数理統計学2022

すべてをまとめたPDF:

- Statistics2022.pdf (https://github.com/genkuroki/Statistics/blob/master/2022/Statistics2022.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/Statistics2022.pdf)]
- Statistics2022handwritten.pdf (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/Statistics2022handwritten.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/Statistics2022handwritten.pdf)]

これはおまけで作っているPDFファイルなので更新は遅れる予定. 以下の個別のPDFファイルの方が常に新しいバージョンになっている.

このファイルのipynb版とpdf版:

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/00-0%20README.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/00-0%20README.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/00-0%20README.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/00-0%20README.pdf)]

記号法については次のメモも参照せよ:

- Memo.md (https://github.com/genkuroki/Statistics/blob/master/2022/Memo.md)
- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/00-1%20Memo.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/00-1%20Memo.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/00-1%20Memo.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/00-1%20Memo.pdf)]

目標

- 確率変数の扱い方
- 大数の法則と中心極限定理
- よく使われる確率分布達に関する知識
- 尤度と推定と要約統計
- 検定と信頼区間を表裏一体のものとして理解すること
- P値函数について理解すること
- **簡単な場合について具体的な数値の計算を行えるようになること(重要!)**

資料

Jupyter (https://jupyter.org/) notebook をオンラインで読むなら nbviewer (https://nbviewer.org/) 経由の閲覧がおすすめです。オフラインでも読めるようにするためには pdf ファイルをダウンロードしてください.

Julia言語 (https://julialang.org/)を使って数値の計算やグラフの作画を行っています。必要最小限の使い方については次の資料を見て下さい:

- 07-2 Julia言語を使った統計学の勉強の仕方 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-2%20How%20to%20use%20Julia%20language%20for%20learning%20statistics.ipynb)

WolframAlpha (https://www.wolframalpha.com/)も併用しています.

明らかな誤りをまだ大量に含んでいるものと思われます. ごめんなさい. 適切に訂正しながら読んでください. 随時訂正更新しています.

01 Bernoulli試行と関連確率分布

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/01%20Bernoulli%20trial%20and%20related%20distributions.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/01%20Bernoulli%20trial%20and%20related%20distributions.ip
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/01%20Bernoulli%20trial%20and%20related%20distributions.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/01%20Bernoulli%20trial%20and%20related%20distributions.pdf)]
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%

[download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%
[download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

定義された確率分布

- **カテゴリカル分布**: $\mathrm{Categorical}(p_1, \dots, p_n)$
- 成功確率 p の **Bernoulli分布**: $\mathrm{Bernoulli}(p)$
- 試行回数 n, 成功確率 p の **Bernoulli試行の分布**: $\mathrm{Bernoulli}(p)^n$
- 試行回数 n, 成功確率 p の **二項分布**: $\mathrm{Binomial}(n, p)$ (n 回のBernoulli試行での成功回数の分布)
- 成功回数 k, 成功確率 p の **負の二項分布**: $\mathrm{NegativeBinomial}(n, p) = \mathrm{NegBin}(n, p)$ (Bernoulli試行をちょうど k 回成功するまで続けたときの失敗の回数の分布)
- 成功確率 p の **幾何分布**: $\mathrm{Geometric}(p) = \mathrm{NegativeBinomial}(1, p)$ (Bernoulli試行を成功するまで続けたときの失敗の回数の分布)
- 0から1のあいだの **一様分布**: $\mathrm{Uniform}(0, 1)$ (コンピュータでの一様乱数函数 `rand()` のモデル化)
- 試行回数 n の **一様乱数生成の繰り返しの分布**: $\mathrm{Uniform}(0, 1)^n$
- 平均 μ, 分散 σ^2 (標準偏差 σ)の **正規分布**: $\mathrm{Normal}(\mu, \sigma)$
- **標準正規分布**: $\mathrm{Normal}() = \mathrm{Normal}(0, 1)$
- 分布 D の **アフィン変換_**: $aD + b$
- 分布 D の **スケール変換**: aD ($a > 0$)
- 分布 D の **平行移動**: $D + b$
- 分布 D の **逆数**: $1/D$

02 Gauss積分, ガンマ函数, ベータ函数

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/02%20Gaussian%20integrals%2C%20Gamma%20and%20Beta%20fu
[nbvewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/02%20Gaussian%20integrals%2C%20Gamma%20and%20Be
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/02%20Gaussian%20integrals%2C%20Gamma%20and%20Beta%20fu
[download (https://github.com/genkuroki/Statistics/raw/master/2022/02%20Gaussian%20integrals%2C%20Gamma%20and%20Beta%20fu
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%
[download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%
[download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

定義された確率分布

- 正規分布 (再): $\mathrm{Normal}(\mu, \sigma) = \mu + \sigma\,\mathrm{Normal}(0, 1)$
- 形状 α, スケール θ の **ガンマ分布**: $\mathrm{Gamma}(\alpha, \theta) = \theta\,\mathrm{Gamma}(\alpha, 1)$ (平均的なイベント発生間隔が θ のときにちょうど α 回イベントが生じるまでにかかる時間の分布)
- 平均 θ の **指数分布**: $\mathrm{Exponential}(\theta) = \mathrm{Gamma}(1, \theta)$ (平均的なイベント発生間隔が θ のときにイベントが発生するまでにかかる時間の分布)
- 中央値 a, スケール b の **Laplace分布**: $\mathrm{Laplace}(a, b)$
- 形状 α, スケール θ の **逆ガンマ分布**: $\mathrm{InverseGamma}(\alpha, \theta) = 1/\mathrm{Gamma}(\alpha, 1/\theta)$
- 自由度 ν の **χ²分布**(カイ二乗分布): $\mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$
- 形状 α, スケール θ の **Weibull分布**: $\mathrm{Weibull}(\alpha, \theta)$
- パラメータ α, β の **ベータ分布**: $\mathrm{Beta}(\alpha, \beta)$ ($\alpha + \beta - 1$ 個の一様乱数の中で α 番目に小さな(= β 番目に大きな)値の分布)
- 自由度 ν の **t分布**: $\mathrm{TDist}(\nu)$ (平均 0 の正規分布で分散の逆数が $\mathrm{Chisq}(\nu)/\nu$ に従う場合, $\mathrm{BetaPrime}(1/2, \nu/2)$ と密接に関係)
- パラメータ α, β の **ベータプライム分布**: $\mathrm{BetaPrime}(\alpha, \beta)$ (確率 t に関するベータ分布 $\mathrm{Beta}(\alpha, \beta)$ をオッズ $u = t/(1-t) = 1/(1-t) - 1$ に関する分布に変換したもの)
- 自由度 ν_1, ν_2 の **F分布**: $\mathrm{FDist}(\nu_1, \nu_2) = (\nu_2/\nu_1)\,\mathrm{BetaPrime}(\nu_1/2, \nu_2/2)$
- **Dirichlet分布**: $\mathrm{Dirichlet}(\alpha_1, \dots, \alpha_n)$ ($n = 2$ の場合がベータ分布と一致)

03 確率分布達の解釈

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/03%20Interpretation%20of%20probability%20distributions.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/03%20Interpretation%20of%20probability%20distributions.ipy
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/03%20Interpretation%20of%20probability%20distributions.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/03%20Interpretation%20of%20probability%20distributions.pdf)]
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88% [download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

定義された確率分布

- **対数正規分布**: $\mathrm{LogNormal}(\mu, \sigma) = \exp(\mathrm{Normal}(\mu, \sigma))$
- 平均 λ の **Poisson分布**(ポアソン分布): $\mathrm{Poisson}(\lambda)$
- **負の二項分布**: $\mathrm{NegativeBinomial}(\alpha, 1/(1+\theta))$ (Poisson分布の平均パラメータ λ がガンマ分布に従う場合)
- a から b のあいだの **一様分布**: $\mathrm{Uniform}(a, b)$
- **ベータ二項分布**: $\mathrm{BetaBinomial}(n, \alpha, \beta)$ (二項分布の成功確率パラメータがベータ分布に従う場合)
- **超幾何分布**: $\mathrm{Hypergeometric}(s, f, n)$ (非復元抽出の分布)
- **ベータ負の二項分布**: $\mathrm{NegativeBetaBinomial}(n, \alpha, \beta)$ (負の二項分布の成功確率パラメータがベータ分布に従う場合)
- 試行回数 n の **Pólyaの壺の確率分布**: $\mathrm{Pólya}(n, \alpha, \beta)$ (Bernoulli試行の分布で成功確率パラメータがベータ分布に従う場合)

04 標本分布について

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/04%20Distribution%20of%20samples.pdf)]
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88% [download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

演習用サンプル

- Anscombeの例(アンスコムの例)
  - csv file (https://github.com/genkuroki/Statistics/blob/master/2022/data/anscombe.csv), raw version (https://raw.githubusercontent.com/genkuroki/Statistics/master/2022/data/anscombe.csv)
  - transposed version (https://github.com/genkuroki/Statistics/blob/master/2022/data/anscombe_transposed.csv), transposed raw version (https://raw.githubusercontent.com/genkuroki/Statistics/master/2022/data/anscombe_transposed.csv)

定義された確率分布

- 分布達 $D_1, \dots, D_n$ の **積**: $D_1 \times \cdots \times D_n$
- 分布 D の **累乗**: D^n
- 分布 D のサイズ n の **標本分布**: D^n

作成された動画

楕円と回帰直線の関係 (https://github.com/genkuroki/Statistics/blob/master/2022/images/ellipse_and_regressionline.gif)

05 大数の法則と中心極限定理

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/05%20Central%20limit%20theorem.pdf)]
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88% [download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

大数の法則の数値例 ($(X_1 + X_2 + \cdots + X_n)/n$ の分布)

Bernoulli試行, 正規分布の標本分布, ガンマ分布 Gamma(2, 3) の標本分布

大数の法則が成立しない場合の数値例

Cachy分布の標本分布, Pólyaの壺＝事前分布がベータ分布のBernoulli試行

ランダムウォークの数値例 $X_1 + X_2 + \cdots + X_n$ の分布

Bernoulli試行, 正規分布の標本分布, ガンマ分布 Gamma}(2, 3) の標本分布, Cachy分布の標本分布, Pólyaの壺＝事前分布がベータ分布のBernoulli試行

中心極限定理の数値例

対称な分布　二項分布, 一様分布の標本分布, ベータ分布 Beta(0.5, 0.5), Bernoulli試行 p = 0.5, Laplace分布の標本分布

非対称な分布　ガンマ分布 Gamma(3, 1) の標本分布, Bernoulli試行 (p = 0.25) の標本分布, Poisson分布 (期待値1) の標本分布, 指数分布 (期待値1) の標本分布, χ²分布 (自由度1) の標本分布, 混合正規分布の標本分布, 対数正規分布の標本分布

正規分布で近似される確率分布の数値例

デルタ法の応用　自由度が大きなχ²分布の逆数, 自由度が大きなχ²分布の対数, 2つのパラメータが両方とも大きなベータ分布

適当なパラメータ領域において正規分布で近似される分布　二項分布, 負の二項分布, Poisson分布, ガンマ分布, χ²分布, 逆ガンマ分布, ベータ分布, t分布, ベータプライム分布, F分布, 超幾何分布, ベータ二項分布, ベータ負の二項分布

06 条件付き確率分布, 尤度, 推定, 記述統計

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estimatic [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estimatic [download (https://github.com/genkuroki/Statistics/raw/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estimatio
- hand-written note (https://github.com/genkuroki/Statistics/blob/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88% [download (https://github.com/genkuroki/Statistics/raw/master/2022/handwritten/%E6%95%B0%E7%90%86%E7%B5%B1%E8%A8%88%E

演習用サンプル

- Datasaurusの例 (データサウルスの例) (http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html)
  - csv file (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_data.csv), raw version (https://raw.githubusercontent.com/genkuroki/Statistics/master/2022/data/Datasaurus_data.csv)
  - x座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X.txt), y座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y.txt)
  - x座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X_with_commas.txt), y座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y_with_commas.txt)

この例に関する詳しい解説が次の場所にある:

- Justin Matejka, George Fitzmaurice. Same Stats, Different Graphs: Generating Datasets with Varied Appearance and Identical Statistics through Simulated Annealing. Honourable Mention, ACM SIGCHI Conference on Human Factors in Computing Systems, 2017 [link (https://www.autodesk.com/research/publications/same-stats-different-graphs)]

関連の情報が datasaurus same stats を検索 (https://www.google.com/search?q=datasaurus+same+stats) すれば得られる.

定義された確率分布

- **2変量正規分布**: $\mathrm{MvNormal}(\mu, \Sigma)$ (**多変量正規分布** でも同じ記号を使う.)
- **2×2の分割表の分布**
  - 4つのPoisson分布の積
  - 四項分布
  - 2つの二項分布の積
  - Fisherの非心超幾何分布

07 例

07-1 例：ベータ函数と二項分布の関係とその応用

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.pdf)]

07-2 Julia言語を使った統計学の勉強の仕方

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/07-2%20How%20to%20use%20Julia%20language%20for%20learning%20statistics.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-2%20How%20to%20use%20Julia%20language%20for%20learning%20statistics.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/07-2%20How%20to%20use%20Julia%20language%20for%20learning%20statistics.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.pdf)]

08 検定と信頼区間(1) 一般論

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/08%20Hypothesis%20testing%20and%20confidence%20interval%20-%20General%20theory.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/08%20Hypothesis%20testing%20and%20confidence%20inter%20General%20theory.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/08%20Hypothesis%20testing%20and%20confidence%20interval%20-%20General%20theory.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/08%20Hypothesis%20testing%20and%20confidence%20interval%20-%20General%20theory.pdf)]

P値函数 の概念が最も重要である.

この一連のノートでは **信頼区間は常に対応するP値函数を使って定義される** ことになる.

関連講義動画

- 京都大学大学院医学研究科 聴講コース
  臨床研究者のための生物統計学「仮説検定とP値の誤解」
  佐藤 俊哉 医学研究科教授
  https://youtu.be/vz9cZnB1d1c (https://youtu.be/vz9cZnB1d1c)

この講義動画は「P値に関するASA声明 (https://www.biometrics.gr.jp/news/all/ASA.pdf)」の考え方の非常に良い解説になっている.

09 検定と信頼区間(2) 比率

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Proportion.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Proportion.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/09%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Proportion.pdf)]

定義された信頼区間

二項検定に付随する4種の信頼区間

- Clopper-Pearsonの信頼区間
- Sterneの信頼区間
- Wilsonの信頼区間
- Waldの信頼区間

作成された動画

n=20の二項分布モデルのP値函数 (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction20.gif)

n=100の二項分布モデルのP値函数 (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction100.gif)

10 検定と信頼区間(3) 平均

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/10%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Mean.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/10%20Hypothesis%20testing%20and%20confidence%20inter%20Mean.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/10%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Mean.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/10%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Mean.pdf)]

定義された信頼区間

平均の信頼区間

- 標準正規分布を使って計算される信頼区間
- t分布を使って計算される信頼区間

後者のみを使用する.

定義された確率分布

- 自由度νのχ分布: $\mathrm{Chi}(\nu) = \sqrt{\mathrm{Chisq}(\nu)}$

作成された動画

n → ∞ でのP値函数と信頼区間の挙動 (https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_mean.gif)

11 検定と信頼区間(4) 比率の違い

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20proportions.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20proportions.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/11%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20proportions.pdf)]

定義された信頼区間

オッズ比の信頼区間

- Wald型の信頼区間
- Pearsonのχ²検定に付随する信頼区間
- Fisher検定に付随する信頼区間
  - Sterne型
  - Clopper-Pearson型

リスク比の信頼区間

- Wald型の信頼区間
- Pearsonのχ²検定に付随する信頼区間

比率の差の信頼区間

- Wald型の信頼区間

定義された確率分布

- Fisherの非心超幾何分布: $\mathrm{FisherNoncentralHypergeometric}(m, n, r, \omega)$

関連講義動画

- 聴講コース 臨床研究者のための生物統計学 (https://ocw.kyoto-u.ac.jp/course/328/)
  - 佐藤俊哉, ランダム化ができないとき, 2018/10/25, 長さ 1:02:31) [https://youtu.be/tUkyjZXU1vc (https://youtu.be/tUkyjZXU1vc)]
  - 佐藤俊哉, 交絡とその調整, 2018/12/20, 長さ 1:00:52) [https://youtu.be/ybdkQFEdCPM (https://youtu.be/ybdkQFEdCPM)]
  - 佐藤俊哉, 回帰モデルと傾向スコア, 2019/02/21, 長さ 1:04:44) [https://youtu.be/cOHN444kBlo (https://youtu.be/cOHN444kBlo)]

「検定と信頼区間(4) 比率の違い」のノートブックの内容をJulia言語で直接実行するためのヒント

「検定と信頼区間(4) 比率の違い」のノートブック (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20interval%%20Two%20proportions.ipynb) の中の

```
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
χ² = (a+b+c+d)*(a*d-b*c)^2/((a+b)*(c+d)*(a+c)*(b+d))
@show χ²
@show pvalue = ccdf(Chisq(1), χ²)
α = 0.05
@show α
confint = find_zeros(0.5, 3.0) do ω
    A, B, C = 1-ω, a+d+ω*(b+c), a*d-ω*b*c
    δ = 2C/(B + √(B^2 - 4A*C))
    χ² = δ^2 * (1/(a-δ) + 1/(b+δ) + 1/(c+δ) + 1/(d-δ))
    ccdf(Chisq(1), χ²) - α
end
@show confint;
```

```
(a, b, c, d) = (49, 965, 26, 854)
χ² = 4.368235947209686
pvalue = ccdf(Chisq(1), χ²) = 0.03661487827607113
α = 0.05
confint = [1.031778110800563, 2.6957054358850328]
```

の部分を直接 Julia (download (https://julialang.org/downloads/))で実行するためには以下のようにすればよい.

注意するべきポイントは `using Distributions, Roots` を忘れないことである.

しかし, どのくらい前もって `using` しておけばよいのか分かり難いので, 上のノートブックの上の方にある `using` 達をすべて前持って実行しておくという手もある. `using` には少し時間が取られるので, そのようにしたJuliaの窓をそのまま閉じずに放置しておき, 超高級電卓として利用し続けると便利だと思われる.

`χ²` などの入力の仕方は, プロンプト `julia>` で `?` と入力して, ヘルプモードのプロンプト `help?>` を表示させて, `χ²` を貼り付けてエンターキーを押せばよい. その結果

```
"χ²" can be typed by \chi<tab>\^2<tab>
```

と表示されている. すなわち, `\chi` と入力してタブキーを押し, さらに `\^2` と入力してタブキーを押せば `χ²` と入力できる.

12 検定と信頼区間(5) 平均の差

- [Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20means.ipynb)](https://github.com/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20means.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb)]
- [pdf (https://github.com/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20means.pdf)](https://github.com/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20means.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/12%20Hypothesis%20testing%20and%20confidence%20interval%20-%20Two%20means.pdf)]

定義された信頼区間

2群の平均の差の信頼区間

- Welchのt検定に付随する信頼区間

Studentのt検定は使用しない.

作成された動画

[データサイズを大きくしたときのP値函数と信頼区間の挙動 (https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_diffmeans.gif)](https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_diffmeans.gif)

13 誤用を避けるための注意

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/13%20How%20to%20avoid%20misuse.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/13%20How%20to%20avoid%20misuse.ipynb)]
- pdf () [download ()]

P値に関する誤用を避けるために最初に読むべき基本文献は

- 統計的有意性とP値に関するASA声明 (https://www.biometrics.gr.jp/news/all/ASA.pdf)

である. ただし, 以下の部分の「データ」は「統計モデル内でランダムに生成された仮想的なデータ」の意味であると解釈する必要がある. (「現実で得た観察データ」であると解釈してはいけない.)

> 2. P 値とは?
> おおざっぱにいうと、P 値とは特定の統計モ デルのもとで、データの統計的要約（たとえば、 2 グループ比較での標本平均の差）が観察され た値と等しいか、それよりも極端な値をとる確率 である。

一連の講義では, 通常の統計学入門の教科書とは違って, 「統計モデル」という用語を多用した. このP値の定義の具体例については他のノートで詳しく説明した.

14 回帰

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/14%20Regression.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/14%20Regression.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/14%20Regression.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/14%20Regression.pdf)]

定義された信頼推定

線形回帰の場合:

- 回帰函数の値の信頼区間
- 回帰函数＋残差の予測区間

線形回帰の場合:

- $t_* = \beta_0 + \beta_1 x_*$ の信頼区間
- $p_* = \mathrm{logistic}(\beta_0 + \beta_1 x_*)$ の信頼区間
- $x_* = 0$ の場合から β_0 の信頼区間が得られる.
- β_1 の信頼区間

定義された確率分布

- 多変量正規分布: $\mathrm{MvNormal}(\mu, \Sigma)$
- ロジスティックモデル: $\mathrm{LogisticModel}(x, \beta)$

作成された動画

線形回帰の信頼区間に対応するP値函数 (https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_linreg_confint_pvalfunc_3d.gif) (x軸は-2から2まで)

ロジスティック回帰のβ_1の信頼区間に対応するP値函数 (https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_logisticreg_beta1_confint_pvalfunc.gif)

15 まとめ

- Jupyter notebook (https://github.com/genkuroki/Statistics/blob/master/2022/15%20Summary.ipynb) [nbviewer (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/15%20Summary.ipynb)]
- pdf (https://github.com/genkuroki/Statistics/blob/master/2022/15%20Summary.pdf) [download (https://github.com/genkuroki/Statistics/raw/master/2022/15%20Summary.pdf)]

1. 二項分布モデルでのClopper-Pearsonの信頼区間
2. P値函数について
3. Welchの t 検定について
4. 数学的な補足: 大数の法則と中心極限定理について

参考資料

以上で解説できなかったことに関するノート

- Kullback-Leibler情報量とSanovの定理 (2016-2018) [pdf (https://genkuroki.github.io/documents/20160616KullbackLeibler.pdf)]
- ガンマ分布の中心極限定理とStirlingの公式 (2016-2018) [pdf (https://genkuroki.github.io/documents/20160501StirlingFormula.pdf)]
- 一般化されたLaplaceの方法 (2016) [pdf (https://genkuroki.github.io/documents/20161014GeneralizedLaplace.pdf)]

- 最尤法とカイ二乗検定の基礎 (2017-2020) [pdf (https://genkuroki.github.io/documents/IntroMLE.pdf)]
- 確率論入門 (2017) [pdf (https://genkuroki.github.io/documents/IntroProbability.pdf)]
- ベイズ統計入門 (手書きのノート, 2019) [pdf (https://genkuroki.github.io/documents/2019-09-03_BayesianStatistics.pdf)]
- ベイズ統計の枠組みと解釈について (2019) [Jupyter notebook (https://nbviewer.org/github/genkuroki/Statistics/blob/master/Introduction%20to%20Bayesian%20Statistics.ipynb)]
- Kullback-Leibler情報量と記述統計 (2019-2020) [Jupyter notebook (https://nbviewer.org/github/genkuroki/Statistics/blob/master/KL%20information%20and%20descriptive%20statistics.ipynb)]

記号法に関するメモ

$\mathbb{Z}$ は整数 $0, \pm 1, \pm 2, \ldots$ 全体の集合.

$\mathbb{R}$ は実数全体の集合.

$\mathbb{R}_{>0}$ は正の実数全体の集合.

$[a, b]$ は a 以上 b 以下の実数全体で構成される閉区間.

(a, b) は a より大きく b 未満の実数全体で構成される開区間.

Γ はガンマ γ の大文字.

B はベータ β の大文字になる場合がある.

$\Gamma(\alpha)$ はガンマ函数で $B(\alpha, \beta)$ はベータ函数:

$$\Gamma(\alpha) = \int_0^\infty e^{-x} x^{\alpha-1}\, dx, \quad B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\, dt.$$

$\leq$ と $\leqq$ は同じ意味.

$\approx$ は近似的に等しいを意味する. その正確な意味は文脈に依存する.

$p(x|\theta)$ はパラメータ θ を持つ x に関する確率密度函数.

$P(x|\theta)$ はパラメータ θ を持つ x に関する確率質量函数.

$\mathrm{Binomial}(n, p)$ は二項分布を表す.

$\mathrm{Normal}(\mu, \sigma)$ は平均 μ と標準偏差 σ を持つ正規分布を表す.

$\mathrm{Gamma}(\alpha, \theta)$ はガンマ分布を表す.

$\mathrm{Chisq}(\nu)$ は自由度 ν の χ^2 分布を表す.

$\mathrm{Beta}(a, b)$ はベータ分布を表す.

$\mathrm{TDist}(\nu)$ は自由度 ν の t 分布を表す.

$X \sim D$ は確率変数 X が確率分布 D に従うことを意味する.

$X \sim D$, approximately は確率分布 X が確率分布 D に近似的に従うことを意味する. 近似的に従うことの正確な意味は文脈依存.

$(X_1, \ldots, X_n) \sim D^n$ は確率変数達 $X_1, \ldots, X_n$ の組が確率分布 D の標本分布に従うことを意味する. これは, $X_1, \ldots, X_n$ が独立同分布な確率変数達でその各々が確率分布 D に従うことと同値である.

$E[f(X)]$ は確率変数 X の函数 $f(X)$ (これも確率変数)の期待値.

$1_{a<X<b}(X)$ は $a < X < b$ という条件が成立するときに 1 になり, それ以外のときに 0 になる函数.

$P(X \leq x)$ は確率変数 X の値が x 以下になる確率.

$P(X = x, Y = y)$ は確率変数 X が x に等しくてかつ確率変数 Y が y に等しい確率.

$P(X = x|Y = y)$ は確率変数 Y が y に等しいという条件における確率変数 X が x に等しい条件付き確率. $P(X = y|Y = y)$ はパラメータ y を持つ x の確率質量函数になっている.

$p(x, y)$ に対する $p(x|y)$ は y で条件付けらえた x に関する条件付き確率分布の密度函数. $p(x|y)$ はパラメータ y を持つ x の確率密度函数になっている.

$E[f(X, Y)|Y = y]$ は $Y = y$ という条件のもとでの確率変数 $f(X, Y)$ の条件付き期待値. これは y の函数とみなされ, 確率変数 $E[f(X, Y)|Y]$ を y を $E[f(X, Y)|Y = y]$ に対応させる函数として定義すると, $E[E[f(X, Y)|Y]] = E[f(X, Y)]$ が成立している.

確率分布 D の累積分布函数(分布 D に従う確率変数の値が x 以下になる確率)を $\mathrm{cdf}(D, x)$ と表す. D が確率質量函数 $P(k)$ を持つとき,

$$\mathrm{cdf}(D, x) = \sum_{k \leq x} P(k)$$

であり, D が確率密度函数 $p(x)$ を持つとき,

$$\mathrm{cdf}(D, x) = \int_{-\infty}^{x} p(t)\, dt.$$

確率分布 D の分位点函数(累積分布函数 $p = \mathrm{cdf}(D, x)$ の逆函数または逆函数がない場合には適切に定義された逆函数のような函数)を $x = \mathrm{quantile}(D, p)$ と表す.

パラメータのまたはパラメータで表された数値としてのオッズ比(odds ratio)は OR のようにハット無しで表す.

確率変数のまたはデータの数値で表されたオッズ比は $\widehat{\mathrm{OR}}$ のようにハットを付けて表す.

基本的にハット付きの記号はハット無しの値の推定量という意味になる.

ハットの有無は標準誤差(standard error)の場合も同様で, モデルのパラメータで表された標準誤差は SE と書くが, 確率変数またはデータの数値で表された数値としての標準誤差の推定値は $\widehat{\mathrm{SE}}$ と表す.

Bernoulli試行と関連確率分布

- 黒木玄

- 2022-04-09〜2022-05-31

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

1 WolframAlphaの使い方
2 一様乱数の生成の繰り返し
3 Bernoulli試行
4 離散分布
4.1 離散分布と確率質量函数
4.2 離散確率変数
4.3 離散確率変数の期待値
4.4 離散確率変数の期待値を取る操作の基本性質
4.5 確率変数の一般的な定義
4.6 離散確率変数に関する確率
4.7 離散確率変数の分散と標準偏差
4.8 問題: 期待値と分散の特徴付け
4.9 問題: 平均二乗誤差のバイアス・バリアンス分解
4.10 カテゴリカル分布の定義
4.11 問題: ルーレットの例の賞金の期待値と分散
4.12 Bernoulli分布の定義
4.13 問題: Bernoulli分布の期待値と分散
4.14 オッズの定義と例
4.15 オッズとロジットおよびロジスティック変換
4.16 Bernoulli試行の確率分布
4.17 問題: n=3, p=0.3 の場合のBernoulli試行の確率分布
4.18 Bernoulli試行から応用上基本的かつ重要な確率分布が大量に得られること
4.19 二項定理
4.20 二項分布の定義
4.21 問題: 二項分布の期待値と分散
4.22 問題: 当たりが30%の確率で出るくじを100回引いた場合
4.23 幾何分布の定義
4.24 幾何分布の期待値と分散
4.25 問題: 当たりが出る確率が1%のガチャを当たりが出るまで回す場合
4.26 負の二項展開
4.27 負の二項分布の定義
4.28 負の二項分布の期待値と分散
4.29 問題: 当たりが出る確率が5%のガチャを当たりが5回出るまで回す場合
4.30 多項分布の定義
5 連続分布など
5.1 連続分布と確率密度函数
5.2 一様分布の定義
5.3 正規分布の定義
5.4 連続確率変数
5.5 連続確率変数に関する確率とその期待値
5.6 確率変数の確率密度函数
5.7 確率変数の確率質量函数
5.8 連続分布における確率変数の期待値を取る操作の基本性質

5.9 問題: Markovの不等式とChebyshevの不等式
5.10 問題: Jensenの不等式
5.11 問題: 相加相乗平均の不等式
5.12 問題: Kullback-Leibler情報量とGibbsの情報不等式
5.13 連続分布における確率変数の分散
5.14 問題: 正規分布モデルによる他の分布の最良近似
5.15 確率変数の一次函数による変換(アフィン変換)の期待値と分散
5.16 分布のアフィン変換, スケール変換, 平行移動, 逆数の定義
5.17 確率変数の累積分布函数
5.18 分位点函数 (quantile function)
5.19 Bernoulli分布の累積分布函数
5.20 カテゴリカル分布の累積分布函数
5.21 二項分布の累積分布函数のグラフ
5.22 負の二項分布の累積分布函数のグラフ
5.23 一様分布の累積分布函数
5.24 標準正規分布の累積分布函数と分位点函数
5.25 問題: 標準正規分布の四分位数
5.26 問題: 連続分布の累積分布函数は一様分布に従う.
5.27 分位点函数への一様乱数の代入で元の分布の乱数を作れること
5.28 一様分布の期待値と分散
5.29 標準正規分布の期待値と分散
5.30 一様乱数の生成の繰り返しの分布

1 WolframAlphaの使い方

WolframAlpha (https://www.wolframalpha.com/)を利用すれば以下のような計算が可能である. 例えば,

$$(10000-2020)^2\cdot 0.1+(3000-2020)^2\cdot 0.2+(1000-2020)^2\cdot 0.3+(300-2020)^2\cdot 0.4$$

は次のように入力すれば計算できる:

- (10000 - 2020)^2 0.1 + (3000 - 2020)^2 0.2 + (1000 - 2020)^2 0.3 + (300 - 2020)^2 0.4 (https://www.wolframalpha.com/input?i=%2810000+-+2020%29%5E2+0.1+%2B+%283000+-+2020%29%5E2+0.2+%2B+%281000+-+2020%29%5E2+0.3+%2B+%28300+-+2020%29%5E2+0.4)

二項係数を含む式

$$\sum_{k=5}^{300}\binom{300}{k}0.01^k(1-0.01)^{300-k}$$

は次のように入力すれば計算できる:

- sum_{k=5}^{300} binom(300, k) 0.01^k (1 - 0.01)^(300-k) (https://www.wolframalpha.com/input?i=sum_%7Bk%3D5%7D%5E%7B300%7D+binom%28300%2C+k%29+0.01%5Ek+%281+-+0.01%29%5E%28300-k%29)

これは当たりが出る確率が1%のガチャを300回まわしたときに幸運にも当たりが5回以上出る確率になっている.

積分

$$2\int_{1.96}^{\infty}\frac{\exp(-x^2/2)}{\sqrt{2\pi}}\,dx$$

は次のように入力すれば計算できる:

- 2 int_{1.96}^{infty} exp(-x^2/2)/sqrt(2π) dx (https://www.wolframalpha.com/input?i=2+int_%7B1.96%7D%5E%7Binfty%7D+exp%28-x%5E2%2F2%29%2Fsqrt%282%CF%80%29+dx)

これはほぼ $5\% = 0.05$ になる.

統計学での練習問題を解くためにはこの手の計算を何らかの手段でできるようになっておく必要がある.

In [1]:
```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:
```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

2 一様乱数の生成の繰り返し

確率変数 の理解に向けて, コンピュータでの `rand()` 函数について説明する.

0以上1未満の区間上の一様分布の(擬似)乱数(以下ではこれを **一様乱数** と短く呼ぶことにする)を10個生成すると以下のようになる.

In [3]:
```julia
[rand() for _ in 1:10]
```

Out[3]:
```
10-element Vector{Float64}:
 0.5105030210947423
 0.13229885279256504
 0.05873542170272006
 0.742812017444488
 0.8417141593154808
 0.44238748406950934
 0.5002487759410339
 0.7049642022880609
 0.6687158029467306
 0.7672724745322447
```

コンピュータでの `rand()` は **値がランダムに決まる量という意味での確率変数** のわかりやすい例になっていると思える.

上と同じ結果を与えるコードは次のようにもっとシンプルに書ける.

In [4]:
```
rand(10)
```

Out[4]:
```
10-element Vector{Float64}:
 0.7702439301833676
 0.6263555234233755
 0.4983977148383013
 0.6011722501239075
 0.18013689329541094
 0.37632266071643605
 0.13414398433355734
 0.9652043335626921
 0.7352525630207614
 0.23381276022709152
```

3 Bernoulli試行

確率 p で 1 を, 確率 $1-p$ で 0 をランダムに毎回独立に生成することの繰り返しを **Bernoulli試行** (ベルヌーイ試行, ベルヌイ試行)と呼ぶ.

Bernoulli試行は上で説明した `rand()` の繰り返しに似ており, 乱数の値が 1 と 0 しかない場合の乱数列生成だとみなされる.

In [5]:
```
@show rand(Bernoulli(0.3), 20);
```

```
rand(Bernoulli(0.3), 20) = Bool[0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1]
```

これは 1 が生成される確率が $p=0.3$ で試行回数が $n=20$ のBernoulli試行の例であり, この場合には生成された 1 の個数の方が 0 の個数より少なくなっている.

乱数の出目が大幅に偏っていれば 1 の個数の方が多くなることもありえる. 1 が生成される確率が $p=0.3$ で試行回数が $n=20$ のBernoulli試行で生成された 1 の個数が 10 より大きくなる確率は約 1.7% になる.

In [6]:
```
ccdf(Binomial(20, 0.3), 10)
```

Out[6]:
```
0.017144816431258425
```

0以上1未満の乱数生成を繰り返して, 乱数の値が p 未満(以下でもよい)ならば 1 を生成し, p 以上(より大きいでもよい)のとき 0 を生成することにしてもBernoulli試行が得られる.

In [7]:
```
T = rand(20)
for i in eachindex(T) @printf "%.16f\n" T[i] end
```

```
0.7086835170539888
0.3129878077458778
0.4432299124211534
0.6380100590067396
0.7457553759007464
0.0042479988608554
0.4282692167397828
0.2591080281694401
0.6056891396222264
0.4718831513039459
0.2359750577106341
0.3658417371594386
0.4169159761488047
0.2234702295437706
0.2267949185649013
0.1054052960574101
0.5401955636162634
0.3620455865151407
0.9332297268604005
0.1009962782059489
```

以下の計算では T の成分が $p=0.3$ 未満のときに 1 に, それ以外のときに 0 に変換している. これによって, 一様乱数列からBernoulli試行が得られる.

In [8]:
```
X = [t < 0.3 ? 1 : 0 for t in T]
for i in eachindex(T) @printf "%.16f → %d\n" T[i] X[i] end
```

```
0.7086835170539888 → 0
0.3129878077458778 → 0
0.4432299124211534 → 0
0.6380100590067396 → 0
0.7457553759007464 → 0
0.0042479988608554 → 1
0.4282692167397828 → 0
0.2591080281694401 → 1
0.6056891396222264 → 0
0.4718831513039459 → 0
0.2359750577106341 → 1
0.3658417371594386 → 0
0.4169159761488047 → 0
0.2234702295437706 → 1
0.2267949185649013 → 1
0.1054052960574101 → 1
0.5401955636162634 → 0
0.3620455865151407 → 0
0.9332297268604005 → 0
0.1009962782059489 → 1
```

Bernoulli試行に限らず, 何らかの確率分布に従う乱数の生成の独立試行は, 0以上1未満の一様乱数の生成を経由して作ることができる. その意味でコンピュータでのプログラミングでおなじみの `rand()` 函数は基本的である.

毎回同確率で当たりが出るくじを繰り返し引くことは, 当たりのときに 1, 外れのときに0とすれば, Bernoulli試行モデルで記述できることの例になっている.

In [9]:
```
AH = [x == 1 ? "当たり" : "外れ" for x in X]
for i in eachindex(T) @printf "%.16f → %d → %s\n" T[i] X[i] AH[i] end
```

```
0.7086835170539888 → 0 → 外れ
0.3129878077458778 → 0 → 外れ
0.4432299124211534 → 0 → 外れ
0.6380100590067396 → 0 → 外れ
0.7457553759007464 → 0 → 外れ
0.0042479988608554 → 1 → 当たり
0.4282692167397828 → 0 → 外れ
0.2591080281694401 → 1 → 当たり
0.6056891396222264 → 0 → 外れ
0.4718831513039459 → 0 → 外れ
0.2359750577106341 → 1 → 当たり
0.3658417371594386 → 0 → 外れ
0.4169159761488047 → 0 → 外れ
0.2234702295437706 → 1 → 当たり
0.2267949185649013 → 1 → 当たり
0.1054052960574101 → 1 → 当たり
0.5401955636162634 → 0 → 外れ
0.3620455865151407 → 0 → 外れ
0.9332297268604005 → 0 → 外れ
0.1009962782059489 → 1 → 当たり
```

ビジネス分野では, 「広告を見てくれる」「リンクをクリックしてくれる」「商品を購入してくれる」などが「当たり」の例として扱われたりする.

現実の統計分析でBernoulli試行モデルを使う場合には, 「当たり」がよいことばかりとは限らない. たとえば, 「病気になる」「死ぬ」「事故が起きる」「機器が壊れる」などもBernoulli試行モデルで扱われる場合がある.

現実への応用において, Bernoulli試行モデルはある処置によって生じる「当たり」が出る確率の変化を推定するために使われることが多い.

例えば, 「薬を飲むと病気になる確率が減るか？」とか「ウェブサイトの構成を変えると商品を買ってもらえる確率が上がるか？」などが問題になる.

しかし, Bernoulli試行モデルにおける「すべての場合に同じ確率で独立に当たりまたは外れが生成される」という条件は相当に強い仮定になっており, 多くの場合には非現実的であり, 現実への応用では様々な工夫が必要になる.

統計分析のために用いられる **統計モデルは現実にはぴったり一致していない** と考えた方がよい. 以下ではこの点について誤解を招きかねない単純化を行う場合が多いので注意して欲しい. 多くの入門的教科書では **統計モデルは現実にはぴったり一致していない** という実践的に非常に重要なことが強調されていないことが多いので注意が必要である.

4 離散分布

4.1 離散分布と確率質量函数

以下, a_i 達は互いに異なると仮定する.

有限集合 $\{a_1, a_2, \ldots, a_r\}$ もしくは可算集合 $\{a_1, a_2, a_3, \ldots\}$ 上の **離散確率分布** (discrete probability distribution)もしくは **離散分布** (discrete distribution)とは, 0以上の有限個の実数達 $p_1, p_2, \ldots, p_r$ もしくは0以上の可算個の実数達 $p_1, p_2, p_3, \ldots$ で総和が 1 になるものであると定める:

$$p_i \geq 0, \quad \sum_i p_i = 1.$$

確率分布を単に分布と呼ぶことがよくある.

そのとき, a_i を p_i に対応させる函数 $P(a)$ をその離散分布の **確率質量函数** (probability mass function, pmf)と呼ぶ:

$$P(a_1) = p_1,\ P(a_2) = p_3,\ P(a_3) = p_3,\ \ldots$$

このとき,

$$P(a_i) \geq 0, \quad \sum_i P(a_i) = 1.$$

離散分布を与えることは確率質量函数を与えることに等しい. 以下では主に確率質量函数を与えることによって離散分布を構成する.

直観的には各 $P(a_i) = p_i$ はこの離散分布に従う乱数において値 a_i が確率 $P(a_i) = p_i$ で生成されることを意味すると考える.

しかし, 実際の数学的取り扱いでは乱数の「ランダム性」には触れずに単なる「確率の値の表」として

x	a_1	a_2	a_3	$\cdots$
$P(x)$	$P(a_1)$	$P(a_2)$	$P(a_3)$	$\cdots$

を扱うことになる.

以下で使いまわすルーレットの例: 例えば, $1, 2, 3, 4$ の値が確率それぞれ $0.1, 0.2, 0.3, 0.4$ で出るルーレットは,

x	1	2	3	4
$P(x)$	0.1	0.2	0.3	0.4

という確率の値の表で定まる離散分布で自然にモデル化される.

4.2 離散確率変数

離散確率変数 (discrete random variable)とは, 確率質量函数 $P(x)$ が与えられた有限集合 $\{a_1, a_2, \ldots, a_r\}$ または可算集合 $\{a_1, a_2, a_3, \ldots\}$ 上の **実数値函数** のことであると定める. (後で, 有限集合や可算集合上の函数でなくても, 取り得る値全体の集合が有限集合や離散的な可算集合になるような確率変数も離散確率変数と呼ぶことになるが, 現時点ではそういう細かいことを気にする必要はない.)

実際には, 複素数値の確率変数やベクトルや行列に値を持つ確率変数も考えることがある. そのような場合には, **複素数値確率変数**, **ベクトル値確率変数**, **行列値確率変数** などと呼ぶことにする.

函数のことを「変数」と呼ぶことに違和感を覚える人がいるかもしれないが, 直観的には確率変数を「ランダムに値が決まる変数」とみなしたい場合も多いので「変数」という呼び方もそう悪くない.

以下では, 有限集合 $\{a_1, a_2, \ldots, a_r\}$ または可算集合 $\{a_1, a_2, a_3, \ldots\}$ は $\mathbb{R}$ の部分集合である場合について考える.

確率質量函数 $P(x)$ が与えられていると仮定する. このとき, $x = a_i$ に実数の $x = a_i$ 自身を対応させる函数として定義される確率変数を X と書き, **確率質量函数 P(x) を持つ離散確率変数** と呼ぶ:

$$X : x = a_i \mapsto x = a_i.$$

このとき,

$$X \sim P(x)$$

という書き方をすることがある. さらに, その確率質量函数で与えられる離散分布に Dist という名前が付けられているとき, X は **分布** Dist **に従う確率変数** であるともいい,

$$X \sim \mathrm{Dist}$$

という書き方をすることがある. 「分布〜に従う確率変数」は "random variable following distribution 〜" の翻訳になっている. $X \sim \mathrm{Dist}$ のような表記は確率分布による統計モデルをコンピュータで実装するために使われている確率プログラミング言語での採用されている記号法でもある.

確率変数は大文字で表すことが多く, 大文字で書かれた確率変数に対応する通常の変数を対応する小文字で表すことが多い. この対応は絶対ではないが, ある程度守っておくと, 記号を読み易くなる場合がある.

例えば, $1, 2, 3, 4$ の値が確率それぞれ $0.1, 0.2, 0.3, 0.4$ で出るルーレットのモデル化の離散分布において, 集合 $\{1, 2, 3, 4\}$ 上の任意の実数値函数は離散確率変数になる.

$x = 1, 2, 3, 4$ のそれぞれにその値自身 $x \in \mathbb{R}$ を対応させる函数が定める確率変数を X と書くことにする. この X は確率質量函数 $P(x) = 0.1x$ $(x = 1, 2, 3, 4)$ を持つ離散確率変数である.

直観的には, この確率変数 X の値はランダムに決まり, その値は確率 $0.1, 0.2, 0.3, 0.4$ でそれぞれ $1, 2, 3, 4$ になると考える. (これはあくまでも直観的な解釈であり, 確率変数は数学的には単なる函数に過ぎない.)

$x \in \{1, 2, 3, 4\}$ を実数 $f(x) \in \mathbb{R}$ に対応させる函数も離散確率変数になる. その確率変数を $f(X)$ と表す.

直観的には, 確率変数 $f(X)$ の値はランダムに決まり, その値は確率 $0.1, 0.2, 0.3, 0.4$ でそれぞれ $f(1), f(2), f(3), f(4)$ になると考える.

一般に確率変数 X の函数 $f(X)$ も確率変数になると考えてよい. (変数の値がランダムに決まるとき, その変数の函数の値もランダムに決まる. 値がランダムに決まると考えたくなるものを確率変数と呼ぶ習慣になっている.)

4.3 離散確率変数の期待値

確率変数の最も重要な性質はその期待値が定義されることである. **確率変数の定義を期待値が定義されている変数(のようなもの)のことだと思ってもよい.** それくらい期待値の概念は確率変数の理解において基本的である.

確率質量函数 $P(x)$ が定められた有限集合 $\{a_1, a_2, \ldots, a_r\}$ または可算集合 $\{a_1, a_2, a_3, \ldots\}$ 上の函数 $f(x)$ が定める離散確率変数を $f(X)$ と書くのであった. このとき, 離散確率変数 $f(X)$ の **期待値**(expectation value) $E[f(X)]$ が次のように定義される:

$$E[f(X)] = \sum_i f(a_i)P(a_i).$$

この和が無限和になる場合には絶対値の期待値 $E[|f(X)|] = \sum_i |f(a_i)|P(a_i)$ が有限の値に収束するとき(すなわち絶対収束するとき)にのみ, 期待値 $E[f(X)] = \sum_i f(a_i)P(a_i)$ が定義されていると考える. (これ以後, こういう細かい話は略す.)

X の従う分布の名前が Dist のとき $E[X]$ を **分布** Dist **の期待値** と呼ぶ.

確率変数の期待値を **平均** または **平均値** と呼ぶこともあるが, あとで定義する標本の平均値とは区別する必要がある. 混乱しないためには常に「確率変数の平均(期待値)」「標本の平均」のように「〜の平均」という言い方をすればよいだろう.

例えば, 上のルーレットの例では,

$$\begin{aligned} E[X] &= 1 \cdot 0.1 + 2 \cdot 0.2 + 3 \cdot 0.3 + 4 \cdot 0.4 = 3, \\ E[f(X)] &= f(1) \cdot 0.1 + f(2) \cdot 0.2 + f(3) \cdot 0.3 + f(4) \cdot 0.4 \end{aligned}$$

上のルーレットの例で, $1, 2, 3, 4$ の値が出たときに得られる賞金の額がそれぞれ10000円, 3000円, 1000円, 300円のとき, 賞金は次の確率変数 $g(X)$ でモデル化される:

$$g(1) = 10000, \quad g(2) = 3000, \quad g(3) = 1000, \quad g(4) = 300.$$

このとき, 賞金 $g(X)$ の期待値は次のように2020円だと計算される:

$$\begin{aligned} E[g(X)] &= g(1)P(1) + g(2)P(2) + g(3)P(3) + g(4)P(4) \\ &= 10000 \cdot 0.1 + 3000 \cdot 0.2 + 1000 \cdot 0.3 + 300 \cdot 0.4 \\ &= 1000 + 600 + 300 + 120 = 2020. \end{aligned}$$

4.4 離散確率変数の期待値を取る操作の基本性質

確率変数の期待値を取る操作(**期待値汎函数**) $f(X) \mapsto E[f(X)]$ は以下の性質を満たしている.

- 線形性: $E[\alpha f(X) + \beta g(X)] = \alpha E[f(X)] + \beta E[g(X)]$.
- 単調性: 任意の $x = a_i$ について $f(x) \leq g(x)$ が成立しているならば, $E[f(X)] \leq E[g(X)]$.
- 規格化条件: $E[\alpha] = \alpha$. (定数 α の期待値は α. 特に $E[1] = 1$).

証明: 確率変数 $f(X), g(X)$ と定数 α, β について,

$$\begin{aligned} E[\alpha f(X) + \beta g(X)] &= \sum_i (\alpha f(a_i) + \beta g(a_i)) P(a_i) \\ &= \alpha \sum_i f(a_i) P(a_i) + \beta \sum_i g(a_i) P(a_i) = \alpha E[f(X)] + \beta E[g(X)] \end{aligned}$$

任意の $x = a_i$ について $f(x) \leq g(x)$ が成立していると仮定すると, $P(a_i) \geq 0$ なので,

$$E[f(X)] = \sum_i f(a_i) P(a_i) \leq \sum_i g(a_i) P(a_i) = E[g(X)].$$

定数 α を定数確率変数とみなすと, $\sum_i P(a_i) = 1$ なので,

$$E[\alpha] = \sum_i \alpha P(a_i) = \alpha \sum_i P(a_i) = \alpha.$$

証明終.

重要: 確率変数に関する多くの計算は上で説明した期待値を取る操作の基本性質のみを使って実行可能なことが多い. 統計学がらみの理論的計算に習熟したいならば, $E[\ ]$ を使った計算の習熟を最初の目標にして勉強するとよいだろう. 以下で説明するように確率変数に関する確率も $E[\ ]$ を使って計算可能である.

4.5 確率変数の一般的な定義

ここで離散とは限らない一般の確率変数を少々大雑把なスタイルで定義しておこう.

変数 X が確率変数であるとは, X の函数 $f(X)$ の期待値 $E[f(X)]$ を取る操作が定義されていて, 期待値を取る操作が前節の基本性質を満たしていることだと定める.

ただし, $E[|f(X)|] = \infty$ になる場合には $f(X)$ の期待値はうまく定義されないと考える.

確率変数には他にも様々な解釈(定義)の仕方があるが, 統計学で必要な様々な理論的な計算をするときには確率変数とは「その函数の期待値が定義されている変数」のことであると了解しておくと, 様々な議論を楽に理解できることが多い.

4.6 離散確率変数に関する確率

$f(X)$ は確率質量函数 $P(x)$ が定められた有限集合 $\{a_1, a_2, \ldots, a_r\}$ または可算集合 $\{a_1, a_2, a_3, \ldots\}$ 上の確率変数であるとする.

このとき, 例えば, $c \leq f(X) \leq d$ となる確率 $P(c \leq f(X) \leq d)$ を次のように定める:

$$P(c \leq f(X) \leq d) = \sum_{c \leq f(a_i) \leq d} P(a_i).$$

確率変数に関する $c \leq f(X) \leq d$ 以外の条件(例えば $f(X) \geq c$ や $f(X) < d$ など)の確率も同様に定める.

このような確率は, 確率変数 $1_{c \leq f(X) \leq d}(X)$ を, 条件 $c \leq f(x) \leq d$ が成立するときに 1 になり, 成立していないとき 0 になる函数として,

$$1_{c \leq f(X) \leq d}(x) = \begin{cases} 1 & (c \leq f(x) \leq d) \\ 0 & (\text{otherwise}) \end{cases}$$

と定めると,

$$P(c \leq f(X) \leq d) = \sum_{c \leq f(a_i) \leq d} P(a_i) = \sum_i 1_{c \leq f(X) \leq d}(a_i) P(a_i) = E[1_{c \leq f(X) \leq d}(X)]$$

と期待値で書けるので, 期待値の概念は確率の概念を含んでいるといえる.

例えば, 上のルーレットの例において, 賞金が500円以上5000円以下になる確率は $P(500 \leq g(X) \leq 5000)$ と書け,

$$P(500 \leq g(X) \leq 5000) = 0.2 + 0.3 = 0.5$$

となる. さらに,

$$1_{500 \leq g(x) \leq 5000}(x) = \begin{cases} 1 & (x = 2, 3) \\ 0 & (\text{otherwise}) \end{cases}$$

となるので,

$$E[1_{500\leq g(X)\leq 5000}(X)] = \sum_{i=1}^{4} 1_{500\leq g(X)\leq 5000}(i)P(i) = 0\cdot 0.1 + 1\cdot 0.2 + 1\cdot 0.3 + 0\cdot 0.4 = 0.5$$

となって, 上の結果と一致する.

4.7 離散確率変数の分散と標準偏差

離散確率変数 $Y = f(X)$ の期待値を

$$\mu_Y = E[Y] = E[f(X)] = \sum_i f(a_i)P(a_i)$$

と書くとき, $Y = f(X)$ の **分散** (variance) $\mathrm{var}(Y) = \mathrm{var}(f(X))$ が次のように定義される:

$$\sigma_Y^2 = \mathrm{var}(Y) = E\left[(Y-\mu_Y)^2\right] = \sum_i (f(a_i)-\mu_Y)^2\, P(a_i).$$

分散の平方根を **標準偏差** (standard deviation) と呼ぶ. 確率変数 Y の標準偏差を次のように表すことにする:

$$\sigma_Y = \mathrm{std}(Y) = \sqrt{\mathrm{var}(Y)}.$$

$a > 0$ のとき, Y を a 倍するとその分散は a^2 倍になるが, 標準偏差ならば Y と同じく a 倍になる. こういう性質を持っているので Y と直接比較する場合には Y の分散ではなく標準偏差を使った方が便利である.

X の従う分布の名前が Dist のとき $\mathrm{var}(X)$ を **分布** Dist **の分散** と呼ぶ. (標準偏差についも同様とする.)

確率変数の分散や標準偏差は確率変数が従う分布の **広がりの大きさ**, **確率的揺らぎの大きさ** の指標の1つになっている. そのような指標の中で **分散と標準偏差は期待値と相性がよい**.

注意: 確率変数の期待値(平均)以外の分布の中心を表す値として, 中央値の確率変数版を定義することもでき, 分布の広がりの大きさの指標として, 四分位数の確率変数版を定義することもできる. 期待値(平均)と分散(と標準偏差)の組み合わせと中央値と四分位数の組み合わせは目的と状況に合わせて使い分けるとよいだろう.

確率変数の分散については

$$\mathrm{var}(Y) = E[Y^2] - E[Y]^2$$

という公式もよく使われる. この公式は以下のようにして証明される:

$$\begin{aligned}\mathrm{var}(Y) &= E\left[(Y-\mu_Y)^2\right] = E\left[Y^2 - 2\mu_Y Y + \mu_Y^2\right] = E[Y^2] - 2\mu_Y E[Y] + \mu_Y^2 \\ &= E[Y^2] - 2\mu_Y\mu_Y + \mu_Y^2 = E[Y^2] - \mu_Y^2 = E[Y^2] - E[Y]^2.\end{aligned}$$

この証明では期待値を取る操作の線形性と規格化条件しか使われていないので, この結果は非常に一般的に成立しており, 後で説明する連続確率変数の分散についても成立している.

注意: コンピュータによる浮動小数点演算で分散を実装するときには, $\mathrm{var}(Y) = E[Y^2] - E[Y]^2$ という公式を使わずに, 分散の定義式 $E\left[(Y-\mu_Y)^2\right]$ を使って実装した方が良い場合がある. なぜならば, 浮動小数点演算では大きな数値の差が小さな数値になるとき桁落ちが生じるからである. 同じ程度の大きさの数値の差の計算になり易い $Y - \mu_Y$ と違って, $E[Y^2] - E[Y]^2$ は大きな数のあいだの差で計算結果が小さな数になる可能性が高いので注意が必要である.

4.8 問題: 期待値と分散の特徴付け

X は有限離散分布に従う確率変数であるとし, 実数 t について $f(t) = E[(X-t)^2]$ とおく. このとき, $f(t)$ を最小化する t の値は X の期待値 $E[X]$ になり, 最小値 $f(E[X])$ は X の分散 $\mathrm{var}(X)$ になることを示せ.

解答例: 期待値を取る操作の基本性質のみを使って示せる. $\mu = E[X]$ とおく. このとき,

$$(X-t)^2 = ((X-\mu)-(t-\mu))^2 = (X-\mu)^2 - 2(t-\mu)(X-\mu) + (t-\mu)^2$$

なので, これに $E[\ ]$ を作用させ, $E[\ ]$ の線形性(足し算と引き算と定数倍を $E[\ ]$ の外に出せること)と規格化条件(定数の期待値はその定数そのものの値になること)および $E[X] = \mu$, $E[(X-\mu)^2] = \mathrm{var}(X)$ を使うと,

$$E[(X-t)^2] = E[(X-\mu)^2] - 2(t-\mu)(E[X]-\mu) + (t-\mu)^2 = \mathrm{var}(X) + (t-\mu)^2.$$

これより, $f(t)$ は $t = \mu = E[X]$ で最小値 $\mathrm{var}(X)$ になることがわかる.

解答終

注意: 上の証明では期待値を取る操作の基本性質しか使っておらず, X が有限離散分布に従う確率変数であることを使っていない. 上の問題の結果は X が有限離散分布に限らないもっと一般の分布に従っている場合にも成立している.

注意: $g(t) = E[|X - t|]$ を最小化する t として確率変数 X の中央値を特徴付けることができる. ただし, 確率変数の中央値は一般には一意(ただ一つ)には決まらない. この結果は数学的にそれなりに非自明なので, 数学に自身がある人は確率変数の中央値を自分で適切に定義して, 中央値であることと $g(t)$ を最小化することが同値であることを証明してみよ. (おそらく初学者にとってこれは難問である.)

注意: a を確率変数 X の中央値とするとき, $E[|X - a|]$ は中央値からの距離の期待値になる. これも確率変数 X の分布の広がりの大きさの指標の1つとして使える. 分散の定義について「どうして2乗するのか?」という質問は非常によくある質問だが, 分布の広がりの大きさの指標は無数にあるので, **分散に特権的な優位性があるわけではない**. 期待値の代わりに中央値を考え, 2乗せずに単なる絶対値を考えるのでもよい. その他にも中央値と四分位数の組み合わせもよく使われている. 分散や四分位数について教えることが多いのは単によく使われるからである. しかし, 期待値と分散の組み合わせを頻繁に目にする数学的理由もあって, **中心極限定理** によって正規分布で近似されると考えられる分布が応用上よく現れるという事情がある.

注意: 標本(サンプル, データ)の平均や分散や中央値と確率変数(もしくはそれが従う分布)の平均や分散や中央値の概念を統計学では厳密に区別する必要がある. 混乱しないように注意して欲しい. 混乱を防ぐためには単に「平均」「分散」などと言うことを少なめにして, できるだけ「確率変数の平均や分散」「標本の平均や分散」のように「○○の平均や分散」と言うようにすればよいと思う.

4.9 問題: 平均二乗誤差のバイアス・バリアンス分解

X は確率変数であるとし, その期待値 $\mu = E[X]$ と分散 $E[(X-\mu)^2]$ がうまく定義されているとする. 定数 θ に対して次が成立することを示せ:

$$E[(X-\theta)^2] = (E[X]-\theta)^2 + E[(X-\mu)^2].$$

注意: 確率変数 X の実現値を定数 θ の推定値とみなす状況において, 左辺の $E[(X-\theta)^2]$ を推定の **平均二乗誤差** と呼び, $E[X]-\theta$ を推定の **バイアス**(bias)と呼び, $E[(X-\mu)^2]$ を **バリアンス** (variance, 分散)と呼び, この公式を **平均二乗誤差のバイアス・バリアンス分解** と呼ぶ.

解答例: $(X-\theta)^2$ の括弧の内側に $0 = -\mu + E[X]$ を足すと,

$$\begin{aligned}(X-\theta)^2 &= ((X-\mu)-(E[X]-\theta))^2 \\ &= (X-\mu)^2 + 2(E[X]-\theta)(X-\mu) + (E[X]+\theta)^2.\end{aligned}$$

$E[X]-\theta$ と $(E[X]+\theta)^2$ が確率変数ではなく, 定数であることに注意しながら, 期待値を取る操作の線形性と規格化条件を使うと,

$$\begin{aligned}(X-\theta)^2 &= E[(X-\mu)^2] + 2(E[X]-\theta)(E[X]-\mu) + (E[X]-\theta)^2 \\ &= E[(X-\mu)^2] + (E[X]-\theta)^2.\end{aligned}$$

たし算の順序を交換すれば欲しい公式が得られる.

解答終

注意: 上の解答例で用いた式の変形は統計学の文脈で数え切れないくらい登場する. 例えば, 前節の問題でもこの方法を用いた.

4.10 カテゴリカル分布の定義

番号の集合 $\{1, 2, \ldots, r\}$ 上の確率質量函数

$$P(i) = p_i \quad (p_1, \ldots, p_r \geq 0,\ p_1 + \cdots + p_r = 1)$$

が定める有限離散分布を **カテゴリカル分布** と呼び, 次のように表す:

$$\mathrm{Categorical}(p_1, p_2, \ldots, p_r).$$

確率の表は次のようになる:

x	1	2	$\cdots$	r
$P(x)$	p_1	p_2	$\cdots$	p_r

多項分布の確率質量函数は次のようにも書ける:

$$P(i) = p_1^{\delta_{i1}} \cdots p_r^{\delta_{ir}}.$$

ここで δ_{ij} は $i = j$ のとき 1 になり, それ以外のとき 0 になるKroneckerのデルタである.

`categ = Categorical(0.1, 0.2, 0.3, 0.4)` のとき `rand(categ)` は値が $1, 2, 3, 4$ にそれぞれ確率 $0.1, 0.2, 0.3, 0.4$ でなる確率変数だとみなされる. (値がランダムに決まる量も確率変数と呼ぶ習慣がある.) `categ` は上の方で繰り返し使ったルーレットの分布に等しい.

In [10]:
```
categ = Categorical(0.1, 0.2, 0.3, 0.4)
```

Out[10]:
```
Categorical{Float64, Vector{Float64}}(support=Base.OneTo(4), p=[0.1, 0.2, 0.3, 0.4])
```

In [11]:
```
@show rand(categ, 20);
```

```
rand(categ, 20) = [4, 4, 4, 3, 4, 2, 3, 3, 4, 3, 2, 2, 1, 4, 2, 4, 4, 4, 4, 4]
```

In [12]:
```
bar(support(categ), a -> pdf(categ, a); label="", title="Categorical(0.1, 0.2, 0.3, 0.4)", alp
```

Out[12]:

4.11 問題: ルーレットの例の賞金の期待値と分散

ルーレットの例の分布は $\mathrm{Categorical}(0.1, 0.2, 0.3, 0.4)$ に等しい. この分布に従う確率変数を X と書く. 賞金の確率変数 $g(X)$ は

$$g(1) = 10000, \quad g(2) = 3000, \quad g(3) = 1000, \quad g(4) = 300,$$

で定義される. X と $g(X)$ の期待値と分散と標準偏差を求めよ. 整数にならない標準偏差は小数点以下第1桁目まで求めよ.

解答例: X の期待値, 分散, 標準偏差はそれぞれ

$$\mu_X = 1, \quad \sigma_X^2 = 1, \quad \sigma_X = 1.$$

$g(X)$ の期待値, 分散, 標準偏差はそれぞれ

$$\mu_{g(X)} = 2020, \quad \sigma_{g(X)}^2 = 8055600, \quad \sigma_{g(X)} \approx 2838.2.$$

詳しい計算法については以下を見よ. 以下と同じ数値が得られれば正解である. 浮動小数点計算では小数点以下3桁まで正しければ正解であるとする(四捨五入すること).

解答終

X の期待値と分散と標準偏差:

In [13]:
```
μ_X = 1 * 0.1 + 2 * 0.2 + 3 * 0.3 + 4 * 0.4
@show μ_X;
```

```
μ_X = 3.0
```

X の分散と標準偏差:

In [14]:
```
σ²_X = (1 - 3)^2 * 0.1 + (2 - 3)^2 * 0.2 + (3 - 3)^2 * 0.3 + (4 - 3)^2 * 0.4
σ_X = √σ²_X
@show σ²_X σ_X;
```

```
σ²_X = 1.0
σ_X = 1.0
```

$g(X)$ の期待値:

In [15]:
```
μ_g = 10000 * 0.1 + 3000 * 0.2 + 1000 * 0.3 + 300 * 0.4 ▷ Int
@show μ_g;
```

```
μ_g = 2020
```

$g(X)$ の分散と標準偏差:

In [16]:
```
σ²_g = (10000 - 2020)^2 * 0.1 + (3000 - 2020)^2 * 0.2 + (1000 - 2020)^2 * 0.3 + (300 - 2020)^2
σ_g = √σ²_g
@show σ²_g round(σ_g; digits=1);
```

```
σ²_g = 8055600
round(σ_g; digits = 1) = 2838.2
```

以下はJulia言語のより高級な機能を使った計算法である.

In [17]:
```
categ = Categorical(1//10, 2//10, 3//10, 4//10)
μ_X = mean(categ) ▷ Int
σ²_X = var(categ) ▷ Int
σ_X = std(categ) ▷ Int
@show μ_X σ²_X σ_X;
```

```
μ_X = 3
σ²_X = 1
σ_X = 1
```

期待値を取る操作は分布 dist と確率変数 f に期待値を対応させる函数 $E(f, \mathrm{dist})$ として定義可能である.

In [18]:
```
E(f, dist::DiscreteUnivariateDistribution; imax = 10^6) =
    sum(x -> f(x) * pdf(dist, x), support(dist)[1:min(end, imax)])
```

Out[18]:
```
E (generic function with 1 method)
```

$E(f, \mathrm{dist})$ を使って X の期待値と分散を計算すると以下のようになる.

In [19]:
```
categ = Categorical(1//10, 2//10, 3//10, 4//10)
μ_X = E(x -> x, categ) ▷ Int
σ²_X = E(x -> (x - μ_X)^2, categ) ▷ Int
σ_X = √σ²_X ▷ Int
@show μ_X σ²_X σ_X;
```

```
μ_X = 3
σ²_X = 1
σ_X = 1
```

In [20]:
```
g(x) = get((10000, 3000, 1000, 300), x, 0) # x = 1,2,3,4 のそれぞれを 10000, 3000, 1000, 300
μ_g = E(g, categ) ▷ Int
σ²_g = E(x -> (g(x) - μ_g)^2, categ) ▷ Int
σ_g = √σ²_g
@show μ_g σ²_g round(σ_g; digits=1);
```

```
μ_g = 2020
σ²_g = 8055600
round(σ_g; digits = 1) = 2838.2
```

4.12 Bernoulli分布の定義

$0 \leq p \leq 1$ と仮定する. 値が $1, 0$ になる確率をそれぞれ $p, 1-p$ とすることによって定まる離散分布を成功確率 p の **Bernoulli分布** (ベルヌーイ分布, ベルヌイ分布)と呼び, 次のように表す:

$$\mathrm{Bernoulli}(p).$$

この分布の確率質量函数は次のようになる:

$$P(1) = p, \quad P(0) = 1 - p.$$

4.13 問題: Bernoulli分布の期待値と分散

Bernoulli分布 $\mathrm{Bernoulli}(p)$ に従う確率変数 X の期待値と分散を求めよ. さらに分散の最大値とそれを与える p を求めよ.

解答例: X の期待値 $E[X]$ と分散 $\mathrm{var}(X)$ は以下のように計算される:

$$
\begin{aligned}
E[X] &= \sum_{x=1,0} xp(x) = 1p + 0(1-p) = p, \\
E[X^2] &= \sum_{x=1,0} x^2 p(x) = 1^2 p + 0^2(1-p) = p, \\
\mathrm{var}(X) &= E[X^2] - E[X]^2 = p - p^2 = p(1-p).
\end{aligned}
$$

$\mathrm{var}(X) = -(p-1/2)^2 + 1/4$ なので分散 $\mathrm{var}(X)$ は $p = 1/2$ のとき最大値 $1/4$ になる.

解答終

In [21]:
```
p = 0.3
bern = Bernoulli(p)
```

Out[21]:
```
Bernoulli{Float64}(p=0.3)
```

In [22]:
```
@show mean(bern) var(bern);
```

```
mean(bern) = 0.3
var(bern) = 0.21
```

In [23]:
```
@show rand(bern, 20);
```

```
rand(bern, 20) = Bool[0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1]
```

In [24]:
```
bar(0:1, x -> pdf(bern, 1 - x); label="", title="Bernoulli(0.3)", alpha=0.5)
plot!(; xtick=(0:1, 1 .- (0:1)), ytick=0:0.1:1)
```

Out[24]:

4.14 オッズの定義と例

成功確率 p に対して, 失敗確率分の成功確率

$$
\frac{p}{1-p} = \frac{1}{1-p} - 1
$$

を **オッズ** (odds)と呼ぶ. オッズは $0 \leq p < 1$ について単調増加函数になり, $p = 1$ のときオッズは ∞ になると考えられる.

オッズは賭け事の世界では普通によく使われる用語である. オッズは勝利確率 p の単調増加函数になっているので, 勝利確率とは別の勝ち易さの指標になっている.

例えば, 勝利確率が p のギャンブルで勝利したときに賭けた金額の何倍の賞金をもらえることにすると公平になるだろうか? 勝利確率が p の賭け事に1万円賭けたとき, 勝利したら賭けた1万円はそのままで賞金として k 万円もらえ, 敗北したら賭けた1万円が全額没収されるとする. そのときの期待値は $pk - (1-p)$ 万円なので, 期待値が 0 の公平なギャンブルにするためには k をオッズの逆数にしなければいけない. オッズの低い勝ち難いギャンブルではその分だけ勝利したときに沢山の賞金を獲得できないと損になる.

例えば, 勝利確率が $1/2$ のギャンブルではオッズは $(1/2)/(1/2) = 1$ になる.

例えば, 勝利確率が $1/3$ のギャンブルではオッズは $(1/3)/(2/3) = 1/2$ になり, 勝利したときに賭けた金額の2倍の賞金が得られるなら公平なギャンブルになる. 実際, そのとき, 1 万円を賭けると, $1/3$ の確率で 2 万円もらえ, $2/3$ の確率で 1 万円を失うので, 確かに期待値は 0 になる. (これはBernoulli分布モデルにおける確率変数の期待値の例にもなっている.)

4.15 オッズとロジットおよびロジスティック変換

確率 p についてそのオッズを $u = p/(1-p) = 1/(1-p) - 1$ と書くと, 逆に確率はオッズで $p = u/(1+u) = 1 - 1/(1+u)$ と書ける. これによって, $0 < p < 1$ と $0 < u < \infty$ のあいだの一対一の対応が得られる. さらに $x = \log(u)$, $u = \exp(x)$ とおくと, $0 < u < \infty$ と $-\infty < x < \infty$ のあいだの一対一に対応も得られる. p と x の関係を直接書き下すと,
$p = \exp(x)/(1+\exp(x)) = 1/(1+\exp(-x))$, $x = \log(p/(1-p))$ となる. 以上によって,

$$p = \frac{u}{1+u} = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}}, \quad u = \frac{p}{1-p} = e^x, \quad x = \log(u) = \log\frac{p}{1-p}$$

によって,

$$0 < p < 1, \quad 0 < u < \infty, \quad -\infty < x < \infty$$

のあいだの一対一対応が得られたことになる. これらは賭け事とは無関係に非常によく使われる変数変換であり, 賭け事由来のオッズの概念はこのような形でも数学的に非常に役に立つことが知られている.

以下の変数変換はそれぞれ **ロジット変換** (logit変換), **ロジスティック変換** (logistic変換)と呼ばれている:

$$x = \mathrm{logit}(p) = \log\frac{p}{1-p}, \quad p = \mathrm{logistic}(x) = \frac{1}{1+e^{-x}}.$$

確率を意味するパラメータの p を $-\infty$ から ∞ のあいだを自由に動ける変数 x に変換してから様々な計算を行った方が都合が良い場合があるので, これらの変換はよく使われる. (既存の統計ソフトが行ってくれる計算においてユーザーから見えない所でこのような変換が使われている場合もある.)

4.16 Bernoulli試行の確率分布

Bernoulli分布 $\mathrm{Bernoulli}(p)$ における $x = 1, 0$ の確率は

$$P(x) = p^x(1-p)^{1-x}$$

と書ける. 実際, $P(1) = p^1(1-p)^0 = p$ となり, $P(0) = p^x(1-p)^{1-x} = p^0(1-p)^1 = 1-p$ となる. これをBernoulli試行の場合に拡張しよう.

試行回数 n のBernoulli試行の結果は長さ n の 1 と 0 からなる数列 $(x_1, \dots, x_n)$ になる. そのような列全体の個数は, 各々の x_k が2通りでそれらが n 個あるから, 2^n 個になる. ($p = 0.5$ の場合を除いて, すべての場合の確率が等しくなるわけではない.)

Bernoulli試行が毎回確率 p で 1 が出て確率 $1-p$ で 0 が出る独立な試行の繰り返しなので, 与えられた $x_1, \dots, x_n = 1, 0$ について, n 回のBernoulli試行の結果が $(x_1, \dots, x_n)$ になる確率 $P(x_1, \dots, x_n)$ は, Bernoulli分布における確率

$$P(x_i) = p^{x_i}(1-p)^{1-x_i}$$

の積

$$P(x_1, \dots, x_n) = P(x_1)\cdots P(x_n) = \prod_{i=1}^{n}\left(p^{x_i}(1-p)^{1-x_i}\right) = p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)}$$

になると考えられる. (数学的にはこれによってBernoulli試行の確率分布を定める.) これは, $x_1, \dots, x_n$ の中の 1 (=「当たり」)の個数を k と書くと,

$$p^k(1-p)^{n-k}$$

とシンプルに書来直される.

このようにして定まる集合 $\{\,(x_1, \dots, x_n) \mid x_1, \dots, x_n = 1, 0\,\}$ 上の離散分布を試行回数 n, 成功確率 p の **Bernoulli試行の確率分布** と呼ぶことにする.

4.17 問題: n=3, p=0.3 の場合のBernoulli試行の確率分布

$n = 3$, $p = 0.3$ のBernoulli試行の確率分布における確率質量函数を具体的に書き下し, 確率の総和がぴったり 1 になることを確認せよ.

解答例: $n = 3$, $p = 0.3$ のBernoulli試行の確率分布を考える. (x_1, x_2, x_3) を単に並べて $x_1x_2x_3$ と書くと,

$$\{\,(x_1, x_2, x_3) \mid x_1, x_2, x_3 = 1, 0\,\} = \{111, 011, 101, 110, 100, 010, 001, 000\}.$$

確率質量函数の全体は以下のように記述される:

$$
\begin{aligned}
&P(111) = 0.3^3 = 0.027, \\
&P(011) = P(101) = P(110) = 0.3^2 \cdot 0.7 = 0.063, \\
&P(100) = P(010) = P(001) = 0.3 \cdot 0.7^2 = 0.147, \\
&P(000) = 0.7^3 = 0.343.
\end{aligned}
$$

これら8通りの確率の和は

$$0.027 + 3 \cdot 0.063 + 3 \cdot 0.147 + 0.343 = 0.027 + 0.189 + 0.441 + 0.343 = 1$$

と確かにぴったり 1 になっている.

解答終

In [25]:
```
27//1000 + 3*63//1000 + 3*147//1000 + 343//1000 ▷ Int
```

Out[25]: 1

4.18 Bernoulli試行から応用上基本的かつ重要な確率分布が大量に得られること

Bernoulli試行から以下の確率分布が得られる:

- Bernoulli分布 (これは $n = 1$ のBernoulli試行の確率分布と $n = 1$ の二項分布に等しい)
- 二項分布 (n 回のBernoulli試行で出た 1 の個数の分布)
- 幾何分布 (これは $k = 1$ の負の二項分布に等しい)
- 負の二項分布 (Bernoulli試行を k 回 1 が出るまで続けたときに出た 0 の個数の分布)

これらの連続極限として,

- 正規分布 (二項分布の中心極限定理)
- Poisson分布 (二項分布の時間連続極限)
- ガンマ分布 (負の二項分布の時間連続極限)

も得られる. 正規分布, ガンマ分布, Poisson分布は極めて基本的な確率分布であり, 非常に頻繁に使われる.

Bernoulli試行について十分に理解すれば, 他の重要かつ基本的な確率分布についても楽に理解できるようになるだろう!

4.19 二項定理

n 個から k 個を選ぶ組み合わせの数を

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)(n-2)\cdots(n-k+1)}{k!}$$

と書き, **二項係数** と呼ぶ.

互いに可換な文字 x, y の和の n 乗の展開したときの $x^k y^{n-k}$ の係数を a_k と書く:

$$(x+y)^n = \sum_{k=0}^{n} a_k x^k y^{n-k}.$$

このとき, a_k は掛けられてる n 個の $x+y$ の中から x をピックアップする k 個を選ぶ組み合わせの数に等しいので, $a_k = \binom{n}{k}$ が成立する. ゆえに

$$(x+y) = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}.$$

特に $y = 1$ のとき,

$$(1+x)^n = \sum_{k=0}^{n} \binom{n}{k} x^k.$$

これらの結果を **二項定理** と呼ぶ.

注意: 高校の教科書のように二項係数を ${}_nC_k$ と書いてもよいが, このノートではこの書き方をしない. 後で二項係数 $\binom{n}{k}$ の n は整数でない場合にも拡張されることになる.

4.20 二項分布の定義

二項分布 は試行回数 n, 成功確率 p のBernoulli試行で生じる 1 と 0 からなる長さ n の列の中の 1 の個数の分布として得られる.

試行回数 n, 成功確率 p のBernoulli試行において 1 と 0 からなる列 $(x_1, x_2, \ldots, x_n)$ が生じる確率は

$$p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)}$$

と書けるのであった. 1 と 0 からなる列 $(x_1, x_2, \ldots, x_n)$ でそれが含む 1 の個数 $x_1+\cdots+x_n$ が k になるものの個数は, n 個の x_i 達から k 個の 1 になるものを選ぶ組み合わせの数

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

になるので, 試行回数 n, 成功確率 p のBernoulli試行において, 1 の個数 $x_1+\cdots+x_n$ が k になる確率は

$$P(k) = \binom{n}{k}p^k(1-p)^{n-k} = \frac{n!}{k!(n-k)!}p^k(1-p)^{n-k}$$

になる. この確率質量函数で定義される集合 $\{0, 1, \ldots, n\}$ 上の有限離散分布を **二項分布** と呼び, 次のように表す:

$$\mathrm{Binomial}(n, p).$$

二項分布における確率の総和が 1 になることは二項定理を使えばただちに確認できる:

$$\sum_{k=0}^{n} P(k) = \sum_{k=0}^{n}\binom{n}{k}p^k(1-p)^{n-k} = (p+(1-p))^n = 1.$$

4.21 問題: 二項分布の期待値と分散

K は二項分布に従う確率変数であるとする:

$$K \sim \mathrm{Binomial}(n, p).$$

K の期待値と分散を求めよ.

解答例: 二項定理を使う. K の期待値は次のようにして得られる:

$$\begin{aligned}
E[K] &= \sum_{k=0}^{n} kP(k) = \sum_{k=0}^{n} k\frac{n!}{k!(n-k)!}p^k(1-p)^{n-k} = \sum_{k=1}^{n} n\frac{(n-1)!}{(k-1)!(n-k)!}p^k(1-p)^{n-k} \\
&= np\sum_{k=1}^{n}\binom{n-1}{k-1}p^{k-1}(1-p)^{n-k} = np(p+(1-p))^{n-1} = np,
\end{aligned}$$

上の計算の5番目の統合で二項定理を使っている. この計算法と同様のアイデアで以下のように計算することができる:

$$\begin{aligned}
E[K(K-1)] &= \sum_{k=0}^{n} k(k-1)P(k) = \sum_{k=2}^{n} k(k-1)\frac{n!}{k!(n-k)!}p^k(1-p)^{n-k} \\
&= \sum_{k=1}^{n} n(n-1)\frac{(n-2)!}{(k-2)!(n-k)!}p^k(1-p)^{n-k} = n(n-1)p^2\sum_{k=2}^{n}\binom{n-2}{k-2}p^{k-2}(1-p)^{n-k} \\
&= n(n-1)p^2(p+(1-p))^{n-2} = n(n-1)p^2,
\end{aligned}$$

この計算の5番目の統合でも上と同様に二項定理を使っている. $\mathrm{var}(K) = E[K^2] - E[K]^2$ を使って K の分散を計算しよう:

$$\begin{aligned}
&E[K(K-1)] = E[K^2] - E[K], \\
&E[K^2] = E[K(K-1)] + E[K] = n(n-1)p^2 + np = n^2p^2 - np^2 + np = np(1-p) + n^2p^2, \\
&\mathrm{var}(K) = E[K^2] - E[K]^2 = np(1-p) + n^2p^2 - (np)^2 = np(1-p).
\end{aligned}$$

解答終

注意: 二項分布 $\mathrm{Binomial}(n, p)$ に従う確率変数の期待値 np と分散 $np(1-p)$ は $n=1$ の場合のBernoulli分布 $\mathrm{Bernoulli}(p)$ の期待値と p と分散 $p(1-p)$ のちょうど n 倍になっている.

期待値についてこの事実は次のように説明される. Bernoulli試行における 1 と 0 の列 $(x_1, \ldots, x_n)$ に x_i を対応させる函数によって定まる確率変数を X_i と書くと, 1 と 0 からなる列の中の 1 の個数を意味する確率変数は $K = X_1 + \cdots + X_n$ になる. K の期待値が二項分布の期待値になる. 各々の X_i は成功確率 p のBernoulii分布に従うので, その期待値は $E[X_i] = p$ になり, 期待値を取る操作の線形性(加法性)より,

$$E[K] = E[X_1] + \cdots + E[X_n] = p + \cdots + p = np.$$

分散についても

$$\mathrm{var}(K) = \mathrm{var}(X_1) + \cdots + \mathrm{var}(X_n) = p(1-p) + \cdots + p(1-p) = np(1-p)$$

が成立することは偶然ではなく, 必然的にこうなる理由がある. この点については後で確率変数の独立性について説明してから詳しく説明する. 答えを先走って言うと, 独立な確率変数の和の分散はそれぞれの確率変数の分散の和に等しくなる.

二項分布における確率質量函数のグラフは以下のようになる.

In [26]:
```
bin = Binomial(16, 0.3)
x = support(bin)
bar(x, x -> pdf(bin, x); label="", title="Binomial(20, 0.3)", alpha=0.5)
plot!(; xtick=x)
```

Out[26]:

二項分布 $\mathrm{Binomial}(n,p)$ はその期待値 np と反対側の期待値 $n(1-p)$ の両方が大きなとき, $\mathrm{Binomial}(n,p)$ と等しい平均 $\mu = np$ と分散 $\sigma^2 = np(1-p)$ を持つ正規分布で近似されること(**二項分布の中心極限定理**)がよく知られている. (実際には np がそう大きくなくても悪くない近似になる. これは階乗のStirlingの公式による近似がそう悪くならないからだと考えられる.)

二項分布の中心極限定理は階乗に関するStirlingの(近似)公式を使えば示せるが, そこそこ面倒な計算が必要になる. そのような計算を避けたい人も避けなかった人も, コンピュータで二項分布と正規分布のグラフを重ねてプロットすれば二項分布が正規分布で近似されることをすぐに納得できると思う. 正規分布については後で詳しく説明する.

In [27]:
```
bin = Binomial(16, 0.3)
μ, σ = mean(bin), std(bin)
x = support(bin)
bar(x, x -> pdf(bin, x); label="Binomial(20, 0.3)", alpha=0.5)
plot!(; xtick=x)
plot!(Normal(μ, σ); label="normal approx.", lw=2, c=:red)
```

Out[27]:

次のセルを見ればわかるように, p が小さな場合の二項分布は期待値 np のPoisson分布で近似される.

より正確に言うと, 固定された $\lambda > 0$ について, $p = \lambda/n$ とおくと, 二項分布 $\mathrm{Binomial}(n,p)$ は $n \to \infty$ でPoisson分布 $\mathrm{Poisson}(\lambda)$ に近付く. Poisson分布(ポアソン分布)の定義を知らない人はこれをPoisson分布の定義だと思ってもよい. この点については後で説明する予定である.

In [28]:

```
λ, n = 3, 100
p = λ/n
bin = Binomial(n, p)
μ, σ = mean(bin), std(bin)
x = 0:5λ
groupedbar(x, [pdf.(bin, x) pdf.(Poisson(n*p), x)];
    label=["Binomial(300, 0.01)" "Poisson($(n*p))"], alpha=0.5)
plot!(; xtick=x, xlim=(-2, 16.5))
plot!(Normal(μ, σ); label="normal approximation", lw=1, c=:red, ls=:dash)
```

Out[28]:

In [29]:

```
λ, n = 3, 1000
p = λ/n
bin = Binomial(n, p)
μ, σ = mean(bin), std(bin)
x = 0:5λ
groupedbar(x, [pdf.(bin, x) pdf.(Poisson(n*p), x)];
    label=["Binomial(300, 0.01)" "Poisson($(n*p))"], alpha=0.5)
plot!(; xtick=x, xlim=(-2, 16.5))
plot!(Normal(μ, σ); label="normal approximation", lw=1, c=:red, ls=:dash)
```

Out[29]:

4.22 問題: 当たりが30%の確率で出るくじを100回引いた場合

当たりが毎回 $p = 30\% = 0.3$ の確率で出るくじを $n = 100$ 回引いたときに当たりが出た回数 K は二項分布 $\mathrm{Binomial}(n, p)$ に従う確率変数とみなされる.

(1) 当たりの回数 K の期待値と分散と標準偏差を求めよ.

(2) 当たりの回数 K が25回以下になる確率と20回以下になる確率を求めよ.

(3) 当たりの回数 K が35回以上になる確率と40回以上になる確率を求めよ.

有効桁3桁で求めよ.

解答例: $p = 0.3$, $n = 100$ とおく.

(1) K の期待値, 分散, 標準偏差:

$$E[K] = np = 30, \quad \mathrm{var}(K) = np(1-p) = 21, \quad \mathrm{std}(K) = \sqrt{\mathrm{var}(K)} \approx 4.58.$$

(2) $K \leq 25$ となる確率:

$$P(K \leq 25) = \sum_{k=0}^{25} \binom{n}{k} p^k (1-p)^{n-k} \approx 0.163.$$

$K \leq 30$ となる確率:

$$P(K \leq 20) = \sum_{k=0}^{20} \binom{n}{k} p^k (1-p)^{n-k} \approx 0.0165.$$

(2) $K \geq 35$ となる確率:

$$P(K \geq 35) = \sum_{k=35}^{100} \binom{n}{k} p^k (1-p)^{n-k} \approx 0.163.$$

$K \geq 40$ となる確率:

$$P(K \geq 40) = \sum_{k=40}^{100} \binom{n}{k} p^k (1-p)^{n-k} \approx 0.0210.$$

解答終

注意: WolframAlpha (https://www.wolframalpha.com/)では確率 $P(K \leq 25)$ を

- sum_{k=0}^{25} binom(100, k) 0.3^k (1 - 0.3)^{100-k} (https://www.wolframalpha.com/input?i=sum_%7Bk%3D0%7D%5E%7B25%7D+binom%28100%2C+k%29+0.3%5Ek+%281+-+0.3%29%5E%7B100-k%7D)

と入力して求めることができる. 他の場合も試してみよ. (もちろんJulia言語を導入して使ってもよい.)

In [30]:

```julia
logP(n, p, k) = logabsbinomial(n, k)[1] + k*log(p) + (n-k)*log(1-p)

p = 0.3
n = 100
@show n*p
@show n*p*(1-p)
@show √(n*p*(1-p))
for Kmax in (25, 20)
    @eval @show exp(logsumexp(logP(n, p, k) for k in 0:$Kmax))
end
for Kmin in (35, 40)
    @eval @show exp(logsumexp(logP(n, p, k) for k in $Kmin:n))
end
```

```
n * p = 30.0
n * p * (1 - p) = 21.0
√(n * p * (1 - p)) = 4.58257569495584
exp(logsumexp((logP(n, p, k) for k = 0:25))) = 0.16313010446635048
exp(logsumexp((logP(n, p, k) for k = 0:20))) = 0.016462853241869316
exp(logsumexp((logP(n, p, k) for k = 35:n))) = 0.16285828837178473
exp(logsumexp((logP(n, p, k) for k = 40:n))) = 0.020988576003924307
```

In [31]:
```
p = 0.3
n = 100
@eval @show bin = Binomial($n, $p)
@show mean(bin)
@show var(bin)
@show std(bin)
for Kmax in (25, 20)
    @eval @show cdf(bin, $Kmax)
end
for Kmin in (35, 40)
    @eval @show ccdf(bin, $Kmin - 1)
end
```

```
bin = Binomial(100, 0.3) = Binomial{Float64}(n=100, p=0.3)
mean(bin) = 30.0
var(bin) = 21.0
std(bin) = 4.58257569495584
cdf(bin, 25) = 0.16313010446635237
cdf(bin, 20) = 0.0164628532418695
ccdf(bin, 35 - 1) = 0.16285828837178704
ccdf(bin, 40 - 1) = 0.020988576003924644
```

4.23 幾何分布の定義

$0 < p \leq 1$ であると仮定し, $m = 0, 1, 2, \ldots$ であるとする.

確率 p で 1 が確率 $1-p$ で 0 が生じるようなBernoulli試行を続けたとき, ちょうど $m+1$ 回目で 1 が初めて出る確率は, m 回 0 が続いて $m+1$ 回目に 1 が出る確率

$$(1-p)^m \times p$$

に等しいので,

$$P(m) = p(1-p)^m$$

になる. この確率質量函数で定義される可算集合 $\{0, 1, 2, \ldots\}$ 上の無限離散分布を **幾何分布** (geometric distribution)と呼び, 次のように表す:

$$\mathrm{Geometric}(p).$$

幾何分布における確率の総和が 1 になることは, 等比級数の和を使って, 次のように証明される:

$$\sum_{m=0}^{\infty} P(m) = p \sum_{m=0}^{\infty} (1-p)^m = \frac{p}{1-(1-p)} = 1.$$

$1-(1-p)$ が p になることがこれ以後頻繁に現れる.

幾何分布の確率質量函数の形は次のようになる.

In [32]:
```
geom = Geometric(0.3)
x = 0:16
bar(x, x -> pdf(geom, x); label="", title="Geometric(0.3)", alpha=0.5)
plot!(; xtick = x)
```

Out[32]:

p が小さい場合の幾何分布は平均 $\lambda = 1/p$ の指数分布で近似される: λ が大きなとき,

$$p(1-p)^m = \frac{1}{\lambda}\left(1-\frac{1}{\lambda}\right)^m \approx \frac{1}{\lambda}\exp\left(-\frac{m}{\lambda}\right).$$

In [33]:
```
geom = Geometric(0.03)
p = succprob(geom)
x = 0:200
plot(x, x -> pdf(geom, x); label="Geometric($p)")
plot!(Exponential(1/p), 0, 200; ls=:dash, label="Exponential(1/$p)")
```

Out[33]:

4.24 幾何分布の期待値と分散

幾何分布に従う確率変数 M の期待値と分散は次のようになる:

$$E[M] = \frac{1-p}{p}, \quad \mathrm{var}(M) = \frac{1-p}{p^2}.$$

詳しくは以下のように計算される. 上で経験した等比級数の和の計算から,

$$\sum_{m=0}^{\infty}(1-p)^m = \frac{1}{p}$$

となることもわかる. 両辺を p で微分して -1 倍する操作を2回施すと,

$$\sum_{m=1}^{\infty} m(1-p)^{m-1} = \frac{1}{p^2}, \quad \sum_{m=2}^{\infty} m(m-1)(1-p)^{m-2} = \frac{2}{p^3}.$$

これらの公式を使うと,

$$\begin{aligned}
E[M] &= \sum_{m=0}^{\infty} mP(m) = \sum_{m=1}^{\infty} mp(1-p)^m = p(1-p)\sum_{m=1}^{\infty} m(1-p)^{m-1} = p(1-p)\frac{1}{p^2} = \frac{1-p}{p}, \\
E[M(M-1)] &= \sum_{m=0}^{\infty} m(m-1)P(m) = \sum_{m=2}^{\infty} m(m-1)p(1-p)^m \\
&= p(1-p)^2\sum_{m=2}^{\infty} m(m-1)(1-p)^{m-2} = p(1-p)^2\frac{2}{p^3} = \frac{2(1-p)^2}{p^2} \\
E[M(M-1)] &= E[M^2] - E[M], \\
E[M^2] &= E[M(M-1)] + E[M] = \frac{2(1-p)^2}{p^2} + \frac{1-p}{p}, \\
\mathrm{var}(M) &= E[M^2] - E[M]^2 = \frac{2(1-p)^2}{p^2} + \frac{1-p}{p} - \frac{(1-p)^2}{p^2} = \frac{(1-p)^2}{p^2} + \frac{p(1-p)}{p^2} = \frac{1-p}{p^2}.
\end{aligned}$$

$N = M+1$ とおく. N は始めて 1 が出るまでの試行回数を意味する確率変数になる. N の確率質量函数は

$$P(n-1) = p(1-p)^{n-1}$$

になる. N の期待値と分散は上の結果より, 次のようになる:

$$E[N] = E[M]+1 = \frac{1}{p}, \quad \mathrm{var}(N) = \mathrm{var}(M) = \frac{1-p}{p^2}.$$

$E[N] = 1/p$ の意味: 当たりが確率 p で出るガチャを回し続けるとき, 当たりが初めて出るまでに回す回数の期待値は $1/p$ になる.

注意・警告: 例えば, 当たりが $1\% = 0.01$ の確率で出るガチャを回し続けたときに, 当たりが初めて出るまでに回す回数の期待値は $1/0.01 = 100$ 回になる. これは「だいたい100回程度ガチャを回せばほぼ確実に当たりが出ること」を意味 **しない**. 以下の問題を解いてみよ.

4.25 問題: 当たりが出る確率が1%のガチャを当たりが出るまで回す場合

当たりが1%の確率で出るガチャを引き続けて当たりが出るまで回す回数を N と書く.

(1) N の期待値と分散と標準偏差を求めよ.

(2) 当たりが初めて出るまでの回数が150回以上になる確率と200回以上になる確率と300回以上になる確率を求めよ.

確率や標準偏差の値は有効桁3桁以上が合っていれば正解とする.

解答例: $p = 0.01$ とおく.

(1) N の期待値と分散と標準偏差:

$$E[N] = \frac{1}{p} = 100, \quad \mathrm{var}(N) = \frac{1-p}{p^2} = 9900, \quad \mathrm{std}(N) = \sqrt{\mathrm{var}(N)} \approx 99.5.$$

(2) N が150回以上になる確率 = 最初の149回分がすべて外れの確率:

$$(1-p)^{149} = 0.99^{149} \approx 0.224.$$

N が200回以上になる確率 = 最初の199回分がすべて外れの確率:

$$(1-p)^{199} = 0.99^{199} \approx 0.135.$$

N が300回以上になる確率 = 最初の199回分がすべて外れの確率:

$$(1-p)^{299} = 0.99^{299} \approx 0.0495.$$

解答終

注意・警告続き: 当たりが出る確率が1%のガチャを当たりが出るまで回し続ける人達が沢山いると, 300回以上回してしまう人が5%程度すなわち20人に1人程度生じるということになる. このようなガチャを当たりが出るまで回し続ける場合にあなたは, その20人の1人になってしまうことを覚悟しておくべきだと思われる.

注意: p が小さな幾何分布の標準偏差は

$$\sqrt{\frac{1-p}{p^2}} \approx \sqrt{\frac{1}{p^2}} = \frac{1}{p}$$

となる. 実際, 上の場合も $\mathrm{var}(N)$ はほぼ $1/p = 100$ になっている.

In [34]:

```
p = 0.01
@show 1/p
@show (1 - p)/p^2
@show √((1 - p)/p^2)
for Nmin in (150, 200, 300)
    @eval @show (1 - p)^($Nmin - 1)
end
```

```
1 / p = 100.0
(1 - p) / p ^ 2 = 9900.0
√((1 - p) / p ^ 2) = 99.498743710662
(1 - p) ^ (150 - 1) = 0.2236886739786474
(1 - p) ^ (200 - 1) = 0.13533300490703204
(1 - p) ^ (300 - 1) = 0.04953625663766235
```

In [35]:
```
@show mean(Geometric(0.01)) + 1
@show var(Geometric(0.01))
@show std(Geometric(0.01))
for Nmin in (150, 200, 300)
    @eval @show ccdf(Geometric(0.01), $(Nmin - 1) - 1)
end
```

```
mean(Geometric(0.01)) + 1 = 100.0
var(Geometric(0.01)) = 9900.0
std(Geometric(0.01)) = 99.498743710662
ccdf(Geometric(0.01), 149 - 1) = 0.22368867397864767
ccdf(Geometric(0.01), 199 - 1) = 0.13533300490703226
ccdf(Geometric(0.01), 299 - 1) = 0.049536256637662465
```

Monte Carlo 法による近似計算:

In [36]:
```
N = rand(Geometric(0.01), 10^8) .+ 1
@show mean(N)
@show var(N)
@show std(N)
@show mean(≥(150), N)
@show mean(≥(200), N)
@show mean(≥(300), N);
```

```
mean(N) = 99.99065754
var(N) = 9898.741484625854
std(N) = 99.4924192319488
mean((≥)(150), N) = 0.22367281
mean((≥)(200), N) = 0.1352877
mean((≥)(300), N) = 0.04951668
```

4.26 負の二項展開

任意の数 a と0以上の整数 m について, **二項係数** (binomial coefficients)を

$$\binom{a}{m} = \frac{a(a-1)(a-2)\cdots(a-m+1)}{m!}$$

と定める. たとえば, $m = 0, 1, 2, 3$ の場合には,

$$\binom{a}{0} = 1,\ \binom{a}{1} = a,\ \binom{a}{2} = \frac{a(a-1)}{2},\ \binom{a}{3} = \frac{a(a-1)(a-2)}{3!}.$$

$a = n$ も0以上の整数のとき, これは組み合わせの数と一致する:

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}.$$

$f(x) = (1+x)^a$ について,

$$f'(x) = a(1+x)^{a-1},\ f''(x) = a(a-1)(1+x)^{a-2},\ \dots\ f^{(m)}(x) = a(a-1)\cdots(a-m+1)(1+x)^{a-m}$$

なので,

$$f^{(m)}(0) = a(a-1)\cdots(a-m+1), \quad \frac{1}{m!}f^{(m)}(0) = \binom{a}{m}.$$

ゆえに, $f(x) = (1+x)^a$ は $x = 0$ で次のようにTaylor展開される:

$$(1+x)^a = f(x) = \sum_{m=0}^{\infty} \frac{1}{m!} f^{(m)}(0) x^m = \sum_{m=0}^{\infty} \binom{a}{m} x^m = 1 + ax + \frac{a(a-1)}{2}x^2 + \frac{a(a-1)(a-2)}{3!}x^3 + \cdots$$

このTaylor展開は $|x| < 1$ で常に収束している. $a = n$ が0以上の整数の場合には $m = 0, 1, \dots, n$ に関する有限和になり, 通常の二項定理が得られる. このTaylor展開を **二項展開** (binomial expansion)と呼ぶ.

次の形の二項係数を **負の二項係数** と呼ぶことにする:

$$(-1)^m \binom{-a}{m} = (-1)^m \frac{(-a)(-a-1)\cdots(-a-m+1)}{m!} = \frac{a(a+1)\cdots(a+m-1)}{m!} = \binom{a+m-1}{m}.$$

上の二項展開の x, a を $-x, -a$ で置き換えることによって, $|x| < 1$ のとき,

$$(1-x)^{-a} = \sum_{m=0}^{\infty}(-1)^m\binom{-a}{m}x^m = \sum_{m=0}^{\infty}\binom{a+m-1}{m}x^m.$$

この公式を **負の二項展開** と呼ぶことにする.

4.27 負の二項分布の定義

$0 < p \leq 1$ であると仮定し, k は正の整数であるとし, $m = 0, 1, 2, \ldots$ であるとする.

確率 p で 1 が確率 $1-p$ で 0 が生じるようなBernoulli試行を続けたとき, 1 が合計でちょうど k 個出たときの試行回数が $k+m$ になる確率は, 最初の $k+m-1$ 回中の 0 の個数が m 個で最後の $m+k$ 回目に 1 が出る確率

$$\binom{k+m-1}{m}p^{k-1}(1-p)^m \times p$$

に等しいので,

$$P(m) = \binom{k+m-1}{m}p^k(1-p)^m = (-1)^m\binom{-k}{m}p^k(1-p)^m$$

になる. この確率質量函数で定義される可算集合 $\{0, 1, 2, 3, \ldots\}$ 上の無限離散分布を **負の二項分布** (negative binomial distribution)と呼び, 次のように表す:

$$\mathrm{NegativeBinomial}(k, p).$$

負の二項分布における確率の総和が 1 になることは, 前節の負の二項展開からただちに得られる:

$$\sum_{m=0}^{\infty}P(m) = p^k\sum_{m=0}^{\infty}\binom{k+m-1}{m}(1-p)^m = p^k(1-(1-p))^{-k} = p^kp^{-k} = 1.$$

注意: 以上および以下の計算は k が正の整数でなくても,

$$\binom{k+m-1}{m} = \frac{k(k+1)\cdots(k+m-1)}{m!} > 0 \quad (k > 0,\ m = 0, 1, 2, 3, \ldots)$$

なので, $k > 0$ ならば確率質量函数

$$P(m) = \binom{k+m-1}{m}p^k(1-p)^m = (-1)^m\binom{-k}{m}p^k(1-p)^m \quad (m = 0, 1, 2, 3, \ldots)$$

は定義されている. 期待値と分散の計算も上で計算した結果と同じになる. このように拡張された離散分布も負の二項分布と呼ばれる.

4.28 負の二項分布の期待値と分散

M は負の二項分布 $\mathrm{NegativeBinomial}(k, p)$ に従う確率変数であるとし, $N = M + k$ とおく.

M は成功確率 p のBernoulli試行をちょうど k 回 1 が出るまで繰り返すときに出た 0 の個数を意味する確率変数にになる.

N はちょうど k 回 1 が出たときの試行回数を意味する確率変数であり, その確率質量函数は

$$P(n-k) = \binom{n-1}{n-k}p^k(1-p)^{n-k} = \binom{n-1}{k-1}p^k(1-p)^{n-k}$$

になる(この式の形は二項分布の確率質量函数に近いが少しだけ違っている).

それらの期待値と分散は次のようになる:

$$E[M] = \frac{k(1-p)}{p}, \quad E[N] = E[M] + k = \frac{k}{p}, \quad \mathrm{var}(M) = \mathrm{var}(N) = \frac{k(1-p)}{p^2}.$$

これらは以下のようにして示される. 二項係数について

$$
\begin{aligned}
m\binom{a+m-1}{m} &= m\frac{a(a+1)\cdots(a+m-1)}{m!} \\
&= a\frac{(a+1)\cdots(a-m+1)}{(m-1)!} = a\binom{a+m-1}{m-1}, \\
m(m-1)\binom{a+m-1}{m} &= m(m-1)\frac{a(a+1)(a+2)\cdots(a+m-1)}{m!} \\
&= a(a+1)\frac{(a+2)\cdots(a+m-1)}{(m-2)!} = a(a+1)\binom{a+m-1}{m-2}
\end{aligned}
$$

が成立することとと負の二項展開を使うと, 負の二項分布に従う確率変数 M の期待値と分散は以下のように計算される:

$$
\begin{aligned}
E[M] &= \sum_{m=0}^{\infty} mP(m) = p^k \sum_{m=1}^{\infty} m\binom{k+m-1}{m}(1-p)^m = p^k \sum_{m=1}^{\infty} k\binom{k+m-1}{m-1}(1-p)^m \\
&= p^k k(1-p) \sum_{m=1}^{\infty} \binom{(k+1)+(m-1)-1}{m-1}(1-p)^{m-1} = p^k k(1-p)(1-(1-p))^{-(k+1)} \\
&= p^k k(1-p)p^{-(k+1)} = \frac{k(1-p)}{p}, \\
E[M(M-1)] &= \sum_{m=0}^{\infty} m(m-1)P(m) = p^k \sum_{m=2}^{\infty} m(m-1)\binom{k+m-1}{m}(1-p)^m = p^k \sum_{m=2}^{\infty} k(k+1)\binom{k+m-1}{m-2}(1-p) \\
&= p^k k(k+1)(1-p)^2 \sum_{m=1}^{\infty} \binom{(k+2)+(m-2)-1}{m-2}(1-p)^{m-2} = p^k k(k+1)(1-p)^2(1-(1-p))^{-(k+2)} \\
&= p^k k(k+1)(1-p)^2 p^{-(k+2)} = \frac{k(k+1)(1-p)^2}{p^2}, \\
E[M(M-1)] &= E[M^2] - E[M], \\
E[M^2] &= E[M(M-1)] + E[M] = \frac{k(k+1)(1-p)^2}{p^2} + \frac{k(1-p)}{p}, \\
\mathrm{var}(M) &= E[M^2] - E[M]^2 = \frac{k(k+1)(1-p)^2}{p^2} + \frac{k(1-p)}{p} - \frac{k^2(1-p)^2}{p^2} \\
&= \frac{-k(1-p)^2}{p^2} + \frac{kp(1-p)}{p^2} = \frac{k(1-p)}{p^2}.
\end{aligned}
$$

上の結果より, $N = M + k$ の期待値と分散は次のようになる:

$$
E[N] = E[M] + k = \frac{k}{p}, \quad \mathrm{var}(N) = \mathrm{var}(M) = \frac{k(1-p)}{p^2}.
$$

注意: 負の二項分布に従う確率変数の期待値と分散は $k = 1$ の場合の幾何分布の期待値と分散のちょうど k 倍になっている. これは偶然ではなく, 必然的にこうなる理由がある. この点については後で確率変数の独立性について説明してから詳しく説明する.

In [37]:

```
@vars k p
k*(k+1)*(1-p)^2/p^2 + k*(1-p)/p - k^2*(1-p)^2/p^2 ▷ simplify
```

Out[37]: $\frac{k(1-p)}{p^2}$

In [38]:

```
k, p = 4, 0.25
negbin = NegativeBinomial(k, p)
@show mean(negbin), k*(1-p)/p
@show mean(negbin) + k, k/p
@show var(negbin), k*(1-p)/p^2;
```

```
(mean(negbin), (k * (1 - p)) / p) = (12.0, 12.0)
(mean(negbin) + k, k / p) = (16.0, 16.0)
(var(negbin), (k * (1 - p)) / p ^ 2) = (48.0, 48.0)
```

In [39]:

```
negbin = NegativeBinomial(10, 0.5)
μ, σ = mean(negbin), std(negbin)
x = 0:round(Int, μ + 4σ)
bar(x, x -> pdf(negbin, x); label="",
    title="NegativeBinomial(10, 0.5)", alpha=0.5)
plot!(; xtick=0:5:maximum(x)+5)
```

Out[39]:

M は負の二項分布 $\mathrm{NegativeBinomial}(k, p)$ に従う確率変数であるとし, $N = M + k$ とおく. L は正の整数であるとし, $\theta = 1/(Lp)$, $T = N/L$ とおくと, L を大きくしたとき, T の分布はガンマ分布 $\mathrm{Gamma}(k, \theta)$ で近似される.

In [40]:

```
k, θ = 5, 2
L = 10
p = 1/(L*θ)

negbin = NegativeBinomial(k, p)
μ, σ = mean(negbin), std(negbin)
m = 0:round(Int, μ + 5σ)
n = m .+ k
t = n ./ L
plot(t, pdf.(negbin, m)*L; label="(NegBin($k,$p)+$k)/$L")
plot!(t, pdf.(Gamma(k, θ), t); label="Gamma($k,$θ)", ls=:dash)
```

Out[40]:

In [41]:

```
k, θ = 5, 2
L = 100
p = 1/(L*θ)

negbin = NegativeBinomial(k, p)
μ, σ = mean(negbin), std(negbin)
m = 0:round(Int, μ + 5σ)
n = m .+ k
t = n ./ L
plot(t, pdf.(negbin, m)*L; label="(NegBin($k,$p)+$k)/$L")
plot!(t, pdf.(Gamma(k, θ), t); label="Gamma($k,$θ)", ls=:dash)
```

Out[41]:

k が大きなとき, 負の二項分布 $\mathrm{NegativeBinomial}(k, p)$ は正規分布で近似される.

In [42]:

```
k, p = 100, 0.4

negbin = NegativeBinomial(k, p)
μ, σ = mean(negbin), std(negbin)
m = round(Int, μ - 4σ):round(Int, μ + 4σ)
n = m .+ k
plot(n, pdf.(negbin, m); label="NegBin($k,$p)+$k")
plot!(Normal(μ+k, σ); label="normal approx.", ls=:dash)
plot!(; legend=:outertop)
```

Out[42]:

In [43]:

```
k, p = 1000, 0.4

negbin = NegativeBinomial(k, p)
μ, σ = mean(negbin), std(negbin)
m = round(Int, μ - 4σ):round(Int, μ + 4σ)
n = m .+ k
plot(n, pdf.(negbin, m); label="NegBin($k,$p)+$k")
plot!(Normal(μ+k, σ); label="normal approx.", ls=:dash)
plot!(; legend=:outertop)
```

Out[43]:

4.29 問題: 当たりが出る確率が5%のガチャを当たりが5回出るまで回す場合

当たりが $p = 5\% = 0.05$ で出るガチャを当たりがちょうど $k = 5$ 回出るまでに回した回数を N と書くことにする. N は負の二項分布 $\mathrm{NegativeBinomial}(k, p)$ に従う確率変数 M によって $N = M + k$ と書ける.

(1) N の期待値と分散と標準偏差を求めよ.

(2) N が150回以上, 200回以上になる確率を求めよ.

確率や標準偏差の値は有効桁3桁以上が合っていれば正解とする.

解答例: $p = 0.05$, $k = 5$ とおく.

(1) N の期待値と分散と標準偏差:

$$E[N] = \frac{k}{p} = 100, \quad \mathrm{var}(N) = \frac{k(1-p)}{p^2} = 1900, \quad \mathrm{std}(N) = \sqrt{\mathrm{var}(N)} \approx 43.58.$$

(2) ($N \geq 150$ となる確率) = 1 − ($N \leq 149$ となる確率):

$$1 - \sum_{n=1}^{149} \binom{n-1}{k-1} p^k (1-p)^{n-k} \approx 0.129.$$

($N \geq 200$ となる確率) = 1 − ($N \leq 199$ となる確率):

$$1 - \sum_{n=1}^{199} \binom{n-1}{k-1} p^k (1-p)^{n-k} \approx 0.0273.$$

解答終

In [44]:

```
p = 5//100
k = 5
@show 5/p 5(1 - p)/p^2 √(5(1 - p)/p^2);
```

```
5 / p = 100//1
(5 * (1 - p)) / p ^ 2 = 1900//1
√((5 * (1 - p)) / p ^ 2) = 43.58898943540674
```

In [45]:
```
p = 0.05
k = 5
for Nmin in (150, 200)
    @show 1 - sum(binomial(n-1, k-1)*p^k*(1-p)^(n-k) for n in 1:Nmin-1)
end
```

```
1 - sum((binomial(n - 1, k - 1) * p ^ k * (1 - p) ^ (n - k) for n = 1:Nmin - 1)) = 0.12922049010
314218
1 - sum((binomial(n - 1, k - 1) * p ^ k * (1 - p) ^ (n - k) for n = 1:Nmin - 1)) = 0.02734419238
495378
```

In [46]:
```
p = 0.05
k = 5
@eval @show negbin = NegativeBinomial($k, $p)
@show mean(negbin) + k
@show var(negbin)
@show std(negbin)
for Nmin in (150, 200)
    @show ccdf(negbin, (Nmin - 1) - k)
end
```

```
negbin = NegativeBinomial(5, 0.05) = NegativeBinomial{Float64}(r=5.0, p=0.05)
mean(negbin) + k = 100.0
var(negbin) = 1899.9999999999995
std(negbin) = 43.58898943540674
ccdf(negbin, (Nmin - 1) - k) = 0.12922049010313866
ccdf(negbin, (Nmin - 1) - k) = 0.027344192384949475
```

Monte Carlo 法による近似計算:

In [47]:
```
M = rand(negbin, 10^8) .+ k
@show mean(N)
@show var(N)
@show std(N)
@show mean(≥(150), N)
@show mean(≥(200), N);
```

```
mean(N) = 99.99065754
var(N) = 9898.741484625854
std(N) = 99.4924192319488
mean((≥)(150), N) = 0.22367281
mean((≥)(200), N) = 0.1352877
```

4.30 多項分布の定義

Bernoulli試行から二項分布が得られたのと同じように, カテゴリカル分布の同分布独立試行から多項分布が得られる.

次のカテゴリカル分布を考える:

$$\mathrm{Categorical}(p_1, \ldots, p_r) = \begin{array}{|c|cccc|} \hline x & 1 & 2 & \cdots & r \\ \hline P(x) & p_1 & p_2 & \cdots & p_r \\ \hline \end{array}$$

このカテゴリカル分布にしたがって, 番号の $1, \ldots, r$ のどれかをランダムに生成することを独立に n 回繰り返すと, 番号 $1, \ldots, r$ からなる長さ n の数列ができる. 総和が n になる0以上の整数達の組 $(k_1, \ldots, k_r)$ が与えられたとき, その数列中に番号 $1, \ldots, r$ がそれぞれ $k_1, \ldots, k_r$ 個含まれている確率は

$$P(k_1, \ldots, k_r) = \frac{n!}{k_1! \cdots k_r!} p_1^{k_1} \cdots p_r^{k_r}$$

になる. この確率質量函数によって定義された有限集合

$$\{, (k_1, \ldots, k_r) \in \mathbb{Z}_{\geq 0}^r \mid k_1 + \cdots + k_r \}$$

上の多変量離散分布を多項分布と呼び,

$$\mathrm{Multinomial}(n, p_1, \ldots, p_r)$$

と表す. $r = 2$ の場合の多項分布は本質的に二項分布に等しい.

5 連続分布など

5.1 連続分布と確率密度函数

$a, b \in \mathbb{R} \cup \{\pm\infty\}$, $a < b$ であるとし, 区間 $(a, b) = \{\, x \in \mathbb{R} \mid a < x < b \,\}$ を考える. ($a = -\infty$ や $b = \infty$ となっている場合があるので注意せよ.) このとき, 実数 $x \in (a, b)$ を0以上の実数に対応させる函数 $p(x)$ で

$$\int_a^b p(x)\,dx = 1$$

を満たすものを区間 (a, b) 上の **確率密度函数** (probability density function, pdf)もしくは単に **密度函数** (density function)と呼ぶ.

密度函数の定義域を実数全体に拡張する場合には区間 (a, b) の外での値は 0 にする.

区間 (a, b) 上の確率密度函数が与えられているとき, 区間 (a, b) 上の **連続確率分布** もしくは **連続分布** が与えられているいう. (区間の両端を区間に含めても含めなくてもよいが, 連続分布の場合にはその違いを気にする必要はない. なぜならば, 連続分布において一点の確率は0だからである.)

確率分布を単に分布と呼ぶことがよくある.

1次元の区間を n 次元の $\mathbb{R}^n$ における領域に一般化して, n 変量の連続分布を定義することもできる. その場合には上の積分は n 重積分になる. 多変量連続分布については後で具体的な例を通して説明することにする.

5.2 一様分布の定義

区間 (a, b) 上の確率密度函数

$$p(x) = \frac{1}{b-a} \quad (a < x < b)$$

によって与えられる連続分布を区間 (a, b) 上の一様分布と呼び, 次のように表す:

$$\mathrm{Uniform}(a, b).$$

半開区間 $[a, b)$, $(a, b]$ や閉区間 $[a, b]$ 上の一様分布も同様に定義されるが, 区間の両端を含めるか否かは本質的ではない.

コンピュータでの `rand()` 函数は近似的に区間 $[0, 1)$ 上の一様分布に従う確率変数とみなされる.

一様分布 $\mathrm{Uniform}(0, 1)$ の確率密度函数のグラフは以下のようになる.

In [48]:
```
plot(Uniform(0, 1), -0.4, 1.4; label="", title="Uniform(0, 1)")
```

Out[48]:

5.3 正規分布の定義

$\mu \in \mathbb{R}$, $\sigma \in \mathbb{R}$, $\sigma > 0$ と仮定する. 実数直線 $\mathbb{R} = (-\infty, \infty)$ 上の確率密度函数

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

によって与えられる連続分布を平均 μ, 分散 σ^2 の **正規分布** (normal distribution)と呼び,

$$\mathrm{Normal}(\mu, \sigma)$$

と表す. パラメータ σ は標準偏差と呼ばれる.

特に平均 0, 分散 1 の正規分布を **標準正規分布** (standard normal distribution)と呼び,

$$\mathrm{Normal}() = \mathrm{Normal}(0, 1)$$

と表すことにする.

分布のパラメータとしての平均 μ や分散 σ^2 と後で出て来る標本の平均や分散を概念的に厳密に区別する必要がある.

互いに密接に関係はしているが異なる概念を同じ用語で呼ぶことが原因で初学者は大抵の場合混乱する.

混乱せずに済ますためには 「平均」や「分散」という用語を使う場合には必ず「○○分布の平均と分散」や「確率変数○○の平均と分散」や「標本○○の平均と分散」のように **「〜の平均と分散」という言い方をすればよいだろう.**

正規分布の確率密度函数のグラフは以下の形になる. この形は「釣鐘型」(bell-shaped)と表現されることが多い.

In [49]:
```
plot(Normal(2, 3), 2-4*3, 2+4*3; label="", title="Normal(2, 3)", xtick=-100:2:100)
```

Out[49]:

In [50]:
```
plot(Normal(), -4, 4; label="", title="Normal()", xtick=-100:100)
```

Out[50]:

標準正規分布において, 値が -1 以上 1 以下になる確率, -2 以上 2 以下になる確率, -1.96 以上 1.96 以下になる確率はそれぞれ以下のようになる.

In [51]:
```
@show cdf(Normal(), 1) - cdf(Normal(), -1)
plot(Normal(), -4, 4; label="", title="Normal()", xtick=-100:100)
plot!(Normal(), -1, 1; label="", c=1, fillrange=0, fillcolor=:red, fillalpha=0.3)
```

```
cdf(Normal(), 1) - cdf(Normal(), -1) = 0.6826894921370861
```

Out[51]:

In [52]:
```
@show cdf(Normal(), 2) - cdf(Normal(), -2)
plot(Normal(), -4, 4; label="", title="Normal()", xtick=-100:100)
plot!(Normal(), -2, 2; label="", c=1, fillrange=0, fillcolor=:red, fillalpha=0.3)
```

```
cdf(Normal(), 2) - cdf(Normal(), -2) = 0.9544997361036416
```

Out[52]:

In [53]:
```
@show cdf(Normal(), 1.96) - cdf(Normal(), -1.96)
plot(Normal(), -4, 4; label="", title="Normal()", xtick=-100:100)
plot!(Normal(), -1.96, 1.96; label="", c=1, fillrange=0, fillcolor=:red, fillalpha=0.3)
```

```
cdf(Normal(), 1.96) - cdf(Normal(), -1.96) = 0.950004209703559
```

Out[53]:

5.4 連続確率変数

区間 (a, b) 上の確率密度函数 $p(x)$ が与えられているとする.

さらに, 区間 (a, b) 上の実数値函数を **連続分布における確率変数** と呼ぶ. (積分が定義できるような良い函数に限ることにする. 複素数値, ベクトル値, 行列値の連続確率変数を考えることもある.)

たとえば, $x \in (a, b)$ をそれ自身 $x \in \mathbb{R}$ に対応させる函数は確率変数とみなされる. 以下ではそれを X と書き, 確率密度函数 $p(x)$ を持つ **連続確率変数** と呼ぶ(確率密度函数を持つ確率変数を連続確率変数と呼ぶ). そのとき,

$$X \sim p(x)$$

と書くことがある. さらに, 確率密度函数 $p(x)$ で与えられる連続分布の名前が Dist のとき, X は分布 Dist に従う確率変数であると言い(「従う」は "follow" の訳語), 以下のように書く:

$$X \sim \mathrm{Dist}.$$

5.5 連続確率変数に関する確率とその期待値

値が $a \leq c \leq d \leq b$ であると仮定する.

確率密度函数 $p(x)$ を持つ確率変数 X が c より大きく d より小さくなる確率 $P(c < X < d)$ を積分を使って

$$P(c < X < d) = \int_c^d p(x)\,dx$$

と定義する. c 以上 d 以下の確率 $P(c \leq X \leq d)$ なども同じ積分の値で定義する:

$$P(c \le X \le d) = \int_c^d p(x)\,dx.$$

特に X の値がぴったり c に等しくなる確率 $P(X=c)$ は次のように 0 になる:

$$P(X=c) = P(c \le X \le c) = \int_c^c p(x)\,dx = 0.$$

$x \in (a,b)$ を含む長さ dx の微小区間 $I = (x - t\,dx, x + u\,dx) \subset (a,b)$ $(t,u \ge 0, t+u=1)$ に値が含まれる確率は

$$P(X \in I) = \int_{x-t\,dx}^{x+u\,dx} p(x')\,dx' \approx p(x)\,dx$$

のように近似される. これが $p(x)$ が確率の密度であることの意味である.

確率変数 $f(X)$ の期待値 $E[f(X)]$ が次のように定義される:

$$E[f(X)] = \int_a^b f(x)p(x)\,dx.$$

ただし, これは $\int_a^b |f(x)|p(x)\,dx$ が有限の値に収束する場合(積分が絶対収束する場合)にのみ定義されていると考える.

X の従う分布の名前が Dist のとき $E[X]$ を **分布** Dist **の期待値** と呼ぶ.

期待値を **平均** または **平均値** と呼ぶこともあるが, あとで定義する標本の平均値と確率変数の平均＝期待値を区別する必要があるので注意が必要である.

確率変数 $1_{c \le f(X) \le d}(X)$ を

$$1_{c \le f(X) \le d}(x) = \begin{cases} 1 & (c \le x \le d) \\ 0 & (\text{otherwise}) \end{cases}$$

と定めると,

$$E[1_{c \le f(X) \le d}(X)] = \int_a^b 1_{c \le f(X) \le d}(x)p(x)\,dx = \int_c^d p(x)\,dx = P(c \le X \le d).$$

このように期待値の概念は確率の概念を含んでいると考えてよい.

再度強調: 確率変数の定義を期待値が定義されている変数(のようなもの)のことだと思ってもよい と言ってよいくらい, 期待値の概念は確率変数について基本的である. 統計学に必要な理論的な計算の多くが期待値を取る操作 $E[\ ]$ の計算に帰着されることになる.

5.6 確率変数の確率密度函数

確率変数 $Y = f(X)$ を考える. ある y の確率密度函数 $p_Y(y)$ が存在して, Y の函数達 $g(Y)$ について

$$E[g(Y)] = \int g(y)\,p_Y(y)\,dy$$

と書けるとき, Y は **確率密度函数** $p_Y(y)$ **を持つ** という. 右辺は定積分であることに注意せよ. 積分区間は場合ごとに異なるので表記を省略した. この説明の曖昧さを除くために例を使って説明しよう.

例: X は一様分布 $\mathrm{Uniform}(0,2)$ に従う確率変数であるとし, $Y = X^2$ とおく. このとき, 積分変数を $x = \sqrt{y}$ によって変換すると, $dx = y^{-1/2}\,dy/2$ なので

$$E[g(Y)] = \frac{1}{2}\int_0^2 g(x^2)\,dx = \int_0^{\sqrt{2}} g(y)\,\frac{1}{4}y^{-1/2}\,dy.$$

ゆえに $Y = X^2$ は確率密度函数

$$p_Y(y) = \frac{1}{4}y^{-1/2} \quad \left(0 < y < \sqrt{2}\right)$$

を持つ.

例: Z は標準正規分布に従う確率変数であるとし, $Y = Z^2$ とおく. このとき z^2 が z の偶函数であることと, 積分の変数変換 $z = \sqrt{y}$ によって,

$$E[g(Y)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(z^2)\, e^{-z^2/2}\, dz = \frac{2}{\sqrt{2\pi}} \int_{0}^{\infty} g(z^2)\, e^{-z^2/2}\, dz = \frac{1}{\sqrt{2\pi}} \int_{0}^{\infty} g(y)\, e^{-y} y^{-1/2}\, dy.$$

ゆえに $Y = Z^2$ は確率密度函数

$$p_Y(y) = \frac{1}{\sqrt{2\pi}} e^{-y} y^{-1/2} \quad (y > 0)$$

を持つ. この確率密度函数で定義される連続分布は **自由度1のχ²分布** (カイ二乗分布)と呼ばれている.

記号法上の注意: 以上における $p_Y(y)$ を単に $p(y)$ と書いて, x と y の変数名の違いによって, X の確率密度函数 $p(x)$ と区別することもある. そういうスタイルは初学者の混乱の原因になりえるが, 手計算をするときには非常に便利であり, 無駄な添え字を付けることを無くせるので結構便利である. 実際によく使われるので, 文献を読むときには知っておかないと困る知識でもある.

5.7 確率変数の確率質量函数

連続確率変数 X の函数 $Y = f(X)$ が離散確率変数になる場合もあることにも注意せよ.

$Y = f(X)$ の取り得る値が離散的な $\{b_1, b_2, b_3, \ldots\}$ に限り, ある確率質量函数 $P_Y(y)$ が存在して,

$$E[g(Y)] = \sum_i g(b_i) P_Y(b_i)$$

が成立するとき, $Y = f(X)$ は **確率質量函数** $P_Y(y)$ **を持つ** という.

例: T は一様分布 $\mathrm{Uniform}(0,1)$ に従う連続確率変数であるとし, $X = f(T)$ を

$$f(t) = \begin{cases} 1 & (t < 0.3) \\ 0 & (t \geq 0.3) \end{cases}$$

によって定義する. このとき,

$$E[g(X)] = \int_0^1 g(f(t))\, dt = \int_0^{0.3} g(1)\, dt + \int_{0.3}^1 g(0)\, dt = g(1) \cdot 0.3 + g(0) \cdot 0.7.$$

ゆえに $X = f(T)$ は確率質量函数

$$P_X(1) = 0.3, \quad P_X(0) = 0.7$$

を持つ. この例は一様乱数からBernoulli試行を作るときにすでに登場している.

5.8 連続分布における確率変数の期待値を取る操作の基本性質

連続確率変数 X の函数 $f(X)$ の期待値を取る操作(期待値汎函数) $E[f(X)]$ は以下の性質を満たしている.

- 線形性: $E[\alpha f(X) + \beta g(X)] = \alpha E[f(X)] + \beta E[g(X)]$,
- 単調性: 任意の $x = a_i$ について $f(x) \leq g(x)$ ならば $E[f(X)] \leq E[g(X)]$,
- 規格化: $E[\alpha] = \alpha$. (特に $E[1] = 1$).

この性質はすでに説明した離散確率変数の基本性質と完全に同じ形をしており, 離散の場合の和を積分に置き換えれば全く同じ方法で証明される.

5.9 問題: Markovの不等式とChebyshevの不等式

X は確率密度函数または確率質量函数を持つ確率変数であるとする.

$a, t > 0$ のとき, 確率変数 $Y = f(X)$ について次が成立することを示せ:

$$P(|Y| \geq t) \leq \frac{1}{t^a} E[|Y|^a].$$

この不等式の $a = 1$ の場合を **Markovの不等式** と呼び, $a = 2$ の場合を **Chebyshevの不等式** と呼ぶ.

解答例: 確率変数 $1_{|Y| \geq t}(X)$ を

$$1_{|Y| \geq t}(x) = \begin{cases} 1 & (|f(x)| \geq t) \\ 0 & (\text{otherwise}) \end{cases}$$

と定めると,

$$P(|Y| \geq t) = E[1_{|Y|\geq t}(X)]$$

となるのであった. $|f(x)| \geq t$ ならば $1_{|Y|\geq t}(x) = 1$ となるので,

$$|f(x)|^a \geq t^a = t^a 1_{|Y|\geq t}(x)$$

となり, $|f(x)| < t$ ならば $1_{|Y|\geq t}(x) = 0$ となるので

$$|f(x)|^a \geq 0 = t^a 1_{|Y|\geq t}(x)$$

が自明に成立する. ゆえに期待値を取る操作の単調性と線形性より,

$$E[|Y|^a] \geq t^a E[1_{|Y|\geq t}(X)] = t^a P(|Y| \geq t).$$

両辺を t^a で割れば示したい不等式が得られる.

解答終

5.10 問題: Jensenの不等式

$y = f(x)$ は上に凸な函数であるとする. ある一次函数 $y = a(x-\mu) + f(\mu)$ が存在して(この一次函数は直観的には $y = f(x)$ の接線である), $f(x) \leq a(x-\mu) + f(\mu)$ という不等式が成立することを認めて, 確率変数 X について以下が成立することを示せ:

(1) $f(x)$ が上に凸な函数ならば $E[f(X)] \leq f(E[X])$

(2) $f(x)$ が下に凸な函数ならば $E[f(X)] \geq f(E[X])$

これらを **Jensenの不等式** (イェンセンの不等式)と呼ぶ.

解答例: $f(x)$ は上に凸な函数であるとし, $\mu = E[X]$ とおく. ある一次函数 $y = a(x-\mu) + f(\mu)$ が存在して

$$f(x) \leq a(x-\mu) + f(\mu)$$

という不等式が成立する. この両辺の x に確率変数 X を代入して期待値を取ると, 期待値を取る操作の単調性より,

$$E[f(X)] \leq E[a(X-\mu) + f(\mu)].$$

右辺に期待値を取る操作の線形性と規格化条件を適用し, $\mu = E[X]$ を代入すると,

$$E[f(X)] \leq a(E[X]-\mu) + f(\mu) = f(E[X]).$$

これで(1)が示された. $f(x)$ が下に凸な場合には $-f(x)$ が上に凸になるので, (1)から(2)が導かれることがわかる.

解答終

注意: 以上のように, Jensenの不等式は期待値を取る操作の基本性質をちょうど全部使えば証明できる. 期待値を取る操作の基本性質が十分に強力であることがこの証明から分かる.

5.11 問題: 相加相乗平均の不等式

Jensenの不等式を使って, $a_1, \ldots, a_n > 0$ のとき, 次が成立することを示せ:

$$(a_1 \cdots a_n)^{1/n} \leq \frac{a_1 + \cdots + a_n}{n}.$$

解答例: $E[f(X)]$ を

$$E[f(X)] = \frac{f(a_1) + \cdots + f(a_n)}{n}$$

と定めると, $E[\ ]$ は期待値を取る操作の基本性質をすべて満たしており, $f(x) = \log x$ のとき

$$E[\log X] = \frac{\log a_1 + \cdots + \log a_n}{n} = \log(a_1 \cdots a_n)^{1/n}, \quad E[X] = \frac{a_1 + \cdots + a_n}{n}.$$

ゆえに $f(x) = \log x$ (これは上に凸)に関するJensenの不等式 $E[\log X] \leq \log E[X]$ より,

$$\log(a_1 \cdots a_n)^{1/n} \leq \log \frac{a_1 + \cdots + a_n}{n}.$$

$\log x$ 単調増加函数なので, $(a_1 \cdots a_n)^{1/n} \leq (a_1 + \cdots + a_n)/n$ が得られる.

解答終

5.12 問題: Kullback-Leibler情報量とGibbsの情報不等式

2つの確率密度函数 $p(x)$, $q(x)$ に対して, **Kullback-Leibler情報量** $D(q||p)$ を

$$D(q||p) = \int q(x) \log \frac{q(x)}{p(x)}$$

と定める. このとき

$$D(q||p) \geq 0$$

となることを示せ. この不等式を **Gibbsの情報不等式** と呼ぶ.

解答例: $x > 0$ について, $f(x) = x \log x$ とおくと, $f'(x) = \log x + 1$, $f''(x) = 1/x > 0$ なので, $f(x)$ は下に凸な函数になる. 確率密度函数 $p(x)$ で定義される分布に従う確率変数を X と書き, $Y = q(x)/p(x)$ とおく. このとき,

$$\begin{aligned} E[Y] &= \int \frac{q(x)}{p(x)}\, p(x)\, dx = \int q(x)\, dx = 1, \\ E[f(Y)] &= \int \frac{q(x)}{p(x)} \log \frac{q(x)}{p(x)}\, p(x)\, dx = \int q(x) \log \frac{q(x)}{p(x)}\, dx = D(q||p). \end{aligned}$$

下の凸な函数 $f(x)$ に関するJensenの不等式より,

$$E[f(Y)] \geq f(E[Y]) = f(1) = 1 \log 1 = 0.$$

以上を合わせるとGibbsの情報不等式 $D(q||p) \geq 0$ が得られる.

解答終

注意: Kullback-Leibler情報量は2つの確率質量函数 $P(x)$, $Q(x)$ についても同様に

$$D(q||p) = \sum_x Q(x) \log \frac{Q(x)}{Q(x)}$$

と定義され, 上の証明における積分を和で置き換えれば, 全く同様にしてGibbsの情報不等式を示せる.

注意: Kullback-Leibler情報量は2つの確率分布の違いの大きさを測る指標として使われるので, Gibbsの情報不等式は基本的である. ($D(q||q) = 0$ となることにも注意せよ.) Kullback-Leibler情報量の **Sanovの定理** より, Kullback-Leibler情報量 $D(q||p)$ は分布 p による分布 q のMonte Carloシミュレーションの誤差の大きさの指標の1つになっていることがわかる. シミュレートする側の p とされる側の q の立場は非対称なので $D(q||p)$ が p と q の交換について不変でないことは自然である. Kullback-Leibler情報量のこの性質は以下の2つの異なる使い方をされることがある:

(1) 未知の真の分布 q がそのモデル化とみなせる分布 p でどれだけ近似されているかを測るための指標として使う. (汎化誤差の概念はこのようにして得られる. この考え方はモデル選択における情報量規準の理論の基礎になっている.)

(2) 求めるには大量の計算が必要な分布 p を計算量が少なくてすむ特殊な形の分布 q で近似したいことがある. そのときに, 特殊な形の分布 q で分布 p から(Monte Carloシミュレーションの意味で)最も出て来やすい分布を選ぶためにはKullback-Leibler情報量 $D(q||p) を最小化すればよい. (ベイズ統計における変分推論はこの典型例になっている.)

5.13 連続分布における確率変数の分散

連続確率変数 X について, 確率変数 $Y = f(X)$ の期待値を

$$\mu_Y = E[Y] = E[f(X)].$$

と書くとき, $Y = f(X)$ の **分散**(variance) $\mathrm{var}(f(X))$ は次のように定義される:

$$\mathrm{var}(Y) = E\left[(Y - \mu_Y)^2\right] = \int_a^b (f(x) - \mu_Y)^2 p(x)\, dx.$$

これが $\mathrm{var}(Y) = E[Y^2] - E[Y]^2$ を満たすことを離散の場合と完全に同様に証明できる.

連続分布における確率変数 Y の場合もその標準偏差を $\mathrm{std}(Y) = \sqrt{\mathrm{var}(Y)}$ と定める.

X の従う分布の名前が Dist のとき $\mathrm{var}(X)$ を **分布** Dist **の分散** と呼ぶ. 標準偏差についても同様とする.

5.14 問題: 正規分布モデルによる他の分布の最良近似

$p(x)$ が平均 μ, 分散 σ^2 を持つ正規分布の確率密度函数であるとする:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/(2\sigma^2)}.$$

確率密度函数 $q(x)$ が定める分布の平均と分散をそれぞれ μ_0, σ_0^2 と書こう:

$$\mu_0 = \int x\, q(x)\, dx, \quad \sigma_0 = \int (x-\mu_0)^2\, q(x)\, dx.$$

ここで $\int \cdots dx$ は実数直線全体での定積分を表す. このとき, 少し上の方で定義したKullback-Leibler情報量

$$D(q||p) = \int q(x) \log \frac{q(x)}{p(x)}\, dx = \int (-\log p(x)) q(x)\, dx + \int q(x) \log q(x)\, dx$$

は次の形になることを示せ:

$$D(q||p) = \frac{\sigma_0^2 + (\mu-\mu_0)^2}{2\sigma^2} + \log \sigma^2 + \log \sqrt{2\pi} + \int q(x) \log q(x)\, dx.$$

特にこの場合には, 固定された $q(x)$ に対して, $D(q||p)$ を最小にする μ, σ^2 の組み合わせは $\mu = \mu_0$, $\sigma^2 = \sigma_0^2$ になることを示せ.

解答例: $(x-\mu)^2 = ((x-\mu_0) - (\mu-\mu_0))^2 = (x-\mu_0)^2 - 2(\mu-\mu_0)(x-\mu_0) + (\mu-\mu_0)^2$ と正規分布の密度函数 $p(x)$ の定義より,

$$\begin{aligned} -\log p(x) &= \frac{(x-\mu)^2}{2\sigma^2} + \log \sigma^2 + \log \sqrt{2\pi} \\ &= \frac{(x-\mu_0)^2 - 2(\mu-\mu_0)(x-\mu_0) + (\mu-\mu_0)^2}{2\sigma^2} + \frac{1}{2}\log \sigma^2 + \log \sqrt{2\pi} \end{aligned}$$

ゆえに μ_0 と σ_0^2 の定義を使うと,

$$D(q||p) = \frac{\sigma_0^2 + (\mu-\mu_0)^2}{2\sigma^2} + \frac{1}{2}\log \sigma^2 + \log \sqrt{2\pi} + \int q(x) \log q(x)\, dx.$$

任意に固定された $\sigma^2 > 0$ について, これを最小化する μ は $\mu = \mu_0$ である. そのとき,

$$D(q||p) = \frac{1}{2}\left(\frac{\sigma_0^2}{\sigma^2} + \log \sigma^2\right) + (\text{constant of } \sigma^2)$$

これを σ^2 の函数とみなして増減表を書くと, これを最小化する σ^2 は $\sigma^2 = \sigma_0^2$ であることがわかる. 実際, $t = \sigma^2$, $t_0 = \sigma_0^2$, $f(t) = t_0/t + \log t$ とおくと, $f'(t) = (t-t_0)/t^2$ なので,

t	0	$\cdots$	t_0	$\cdots$	∞
$f'(t)$		$-$	0	$+$	
$f(t)$	∞	$\searrow$	$1+\log t_0$	$\nearrow$	∞

と増減表を書ける.

解答終

注意: 以上の結果は, 与えられた分布 $q(x)$ をKullback-Leibler情報量 $D(q||p)$ の最小化の意味で裁量近似する正規分布 $p(x)$ は $q(x)$ と同じ平均と分散を持つ正規分布になることを意味している. これ以後, 分布を定義するごとにその期待値(平均)と分散を計算して行くことになるが, そのときやっていることは本質的に「正規分布による最良近似」の計算だということになる.

5.15 確率変数の一次函数による変換(アフィン変換)の期待値と分散

離散および連続な確率変数 Y と定数 a, b について,

$$E[aY+b] = aE[Y] + b, \quad \mathrm{var}(aY+b) = a^2\, \mathrm{var}(Y)$$

となることを以下のようにして, $E[\ ]$ の基本性質の線形性と規格化条件のみを使って示せる.

$$\begin{aligned} &E[aY+b] = aE[Y] + E[b] = aE[Y] + b. \\ &\mathrm{var}(aY+b) = E[(aY+b-(aE[Y]+b))^2] \\ &\qquad = E[a^2(Y-E[Y])^2] = a^2 E[(Y-E[Y])^2] = a^2\, \mathrm{var}(Y). \end{aligned}$$

5.16 分布のアフィン変換, スケール変換, 平行移動, 逆数の定義

確率変数 Y が分布 D に従うとき, $aY+b$ ($a, b \in \mathbb{R}$, $a \neq 0$) が従う分布を D の **アフィン変換** (affine transformation)と呼び,

$$aD+b$$

と書く. $a>0$ のとき aD を D の **スケール変換** (scale transformation)と呼び, $D+b$ を分布 D の **平行移動** (translation)と呼ぶ.

さらに, $1/Y$ が従う分布を

$$1/D$$

と書き, D の **逆数** (inverse)と呼ぶことにする.

5.17 確率変数の累積分布函数

確率変数 Y に対して

$$F_Y(y) = P(Y \leq y)$$

で定義される実数 y の函数 $F_Y(y)$ を Y もしくは Y が従う確率分布の **累積分布函数** (cumulative distribution function, cdf)と呼ぶ.

Y が連続確率変数で確率密度函数 $p_Y(y)$ を持つとき,

$$F_Y(y) = \int_{-\infty}^{y} p_Y(y)\,dy.$$

ただし, 確率密度函数 $p_Y(y)$ の定義域は拡張した部分での値が 0 になるように実数直線全体に拡張しておく. 連続分布の累積分布函数の導函数は確率密度函数になる.

Y が離散確率変数で取り得る値の集合が $\{b_1, b_2, \ldots\}$ で確率質量函数 $P_Y(y)$ を持つとき,

$$F_Y(y) = \sum_{b_i \leq y} P_Y(b_i).$$

累積分布函数 $F_Y(y)$ は y について(広義)単調増加函数になり, $y \to -\infty$ で 0 に収束し, $y \to \infty$ で 1 に収束する.

さらに, $F_Y(y)$ の定義 $P(Y \leq y)$ において不等号 $\leq$ に等号が含まれていることより, 累積分布函数 $F_Y(y)$ は y について右連続になる. すなわち, y が数直線上で右から b に近付くとき, $F_Y(y)$ は $F_Y(b)$ に近付く:

$$\lim_{y \searrow b} F_Y(y) = F_Y(b).$$

5.18 分位点函数 (quantile function)

確率分布の累積分布函数 $p = F_Y(y)$ の逆函数 $y = Q_Y(p)$ を **分位点函数** (**クォンタイル函数**, quantile function)と呼ぶ. 逆函数が存在しない場合には逆函数の代わりに使える函数を適切に定義して分位点函数と呼んで利用する.

分位点函数が定義されているとき, $Q_2 = Q_Y(1/2)$ は Y の分布の中央値と呼ばれ, $Q_1 = Q_Y(1/4)$, $Q_3 = Q_Y(3/4)$ は Y の分布の第1四分位点, 第3四分位点と呼ばれる(「四分位数」と書いてもよい).

ただし, これらは確率変数もしくはそれが従う分布の中央値や四分位点であり, 標本(サンプル, データ)の中央値や四分位点とは概念的に明瞭に区別しておかなければいけない.

5.19 Bernoulli分布の累積分布函数

Bernoulli分布 $\mathrm{Bernoulli}(p)$ の累積分布函数は次になる:

$$F(x) = \begin{cases} 0 & (x < 0) \\ 1-p & (0 \leq x < 1) \\ 1 & (1 \leq x) \end{cases}$$

In [54]:
```
ber = Bernoulli(0.3)
x = vcat([-1:0.02:-0.02; NaN], [0:0.02:0.98; NaN], 1:0.02:2)
plot(x, x -> cdf(ber, x); label="")
title!("cdf of Bernoulli(0.3)")
plot!(; ytick = 0:0.1:1)
scatter!(0:1, x -> cdf(ber, x - 0.02) ; label="", c=:white, ms=3)
scatter!(0:1, x -> cdf(ber, x) ; label="", c=1, ms=3)
```

Out[54]:

5.20 カテゴリカル分布の累積分布函数

カテゴリカル分布 $\mathrm{Categorical}(0.1, 0.2, 0.3, 0.4)$ の累積分布函数は次になる:

$$F(x) = \begin{cases} 0 & (x < 1) \\ 0.1 & (1 \le x < 2) \\ 0.1 + 0.2 & (2 \le x < 3) \\ 0.1 + 0.2 + 0.3 & (3 \le x < 4) \\ 1 & (0.4 \le x) \end{cases}$$

In [55]:
```
categ = Categorical(0.1, 0.2, 0.3, 0.4)
x = vcat([-0.5:0.02:0.98; NaN], ([k:0.02:k+0.98; NaN] for k in 1:3)..., 4:0.02:5.5)
plot(x, x -> cdf(categ, x); label="")
title!("cdf of Categorical(0.1, 0.2, 0.3, 0.4)")
plot!(; xtick = -1:6, ytick = 0:0.1:1)
scatter!(1:4, x -> cdf(categ, x - 0.02) ; label="", c=:white, ms=3)
scatter!(1:4, x -> cdf(categ, x) ; label="", c=1, ms=3)
```

Out[55]:

5.21 二項分布の累積分布函数のグラフ

In [56]:

```
bin = Binomial(10, 0.4)
x = vcat([-2:0.02:-0.02; NaN], ([k:0.02:k+0.98; NaN] for k in 0:9)..., 10:0.02:12)
plot(x, x -> cdf(bin, x); label="")
title!("cdf of Binomial(10, 0.4)")
plot!(; xtick = -2:12, ytick = 0:0.1:1)
scatter!(0:10, x -> cdf(bin, x - 0.02) ; label="", c=:white, ms=3)
scatter!(0:10, x -> cdf(bin, x) ; label="", c=1, ms=3)
```

Out[56]:

5.22 負の二項分布の累積分布函数のグラフ

In [57]:

```
negbin = NegativeBinomial(4, 0.5)
x = vcat([-2:0.02:-0.02; NaN], ([k:0.02:k+0.98; NaN] for k in 0:13)...)
plot(x, x -> cdf(negbin, x); label="")
title!("cdf of NegativeBinomial(4, 0.5)")
plot!(; xtick = -2:14, ytick = 0:0.1:1)
scatter!(0:14, x -> cdf(negbin, x - 0.02) ; label="", c=:white, ms=3)
scatter!(0:14, x -> cdf(negbin, x) ; label="", c=1, ms=3)
```

Out[57]:

5.23 一様分布の累積分布函数

$a < b$ のとき, 一様分布 $\mathrm{Uniform}(a, b)$ の確率密度函数は

$$p(x) = \begin{cases} 1/(b-a) & (a < x < b) \\ 0 & (\text{otherwise}) \end{cases}$$

である. ゆえに, 一様分布 $\mathrm{Uniform}(a, b)$ の累積分布函数は

$$F(x) = \int_{-\infty}^{x} p(x')\,dx' = \begin{cases} 0 & (x \le a) \\ (x-a)/(b-a) & (a \le x \le b) \\ 1 & (b \le x) \end{cases}$$

になる.

In [58]:
```
plot(x -> cdf(Uniform(-1, 2), x), -2.5, 3.5; label="")
title!("cdf of Uniform(-1, 2)")
yticks!(0:0.1:1)
```

Out[58]:

5.24 標準正規分布の累積分布函数と分位点函数

誤差函数が

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-u^2)\, du$$

と定義される. この函数はコンピュータでの基本特殊函数ライブラリに含まれており, 効率的に計算可能である. この函数を使うと, 正規分布の累積分布函数は次のように書ける: $t = \sqrt{2}\, u$ とおくと,

$$\begin{aligned} F(z) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-t^2/2}\, dt = \frac{1}{2} + \frac{1}{\sqrt{2\pi}} \int_0^z e^{-t^2/2}\, dt \\ &= \frac{1}{2} + \frac{1}{\sqrt{\pi}} \int_0^{z/\sqrt{2}} e^{-u^2}\, du = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\left(\frac{z}{\sqrt{2}}\right) = \frac{1 + \mathrm{erf}(z/\sqrt{2})}{2}. \end{aligned}$$

In [59]:
```
plot(z -> cdf(Normal(), z), -4, 4;
    label="cdf of Normal()", legend=:topleft)
plot!(z -> (1 + erf(z/√2))/2, -4, 4;
    label="(1 + erf(z/√2))/2", ls=:dash)
plot!(; xtick=-5:5, ytick = 0:0.1:1,
    xlabel="z", ylabel="p")
plot!(; size=(300, 300))
```

Out[59]:

$y = \mathrm{erf}(x)$ の逆函数 $x = \mathrm{erfinv}(y)$ もコンピュータでの基本特殊函数ライブラリに含まれており, 効率的に計算可能である. 標準正規分布の累積分布函数 $p = F(z)$ の逆函数(分位点函数, quantile function)は

$$z = Q_{\mathrm{Normal}()}(p) = \sqrt{2}\ \mathrm{erfinv}(2p - 1)$$

と書ける. 標準正規分布の分位点函数は統計学で非常によく使われる. 例えば

$$Q_{\mathrm{Normal}()}(0.975) \approx 1.96$$

は非常に有名な数値である. これは標準正規分布において値が 1.96 以上になる確率が 2.5% になることを意味している.

In [60]:
```
quantile(Normal(), 0.975)
```

Out[60]: 1.9599639845400576

In [61]:
```
plot(p -> quantile(Normal(), p), 0, 1;
    label="quantile function of Normal()", legend=:topleft)
plot!(p -> √2*erfinv(2p-1), 0, 1;
    label="z = √2 erfinv(2p-1)", ls=:dash)
plot!(; xtick = 0:0.1:1, ytick=-5:5,
    xlabel="p", ylabel="z", ylim=(-4.2, 4.2))
plot!(; size=(300, 300))
```

Out[61]:

5.25 問題: 標準正規分布の四分位数

標準正規分布の累積分布函数と分位点函数はそれぞれ

$$p = F(z) = \int_{-\infty}^{z} \frac{e^{-t^2/2}}{\sqrt{2\pi}}\,dt = \frac{1+\mathrm{erf}(z/\sqrt{2})}{2},$$
$$z = Q(p) = F^{-1}(p) = \sqrt{2}\ \mathrm{erfinv}(2p-1)$$

と書ける. 標準正規分布の四分位数 $Q(0.25)$, $Q(0.75)$ をコンピュータを使って小数点以下第2桁まで求めよ.

解答例: $Q(0.25) \approx -0.6745$, $Q(0.75) \approx 0.6745$. **解答終**

In [62]:
```
quantile(Normal(), 0.25), quantile(Normal(), 0.75)
```

Out[62]: (-0.6744897501960818, 0.6744897501960818)

WolframAlpha (https://www.wolframalpha.com/) を使って計算するにはこう入力すればよい:

- √2 erfinv(2 0.25 - 1) (https://www.wolframalpha.com/input?i=%E2%88%9A2+erfinv%282+0.25+-+1%29)
- √2 erfinv(2 0.75 - 1) (https://www.wolframalpha.com/input?i=%E2%88%9A2+erfinv%282+0.75+-+1%29)

WolframAlpha にはこう聞いても答えを教えてくれる:

- quantile of standard normal distribution (https://www.wolframalpha.com/input?i=quantile+of+standard+normal+distribution)

5.26 問題: 連続分布の累積分布函数は一様分布に従う.

X は連続分布に従う確率変数であるとし, その累積分布函数を $F(x) = P(X \leq x)$ と書く. このとき, 累積分布函数 $F(x)$ に確率変数 X を代入してできる確率変数 $U = F(x)$ は一様分布 $\mathrm{Uniform}(0,1)$ に従うことを示せ.

注意: このことから, T を一様分布 $\mathrm{Uniform}(0,1)$ に従う確率変数とするとき, X が従う分布の分位点函数 $Q(t) = F^{-1}(t)$ に T を代入して得られる確率変数 $Q(T) = F^{-1}(T)$ は X と同じ連続分布に従うことがわかる. (後でコンピュータで数値的にこのことを確認する.)

解答例: X は連続分布に従うと仮定していたので, X は確率密度函数 $p(x) = F'(x)$ を持つ. 累積分布函数の値は0以上1以下なので, ゆえに,

$$E[g(U)] = E[g(F(X))] = \int_{\mathbb{R}} g(F(x))p(x)\,dx = \int_{\mathbb{R}} g(F(x))\,dF(x) = \int_0^1 g(u)\,du.$$

これで $U = F(X)$ が一様分布 $\mathrm{Uniform}(0,1)$ に従うことがわかった.

解答終

5.27 分位点函数への一様乱数の代入で元の分布の乱数を作れること

In [63]:
```
function plot_quantile_uniform(dist; L=10^6, tlim=(0, 1), title="", kwargs...)
    T = rand(L) # 一様乱数を大量に生成
    X = quantile.(dist, T) # 分位点函数で一様乱数を変換
    xlim = quantile.(Ref(X), tlim) .+ (-0.1, 0.1)
    X = X[first(xlim) .< X .< last(xlim)]
    if dist isa ContinuousUnivariateDistribution
        stephist(X; norm=true, label="")
        plot!(dist, xlim...; ls=:dash, label="", xlim)
    else
        bin = range(round.(extrema(X)) .+ (-0.5, 0.5)...; step=1)
        stephist(X; norm=true, bin, label="")
        plot!(x -> pdf(dist, round(x)), xlim...; ls=:dash, label="", xlim)
    end
    title!(title; kwargs...)
end
```

Out[63]:
```
plot_quantile_uniform (generic function with 1 method)
```

In [64]:
```
plot(plot_quantile_uniform(Normal(); title="Normal(0, 1)"),
    plot_quantile_uniform(Gamma(2, 3); title="Gamma(2, 3)", tlim=(0, 0.995));
    size=(800, 250))
```

Out[64]:

In [65]:
```
plot(plot_quantile_uniform(InverseGamma(2, 3); title="InverseGamma(2, 3)", tlim=(0, 0.98)),
    plot_quantile_uniform(LogNormal(); title="LogNormal(0, 1)", tlim=(0, 0.98));
    size=(800, 250))
```

Out[65]:

In [66]:
```
plot(plot_quantile_uniform(Beta(0.5, 0.5); title="Beta(0.5, 0.5)", xlim=(-0.1, 1.1), ylim=(-0.
    plot_quantile_uniform(Beta(10, 30); title="Beta(10, 30)", tlim=(0.005, 0.995));
    size=(800, 250))
```

Out[66]:

In [67]:
```
plot(plot_quantile_uniform(Binomial(10, 0.3), title="Binomial(10, 0.3)"),
    plot_quantile_uniform(Poisson(5); title="Poisson(5)");
    size=(800, 250))
```

Out[67]:

5.28 一様分布の期待値と分散

区間 (a, b) 上の確率密度函数

$$p(x) = \frac{1}{b-a} \quad (a < x < b)$$

によって与えられる一様分布

$$\mathrm{Uniform}(a, b)$$

を考え, X はこれに従う確率変数であるとする.

このとき, X の期待値(平均) μ と分散 σ^2 は以下のように計算される:

$$\mu = E[X] = \int_a^b xp(x)\,dx = \frac{1}{b-a}\int_a^b x\,dx = \frac{b^2-a^2}{2(b-a)} = \frac{a+b}{2}.$$

$$E[X^2] = \int_a^b x^2 p(x)\,dx = \frac{1}{b-a}\int_a^b x^2\,dx = \frac{b^3-a^3}{3(b-a)} = \frac{a^2+ab+b^2}{3}.$$

$$\mathrm{var}(X) = E[X^2] - E[X]^2 = \frac{4(a^2+ab+b^2)-3(a+b)^2}{12} = \frac{a^2-2ab+b^2}{12} = \frac{(b-a)^2}{12}.$$

In [68]:
```
function E(f, dist::ContinuousUnivariateDistribution)
    # quadgk(f, a, b)[1] は数値積分してくれる函数
    quadgk(x -> f(x) * pdf(dist, x), extrema(dist)...)[1]
end
```

Out[68]: E (generic function with 2 methods)

In [69]:
```
mean(Uniform(0, 1//0)), var(Uniform(0, 1//1))
```

Out[69]: (1//0, 1//12)

In [70]:
```
E(x -> x, Uniform(0, 1)), E(x -> x^2, Uniform(0, 1)) - E(x -> x, Uniform(0, 1))^2
```

Out[70]: (0.5, 0.08333333333333331)

In [71]:
```
mean(Uniform(0, 5//1)), var(Uniform(0, 5//1))
```

Out[71]: (5//2, 25//12)

In [72]:
```
E(x -> x, Uniform(0, 5)), E(x -> x^2, Uniform(0, 5)) - E(x -> x, Uniform(0, 5))^2
```

Out[72]: (2.5, 2.0833333333333357)

5.29 標準正規分布の期待値と分散

Z は標準正規分布に従う確率変数であるとする. このとき Z は確率密度函数

$$p(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2}$$

を持つ. このとき, $ze^{-z^2/2}$ が奇函数であり, 奇函数の $\mathbb{R}$ 全体での積分は 0 になるので,

$$E[Z] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} xe^{-z^2/2}\,dz = 0.$$

$z^2e^{-z^2/2}$ が偶函数であることと, $z = x^{1/2}$ と変数変換すると, $dz = x^{-1/2}\,dx/2$ となることを使うと,

$$\begin{aligned}\mathrm{var}(Z) = E[Z^2] &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} z^2e^{-z^2/2}\,dz = \frac{2}{\sqrt{2\pi}} \int_0^{\infty} z^2e^{-z^2/2}\,dz \\ &= \frac{2}{\sqrt{2\pi}} \int_0^{\infty} xe^{-x/2}\frac{x^{-1/2}\,dx}{2} = \frac{1}{\sqrt{2\pi}} \int_0^{\infty} x^{1/2-1}e^{-x/2}\,dx = \frac{2^{1/2}\Gamma(1/2)}{\sqrt{2\pi}} = 1.\end{aligned}$$

ここで, 5番目の等号では次の公式を使った:

$$\int_0^{\infty} x^{\alpha-1}e^{-x/\theta}\,dx = \theta^{\alpha}\Gamma(\alpha) \quad (\alpha, \theta > 0).$$

最後の等号では $\Gamma(1/2) = \sqrt{\pi}$ を使った. ガンマ函数はこれらの形で使われることが多い. これらの辺の公式については別のノートで解説することにする.

平均 μ と分散 σ^2 を持つ正規分布に従う確率変数 X は $X = \mu + \sigma Z$ と作れる.

注意: $\mathrm{var}(Z) = 1$ は以下のようにしても示される. Gauss積分の公式

$$\int_{-\infty}^{\infty} e^{-x^2}\,dx = \sqrt{\pi}$$

において, $a > 0$ を任意に取って, $x = \sqrt{a}\,z$ とおくと,

$$\sqrt{a}\int_{-\infty}^{\infty} e^{-az^2}\,dz = \sqrt{\pi}, \qquad \therefore \int_{-\infty}^{\infty} e^{-az^2}\,dz = \sqrt{\pi}\,a^{-1/2}.$$

両辺を a で微分してから -1 倍して(微分と積分の順序は交換できる), $a = 1/2$ とおくと,

$$\int_{-\infty}^{\infty} z^2e^{-az^2}\,dz = \frac{1}{2}\sqrt{\pi}\,a^{-3/2}, \quad \int_{-\infty}^{\infty} z^2e^{-z^2/2}\,dz = \sqrt{2\pi}.$$

ゆえに, $\mathrm{var}(Z) = E[Z^2] = (1/\sqrt{2\pi})\int_{-\infty}^{\infty} z^2e^{-z^2/2}\,dz = 1.$

In [73]:
```
mean(Normal(0, 1//1)) ▷ Int, var(Normal(0, 1//1)) ▷ Int
```
Out[73]:
```
(0, 1)
```

In [74]:
```
E(x → x, Normal(0, 1)), E(x → x^2, Normal(0, 1)) - E(x → x, Normal(0, 1))^2
```
Out[74]:
```
(4.2013717612707503e-16, 1.0)
```

In [75]:
```
mean(Normal(2, 3//1)) ▷ Int, var(Normal(2, 3//1)) ▷ Int
```
Out[75]:
```
(2, 9)
```

In [76]:
```
E(x → x, Normal(2, 3)), E(x → x^2, Normal(2, 3)) - E(x → x, Normal(2, 3))^2
```
Out[76]:
```
(1.999999999999999, 9.0)
```

標準正規分布の歪度(わいど, skewness):

In [77]:
```
skewness(Normal(0, 1//1)) ▷ Int
```
Out[77]:
```
0
```

In [78]:
```
E(z → z^3, Normal())
```
Out[78]:
```
-5.974359421244178e-16
```

標準正規分布の尖度(せんど, kurtosis)に 3 を足した値:

In [79]:
```
kurtosis(Normal(0, 1//1)) + 3 ▷ Int
```
Out[79]:
```
3
```

In [80]:
```
E(z → z^4, Normal())
```
Out[80]:
```
2.999999999999997
```

質問: 尖度の定義でどうして 3 を引いておくか?

答え: 正規分布の尖度を 0 にするためには4次のモーメントから 3 を引いておく必要がある.

In [81]:
```
@syms t κ3 κ4 κ5 μ3 μ4 μ5
series(exp(t^2//2 + κ3 * t^3//6 + κ4 * t^4//24 + κ5 * t^5//120), t)
```
Out[81]:

$$1 + \frac{t^2}{2} + \frac{t^3\kappa_3}{6} + t^4\left(\frac{\kappa_4}{24} + \frac{1}{8}\right) + t^5\left(\frac{\kappa_3}{12} + \frac{\kappa_5}{120}\right) + O\left(t^6\right)$$

In [82]:
```
series(log(1 + t^2//2 + μ3 * t^3//6 + μ4 * t^4//24 + μ5 * t^5//120), t)
```
Out[82]:

$$\frac{t^2}{2} + \frac{t^3\mu_3}{6} + t^4\left(\frac{\mu_4}{24} - \frac{1}{8}\right) + t^5\left(-\frac{\mu_3}{12} + \frac{\mu_5}{120}\right) + O\left(t^6\right)$$

$Z \sim \mathrm{Normal}()$ (標準正規分布)のとき, $E[Z^{2k}] = 1 \cdot 3 \cdots (2k-1)$ となる.

In [83]:
```
[(prod(1:2:2k-1), E(z → z^(2k), Normal())) for k in 1:7]
```
Out[83]:
```
7-element Vector{Tuple{Int64, Float64}}:
 (1, 1.0)
 (3, 2.999999999999997)
 (15, 14.999999999999998)
 (105, 105.00000000000004)
 (945, 945.0000000000093)
 (10395, 10395.000000000617)
 (135135, 135135.0000000145)
```

5.30 一様乱数の生成の繰り返しの分布

0 と 1 のあいだにある n 個の実数達 $t_1, \ldots, t_n$ の組 $(t_1, \ldots, t_n)$ に関する確率密度函数 $p(t_1, \ldots, t_n)$ を一様分布 $\mathrm{Uniform}(0,1)$ の確率密度函数

$$p(t) = \begin{cases} 1 & (0 < t < 1) \\ 0 & (\text{otherwise}) \end{cases}$$

の積で次のように定める:

$$p(t_1, \ldots, t_n) = p(t_1) \cdots p(t_n) = \begin{cases} 1 & (0 < t_1, \ldots, t_n < 1) \\ 0 & (\text{otherwise}) \end{cases}$$

これが実際に n 変数の確率密度函数になっていることは次のように確認できる:

$$\begin{aligned} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p(t_1, \ldots, t_n)\, dt_1 \cdots dt_n &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p(t_1) \cdots p(t_n)\, dt_1 \cdots dt_n \\ &= \int_{-\infty}^{\infty} p(t_1)\, dt_1 \cdots \int_{-\infty}^{\infty} p(t_n)\, dt_n = 1 \cdots 1 = 1. \end{aligned}$$

この確率密度函数 $p(t_1, \ldots, t_n)$ で定まる $\mathbb{R}^n$ 上の連続確率分布(n 変量の連続分布)を **一様乱数の生成の繰り返しの分布** と呼び, $\mathrm{Uniform}(0,1)^n$ と表す

この分布はコンピュータにおける一様乱数函数 `rand()` を n 回実行して得られる長さ n の 0 から 1 のあいだの乱数の列の分布のモデル化になっている.

この分布は後で別のノートで説明するように, 一様分布 $\mathrm{Uniform}(0,1)$ のサイズ n の **標本分布** (sample distribution)もしくは各々が一様分布 $\mathrm{Uniform}(0,1)$ に従う n 個の **独立同分布確率変数達** (independent and identically distributed random variables)が従う同時確率分布になっている.

$(T_1, \ldots, T_n)$ がこの分布に従う確率変数の組であるとき, その函数 $f(T_1, \ldots, T_n)$ の期待値は

$$E[f(T_1, \ldots, T_n)] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(t_1, \ldots, t_n) p(t_1, \ldots, t_n)\, dt_1 \cdots dt_n$$

と定義される. $f(T_1, \ldots, T_n)$ が $f(T_1, \ldots, T_n) = f_1(T_1) \ldots f_n(T_n)$ と各々の T_i のみの函数の積になっている場合には

$$\begin{aligned} E[f_1(T_1) \cdots f_n(T_n)] &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_1(t_1) \cdots f_n(t_n) p(t_1) \cdots p(t_n)\, dt_1 \cdots dt_n \\ &= \int_{-\infty}^{\infty} f_1(t_1) p(t_1)\, dt_1 \cdots \int_{-\infty}^{\infty} f_n(t_n) p(t_n)\, dt_n = E[f_1(T_1)] \cdots E[f_n(T_n)] \end{aligned}$$

が成立する. このように確率変数の積の期待値が各々の期待値の積に等しくなるとき, それらの確率変数は **独立** (independent)であるという.

以上のようなものは, $\mathrm{Uniform}(0,1)$ に限らず, 任意の確率分布について考えることができる.

例えば, Bernoulli試行の確率分布はBernoulli分布の標本分布になっている.

In [ ]:

```
1
```

Gauss積分, ガンマ函数, ベータ函数

- 黒木玄

- 2022-04-10〜2022-06-30

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 Gauss積分
  - 1.1 Gauss積分の定義
  - 1.2 問題: 正規分布での確率の総和
  - 1.3 正規分布の定義
  - 1.4 $p(x|\theta)p(x|\theta)$ という記号法について
  - 1.5 問題: Gauss積分の拡張とガンマ函数の関係
- 2 ガンマ函数
  - 2.1 ガンマ函数の定義
  - 2.2 問題: ガンマ函数はこの形でよく使われる
  - 2.3 ガンマ分布の定義
  - 2.4 問題: ガンマ分布の期待値と分散
  - 2.5 指数分布の定義
  - 2.6 Laplace分布の定義
  - 2.7 問題: Laplace分布の期待値と分散
  - 2.8 問題: 逆ガンマ分布の定義
  - 2.9 問題: 逆ガンマ分布の期待値と分散
  - 2.10 χ²分布の定義
  - 2.11 χ²分布と多変量正規分布の関係
  - 2.12 問題: 自由度 nn のχ²分布の nn 個の標準正規分布による表示
  - 2.13 Weibull分布の定義
  - 2.14 問題: Weibull分布の期待値と分散
  - 2.15 ガンマ函数の函数等式
  - 2.16 ガンマ函数の函数等式のよく見る使い方
  - 2.17 階乗のガンマ函数表示
  - 2.18 問題: 正の半整数でのガンマ函数の値
  - 2.19 問題: 標準正規分布の偶数次のモーメント
  - 2.20 問題: 標準正規分布のモーメント母函数とキュムラント母函数
  - 2.21 問題: n-1次元単位球面の面積のガンマ函数表示
  - 2.22 $n-1n-1$ 次元単位球面上の座標を使った面積の計算
  - 2.23 問題: n次元単位球体の面積のガンマ函数表示
  - 2.24 Stirlingの公式
  - 2.25 問題: 小さな nn でもStirlingの公式による $n!n!$ の近似精度がそう悪くないこと
  - 2.26 Stirlingの公式とガンマ分布の正規分布近似(中心極限定理)の関係
- 3 ベータ函数
  - 3.1 ベータ函数の定義
  - 3.2 ベータ分布の定義
  - 3.3 問題: ベータ分布の期待値と分散
  - 3.4 ベータ函数は $\alpha\alpha$ と $\beta\beta$ を交換しても不変
  - 3.5 問題: ベータ分布の三角函数を使った表示
  - 3.6 ベータ函数の極限としてガンマ函数を表せること
  - 3.7 問題: ベータ分布の極限でガンマ分布が得られること
  - 3.8 ベータ函数の別の基本的表示

3.9 問題: tt 分布の基礎になる公式
3.10 tt 分布の定義
3.11 tt 分布の期待値と分散
3.12 問題: tt 分布の自由度無限大の極限で標準正規分布が得られること
3.13 tt 分布と標準正規分布を同時プロット
4 ベータ函数のガンマ函数表示
4.1 積分 $J[f]J[f]$ を導入とベータ函数のガンマ函数表示の証明
4.2 積分 $J[f]J[f]$ に関する公式の証明
4.3 問題: ガンマ分布とベータ分布の関係 (ベータ分布のガンマ分布表示)
4.4 問題: ガンマ分布の再生成
4.5 積分 $K[f]K[f]$ を導入とベータ函数のガンマ函数表示の別証明
4.6 積分 $K[g]K[g]$ に関する公式の証明
4.7 ベータプライム分布の定義
4.8 FF 分布の定義
4.9 FF 分布とベータ分布やガンマ分布やχ²分布との関係
4.10 問題: ガンマ函数のGaussの乗法公式
4.11 問題: $\sin\sin$ の無限積表示
4.12 問題: Wallisの公式
4.13 問題: ガンマ函数の相反公式(reflection formula)
4.14 ディガンマ函数, トリガンマ函数, ポリガンマ函数, Hurwitzのゼータ函数
5 Dirichlet積分=多変数ベータ函数
5.1 Dirichlet積分=多変数ベータ函数の定義とそのガンマ函数表示
5.2 Dirichlet分布の定義
5.3 多変数ベータ函数のガンマ函数表示の証明
5.4 ガンマ分布とDirichlet分布の関係
5.5 問題: Dirichlet (1839)の公式の証明
5.6 Dirichlet氏の1839年の論文との比較

In [1]:

```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:

```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

1 Gauss積分

1.1 Gauss積分の定義

次の積分 I を **Gauss積分** と呼ぶ:

$$I = \int_{-\infty}^{\infty} e^{-x^2}\,dx = \sqrt{\pi}.$$

この積分の値が実際に円周率の平方根になることを以下で示そう.

Gauss積分の2乗

$$I^2 = \int_{-\infty}^{\infty} e^{-x^2}\,dx \cdot \int_{-\infty}^{\infty} e^{-y^2}\,dy = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-(x^2+y^2)}\,dx\,dy$$

は次のグラフで示された高さ 1 の山の体積になる.

In [3]:

```
x = y = range(-3, 3, 61)
wireframe(x, y, (x, y) → exp(-(x^2+y^2)); camera=(30, 70), colorbar=false)
plot!(; zlim=(-0.02, 1.02), size=(600, 400))
plot!(; xlabel="x", ylabel="y", zlabel="z")
title!("z = exp(-(x²+y²))"; titlefontsize=12)
```

Out[3]:

この山は z 軸 $x = y = 0$ を中心にして回転対称である. 次のグラフは横軸を半径 $r = \sqrt{x^2 + y^2}$ とした場合のこの山の高さのグラフである.

In [4]:

```
plot(r → exp(-r^2), 0, 3; label="", title="z = exp(-r²) ⇔ r = √(-log(z))")
plot!(; xlabel="r = √(x² + y²)", ylabel="z")
plot!(; ytick=0:0.1:1)
z = 0.3
plot!([(0, z), (√(-log(z)), z)]; label="", lw=2)
plot!(; xlim=(0, 3))
annotate!(1.1, 0.2, ("√(-log(0.3)) = r", 10, :right))
annotate!(1.2, 0.3, ("← 0.3 = exp(-r²)", 10, :left))
```

Out[4]:

この図より, 高さ $0 < z \leq 1$ での山の切断面は, 半径 $\sqrt{-\log z}$ の円盤になる. その円盤の面積 $\pi(-\log z)$ を $0 < z \leq 1$ について積分すれば上の山の体積(= Gauss積分の2乗 I^2 の値)が得られる:

$$I^2 = \int_0^1 \pi(-\log z)\,dz = -\pi[z\log z - z]_0^1 = -\pi(-1) = \pi.$$

ゆえに

$$I = \int_{-\infty}^{\infty} e^{-x^2}\,dx = \sqrt{\pi}.$$

1.2 問題: 正規分布での確率の総和

$\mu, \sigma \in \mathbb{R}$, $\sigma > 0$ であると仮定する. 次を示せ:

$$\frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = 1.$$

解答例: 左辺の積分部分が $\sqrt{2\pi\sigma^2}$ に等しいことを示せばよい.

$$\int_{-\infty}^{\infty} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = \int_{-\infty}^{\infty} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = \sqrt{2\sigma^2}\int_{-\infty}^{\infty} e^{-z^2}\,dz = \sqrt{2\sigma^2}\sqrt{\pi} = \sqrt{2\pi\sigma^2}$$

最初の等号で $x = y + \mu$ とおき, 2つめの等号で $y = \sqrt{2\sigma^2}\,z$ とおき, 3つめの等号でGauss積分を使った.

解答終

1.3 正規分布の定義

$\mu, \sigma \in \mathbb{R}$, $\sigma > 0$ であると仮定する.

次の確率密度函数によって定義される連続分布を平均 μ, 分散 σ^2 (もしくは標準偏差 σ)の **正規分布** (normal distribution)と呼び $\mathrm{Normal}(\mu, \sigma)$ と表す:

$$p(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

特に $\mu = 0$, $\sigma = 1$ の正規分布を **標準正規分布** (standard normal distribution)と呼び, $\mathrm{Normal}()$ と表すことにする:

$$p(x|0,1) = \frac{1}{\sqrt{2\pi}} e^{-x^2}.$$

1.4 $p(x|\theta)$ という記号法について

パラメータ θ を持つ x に関する確率密度函数を $p(x|\theta)$ と書く.

この記号法によって, 縦棒 $|$ の左側の変数 x について確率密度函数になっているが, 縦棒 $|$ の右側の変数 θ については単なる函数で確率密度函数ではないことを表す.

パラメータ θ を持つ x に関する確率質量函数 $P(x|\theta)$ についても同様である.

1.5 問題: Gauss積分の拡張とガンマ函数の関係

$\alpha \in \mathbb{R}$, $\alpha > 0$ であると仮定する. 次を示せ:

$$\int_{-\infty}^{\infty} e^{-x^2}|x|^{2\alpha-1}\,dx = \int_0^{\infty} e^{-y}y^{\alpha-1}\,dy$$

解答例: $x = \sqrt{y}$ とおくと, $dx/x = d\log x = (1/2)d\log y = dy/(2y)$ なので,

$$\int_{-\infty}^{\infty} e^{-x^2}|x|^{2\alpha-1}\,dx = 2\int_0^{\infty} e^{-x^2}x^{2\alpha}\,\frac{dx}{x} = 2\int_0^{\infty} e^{-y}y^{\alpha}\,\frac{dy}{2y} = \int_0^{\infty} e^{-y}y^{\alpha-1}\,dy.$$

最初の等号で被積分函数が偶函数であることを用い, 第2の等号で $x = \sqrt{y}$ とおいた.

解答終

この問題では単なる $\exp(-x^2)$ の積分であるGauss積分をそれに $|x|$ のべきをかけたものの積分に拡張した. 実はこの拡張は次の節で扱うガンマ函数の定義そのものになっている. すなわち, ガンマ函数はGauss積分の拡張の一種になっている.

正規分布の確率密度函数は釣鐘型(bell-shaped)になる.

In [5]:
```
plot(Normal(-1, 2); label="", title="Normal(-1, 2)", xtick = -100:100)
```

Out[5]:

2 ガンマ函数

2.1 ガンマ函数の定義

$\alpha > 0$ について, ガンマ函数 $\Gamma(\alpha)$ を次のように定義する:

$$\Gamma(\alpha) = \int_0^\infty e^{-x} x^{\alpha-1}\, dx.$$

前節の問題の結果より, これは

$$\Gamma(\alpha) = \int_{-\infty}^\infty e^{-x^2} |x|^{2\alpha-1}\, dx$$

とGauss積分の拡張形でも表される. 特に

$$\Gamma(1/2) = \int_{-\infty}^\infty e^{-x^2}\, dx = \sqrt{\pi}.$$

このようにガンマ函数 $\Gamma(\alpha)$ はGauss積分 $\Gamma(1/2)$ の拡張になっている.

$\Gamma(1) = 1$ となることも容易に示せる:

$$\Gamma(1) = \int_0^\infty e^{-x}\, dx = [-e^{-x}]_0^\infty = -(-1) = 1.$$

注意: Gauss積分は正規分布の定義の基礎になっている. ここで扱っているようなGauss積分の拡張形は平均が0の正規分布の絶対モーメントになっており, 正規分布に関係した計算には自然にガンマ函数が出て来る.

2.2 問題: ガンマ函数はこの形でよく使われる

$\alpha, \theta > 0$ について次の公式が成立することを示せ:

$$\int_0^\infty e^{-x/\theta} x^{\alpha-1}\, dx = \theta^\alpha \Gamma(\alpha).$$

解答例: $x = \theta y$ とおくと, $dx/x = \theta\, dy/(\theta y) = dy/y$ なので,

$$\int_0^\infty e^{-x/\theta} x^{\alpha-1}\, dx = \int_0^\infty e^{-x/\theta} x^\alpha \frac{dx}{x} = \int_0^\infty e^{-y} (\theta y)^\alpha \frac{dy}{y} = \theta^\alpha \int_0^\infty e^{-y} y^{\alpha-1}\, dy = \theta^\alpha \Gamma(\alpha).$$

2つめの等号で $x = \theta y$ を代入して, 4つめの等号でガンマ函数の定義を使った.

解答終

ガンマ函数は主にこの形で使う と覚えておくと便利だと思う.

例えば, この問題の結果から,

$$\int_0^\infty \frac{e^{-x/\theta}x^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}\,dx = 1$$

という公式が得られる. これは「ガンマ分布において確率の総和が1になる」という意味を持つ.

2.3 ガンマ分布の定義

$\alpha, \theta > 0$ であると仮定する.

次の確率密度函数によって定義される連続分布を**形状パラメータ** α, **スケールパラメータ** θ の **ガンマ分布** (Gamma distribution)と呼び $\mathrm{Gamma}(\alpha, \theta)$ と表す:

$$p(x|\alpha,\theta) = \frac{1}{\theta^\alpha\Gamma(\alpha)}e^{-x/\theta}x^{\alpha-1} \quad (x > 0).$$

ガンマ分布は正規分布から派生して出て来る分布の中で非常に基本的なものになっている.

負の二項分布の連続極限としてガンマ分布を出すことによって, ガンマ分布には「1回起こるまでにかかる時間の期待値が θ であるようなイベントが α 回起こるまでにかかる時間の分布」だと解釈できる. この点については後で「確率分布達の解釈」のノートで説明する.

スケールパラメータを変えてもガンマ分布の確率密度函数 $p(x|\alpha,\theta)$ のグラフの形状は変化せず, 単に x 軸方向のスケールが変わって, 密度の値もそれに応じてスケールされるだけになる. ゆえに, ガンマ分布の形状を知りたければ, 形状パラメータ α のみを動かして, ガンマ分布のグラフを描けばよい.

$\alpha = 1/2$ のガンマ分布の確率密度函数は次の形になる.

In [6]:
```
α, θ = 1/2, 1
plot(Gamma(α, θ), 0.01, 2; label="", title="Gamma($α, $θ)")
```

Out[6]:

$\alpha = 3$ のガンマ分布の確率密度函数は次の形になる.

In [7]:
```
α, θ = 3, 1
plot(Gamma(α, θ); label="", title="Gamma($α, $θ)")
```

Out[7]:

$\alpha = 30$ のガンマ分布の確率密度函数は次の形になる.

In [8]:
```
α, θ = 30, 1
plot(Gamma(α, θ); label="", title="Gamma($α, $θ)")
```

Out[8]:

形状パラメータ α が大きなガンマ分布は正規分布で近似される.

In [9]:
```
function plot_gamma_normal(α, θ; kwargs...)
    gam = Gamma(α, θ)
    μ, σ = mean(gam), std(gam)
    x = range(max(0, μ - 5σ), μ + 5σ, 400)
    P1 = plot(x, x -> pdf(gam, x); label="Gamma(α, θ)", kwargs...)
    α > 2 && plot!(x, x -> pdf(Normal(μ, σ), x); label="normal approx.", ls=:dash)
    title!("α=$α, θ=$θ")
end
```

Out[9]: plot_gamma_normal (generic function with 1 method)

In [10]:
```
plot(plot_gamma_normal(0.5, 1), plot_gamma_normal(1, 1); size=(800, 250))
```

Out[10]:

In [11]:
```
plot(plot_gamma_normal(1.5, 1), plot_gamma_normal(2, 1); size=(800, 250))
```

Out[11]:

In [12]:
```
plot(plot_gamma_normal(5, 1), plot_gamma_normal(15, 1); size=(800, 250))
```

Out[12]:

In [13]:
```
plot(plot_gamma_normal(50, 1), plot_gamma_normal(150, 1); size=(800, 250))
```

Out[13]:

ガンマ分布の確率密度函数は $(\alpha-1)\theta$ で最大になることを示せる. それに対して, それを近似する正規分布の期待値はもとのガンマ分布の期待値 $\alpha\theta$ に等しくなるので, $\alpha=30$, $\theta=1$ の場合にはその分だけずれが生じている.

In [14]:
```
plot_gamma_normal(300, 1)
```

Out[14]:

注意: 形状パラメータ α が大きなガンマ分布は正規分布でよく近似されることは本質的に階乗に関するStirlingの(近似)公式に同等である.

2.4 問題: ガンマ分布の期待値と分散

分布 $\mathrm{Gamma}(\alpha,\theta)$ の期待値と分散がそれぞれ $\alpha\theta$, $\alpha\theta^2$ となることを示せ.

解答例: $X\sim\mathrm{Gamma}(\alpha,\theta)$ と仮定する. $k=0,1,2,\ldots$ について

$$
\begin{aligned}
\Gamma(\alpha+k) &= (\alpha+k-1)\Gamma(\alpha+k-1) = \cdots = (\alpha+k-1)\cdots(\alpha+1)\alpha\Gamma(\alpha) \\
&= \alpha(\alpha+1)\cdots(\alpha+k-1)\Gamma(\alpha)
\end{aligned}
$$

なので

$$
E[X^k] = \frac{1}{\theta^\alpha\Gamma(\alpha)}\int_0^\infty e^{-x/\theta}x^{\alpha+k-1}\,dx = \frac{\theta^{\alpha+k}\Gamma(\alpha+k)}{\theta^\alpha\Gamma(\alpha)} = \theta^k\alpha(\alpha+1)\cdots(\alpha+k-1).
$$

ゆえに,

$$
E[X] = \theta\alpha = \alpha\theta, \quad \mathrm{var}(X) = E[X^2] - E[X]^2 = \theta^2\alpha(\alpha+1) - \theta^2\alpha^2 = \theta^2\alpha = \alpha\theta^2.
$$

解答終

2.5 指数分布の定義

$\alpha = 1$ のガンマ分布の密度函数は

$$
p(x|1,\theta) = \frac{1}{\theta}e^{-x/\theta} \quad (x > 0)
$$

の形になる. この形の確率密度函数で定義される連続確率分布を **指数分布** (exponential distribution)と呼び, $\mathrm{Exponential}(\theta)$ と表す:

$$
\mathrm{Exponential}(\theta) = \mathrm{Gamma}(1,\theta).
$$

ガンマ分布の期待値と分散に関する結果より, 指数分布の期待値と分散はそれぞれ θ, θ^2 になる.

指数分布は「1回起こるまでにかかる時間の期待値が θ であるようなイベントが起こるまでにかかる時間の分布」だと解釈可能である. この点については後で「確率分布達の解釈」のノートで説明する.

注意: 家計が保有する金融資産の分布は指数分布に近い形の分布になり, そのことには統計力学的な解釈がある.

指数分布の確率密度函数は次の形になる.

In [15]:
```
plot_exp(θ) = plot(Exponential(θ); label="", title="Exponential($θ)")
plot(plot_exp(1), plot_exp(3), plot_exp(10); size=(800, 200), layout=(1, 3))
```

Out[15]:

指数分布のグラフはどれも同じ形状になる. 指数分布 $\mathrm{Exponential}(\theta)$ の期待値と分散はそれぞれ θ, θ^2 になる.

2.6 Laplace分布の定義

$a \in \mathbb{R}$, $b > 0$ であると仮定する. 確率密度函数

$$
p(x|a,b) = \frac{1}{2b}\exp\left(-\frac{|x-a|}{b}\right)
$$

が定める $\mathbb{R}$ 上の連続確率分布を **Laplace分布** (ラプラス分布)と呼び,

$$
\mathrm{Laplace}(a,b)
$$

と書くことにする. Laplace分布は $x = a$ の左右両側に指数分布を並べた分布になっているので, **ダブル指数分布** (double exponential distribution)と呼ばれることもある.

$a = 0$, $b = 1$ のLaplace分布 $\mathrm{Laplace}(0,1)$ を **標準Laplace分布** と呼ぶ.

In [16]:

```
plot_laplace(a, b) = plot(Laplace(a, b); label="", title="Laplace($a,$b)")
plot(plot_laplace(0, 1), plot_laplace(0, 0.5), plot_laplace(2, 3);
    size=(800, 200), layout=(1, 3))
```

Out[16]:

2.7 問題: Laplace分布の期待値と分散

Laplace分布 $\mathrm{Laplace}(a, b)$ に従う確率変数 X について

$$E[X] = a, \quad \mathrm{var}(X) = 2b^2$$

となっていることを示せ.

解答例: Y が標準Laplace分布に従う場合には, $ye^{-|y|}$ が奇函数なので,

$$E[Y] = \frac{1}{2}\int_{-\infty}^{\infty} ye^{-|y|}\,dy = 0$$

となり, さらに,

$$\mathrm{var}(Y) = \frac{1}{2}\int_{-\infty}^{\infty} y^2 e^{-|y|}\,dy = \int_{-\infty}^{\infty} y^2 e^{-y}\,dy = \Gamma(3) = 2! = 2.$$

$X = a + bY$ とおくと, $X \sim \mathrm{Laplace}(a, b)$ となる. このとき,

$$E[X] = a + bE[Y] = a, \quad \mathrm{var}(X) = b^2\,\mathrm{var}(Y) = 2b^2.$$

解答終

2.8 問題: 逆ガンマ分布の定義

X がガンマ分布 $\Gamma(\alpha, 1/\theta)$ に従う確率変数であるとき, その逆数 $Y = 1/X$ について

$$E[f(Y)] = \int_0^{\infty} f(y)\frac{e^{-\theta/y}y^{-\alpha-1}}{(1/\theta)^{\alpha}\Gamma(\alpha)}\,dy$$

を満たすことを示せ. 確率密度函数

$$p(y|\alpha, \theta) = \frac{e^{-\theta/y}y^{-\alpha-1}}{(1/\theta)^{\alpha}\Gamma(\alpha)} = \frac{\theta^{\alpha}}{\Gamma(\alpha)}e^{theta/y}y^{-\alpha-1} \quad (y > 0)$$

で定義される連続分布を **逆ガンマ分布** (inverse Gamma distribution)と呼び,

$$\mathrm{InverseGamma}(\alpha, \theta)$$

と表すことにする. 逆ガンマ分布はガンマ分布に従う確率変数の逆数が従う分布のことである.

解答例: 積分変数 x を $x = 1/y$ で置換すると, $dx = -dy/y^2$ より,

$$\begin{aligned} E[f(Y)] = E[f(1/X)] &= \int_0^{\infty} f(1/x)\frac{e^{-\theta x}x^{\alpha-1}}{(1/\theta)^{\alpha}\Gamma(\alpha)}\,dx \\ &= \int_{\infty}^{0} f(y)\frac{e^{-\theta/y}y^{-\alpha+1}}{(1/\theta)^{\alpha}\Gamma(\alpha)}\frac{-dy}{y^2} = \int_0^{\infty} f(y)\frac{e^{-\theta/y}y^{-\alpha-1}}{(1/\theta)^{\alpha}\Gamma(\alpha)}\,dy. \end{aligned}$$

解答終

In [17]:
```
p_inversegamma(α, θ, x) = x ≥ 0 ? pdf(InverseGamma(α, θ), x) : zero(x)
function plot_inversegamma_normal(α, θ; kwargs...)
    invgam = InverseGamma(α, θ)
    if α > 2
        μ, σ = mean(invgam), std(invgam)
        x = range(max(0, μ - 5σ), μ + 5σ, 1000)
    else
        x = range(0, quantile(invgam, 0.75 + 0.1α), 1000)
    end
    P1 = plot(x, x -> p_inversegamma(α, θ, x); label="InvGam(α, θ)", kwargs...)
    α > 2 && plot!(x, x -> pdf(Normal(μ, σ), x); label="normal approx.", ls=:dash)
    title!("α=$α, θ=$θ")
end
```

Out[17]: plot_inversegamma_normal (generic function with 1 method)

In [18]:
```
plot(plot_inversegamma_normal(0.5, 2), plot_inversegamma_normal(1, 1); size=(800, 250))
```

Out[18]:

In [19]:
```
plot(plot_inversegamma_normal(1.5, 1), plot_inversegamma_normal(2, 1); size=(800, 250))
```

Out[19]:

In [20]:
```
plot(plot_inversegamma_normal(5, 1), plot_inversegamma_normal(15, 1); size=(800, 250))
```

Out[20]:

In [21]:
```
plot(plot_inversegamma_normal(50, 1), plot_inversegamma_normal(150, 1); size=(800, 250))
```

Out[21]:

2.9 問題: 逆ガンマ分布の期待値と分散

確率変数 X は $X \sim \mathrm{InverseGamma}(\alpha, \theta)$ を満たしていると仮定する. このとき, 以下が成立することを示せ:

$$\alpha > 1 \implies E[X] = \frac{\theta}{\alpha - 1}$$
$$\alpha > 2 \implies \mathrm{var}(X) = \frac{\theta^2}{(\alpha-1)^2(\alpha-2)}.$$

解答例: $\alpha > 0$ のとき, 逆ガンマ分布の確率の総和が 1 であることより,

$$\int_0^\infty e^{-\theta/x} x^{-\alpha-1}\, dx = \frac{\Gamma(\alpha)}{\theta^\alpha} E[1] = \frac{\Gamma(\alpha)}{\theta^\alpha}.$$

$\alpha > 1$ のとき,

$$E[X] = \frac{\theta^\alpha}{\Gamma(\alpha)} \int_0^\infty e^{-\theta/x} x^{-(\alpha-1)-1}\, dx = \frac{\theta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha-1)}{\theta^{\alpha-1}} = \frac{\theta}{\alpha-1}.$$

$\alpha > 2$ のとき,

$$E[X^2] = \frac{\theta^\alpha}{\Gamma(\alpha)} \int_0^\infty e^{-\theta/x} x^{-(\alpha-2)-1}\, dx = \frac{\theta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha-2)}{\theta^{\alpha-2}} = \frac{\theta^2}{(\alpha-1)(\alpha-2)}.$$

ゆえに

$$\mathrm{var}(X) = E[X^2] - E[X]^2 = \frac{\theta^2}{(\alpha-1)^2(\alpha-2)}.$$

解答終

In [22]:
```
@vars α θ
E_X = θ/(α - 1)
E_X² = θ^2/((α - 1)*(α - 2))
var_X = E_X² - E_X^2 ▷ factor
```

Out[22]:

$$\frac{\theta^2}{(\alpha-2)(\alpha-1)^2}$$

2.10 χ²分布の定義

指数分布はガンマ分布 $\Gamma(\alpha, \theta)$ の形状パラメータ α を $\alpha = 1$ に固定して得られる分布であった.

χ²分布 は本質的にガンマ分布 $\Gamma(\alpha, \theta)$ のスケールパラメータ θ を $\theta = 2$ に固定して得られる分布のことである. ただし, $\alpha = \nu/2$ とおいて, ν を **自由度** (degree of freedom, df)と呼ぶ. すなわち, 分布 $\mathrm{Gamma}(\nu/2, 2)$ を自由度 ν の **χ²分布** (カイ二乗分布, chi-squared distribution)と呼び,

$$\mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$$

と表す. χ²分布の確率密度函数は次の形になる:

$$p(x|\nu) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} e^{x/2} x^{\nu/2-1} \quad (x > 0)$$

ガンマ分布の期待値と分散に関する結果より, 指数分布の期待値と分散はそれぞれ ν, 2ν になる.

ガンマ分布は α が大きなとき正規分布で近似されるので, **自由度 ν が大きなχ²分布は正規分布で近似される.**

In [23]:
```
p_chisq(ν, x) = (ν > 2 || x ≥ 0 || x ≤ -0.1) ? pdf(Chisq(ν), x) : NaN
function plot_chisq_normal(ν; kwargs...)
    x = range(ν - 5√(2ν), ν + 5√(2ν), 1000)
    P1 = plot(x, x -> p_chisq(ν, x); label="Chisq(ν)", kwargs...)
    plot!(x, x -> pdf(Normal(ν, √(2ν)), x); label="Normal(ν,√(2ν))", ls=:dash)
    title!("ν = $ν")
end
```

Out[23]: plot_chisq_normal (generic function with 1 method)

In [24]:
```
plot(plot_chisq_normal(1; ylim=(-0.05, 1.55)), plot_chisq_normal(2); size=(800, 250))
```

Out[24]:

In [25]:
```
plot(plot_chisq_normal(3), plot_chisq_normal(4); size=(800, 250))
```

Out[25]:

In [26]:
```
plot(plot_chisq_normal(10), plot_chisq_normal(30); size=(800, 250))
```

Out[26]:

In [27]:
```
plot(plot_chisq_normal(100), plot_chisq_normal(300); size=(800, 250))
```

Out[27]:

2.11 χ²分布と多変量正規分布の関係

ν が正の整数のとき, カイ二乗分布は標準正規分布に従う独立な ν 個の確率変数達の和が従う分布になっている(次の問題を見よ).

この段階では多変量正規分布を導入していないので詳しくは説明できないのだが, より一般に, $(X_1, \dots, X_\nu)$ が確率密度函数が

$$p(x_1, \dots, x_n) = \mathrm{const.}\exp\left(-\frac{1}{2}\sum_{i,j=1}^{\nu} a_{ij}x_ix_j\right)$$

の形の多変量正規分布に従うときに,

$$X^2 = \sum_{i,j=1}^{\nu} a_{ij}X_iX_j$$

の従う分布は自由度 ν のχ²分布になる. このようにχ²分布は正規分布と直接的に関係した分布になっている.

$X_1, \dots, X_\nu$ は各々が標準正規分布に従う独立な確率変数達であることは, それらの同時確率密度函数が

$$p(x_1, \dots, x_\nu) = \frac{1}{(2\pi)^{\nu/2}}\exp\left(-\frac{1}{2}(x_1^2 + \cdots + x_\nu^2)\right)$$

となることと同値なので, この場合には上で述べたように $X^2 = X_1^2 + \cdots + X_\nu^2$ が従う分布が自由度 ν のχ²分布になる. χ²分布の自由度は「独立な正規分布が何個あるか」を意味していると考えられる.

2.12 問題: 自由度 n のχ²分布の n 個の標準正規分布による表示

確率変数達 $X_1, \dots, X_n$ は同時確率密度函数

$$p(x_1, \dots, x_n) = \frac{1}{(2\pi)^{n/2}}\exp\left(-\frac{1}{2}(x_1^2 + \cdots + x_n^2)\right)$$

を持つと仮定する. すなわち,

$$E[f(X_1, \dots, X_n)] = \int\cdots\int_{\mathbb{R}^n} f(x_1, \dots, x_n)p(x_1, \dots, x_n)\,dx_1\cdots x_n$$

が成立していると仮定する. このとき, 確率変数 $Y = X^2$ を

$$Y = X^2 = X_1^2 + \cdots + X_n^2$$

とおくと, Y は自由度 n のχ²分布に従うことを示せ. すなわち,

$$E[f(Y)] = \frac{1}{2^{n/2}\Gamma(n/2)}\int_0^\infty f(y)e^{y/2}y^{n/2-1}\,dy$$

となることを示せ. この問題の解答では, ずっと下の方で解説した n 次元球面の面積 A_{n-1} に関する

$$A_{n-1} = \frac{2\pi^{n/2}}{\Gamma(n/2)}$$

という結果とこれを示すために使った方法をに自由に使ってよい.

解答例1: $Y = X_1^2 + \cdots + X_n^2$ の函数の期待値は次のように書ける:

$$E[f(Y)] = \frac{1}{(2\pi)^{n/2}} \int \cdots \int_{\mathbb{R}^n} f(x_1^2 + \cdots + x_n^2) p(x_1, \ldots, x_n) \exp\left(-\frac{1}{2}(x_1^2 + \cdots + x_n^2)\right) dx_1 \cdots x_n$$

この積分中の被積分函数は球対称なので, 半径 r に関する積分と単位球面 S^{n-1} 上の積分に分解できる:

$$\begin{aligned} E[f(Y)] &= \frac{1}{(2\pi)^{n/2}} \int_{S^{n-1}} \left(\int_0^\infty f(r^2) e^{-r^2} r^{n-1} \, dr \right) d\omega = \frac{A_{n-1}}{(2\pi)^{n/2}} \int_0^\infty f(r^2) e^{-r^2} r^{n-1} \, dr \\ &= \frac{2\pi^{n/2}}{(2\pi)^{n/2} \Gamma(n/2)} \int_0^\infty f(y) e^{-y} y^{(n-1)/2} \frac{1}{2} y^{-1/2} \, dy = \frac{1}{2^{n/2} \Gamma(n/2)} \int_0^\infty f(y) e^{-y} y^{n/2-1} \, dy. \end{aligned}$$

2つめの等号で単位球面上 S^{n-1} での定数の積分を実行し(単位球面の面積 A_{n-1} 倍になる), 3つめの等号で $r = \sqrt{y}$ とおいた.

解答終

解答例2: ずっと下の方で示すガンマ分布の再生成に帰着する方法でこの問題を解くこともできる. 以下の2つを証明できれば, $X_1, X_2, \ldots, X_n$ が独立な確率変数達でかつ各 X_i が標準正規分布に従うとき, $Y = X^2 = X_1^2 + \cdots + X_n^2$ が自由度 n のχ²分布に従うことがわかる:

(1) Z が標準正規分布に従う確率変数であるとき, $Y = Z^2$ は自由度 1 のχ²分布に従う.

(2) 独立な確率変数達 Y_1, Y_2 の各々が自由度 ν_1, ν_2 のχ²分布に従うとき $Y_1 + Y_2$ は自由度 $\nu_1 + \nu_2$ のχ²分布に従う.

この(2)は $\mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$ なのでガンマ分布の再生成の特別な場合に過ぎない. そして(1)は以下のように容易に証明される:

$$\begin{aligned} E[f(Y)] &= \int_{-\infty}^\infty f(z^2) \frac{e^{-z^2/2}}{\sqrt{2\pi}} \, dz = 2 \int_0^\infty f(z^2) \frac{e^{-z^2/2}}{\sqrt{2\pi}} \, dz \\ &= \int_0^\infty f(y) \frac{e^{-y/2}}{\sqrt{2\pi}} y^{-1/2} \, dy = \int_0^\infty f(y) \frac{e^{-y/2} y^{1/2-1}}{2^{1/2} \Gamma(1/2)} \, dy \end{aligned}$$

3つめの等号で $z = \sqrt{y}$ とおき, 最後の等号でGauss積分 $\Gamma(1/2) = \sqrt{2}$ を使った. これで(1)も示された.

解答終

2.13 Weibull分布の定義

T は平均 1 の指数分布 $\mathrm{Exponential}(1)$ に従う確率変数であるとする:

$$E[f(T)] = \int_0^\infty f(t) e^{-t} \, dt.$$

$\alpha, \theta > 0$ について, 確率変数 $W = \theta T^{1/\alpha}$ が従う分布を **形状**(shape)が α で **スケール** (scale)が θ のWeilbull分布と呼び,

$$\mathrm{Weibull}(\alpha, \theta)$$

と表すことにする. このとき, $W = \theta T^{1/\alpha}$ は T について $T = (W/\theta)^\alpha$ と解けるので,

$$E[f(W)] = \int_0^i nftyf(\theta t^{1/\alpha}) e^{-t} \, dt = \int_0^\infty f(w) e^{-(w/\theta)^\alpha} \frac{\alpha}{\theta} \left(\frac{w}{\theta} \right)^{\alpha-1} dw$$

なので, Weilbull分布の確率密度函数は次の形になることがわかる:

$$p(w|\alpha, \theta) = \frac{\alpha}{\theta} \left(\frac{w}{\theta} \right)^{\alpha-1} e^{-(w/\theta)^\alpha} \quad (w > 0)$$

注意: 平均 1 の指数分布は単位時間に平均して 1 回起こるイベントが起こるまでにかかる時間の分布とみなされる(「確率分布達の解釈」のノートで説明する). Weibull分布はイベントが起こる感覚が段々減速したり加速したりするときのイベントが起こるまでにかかる時間の分布だとみなされる.

注意: Weibull分布ではガンマ分布と違って形状パラメータを大きくしても正規分布で近似されるようにならない.

In [28]:
```
function plot_weibull_normal(α, θ; kwargs...)
    weib = Weibull(α, θ)
    μ, σ = mean(weib), std(weib)
    x = range(max(0, μ - 5σ), μ + 5σ, 400)
    plot(x, x -> pdf(weib, x); label="Weibull(α, θ)", kwargs...)
    plot!(x, x -> pdf(Normal(μ, σ), x); label="normal approx.", ls=:dash)
    title!("α=$α, θ=$θ")
end
```

Out[28]: plot_weibull_normal (generic function with 1 method)

In [29]:
```
plot(plot_weibull_normal(0.5, 1), plot_weibull_normal(1, 1); size=(800, 250))
```

Out[29]:

In [30]:
```
plot(plot_weibull_normal(1.5, 1), plot_weibull_normal(2, 1); size=(800, 250))
```

Out[30]:

In [31]:
```
plot(plot_weibull_normal(5, 1), plot_weibull_normal(15, 1); size=(800, 250))
```

Out[31]:

In [32]:
```
plot(plot_weibull_normal(50, 1), plot_weibull_normal(150, 1); size=(800, 250))
```

Out[32]:

2.14 問題: Weibull分布の期待値と分散

確率変数 W が形状 α, スケール θ のWeibull分布に従っているとき, 以下が成立することを示せ:

$$E[W] = \theta\Gamma\left(1+\frac{1}{\alpha}\right), \quad \mathrm{var}(W) = \theta^2\left(\Gamma\left(1+\frac{2}{\alpha}\right) - \Gamma\left(1+\frac{1}{\alpha}\right)^2\right).$$

解答例: $W = \theta T^{1/\alpha}$, $T \sim \mathrm{Exponential}(1)$ と書ける. ゆえに,

$$E[W^m] = E[\theta^m T^{m/\alpha}] = \theta^m \int_0^\infty t^{m/\alpha} e^{-t}\,dt = \theta^m \Gamma\left(1+\frac{m}{\alpha}\right).$$

ゆえに,

$$\begin{aligned} E[W] &= \theta\Gamma\left(1+\frac{1}{\alpha}\right), \\ \mathrm{var}(W) &= E[W^2] - E[W]^2 = \theta^2\left(\Gamma\left(1+\frac{2}{\alpha}\right) - \Gamma\left(1+\frac{1}{\alpha}\right)^2\right). \end{aligned}$$

解答終

2.15 ガンマ函数の函数等式

ガンマ函数の函数等式

$$\Gamma(\alpha+1) = \alpha\Gamma(\alpha) \quad (\alpha > 0)$$

を示そう. $(-e^{-x})' = e^{-x}$ に関する部分積分を行うと,

$$\begin{aligned} \Gamma(\alpha+1) &= \int_0^\infty e^{-x}x^\alpha\,dx = \int_0^\infty (-e^{-x})'x^\alpha\,dx \\ &= [-e^{-x}x^\alpha]_0^\infty + \int_0^\infty e^{-x}\alpha x^{\alpha-1}\,dx = \alpha\int_0^\infty e^{-x}x^{\alpha-1}\,dx = \alpha\Gamma(\alpha). \end{aligned}$$

3つめの等号で部分積分を行った. 4つめの等号では $\alpha > 0$ より $\lim_{x\searrow 0} x^\alpha = 0$ となることと, $\lim_{x\to\infty} e^{-x}x^\alpha = 0$ となることを使った. 最初と最後の等号でガンマ函数の定義を使った.

2.16 ガンマ函数の函数等式のよく見る使い方

ガンマ函数の函数等式より, $k=0,1,2,\ldots$ について,

$$\Gamma(\alpha+k)=(\alpha+k-1)\Gamma(\alpha+k-1)=\cdots=(\alpha+k-1)\cdots(\alpha+2)\alpha\Gamma(\alpha)$$

なので

$$\frac{\Gamma(\alpha+k)}{\Gamma(\alpha)}=\alpha(\alpha+k-1)\cdots(\alpha+k-1).$$

ガンマ函数の函数等式はこの形で使われることがよくある.

2.17 階乗のガンマ函数表示

ガンマ函数の函数等式と易しく証明できる $\Gamma(1)=1$ から,

$$\Gamma(n+1)=n! \quad (n=0,1,2,\ldots)$$

となることを示せる. 実際, n が 0 以上の整数のとき,

$$\Gamma(n+1)=n\Gamma(n)=n(n-1)\Gamma(n-1)=\cdots=n(n-1)\cdots 2\cdot 1\,\Gamma(1)=n!.$$

ガンマ函数の函数等式を n 回使い, 最後に $\Gamma(1)=1$ を使った.

2.18 問題: 正の半整数でのガンマ函数の値

次を示せ:

$$\Gamma\left(\frac{2n+1}{2}\right)=\frac{1\cdot 3\cdots(2n-1)}{2^n}\sqrt{\pi}=\frac{(2n)!}{2^{2n}n!}\sqrt{\pi} \quad (n=0,1,2,\ldots)$$

解答例: n が0以上の整数であるとき,

$$\begin{aligned}\Gamma\left(\frac{2n+1}{2}\right)&=\frac{2n-1}{2}\Gamma\left(\frac{2n-1}{2}\right)=\frac{2n-1}{2}\frac{2n-3}{2}\Gamma\left(\frac{2n-3}{2}\right)=\cdots\\&=\frac{2n-1}{2}\frac{2n-3}{2}\cdots\frac{3}{2}\frac{1}{2}\Gamma\left(\frac{1}{2}\right)=\frac{1\cdot 3\cdots(2n-1)}{2^n}\sqrt{\pi}.\end{aligned}$$

ガンマ函数の函数等式を n 回使い, 最後にGauss積分 $\Gamma(1/2)=\sqrt{\pi}$ を使った.

さらに, 分子分母に $2\cdot 4\cdots(2n)=2^n n!$ をかけることによって,

$$\Gamma\left(\frac{2n+1}{2}\right)=\frac{1\cdot 3\cdots(2n-1)}{2^n}\sqrt{\pi}=\frac{1\cdot 3\cdots(2n-1)}{2^n}\frac{2\cdot 4\cdots(2n)}{2^n n!}\sqrt{\pi}=\frac{(2n)!}{2^{2n}n!}\sqrt{\pi}.$$

解答終

2.19 問題: 標準正規分布の偶数次のモーメント

次を示せ:

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}e^{-x^2/2}x^{2n}\,dx=\frac{2^n}{\sqrt{\pi}}\Gamma\left(\frac{2n+1}{2}\right)=\frac{(2n)!}{2^n\,n!}=1\cdot 3\cdots(2n-1) \quad (n=0,1,2,\ldots)$$

解答例: 被積分函数が偶函数であることを使ってから, $x=\sqrt{y}$ とおくと $dx=y^{-1/2}\,dy/2$ となることより,

$$\begin{aligned}&\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}e^{-x^2/2}x^{2n}\,dx=\frac{2}{\sqrt{2\pi}}\int_0^{\infty}e^{-x^2/2}x^{2n}\,dx=\frac{1}{\sqrt{2\pi}}\int_0^{\infty}e^{-y/2}y^{n-1/2}\,dy\\&=\frac{1}{\sqrt{2\pi}}\int_0^{\infty}e^{-y/2}y^{(2n+1)/2-1}\,dy=\frac{1}{\sqrt{2\pi}}2^{(2n+1)/2}\Gamma\left(\frac{2n+1}{2}\right)\\&=\frac{2^n}{\sqrt{\pi}}\Gamma\left(\frac{2n+1}{2}\right)=\frac{(2n)!}{2^n\,n!}=1\cdot 3\cdots(2n-1).\end{aligned}$$

最後から4つめの等号でガンマ函数のよく使われる形 $\int_0^{\infty}e^{-x/\theta}x^{\alpha-1}\,dx=\theta^{\alpha}\Gamma(\alpha)$ を使い, 終わりの2つの等号で正の半整数でのガンマ函数の値を使った.

解答終

注意: 一般に確率変数 X について $E[X^k]$ $(k=0,1,2,\ldots)$ を X もしくは X が従う分布の **モーメント** (積率, moment)と呼ぶ. 上の結果は標準正規分布に従う確率変数 X のモーメントについて,

$$E[X^{2n}] = \frac{(2n)!}{2^n\,n!} = 1\cdot 3\cdots(2n-1) \quad (n=0,1,2,\ldots)$$

が成立することを意味している. x^{2n+1} は奇函数になるので $E[X^{2n+1}]=0$ となる.

2.20 問題: 標準正規分布のモーメント母函数とキュムラント母函数

確率変数 X について $E[e^{tX}]$ と $\log E[e^{tX}]$ をそれぞれ X の **モーメント母函数** (moment generating function, mgf)と **キュムラント母函数** (cumulant generating function, cgf)と呼ぶ.

標準正規分布に従う確率変数 X のときのモーメント母函数とキュムラント母函数がそれぞれ次のようになることを示せ:

$$E[e^{tX}] = e^{t^2/2}, \quad \log E[e^{tX}] = \frac{t^2}{2}.$$

注意: 標準正規分布のキュムラント母函数がこのようなシンプルな形になることは, 中心極限定理(正規分布による普遍的な近似定理)や正規分布で近似したときの誤差の見積もりなどで使われる.

解答例1 1つ前の問題の結果を使うと,

$$E[e^{tX}] = \sum_{n=0}^{\infty}\frac{t^{2n}E[X^{2n}]}{(2n)!} = \sum_{n=0}^{\infty}\frac{t^{2n}}{(2n)!}\frac{(2n)!}{2^n\,n!} = \sum_{n=0}^{\infty}\frac{(t^2/2)^n}{n!} = e^{t^2/2}, \quad \log E[e^{tX}] = \frac{t^2}{2}.$$

解答終

解答例2: $tx - x^2/2 = t^2/2 - (x-t)^2/2$ であることより,

$$E[e^{tX}] = \int_{-\infty}^{\infty} e^{tx}\frac{e^{-x^2/2}}{\sqrt{2\pi}}\,dx = e^{t^2/2}\int_{-\infty}^{\infty}\frac{e^{-(x-t)^2/2}}{\sqrt{2\pi}}\,dx = e^{t^2/2}, \quad \log E[e^{tX}] = \frac{t^2}{2}.$$

解答終

2.21 問題: n-1次元単位球面の面積のガンマ函数表示

$n-1$ 次元単位球面 S^{n-1} を

$$S^{n-1} = \{\,(x_1,\ldots,x_n)\in\mathbb{R}^n \mid x_1^2+\cdots+x_n^2 = 1\,\}$$

と定める(Sはsphereの頭文字). 例えば, S^1 は平面上の半径が 1 の円周(曲線なので1次元の図形)になり, S^2 は3次元空間上の半径が 1 の球面(曲面なので2次元の図形)になる. S^{n-1} の面積を A_{n-1} がガンマ函数によって次のように表されることを示せ:

$$A_{n-1} = \frac{2\pi^{n/2}}{\Gamma(n/2)}.$$

解答例: Gauss積分の n 乗

$$I^n = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} e^{-(x_1^2+\cdots+x_n^2)}\,dx_1\cdots dx_n$$

を二通りに計算することによって, $n-1$ 次元単位球面 S^{n-1} の面積 A_{n-1} を求めよう. まず, これがGauss積分の n 乗であることから,

$$I^n = \left(\int_{-\infty}^{\infty} e^{-x^2}\,dx\right)^2 = (\sqrt{\pi})^2 = \pi^{n/2}.$$

$n-1$ 次元球面上の微小領域の面積を $d\omega$ と書くと, 半径 r の $n-1$ 次元球面上の対応する微小領域の面積はその r^{n-1} 倍の $r^{n-1}\,d\omega$ になり, さらにそれに dr の厚みを持たせて作られる n 次元の微小領域の体積はほぼそれらの積 $r^{n-1}\,dr\,d\omega$ に等しくなる. $y = r^2 = x_1^2+\cdots+x_n^2$ とおくと, $dr = d\sqrt{y} = y^{-1/2}\,dy/2$ なので,

$$r^{n-1}\,dr\,d\omega = \frac{1}{2}y^{(n-1)/2-1/2}\,dy\,d\omega = \frac{1}{2}y^{n/2-1}\,dy\,d\omega.$$

$dx_1\cdots dx_n$ に関する積分は $\frac{1}{2}y^{n/2-1}\,dy\,d\omega$ に関する積分に書き換えられるので,

$$\pi^{n/2} = I^n = \int_{S^{n-1}}\left(\int_0^{\infty} e^{-y}\frac{1}{2}y^{n/2-1}\,dy\right)d\omega = \frac{A_{n-1}}{2}\int_0^{\infty} e^{-y}y^{n/2-1}\,dy = \frac{A_{n-1}}{2}\Gamma(n/2).$$

すなわち, $A_{n-1} = 2\pi^{n/2}/\Gamma(n/2)$.

解答終

注意: 半径 r の $n-1$ 次元球面の面積は単位球面の面積の r^{n-1} 倍になるので

$$A_{n-1}r^{n-1} = \frac{2\pi^{n/2}}{\Gamma(n/2)}r^{n-1}$$

になる. 例えば,

$$\Gamma(1/2) = \sqrt{\pi}, \quad \Gamma(1) = 1, \quad \Gamma(3/2) = \frac{1}{2}\sqrt{\pi}, \quad \Gamma(2) = 1, \quad \Gamma(5/2) = \frac{3}{4}\sqrt{\pi}$$

なので,

$$\begin{aligned} A_0 &= \frac{2\pi^{1/2}}{\Gamma(1/2)} = 2, \quad A_1 r = \frac{2\pi^{2/2}}{\Gamma(2/2)}r = 2\pi r, \quad A_2 r^2 = \frac{2\pi^{3/2}}{\Gamma(3/2)}r^2 = 4\pi r^2, \\ A_3 r^3 &= \frac{2\pi^{4/2}}{\Gamma(4/2)}r^3 = 2\pi^2 r^3, \quad A_4 r^4 = \frac{2\pi^{5/2}}{\Gamma(5/2)}r^4 = \frac{8\pi^2 r^4}{3}. \end{aligned}$$

2 は集合 $S^0 = \{-1, 1\}$ の点の個数である. $2\pi r$ や $4\pi r^2$ は高校生にとってもおなじみの公式である.

2.22 $n-1$ 次元単位球面上の座標を使った面積の計算

上の問題の解答例中で単位球面上の自然な積分を $\int_{S^{n-1}} \cdots d\omega$ と書いた. その解答例のポイントは単位球面上の積分の具体的な座標表示を使わずに, そのように抽象的に書いたまま計算が最後まで遂行できることである.

しかし, 具体的に座標で表示しても同じ結果が得られるかどうがについて疑問に思った人もいるのではないだろうか?

そのような人のために以下では $n-1$ 次元単位球面上の座標を具体的に与えて面積を計算し直してみよう.

$n-1$ 次元球面

$$S^{n-1} = \{\, (x_1, \dots, x_n) \in \mathbb{R}^n \mid x_1^2 + \cdots + x_n^2 = 1 \,\}$$

の上の点は次のようにパラメトライズできる:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \cos\theta_1 \cos\theta_2 \cos\theta_3 \cdots \cos\theta_{n-1} \\ \sin\theta_1 \cos\theta_2 \cos\theta_3 \cdots \cos\theta_{n-1} \\ \sin\theta_2 \cos\theta_3 \cdots \cos\theta_{n-1} \\ \sin\theta_3 \cdots \cos\theta_{n-1} \\ \ddots \quad \vdots \\ \sin\theta_{n-1} \end{bmatrix} \qquad \begin{pmatrix} -\pi \le \theta_1 \le \pi, \\ -\frac{\pi}{2} \le \theta_2, \dots, \theta_{n-1} \le \frac{\pi}{2} \end{pmatrix}$$

これは次のような手続きで得られる:

1. (x_1, x_2) 平面上の単位円周を $(x_1, x_2) = (\cos\theta_1, \sin\theta_1)$ とパラメトライズする.
2. (x_1, x_2) 上の単位円周上の点を角度 θ_2 で x_3 軸方向に持ち上げることによって, (x_1, x_2, x_3) 空間内の単位球面をパラメトライズする.
3. (x_1, x_2, x_3) 内の単位球面の点を角度 θ_3 で x_4 軸方向に持ち上げることによって, (x_1, x_2, x_3, x_4) 空間内の単位球面をパラメトライズする.
4. 以下これを繰り返す.

これによって, 単位球面上の面積は以下のようにして計算できることがわかる.

1. (x_1, x_2) 平面上の単位円周上の弧の長さは $d\omega_1 = d\theta_1 = \cos^0\theta_1\, d\theta_1$ の積分で計算できる.
2. (x_1, x_2, x_3) 空間内の単位球面上の面積は $d\omega_1$ を $\cos\theta_2$ 倍に縮小した結果と $d\theta_2$ の積である $d\omega_2 = \cos^0\theta_1 \cos^2\theta_2\, d\theta_1\, d\theta_2$ を積分すれば計算できる.
3. (x_1, x_2, x_3, x_4) 空間内の単位球面上の面積は $d\omega_2$ が2次元の面積要素なのでそれを $\cos\theta_2$ の2乗倍した結果と $d\theta_3$ の積である $d\omega_2 = \cos^0\theta_1 \cos^1\theta_2 \cos^2\theta_3\, d\theta_1\, d\theta_2\, d\theta_3$ を積分すれば計算できる.
4. 以下これを繰り返す.

ゆえに, $n-1$ 次元単位球面 S^{n-1} 上の領域の面積は

$$d\omega_{n-1} = \cos^0\theta_1 \cos^1\theta_2 \cos^2\theta_3 \cdots \cos^{n-2}\theta_{n-1}\, d\theta_1\, d\theta_2\, d\theta_3 \cdots d\theta_{n-1}$$

を積分すれば計算できる. 特に S^{n-1} の面積 A_{n-1} は以下のように表される:

$$A_{n-1} = 2\prod_{k=1}^{n-1}\int_{-\pi/2}^{\pi/2}\cos^{k-1}\theta_k\,d\theta_k.$$

θ_1 に関する $-\pi$ から π までの積分を $-\pi/2$ から $\pi/2$ までの積分に変更し, その代わりに 2 倍してつじつまを合わせたことに注意せよ. (S^0 が2点集合であることを思い出せばこれは自然な処理であることがわかる.)

後で示すベータ函数の三角函数による表示とベータ函数をガンマ函数で表す公式

$$2\int_0^{\pi/2}(\cos\theta)^{2\alpha-1}(\sin\theta)^{2\beta-1}\,d\theta = B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha,\beta)}$$

と $\Gamma(1/2)=\sqrt{\pi}$ を使うと,

$$\int_{-\pi/2}^{\pi/2}\cos^k\theta\,d\theta = 2\int_0^{\pi/2}\cos^{k-1}\theta\,d\theta = B(k/2,1/2) = \frac{\Gamma(k/2)\sqrt{\pi}}{\Gamma((k+1)/2)}.$$

ゆえに

$$A_{n-1} = 2\frac{\sqrt{\pi}\sqrt{\pi}}{\Gamma(2/2)}\frac{\Gamma(2/2)\sqrt{\pi}}{\Gamma(3/2)}\frac{\Gamma(3/2)\sqrt{\pi}}{\Gamma(4/2)}\cdots\frac{\Gamma((n-1)/2)\sqrt{\pi}}{\Gamma(n/2)} = \frac{2\pi^{n/2}}{\Gamma(n/2)}.$$

上の問題と同じ結果が得られた.

2.23 問題: n次元単位球体の面積のガンマ函数表示

n 次元単位球体 B^n を

$$B^n = \{\,(x_1,\ldots,x_n)\in\mathbb{R}^n \mid x_1^2+\cdots+x_n^2\le 1\,\}$$

と定める(Bはballの頭文字). 例えば, B^1 は直線上の線分 $[-1,1]$ になり, B^2 は2次元平面上の半径が 1 の円盤になり, B^3 は3次元空間内の半径 1 の単位球体になる. B^n の体積を V_n がガンマ函数によって次のように表されることを示せ:

$$V_n = \frac{\pi^{n/2}}{\Gamma(n/2+1)}.$$

解答例: 半径 r の $n-1$ 次元球面の面積

$$A_{n-1}r^{n-1} = \frac{2\pi^{n/2}}{\Gamma(n/2)}r^{n-1}$$

を r について 0 から 1 まで積分すれば, n 次元単位球体の体積が得られる:

$$V_n = \int_0^1 A_{n-1}r^{n-1}\,dr = \frac{A_{n-1}}{n} = \frac{2\pi^{n/2}}{n\Gamma(n/2)} = \frac{\pi^{n/2}}{(n/2)\Gamma(n/2)} = \frac{\pi^{n/2}}{\Gamma(n/2+1)}.$$

最後の等号でガンマ函数の函数等式を使った.

解答終

注意: この問題の大幅な一般化と解答を19世紀にDirichlet氏が与えている. このノートのずっと下の方で紹介されているDirichlet積分=多変数ベータ函数の解説を参照せよ.

注意: 半径 r の n 次元球体の体積は単位球体の体積の r^n 倍になるので

$$V_nr^n = \frac{\pi^{n/2}}{\Gamma(n/2+1)}r^n$$

になる. 例えば,

$$\Gamma(3/2) = \frac{1}{2}\sqrt{\pi},\quad \Gamma(2)=1,\quad \Gamma(5/2) = \frac{3}{4}\sqrt{\pi}\Gamma(3) = 2\sqrt{\pi}\Gamma(7/2) = \frac{15}{8}\sqrt{\pi}$$

なので,

$$V_1r = \frac{\pi^{1/2}}{\Gamma(1/2+1)}r = 2r,\quad V_2r^2 = \frac{\pi^{2/2}}{\Gamma(2/2+1)}r^2 = \pi r^2,\quad V_3r^3 = \frac{\pi^{3/2}}{\Gamma(3/2+1)}r^3 = \frac{4\pi^2r^3}{3},$$
$$V_4r^4 = \frac{\pi^{4/2}}{\Gamma(4/2+1)}r^4 = \frac{\pi^2r^4}{2},\quad V_5r^5 = \frac{\pi^{5/2}}{\Gamma(5/2+1)}r^5 = \frac{8\pi^2r^5}{15}.$$

$2r$ は線分 $[-r, r]$ の長さである. πr^2 や $4\pi r^3/3$ は高校生にとってもおなじみの公式だろう.

注意: m は0以上の整数であるとし, $n = 2m$ または $n = 2m+1$ と仮定する. このとき, n 次元球体の球体の体積の公式に π^m が含まれる理由は, $n = 2m, 2m+1$ 次元の空間では互いに可換な回転軸を m 本取れるからである. 1本の回転軸ごとに1つの π が出て来る. 例えば, $m = 2, n = 5$ のとき, 5 次元空間 $\mathbb{R}^5 = \{\, (x_1, x_2, x_3, x_4, x_5) \mid x_1, x_2, x_3, x_4, x_5 \in \mathbb{R} \,\}$ における互いに可換な回転軸として, (x_1, x_2) 平面の回転軸と (x_3, x_4) 平面の回転軸の2本が取れて, 5 次元球体の体積の公式には π^2 が含まれている.

2.24 Stirlingの公式

次の階乗の近似公式を **Stirlingの公式** と呼ぶ:

$$n! = n^n e^{-n} \sqrt{2\pi n}\,(1 + \varepsilon_n) \quad (\varepsilon_n \to 0 \text{ as } n \to \infty)$$

Stirlingの公式は後で空気のごとく使われることになる.

Stirlingの公式はガンマ函数の応用として以下のように示される.

$n!/(n^n e^{-n}\sqrt{n})$ が $n \to \infty$ で $\int_{-\infty}^{\infty} e^{-y^2/2}\,dy = \sqrt{2\pi}$ に収束することを示せばよい.

$n! = \Gamma(n+1) = \int_0^\infty e^{-x} x^n\,dx$ で

$$x = n + \sqrt{n}\,y = n\left(1 + \frac{y}{\sqrt{n}}\right)$$

とおくと(これがポイント!),

$$n! = \int_{-\sqrt{n}}^{\infty} e^{-n} e^{-\sqrt{n}\,y} n^n \left(1 + \frac{y}{\sqrt{n}}\right)^n \sqrt{n}\,dy = n^n e^{-n}\sqrt{n} \int_{-\sqrt{n}}^{\infty} e^{-\sqrt{n}\,y}\left(1 + \frac{y}{\sqrt{n}}\right)^n dy.$$

そして,

$$\begin{aligned}
\log\left(e^{-\sqrt{n}\,y}\left(1 + \frac{y}{\sqrt{n}}\right)^n\right) &= n\log\left(1 + \frac{y}{\sqrt{n}}\right)^n - \sqrt{n}\,y \\
&= n\left(\frac{y}{\sqrt{n}} - \frac{y^2}{2n} + \frac{y^3}{3n\sqrt{n}} - \cdots\right) - \sqrt{n}\,y = -\frac{y^2}{2} + \frac{y^3}{3\sqrt{n}} - \cdots \\
&\to -\frac{y^2}{2} \quad (\text{as } n \to \infty)
\end{aligned}$$

ゆえに, $e^{-\sqrt{n}\,y}\bigl(1 + y/\sqrt{n}\bigr)^n \to e^{-y^2/2}$ $(n \to \infty)$.

したがって, $n \to \infty$ のとき,

$$\frac{n!}{n^n e^{-n}\sqrt{n}} = \int_{-\sqrt{n}}^{\infty} e^{-\sqrt{n}\,y}\left(1 + \frac{y}{\sqrt{n}}\right)^n dy \to \int_{-\infty}^{\infty} e^{-y^2/2} = \sqrt{2\pi}.$$

これで示すべきことが示された.

注意: 極限と積分の交換可能性が気になる読者はLebesgueの収束定理の適用を考えてみよ. もっと初等的な方法でも極限と積分の交換可能性を示せる. この手の細かいことを気にならない人はそのまま気にする必要はないが, 関連のグラフを大量に描いてみたり, 数値積分を試してみたり, 様々な試行錯誤によって理解を深める努力をした方がよい.

2.25 問題: 小さな n でもStirlingの公式による $n!$ の近似精度がそう悪くないこと

$n = 1, 2, 3, \ldots, 10$ について $n!$ と

$$\mathrm{stirling}(n) = n^n e^{-n}\sqrt{2\pi n}$$

の両方を計算し, 前者の後者による近似の相対誤差を求めよ. ここで $a > 0$ の b による近似の相対誤差の定義は $b/a - 1$ であるとする. $\mathrm{stirling}(n)$ については小数点以下第3桁まで, 相対誤差について小数点以下第5桁まで計算せよ.

拡張版解答例＋解説: $1/(12n)$ によって補正した近似公式

$$\mathrm{stirling}_1(n) = n^n e^{-n}\sqrt{2\pi n}\left(1 + \frac{1}{12n}\right)$$

についてもあわせて計算してみよう.

n	$n!$	stirling(n)	relative error	$n!$	stirling$_1(n)$	relative error
1	1	0.922	-0.07786	1	0.99898	-0.001018
2	2	1.919	-0.04050	2	1.99896	-0.000519
3	6	5.836	-0.02730	6	5.99833	-0.000279
4	24	23.506	-0.02058	24	23.99589	-0.000171
5	120	118.019	-0.01651	120	119.98615	-0.000115
6	720	710.078	-0.01378	720	719.94038	-0.000083
7	5040	4980.396	-0.01183	5040	5039.68626	-0.000062
8	40320	39902.395	-0.01036	40320	40318.04541	-0.000048
9	362880	359536.873	-0.00921	362880	362865.91796	-0.000039
10	3628800	3598695.619	-0.00830	3628800	3628684.74890	-0.000032

この結果を見れば, $n!$ のStirling公式による近似の相対誤差の絶対値は $n=2$ ですでに 5% を切っており, $n=9$ で 1% を切っていることがわかる. $1/(12n)$ で補正した場合には $n=1$ の段階ですでに相対誤差の絶対値はほぼ 0.1% になっていて非常に小さい. この事実は

$$\frac{13\sqrt{2\pi}}{12e} \approx 0.9989817596371048$$

が非常に 1 に近いことが偶然ではないことも意味している. このように, $1/(12n)$ による補正で誤差が非常に小さくなるという事実は, Stirlingの公式による階乗の近似の相対誤差の大きさがほぼ $1/(12n)$ になることを意味している.

階乗は確率の記述でよく使われるので, 階乗のシンプルな近似式であるStirlingの公式は確率が絡む議論において非常に有用である.

拡張版解答終

In [33]:

```
stirling(n) = n^n * exp(-n) * √(2π*n)
stirling1(n) = n^n * exp(-n) * √(2π*n) * (1 + 1/(12n))
for n in 1:10
    a = factorial(n)
    b = stirling(n)
    c = stirling1(n)
    @printf("%2d & %7d & %11.3f & %8.5f & %7d & %13.5f & %9.6f \\\\\n", n, a, b, b/a - 1, a, c
end
```

```
 1 &       1 &       0.922 & -0.07786 &       1 &       0.99898 & -0.001018 \\
 2 &       2 &       1.919 & -0.04050 &       2 &       1.99896 & -0.000519 \\
 3 &       6 &       5.836 & -0.02730 &       6 &       5.99833 & -0.000279 \\
 4 &      24 &      23.506 & -0.02058 &      24 &      23.99589 & -0.000171 \\
 5 &     120 &     118.019 & -0.01651 &     120 &     119.98615 & -0.000115 \\
 6 &     720 &     710.078 & -0.01378 &     720 &     719.94038 & -0.000083 \\
 7 &    5040 &    4980.396 & -0.01183 &    5040 &    5039.68626 & -0.000062 \\
 8 &   40320 &   39902.395 & -0.01036 &   40320 &   40318.04541 & -0.000048 \\
 9 &  362880 &  359536.873 & -0.00921 &  362880 &  362865.91796 & -0.000039 \\
10 & 3628800 & 3598695.619 & -0.00830 & 3628800 & 3628684.74890 & -0.000032 \\
```

In [34]:

```
13√(2π)/(12e)
```

Out[34]: 0.9989817596371048

In [35]:
```
N = 10
plot(; legend=:topleft, xtick=1:N)
scatter!(1:N, factorial; label="n!")
plot!(stirling, 1, N; label="nⁿ e⁻ⁿ √(2πn)")
title!("Stirling approximation of n!")
```

Out[35]:

In [36]:
```
N = 10
plot(; legend=:topleft, xtick=1:N)
scatter!(1:N, factorial; label="n!")
plot!(stirling, 1, N; label="nⁿ e⁻ⁿ √(2πn)")
title!("Stirling approximation of n!")
plot!(;yscale=:log10)
```

Out[36]:

2.26 Stirlingの公式とガンマ分布の正規分布近似(中心極限定理)の関係

この節では, Stirlingの公式は本質的にガンマ分布の正規分布近似(中心極限定理ともいう)から出て来ることを説明する.

スケールパラメータが $\theta = 1$ で形状パラメータが $\alpha = n + 1$ のガンマ分布 $\mathrm{Gamma}(n+1, 1)$ の確率密度函数は

$$p_n(x) = \frac{e^{-x}x^n}{\Gamma(n+1)} = \frac{e^{-x}x^n}{n!} \quad (x > 0)$$

という形になる. この $p_n(x)$ の分子の対数を $f_n(x) と書く:

$$f_n(x) = \log(e^{-x}x^n) = n\log x - x.$$

$f_n(x)$ を $x = n$ でTaylor展開してみよう.

$$f_n(n) = n\log n - n, \quad f_n'(x) = \frac{n}{x} - 1, \quad f_n'(n) = 0, \quad f_n''(x) = -\frac{n}{x^2}, \quad f_n''(n) = -\frac{1}{n}.$$

これより, x が n に近いとき,

$$f_n(x) \approx n\log n - n - \frac{(x-n)^2}{2n}.$$

ゆえに, x が n に近いとき, $f_n(x)$ の $x = n$ でのTaylor展開より,

$$e^{-n}x^n = \exp(f_n(x)) \approx \exp\left(n\log n - n - \frac{(x-n)^2}{2n}\right) = n^n e^{-n}\exp\left(-\frac{(x-n)^2}{2n}\right).$$

n が大きなとき, $e^{-n}x^n$ も $\exp(-(x-n)^2/(2n))$ も x が n から離れると急激に 0 に近付くので, それぞれの $x > 0$, $-\infty < x < \infty$ における積分は n に近い x における積分で近似される. したがって,

$$
\begin{aligned}
1 &= \int_0^\infty p_n(x)\,dx = \frac{1}{n!}\int_0^\infty e^{-n}x^n\,dx \\
&\approx \frac{1}{n!}\int_{\text{around } n} e^{-n}x^n\,dx \approx \frac{n^n e^{-n}}{n!}\int_{\text{around } n} \exp\left(-\frac{(x-n)^2}{2n}\right)dx \\
&\approx \frac{n^n e^{-n}}{n!}\int_{-\infty}^\infty \exp\left(-\frac{(x-n)^2}{2n}\right)dx = \frac{n^n e^{-n}\sqrt{2\pi n}}{n!}.
\end{aligned}
$$

これで $n! \approx n^n e^{-n}\sqrt{2\pi n}$ が再び示された.

以上の計算によってガンマ分布 $\mathrm{Gamma}(n+1,1)$ の密度函数 $p_n(x)$ が平均 n, 分散 n の正規分布の密度函数で近似されることも示せている:

$$
p_n(x) = \frac{e^{-x}x^n}{n!} \approx \frac{1}{\sqrt{2\pi n}}\exp\left(-\frac{(x-n)^2}{2n}\right).
$$

以上はかなり雑な議論だが, 議論の全体を正確に作り直すこともできるし, 以下で示すようにグラフを描くことによっても結果が正しいことを確認できる.

In [37]:

```
p_gamma(n, x) = x > 0 ? exp(n*log(x) - x - loggamma(n+1)) : zero(x)
p_gamma_normalapprox(n, x) = 1/√(2π*n) * exp(-(x - n)^2/(2n))
function plot_gamma_normal(n; kwargs...)
    x = range(n - 5√n, n + 6√n, 1000)
    P1 = plot(x, x -> p_gamma(n, x); label="Gamma(n+1,1)", kwargs...)
    plot!(x, x -> p_gamma_normalapprox(n, x); label="Normal(n,√n)", ls=:dash)
    vline!([n]; label="", c=:black, ls=:dot)
    title!("n = $n")
    P2 = plot(x, x -> p_gamma_normalapprox(n, x) - p_gamma(n, x); label="difference", kwargs..
    plot!(x, x -> p_gamma(n, 2n - x) - p_gamma_normalapprox(n, 2n - x); label="180° rotation")
    vline!([n]; label="", c=:black, ls=:dot)
    hline!([0]; label="", c=:black, ls=:dot)
    plot!(ylim=(-1.05p_gamma(n,n), 1.05p_gamma(n,n)))
    plot(P1, P2; size=(800, 250))
end
```

Out[37]:

```
plot_gamma_normal (generic function with 2 methods)
```

In [38]:

```
plot_gamma_normal(3; xtick=-20:20)
```

Out[38]:

In [39]:
```
plot_gamma_normal(10)
```

Out[39]:

In [40]:
```
plot_gamma_normal(30)
```

Out[40]:

In [41]:
```
plot_gamma_normal(100)
```

Out[41]:

3 ベータ函数

3.1 ベータ函数の定義

$\alpha, \beta > 0$ について, ベータ函数 $B(\alpha, \beta)$ を次のように定める:

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt.$$

ベータ函数のこの表示は通常の(第1種)ベータ分布だけではなく, F 分布(Fisherの F 分布)とも関係している. ベータ函数の様々な表示が正規分布に関連した重要な分布達の基礎を与える.

3.2 ベータ分布の定義

$\alpha, \beta > 0$ と仮定する.

次の確率密度函数によって定義される連続分布をパラメータ α, β の **ベータ分布** (Beta distribution)と呼び,

$$\mathrm{Beta}(\alpha, \beta)$$

と表す:

$$p(t|\alpha,\beta) = \frac{1}{B(\alpha,\beta)} t^{\alpha-1}(1-t)^{\beta-1} \quad (0 < t < 1)$$

二項分布がBernoulli試行(値が 1 と 0 のみの乱数生成の繰り返し)から得られたのと同じように, パラメータが正の整数の場合のベータ分布は一様乱数の生成の繰り返しから得られる. この事実については後で別のノートで解説する予定である. この段階ではベータ分布にも後で説明されるように直観的な意味付けがあることを覚えておけばよい. 実用的に使用される確率分布の世界は豊富かつ複雑であり, それらを単に暗記しようとすることには無理がある. 直観的な意味付けを十分に理解して先に進む方が効率的な場合が多い.

ベータ分布の密度函数は $\alpha < 0$ ならば $t \searrow 0$ のとき無限大になり, $\beta > 0$ ならば $t \nearrow 1$ のとき無限大に発散することには注意しなければいけない.

α と β の両方が大きなとき, ベータ分布は正規分布で近似される.

ベータ分布の密度函数のグラフは以下のようになる.

In [42]:
```
function plot_beta(α, β; kwargs...)
    beta = Beta(α, β)
    normal = Normal(mean(beta), std(beta))
    if α < 1 || β < 1
        x = range(eps(), 1-eps(), 1000)
        plot(x, x -> pdf(beta, x); ylim=(-0.1, 4.1), label="Beta(α, β)")
    else
        plot(x -> pdf(beta, x), 0, 1; label="Beta(α, β)")
    end
    plot!(x -> pdf(normal, x), 0, 1; label="normal approx.", ls=:dash)
    title!("α = $α,  β = $β")
    plot!(; xtick=0:0.1:1, kwargs...)
end
```

Out[42]: plot_beta (generic function with 1 method)

In [43]:
```
plot(plot(plot_beta(0.5, 2)), plot(plot_beta(2, 0.5); legend=:topleft); size=(800, 250))
```

Out[43]:

In [44]:
```
plot(plot(plot_beta(0.5, 0.5); legend=:top),
    plot(plot_beta(1, 1); legend=:bottom, ylim=(-0.05, 1.5)); size=(800, 250))
```

Out[44]:

In [45]:
```
plot(plot(plot_beta(3, 7)), plot(plot_beta(6, 14)); size=(800, 250))
```

Out[45]:

In [46]:
```
plot(plot(plot_beta(30, 70)), plot(plot_beta(120, 280)); size=(800, 250))
```

Out[46]:

3.3 問題: ベータ分布の期待値と分散

分布 $\mathrm{Beta}(\alpha, \beta)$ の期待値と分散がそれぞれ

$$\frac{\alpha}{\alpha+\beta}, \quad \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$$

となることを示せ. ただし, あとで示すことになる公式 $B(\alpha,\beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta)$ を自由に用いてよい.

解答例: T は分布 $\mathrm{Beta}(\alpha, \beta)$ に従う確率変数とする.

$$\begin{aligned} E[T^k] &= \frac{1}{B(\alpha,\beta)} \int_0^1 t^{\alpha+k-1}(1-t)^{\beta-1}\,dt = \frac{B(\alpha+k,\beta)}{B(\alpha,\beta)} \\ &= \frac{\Gamma(\alpha+k)\Gamma(\beta)}{\Gamma(\alpha+\beta+k)} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} = \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+k-1)}. \end{aligned}$$

ゆえに $E[T] = \alpha/(\alpha+\beta)$,

$$\begin{aligned} \mathrm{var}(T) &= E[T^2] - E[T]^2 = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)} - \frac{\alpha^2}{(\alpha+\beta)^2} \\ &= \frac{\alpha((\alpha+1)(\alpha+\beta) - \alpha(\alpha+\beta+1))}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \end{aligned}$$

解答終

3.4 ベータ函数は α と β を交換しても不変

$t = 1-u$ とおくと,

$$B(\beta,\alpha) = \int_0^1 t^{\beta-1}(1-t)^{\alpha-1}\,dt = \int_1^0 (1-u)^{\beta-1}u^{\alpha-1}\,(-du) = \int_0^1 u^{\alpha-1}(1-u)^{\beta-1}\,du = B(\alpha,\beta).$$

3.5 問題: ベータ分布の三角函数を使った表示

次を示せ:

$$B(\alpha,\beta) = 2\int_0^{\pi/2} (\cos\theta)^{2\alpha-1}(\sin\theta)^{2\beta-1}\,d\theta \quad (\alpha,\beta>0).$$

解答例: $\alpha,\beta>0$ と仮定する. $t=\cos^2\theta$ とおくと, $dt=-2\cos\theta\,\sin\theta\,d\theta$ なので,

$$\begin{aligned} B(\alpha,\beta) &= \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt = \int_{\pi/2}^0 (\cos\theta)^{2(\alpha-1)}(\sin\theta)^{2(\beta-1)}\,(-2\cos\theta\,\sin\theta\,d\theta) = \\ &= 2\int_0^{\pi/2} (\cos\theta)^{2\alpha-1}(\sin\theta)^{2\beta-1}\,d\theta. \end{aligned}$$

解答終

注意: 特に

$$B(1/2,1/2) = 2\int_0^{\pi/2} d\theta = \pi.$$

となることがわかる. 次の積分は **Wallis積分** と呼ばれている:

$$\int_0^{\pi/2} \sin^n\theta\,d\theta = \frac{1}{2}B(1/2,(n+1)/2).$$

3.6 ベータ函数の極限としてガンマ函数を表せること

$\alpha>0$, $b\in\mathbb{R}$, $n+b>0$ と仮定する. このとき, 次が成立している:

$$\lim_{n\to\infty} n^\alpha B(\alpha,n+b) = \Gamma(\alpha).$$

積分変数を $t=x/n$ で x に変換すると,

$$\begin{aligned} n^\alpha B(\alpha,n+b) &= n^\alpha \int_0^1 t^{\alpha-1}(1-t)^{n+b-1}\,dt \\ &= n^\alpha \int_0^n \left(\frac{x}{n}\right)^{\alpha-1}\left(1-\frac{x}{n}\right)^{n+b-1}\frac{dx}{n} = \int_0^n x^{\alpha-1}\left(1-\frac{x}{n}\right)^{n+b-1}dx \end{aligned}$$

なので, $n\to\infty$ のとき, $(1-x/n)^{n+b-1}\to e^{-x}$ となることを使うと,

$$n^\alpha B(\alpha,n+b) \to \int_0^\infty x^{\alpha-1}e^{-x}\,dx = \Gamma(\alpha).$$

例: t 分布の自由度無限大の極限の計算で使った $\sqrt{\nu}B(1/2,\nu/2)$ の $\nu\to\infty$ の極限は上の結果を使うと以下のようにして計算される:

$$\sqrt{\nu}B(1/2,\nu/2) = \sqrt{2}\sqrt{\nu/2}B(1/2,\nu/2) \to \sqrt{2}\,\Gamma(1/2) = \sqrt{2\pi}.$$

In [47]:
```
beta_approx_of_gamma(n, α) = exp(α*log(n) + logbeta(α, n))
x = range(0, 30, 1000)
plot(; legend=:topleft, yscale=:log10)
plot!(x, gamma; label="Γ(x)", ls=:dash)
for n in (10, 30, 100, 300)
    plot!(x, x -> beta_approx_of_gamma(n, x); label="n = $n")
end
title!("y = nˣ B(x, n)")
```

Out[47]:

3.7 問題: ベータ分布の極限でガンマ分布が得られること

$\alpha, \theta > 0, b \in \mathbb{R}$ と仮定する. T_n はパラメータ $\alpha, n/\theta + b > 0$ を持つベータ分布に従う確率変数であるとする:

$$T_n \sim \mathrm{Beta}(\alpha, n/\theta + b).$$

このとき, nT_n が従う分布は $n \to \infty$ でガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ に近付くことを示せ. すなわち次が成立することを示せ:

$$\lim_{n\to\infty} E[f(nT_n)] = \int_0^\infty f(x) \frac{e^{-x/\theta} x^{\alpha-1}}{\theta^\alpha \Gamma(\alpha)}\, dx.$$

解答例: 定義に戻って $E[f(nT_n)]$ を計算すると $n \to \infty$ で以下が成立することがわかる:

$$\begin{aligned}
E[f(nT_n)] &= \int_0^1 f(nt) \frac{t^{\alpha-1}(1-t)^{n/\theta+b-1}}{B(\alpha, n/\theta+b)}\, dt \\
&= \frac{1}{B(\alpha, n/\theta+b)} \int_0^n f(x) \left(\frac{x}{n}\right)^{\alpha-1} \left(1 - \frac{x}{n}\right)^{n/\theta+b-1} \frac{dx}{n} \\
&= \frac{1}{n^\alpha B(\alpha, n/\theta+b)} \int_0^n f(x) x^{\alpha-1} \left(1 - \frac{x}{n}\right)^{n/\theta+b-1} dx \\
&= \frac{1}{\theta^\alpha \underbrace{(n/\theta)^\alpha B(\alpha, n/\theta+b)}_{\to \Gamma(\alpha)}} \int_0^n f(x) \underbrace{\left(1 - \frac{x}{n}\right)^{n/\theta+b-1}}_{\to \exp(-x/\theta)} x^{\alpha-1}\, dx \\
&\to \frac{1}{\theta^\alpha \Gamma(\alpha)} \int_0^\infty f(x) e^{-x/\theta} x^{\alpha-1}\, dx.
\end{aligned}$$

2つめの等号で $t = x/n$ とおいた. これで示すべきことが示された.

解答例

注意: 正規分布モデルの統計学ではベータ分布がらみの確率分布達(例えば t 分布, F 分布)が大活躍する. 上の結果はそれらが, 正規分布またはガンマ分布がらみの確率分布(例えばχ²分布)の裾野を太くする拡張になっていることを本質的に含んでいる.

In [48]:
```
pdf_Tn(n, α, θ, b, x) = pdf(Beta(α, n/θ + b), x/n)/n

function plot_nbeta(n, α, θ, b=1)
    xmin, xmax = max(0, α*θ - 4√α*θ), α*θ + 4√α*θ
    plot(x -> pdf_Tn(n, α, θ, b, x), xmin, xmax; label="n Beta(α,n/θ+b)")
    plot!(x -> pdf(Gamma(α, θ), x), xmin, xmax; label="Gamma(α,β)", ls=:dash)
    α < 1 && plot!(; ylim=(-0.3, 3.3))
    title!("n=$n, α=$α, θ=$θ, b=$b")
end
```

Out[48]: plot_nbeta (generic function with 2 methods)

In [49]:
```
plot(plot_nbeta.((1.0, 1.5, 2.0), 0.5, 1)...; size=(800, 200), layout=(1, 3))
```

Out[49]:

In [50]:
```
plot(plot_nbeta.((3, 10, 30), 1, 2)...; size=(800, 200), layout=(1, 3))
```

Out[50]:

In [51]:
```
plot(plot_nbeta.((10, 30, 100), 3, 1)...; size=(800, 200), layout=(1, 3))
```

Out[51]:

3.8 ベータ函数の別の基本的表示

統計学的にはベータ函数の次の表示も重要である:

$$B(\alpha,\beta)=\int_0^\infty \frac{u^{\alpha-1}\,du}{(1+u)^{\alpha+\beta}}=\int_0^\infty \frac{u^{\beta-1}\,du}{(1+u)^{\alpha+\beta}}.$$

この表示は t 分布(Studentの t 分布)と関係している.

上の表示は以下のようにして証明される.

$t=u/(1+u)=1-1/(1+u)$ とおくと, $dt=du/(1+u)^2$ となるので,

$$B(\alpha,\beta)=\int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt=\int_0^\infty\left(\frac{u}{1+u}\right)^{\alpha-1}\left(\frac{1}{1+u}\right)^{\beta-1}\frac{du}{(1+u)^2}=\int_0^\infty\frac{u^{\alpha-1}\,du}{(1+u)^{\alpha+\beta}}$$

となる. ベータ函数は α と β の交換で不変なので被積分函数の分子の α を β で置き換えても等しくなる.

注意: $0<t<1$ と $0<u<\infty$ と $-\infty<x<\infty$ のあいだの変数変換

$$t=\frac{u}{1+u}=\frac{1}{1+e^{-x}}=\mathrm{logistic}(x),\qquad \frac{t}{1-t}=u=e^x,\qquad \mathrm{logit}(t)=\log\frac{t}{1-t}=\log u=x$$

は統計学に関連した計算で非常によく現れる鉄板の基本パターンの一つである. t を確率とみなすとき, $u=t/(1-t)$ をその **オッズ** と呼び, $x=\log(t/(1-t))$ をその **ロジット変換** と呼ぶ.

3.9 問題: t 分布の基礎になる公式

次を示せ:

$$\int_{-\infty}^{\infty}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}dt=\sqrt{\nu}\,B(1/2,\nu/2)\quad(\nu>0).$$

解答例: $\nu>0$ と仮定する.

$$\begin{aligned}\int_{-\infty}^{\infty}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}dt&=2\int_{0}^{\infty}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}dt=2\sqrt{\nu}\int_{0}^{\infty}(1+s^2)^{-(\nu+1)/2}\,ds\\&=\sqrt{\nu}\int_{0}^{\infty}(1+u)^{-(\nu+1)/2}\,u^{-1/2}\,du=\sqrt{\nu}\int_{0}^{\infty}\frac{u^{1/2-1}\,du}{(1+u)^{1/2+\nu/2}}=\sqrt{\nu}\,B(1/2,\nu/2).\end{aligned}$$

最初の等号で被積分函数が偶函数であることを使った. 2つめの等号で $t=\sqrt{\nu}\,s$ とおいた. 3つめの等号で $s=\sqrt{u}$, $ds=u^{-1/2}\,du/2$ とおいた. 最後の等号でベータ函数の別の基本的表示を使った.

解答終

注意: 上の計算での積分変数の変換では $t=\sqrt{\nu u}$, $u=t^2/\nu$ となっている.

3.10 t 分布の定義

$\nu>0$ と仮定する.

次の確率密度函数によって定義される連続分布を **自由度** (degree of freedom) が ν の t **分布** (**Student の** t **分布**, Student's t-distribution)と呼び,

$$\mathrm{TDist}(\nu)$$

と表す:

$$p(t|\nu)=\frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}.$$

t 分布は正規分布モデルによる平均に関する検定や推定で基本的な役目を果たすことになる. この点については後で別のノートで詳しく説明する.

3.11 t 分布の期待値と分散

$T\sim\mathrm{TDist}(\nu)$ と仮定する. $|t|\to\infty$ で自由度 ν の t 分布の確率密度函数は

$$p(t|\nu)=\frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}=O\left(|t|^{-(\nu+1)}\right)$$

のように振る舞うので, $m\geq 0$ に対する $E[|T|^m]$ が有限の値になって m 次のモーメント $E[T^m]$ が有限の値としてうまく定義されるための必要十分条件は $m<\nu$ が成立することである.

ゆえに, 期待値 $E[T]$ が定義されるための必要十分条件は $\nu>1$ となることである. t 分布の確率密度函数は偶函数なので, $\nu>1$ のとき期待値は $E[T]=0$ となる.

$\nu>1$ のとき, 分散 $E[T^2]$ が有限の値になるための必要十分条件は $\nu>2$ となることである. t 分布の分散は次のようになることを示そう:

$$\nu>2\implies\mathrm{var}(K)=\frac{\nu}{\nu-2}.$$

証明: $\nu>m\geq 0$ のとき,

$$
\begin{aligned}
E[|T|^m] &= \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{-\infty}^{\infty} |t|^m \left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2} dt \\
&= \frac{2}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{0}^{\infty} |t|^m \left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2} dt \\
&= \frac{2}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{0}^{\infty} \nu^{m/2} u^{m/2} (1+u)^{-(\nu+1)/2} \frac{1}{2}\sqrt{\nu}\,u^{-1/2}\,du \\
&= \frac{\nu^{m/2}}{B(1/2,\nu/2)} \int_{0}^{\infty} \frac{u^{(m+1)/2-1}}{(1+u)^{((m+1)+(\nu-m))/2}}\,du \\
&= \frac{\nu^{m/2} B((m+1)/2,(\nu-m)/2)}{B(1/2,\nu/2)} \\
&= \nu^{m/2} \frac{\Gamma((m+1)/2)}{\Gamma(1/2)} \frac{\Gamma((\nu-m)/2)}{\Gamma(\nu/2)}.
\end{aligned}
$$

3つめの等号で $t=\sqrt{\nu u}$ とおいた. 特に $m=2k<\nu$ $(k=0,1,2,\ldots)$ のとき,

$$
\begin{aligned}
E[T^{2k}] &= \nu^k \frac{\Gamma(1/2+k)}{\Gamma(1/2)} \frac{\Gamma(\nu/2-k)}{\Gamma(\nu/2)} \\
&= \nu^k \frac{(1/2)(3/2)\cdots((2k-1)/2)}{((\nu-2)/2)((\nu-4)/2)\cdots((\nu-2k)/2)} \\
&= \frac{1\cdot 3\cdots(2k-1)\,\nu^k}{(\nu-2)(\nu-4)\cdots(\nu-2k)}.
\end{aligned}
$$

特に $k=1$ の場合より,

$$
\mathrm{var}(T) = E[T^2] = \frac{\nu}{\nu-2}.
$$

証明終

3.12 問題: t 分布の自由度無限大の極限で標準正規分布が得られること

t 分布の確率密度函数 $p(t|\nu)$ について次を示せ:

$$
\lim_{\nu\to\infty} p(t|\nu) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}.
$$

解答例: $t\in\mathbb{R}$ であるとする.

$$
p(t|\nu) = \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)} \left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}
$$

なので, $\nu\to\infty$ のとき, $\sqrt{\nu}\,B(1/2,\nu/2)\to\sqrt{2\pi}$ かつ $(1+t^2/\nu)^{-(\nu+1)/2}\to e^{-t^2}$ となることを示せばよい.

$N\to\infty$ のとき $(1+x/N)^N\to e^x$ より, $\nu\to\infty$ とすると,

$$
\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2} = \left(1+\frac{t^2/2}{\nu/2}\right)^{-\nu/2} \left(1+\frac{t^2/2}{\nu/2}\right)^{-1/2} \to e^{-t^2/2}.
$$

$u=x/(\nu/2)$ とおくと, $du/u=dx/x$ なので,

$$
\begin{aligned}
\sqrt{\nu}\,B(1/2,\nu/2) &= \sqrt{\nu}\int_0^\infty \frac{u^{1/2-1}\,du}{(1+u)^{1/2+\nu/2}} = \frac{\sqrt{\nu}}{(\nu/2)^{1/2}} \int_0^\infty \frac{x^{1/2-1}\,dx}{(1+x/(\nu/2))^{1/2+\nu/2}} \\
&= \sqrt{2}\int_0^\infty \frac{x^{1/2-1}\,dx}{(1+x/(\nu/2))^{1/2+\nu/2}}
\end{aligned}
$$

なので $\nu\to\infty$ とすると,

$$
\sqrt{\nu}\,B(1/2,\nu/2) \to \sqrt{2}\int_0^\infty \frac{x^{1/2-1}\,dx}{e^x} = \sqrt{2}\int_0^\infty e^{-x}x^{1/2-1}\,dx = \sqrt{2}\,\Gamma(1/2) = \sqrt{2\pi}.
$$

解答終

注意: t 分布という新しい複雑に見える確率分布を定義したが, 自由度が大きな場合には, t 分布の代わりに正規分布を使った議論と実践的には(近似的には)同じ結果が得られることになる. 自由度が大体 30 程度で実践的には t 分布はほぼ正規分布だと思ってよい場合が多い. t 分布の自由度はほぼ標本サイズになることが多いので, 標本サイズが大きな場合には複雑に見える t 分布の代わりに正規分布を使っても大差ないことになる. このように, ざくっとどんぶり勘定することによって, 複雑に見える議論を単純化することは統計学を実用的に使うときに重要である.

3.13 t 分布と標準正規分布を同時プロット

In [52]:
```
function plot_tdist(ν)
    P = plot(x -> pdf(TDist(ν), x), -6, 6; label="", title="TDist($ν)")
    plot!(x -> pdf(Normal(), x), -6, 6; label="Normal()", ls=:dash)
end
```

Out[52]: plot_tdist (generic function with 1 method)

In [53]:
```
plot(plot_tdist(1), plot_tdist(2); size=(800, 250))
```

Out[53]:

In [54]:
```
plot(plot_tdist(3), plot_tdist(4); size=(800, 250))
```

Out[54]:

In [55]:
```
plot(plot_tdist(10), plot_tdist(20); size=(800, 250))
```

Out[55]:

4 ベータ函数のガンマ函数表示

ベータ函数をガンマ函数で表せることから, ガンマ分布とベータ分布の関係が得られる.

$\alpha, \beta > 0$ について次が成立している:

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

これを示すためには

$$\Gamma(\alpha)\Gamma(\beta) = \Gamma(\alpha+\beta)B(\alpha, \beta)$$

を示せばよい.

4.1 積分 $J[f]$ を導入とベータ函数のガンマ函数表示の証明

後で, ガンマ分布とベータ分布を関係付けるために, 函数 $f(z,t)$ について次の積分を考える:

$$J[f] = \int_0^\infty \int_0^\infty f\left(x+y, \frac{x}{x+y}\right) e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx\, dy.$$

このとき, 次が成立している:

$$J[f] = \int_0^\infty \left(\int_0^1 f(z,t)\, e^{-z} z^{\alpha+\beta-1}\, t^{\alpha-1}(1-t)^{\beta-1}\, dt \right) dz$$

もしもこの公式が証明されたならば, $J[f]$ の定義より, $f(z,t) = 1$ のとき,

$$\begin{aligned} J[1] &= \int_0^\infty \int_0^\infty e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx\, dy \\ &= \int_0^\infty e^{-x} x^{\alpha-1}\, dx \cdot \int_0^\infty e^{-y} y^{\beta-1}\, dy = \Gamma(\alpha)\Gamma(\beta) \end{aligned}$$

となり, 上の公式より,

$$\begin{aligned} J[1] &= \int_0^\infty \left(\int_0^1 e^{-z} z^{\alpha+\beta-1}\, t^{\alpha-1}(1-t)^{\beta-1}\, dt \right) dz \\ &= \int_0^\infty e^{-z} z^{\alpha+\beta-1}\, dz \cdot \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\, dt = \Gamma(\alpha+\beta)B(\alpha,\beta). \end{aligned}$$

以上を比較すれば $\Gamma(\alpha)\Gamma(\beta) = \Gamma(\alpha+\beta)B(\alpha,\beta)$ が得られる.

4.2 積分 $J[f]$ に関する公式の証明

前節で説明した $J[f]$ に関する公式は以下のようにして示される. 積分変数 y を $y = z - x$ によって z に変換すると, $0 < y < \infty$ と $x < z < \infty$ のあいだの一対一対応が得られるので,

$$\begin{aligned} J[f] &= \int_0^\infty \left(\int_0^\infty f(x+y, x/(x+y))\, e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dy \right) dx \\ &= \int_0^\infty \left(\int_x^\infty f(z, x/z)\, e^{-z} x^{\alpha-1} (z-x)^{\beta-1}\, dz \right) dx. \end{aligned}$$

この二重積分で (x,z) の動く範囲は $0 < x < z < \infty$ である. その範囲で (x,z) を動かすためには, まず $0 < z < \infty$ を満たす z を与えてから, $0 < x < z$ を満たす x を与えればよい. ゆえにその二重積分の積分順序を次のように交換できる:

$$J[f] = \int_0^\infty \left(\int_0^z f(z, x/z)\, e^{-z} x^{\alpha-1} (z-x)^{\beta-1}\, dx \right) dz.$$

積分変数 x を $x = zt$ によって t に変換すると, $0 < x < z$ と $0 < t < 1$ のあいだの一対一対応が得られるので,

$$J[f] = \int_0^\infty \left(\int_0^1 f(z,t)\, e^{-z} (zt)^{\alpha-1} (z(1-t))^{\beta-1}\, z\, dt \right) dz$$

この二重積分の被積分函数の $f(z,t)$ 以外の因子は z, t のそれぞれの単独変数函数の積の形になっており, 次のように整理される:

$$J[f] = \int_0^\infty \left(\int_0^1 f(z,t)\, e^{-z} z^{\alpha+\beta-1}\, t^{\alpha-1}(1-t)^{\beta-1}\, dt \right) dz.$$

これで示したい公式が示された.

4.3 問題: ガンマ分布とベータ分布の関係 (ベータ分布のガンマ分布表示)

$\alpha, \beta > 0$ と仮定する. 正の実数に値を持つ確率変数 X, Y の組 (X, Y) の確率密度函数は, ガンマ分布達 $\mathrm{Gamma}(\alpha, 1)$, $\mathrm{Gamma}(\beta, 1)$ の確率密度函数の積になっていると仮定する:

$$p(x, y) = \frac{e^{-x}x^{\alpha-1}}{\Gamma(\alpha)}\frac{e^{-y}y^{\beta-1}}{\Gamma(\beta)} = \frac{e^{-(x+y)}x^{\alpha-1}y^{\beta-1}}{\Gamma(\alpha)\Gamma(\beta)} \quad (x, y > 0)$$

この $p(x, y)$ を確率変数 X, Y の **同時確率密度函数** (simultaneous probability density function)と呼ぶ. 同時確率密度函数がこのように各々の確率変数の確率密度函数の積になっているとき, それらの確率変数達は **独立** (independent)であるという. 以上の設定を次のように書くことにする:

$$X \sim \mathrm{Gamma}(\alpha, 1), \quad Y \sim \mathrm{Gamma}(\beta, 1) \quad (\text{independent})$$

このとき,

$$X + Y \sim \mathrm{Gamma}(\alpha + \beta, 1), \quad \frac{X}{X+Y} \sim \mathrm{Beta}(\alpha, \beta) \quad (\text{independent})$$

となることを示せ. すなわち, $Z = X + Y, T = X/(X+Y) = X/Z$ とおき, 分布 $\mathrm{Gamma}(\alpha + \beta, 1)$ の確率密度函数を $p_Z(z)$ と書き, 分布 $\mathrm{Beta}(\alpha, \beta)$ の確率密度函数を $p_T(t)$ と書くとき, 次が成立することを示せ:

$$E[f(Z, T)] = \int_0^\infty \left(\int_0^1 f(z, t) p_Z(z) p_T(t)\, dt \right) dz.$$

解答例: 分布 $\mathrm{Gamma}(\alpha + \beta, 1)$ の確率密度函数 $p_Z(z)$ と分布 $\mathrm{Beta}(\alpha, \beta)$ の確率密度函数を $p_T(t)$ はそれらの定義より次の形をしている:

$$p_Z(z) = \frac{e^{-z}z^{\alpha+\beta-1}}{\Gamma(\alpha+\beta)}, \quad p_T(t) = \frac{t^{\alpha-1}(1-t)^{\beta-1}}{B(\alpha, \beta)} \quad (z > 0,\ 0 < t < 1)$$

確率変数の期待値の定義と前節の証明で使った $J[f]$ の定義を使うと,

$$\begin{aligned} E[f(Z, T)] &= \int_0^\infty \int_0^\infty f(x+y, x/(x+y)) p(x, y)\, dx\, dy \\ &= \int_0^\infty \int_0^\infty f(x+y, x/(x+y)) \frac{e^{-(x+y)}x^{\alpha-1}y^{\beta-1}}{\Gamma(\alpha)\Gamma(\beta)}\, dx\, dy = \frac{J[f]}{\Gamma(\alpha)\Gamma(\beta)} =. \end{aligned}$$

$\Gamma(\alpha)\Gamma(\beta) = \Gamma(\alpha+\beta)B(\alpha, \beta)$ と前節で示した $J[f]$ に関する公式と $p_Z(z), p_T(t)$ の定義を使うと,

$$\begin{aligned} E[f(Z, T)] &= \frac{J[f]}{\Gamma(\alpha+\beta)B(\alpha, \beta)} \\ &= \int_0^\infty \left(\int_0^1 f(z, t) \frac{e^{-z}z^{\alpha+\beta-1}}{\Gamma(\alpha+\beta)} \frac{t^{\alpha-1}(1-t)^{\beta-1}}{B(\alpha, \beta)}\, dt \right) dz \\ &= \int_0^\infty \left(\int_0^1 f(z, t) p_Z(z) p_T(t)\, dt \right) dz. \end{aligned}$$

これで示すべきことが示された.

解答終

注意: 以上においては計算を簡単にするためにガンマ分布のスケールパラメータを $\theta = 1$ とおいたが, $\theta > 0$ の一般的な場合にも,

$$X \sim \mathrm{Gamma}(\alpha, \theta) \iff \frac{X}{\theta} \sim \mathrm{Gamma}(\alpha, 1)$$

であることから,

$$X \sim \mathrm{Gamma}(\alpha, \theta), \quad Y \sim \mathrm{Gamma}(\beta, \theta) \quad (\text{independent})$$

ならば

$$X + Y \sim \mathrm{Gamma}(\alpha + \beta, \theta), \quad \frac{X}{X+Y} \sim \mathrm{Beta}(\alpha, \beta) \quad (\text{independent})$$

となることがわかる.

注意: 上の問題の結果はガンマ分布達 $\mathrm{Gamma}(\alpha, 1)$, $\mathrm{Gamma}(\beta, 1)$ に従う独立な確率変数 X, Y からガンマ分布 $\mathrm{Gamma}(\alpha+\beta, 1)$ に従う確率変数を $X+Y$ の形で作れてかつ, ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従う確率変数を $X/(X+Y)$ の形で作れることを示した.

これは, ガンマ分布達 $\mathrm{Gamma}(\alpha, 1)$, $\mathrm{Gamma}(\beta, 1)$ に従う乱数達から, ガンマ分布 $\mathrm{Gamma}(\alpha+\beta, 1)$ に従う乱数とベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従う乱数を作れることを意味している.

上のように積分の煩雑な計算をするだけだと何をやっているのかがピンとこない人が多く出て来るかもしれないが, 確率変数達の以上のような関係は乱数の生成法のあいだの関係のことだと了解しておけば結果的に何をやっているのかわかりやすくなる.

そして, 以下のようにコンピュータによる計算によって数学的結果の正しさを数値的に確認することもできる.

ガンマ分布達の乱数を大量に生成.

In [56]:
```
α, β = 3, 5
n = 10^6
X = rand(Gamma(α, 1), n)
Y = rand(Gamma(β, 1), n);
```

ガンマ函数達の乱数達 X, Y の分布の確認.

In [57]:
```
P1 = histogram(X; norm=true, alpha=0.3, bin=140, xlim=(-1, 23), label="X")
plot!(Gamma(α, 1); label="Gamma($α, 1)", lw=2, ls=:dash)

P2 = histogram(Y; norm=true, alpha=0.3, bin=140, xlim=(-1, 23), label="Y")
plot!(Gamma(β, 1); label="Gamma($β, 1)", lw=2, ls=:dash)

plot(P1, P2; size=(800, 250))
```

Out[57]:

ガンマ函数の乱数達から Z と T を作成.

In [58]:
```
Z = @. X + Y
T = @. X / Z;
```

Z, T の分布達がそれぞれガンマ分布とベータ分布になっていることの確認.

In [59]:

```
P3 = histogram(Z; norm=true, alpha=0.3, bin=140, xlim=(-1, 23), label="Z = X + Y")
plot!(Gamma(α + β, 1); label="Gamma($α + $β, 1)", lw=2, ls=:dash)

P4 = histogram(T; norm=true, alpha=0.3, bin=140, xlim=(-0.02, 1.02), label="T = X / (X + Y)")
plot!(Beta(α, β); label="Beta($α, $β)", lw=2, ls=:dash)

plot(P3, P4; size=(800, 250))
```

Out[59]:

4.4 問題: ガンマ分布の再生成

以下を示せ:

(1) X がスケールパラメータ 1 のガンマ分布 $\mathrm{Gamma}(\alpha, 1)$ に従う確率変数ならば θX はスケールパラメータ θ のガンマ分布に従う.

(2) **ガンマ分布の再生成:** 独立な確率変数達 X, Y のそれぞれが同じスケールパラメータのガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$, $\mathrm{Gamma}(\beta, \theta)$ に従うとき, $X + Y$ は形状パラメータがそれらの和であるようなガンマ分布 $\mathrm{Gamma}(\alpha + \beta, \theta)$ に従う.

解答例: (1) X がスケールパラメータ 1 のガンマ分布 $\mathrm{Gamma}(\alpha, 1)$ に従う確率変数であるとき,

$$\begin{aligned} E[f(\theta X)] &= \int_0^\infty f(\theta x) \frac{e^{-x} x^{\alpha-1}}{\Gamma(\alpha)}\, dx \\ &= \int_0^\infty f(y) \frac{e^{-y/\theta} (y/\theta)^{\alpha-1}}{\Gamma(\alpha)} \frac{dy}{\theta} = \int_0^\infty f(y) \frac{e^{-y/\theta} y^{\alpha-1}}{\theta^\alpha \Gamma(\alpha)}\, dy. \end{aligned}$$

ゆえに $Y = \theta X$ はスケール θ のガンマ分布 $\Gamma(\alpha, \theta)$ に従う.

(2) 独立な確率変数達 X, Y のそれぞれが同じスケールパラメータのガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$, $\mathrm{Gamma}(\beta, \theta)$ に従っていると仮定する.

このとき, (1)の結果を使うと, X, Y はスケールパラメータが 1 のガンマ分布達 $\mathrm{Gamma}(\alpha, 1)$, $\mathrm{Gamma}(\beta, 1)$ にそれぞれが従う独立な確率変数達 X', Y' によって, $X = \theta X'$, $Y = \theta Y'$ と表される.

ガンマ分布とベータ分布の関係の問題より, $X' + Y'$ はガンマ分布 $\mathrm{Gamma}(\alpha + \beta, 1)$ に従うことがわかる.

ゆえに(1)の結果より, $X + Y = \theta(X' + Y')$ はガンマ分布 $\mathrm{Gamma}(\alpha + \beta, \theta)$ に従う.

解答終

4.5 積分 $K[f]$ を導入とベータ函数のガンマ函数表示の別証明

函数 $g(z, u)$ に関する以下の積分を使って別証明を作っておく:

$$K[g] = \int_0^\infty \int_0^\infty g\left(x + y, \frac{x}{y}\right) e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx\, dy.$$

このとき, 次が成立している:

$$K[g] = \int_0^\infty \int_0^\infty g(z, u)\, e^{-z} z^{\alpha+\beta-1} \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\, du\, dz.$$

もしもこの公式が証明されたならば, $K[g]$ の定義より, $g(z, u) = 1$ のとき,

$$
\begin{aligned}
K[1] &= \int_0^\infty \int_0^\infty e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx\, dy \\
&= \int_0^\infty e^{-x} x^{\alpha-1}\, dx \cdot \int_0^\infty e^{-y} y^{\beta-1}\, dy = \Gamma(\alpha)\Gamma(\beta)
\end{aligned}
$$

となり, 上の公式より,

$$
\begin{aligned}
K[1] &= \int_0^\infty \left(\int_0^\infty e^{-z} z^{\alpha+\beta-1} \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\, dt \right) dz \\
&= \int_0^\infty e^{-z} z^{\alpha+\beta-1}\, dz \cdot \int_0^\infty \frac{u^{\alpha-1}\, du}{(1+u)^{\alpha+\beta}} = \Gamma(\alpha+\beta) B(\alpha,\beta).
\end{aligned}
$$

以上を比較すれば $\Gamma(\alpha)\Gamma(\beta) = \Gamma(\alpha+\beta)B(\alpha,\beta)$ が得られる.

4.6 積分 $K[g]$ に関する公式の証明

前節で説明した $K[g]$ に関する公式は以下のようにして示される. 積分変数 x を $x = yu$ によって u に変換すると, $0 < x < \infty$ と $0 < u < \infty$ のあいだの一対一対応が得られるので,

$$
\begin{aligned}
K[g] &= \int_0^\infty \left(\int_0^\infty g(x+y, x/y)\, e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx \right) dy \\
&= \int_0^\infty \left(\int_0^\infty g((1+u)y, u)\, e^{-(1+u)y} u^{\alpha-1} y^{\alpha+\beta-1}\, du \right) dy.
\end{aligned}
$$

この二重積分の積分順序を次のように交換すると,

$$
K[g] = \int_0^\infty \left(\int_0^\infty g((1+u)y, u)\, e^{-(1+u)y} u^{\alpha-1} y^{\alpha+\beta-1}\, dy \right) du.
$$

積分変数 y を $y = z/(1+u)$ によって z に変換すると,

$$
K[g] = \int_0^\infty \left(\int_0^\infty g(z,u)\, e^{-z} u^{\alpha-1} \left(\frac{z}{1+u} \right)^{\alpha+\beta-1} \frac{dz}{1+u} \right) du
$$

この二重積分の被積分函数の $g(z,u)$ 以外の因子は z, u のそれぞれの単独変数函数の積の形になっており, 次のように整理される:

$$
K[g] = \int_0^\infty \int_0^\infty g(z,u)\, e^{-z} z^{\alpha+\beta-1} \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\, dz\, du.
$$

これで示したい公式が示された.

$J[f]$ に関する結果を使った別証明: $f(z,t) = g(z, t/(1-t))$ とおくと, $u = t/(1-t) \iff t = u/(1+u)$ より, $g(z,u) = f(z, u/(1+u))$ なので,

$$
\begin{aligned}
K[g] = J[f] &= \int_0^\infty \int_0^\infty f\left(x+y, \frac{x}{x+y}\right)\, e^{-(x+y)} x^{\alpha-1} y^{\beta-1}\, dx\, dy \\
&= \int_0^\infty \left(\int_0^1 f(z,t)\, e^{-z} z^{\alpha+\beta-1}\, t^{\alpha-1} (1-t)^{\beta-1}\, dt \right) dz.
\end{aligned}
$$

$t = u/(1+u) = 1 - 1/(1+u)$, $dt = du/(1+u)^2$ とおくと, $0 < t < 1 \iff 0 < u < \infty$ なので,

$$
\begin{aligned}
K[g] = J[f] &= \int_0^\infty \left(\int_0^\infty f(z, u/(1+u))\, e^{-z} z^{\alpha+\beta-1} \left(\frac{u}{1+u} \right)^{\alpha-1} \left(\frac{1}{1+u} \right)^{\beta-1} \frac{du}{(1+u)^2} \right) dz \\
&= \int_0^\infty \left(\int_0^\infty g(z,u)\, e^{-z} z^{\alpha+\beta-1} \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\, du \right) dz \\
&= \int_0^\infty \int_0^\infty g(z,u)\, e^{-z} z^{\alpha+\beta-1} \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\, dz\, du.
\end{aligned}
$$

4.7 ベータプライム分布の定義

$\alpha, \beta > 0$ であると仮定する. 確率密度函数

$$p(u|\alpha,\beta) = \frac{1}{B(\alpha,\beta)}\frac{u^{\alpha-1}}{(1+u)^{\alpha,\beta}} \quad (u>0)$$

によって定義される連続分布をパラメータ (α,β) の **ベータプライム分布** (Beta prime distribution)と呼び,

$$\mathrm{BetaPrime}(\alpha,\beta)$$

と表す.

ベータプライム分布はオッズに座標変換したベータ分布である. $T \sim \mathrm{Beta}(\alpha,\beta)$ のとき, そのオッズ $U = T/(1-T)$ は分布 $\mathrm{BetaPrime}(\alpha,\beta)$ に従い, 逆に $U \sim \mathrm{BetaPrime}(\alpha,\beta)$ のとき, $T = U/(1+U)$ は分布 $\mathrm{Beta}(\alpha,\beta)$ に従う. この結果は本質的にベータ分布の2つの異なる表示の存在

$$B(\alpha,\beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,dt = \int_0^\infty \frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}\,du$$

と同じことである.

ベータプライム分布と以下で説明する F 分布はスケール変換の違いを除けば同じ分布である. 一般に F 分布の方が使われることが多いように思われるので, 詳しい説明は F 分布の方ですることにし, ベータプライム分布については定義をするだけで詳しい説明を省略することにする.

In [60]:

```
PP = []
α = 0.5
for β in  (1.5, 5, 15)
    bp = BetaPrime(α, β)
    P = plot(bp, 0, α/β + 4√(α*(α+β)/β^2); label="BetaPrime(α, β)")
    α < 1 && plot!(; ylim = (-0.2, 4.2))
    title!("α = $α, β = $β")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[60]:

In [61]:

```
PP = []
α = 2
for β in  (1.5, 5, 15)
    bp = BetaPrime(α, β)
    P = plot(bp, 0, α/β + 4√(α*(α+β)/β^2); label="BetaPrime(α, β)")
    α < 1 && plot!(; ylim = (-0.2, 4.2))
    title!("α = $α, β = $β")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[61]:

4.8 F 分布の定義

$\nu_1, \nu_2 > 0$ であると仮定する. 確率密度函数

$$p(w|\nu_1,\nu_2) = \frac{1}{B(\nu_1/2,\nu_2/2)}\left(\frac{\nu_1 w}{\nu_1 w+\nu_2}\right)^{\nu_1/2}\left(1-\frac{\nu_1 w}{\nu_1 w+\nu_2}\right)^{\nu_2/2} w^{-1} \quad (w>0)$$

によって定義される $w>0$ に関する連続分布を自由度 (ν_1,ν_2) の F **分布** (F-distribution)と呼び,

$$\mathrm{FDist}(\nu_1,\nu_2)$$

と表す. F 分布はベータプライム分布のスケール変換に過ぎないが, こちらの方がよく使われているように思われる.

4.9 F 分布とベータ分布やガンマ分布やχ²分布との関係

$\nu_1,\nu_2,\theta>0$ であるとし,

$$X \sim \mathrm{Gamma}(\nu_1/2,\theta), \quad Y \sim \mathrm{Gamma}(\nu_2/2,\theta) \quad (\text{independent})$$

のとき,

$$T = \frac{X}{X+Y}, \quad W = \frac{\nu_2}{\nu_1}\frac{T}{1-T} = \frac{X/\nu_1}{Y/\nu_2}$$

とおくと,

$$T \sim \mathrm{Beta}(\nu_1/2,\nu_2/2), \quad W \sim \mathrm{FDist}(\nu_1,\nu_2).$$

特に $\theta=1/2$ のとき, $\mathrm{Chisq}(\nu)=\mathrm{Gamma}(\nu/2,2)$ だったので,

$$X \sim \mathrm{Chisq}(\nu_1), \quad Y \sim \mathrm{Chisq}(\nu_2) \quad (\text{independent})$$

の場合にも上と同じ結論が成立する.

証明: F 分布の確率密度函数の形は煩雑に見えるので, 見たことがあるよりシンプルな式に変形しよう. オッズ u と確率 t の関係

$$u = \frac{\nu_1}{\nu_2}w, \quad t = \frac{u}{1+u} = \frac{\nu_1 w}{\nu_1 w+\nu_2} = \frac{\nu_1 w}{\nu_1 w+\nu_2}, \quad 1-t = \frac{1}{1+u}$$

を念頭に置きながら整理し直すと, F 分布の確率密度函数は以下のように変形される:

$$\begin{aligned} p(w|\nu_1,\nu_2) &= \frac{1}{B(\nu_1/2,\nu_2/2)}\left(\frac{\nu_1 w/\nu_2}{1+\nu_1 w/\nu_2}\right)^{\nu_1/2}\left(\frac{1}{1+\nu_1 w/\nu_2}\right)^{\nu_2/2}\left(\frac{\nu_1}{\nu_2}w\right)^{-1}\frac{\nu_2}{\nu_1} \\ &= \frac{1}{B(\nu_1/2,\nu_2/2)}\frac{(\nu_1 w/\nu_2)^{\nu_1-1}}{(1+\nu_1 w/\nu_2)^{(\nu_1+\nu_2)/2}}\frac{\nu_2}{\nu_1} \end{aligned}$$

これより,

$$w = \frac{\nu_1}{\nu_2}u, \quad \alpha = \frac{\nu_1}{2}, \quad \beta = \frac{\nu_1}{2}$$

とおくと,

$$p(w|\nu_1,\nu_2)\,dw = \frac{1}{B(\nu_1/2,\nu_2/2)}\frac{u^{\alpha-1}\,du}{(1+u)^{\alpha+\beta}}.$$

これはすでに何度も見たことがある形の式である! ゆえに $u=t/(1-t)$ とおくと,

$$\frac{u^{\alpha-1}\,du}{(1+u)^{\alpha+\beta}} = t^{\alpha-1}(1-t)^{\beta-1}\,dt$$

となることをすでに知っており,

$$w = \frac{\nu_1}{\nu_2}u = \frac{\nu_1}{\nu_2}\frac{t}{1-t}, \quad \alpha = \frac{\nu_1}{2}, \quad \beta = \frac{\nu_1}{2}$$

とおくと, 次が成立することがわかった:

$$p(w|\nu_1,\nu_2)\,dw = \frac{1}{B(\nu_1/2,\nu_2/2)}t^{\alpha-1}(1-t)^{\beta-1}\,dt.$$

この公式は, ベータ分布に従う確率変数 $T \sim \mathrm{Beta}(\alpha,\beta)$ について

$$W = \frac{\nu_1}{\nu_2}\frac{T}{1-T} = \frac{T/\nu_1}{(1-T)/\nu_2}, \alpha = \frac{\nu_1}{2}, \quad \beta = \frac{\nu_1}{2}$$

とおくと, $W \sim \mathrm{FDist}(\nu_1, \nu_2)$ が成立することを意味している.

さらに,

$$X \sim \mathrm{Gamma}(\alpha, \theta), \quad Y \sim \mathrm{Gamma}(\beta, \theta) \quad \text{(independent)}$$

のとき,

$$X + Y \sim \mathrm{Gamma}(\alpha + \beta, \theta), \quad \frac{X}{X+Y} \sim \mathrm{Beta}(\alpha, \beta) \quad \text{(independent)}$$

となることがすでにわかっているので, $T = X/(X+Y)$ とおくと,

$$W = \frac{T/\nu_1}{(1-T)/\nu_2} = \frac{X/\nu_1}{Y/\nu_2} \alpha = \frac{\nu_1}{2}, \quad \beta = \frac{\nu_1}{2}$$

とおくと, $W \sim \mathrm{FDist}(\nu_1, \nu_2)$ となることもわかる.

特に $\theta = 1/2$ とおくと,

$$X \sim \mathrm{Chisq}(\nu_1), \quad Y \sim \mathrm{Gamma}(\nu_2) \quad \text{(independent)}$$

ならば

$$W = \frac{X/\nu_1}{Y/\nu_2} \sim \mathrm{FDist}(\nu_1, \nu_2)$$

となることがわかる.

証明終

注意: $X_\nu \sim \mathrm{Chisq}(\nu)$ のとき, $\nu \to \infty$ で X_ν/ν の分布は 1 に集中して行く.

In [62]:

```
PP = []
for ν in (10, 100, 1000)
    P = plot(x -> pdf(Chisq(ν), ν*x)*ν, 0, 2; label="", title="Chisq($ν)/$ν")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[62]:

ゆえに $X_1 \sim \mathrm{Chisq}(\nu_1)$, $X_2 \sim \mathrm{Chisq}(\nu_2)$ でそれらが独立なとき, $W = (X_1/\nu_1)/(X_2/\nu_2)$ が従う分布は ν_2 が大きなとき, X_1/ν_1 がしたがう分布で近似される. ν_2 が大きくないとき, F 分布は裾が太い分布になる.

In [63]:

```
PP = []
ν₁ = 4
for ν₂ in  (3, 10, 30)
    P = plot(FDist(ν₁, ν₂), 0, 1+5√(2/ν₁); label="FDist(ν₁, ν₂)")
    plot!(x → pdf(Chisq(ν₁), ν₁*x)*ν₁, 0, 1+5√(2/ν₁); label="Chisq(ν₁)/ν₁", ls=:dash)
    ν₁ < 2 && plot!(; ylim = (-0.1, 1.1))
    title!("ν₁ = $ν₁, ν₂ = $ν₂")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[63]:

$T \sim \mathrm{TDist}(\nu) \implies T^2 \sim \mathrm{Fdist}(1, \nu)$ でかつ $Z \sim \mathrm{Normal}(0, 1) \implies Z^2 \sim \mathrm{Chisq}(1)$ なので, 上で述べた結果は t 分布 $\mathrm{TDist}(\nu)$ が ν を大きくすると標準正規分布に近付くことや, ν が大きくないとき裾が太い分布になることの一般化になっている.

In [64]:

```
PP = []
for ν in  (1, 3, 10)
    P = plot(FDist(1, ν), 0, 5; label="FDist(1, ν)")
    plot!(x → pdf(TDist(ν), √x)/√x, 0, 5; label="TDist(ν)²", ls=:dash)
    plot!(x → pdf(Chisq(1), x), 0, 5; label="Chisq(1)", ls=:dashdot)
    plot!(; ylim = (-0.1, 1.1))
    title!("ν = $ν")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[64]:

4.10 問題: ガンマ函数のGaussの乗法公式

ベータ函数のガンマ函数表示とガンマ函数の函数等式とベータ函数の極限でガンマ函数を表せることを用いて, 次の公式を証明せよ:

$$\lim_{n\to\infty} \frac{n^\alpha\, n!}{\alpha(\alpha+1)\cdots(\alpha+n)} = \Gamma(\alpha).$$

解答例: ガンマ函数の函数等式より $\Gamma(\alpha+n+1) = (\alpha+n)\cdots(\alpha+1)\alpha\Gamma(\alpha)$ なので, $\Gamma(n+1) = n!$ とベータ函数のガンマ函数表示より,

$$n^\alpha B(\alpha, n+1) = \frac{n^\alpha\Gamma(\alpha)\Gamma(n+1)}{\Gamma(\alpha+n+1)} = \frac{n^\alpha\Gamma(\alpha)n!}{(\alpha+n)\cdots(\alpha+1)\alpha\Gamma(\alpha)} = \frac{n^\alpha\, n!}{\alpha(\alpha+1)\cdots(\alpha+n)}.$$

ベータ函数の極限でガンマ函数が表せることより, $n \to \infty$ のとき,

$$\frac{n^\alpha\, n!}{\alpha(\alpha+1)\cdots(\alpha+n)} = n^\alpha B(\alpha, n+1) \to \Gamma(\alpha)$$

が得られる.

解答終

注意: 上の公式をガンマ函数に関する **Gaussの乗法公式** と呼ぶ. 上の公式の極限は α が0以下の整数以外の任意の複素数であっても収束している.

4.11 問題: $\sin$ の無限積表示

$f_\alpha(x)$ は $\cos(\alpha x)$ の $-\pi \leq x \leq \pi$ への制限の定義域を周期 2π で $\mathbb{R}$ 全体に拡張して得られる連続函数であるとする. このとき, $f_\alpha(x)$ の周期 2π でのFourier級数は $f_n(x)$ に収束することを認めて, 次の $\sin$ の無限積表示を示せ:

$$\sin(\pi\alpha) = \pi\alpha \prod_{k=1}^{\infty} \left(1 - \frac{\alpha^2}{k^2}\right).$$

解答例: 偶函数 $f_\alpha(x)$ の周期 2π のFourier級数を計算しよう. $k = 0, 1, 2, \ldots$ について,

$$\begin{aligned} a_k := \int_{-\pi}^{\pi} f_\alpha(x)\cos(kx)\,dx &= \int_{-\pi}^{\pi} \cos(\alpha x)\cos(kx)\,dx = \frac{1}{2}\int_{-\pi}^{\pi} (\cos((\alpha+k)x) + \cos((\alpha-k)x))\,dx \\ &= \frac{1}{2}\left[\frac{\sin((\alpha+k)x)}{\alpha+k} + \frac{\sin((\alpha-k)x)}{\alpha-k}\right]_{-\pi}^{\pi} = (-1)^k \sin(\pi\alpha)\frac{2\alpha}{\alpha^2-k^2}. \end{aligned}$$

$\int_{-\pi}^{\pi} \cos^2(0x)\,dx = 2\pi$ でかつ $k = 1, 2, \ldots$ のとき

$$\int_{-\pi}^{\pi} \cos^2(kx)\,dx = \int_{-\pi}^{\pi} \frac{\cos(2kx)+1}{2}\,dx = \int_{-\pi}^{\pi} \frac{1}{2}\,dx = \pi.$$

ゆえに, $-\pi \leq x \leq \pi$ のとき,

$$\cos(\alpha x) = f_\alpha(x) = \frac{a_0}{2\pi} + \frac{1}{\pi}\sum_{k=1}^{\infty} a_k \cos(kx) = \sin(\pi\alpha)\left(\frac{1}{2\pi}\frac{2}{\alpha} + \frac{1}{\pi}\sum_{k=1}^{\infty}(-1)^k \frac{2\alpha}{\alpha^2-k^2}\cos(kx)\right).$$

両辺を $\sin(\pi\alpha)/\pi$ で割って, $x = \pi$ とおくと, $\cos(k\pi) = (-1)^k$ より,

$$\pi\cot(\pi\alpha) = \frac{1}{s} + \sum_{k=1}^{\infty} \frac{2\alpha}{\alpha^2-k^2}. \tag{$*$}$$

$\sin$ の無限積表示

$$\frac{\sin(\pi\alpha)}{\pi\alpha} = \prod_{k=1}^{\infty}\left(1 - \frac{\alpha^2}{k^2}\right)$$

を証明しよう. $\alpha \to 0$ とするとこの両辺はどちらも 1 に等しくなる. ゆえに両辺の対数の導函数が等しければ, 両辺が等しいことがわかる. 左辺の対数の導函数は

$$\frac{\pi\cos(\pi\alpha)}{\sin(\pi\alpha)} - \frac{\pi}{\pi\alpha} = \pi\cot(\pi\alpha) - \frac{1}{\alpha}.$$

右辺の対数の導函数は

$$\sum_{k=1}^{\infty} \frac{-2\alpha/k^2}{1-\alpha^2/k^2} = \sum_{k=1}^{\infty} \frac{2\alpha}{\alpha^2-k^2}.$$

ゆえに$(*)$より, 両辺の対数の導函数は互いに等しい. これで $\sin$ の無限積表示が証明された.

解答終

4.12 問題: Wallisの公式

Gaussの乗法公式と $\sin$ の無限積表示

$$\lim_{n\to\infty} \frac{n^\alpha\, n!}{\alpha(\alpha+1)\cdots(\alpha+n)} = \Gamma(\alpha), \quad \prod_{k=1}^{\infty}\left(1 - \frac{\alpha^2}{k^2}\right) = \frac{\sin(\pi\alpha)}{\pi\alpha}$$

を使って, 次の公式を示せ:

$$\lim_{n\to\infty} \frac{2^{2n}(n!)^2}{\sqrt{n}\,(2n)!} = \sqrt{\pi}, \quad \prod_{k=1}^{\infty} \frac{2k\cdot 2k}{(2k-1)(2k+1)} = \frac{\pi}{2}.$$

これらを**Wallisの公式** と呼ぶ.

解答例: $\alpha = 1/2$ とおけばよい. 分子分母に $2 \cdot 4 \cdots (2n) = 2^n n!$ をかけることによって,

$$\frac{1}{2}\left(\frac{1}{2}+1\right)\cdots\left(\frac{1}{2}+n\right) = \frac{1\cdot 3\cdots(2n-1)}{2^n}\left(\frac{1}{2}+n\right) = \frac{(2n)!}{2^{2n}n!}\left(\frac{1}{2}+n\right).$$

Gaussの乗法公式で極限を取る前の式で $\alpha = 1/2$ とおくと,

$$\frac{\sqrt{n}\,n!}{(1/2)(1/2+1)\cdots(1/2+n)} = \frac{\sqrt{n}\,2^{2n}(n!)^2}{(2n)!(1/2+n)} = \frac{2^{2n}(n!)^2}{\sqrt{n}\,(2n)!}\frac{n}{1/2+n}$$

なので,

$$\lim_{n\to\infty}\frac{2^{2n}(n!)^2}{\sqrt{n}\,(2n)!} = \Gamma(1/2) = \sqrt{\pi}.$$

sin の無限積表示の各因子は $\alpha = 1/2$ とおくと,

$$1 - \frac{(1/2)^2}{k^2} = \frac{(2k)^2-1}{(2k)^2} = \frac{(2k-1)(2k+1)}{2k\cdot 2k}$$

になるので,

$$\prod_{k=1}^{\infty}\frac{(2k-1)(2k+1)}{2k\cdot 2k} = \frac{\sin(\pi/2)}{\pi/2} = \frac{2}{\pi}.$$

これの逆数を取ればもう1つのWallisの公式も得られる.

解答終

以下はWallisの公式の数値的な確認. 階乗を数値計算で扱うときには常に $\log n!$ を扱うようにし, かけ算やわり算を直接的に使わずに対数を取ったものの和や差を計算して, 最後に指数函数を作用させる必要がある.

In [65]:
```
n = 10^6
@show n
@show exp(2n*log(2) + 2logfactorial(n) - (1/2)*log(n) - logfactorial(2n))^2
@show 2prod(2k*2k/((2k-1)*(2k+1)) for k in 1:n);
```

```
n = 1000000
exp((((2n) * log(2) + 2 * logfactorial(n)) - (1 / 2) * log(n)) - logfactorial(2n)) ^ 2 = 3.14159
34154213083
2 * prod((((2k) * (2k)) / ((2k - 1) * (2k + 1)) for k = 1:n)) = 3.141591868192149
```

4.13 問題: ガンマ函数の相反公式(reflection formula)

sin の無限積表示

$$\sin(\pi\alpha) = \pi\alpha\prod_{k=1}^{\infty}\left(1-\frac{\alpha^2}{k^2}\right)$$

を使って, ガンマ函数のGaussの乗法公式から次の公式を導け:

$$B(\alpha, 1-\alpha) = \Gamma(\alpha)\Gamma(1-\alpha) = \frac{\pi}{\sin(\pi\alpha)}.$$

これをガンマ函数の **相反公式** (reflection formula)と呼ぶ.

解答例: 相反公式の両辺に α をかけて, $\alpha\Gamma(\alpha) = \Gamma(1+\alpha)$ を使った後に, 逆数を取ることによって得られる次の公式を示せば十分である:

$$\frac{1}{\Gamma(1+\alpha)\Gamma(1-\alpha)} = \frac{\sin(\pi\alpha)}{\pi\alpha}.$$

sin の無限積表示を認めて使うと, この公式は次と同値である:

$$\frac{1}{\Gamma(1+\alpha)\Gamma(1-\alpha)} = \prod_{k=1}^{\infty}\left(1-\frac{\alpha^2}{k^2}\right).$$

これをGaussの乗法公式から導けばよい.

Gaussの情報公式より, 次の式は $n \to \infty$ で $1/(\Gamma(1+\alpha)\Gamma(1-\alpha))$ に収束する:

$$
\begin{aligned}
&\frac{(1+\alpha)(2+\alpha)\cdots(n+1+\alpha)}{n^{1+\alpha}\,n!}\frac{(1-\alpha)(2-\alpha)\cdots(n+1-\alpha)}{n^{1-\alpha}\,n!} \\
&= \frac{(n+1)^2}{n^2}\left(1+\frac{\alpha}{1}\right)\left(1+\frac{\alpha}{2}\right)\cdots\left(1+\frac{\alpha}{n+2}\right)\left(1-\frac{\alpha}{1}\right)\left(1-\frac{\alpha}{2}\right)\cdots\left(1-\frac{\alpha}{n+2}\right) \\
&= \frac{(n+1)^2}{n^2}\prod_{k=1}^{n+1}\left(1-\frac{\alpha^2}{k^2}\right)
\end{aligned}
$$

最初の等号で, 分母の n^α と $n^{-\alpha}$ をキャンセルさせて, 分子分母を $(n+1)! \times (n+1)!$ で割った. 上の最後の式は $n \to \infty$ で次に

$$
\prod_{k=1}^{\infty}\left(1-\frac{\alpha^2}{k^2}\right)
$$

に収束する. これで示すべきことが示された.

解答終

4.14 ディガンマ函数, トリガンマ函数, ポリガンマ函数, Hurwitzのゼータ函数

統計学関係の計算ではガンマ函数やベータ函数の導函数達が必要になる場合がある. ベータ函数はガンマ函数で表示されるのでベータ函数の偏導函数の計算はガンマ函数の導函数の計算に帰着する. ガンマ函数の導函数が必要になる場合として, ガンマ分布の最尤法や二項分布モデルやPoisson分布モデルのベイズ統計における情報量規準の計算などがある.

コンピュータでの基本特殊函数ライブラリの中にはガンマ函数の導函数の計算のために必要な **ディガンマ函数** (digamma function), **トリガンマ函数** (trigamma function), **ポリガンマ函数** (polygamma function)が含まれている. それらはガンマ函数そのものではなく, ガンマ函数の対数の導函数として定義される. ディガンマ函数 $\psi(\alpha)$ は

$$
\psi(\alpha) = \frac{d}{d\alpha}\log\Gamma(\alpha) = \frac{\Gamma'(\alpha)}{\Gamma(\alpha)}
$$

と定義される. これの導函数

$$
\psi'(\alpha) = \frac{\Gamma(\alpha)\Gamma''(\alpha) - \Gamma'(\alpha)^2}{\Gamma(\alpha)^2}
$$

をトリガンマ函数と呼び, 高階の導函数 $\psi^{(k)}(\alpha)$ をポリガンマ函数と呼ぶ.

ガンマ函数の函数等式 $\gamma(\alpha+1) = \alpha\Gamma(\alpha)$ の両辺の対数を微分することによって,

$$
\psi(\alpha+1) = \psi(\alpha) + \frac{1}{\alpha},\ \psi'(\alpha+1) = \psi'(\alpha) - \frac{1}{\alpha^2},\ \psi''(\alpha+1) = \psi''(\alpha) + \frac{2}{\alpha^3},\ \ldots \tag{$*$}
$$

ガンマ函数のGaussの乗法公式

$$
\lim_{n\to\infty}\frac{n^\alpha\,n!}{\alpha(\alpha+1)\cdots(\alpha+n)} = \Gamma(\alpha).
$$

と

$$
\frac{d}{d\alpha}\log\frac{n^\alpha\,n!}{\alpha(\alpha+1)\cdots(\alpha+n)} = \log n - \sum_{k=0}^{n}\frac{1}{\alpha+k}
$$

より,

$$
\psi(\alpha) = \lim_{n\to\infty}\left(\log n - \sum_{k=0}^{n}\frac{1}{\alpha+k}\right),\ \psi'(\alpha) = \sum_{k=0}^{\infty}\frac{1}{(\alpha+k)^2},\ \psi''(\alpha) = -2\sum_{k=0}^{\infty}\frac{1}{(\alpha+k)^3},\ \ldots \tag{$**$}
$$

この($**$)から上の($*$)が成立することを確認することもできる.

調和級数 $\sum_{n=1}^{N} 1/n$ は $N \to \infty$ で無限大に発散するが, それから $\log N$ を引いたものは有限の値 $\gamma \approx 0.5772$ に収束することが知られている. この γ を **Eulerの** γ (オイラーの γ, Euler's gamma)と呼ぶことがある. ($**$)の最初の式より,

$$
-\psi(1) = \lim_{N\to\infty}\left(\sum_{n=1}^{N+1}\frac{1}{n} - \log N\right) = \lim_{N\to\infty}\left(\sum_{n=1}^{N}\frac{1}{n} - \log N\right) = \gamma \approx 0.5772.
$$

これと(*)の最初の式から, 次が得られる:

$$\psi(n+1) = \frac{1}{1} + \frac{1}{2} + \cdots + \frac{1}{n} - \gamma \quad (n = 0, 1, 2, \ldots)$$

ガンマ函数は階乗 $\Gamma(n+1) = n!$ の一般化になっていたが, ディガンマ函数はEulerの γ の情報を含む調和級数の一般化になっていると考えられる.

Hurwitzのゼータ函数 (フルヴィッツのゼータ函数)が

$$\zeta(s, \alpha) = \sum_{k=0}^{\infty} \frac{1}{(\alpha + k)^s}$$

と定義される. トリガンマ函数以降のポリガンマ函数達は, 定数倍の違いを除いて, Hurwitzのゼータ函数で s を2以上の整数にした場合になっている:

$$\psi^{(k)}(\alpha) = (-1)^{k+1} k!\, \zeta(k+1, \alpha) \quad (k = 1, 2, 3, \ldots)$$

$\zeta(s) = \zeta(s, 1) = \sum_{n=1}^{\infty} n^{-s}$ は **Riemannのゼータ函数** (リーマンのゼータ函数)と呼ばれている. ゼータ函数達は数論の基本的研究対象である. ガンマ函数のよく使われる形から得られる

$$\frac{1}{(\alpha + k)^s} = \frac{1}{\Gamma(s)} \int_0^{\infty} e^{-(\alpha+k)t} t^{s-1}\, dt$$

という公式を使えば, Hurwitzのゼータ函数はガンマ函数のような積分表示を持つこともわかる:

$$\zeta(s, \alpha) = \frac{1}{\Gamma(s)} \int_0^{\infty} \sum_{k=0}^{\infty} e^{-(\alpha+k)t} t^{s-1}\, dt = \frac{1}{\Gamma(s)} \int_0^{\infty} \frac{e^{-\alpha t}}{1 - e^{-t}} t^{s-1}\, dt.$$

ガンマ函数の対数(対数ガンマ函数 (logarithmic Gamma function))は, 定数差を除いて, Hurwitzのゼータ函数の $s = 0$ での微係数に等しいことも知られている(**Lerchの定理**(レルヒの定理)):

$$\log \Gamma(\alpha) = \zeta_s(0, \alpha) + \log \sqrt{2\pi}.$$

この公式を使うと階乗のStirling近似の精密化(**Binetの公式**)も自然に得られる:

$$\varphi(x) = \int_0^{\infty} \left(\frac{1}{e^t - 1} - \frac{1}{t} + \frac{1}{2} \right) e^{-xt} t^{-1}\, dt.$$

とおくと,

$$\log \Gamma(x+1) = x \log x - x + \frac{1}{2} \log x + \log \sqrt{2\pi} + \varphi(x).$$

これはStirlingの公式 $\Gamma(n+1) = n! \approx n^n e^{-n} \sqrt{2\pi n}$ から得られる近似

$$\log \Gamma(n+1) \approx n \log n - n + \frac{1}{2} \log n + \log \sqrt{2\pi}$$

の精密化になっており, この近似の誤差が $\varphi(n)$ と書けることを意味している.

以上のようなことに興味がある人は次のリンク先を参照せよ:

- [10 Gauss積分, ガンマ函数, ベータ函数 (https://nbviewer.org/github/genkuroki/Calculus/blob/master/10%20Gauss%2C%20Gamma%2C%20Beta.ipynb#Lerch%E3%81%

In [66]:
```
plot(loggamma, eps(), 8; label="y = log Γ(x)", ylim=(-0.5, 10.2))
plot!(x -> x*log(x) - x - (1/2)*log(x) + log(√(2π));
    label="Stirling approx.", ls=:dash)
plot!(; legend=:top)
```

Out[66]:

In [67]:
```
plot(digamma, 0.1, 8; label="y = ψ(x) = digamma(x)")
plot!(x -> log(x) - 1/(2x); label="y = log x - 1/(2x)", ls=:dash)
plot!(; legend=:bottomright)
```

Out[67]:

In [68]:
```
plot(trigamma, 0.1, 8; label="y = ψ'(x) = trigamma(x)", ylim=(-0.2, 10.2))
plot!(x -> 1/x + 1/(2x^2); label="y = 1/x + 1/(2x²)", ls=:dash)
```

Out[68]:

5 Dirichlet積分=多変数ベータ函数

n 次元球体の体積を求める問題は, $a_i, p_i > 0$ に関する

$$X = \left\{ (x_1, \dots, x_n) \in \mathbb{R}^n_{>0} \;\middle|\; \left(\frac{x_1}{a_1}\right)^{p_1} + \cdots + \left(\frac{x_n}{a_n}\right)^{p_n} < 1 \right\}$$

の体積を求める問題に一般化される. ここで $\mathbb{R}_{>0}$ は正の実数全体の集合を表す.

さらにこの問題は $\alpha_i > 0$ も与えた場合の次の積分を計算する問題に一般化される:

$$V = \int \cdots \int_X x_1^{\alpha_1 - 1} \cdots x_n^{\alpha_n - 1} \, dx_1 \cdots dx_n$$

$\alpha_1 = \cdots = \alpha_n = 1$ のときこの積分は X の体積になる. この問題の解答は

$$V = \frac{a_1^{\alpha_1} \cdots a_n^{\alpha_n}}{p_1 \cdots p_n} \frac{\Gamma(\alpha_1/p_1) \cdots \Gamma(\alpha_n/p_n)}{\Gamma(\alpha_1/p_1 + \cdots + \alpha_n/p_n + 1)}.$$

例えば, $\alpha_1 = \cdots = \alpha_n = 1$, $a_1 = \cdots = a_n = 1$, $p_1 = \cdots = p_n = 2$ のとき, V の 2^n 倍の

$$2^n V = \frac{\Gamma(1/2)^n}{\Gamma(n/2+1)} = \frac{\pi^{n/2}}{\Gamma(n/2+1)}$$

は確かに n 次元単位球体の体積になっている.

このような一般化にはどのような価値があるのだろうか. 上の問題の解答はDirichlet(ディリクレ)の1839年の論文

- Lejeune-Dirichlet. Sur une nouvelle méthode pour la détermination des intégrales multiples. Journal de Mathématiques Pures et Appliquées (1839), 164-168. (link (https://eudml.org/doc/235403))

で与えられている. その論文でDirichletはベータ函数の多変数化であるDirichlet積分を発見した. Dirichlet積分はカテゴリカル分布と多項分布の共役事前分布であるDirichlet分布の「分母」となる統計学的にも重要な積分である. 統計学的にも重要なDirichlet積分は

- 高木貞治『解析概論』第8章第96節例2, pp.358-359
- 寺沢寛一『自然科学者のための 数学概論 [増訂版]』第5.21節[4], pp.216-217

でも紹介されている. これらはガンマ函数入門のための文献としても優れている.

5.1 Dirichlet積分=多変数ベータ函数の定義とそのガンマ函数表示

n 次元の領域 Δ_n を次のように定める:

$$\Delta_n = \{ (t_1, \dots, t_n) \in \mathbb{R}^n_{>0} \mid t_1 + \cdots + t_n < 1 \}.$$

ここで $\mathbb{R}_{>0}$ は正の実数全体の集合を表す.

例えば, $\Delta_1 = \{ t_1 \mid 0 < t_1 < 1 \}$ は開区間になり, Δ_2 は頂点 $(1,0)$, $(0,1)$, $(0,0)$ を持つ直角二等辺三角形の内側になり, Δ_3 は頂点 $(1,0,0)$, $(0,1,0)$, $(0,0,1)$, $(0,0,0)$ を持つ四面体の内側になる. (Δ_n を n **次元開単体** と呼ぶことがある.)

Dirichlet積分 とは次の積分のことである:

$$B(\alpha_1, \dots, \alpha_{n+1}) = \int \cdots \int_{\Delta_n} t_1^{\alpha_1 - 1} \cdots t_n^{\alpha_n - 1} (1 - (t_1 + \cdots + t_n))^{\alpha_{n+1} - 1} \, dt_1 \cdots dt_n \quad (\alpha_i > 0)$$

$B(\alpha_1, \dots, \alpha_{n+1})$ を **多変数ベータ函数** とも呼ぶことにする. 多変数ベータ函数は次のガンマ函数表示を持つ.

$$B(\alpha_1, \dots, \alpha_{n+1}) = \frac{\Gamma(\alpha_1) \cdots \Gamma(\alpha_{n+1})}{\Gamma(\alpha_1 + \cdots + \alpha_{n+1})}.$$

5.2 Dirichlet分布の定義

$\mathbb{R}_{>0}$ は正の実数全体の集合を表すとし, $\alpha_1, \dots, \alpha_r \in \mathbb{R}_{>0}$ と仮定する. このとき,

$$\Delta_{r-1} = \{ (t_1, \dots, t_{r-1}) \in \mathbb{R}^{r-1}_{>0} \mid t_1 + \cdots + t_{r-1} < 1 \}$$

の上で定義された確率密度函数を

$$p(t_1, \dots, t_{r-1} | \alpha_1, \dots, \alpha_r) = \frac{t_1^{\alpha_1 - 1} \cdots t_{r-1}^{\alpha_{r-1} - 1} (1 - (t_1 + \cdots + t_{r-1}))^{\alpha_r - 1}}{B(\alpha_1, \dots, \alpha_r)}$$

と定めることができる. この密度函数が定める多変量連続分布をDirichlet分布と呼び,

$$\mathrm{Dirichlet}(\alpha_1, \ldots, \alpha_r)$$

と表す. 対称性のためには

$$t_r = 1 - (t_1 + \cdots + t_{r-1})$$

とおいて, r 次元空間 $\mathbb{R}^r$ 内に浮かぶ $r-1$ 次元の開単体

$$\Delta^{r-1} = \{\, (t_1, \ldots, t_r) \in \mathbb{R}^r_{>0} \mid t_1 + \cdots + t_r = 1 \,\}$$

上の確率分布だとみなした方がよい. これによって, Dirichlet分布は, カテゴリカル分布

$$\mathrm{Categorical}(t_1, \ldots, t_r) \quad (t_i > 0,\ t_1 + \cdots + t_r = 1)$$

および多項分布

$$\mathrm{Multinomial}(n, t_1, \ldots, t_r) \quad (t_i > 0,\ t_1 + \cdots + t_r = 1)$$

のパラメータ $(t_1, \ldots, t_r)$ に関する確率分布だとみなされる. Dirichlet分布はカテゴリカル分布や多項分布の共役事前分布として重要である.

注意: $\Delta_{r-1} \subset \mathbb{R}^{r-1}$ と $\Delta^{r-1} \subset \mathbb{R}^r$ の微妙な違いに注意せよ. 一点, 線分, 三角形, 四面体, ... を単体と呼ぶ. 例えば, Δ^1 は両端が $(1,0)$ と $(0,1)$ の線分から両端の点を除いたものになり, Δ^2 は頂点 $(1,0,0), (0,1,0), (0,0,1)$ を持つ正三角形(から境界を除いたもの)になる.

In [69]:

```
function _pdf_dirichlet(d::Dirichlet, p, q; valmax=Inf)
    0 ≤ q ≤ min(2p, -2p+2) || return NaN
    x, y = p - 0.5q, q
    clamp(pdf(d, SVector(x, y, 1 - x - y)), 0, valmax)
end

function plot_dirichlet!(d::Dirichlet; valmax=Inf, kwargs...)
    @assert length(d) == 3
    f(p, q) = _pdf_dirichlet(d, p, q; valmax)
    p = range(0, 1, length=401)
    q = range(0, 1, length=401)
    annotate!((1.00, 0.10, text("(1,0,0)", 8, -58.0, :right)))
    annotate!((0.55, 1.00, text("(0,1,0)", 8, -58.0, :left)))
    annotate!((0.00, 0.08, text("(0,0,1)", 8,  58.0, :left)))
    heatmap!(p, q, f.(p',q); kwargs...)
end

function plot_dirichlet(d::Dirichlet ; valmax=Inf, kwargs...)
    plot(; colorbar=false, tick=false, axis=false)
    plot_dirichlet!(d; valmax, kwargs...)
    plot!(; size=(250, 200))
    a, b, c = round.(d.alpha; digits=3)
    title!("Dirichlet($a, $b, $c)")
    plot!(; b=:gray)
end

plot_dirichlet(α₁, α₂, α₃ ; valmax=Inf, kwargs...) =
    plot_dirichlet(Dirichlet([α₁, α₂, α₃]) ; valmax, kwargs...)
```

Out[69]:

```
plot_dirichlet (generic function with 2 methods)
```

以下はDirichlet分布のヒートマップである. 明るい部分ほど確率密度が大きい. 正三角形は $t_1, t_2, t_3 > 0$, $t_1 + t_2 + t_3 = 1$ を表し, 右下, 上, 左下の頂点はそれぞれ $(t_1, t_2, t_3) = (1,0,0), (0,1,0), (0,0,1)$ に対応しており, 正三角形の中心の座標は $(1/3, 1/3, 1/3)$ になる.

In [70]:
```
plot(plot_dirichlet(0.5, 0.5, 0.5; valmax=3),
    plot_dirichlet(0.5, 0.5, 1; valmax=3),
    plot_dirichlet(0.5, 1, 1; valmax=3); size=(800, 240), layout=(1,3))
```

Out[70]:

In [71]:
```
plot(plot_dirichlet(2, 3, 5), plot_dirichlet(4, 6, 10), plot_dirichlet(8, 12, 20);
    size=(800, 240), layout=(1,3))
```

Out[71]:

In [72]:
```
plot(plot_dirichlet(10, 15, 25), plot_dirichlet(20, 30, 60), plot_dirichlet(40, 60, 120);
    size=(800, 240), layout=(1,3))
```

Out[72]:

パラメータの大きなDirichlet分布は多変量正規分布で近似され, パラメータを大きくして行くと分布は一点に集中して行く.

5.3 多変数ベータ函数のガンマ函数表示の証明

$z = x_1 + \cdots + x_{n+1}, t_i = x_i/z$ とおく. 函数 $f(z, t_1, \ldots, t_n)$ について次の積分を考える:

$$J[f] = \int_0^\infty \cdots \int_0^\infty f(z, t_1, \ldots, t_n)\, e^{-(x_1+\cdots+x_{n+1})} x_1^{\alpha_1-1} \cdots x_{n+1}^{\alpha_{n+1}-1}\, dx_1 \cdots dx_{n+1}.$$

本当は z, t_i に $z = x_1 + \cdots + x_{n+1}$, $t_i = x_i/z$ を代入した式を書きたかったのだが, 式が煩雑に横に長くなり過ぎるのでそうしていない. この式の z, t にはそのように代入してから積分を行うという意味だと解釈する必要がある. そのとき

$$J[f] = \int_0^\infty \cdots \int_0^\infty \left(\int_0^\infty f(z, t_1, \ldots, t_n)\, e^{-(x_1+\cdots+x_{n+1})} x_1^{\alpha_1-1} \cdots x_{n+1}^{\alpha_{n+1}-1}\, dx_{n+1} \right) dx_1 \cdots dx_n$$

における内側の積分変数 x_{n+1} を $x_{n+1} = z - (x_1 + \cdots + x_n)$ を使って z に変換すると, z の動く範囲は $x_1 + \cdots + x_n < z < \infty$ になるので,

$$J[f] = \int_0^\infty \cdots \int_0^\infty \left(\int_{x_1+\cdots+x_n}^\infty f(z, t_1, \ldots, t_n)\, e^{-z} x_1^{\alpha_1-1} \cdots x_n^{\alpha_n-1} (z - (x_1 + \cdots + x_n))^{\alpha_{n+1}-1}\, dz \right) dx_1 \cdots dx_n.$$

この $n+1$ 重積分において $(x_1, \ldots, x_n, z)$ は $x_1, \ldots, x_n > 0$ かつ $x_1 + \cdots + x_n < z$ を満たす範囲を動く. ゆえに,

$$\Delta_n(z) = \{\, (x_1, \ldots, x_n) \in \mathbb{R}^n_{>0} \mid x_1 + \cdots + x_n < z \,\}$$

とおくと, 上の $n+1$ 重積分は次のように積分順序を交換できる:

$$J[f] = \int_0^\infty \left(\int \cdots \int_{\Delta_n(z)} f(z, t_1, \ldots, t_n)\, e^{-z} x_1^{\alpha_1-1} \cdots x_n^{\alpha_n-1} (z - (x_1 + \cdots + x_n))^{\alpha_{n+1}-1}\, dx_1 \cdots dx_n \right) dz.$$

積分変数 $x_1, \ldots, x_n$ を $x_i = zt_i$ によって $t_1, \ldots, t_n$ に変換すると, $(t_1, \ldots, t_n)$ は$t_1, \ldots, t_n > 0$ かつ $t_1 + \cdots + t_n < 1$ を満たす範囲を動くので,

$$\Delta_n = \Delta_n(1) = \{\, (t_1, \ldots, t_n) \in \mathbb{R}^n_{>0} \mid t_1 + \cdots + t_n < 1 \,\}$$

に注意すれば, 次が成立することがわかる:

$$J[f] = \int_0^\infty \left(\int \cdots \int_{\Delta_n} f(z, t_1, \ldots, t_n)\, e^{-z} (zt_1)^{\alpha_1-1} \cdots (zt_n)^{\alpha_n-1} (z(1 - (t_1 + \cdots + t_n)))^{\alpha_{n+1}-1}\, (z\, dt_1) \cdots (z\, dt_n) \right) dz.$$

これは次のように整理される:

$$J[f] = \int_0^\infty \left(\int \cdots \int_{\Delta_n} f(z, t_1, \ldots, t_n)\, e^{-z} z^{\alpha_1+\cdots+\alpha_{n+1}-1}\, t_1^{\alpha_1-1} \cdots t_n^{\alpha_n-1} (1 - (t_1 + \cdots + t_n))^{\alpha_{n+1}-1}\, dt_1 \cdots dt_n \right) dz.$$

これが $J[f]$ について示したかった公式である.

特に $f = 1$ のとき, $J[f]$ の定義より

$$J[1] = \Gamma(\alpha_1) \cdots \Gamma(\alpha_{n+1})$$

であることがわかり, 上で示した公式より,

$$J[1] = \Gamma(\alpha_1 + \cdots + \alpha_{n+1}) B(\alpha_1, \ldots, \alpha_{n+1})$$

となることがわかる. ゆえに

$$\Gamma(\alpha_1) \cdots \Gamma(\alpha_{n+1}) = \Gamma(\alpha_1 + \cdots + \alpha_{n+1}) B(\alpha_1, \ldots, \alpha_{n+1}).$$

すなわち

$$B(\alpha_1, \ldots, \alpha_{n+1}) = \frac{\Gamma(\alpha_1) \cdots \Gamma(\alpha_{n+1})}{\Gamma(\alpha_1 + \cdots + \alpha_{n+1})}.$$

5.4 ガンマ分布とDirichlet分布の関係

前節の結果を使えば, ガンマ分布とベータ分布の関係の証明と全く同様に以下の結果を示せる.

$X_i \sim \mathrm{Gamma}(\alpha_i, 1)$ $(i = 1, \ldots, n)$ でかつそれらは独立だと仮定する. それらが独立であるとは, $(X_1, \ldots, X_n)$ について

$$E[f(X_1, \ldots, X_n)] = \frac{1}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_n)} \int \cdots \int f(x_1, \ldots, x_n) e^{-(x_1+\cdots+x_n)} x_1^{\alpha_1-1} \cdots x_n^{\alpha_n-1}\, dx_1 \cdots dx_n$$

が成立することと同値である. このとき,

$$Z = X_1 + \cdots + X_n, \quad T_i = \frac{X_i}{Z}$$

とおくと

$$Z \sim \mathrm{Gamma}(\alpha_1 + \cdots + \alpha_n), \quad (T_1, \ldots, T_{n-1}) \sim \mathrm{Dirichlet}(\alpha_1, \ldots, \alpha_n).$$

Z と $(T_1, \ldots, T_{n-1})$ の独立性(同時確率密度函数がそれぞれの確率密度函数の積になること)も示せる.

5.5 問題: Dirichlet (1839)の公式の証明

$a_i, p_i, \alpha_i > 0$ と仮定する. 領域 X を

$$X = \left\{ (x_1, \ldots, x_n) \in \mathbb{R}^n_{>0} \;\middle|\; \left(\frac{x_1}{a_1}\right)^{p_1} + \cdots + \left(\frac{x_n}{a_n}\right)^{p_n} < 1 \right\}$$

と定める. ここで $\mathbb{R}_{>0}$ は正の実数全体の集合を表す. 次の積分を求めよ:

$$V = \int \cdots \int_X x_1^{\alpha_1 - 1} \cdots x_n^{\alpha_n - 1} \, dx_1 \cdots dx_n.$$

解答例: 変数 x_i 達と変数 t_i 達のあいだの

$$t_i = \left(\frac{x_i}{a_i}\right)^{p_i} \iff x_i = a_i t_i^{1/p_i}$$

という変換によって, 領域 X と領域

$$\Delta_n = \{ (t_1, \ldots, t_n) \in \mathbb{R}^n_{>0} \mid t_1 + \cdots + t_n < 1 \}$$

のあいだの一対一対応が得られる. ゆえに

$$\begin{aligned}
V &= \int \cdots \int_{\Delta_n} (a_i t_1^{1/p_1})^{\alpha_1 - 1} \cdots (a_n t_n^{1/p_n})^{\alpha_n - 1} \left(\frac{a_1}{p_1} t_1^{1/p_1 - 1} dt_1 \right) \cdots \left(\frac{a_n}{p_n} t_n^{1/p_n - 1} dt_n \right) \\
&= \frac{a_1^{\alpha_1} \cdots a_n^{\alpha_n}}{p_1 \cdots p_n} \int \cdots \int_{\Delta_n} t_1^{\alpha_1/p_1 - 1} \cdots t_n^{\alpha_n/p_n - 1} \, dt_1 \cdots dt_n \\
&= \frac{a_1^{\alpha_1} \cdots a_n^{\alpha_n}}{p_1 \cdots p_n} B(\alpha_1/p_1, \ldots, \alpha_n/p_n, 1) \\
&= \frac{a_1^{\alpha_1} \cdots a_n^{\alpha_n}}{p_1 \cdots p_n} \frac{\Gamma(\alpha_1/p_1) \cdots \Gamma(\alpha_n/p_n) \Gamma(1)}{\Gamma(\alpha_1/p_1 + \cdots + \alpha_n/p_n + 1)} \\
&= \frac{a_1^{\alpha_1} \cdots a_n^{\alpha_n}}{p_1 \cdots p_n} \frac{\Gamma(\alpha_1/p_1) \cdots \Gamma(\alpha_n/p_n)}{\Gamma(\alpha_1/p_1 + \cdots + \alpha_n/p_n + 1)}.
\end{aligned}$$

解答終

5.6 Dirichlet氏の1839年の論文との比較

以下は論文 Dirichlet (1839) (http://portail.mathdoc.fr/JMPA/afficher_notice.php?id=JMPA_1839_1_4_A11_0)の最後のページより.

Comme second exemple, j'indiquerai l'intégrale

$$V = \int x^{a-1} y^{b-1} z^{c-1} \ldots dx dy dz \ldots,$$

qui doit être étendue à toutes les valeurs positives de $x, y, z, \ldots$ telles qu'on ait

$$\left(\frac{x}{\alpha}\right)^p + \left(\frac{y}{\beta}\right)^q + \left(\frac{z}{\gamma}\right)^r + \ldots < 1,$$

les constantes $a, b, c, \ldots p, q, r, \ldots \alpha, \beta, \gamma, \ldots$ étant également positives. Par une analyse toute semblable, on parvient à cette expression très simple, qu'on peut aussi obtenir par d'autres moyens, et qui renferme un grand nombre de résultats relatifs aux volumes, aux centres de gravité, moments d'inertie, etc. :

$$V = \frac{\alpha^a \beta^b \gamma^c \ldots}{pqr} \frac{\Gamma\left(\frac{a}{p}\right) \Gamma\left(\frac{b}{q}\right) \Gamma\left(\frac{c}{r}\right) \ldots}{\Gamma\left(1 + \frac{a}{p} + \frac{b}{q} + \frac{c}{r} + \ldots\right)}.$$

これを見れば上の解答例によって, Dirichlet氏の1839年の論文の結果がぴったり再現されていることが分かる. ただし, 上で $x_1, x_2, \ldots, x_n$ と書いたところを, Dirichlet氏の時代には $x, y, z, \ldots$ のように書くことが普通であった. しかし, 以上を見れば, 19世紀のフランス語で書かれた数学の論文を現代の我々も結構読めそうなことがわかると思う.

Dirichlet氏が $a, b, c, \ldots$ と書いたところを上では $\alpha_1, \alpha_2, \ldots, \alpha_n$ と書き, Dirichlet氏が $\alpha, \beta, \gamma, \ldots$ と書いたところを上では $a_1, a_n, \ldots, a_n$ と書いた. それらの部分では, ギリシャ文字にする方が逆になっていることに注意せよ. この部分についてこのノートでDirichlet氏と逆のスタイルを採用せざるを得なかった理由は, 確率分布のパラメータをギリシャ文字で書く流儀を採用したからである.

Dirichlet氏が $p, q, r, \ldots$ と書いたところ上では $p_1, p_2, \ldots, p_n$ と書いた. 上に引用した論文の中にある2つめの $V =$ の式の右辺の分母の pqr は正しくは $pqr \cdots$ と $\cdots$ が必要である. このような細かい粗は昔の論文にも散見される. 数学の文献はそこに書いてあることを文字通りに正しいと思って読むのではなく, 内容を理解して適切に訂正しながら読む必要がある.

このように19世紀の数学者のDirichlet氏は, n 次元の球体の体積がガンマ函数で書けることの大幅な一般化を発見したのだが, その結果が, 後に統計学や機械学習での定番の道具の一つとして利用されることになることを予想できなかったに違いない.

In [ ]:

```
1
```

確率分布達の解釈

- 黒木玄

- 2022-04-11〜2022-06-08

このノートでは[Julia言語 (https://julialang.org/)](https://julialang.org/)を使用している:

- [Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)](https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

以下の図の内容を理解できればこのノートの内容をかなりよく理解できたことになる.

$\boxed{\text{ガンマ分布・正規分布ワールド}}$ $\boxed{\text{ベータ分布・t分布ワールド}}$ $\boxed{\text{ガンマ分布・正規分布ワールド}}$

$$\begin{tikzcd}
\mathrm{Gamma}(\alpha,\theta)\times\mathrm{Gamma}(\beta,\theta) \arrow[r, "(X,Y)\mapsto P=\frac{X}{X+Y}"] \arrow[d, equal] & \mathrm{Beta}(\alpha,\beta) \arrow[r, "\text{スケール極限 } \beta\to\infty"] \arrow[d, shift right, "U=\frac{P}{1-P}"'] & \mathrm{Gamma}(\alpha,\theta) \arrow[d, equal] \\
\mathrm{Gamma}(\alpha,\theta)\times\mathrm{Gamma}(\beta,\theta) \arrow[r, "(X,Y)\mapsto U=\frac{X}{Y}"] \arrow[d, shift right, "\text{特殊化}"'] & \mathrm{BetaPrime}(\alpha,\beta) \arrow[r, "\beta\to\infty"] \arrow[u, shift right, "P=\frac{U}{1+U}"'] \arrow[d, shift right, "\alpha=\frac{\nu_1}{2}\ \|\ \beta=\frac{\nu_2}{2}"'] & \mathrm{Gamma}(\alpha,\theta) \arrow[d, "\alpha=\frac{\nu_1}{2}\ \|"] \\
\mathrm{Gamma}(\frac{\nu_1}{2},2)\times\mathrm{Gamma}(\frac{\nu_2}{2},2) \arrow[r, "(X,Y)\mapsto U=\frac{X}{Y}"] \arrow[u, shift right, "\text{一般化}"'] \arrow[d, equal] & \mathrm{BetaPrime}(\frac{\nu_1}{2},\frac{\nu_2}{2}) \arrow[r, "\nu_2\to\infty"] \arrow[d, leftrightarrow, "\text{スケール変換}"] & \mathrm{Gamma}(\frac{\nu_1}{2},\theta) \arrow[d, leftrightarrow, "\text{スケール変換}"] \\
\mathrm{Chisq}(\nu_1)\times\mathrm{Chisq}(\nu_2) \arrow[r, "(X,Y)\mapsto F=\frac{X/\nu_1}{Y/\nu_2}"] \arrow[d, shift right, "\text{特殊化}"'] & \mathrm{FDist}(\nu_1,\nu_2) \arrow[r, "\nu_2\to\infty"] \arrow[d, shift right, "\text{特殊化}"'] & \mathrm{Chisq}(\nu_1) \arrow[d, shift right] \\
\mathrm{Chisq}(1)\times\mathrm{Chisq}(\nu) \arrow[r, "(Z^2,Y)\mapsto T^2=\frac{Z^2}{Y/\nu}"] \arrow[u, shift right, "\text{一般化}"'] & \mathrm{FDist}(1,\nu) \arrow[r, "\nu\to\infty"] \arrow[u, shift right, "\text{一般化}"'] & \mathrm{Chisq}(1) \arrow[u, shift right] \\
\mathrm{Normal}(0,1)\times\mathrm{Chisq}(\nu=n-1) \arrow[r, "(Z,Y)\mapsto T=\frac{Z}{\sqrt{Y/\nu}}"] \arrow[u, "Z^2"] \arrow[d, leftrightarrow, "\text{アフィン変換}"'] & \mathrm{TDist}(\nu) \arrow[r, "\nu\to\infty"] \arrow[u, "T^2"] & \mathrm{Normal}(0,1) \arrow[u, "Z^2"] \\
\mathrm{Normal}(\mu,\sigma/\sqrt{n})\times\mathrm{Gamma}(\frac{n-1}{2},\frac{2\sigma^2}{n-1}) & & \\
\bar{X}=\frac{1}{n}\sum_{i=1}^{n}X_i \quad S^2=\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X}) \arrow[u] & & \\
\mathrm{Normal}(\mu,\sigma)^n \quad \text{標本分布} & &
\end{tikzcd}$$

$$\begin{tikzcd}
\text{ベータ分布} \arrow[d, leftrightarrow] & \mathrm{Beta}(\alpha,\beta) \arrow[d, leftrightarrow] & \frac{1}{B(\alpha,\beta)}p^{\alpha-1}(1-p)^{\beta-1}dp \arrow[d, leftrightarrow] & p=\frac{u}{1+u} \arrow[d, leftrightarrow] \\
\text{ベータプライム分布} & \mathrm{BetaPrime}(\alpha,\beta) & \frac{1}{B(\alpha,\beta)}\frac{u^{\alpha-1}}{(1+u)^{\alpha+\beta}}du & u=\frac{p}{1-p}
\end{tikzcd}$$

これはほぼF分布

$$\begin{tikzcd}
\boxed{\text{Bernoulli試行}} \arrow[r, "t=n/L", "L\to\infty"'] & \boxed{\text{連続時間極限}} \\
n\text{回中平均 } np\text{回} \arrow[r, "n=L", "p=\frac{\lambda}{L}"'] & \text{単位時間に平均 } \lambda\text{回} \\
K\sim\mathrm{Binomial}(n,p) \arrow[d, "p\sim\mathrm{Beta}(\alpha,\beta)"'] & K\sim\mathrm{Poisson}(\lambda) \arrow[d, "\lambda\sim\mathrm{Gamma}(\alpha,\theta)"] \\
K\sim\mathrm{BetaBinomial}(n,\alpha,\beta) \arrow[r, "n=L", "\beta=\frac{L}{\theta}"'] & K\sim\mathrm{NegativeBinomial}(\alpha,\frac{1}{1+\theta})
\end{tikzcd}$$

Pólya's urn: $p=\alpha/(\alpha+\beta)$で
当たりが出ると $(\alpha,\beta)\to(\alpha+1,\beta)$
はずれが出ると $(\alpha,\beta)\to(\alpha,\beta+1)$

$$\begin{tikzcd}
N-k\sim\mathrm{BetaNegativeBinomial}(k,\alpha,\beta) \arrow[r, "\beta=\frac{L}{\theta}"] & T\sim\frac{1}{\theta}\mathrm{BetaPrime}(k,\alpha) \\
p\sim\mathrm{Beta}(\alpha,\beta) \arrow[u] & \lambda\sim\mathrm{Gamma}(\alpha,\theta) \arrow[u] \\
N-k\sim\mathrm{NegativeBinomial}(k,p) \arrow[r, "N=LT", "p=\frac{\lambda}{L}"'] & T\sim\mathrm{Gamma}(k,\frac{1}{\lambda}) \\
\text{ちょうど}k\text{回イベントが起こるまでN回} & \text{ちょうど}k\text{回イベントが起こるまでの時間がT} \\
\boxed{\text{Bernoulli試行}} \arrow[r, "L\to\infty", "\text{単位時間にL回試行}"'] & \boxed{\text{連続時間極限}}
\end{tikzcd}$$

目次

- 1 正規分布
  - 1.1 分布のアフィン変換の定義
  - 1.2 正規分布のアフィン変換も正規分布
  - 1.3 問題: 正規分布の平均と分散
  - 1.4 問題: 正規分布に従う独立な確率変数達の和も正規分布に従う
  - 1.5 問題: 標準正規分布に従う独立な確率変数の和
  - 1.6 必ず解いて欲しい問題: 正規分布における確率が95%または99%になる区間
  - 1.7 問題: 正規分布のモーメント母函数とキュムラント母函数
  - 1.8 問題: キュムラント母函数と期待値と分散
  - 1.9 問題: 対数正規分布の確率密度函数
  - 1.10 問題: 対数正規分布の期待値と分散
- 2 t分布の「分散が確率的に揺らいでいる正規分布」という解釈
  - 2.1 分散パラメータが確率分布に従う正規分布について
  - 2.2 問題: 分散がχ^2分布のv分の1に従う正規分布は自由度vのt分布になる
  - 2.3 必ず解いて欲しい問題: t分布における確率が95%または99%になる区間
- 3 Poisson分布の導入とその解釈
  - 3.1 Poisson分布の定義
  - 3.2 Poisson分布における確率の総和が1になることの確認
  - 3.3 問題: Poisson分布のキュムラント母函数と期待値と分散
  - 3.4 二項分布の連続時間極限
  - 3.5 Poisson分布は単位時間内に起こるイベントの回数の分布だとみなされる
  - 3.6 Poisson分布の中心極限定理と二項分布の中心極限定理の関係
  - 3.7 問題: Poisson分布の中心極限定理の直接証明
- 4 負の二項分布の「期待値＝分散が確率的に揺らいでいるPoisson分布」という解釈
  - 4.1 Poisson分布のパラメータがガンマ分布に従っていれば負の二項分布が得られる
  - 4.2 問題: Poisson分布とガンマ分布から負の二項分布が得られる
- 5 ガンマ分布の解釈
  - 5.1 問題: 標準正規分布に従う確率変数の2乗は自由度1のχ^2分布に従う
  - 5.2 問題: 標準正規分布に従うn個の独立な確率変数達の2乗は自由度nのχ^2分布に従う
  - 5.3 負の二項分布の連続時間極限
  - 5.4 負の二項分布の連続時間極限の様子をプロット
  - 5.5 ガンマ分布はイベントがα回起こるまでにかかる時間の分布とみなされる
  - 5.6 Poisson分布の累積分布函数とガンマ分布の累積分布函数の関係
  - 5.7 問題: Poisson分布の累積分布函数とガンマ分布の累積分布函数の関係の直接証明
  - 5.8 必ず解いて欲しい問題: χ^2分布における確率が95%または99%になる範囲
  - 5.9 問題: 自由度1のχ^2分布と標準正規分布の関係の数値例
- 6 ベータ分布の一様乱数生成の繰り返しによる解釈
  - 6.1 一様分布のサイズnの標本分布の順序統計量
  - 6.2 一様分布のサイズnの標本分布の順序統計量が従う分布
  - 6.3 一様乱数生成の繰り返しからベータ分布が出て来ることの数値的確認
  - 6.4 問題: 0からθのあいだの一様分布の上限θの不偏推定量
  - 6.5 二項分布の累積分布函数のベータ分布の累積分布函数表示
  - 6.6 二項分布とベータ分布の関係のClopper-PearsonのP値函数への応用
  - 6.7 おまけ: 二項分布とベータ分布の関係のBayes統計への応用
    - 6.7.1 平坦事前分布とimproper事前分布Beta(0,1)の事後分布の比較
  - 6.8 おまけ関連問題: ベータ分布は二項分布の共役事前分布である
    - 6.8.1 様々な事前分布の事後分布のプロット
  - 6.9 負の二項分布の累積分布函数のベータ分布の累積分布函数表示
  - 6.10 負の二項分布とベータ分布の関係の非整数パラメータケース
  - 6.11 おまけ: 二項分布と負の二項分布のP値のBayes統計的解釈への応用
  - 6.12 二項分布の累積分布函数と負の二項分布の累積分布函数の関係(まとめ)
  - 6.13 Poisson分布の累積分布函数とガンマの累積分布函数の関係(再掲, まとめ)
- 7 ベータ二項分布
  - 7.1 ベータ二項分布の定義
  - 7.2 ベータ二項分布の「成功確率が確率的に揺らいでいる二項分布」という解釈
  - 7.3 ベータ二項分布での確率の総和が 1 になることの確認
  - 7.4 問題: ベータ二項分布の期待値と分散
  - 7.5 問題: ベータ二項分布 = 負の超幾何分布
  - 7.6 超幾何分布, 二項分布, ベータ二項分布の統一的な理解
  - 7.7 問題: 超幾何分布, 二項分布, ベータ二項分布の期待値と分散の統一的な公式
  - 7.8 問題: ベータ二項分布の極限として負の二項分布が得られる
  - 7.9 問題: 二項分布とベータ二項分布のモーメント母函数
- 8 ベータ負の二項分布とベータプライム分布
  - 8.1 ベータ負の二項分布の定義
  - 8.2 問題: ベータ負の二項分布の期待値と分散
  - 8.3 負の二項分布とベータ負の二項分布の期待値と分散の比較

8.4 負の二項分布の連続時間極限の復習
8.5 ベータ分布の極限の復習
8.6 ベータ負の二項分布の連続極限
8.7 ベータ負の二項分布の解釈 (Pólyaの壺)
8.8 Pólyaの壺の確率分布の定義

In [1]:
```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:
```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:
```
mybinomial(a, k) = exp(loggamma(a+1) - loggamma(k+1) - loggamma(a-k+1))
```

Out[3]:
```
mybinomial (generic function with 1 method)
```

1 正規分布

$\mu, \sigma \in \mathbb{R}$, $\sigma > 0$ と仮定する. 確率密度函数

$$p(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

で定義される連続分布を平均 μ, 分散 σ^2 (標準偏差 σ)の **正規分布** (normal distribution)と呼び, $\mathrm{Normal}(\mu,\sigma)$ と書くのであった.

さらに, 平均 0, 分散 1^2 の正規分布を **標準正規分布** (standard normal distribution)と呼び, $\mathrm{Normal}()$ と書くのであった. 標準正規分布の密度函数は次の形になる:

$$p(z) = \frac{e^{-z^2/2}}{\sqrt{2\pi}}.$$

1.1 分布のアフィン変換の定義

確率変数 X が従う分布を D と書くとき, $a, b \in \mathbb{R}$, $a \neq 0$ に対する $aX + b$ が従う分布を $aD + b$ と書いて, 分布 D の **アフィン変換** (affine transformation)と呼ぶ.

1.2 正規分布のアフィン変換も正規分布

$a, b \in \mathbb{R}$, $a \neq 0$ と仮定する. 一般に確率変数 X について,

$$E[aX+b] = aE[X]+b, \quad \mathrm{var}(aX+b) = a^2\,\mathrm{var}(X)$$

が成立するのであった. X が正規分布に従う確率変数の場合には $aX+b$ も正規分布に従うことを示せる:

$$X \sim \mathrm{Normal}(\mu,\sigma) \implies aX+b \sim \mathrm{Normal}(a\mu+b, |a|\sigma).$$

実際, $X \sim \mathrm{Normal}(\mu,\sigma)$ のとき, $y = ax+b$ すなわち $x = (y-b)/a$ とおくと,

$$\begin{aligned}
E[f(aX+b)] &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} f(ax+b) \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} f(y) \exp\left(-\frac{((y-b)/a-\mu)^2}{2\sigma^2}\right) \frac{dy}{|a|} \\
&= \frac{1}{\sqrt{2\pi(a^2\sigma)^2}} \int_{-\infty}^{\infty} f(y) \exp\left(-\frac{(y-(a\mu+b))^2}{2a^2\sigma^2}\right) dy
\end{aligned}$$

なので, $aX+b$ は平均 $a\mu+b$, 分散 $a^2\sigma^2$ の正規分布に従う: $aX+b \sim \mathrm{Normal}(a\mu+b, |a|\sigma)$.

1.3 問題: 正規分布の平均と分散

分布 $\mathrm{Normal}(\mu,\sigma)$ の平均と分散がそれぞれ μ, σ^2 になることを示せ.

解答例1: すでに標準正規分布 $\mathrm{Normal}(0,1)$ の平均と分散がそれぞれ 0 と 1^2 になることは示してある. $X \sim \mathrm{Normal}(\mu,\sigma)$ のとき, $Z = (X-\mu)/\sigma$ とおくと $Z \sim \mathrm{Normal}((\mu-\mu)/\sigma, \sigma/\sigma) = \mathrm{Normal}(0,1)$ となるので, $X = \sigma Z + \mu$ の平均と分散はそれぞれ $\sigma\cdot 0+\mu = \mu$, $\sigma^2\cdot 1^2 = \sigma^2$ になる.

解答終

解答例2: 直接的に計算してみよう. $X \sim \mathrm{Normal}(\mu,\sigma)$ のとき, 積分変数を $x = \sigma z + \mu$ と変換すると,

$$E[X] = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} x e^{-(x-\mu)^2/(2\sigma^2)}\,dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (\sigma z+\mu) e^{-z^2/2}\,dz = \mu.$$

最後の等号で $\int_{-\infty}^{\infty} z e^{-z^2/2}\,dz = 0$ と $\int_{-\infty}^{\infty} e^{-z^2/2}\,dy = \sqrt{2\pi}$ を使った. この結果を使うと,

$$\mathrm{var}(X) = E[(X-\mu)^2] = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} (x-\mu)^2 e^{-(x-\mu)^2/(2\sigma^2)}\,dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \sigma^2 z^2 e^{-z^2/2}\,dx = \sigma^2.$$

最後の等号で $\int_{-\infty}^{\infty} z^2 e^{-z^2/2}\,dz = \sqrt{2\pi}$ を使った. この結果は

$$\int_{-\infty}^{\infty} e^{-\alpha z^2}\,dz = \sqrt{\pi}\,\alpha^{-1/2}$$

の両辺を α で微分して -1 倍して $\alpha = 1/2$ とおいても得られるし, ガンマ函数に帰着する方法でも得られる.

解答終

1.4 問題: 正規分布に従う独立な確率変数達の和も正規分布に従う

確率変数達 X, Y は独立で(同時確率密度函数がそれぞれの確率密度函数の積になっており),

$$X \sim \mathrm{Normal}(\mu_X, \sigma_X), \quad Y \sim \mathrm{Normal}(\mu_Y, \sigma_Y)$$

となっているならば,

$$X+Y = \mathrm{Normal}\Bigl(\mu_X+\mu_Y, \sqrt{\sigma_X^2+\sigma_Y^2}\Bigr)$$

となることを示せ.

解答例: $X \sim \mathrm{Normal}(\mu_X,\sigma_X)$, $Y \sim \mathrm{Normal}(\mu_Y,\sigma_Y)$ でかつ X, Y は独立であると仮定する.

X, Y は独立なので $X+Y$ の平均と分散はそれぞれ $\mu_X+\mu_Y$, $\sigma_X^2+\sigma_Y^2$ になる.

$X' = X-\mu_X$, $Y' = Y-\mu_Y$ とおくと, $X' \sim \mathrm{Normal}(0,\sigma_X)$, $Y \sim \mathrm{Normal}(0,\sigma_Y)$, $X+Y = (X'+Y')+(\mu_X+\mu_Y)$ となるので, $X'+Y'$ が正規分布に従うことを示せば十分である. ゆえに $\mu_X = 0$, $\mu_Y = 0$ と仮定してよいので, そのように仮定する.

$X+Y$ の確率密度函数を計算して, それが正規分布の密度函数になっていることを示せばよい.

$$E[f(X+Y)] = \frac{1}{\sqrt{(2\pi)^2\sigma_X^2\sigma_Y^2}} \iint_{\mathbb{R}^2} f(x+y)\exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_X^2}+\frac{y^2}{\sigma_Y^2}\right)\right) dx\,dy$$

であり, $x = x+y$ すなわち $y = z-x$ とおいて, x について平方完成すると,

$$\frac{x^2}{\sigma_X^2}+\frac{y^2}{\sigma_Y^2} = \frac{x^2}{\sigma_X^2}+\frac{(z-x)^2}{\sigma_Y^2} = \frac{\sigma_X^2+\sigma_Y^2}{\sigma_X^2\sigma_Y^2}\left(x-\frac{\sigma_X^2 z}{\sigma_X^2+\sigma_Y^2}\right)^2+\frac{z^2}{\sigma_X^2+\sigma_Y^2}.$$

となる(この計算を自分で実行してみること). ゆえに,

$$\begin{aligned}
&E[f(X+Y)] \\
&= \frac{1}{\sqrt{(2\pi)^2\sigma_X^2\sigma_Y^2}} \int_{\mathbb{R}} \left(\int_{\mathbb{R}} f(z)\exp\left(-\frac{\sigma_X^2+\sigma_Y^2}{2\sigma_X^2\sigma_Y^2}\left(x-\frac{\sigma_X^2 z}{\sigma_X^2+\sigma_Y^2}\right)^2-\frac{z^2}{2(\sigma_X^2+\sigma_Y^2)}\right) dx \right) dz \\
&= \frac{1}{\sqrt{(2\pi)^2\sigma_X^2\sigma_Y^2}} \sqrt{\frac{2\pi\sigma_X^2\sigma_Y^2}{\sigma_X^2+\sigma_Y^2}} \int_{\mathbb{R}} f(z)\exp\left(-\frac{z^2}{2(\sigma_X^2+\sigma_Y^2)}\right) dz \\
&= \frac{1}{\sqrt{2\pi(\sigma_X^2+\sigma_Y^2)}} \int_{\mathbb{R}} f(z)\exp\left(-\frac{z^2}{2(\sigma_X^2+\sigma_Y^2)}\right) dz
\end{aligned}$$

以上によって $X+Y \sim \mathrm{Normal}\left(\mu_X+\mu_Y, \sqrt{\sigma_X^2+\sigma_Y^2}\right)$ を示せた.

解答終

別解: $z = x+y$, $y = z-x$ とおくと密度函数中の指数函数の中身が z についでの二次式になることから, $Z = X+Y$ が正規分布に従うことは計算しなくても明らかだと考えることもできる. 上のように z 以外の変数は積分して消せる. そして, $Z = X+Y$ の期待値は X, Y の期待値の和になり, X, Y が独立という仮定からそれらは無相関になるので $Z = X+Y$ の分散は X, Y の分散の和になることもわかる. これだけで $X+Y \sim \mathrm{Normal}\left(\mu_X+\mu_Y, \sqrt{\sigma_X^2+\sigma_Y^2}\right)$ を証明できたともみなせる.

上の解答中の最も面倒な部分の計算が正しいことはコンピュータで以下のように確認できる.

$a = \sigma_X^2$, $b = \sigma_Y^2$ とおくと,

In [4]:
```
@vars a b x z
(a+b)/(a*b)*(x - a*z/(a+b))^2 + z^2/(a+b) - x^2/a - (z-x)^2/b ▷ simplify
```

Out[4]: 0

以下のように素朴に計算することもできる.

In [5]:
```
@vars a b positive=true
@vars x y z t
expr = x^2/a + y^2/b
```

Out[5]: $\frac{y^2}{b}+\frac{x^2}{a}$

In [6]:
```
expr = expr(y⇒z-x).expand()
```

Out[6]: $\frac{x^2}{b}-\frac{2xz}{b}+\frac{z^2}{b}+\frac{x^2}{a}$

In [7]:
```
A, B = 1/a+1/b, z/b
expr = sympy.poly(expr(x ⇒ t + B/A), t)
```

Out[7]: $\mathrm{Poly}\left(\frac{a+b}{ab}t^2+\frac{z^2}{a+b}, t, domain = \mathbb{Z}(z,a,b)\right)$

In [8]:

```
# 以下のセルでの説明で使う図の準備

μ, σ, c = 2, 3, 1.5
normal = Normal(μ, σ)

P1 = plot(normal, μ - 4σ, μ + 4σ; label="Normal(μ,σ)", xlabel="x = σz + μ")
vline!([μ]; label="x=μ", xtick = ([μ-c*σ, μ, μ+c*σ], ["μ-cσ", "μ", "μ+cσ"]), ytick=false)
plot!(normal, μ - 4σ, μ - c*σ; label="", c=1, frange=0, fc=:red, fa=0.5)
plot!(normal, μ + c*σ, μ + 4σ; label="", c=1, frange=0, fc=:red, fa=0.5)
annotate!(μ - 1.3c*σ, 0.8pdf(normal, μ - c*σ), ("P(X≤μ-cσ)", 10, :red, :right))
annotate!(μ - 1.5c*σ, 0.5pdf(normal, μ - c*σ), ("= α/2", 10, :red, :right))
annotate!(μ + 1.5c*σ, 0.5pdf(normal, μ + c*σ), ("= α/2", 10, :red, :left))
annotate!(μ + 1.3c*σ, 0.8pdf(normal, μ + c*σ), ("1-P(X≤μ+cσ)", 10, :red, :left))

P2 = plot(Normal(), -4, 4; label="Normal(0,1)", xlabel="z = (x - μ)/σ")
vline!([0]; label="z=0", xtick = ([-c, 0, c], ["-c", "0", "c"]), ytick=false)
plot!(Normal(), -4, -c; label="", c=1, frange=0, fc=:red, fa=0.5)
plot!(Normal(),  c,  4; label="", c=1, frange=0, fc=:red, fa=0.5)
annotate!(-1.3c, 0.8pdf(Normal(), -c), ("P(Z≤-c)", 10, :red, :right))
annotate!(-1.5c, 0.5pdf(Normal(), -c), ("= α/2", 10, :red, :right))
annotate!( 1.5c, 0.5pdf(Normal(),  c), ("= α/2", 10, :red, :left))
annotate!( 1.3c, 0.8pdf(Normal(),  c), ("1-P(Z≤c)", 10, :red, :left))

var"P(μ-cσ ≤ X ≤ μ+cσ) = 1-α ⇔ 1 - P(Z ≤ c) = α/2" =
    plot(P1, P2; size=(800, 250), bottommargin=4Plots.mm);
```

1.5 問題: 標準正規分布に従う独立な確率変数の和

確率変数 $X_1, \ldots, X_n$ の各々は標準正規分布に従い, それらは独立な確率変数達である(すなわちそれらの同時確率密度函数が標準正規分布の確率密度函数の積になっている)と仮定する. このとき

$$Z_n = \frac{X_1 + \cdots + X_n}{\sqrt{n}}$$

も標準正規分布に従うことを示せ.

解答例: $X_i/\sqrt{n}$ は平均 0, 分散 $1/n$ の正規分布に従し, $X_1/\sqrt{n}, \ldots, X_n/\sqrt{n}$ は独立になるので, 上の問題の結果より, それらの和は平均 $0 + \cdots + 0 = 0$, 分散 $1/n + \cdots + 1/n = 1$ の正規分布つまり標準正規分布に従う.

解答終

注意: この問題の結果より, 各々が標準正規分布に従う n 個の独立な確率変数達の和の $\sqrt{n}$ 分の 1 が従う分布も標準正規分布になる.

分布 D は平均 0 と分散 1 を持つ分布であるとき, 各々が分布 D に従う n 個の独立な確率変数達の和の $\sqrt{n}$ 分の 1 (以下 Z_n と書く)が従う分布はどうなるだろうか?

その場合にも Z_n の平均と分散はそれぞれ 0 と 1 になるが, Z_n が従う分布は一般に D とは異なる分布になる.

しかし, 応用上無理のないゆるい仮定のもとで, Z_n が従う分布は n を大きくすると標準正規分布に近付くことを示せる.

これが一般的な **中心極限定理** の主張である. もとの分布が標準正規分布なら Z_n が従う分布はぴったり標準正規分布になるが, それ以外の一般の場合にも n を大きくすれば Z_n が従う分布が近似的に標準正規分布になることはいえるのである.

1.6 必ず解いて欲しい問題: 正規分布における確率が95%または99%になる区間

$X \sim \mathrm{Normal}(\mu, \sigma)$ であると仮定する. このとき, X が区間 $[\mu - c\sigma, \mu + c\sigma]$ に含まれる確率

$$P(\mu - c\sigma \leq X \leq \mu + c\sigma)$$

が $1 - \alpha$ に等しくなるような c を誤差函数

$$y = \mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-u^2)\, du$$

の逆函数 $x = \mathrm{erfinv}(y)$ を使って表せ. (正規分布の定義に戻って地道に計算せよ.)

さらに, 上の確率が 95% になる c と 99% になる c を小数点以下第2桁目まで求めよ.

解答例: 標準正規分布の場合の計算に帰着することを考えよう.

$Z = (X-\mu)/\sigma$ とおくと Z は標準正規分布に従うのであった: $Z \sim \mathrm{Normal}(0,1)$. このとき,

$$\mu - c\sigma \le X \le \mu + c\sigma \iff -c \le Z = \frac{X-\mu}{\sigma} \le c$$

なので, 標準正規分布に従う確率変数 Z について $P(-c \le Z \le c)$ が 95% になる c と 99% になる c を求めればよい. 正規分布は左右対称なので, $P(-c \le Z \le c) = 1-\alpha$ となることと, $1 - P(Z \le c) = \alpha/2$ すなわち

$$P(Z \le c) = 1 - \alpha/2$$

となることは同値である. 下の図を見よ.

In [9]:
```
var"P(μ-cσ ≤ X ≤ μ+cσ) = 1-α ⇔ 1 - P(Z ≤ c) = α/2"
```

Out[9]:

標準正規分布の累積分布函数

$$F(z) = P(Z \le z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{t^2/2}\, dt$$

はコンピュータでの基本特殊函数ライブラリに含まれている誤差函数

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-u^2)\, du$$

を使えば

$$F(z) = P(Z \le z) = \frac{1 + \mathrm{erf}(z/\sqrt{2})}{2}.$$

と書けるのであった. この公式を証明するためには,

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{0} e^{t^2/2}\, dt = \frac{1}{2}$$

を使ってから, $t = \sqrt{2}\, u$ とおけばよい. ゆえに誤差函数の逆函数 $\mathrm{erfinv}(y)$ (これもコンピュータでの基本特殊函数ライブラリに含まれている)を使えば, 標準正規分布の累積分布函数 $p = F(z)$ の逆函数(分位点函数)は

$$z = Q(p) = F^{-1}(p) = \sqrt{2}\ \mathrm{erfinv}(2p-1)$$

と書ける. これを使えば $F(c) = P(Z \le c) = 1 - \alpha/2$ となる c を

$$c = Q(1-\alpha/2) = \sqrt{2}\ \mathrm{erfinv}(1-\alpha)$$

と求めることができる.

- $1-\alpha = 95\%$ のとき, $c = \sqrt{2}\ \mathrm{erfinv}(0.95) \approx 1.96$
- $1-\alpha = 99\%$ のとき, $c = \sqrt{2}\ \mathrm{erfinv}(0.99) \approx 2.58$

解答終

[Julia言語 (https://julialang.org/)](https://julialang.org/)では以下のように計算できる. (ただし, `using SpecialFunctions, Distributions` が必要.)

In [10]:
```
@show √2 * erfinv(0.95)
@show √2 * erfinv(0.99);
```

```
√2 * erfinv(0.95) = 1.9599639845400576
√2 * erfinv(0.99) = 2.5758293035489053
```

In [11]:
```
@show quantile(Normal(), 0.975)
@show quantile(Normal(), 0.995);
```

```
quantile(Normal(), 0.975) = 1.9599639845400576
quantile(Normal(), 0.995) = 2.5758293035489053
```

WolframAlpha (https://www.wolframalpha.com/)では以下のように計算できる:

- √2 erfinv(0.95) (https://www.wolframalpha.com/input?i=%E2%88%9A2+erfinv%280.95%29)
- √2 erfinv(0.99) (https://www.wolframalpha.com/input?i=%E2%88%9A2+erfinv%280.99%29)
- quantile(NormalDistribution(0,1), 0.975) (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.975%29)
- quantile(NormalDistribution(0,1), 0.995) (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.995%29)

1.7 問題: 正規分布のモーメント母函数とキュムラント母函数

$X \sim \mathrm{Normal}(\mu, \sigma)$ のとき次が成立することを示せ:

$$E[e^{tX}] = e^{\mu t + \sigma^2 t^2/2}, \quad \log E[e^{tX}] = \mu t + \sigma^2 \frac{t^2}{2}.$$

注意: 一般に確率変数 X に対して, $E[e^{tX}]$ を X の **モーメント母函数** (moment generating function, mgf)と呼び, $\log E[e^{tX}]$ を X の **キュムラント母函数** (cumulant generating function, cgf)と呼ぶ.

注意: 正規分布のキュムラント母函数は上のように非常に単純な形になる. キュムラント母函数が t について2次式になることと分布が正規分布であることは同値であり, キュムラント母函数中の t について3次以上の項は分布が正規分布とどのように違うかを表している. 特にそのうちの最初の2つである $t^3/3!$ と $t^4/4!$ の係数はそれぞれ **歪度** (わいど, skewness) と **尖度** (せんど, kurtosis) と呼ばれている.

注意: モーメント母函数とキュムラント母函数は物理での統計力学での分配函数と自由エネルギーの統計学での類似物になっており, 極めて便利な母函数になっている. 上の t は物理的には逆温度 β の -1 倍の $-\beta$ に対応している. (t が小さい場合の物理的類似は温度が低い場合になる. $t = -\beta$ が小さくなると温度の逆数 β は大きくなる. 温度の逆数 β が大きくなると温度は小さくなる. 注意: 絶対値が大きな負の温度は絶対値が小さな負の逆温度に対応し, 絶対温度無限大は $\beta = 0$ の場合に対応しているので, 絶対値が大きな負の温度は絶対温度無限大よりも少しだけ高温の場合に対応している. 絶対温度ではなく, その逆数の逆温度 $\beta = -t$ で考えた方が混乱が少なくて済む.)

解答例:

$$\begin{aligned}
tx - \frac{(x-\mu)^2}{2\sigma^2} &= -\frac{(x-\mu)^2 - 2\sigma^2 tx}{2\sigma^2} = -\frac{x^2 - 2(\mu + \sigma^2 t)x + \mu^2}{2\sigma^2} \\
&= -\frac{(x - (\mu + \sigma^2 t))^2 + \mu^2 - (\mu + \sigma^2 t)^2}{2\sigma^2} \\
&= -\frac{(x - (\mu + \sigma^2 t))^2 - 2\sigma^2 \mu t + \sigma^4 t^2}{2\sigma^2} \\
&= -\frac{(x - (\mu + \sigma^2 t))^2}{2\sigma^2} + \mu t + \sigma^2 \frac{t^2}{2}
\end{aligned}$$

より

$$\begin{aligned}
E[e^{tX}] &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{tx} e^{-(x-\mu)^2)/(2\sigma^2)}\, dx \\
&= \frac{e^{\mu t + \sigma^2 t^2/2}}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-(x-(\mu+\sigma^2 t))^2)/(2\sigma^2)}\, dx = e^{\mu t + \sigma^2 t^2/2}.
\end{aligned}$$

こらから $\log E[e^{tX}] = \mu t + \sigma^2 t^2/2$ はただちに得られる.

解答終

1.8 問題: キュムラント母函数と期待値と分散

期待値 μ と分散 σ を持つ確率変数 X について次が成立することを示せ:

$$\log E[e^{tX}] = \mu t + \sigma^2 \frac{t^2}{2} + O(t^3)$$

この結果は期待値と分散の計算に有用な場合がある. この結果は今後空気のごとく使われる.

解答例: $e^{tX} = 1 + Xt + X^2t^2/2 + O(t^3)$ より,

$$E[e^{tX}] = 1 + E[X]t + E[X^2]\frac{t^2}{2} + O(t^3).$$

$\log(1+a) = a - a^2/2 + O(a^3)$ を使うと,

$$\begin{aligned}\log E[e^{tX}] &= E[X]t + E[X^2]\frac{t^2}{2} - \frac{E[X]t)^2}{2} + O(t^3) \\ &= E[X]t + (E[X^2] - E[X]^2)\frac{t^2}{2} + O(t^3) = \mu t + \sigma^2\frac{t^2}{2} + O(t^3).\end{aligned}$$

解答終

1.9 問題: 対数正規分布の確率密度函数

$X \sim \mathrm{Normal}(\mu, \sigma)$ のときの $Y = e^X$ が従う分布を対数正規分布と呼び,

$$\mathrm{LogNormal}(\mu, \sigma)$$

と表す. 対数正規分布の確率密度函数を求めよ.

解答例: $x = \log y$ と積分変数を変換すると, $dx = dy/y$ なので,

$$E[f(Y)] = \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} f(e^x)\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)dx = \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} f(y)\exp\left(-\frac{(\log y-\mu)^2}{2\sigma^2}\right)\frac{dy}{y}.$$

ゆえに対数正規分布の確率密度函数は次の形になる:

$$p(y) = \frac{1}{\sqrt{2\pi\sigma^2}\,y}\exp\left(-\frac{(\log y-\mu)^2}{2\sigma^2}\right).$$

解答終

In [12]:
```
plot(LogNormal(0, 1), 0, 8; label="", title="LogNormal(0, 1)")
```

Out[12]:

1.10 問題: 対数正規分布の期待値と分散

$Y \sim \mathrm{LogNormal}(\mu, \sigma)$ のとき次が成立することを示せ:

$$E[Y^m] = e^{m\mu + m^2\sigma^2/2}, \quad E[Y] = e^{\mu+\sigma^2/2}, \quad \mathrm{var}(Y) = e^{2\mu+\sigma^2}(e^{\sigma^2} - 1).$$

解答例: $X = \log Y \sim \mathrm{Normal}(\mu, \sigma)$ となるので, $E[Y^m]$ の計算で正規分布のモーメント母函数に関する結果を使え, 次が得られる:

$$E[Y^m] = E[e^{mX}] = e^{m\mu + m^2\sigma^2/2}.$$

ゆえに

$$
\begin{aligned}
&E[Y] = e^{\mu+\sigma^2/2}, \\
&\mathrm{var}(Y) = E[Y^2] - E[Y]^2 = e^{2\mu+2\sigma^2} - e^{2\mu+\sigma^2} = e^{2\mu+\sigma^2}(e^{\sigma^2}-1).
\end{aligned}
$$

解答終

2 t分布の「分散が確率的に揺らいでいる正規分布」という解釈

自由度 $\nu > 0$ のt 分布 $\mathrm{TDist}(\nu)$ の確率密度函数は

$$
p(t|\nu) = \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)}\left(1+\frac{t^2}{\nu}\right)^{(\nu+1)/2}
$$

であり, これは $\nu \to \infty$ で標準正規分布の確率密度函数 $e^{-t^2/2}/\sqrt{2\pi}$ に収束するのであった.

2.1 分散パラメータが確率分布に従う正規分布について

Z は標準正規分布に従う確率変数であるとする.

$\sigma > 0$ について, $Z/\sqrt{\sigma^{-2}} = \sigma Z \sim \mathrm{Normal}(0,\sigma)$ となる.

もしも, 値が正の実数になる確率変数 Y の期待値が ν で標準偏差が ν との比較で小さいならば Y/ν の分布はその期待値 1 の近くに集中する. (以下, Z, Y は独立であると仮定する.) そのとき, $Z/\sqrt{Y/\nu}$ が従う分布は分散が 1 の周囲で確率的に揺らぐ「正規分布」に従うことになる.

Y が確率密度函数 $p(y|\nu)$ を持つとき, $z = \sqrt{y/\nu}\,t$, $dz = \sqrt{y/\nu}\,dt$ とおくと,

$$
\begin{aligned}
E[f(Z/\sqrt{Y/\nu})] &= \int_0^\infty \left(\int_{-\infty}^\infty f\left(z/\sqrt{y/\nu}\right)\frac{e^{-z^2/2}}{\sqrt{2\pi}}p(y|\nu)\,dz\right)dy \\
&= \int_0^\infty \left(\int_{-\infty}^\infty f(t)\frac{e^{-yt^2/(2\nu)}}{\sqrt{2\pi}}p(y|\nu)\sqrt{\frac{y}{\nu}}\,dt\right)dy \\
&= \int_{-\infty}^\infty f(t)\left(\frac{1}{\sqrt{2\pi\nu}}\int_0^\infty e^{-t^2y/(2\nu)}y^{1/2}p(y|\nu)\,dy\right)dt
\end{aligned}
$$

なので, 確率変数 $T = Z/\sqrt{Y/\nu}$ は確率密度函数

$$
p(t) = \frac{1}{\sqrt{2\pi\nu}}\int_0^\infty e^{-t^2y/(2\nu)}y^{1/2}p(y|\nu)\,dy
$$

を持つことになる.

以上のように複数の確率変数を組み合わせて統計モデルを構築することが多い. 実は t 分布もそのような統計モデルの一種とみなせることを次の問題で確認してもらう. 以上を踏まえて次の問題を解け.

2.2 問題: 分散がχ²分布のv分の1に従う正規分布は自由度vのt分布になる

$\nu > 0$ だと仮定する. 確率変数達 Y, Z は独立であるとし(同時確率密度函数がそれぞれの密度函数の積になる),

$$
Z \sim \mathrm{Normal}(0,1), \quad Y \sim \mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)
$$

と仮定する. $\mathrm{Chisq}(\nu)$ の期待値は $(\nu/2)2 = \nu$ になり, 分散は $(\nu/2)2^2 = 2\nu$ になり, 標準偏差は $\sqrt{2\nu}$ になるので, ν が 2 より大きければ, 標準偏差は期待値よりも小さくなる. このとき,

$$
T = \frac{Z}{\sqrt{Y/\nu}} \sim \mathrm{TDist}(\nu)
$$

となることを示せ.

解答例: $Y \sim \mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$ の確率密度函数は

$$
p(y|\nu) = \frac{1}{2^{\nu/2}\,\Gamma(\nu/2)}e^{-y/2}y^{\nu/2-1} \quad (y > 0)
$$

になるので, 前節の結果より, T の確率密度函数が以下のように計算される:

$$
\begin{aligned}
p(t) &= \frac{1}{\sqrt{2\pi\nu}\,2^{\nu/2}\Gamma(\nu/2)}\int_0^\infty e^{-t^2y/(2\nu)}y^{1/2}e^{-y/2}y^{\nu/2-1}\,dy \\
&= \frac{1}{\sqrt{\nu}\,2^{(\nu+1)/2}\sqrt{\pi}\,\Gamma(\nu/2)}\int_0^\infty \exp\left(-\frac{1+t^2/\nu}{2}y\right)y^{(\nu+1)/2-1}\,dy \\
&= \frac{1}{\sqrt{\nu}\,2^{(\nu+1)/2}\Gamma(1/2)\,\Gamma(\nu/2)}\left(\frac{1+t^2/\nu}{2}\right)^{-(\nu+1)/2}\Gamma((\nu+1)/2) \\
&= \frac{\Gamma((\nu+1)/2)}{\sqrt{\nu}\,\Gamma(1/2)\Gamma(\nu/2)}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2} \\
&= \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)}\left(1+\frac{t^2}{\nu}\right)^{-(\nu+1)/2}
\end{aligned}
$$

3つめの等号で $\Gamma(1/2)=\sqrt{\pi}$ とガンマ函数のよくある使用法を使い, 最後の等号で $\Gamma(\alpha)\Gamma(\beta)=\Gamma(\alpha+\beta)B(\alpha,\beta)$ を使った. 最後の式は t 分布の確率密度函数である. ゆえに $T\sim\mathrm{TDist}(\nu)$.

解答終

以上のような積分の計算をすることがどうしても嫌な人は, 以下のようにコンピュータで乱数を発生させてその分布を比較して納得するとよい. 上のような数学の計算が得意な人であっても, 具体的な計算をコンピュータに大量にさせて, その結果を視覚化して確認した方がよい. 「百聞は一見に如かず」は確率分布の世界でも正しい.

In [13]:

```
function plot_nct(ν; L = 10^6)
    Z = rand(Normal(), L) # 標準正規分布の乱数を大量に生成
    Y = rand(Chisq(ν), L) # χ² 分布で乱数達を大量生成
    T_nc = @. Z/√(Y/ν)

    # 比較のための同時プロット
    binmin, binmax = round.(quantile.(TDist(ν), (0.001, 0.999)))
    bin = binmin:0.1:binmax
    histogram(T_nc; norm=true, alpha=0.3, bin, xlim=(-6, 6), label="Z/√(Y/ν)")
    plot!(TDist(ν), binmin, binmax; label="TDist(ν)", lw=1.5)
    plot!(Normal(), binmin, binmax; label="Normal(0,1)", lw=1.5, ls=:dash)
    title!("ν = $ν")
end
```

Out[13]:

```
plot_nct (generic function with 1 method)
```

In [14]:

```
plot(plot_nct(1), plot_nct(2); size=(800, 250))
```

Out[14]:

In [15]:
```
plot(plot_nct(3), plot_nct(4); size=(800, 250))
```

Out[15]:

In [16]:
```
plot(plot_nct(10), plot_nct(20); size=(800, 250))
```

Out[16]:

注意: 一般に $Y \sim \mathrm{Gamma}(\alpha, \theta)$ のとき Y の平均と標準偏差はそれぞれ $\alpha\theta$ と $\sqrt{\alpha}\,\theta$ になるので, で $Y/(\alpha\theta)$ の平均と標準偏差はそれぞれ 1 と $1/\sqrt{\alpha}$ になる. これは α が大きなとき, $Y/(\alpha\theta)$ の分布は 1 の近くに集中することを意味している.

In [17]:
```
function plot_gamoαθ(α, θ = 1; L = 10^6) # Gamma over αθ
    Y = rand(Gamma(α, θ), L) # χ² 分布で乱数達を大量生成
    histogram(Y/(α*θ); norm=true, alpha=0.3, bin=0:0.025:2, label="Y/(αθ)")
    title!("α = $α")
end
```

Out[17]:
```
plot_gamoαθ (generic function with 2 methods)
```

In [18]:
```
plot(plot_gamoαθ(10), plot_gamoαθ(30), plot_gamoαθ(100); size=(800, 200), layout=(1, 3))
```

Out[18]:

2.3 必ず解いて欲しい問題: t分布における確率が95%または99%になる区間

$0 < \alpha < 1, \nu > 0$ であるとし, $T \sim \mathrm{TDist}(\nu)$ と仮定する. このとき, T が区間 $[-c, c]$ に含まれる確率

$$P(-c \le T \le c)$$

が $1-\alpha$ になるような c を **正則化された不完全ベータ函数**

$$y = I(x|\alpha, \beta) = I_x(\alpha, \beta) = \frac{\int_0^x v^{\alpha-1}(1-v)^{\beta-1}\,dv}{B(\alpha, \beta)}$$

の逆函数 $x = I^{-1}(y|\alpha,\beta)$ を使って表せ. (注意: $I_x(\alpha,\beta)$ は広く使われている標準的な記号法だが, $I(x|\alpha,\beta)$ と $I^{-1}(y|\alpha,\beta)$ はここだけの記号法である.)

さらに, $\nu = 9$ と $\nu = 19$ と $\nu = 29$ の場合に $P(-c \leq T \leq c)$ が 95% になる c と 99% になる c を小数点以下第2桁目まで求めよ.

お願い: この問題の前半部分は少し難しいかもしれないので, どうしても無理ならば以下の解答例をざっと見るだけでもよい. しかし, 後半の「さらに」以降の具体的な数値を求める部分は必ず解けるようになっておいて欲しい. コンピュータを使ってよい. (より正確に言えばコンピュータを使った計算の仕方を1つ以上マスターしておくこと!)

注意: 正規分布モデルの統計学で t 分布に関する計算が不完全ベータ函数に帰着できるという事実から, 正規分布モデルの統計学においてベータ分布が必須であることがわかる.

解答例: 以下 $t \geq 0$ と仮定する. t 分布の累積分布函数を $F(t) = P(T \leq t)$ と書くと,

$$F(t) = \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{-\infty}^{t} \left(1 + \frac{s^2}{\nu}\right)^{-(\nu+2)/2} ds = \frac{1}{2} + \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{0}^{t} \left(1 + \frac{s^2}{\nu}\right)^{-(\nu+2)/2} ds.$$

ベータ函数の様々な表示の計算で学んだように, この積分変数を $s = \sqrt{\nu u}$ ($\iff$ $u = s^2/\nu$)とおいて, さらに $u = v/(1-v) = 1/(1-v) - 1$ ($\iff$ $v = u/(1+u)$)とおくとベータ分布の累積分布函数すなわち正則化された不完全ベータ函数に帰着できることがわかる:

$$\begin{aligned}
F(t) &= \frac{1}{2} + \frac{1}{\sqrt{\nu}\,B(1/2,\nu/2)} \int_{0}^{t^2/\nu} (1+u)^{-(\nu+2)/2} \frac{\sqrt{\nu}}{2} u^{-1/2}\,du \\
&= \frac{1}{2} + \frac{1}{2B(1/2,\nu/2)} \int_{0}^{t^2/\nu} \frac{u^{1/2-1}\,du}{(1+u)^{1/2+\nu/2}} \\
&= \frac{1}{2} + \frac{1}{2B(1/2,\nu/2)} \int_{0}^{(t^2/\nu)/(1+t^2/\nu)} \left(\frac{v}{1-v}\right)^{1/2-1} (1-v)^{1/2+\nu/2} \frac{dv}{(1-v)^2} \\
&= \frac{1}{2} + \frac{1}{2} \frac{\int_0^{(t^2/\nu)/(1+t^2/\nu)} v^{1/2-1}(1-v)^{\nu/2-1}\,dv}{B(1/2,\nu/2)} \\
&= \frac{1}{2}\left(1 + I\left(\frac{t^2/\nu}{1+t^2/\nu}\middle|\alpha,\beta\right)\right).
\end{aligned}$$

ゆえに, $p = u/(1+u)$ と $u = p/(1-p) = 1/(1-p) - 1$ が同値であることを使うと, $t \geq 0$ のとき,

$$\begin{aligned}
p = F(t) &\iff I^{-1}(2p-1|\alpha,\beta) = \frac{t^2/\nu}{1+t^2/\nu} \\
&\iff \frac{t^2}{\nu} = \frac{I^{-1}(2p-1|\alpha,\beta)}{1-I^{-1}(2p-1|\alpha,\beta)} = \frac{1}{1-I^{-1}(2p-1|\alpha,\beta)} - 1 \\
&\iff t = \sqrt{\nu\left(\frac{1}{1-I^{-1}(2p-1|\alpha,\beta)} - 1\right)}.
\end{aligned}$$

ゆえに $t \geq 0$ における $p = F(t)$ の逆函数である分位点函数は

$$t = Q(p) = F^{-1}(p) = \sqrt{\nu\left(\frac{1}{1-I^{-1}(2p-1|\alpha,\beta)} - 1\right)}$$

と書ける.

$P(-c \leq T \leq c) = 1 - \alpha$ と $F(c) = P(T \leq c) = 1 - \alpha/2$ は同値なので, そのような $c \geq 0$ は次のように表される:

$$c = Q(1-\alpha/2) = \sqrt{\nu\left(\frac{1}{1-I^{-1}(2p-1|\alpha,\beta)} - 1\right)}.$$

$\nu = 9$ のとき,

- $P(-c \leq T \leq c) = 95\%$ となる $c \geq 0$ は $c \approx 2.26$
- $P(-c \leq T \leq c) = 99\%$ となる $c \geq 0$ は $c \approx 3.25$

$\nu = 19$ のとき,

- $P(-c \leq T \leq c) = 95\%$ となる $c \geq 0$ は $c \approx 2.09$
- $P(-c \leq T \leq c) = 99\%$ となる $c \geq 0$ は $c \approx 2.86$

$\nu = 29$ のとき,

- $P(-c \le T \le c) = 95\%$ となる $c \ge 0$ は $c \approx 2.05$
- $P(-c \le T \le c) = 99\%$ となる $c \ge 0$ は $c \approx 2.76$

ちなみに $Z \sim \mathrm{Normal}(0, 1)$ のとき,

- $P(-c \le Z \le c) = 95\%$ となる $c \ge 0$ は $c \approx 1.96$
- $P(-c \le Z \le c) = 99\%$ となる $c \ge 0$ は $c \approx 2.58$

解答終

In [19]:

```
quantile_tdist(ν, p) = √(ν * (1/(1 - beta_inc_inv(1/2, ν/2, 2p - 1)[1]) - 1))
quantile_stdnormal(p) = √2 * erfinv(2p - 1)

for ν in (9, 19, 29)
    @show ν
    @show quantile_tdist(ν, 0.975)
    @show quantile_tdist(ν, 0.995)
    println()
end
ν = Inf
@show ν
@show quantile_stdnormal(0.975)
@show quantile_stdnormal(0.995);
```

```
ν = 9
quantile_tdist(ν, 0.975) = 2.262157162798204
quantile_tdist(ν, 0.995) = 3.2498355415921254

ν = 19
quantile_tdist(ν, 0.975) = 2.0930240544083096
quantile_tdist(ν, 0.995) = 2.860934606464978

ν = 29
quantile_tdist(ν, 0.975) = 2.0452296421327034
quantile_tdist(ν, 0.995) = 2.756385903670605

ν = Inf
quantile_stdnormal(0.975) = 1.9599639845400576
quantile_stdnormal(0.995) = 2.5758293035489053
```

In [20]:

```
for ν in (9, 19, 29)
    @show ν
    @show quantile(TDist(ν), 0.975)
    @show quantile(TDist(ν), 0.995)
    println()
end
ν = Inf
@show ν
@show quantile(TDist(ν), 0.975)
@show quantile(TDist(ν), 0.995);
```

```
ν = 9
quantile(TDist(ν), 0.975) = 2.262157162798205
quantile(TDist(ν), 0.995) = 3.249835541592126

ν = 19
quantile(TDist(ν), 0.975) = 2.093024054408309
quantile(TDist(ν), 0.995) = 2.860934606464979

ν = 29
quantile(TDist(ν), 0.975) = 2.0452296421327034
quantile(TDist(ν), 0.995) = 2.756385903670605

ν = Inf
quantile(TDist(ν), 0.975) = 1.9599639845400536
quantile(TDist(ν), 0.995) = 2.5758293035489
```

例えば $\nu = 9$ のとき, WolframAlphaでは以下のように計算できる:

- sqrt(n (1/(1 - InverseBetaRegularized(0.95, 1/2, n/2)) - 1)) where n = 9 (https://www.wolframalpha.com/input?i=sqrt%28n+%281%2F%281+-+InverseBetaRegularized%280.95%2C+1%2F2%2C+n%2F2%29%29+-+1%29%29+where+n+%3D+9),

- sqrt(n (1/(1 - InverseBetaRegularized(0.99, 1/2, n/2)) - 1)) where n = 9 (https://www.wolframalpha.com/input?i=sqrt%28n+%281%2F%281+-+InverseBetaRegularized%280.99%2C+1%2F2%2C+n%2F2%29%29+-+1%29%29+where+n+%3D+9),
- quantile(StudentTDistribution(9), 0.995) (https://www.wolframalpha.com/input?i=quantile%28StudentTDistribution%289%29%2C+0.995%29)
- quantile(StudentTDistribution(9), 0.975) (https://www.wolframalpha.com/input?i=quantile%28StudentTDistribution%289%29%2C+0.975%29)

3 Poisson分布の導入とその解釈

3.1 Poisson分布の定義

$\lambda > 0$ と仮定する. 確率質量函数

$$p(k|\lambda) = e^{-\lambda}\frac{\lambda^k}{k!} \quad (k = 0, 1, 2, \ldots)$$

で定義される無限離散分布を **Poisson分布** (ポアソン分布)と呼び, 次のように表すことにする:

$$\mathrm{Poisson}(\lambda).$$

Poisson分布は一定期間内に起こるイベントの回数の分布のモデル化としてよく使われている.

In [21]:

```
PP = []
for (λ, s) in ((1, 1), (3, 3), (10, 5), (30, 10) )
    x = max(0, round(λ-4√λ)):λ+4√λ+3/√λ
    P = bar(x, k -> pdf(Poisson(λ), k);
        alpha=0.3, label="", title="Poisson($λ)", xtick=0:s:maximum(x))
    push!(PP, P)
end
plot(PP...; size=(800, 150), layout=(1, 4))
```

Out[21]:

以上はPoisson分布 $\mathrm{Poisson}(\lambda)$ の確率質量函数のグラフの例である. ここでは λ が整数の場合のみを扱ったが, λ は整数でなくてもよい. $\mathrm{Poisson}(30)$ のグラフは正規分布のグラフに近くなっている. **パラメータ λ が大きなPoisson分布は正規分布で近似される.**

3.2 Poisson分布における確率の総和が1になることの確認

指数函数のTaylor展開によって,

$$\sum_{k=0}^{\infty} e^{-\lambda}\frac{\lambda^k}{k!} = e^{-\lambda}\sum_{k=0}^{\infty}\frac{\lambda^k}{k!} = e^{-\lambda}e^{\lambda} = 1.$$

Poisson分布は指数函数のTaylor展開から得られる離散分布である.

Poisson分布よりも複雑な負の二項分布は函数 $p^{-\alpha} = (1-(1-p))^{-\alpha}$ の $1-p$ に関するTaylor展開

$$p^{-\alpha} = (1-(1-p))^{-\alpha} = \sum_{m=0}^{\infty}(-1)^m\binom{-\alpha}{m}(1-p)^m = \sum_{m=0}^{\infty}\binom{\alpha+m-1}{m}(1-p)^m$$

から得られるのであった. このパターンはよく現れる.

3.3 問題: Poisson分布のキュムラント母函数と期待値と分散

$K_\lambda \sim \mathrm{Poisson}(\lambda)$ のとき次が成立することを示せ:

$$\log E[e^{tK_\lambda}] = \lambda(e^t - 1), \quad E[K_\lambda] = \mathrm{var}(K_\lambda) = \lambda.$$

さらに K_λ の標準化 Z_λ を $Z_\lambda = (K_\lambda - \lambda)/\sqrt{\lambda}$ と定めると次が成立することも示せ:

$$\log E[e^{tZ_\lambda}] = \log E\left[e^{t(K_\lambda-\lambda)/\sqrt{\lambda}}\right] = \frac{t^2}{2} + O(\lambda^{-1/2}).$$

ここで $O(\lambda^{-1/2})$ の部分は $O(\lambda)^{-1/2} = \lambda^{-1/2}\times$ ($\lambda \to \infty$ で有界な量)であることを表している. たとえば, $f(\lambda)$ が $\lambda \to \infty$ で収束するとき, $\lambda^{-1/2}f(\lambda)$ は $O(\lambda^{-1/2})$ と表される. 特に $O(\lambda^{-1/2})$ の部分は $\lambda \to \infty$ で 0 に収束する.

解答例:

$$E[e^{tK_\lambda}] = e^{-\lambda}\sum_{k=0}^{\infty} e^{tk}\frac{\lambda^k}{k!} = e^{-\lambda}\sum_{k=0}^{\infty}\frac{(\lambda e^t)^k}{k!} = e^{\lambda(e^t-1)},$$
$$\log E[e^{tK_\lambda}] = \lambda(e^t-1) = \lambda\left(t + \frac{t^2}{2} + O(t^3)\right) = \lambda t + \lambda\frac{t^2}{2} + O(t^3).$$

$\log E[e^{tK_\lambda}]$ の展開における t, $t^2/2$ の係数がそれぞれ K_λ の期待値, 分散になるので

$$E[K_\lambda] = \mathrm{var}(K_\lambda) = \lambda.$$

さらに, $Z_\lambda = (K_\lambda - \lambda)/\sqrt{\lambda})$ について

$$\begin{aligned}\log E[e^{tZ_\lambda}] &= \log\left(e^{-\sqrt{\lambda}t}E\left[e^{\left(t/\sqrt{\lambda}\right)K_\lambda}\right]\right) = \log E\left[e^{\left(t/\sqrt{\lambda}\right)K_\lambda}\right] - \sqrt{\lambda}t)\\ &= \lambda\left(e^{t/\sqrt{\lambda}}-1\right) - \sqrt{\lambda}t) = \lambda\left(\frac{t}{\sqrt{\lambda}} + \frac{t^2}{2\lambda} + O(\lambda^{-3/2})\right) - \sqrt{\lambda}t\\ &= \left(\sqrt{\lambda}t + \frac{t^2}{2} + O(\lambda^{-1/2})\right) - \sqrt{\lambda}t = \frac{t^2}{2} + O(\lambda^{-1/2}).\end{aligned}$$

解答終

注意: 上の問題の結果より, $K_\lambda \sim \mathrm{Poisson}(\lambda)$, $Z_\lambda = (K_\lambda - \lambda)/\sqrt{\lambda}$, $Z_\infty \sim \mathrm{Normal}(0,1)$ のとき, λ を大きくすると $\log E[e^{tZ_\lambda}]$ が $\log E[e^{tZ_\infty}]$ に収束する. (一般に確率変数 X の分布はそのキュムラント母函数 $\log E[e^{tX}]$ から一意的に決まる. この点については別のノートで説明する.) これは実は λ が大きなとき, $Z_\lambda = (K_\lambda - \lambda)/\sqrt{\lambda}$ の従う分布が標準正規分布で近似されることを意味している. そのとき, もとの K_λ の分布は平均 λ, 分散 λ の正規分布で近似される. この結果を, **Poisson分布の中心極限定理** と呼ぶ.

In [22]:

```
PP = []
for λ in (10, 30)
    P = bar(max(0, round(λ-4√λ)):λ+4√λ+3/√λ, k -> pdf(Poisson(λ), k);
        alpha=0.3, label="", title="Poisson(λ = $λ)")
    plot!(Normal(λ, √λ); label="Normal(λ, √λ)", lw=2)
    push!(PP, P)
end
plot(PP...; size=(800, 250), layout=(1, 2))
```

Out[22]:

3.4 二項分布の連続時間極限

期待値 λ のPoisson分布は同じ期待値を持つ p が小さな二項分布によって近似されることを示そう. 二項分布 $\mathrm{Bonomial}(n,p)$ の期待値は np なので, $p = \lambda/n$ とおくと期待値は λ になる. そのように p を定めて, $n \to \infty$ とすると, 以下のように, 二項分布の確率質量函数はPoisson分布の確率質量函数に収束する:

$$
\begin{aligned}
\binom{n}{k} p^k (1-p)^{n-k} &= \frac{n(n-1)\cdots(n-k+1)}{k!}\left(\frac{\lambda}{n}\right)^k \left(1-\frac{\lambda}{n}\right)^{n-k} \\
&= \underbrace{\left(1-\frac{1}{n}\right)\cdots\left(1-\frac{k-1}{n}\right)}_{\to 1} \underbrace{\left(1-\frac{\lambda}{n}\right)^{n-k}}_{\to \exp(-\lambda)} \frac{\lambda^k}{k!} \to e^{-\lambda}\frac{\lambda^k}{k!}.
\end{aligned}
$$

1つめの等号で二項係数の定義と $p = \lambda/n$ を使い, 2つめの等号で分子の $n(n-1)\cdots(n-k+1)$ を $(\lambda/n)^k = \lambda^k/n^k$ の分母の n^k で割った. 最後に $n \to \infty$ の極限を取った.

3.5 Poisson分布は単位時間内に起こるイベントの回数の分布だとみなされる

前節の結果は以下のように解釈される.

(1) 二項分布は n 回のBernoulli試行で出る 1 の個数の分布であった. 1 が出ることを「イベントが起こった」と解釈することにする.

(2) n 回の試行を単位時間内に等間隔で行う状況を考える. これは時間の刻み幅が $1/n$ の離散時間を考えることに相当する. そのとき, 二項分布は単位時間のあいだに平均して np 回起こるイベントが単位時間内で何回起こるかに関する分布になっている.

(3) 単位時間内に起こる二項分布に従うイベントの回数の期待値 np を $np = \lambda$ と固定したままで, $n \to \infty$ とすることは, 時間の刻み幅を細かくする連続時間極限を取ることだと考えられる. この極限によって, 二項分布はPoisson分布に収束する.

(4) **Poisson分布は, 連続時間の場合に単位時間のあいだに平均して λ 回起こるイベントが単位時間内で起こる回数の分布になっている.**

In [23]:
```
function plot_binpoi(λ, n)
    @assert n > λ
    p = λ/n
    x = range(max(-0.5, λ - 5√λ), λ + 6√λ, 1000)
    plot(x, x -> pdf(Binomial(n, p), round(Int, x)); label="Binomial(n, p)")
    plot!(x, x -> pdf(Poisson(λ), round(Int, x)); label="Poisson(λ)", ls=:dash)
    title!("λ = $λ, n = $n, p = λ/n")
end
```

Out[23]: plot_binpoi (generic function with 1 method)

In [24]:
```
plot(plot_binpoi(1, 5), plot_binpoi(1, 15), plot_binpoi(1, 50); size=(800, 200), layout=(1, 3)
```

Out[24]:

In [25]:
```
plot(plot_binpoi(3, 15), plot_binpoi(3, 45), plot_binpoi(3, 150); size=(800, 200), layout=(1,
```

Out[25]:

In [26]:
```
plot(plot_binpoi(10, 50), plot_binpoi(10, 150), plot_binpoi(10, 500); size=(800, 200), layout=
```

Out[26]:

以上のグラフを見れば, $p = \lambda/n$ とおくとき, n を大きくすると, 二項分布が確かにPoisson分布で近似されていることがわかる.

3.6 Poisson分布の中心極限定理と二項分布の中心極限定理の関係

一般に期待値 μ, 標準偏差 σ を持つ確率変数 X の **標準化** Z は $Z = (X - \mu)/\sigma$ と定義される.

そのとき $\log E[e^{tZ}]$ を X の **標準化キュムラント母函数** と呼ぶ.

標準化キュムラント母函数が正規分布の標準化キュムラント母函数 $t^2/2$ にどれだけ近いかは, その分布が正規分布にどれだけ近いかを表している(この点については後で中心極限定理について一般的に説明するときに再度触れる).

Poisson分布のキュムラント母函数に関する問題の結果によれば,

$$K_\lambda \sim \mathrm{Poisson}(\lambda), \quad Z_\lambda = \frac{K_\lambda - \lambda}{\sqrt{\lambda}} \quad \Longrightarrow \quad \log E[e^{tZ_\lambda}] = \frac{t^2}{2} + O(\lambda^{-1/2}).$$

これより, λ を大きくすると, Poisson分布は正規分布で近似されるようになることがわかる. この結果を**Poisson分布の中心極限定理**と呼ぶことにする.

さらに, 前節の結果より, $\lambda = np$ を一定としたままで, n を大きくすると, 二項分布 $\mathrm{Binomial}(n, p)$ はPoisson分布 $\mathrm{Poisson}(\lambda)$ で近似される.

そのとき注意するべきことは $\mathrm{Binomial}(n, p)$ の分散 $np(1-p)$ は $np = \lambda$ のとき $np(1-p) = \lambda(1-p)$ になり, p が小さくないと, $\mathrm{Poisson}(\lambda)$ の分散 λ と全然違う値になってしまうことである(たとえば $p = 1/2$ だと二項分布の分散が対応するPoisson分布の分散の半分になってしまう). 二項分布によるPoisson分布の近似は p が小さくないと精度が低くなる.

しかし, 以上の2つの結果を合わせると, Poisson分布の正規分布近似(=中心極限定理)を通して, p は小さいが np が大きい場合の二項分布 $\mathrm{Binomial}(n, p)$ が正規分布で近似されることがわかる. 実際には, p が小さくなくても, np と $n(1-p)$ が大きければ, 二項分布は正規分布でよく近似される. 二項分布が正規分布で近似されるという結果を **二項分布の中心極限定理** と呼ぶ.

p が小さい場合の二項分布の中心極限定理は応用上あまりにも不完全なので, 後で二項分布を直接正規分布で近似するという p が小さくない場合にも通用する方法で二項分布の中心極限定理を証明する.

注意: 「中心極限定理」は「極限として中心に収束する定理」という意味では **ない**. 「中心極限定理」は「確率論における中心的な極限定理」という意味である. 「中心極限定理」はある種の状況で分布が正規分布で近似されるようになるという結果に付けられた名前である.

In [27]:

```
PP = []
for (λ, n) in ((15, 20), (15, 300))
    p = λ/n
    ks = max(0, round(Int, λ-5.5√λ)):round(Int, λ+5.5√λ+3/√λ)
    P = groupedbar(ks,
        [pdf.(Poisson(λ), ks) pdf.(Binomial(n, λ/n), ks)];
        alpha=0.3,
        label=["Poisson(λ)" "Binomial(n,λ/n)"],
        title="λ = $λ,  n = $n,  p = λ/n = $p")
    plot!(Normal(λ, √λ); label="Normal(λ,√λ)", lw=1.5, c=:blue)
    plot!(Normal(λ, √(λ*(1-p))); label="Normal(λ,√(λ(1-p)))", lw=1.5, ls=:dash, c=:red)
    push!(PP, P)
end
plot(PP...; size=(800, 250), layout=(1, 2))
```

Out[27]:

左側のグラフでは $p = \lambda/n = 3/4$ なので, 二項分布の側の分散はPoisson分布の側の分散の4分の1になっており, 標準偏差については半分になっている.

右側のグラフでは $p = \lambda/n = 1/20$ が小さくなっているので, 二項分布のグラフとPoisson分布のグラフはほとんど一致し, 正規分布による近似もほとんど一致する.

3.7 問題: Poisson分布の中心極限定理の直接証明

パラメータ $\lambda > 0$ を持つPoisson分布の確率質量函数の定義は

$$p(k|\lambda) = e^{-\lambda}\frac{\lambda^k}{k!} \quad (k = 0, 1, 2, \ldots)$$

であった. k は λ ごとに異なる値を取るものとし, 固定された x について

$$k = \lambda + \sqrt{\lambda}\,x + o(\sqrt{\lambda})$$

を満たしていると仮定する. ここで $o(\sqrt{\lambda})$ は $\sqrt{\lambda}$ で割ると $\lambda \to \infty$ で 0 に収束する量を表す. このとき次が成立することを示せ:

$$p(k|\lambda) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}\frac{1}{\sqrt{\lambda}}\,(1 + o(1)).$$

ここで $o(1)$ は $\lambda \to \infty$ で 0 に収束する量である.

注意: 以下の解答例は二項分布の中心極限定理の証明のひな型になる.

解答例: まず, 問題文よりも弱い次の条件を仮定する:

$$k = \lambda + o(\lambda) = \lambda(1 + o(1))$$

ここで $o(\lambda)$ は λ で割ると $\lambda \to \infty$ で 0 に収束する量である. このとき, $k!$ にStirlingの公式を適用すると,

$$p(k|\lambda) = e^{-\lambda}\frac{\lambda^k}{k^k e^{-k}\sqrt{2\pi k}}(1 + o(1)) = \frac{1}{\sqrt{2\pi\lambda}}\left(\frac{k}{\lambda}\right)^{-k} e^{k-\lambda}(1 + o(1)).$$

そして,

$$-\log\left(\left(\frac{k}{\lambda}\right)^{-k} e^{k-\lambda}\right) = k\log\frac{k}{\lambda} - (k - \lambda)$$

を $f(k)$ と書くと, $f(\lambda) = 0$, $f'(k) = \log(k/\lambda)$, $f'(\lambda) = 0$, $f''(k) = 1/k$, $f''(\lambda) = 1/\lambda$, $f'''(k) = -1/k^2$, $f'''(\lambda) = -1/\lambda^2$, etc. なので, $f(k)$ は $k = \lambda$ で次のようにTaylor展開される:

$$k\log\frac{k}{\lambda} + k - \lambda = f(k) = \frac{1}{2}\frac{(k-\lambda)^2}{\lambda} - \frac{1}{3!}\frac{(k-\lambda)^3}{\lambda^2} + \cdots$$

ゆえに, 問題文のように

$$k = \lambda + \sqrt{\lambda}\,x + o(\sqrt{\lambda})$$

と仮定すると,

$$\frac{k-\lambda}{\lambda} = \frac{x}{\sqrt{\lambda}} + o(\lambda^{-1/2}) = O(\lambda^{-1/2}), \quad \frac{(k-\lambda)^2}{\lambda^2} = \frac{x^2}{\lambda} + o(\lambda^{-1}) = O(\lambda^{-1}), \quad \text{etc.}$$

なので

$$\frac{(k-\lambda)^2}{\lambda} = x^2 + o(1), \quad \frac{(k-\lambda)^3}{\lambda^2} = O(\lambda^{-1/2}) = o(1), \quad \text{etc.}$$

となり, 次が得られる:

$$k\log\frac{k}{\lambda} + k - \lambda = \frac{x^2}{2} + o(1).$$

したがって,

$$p(k|\lambda) = \frac{1}{\sqrt{2\pi\lambda}}\exp\left(-\frac{x^2}{2} + o(1)\right)(1 + o(1)) = \frac{1}{\sqrt{2\pi\lambda}}e^{-x^2/2}(1 + o(1)).$$

これで示したい公式が得られた.

解答終

注意: 上の計算の途中で出て来た式

$$\log\left(\left(\frac{k}{\lambda}\right)^{-k} e^{k-\lambda}\right) = -k\log\frac{k}{\lambda} + k - \lambda$$

における $k\log(k/\lambda)$ の部分は **Kullback-Leibler情報量**

$$D(q||p) = \sum_i q_i \log\frac{q_i}{p_i}$$

の各項に化ける式になっている.

注意: 上の計算の途中で出て来た式

$$\frac{(k-\lambda)^2}{\lambda} = x^2 + o(1)$$

の左辺は, **Pearsonのχ^2検定量**

$$\sum_i \frac{(\text{observation}_i - \text{expectation}_i)^2}{\text{expectation}_i}$$

の各項に化ける式になっている.

注意: このように, Poisson分布の中心極限定理を直接証明することには, 統計学的に重要な数学的対象の構成要素を得る行為にもなっている!

4 負の二項分布の「期待値＝分散が確率的に揺らいでいるPoisson分布」という解釈

4.1 Poisson分布のパラメータがガンマ分布に従っていれば負の二項分布が得られる

Poisson分布は単位時間内に起こるイベントの回数の分布とみなされるのであった.

Poisson分布でモデル化できると期待される現実の現象のデータを複数の対象について取得したとする. そのとき, そのすべてについて, 一定期間のあいだにイベントが起こる回数の期待値 λ が等しいと期待できるならば, Poisson分布モデルを単純に適用すればよい.

しかし, データを取得した対象ごとに, 一定期間のあいだにイベントが起こる回数の期待値 λ が異なると考えられる場合にはどうすればよいだろうか?

例えば, 野球選手達が年間何本ホームランを打つかについてデータを取得したとする. 選手ごとに打つホームランの本数の期待値は大きく異なる.

そのような場合には, ホームランの本数の期待値の選手間でのばらつき方も確率分布でモデル化すればよい.

たとえば一定期間内で打つホームランの本数の期待値の分布をガンマ分布でモデル化するとどうなるだろうか?

実はその場合には, ホームランの本数の分布を負の二項分布でモデル化したのと同じことになる.

次の問題ではそのことを実際に計算して確認してもらおう.

4.2 問題: Poisson分布とガンマ分布から負の二項分布が得られる

$\alpha, \theta > 0$ であるとし, 確率変数 Λ は次のようにガンマ分布に従っていると仮定する:

$$\Lambda \sim \mathrm{Gamma}(\alpha, \theta).$$

さらに確率変数 M はパラメータが確率変数 Λ であるようなPoisson分布に従っていると仮定する:

$$M \sim \mathrm{Poisson}(\Lambda).$$

これの意味は乱数の生成を考えるとわかりやすい. この確率変数 M に対応する乱数は以下の手続きで生成されると考える:

1. 分布 $\mathrm{Gamma}(\alpha, \theta)$ に従って乱数 Λ を生成する.
2. 分布 $\mathrm{Poisson}(\Lambda)$ に従って乱数 M を生成する.

このとき, M の分布が従う確率質量函数は次になることを示せ:

$$P(m|\alpha, \theta) = \binom{\alpha + m - 1}{m} \left(\frac{1}{1+\theta}\right)^{\alpha} \left(1 - \frac{1}{1+\theta}\right)^{m} \quad (m = 0, 1, 2, \ldots)$$

これは負の二項分布 $\mathrm{NegativeBinomial}(\alpha, \theta/(1+\theta))$ の確率質量函数に一致する. これによって, 負の二項分布の通常の解釈「α 回成功するまでに失敗する回数の分布」とは別の解釈が得られた. 負の二項分布は「期待値パラメータが固定されてなくてガンマ分布に従っているようなPoisson分布」であるとも考えられる. (この場合には「成功回数」にあたる α は整数でなくても任意の正の実数になれる.)

ただし, $\mathrm{Gamma}(\alpha, \theta)$ の確率密度函数を $p(\lambda|\alpha, \theta)$ と書き, $\mathrm{Poisson}(\lambda)$ の確率質量函数を $P(m|\lambda)$ と書くとき, 上のような状況における M が従う分布の確率質量函数 $P(m|\alpha, \theta)$ は

$$P(m|\alpha, \theta) = \int_0^{\infty} P(m|\lambda) p(\lambda|\alpha, \theta)\, d\lambda$$

と定義される.

解答例: $p(\lambda|\alpha, \theta)$, $P(k|\lambda)$ の定義は次の通り:

$$p(\lambda|\alpha, \theta) = \frac{e^{-\lambda/\theta} x^{\alpha-1}}{\theta^{\alpha}\Gamma(\alpha)}, \quad P(k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}.$$

ゆえに, ガンマ函数の基本的な使い方である

$$\int_0^\infty e^{-\lambda/\eta}\lambda^{\beta-1}\,d\lambda = \eta^\beta\Gamma(\beta)$$

を $\eta = \theta/(1+\theta)$, $\beta = \alpha + m$ の場合に使うと,

$$\begin{aligned}
P(m|\alpha,\theta) &= \int_0^\infty P(m|\lambda)p(\lambda|\alpha,\theta)\,d\lambda = \int_0^\infty \frac{e^{-\lambda}\lambda^m}{m!}\frac{e^{-\lambda/\theta}x^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}\,d\lambda \\
&= \frac{1}{m!\theta^\alpha\Gamma(\alpha)}\int_0^\infty e^{-((1+\theta)/\theta)\lambda}\lambda^{\alpha+m-1}\,d\lambda\,d\lambda \\
&= \frac{1}{m!\theta^\alpha\Gamma(\alpha)}\left(\frac{\theta}{1+\theta}\right)^{\alpha+m}\Gamma(\alpha+m) \\
&= \frac{(\alpha+m-1)\cdots(\alpha+1)\alpha}{m!}\left(\frac{1}{1+\theta}\right)^\alpha\left(\frac{\theta}{1+\theta}\right)^m \\
&= \binom{\alpha+m-1}{m}\left(\frac{1}{1+\theta}\right)^\alpha\left(1-\frac{1}{1+\theta}\right)^m.
\end{aligned}$$

解答終

この手の積分計算をサボりたい人は, 以下のようにコンピュータでガンマ分布の乱数を発生させて, それをパラメータとするPoisson分布の乱数を発生させることを大量に実行し, それと負の二項分布を直接使って生成した乱数の分布を比較してみるとよい.

In [28]:
```
function plot_gampoi(α, θ; L = 10^6)
    p = 1/(1 + θ)
    Λ = rand(Gamma(α, θ), L) # ガンマ分布で乱数達を大量生成
    M_gampoi = @. rand(Poisson(Λ)) # その各々を期待値とするPoisson分布の乱数を生成
    M_negbin = rand(NegativeBinomial(α, p), L) # 直接的に負の二項分布の乱数を大量生成

    # 比較のための同時プロット
    binmin, binmax = round.(quantile.(Ref(M_negbin), (0.001, 0.999)))
    stephist(M_gampoi; norm=true, bin=binmin-0.5:binmax+0.5, label="Gamma-Poisson")
    stephist!(M_negbin; norm=true, bin=binmin-0.5:binmax+0.5, ls=:dash, label="NegativeBinomia
end
```

Out[28]:
```
plot_gampoi (generic function with 1 method)
```

In [29]:
```
plot(plot_gampoi(1, 3), plot_gampoi(3, 1); size=(800, 250), layout=(1, 2))
```

Out[29]:

In [30]:
```
plot(plot_gampoi(6, 2), plot_gampoi(10, 1); size=(800, 250), layout=(1, 2))
```

Out[30]:

このようにMonte Carloシミュレーション(コンピュータで乱数を発生させることによるシミュレーション)でも, 期待値パラメータがガンマ分布に従っているようなPoisson分布と負の二項分布はぴったり一致している.

注意: $p = 1/(1+\theta)$ とおくと, $1-p = \theta/(1+\theta)$ になるので, 負の二項分布 $\mathrm{NegativeBinomial}(\alpha, \theta/(1+\theta))$ の期待値と分散はそれぞれ次のように書ける:

$$E[M] = \frac{\alpha(1-p)}{p} = \alpha\theta, \quad \mathrm{var}(M) = \frac{\alpha(1-p)}{p^2} = \alpha\theta(1+\theta).$$

Poisson分布では期待値と分散が等しかったが, この場合には $1+\theta$ 倍の分だけ分散が期待値よりも大きくなっている. これより, $\mu = \alpha\theta$ を一定に保ったままで $\theta \searrow 0$ とすると負の二項分布 $\mathrm{NegativeBinomial}(\alpha, \theta/(1+\theta))$ がPoisson分布 $\mathrm{Poissom}(\mu)$ に収束すると予想されるが実際にそうなる: $\alpha = \mu/\theta$ とおいて, $\theta \searrow 0$ とすると,

$$\begin{aligned}
P(m|\alpha,\theta) &= \binom{\alpha+m-1}{m}\left(\frac{1}{1+\theta}\right)^{\alpha}\left(1-\frac{1}{1+\theta}\right)^{m} \\
&= \frac{(\mu/\theta+m-1)\cdots(\mu/\theta+1)(\mu/\theta)}{m!}\left(\frac{1}{1+\theta}\right)^{\mu/\theta}\left(\frac{\theta}{1+\theta}\right)^{m} \\
&= \underbrace{\frac{(\mu+(m-1)\theta)\cdots(\mu+\theta)\mu}{m!}}_{\to\mu^m/m!}\underbrace{\left(\frac{1}{1+\theta}\right)^{\mu/\theta}}_{\to\exp(-\mu)}\underbrace{\left(\frac{1}{1+\theta}\right)^{m}}_{\to 1} \to \frac{e^{-\mu}\mu^m}{m!}.
\end{aligned}$$

$\alpha = \mu/\theta$ とおいて, θ を小さくすると, $\mathrm{Gamma}(\mu/\theta, \theta)$ の分布は μ に集中して行くのでこうなるとも考えられる.

In [31]:

```
PP = []
μ = 3
for α in (10, 30, 100)
    θ = μ/α
    P = plot(Gamma(α,θ), 0, 6; label="", title="Gamma($α, $θ)")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[31]:

In [32]:

```
PP = []
μ = 3
for α in (10, 30, 100)
    θ = μ/α
    p = 1/(1 + θ)
    x = max(0, round(Int, μ - 4√μ)):round(Int, μ + 4√μ)
    y1 = pdf.(NegativeBinomial(α, p), x)
    y2 = pdf.(Poisson(μ), x)
    P = groupedbar(x, [y1 y2]; alpha=0.5, label=["NegBin" "Poisson"])
    title!("μ = $μ, α = $α, θ = $θ")
    push!(PP, P)
end
plot(PP...; size=(800, 200), layout=(1, 3))
```

Out[32]:

5 ガンマ分布の解釈

形状パラメータ $\alpha > 0$, スケールパラメータ $\theta > 0$ を持つガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ の確率密度函数は

$$p(y|\alpha, \beta) = \frac{1}{\theta^\alpha \Gamma(\alpha)} e^{-y/\theta} y^{\alpha-1} \quad (y > 0)$$

であった. 我々は別のノートでガンマ分布を「正規分布に関係した分布」として導入したのであった(χ²分布はガンマ分布の特別な場合であった). 以下の2つの問題を見よ. その後にガンマ分布の別の解釈を説明する.

5.1 問題: 標準正規分布に従う確率変数の2乗は自由度1のχ²分布に従う

標準正規分布に従う確率変数の2乗は自由度1のχ²分布に従うことを示せ.

解答例: $Z \sim \mathrm{Normal}(0, 1)$ のとき, $Y = Z^2$ とおくと,

$$\begin{aligned} E[f(Y)] &= \int_{-\infty}^{\infty} f(z^2) \frac{e^{-z^2/2}}{\sqrt{2\pi}} \, dz = 2 \int_0^{\infty} f(z^2) \frac{e^{-z^2}/2}{\sqrt{2\pi}} \, dz \\ &= \int_0^{\infty} f(y) \frac{e^{-y/2}}{\sqrt{2\pi}} y^{-1/2} \, dy = \int_0^{\infty} f(y) \frac{e^{-y/2} y^{1/2-1}}{2^{1/2}\Gamma(1/2)} \, dy \end{aligned}$$

2つめの等号で被積分函数が偶函数であることを使い, 3つめの等号で $z = \sqrt{y}$ とおき, 最後の等号で $\Gamma(1/2) = \sqrt{\pi}$ を使った. これは Y がχ²分布に従うことを意味している.

解答終

In [33]:

```
L = 10^6
Z = rand(Normal(), L)
Y = @. Z^2
histogram(Y; norm=true, alpha=0.3, label="Y = Z²")
plot!(Chisq(1); label="Chisq(1)", lw=1.5)
plot!(; xlim=(-0.2, 4.2), ylim=(-0.2, 4.2))
```

Out[33]:

5.2 問題: 標準正規分布に従うn個の独立な確率変数達の2乗は自由度nのχ²分布に従う

標準正規分布に従う n 個の独立な確率変数達の2乗は自由度 n のχ²分布に従うことを示せ.

解答例: $Z_1, \ldots, Z_n$ は独立な確率変数達であるとし, $Z_i \sim \mathrm{Normal}(0, 1)$ となっていると仮定する. このとき, 1つ上の問題より, 各 i ごとに $Z_i^2 \sim \mathrm{Chisq}(1) = \mathrm{Gamma}(1/2, 1/2)$ となり, $Z_1^2, \ldots, Z_n^2$ は独立な確率変数達になる.

一般に, X, Y が独立な確率変数達で $X \sim \mathrm{Gamma}(\alpha, \theta)$, $Y \sim \mathrm{Gamma}(\beta, \theta)$ のとき, $X + Y \sim \mathrm{Gamma}(\alpha + \beta, \theta)$, $X/(X+Y) \sim \mathrm{Beta}(\alpha, \beta)$ で $X + Y$, $X/(X+Y)$ は独立になるのであった. この結果を上の状況で使うと,

$$Z_1^2 + \cdots + Z_n^2 \sim \mathrm{Gamma}(\underbrace{1/2 + \cdots + 1/2}_{n}, 1/2) = \mathrm{Gamma}(n/2, 1/2) = \mathrm{Chisq}(n)$$

となることがわかる.

解答終

In [34]:

```
L = 10^6
n = 5
Z = rand(Normal(), n, L)
Y = [sum(z -> z^2, Z) for Z in eachcol(Z)]
histogram(Y; norm=true, alpha=0.3, bin=100, label="Z₁² + ⋯ + Zₙ²")
plot!(Chisq(n); label="Chisq(1)", lw=1.5)
plot!(; xlim=(-1, 20))
title!("n = $n")
```

Out[34]:

5.3 負の二項分布の連続時間極限

ガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ が負の二項分布の連続極限になっていることを説明する. 以下ではガンマ分布が時間幅に関する確率分布だとみなしたいので, ガンマ分布に従う確率変数を T と書き, 対応する通常の変数を t と書くことにする.

負の二項分布は成功確率 $0 < p \leq 1$ のBernoulli試行を $\alpha > 0$ 回成功するまで続けたときに失敗する回数 M が従う分布であり, その確率質量函数は

$$P(m|\alpha, p) = \binom{\alpha + m - 1}{m} p^\alpha (1-p)^m \quad (m = 0, 1, 2, \ldots)$$

と書けるのであった(分布がうまく定義されるために α は整数でなくてもよい). このときの総試行回数 N は $N = \alpha + M$ と書ける. N が取り得る値の全体は $\alpha, \alpha+1, \alpha+2, \ldots$ になる.

一般に二項係数は

$$\binom{\alpha + m - 1}{m} = \frac{\Gamma(\alpha+m)}{\Gamma(m+1)\Gamma(\alpha)} = \frac{\Gamma(\alpha+m+1)}{(\alpha+m)\Gamma(m+1)\Gamma(\alpha)} = \frac{1}{(\alpha+m)B(\alpha, m+1)}$$

とベータ函数で書けるので, N が従う分布の確率質量函数は次のように書ける:

$$P(n-\alpha|\alpha, p) = \frac{1}{nB(\alpha, n-\alpha+1)} p^\alpha (1-p)^{n-\alpha}$$

$L > 0$ であるとし, 単位時間に L 回にBenoulli試行を行い, 単位時間内での成功回数(イベントが起こる回数)の期待値 Lp は一定であり,

$$Lp = \frac{1}{\theta}$$

を満たしていると仮定する(p はこれによって L の函数とみなす). これは固定された θ について, 単位時間に平均して $1/\theta$ 回のイベントが起こるという仮定であり, この仮定はイベントが起こる間隔の期待値が θ であるという仮定と同等である.

さらにこのとき, 試行回数を表す通常の変数 n から離散時間変数 t が

$$t = \frac{n}{L}$$

と定義される. このとき単位時間分の試行が行われたとき $n = L$ となり, $t = 1$ となるのでつじつまが合っている.

以上の設定のもとで, 時間の刻み幅を 0 に近付ける $L \to \infty$ で負の二項分布の確率質量函数がどのように振る舞うかを調べよう. $n = Lt$ と $p = 1/(L\theta)$ を上の式に代入すると, $L \to \infty$ において,

$$
\begin{aligned}
P(Lt-\alpha|\alpha,1/(L\theta)) &= \frac{1}{Lt\,B(\alpha,Lt-\alpha+1)}\frac{1}{(L\theta)^\alpha}\left(1-\frac{1}{L\theta}\right)^{Lt-\alpha} \\
&= \frac{1}{(Lt)^\alpha\,B(\alpha,Lt-\alpha+1)}\frac{(Lt)^{\alpha-1}}{(L\theta)^\alpha}\left(1-\frac{1}{L\theta}\right)^{Lt-\alpha} \\
&= \frac{1}{\theta^\alpha\underbrace{(Lt)^\alpha\,B(\alpha,Lt-\alpha+1)}_{\to\Gamma(\alpha)}}\underbrace{\left(1-\frac{1}{L\theta}\right)^{Lt-\alpha}}_{\to\exp(-t/\theta)}t^{\alpha-1}\frac{1}{L} \\
&= \frac{1}{\theta^\alpha\Gamma(\alpha)}e^{-t/\theta}t^{\alpha-1}\frac{1}{L}\,(1+O(1/L)).
\end{aligned}
$$

ここで $K^\alpha B(\alpha,K+b)\to\Gamma(\alpha)$ $(K\to\infty)$ となることを使った.

$1/L$ は離散時間 t の刻み幅なので, この式は時間の刻み幅を 0 に近付ける $L\to\infty$ の極限で, 負の二項分布がガンマ分布に収束することを意味している:

$$
\frac{\alpha+\mathrm{NegativeBinomial}(\alpha,p=1/(L\theta))}{L}\to\mathrm{Gamma}(\alpha,\theta)\quad(L\to\infty)
$$

5.4 負の二項分布の連続時間極限の様子をプロット

前節の計算より, L が大きなとき, ガンマ分布 $\Gamma(\alpha,\theta)$ の確率密度函数 $p(t|\alpha,\theta)$ を近似する函数を, 負の二項分布の確率質量函数 $P(m|\alpha,p)$ を使って次のように作れることがわかる:

$$
p_{\mathrm{approx}}(t|\alpha,\theta,L)=P\big(\mathrm{round}(Lt-\alpha)\mid\alpha,1/(L\theta)\big)L.
$$

$T=N/L$ を $M=N-\alpha$ について解くと, $M=(LT-\alpha)$ であることを使った. 前節の計算から右辺には L をかけておかばければいけないことがわかる. ここで $\mathrm{round}(x)$ は x を四捨五入して整数に変換する函数であるとする.

この事実を使って, 負の二項分布とガンマ分布を比較するグラフを作成してみよう.

In [35]:
```
function plot_nbgam(α, θ, L)
    p = 1/(L*θ)
    @show mean((α + NegativeBinomial(α, p))/L), mean(Gamma(α, θ))
    @show std((α + NegativeBinomial(α, p))/L), std(Gamma(α, θ))
    t = range(max(-1/(2L), α*θ - 4√α*θ), α*θ + 6√α*θ, 1000)
    plot(t, t -> pdf(NegativeBinomial(α, p), round(Int, L*t - α))*L; label="(α+NB(α,p))/L")
    plot!(t, t -> pdf(Gamma(α, θ), t); label="Gamma(α, θ)", ls=:dash)
    title!("α=$α, θ=$θ, L=$L, p=1/(Lθ)")
end
```

Out[35]:
```
plot_nbgam (generic function with 1 method)
```

In [36]:
```
plot(plot_nbgam.(1, 1, (1.2, 3, 10))...; size=(800, 200), layout=(1, 3))
```

```
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (1.0, 1.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (0.40824829046386296, 1.0)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (1.0, 1.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (0.816496580927726, 1.0)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (1.0, 1.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (0.9486832980505137, 1.0)
```

Out[36]:

In [37]:
```
plot(plot_nbgam.(2, 2, (1.2, 3, 10)./2)...; size=(800, 200), layout=(1, 3))
```

```
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (4.0, 4.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (1.1547005383792515, 2.8284271247461
903)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (4.0, 4.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (2.3094010767585034, 2.8284271247461
903)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (4.0, 4.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (2.6832815729997477, 2.8284271247461
903)
```

Out[37]:

In [38]:
```
plot(plot_nbgam.(5, 4, (1.2, 3, 10)./4)...; size=(800, 200), layout=(1, 3))
```

```
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (20.0, 20.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (3.651483716701107, 8.9442719099991
6)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (20.0, 20.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (7.302967433402216, 8.9442719099991
6)
(mean((α + NegativeBinomial(α, p)) / L), mean(Gamma(α, θ))) = (20.0, 20.0)
(std((α + NegativeBinomial(α, p)) / L), std(Gamma(α, θ))) = (8.48528137423857, 8.94427190999916)
```

Out[38]:

5.5 ガンマ分布はイベントがα回起こるまでにかかる時間の分布とみなされる

前節の結果は以下のように解釈される.

(1) 負の二項分布は成功確率 p のBernoulli試行を 1 が α 回出るまでに出た 0 の個数 M の分布であった. 以下では 1 が出たことを「イベントが起こった」と解釈することにし, イベントが α 回起こるまでの試行回数 $N = \alpha + M$ の分布を考える.

(2) さらに, 試行を単位時間あたり L 回の等間隔で行う状況を考える. これは刻み幅が $1/L$ の離散時間を考えることに相当する. そのとき, $T = N/L$ は α 回イベントが起こるまでにかかる時間を意味する.

(3) 単位時間内に起こるイベントの回数の期待値 $Lp = 1/\theta$ を固定したままで, $L \to \infty$ とすることは, イベントが起こる間隔の期待値を θ に固定して連続時間極限を取ることだと考えられる. この極限によって, 負の二項分布における $T = N/L$ の分布はガンマ分布に収束する.

(4) **ガンマ分布は, 連続時間の場合に, イベント起こる間隔の期待値が θ であるときに, α 回のイベントが起こるまでにかかる時間の分布になっている.**

5.6 Poisson分布の累積分布函数とガンマ分布の累積分布函数の関係

Poisson分布は単位時間内に生じるイベントの回数の分布だと解釈でき, ガンマ分布はイベントがα回おこるまでにかかる時間の分布だと解釈できるのであった. これらの解釈から, Poisson分布における確率をガンマ分布における確率と解釈したり, その逆向きの解釈をしたりできることがわかる.

Poisson分布 $\mathrm{Poisson}(\lambda)$ は単位時間のあいだに平均して λ 回起こるイベントが単位時間内で起こる回数の分布だと解釈できるのであった. その状況のもとでは, より一般に時間 t のあいだにイベントが起こる回数 K の分布は $\mathrm{Poisson}(\lambda t)$ になる.

ガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ は, イベントが起こる時間間隔の期待値が θ であるときに, α 回イベントが起こるまでにかかる時間 T の分布だと解釈できるのであった.

イベントが起こる時間間隔の期待値が θ であることは, 単位時間のあいだに平均して $1/\theta$ 回のイベントが起こることと同値である. そこで, 以下では次のように仮定する:

$$\lambda = 1/\theta \quad \theta = 1/\lambda.$$

「時間 τ 以内にイベントが起こる回数が κ 回未満であること」と「イベントが κ 回起こるまでにかかる時間が τ より大きいこと」は論理的に同値なので, それらの確率も等しくなるはずである: $\tau > 0$, $\kappa = 1, 2, \ldots$ のとき,

$$K \sim \mathrm{Poisson}(\lambda\tau), \;\; T \sim \mathrm{Gamma}(\alpha = \kappa, \theta = 1/\lambda), \quad \Longrightarrow \;\; P(K < \kappa) = P(T > \tau).$$

実際にこれが成立している. それを式で書くと,

$$\sum_{k=0}^{\kappa-1} \frac{e^{-\lambda\tau}(\lambda\tau)^k}{k!} = \frac{\int_\tau^\infty e^{-\lambda t} t^{\kappa-1}\,dt}{\Gamma(\kappa)/\lambda^\kappa}.$$

注意: この結果より, Poisson分布における累積分布函数の大量の和が必要な計算をガンマ分布における累積分布函数の計算に帰着できることがわかる. ガンマ分布の累積分布函数は基本特殊函数ライブラリに含まれている正則化された(不完全)ガンマ函数を使って効率的に計算できる.

5.7 問題: Poisson分布の累積分布函数とガンマ分布の累積分布函数の関係の直接証明

次の公式を証明せよ:

$$\sum_{k=0}^{\kappa-1} \frac{e^{-\lambda\tau}(\lambda\tau)^k}{k!} = \frac{\int_\tau^\infty e^{-\lambda t} t^{\kappa-1}\,dt}{\Gamma(\kappa)/\lambda^\kappa} \quad (\lambda, \tau > 0,\ \kappa = 1, 2, \ldots)$$

解答例: 左辺を $f(\tau)$ とおく. そのとき,

$$f(t) = \sum_{k=0}^{\kappa-1} \frac{e^{-\lambda t}\lambda^k t^k}{k!}.$$

$t \searrow 0$ のとき $f(t) \to 0$ となり,

$$\begin{aligned} f'(t) &= \sum_{k\le\kappa-1} \frac{e^{-\lambda t}\lambda^k k t^{k-1}}{k!} - \sum_{k\le\kappa-1} \frac{e^{-\lambda t}\lambda\lambda^k t^k}{k!} = \sum_{k\le\kappa-1} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1)!} - \sum_{k\le\kappa-1} \frac{e^{-\lambda t}\lambda^{k+1} t^k}{k!} \\ &= \sum_{k\le\kappa-1} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1)!} - \sum_{k\le\kappa} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1)!} = \frac{e^{-\lambda t}\lambda^\kappa t^{\kappa-1}}{(\kappa-1)!} = \frac{e^{-\lambda t} t^{\kappa-1}}{\Gamma(\kappa)/\lambda^\kappa}. \end{aligned}$$

ゆえに

$$f(\tau) = \frac{\int_\tau^\infty e^{-\lambda t} t^{\kappa-1}\,dt}{\Gamma(\kappa)/\lambda^\kappa}.$$

解答終

上のような煩雑な計算をやりたくない人も楽々遂行できる人も以下のように結果が本当に正しいかをコンピュータで計算して確認した方がよい.

In [39]:
```
function plot_cdfpoigam(λ, τ)
    poi = Poisson(λ*τ)
    gam(κ) = Gamma(κ, 1/λ)
    κ = range(1, round(mean(poi) + 5*std(poi))+0.5, 1000)
    plot(κ, κ -> cdf(poi, κ-1); label="cdf(Poisson(λτ),κ-1)")
    plot!(κ, κ -> ccdf(gam(floor(κ)), τ); ls=:dash, label="ccdf(Gamma(⌊κ⌋),1/λ),τ)")
    plot!(; legend=:bottomright, xlabel="κ")
    title!("λ=$λ, τ=$τ")
end
```

Out[39]: plot_cdfpoigam (generic function with 1 method)

In [40]:
```
plot(plot_cdfpoigam(3, 1), plot_cdfpoigam(3, 5); size=(800, 250), bottommargin=4Plots.mm)
```

Out[40]:

5.8 必ず解いて欲しい問題: χ²分布における確率が95%または99%になる範囲

$0 < \alpha < 1, \nu > 0$ であるとし, $Y \sim \mathrm{Chisq}(\nu)$ と仮定する. このとき, Y が c 以下になる確率

$$P(Y \le c)$$

が $1-\alpha$ になるような c を **正則化された(不完全)ガンマ函数** (regularized (incomplete) gamma function)

$$y = P(x|\alpha) = P(\alpha, x) = \frac{\int_0^x e^{-u} u^{\alpha-1}\,du}{\Gamma(\alpha)} \quad (x > 0)$$

の逆函数 $x = P^{-1}(y|\alpha)$ を使って表せ. (注意: $P(\alpha, x)$ は広く使われている標準的な記号法だが, $P(x|\alpha)$ と $P^{-1}(y|\alpha)$ はここだけの記号法である.)

さらに, $\nu = 1$ と $\nu = 2$ と $\nu = 3$ の場合に $P(Y \le c)$ が 95% になる c と 99% になる c を小数点以下第2桁目まで求めよ.

お願い: 後半の「さらに」以降の具体的な数値を求める部分は必ず解けるようになっておいて欲しい. コンピュータを使ってよい. (より正確に言えばコンピュータを使った計算の仕方を1つ以上マスターしておくこと!)

注意: 正規分布モデルの統計学でχ²分布に関する計算が不完全ガンマ函数に帰着できるという事実から, 正規分布モデルの統計学においてガンマ分布が必須であることがわかる.

解答例: $F(y) = P(Y \le y)$ とおく. $p = F(y)$ の逆函数 $y = Q(p) = F^{-1}(p)$ を正則化された不完全ガンマ函数を使って表そう:

$$\begin{aligned} F(y) = P(Y \le y) &= \frac{1}{2^{\nu/2}\Gamma(\nu/2)} \int_0^y e^{-x/2} x^{\nu/2-1}\,dx \\ &= \frac{1}{\Gamma(\nu/2)} \int_0^{y/2} e^{-u} u^{\nu/2-1}\,du = P(y/2|\nu/2). \end{aligned}$$

2つめの等号で $x = 2u$ とおいた. これより, $p = F(y)$ の逆函数(分位点函数) $y = Q(p) = F^{-1}(p)$ が次のように表されることがわかる:

$$y = Q(p) = 2P^{-1}(p|\nu/2).$$

したがって, $P(Y \le c) = 1-\alpha$ となる c は

$$c = 2P^{-1}(1-\alpha|\nu/2)$$

と表される.

$\nu = 1$ のとき,

- $P(Y \leq c) = 95\%$ となる c は $c \approx 3.84$
- $P(Y \leq c) = 99\%$ となる c は $c \approx 6.63$

$\nu = 3$ のとき,

- $P(Y \leq c) = 95\%$ となる c は $c \approx 5.99$
- $P(Y \leq c) = 99\%$ となる c は $c \approx 9.21$

$\nu = 3$ のとき,

- $P(Y \leq c) = 95\%$ となる c は $c \approx 7.81$
- $P(Y \leq c) = 99\%$ となる c は $c \approx 11.34$

解答終

In [41]:

```
quantile_chisq(ν, p) = 2gamma_inc_inv(ν/2, p, 1-p)

for ν in (1, 2, 3)
    @show ν
    @show quantile_chisq(ν, 0.95)
    @show quantile_chisq(ν, 0.99)
    println()
end
```

```
ν = 1
quantile_chisq(ν, 0.95) = 3.8414588206941245
quantile_chisq(ν, 0.99) = 6.634896601021214

ν = 2
quantile_chisq(ν, 0.95) = 5.99146454710798
quantile_chisq(ν, 0.99) = 9.21034037197618

ν = 3
quantile_chisq(ν, 0.95) = 7.814727903251177
quantile_chisq(ν, 0.99) = 11.34486673014437
```

In [42]:

```
for ν in (1, 2, 3)
    @show ν
    @show quantile(Chisq(ν), 0.95)
    @show quantile(Chisq(ν), 0.99)
    println()
end
```

```
ν = 1
quantile(Chisq(ν), 0.95) = 3.8414588206941245
quantile(Chisq(ν), 0.99) = 6.634896601021214

ν = 2
quantile(Chisq(ν), 0.95) = 5.99146454710798
quantile(Chisq(ν), 0.99) = 9.21034037197618

ν = 3
quantile(Chisq(ν), 0.95) = 7.814727903251177
quantile(Chisq(ν), 0.99) = 11.34486673014437
```

例えば $\nu = 2$ の場合について, WolframAlphaでは以下のように計算できる:

- 2 InverseGammaRegularized(n/2, 1 - 0.95) where n = 2 (https://www.wolframalpha.com/input?i=2+InverseGammaRegularized%28n%2F2%2C+1+-+0.95%29+where+n+%3D+2)
- 2 InverseGammaRegularized(n/2, 1 - 0.99) where n = 2 (https://www.wolframalpha.com/input?i=2+InverseGammaRegularized%28n%2F2%2C+1+-+0.99%29+where+n+%3D+2)
- quantile(ChisqDistribution(2), 0.95) (https://www.wolframalpha.com/input?i=quantile%28ChisqDistribution%282%29%2C+0.95%29)
- quantile(ChisqDistribution(2), 0.99) (https://www.wolframalpha.com/input?i=quantile%28ChisqDistribution%282%29%2C+0.99%29)

注意: このノートでは分位点函数を $Q(p)$ と書いていて紛らわしいのだが, 不完全ガンマ函数として

$$Q(x|\alpha) = Q(\alpha, x) = 1 - P(\alpha, x) = \frac{\int_x^\infty e^{-u} u^{\alpha-1}}{\Gamma(\alpha)}$$

の側が使われることもよくある. $p = Q(x|\alpha)$ の逆函数を $Q^{-1}(p|\alpha)$ と書くとき, $P(Y \leq y) = 2Q^{-1}(1-p|\alpha)$ となることに注意せよ. 上のWolframAlphaでは実際にそうなっている.

5.9 問題: 自由度1のχ²分布と標準正規分布の関係の数値例

$Z \sim \mathrm{Normal}(0,1)$, $Y \sim \mathrm{Chisq}(1)$ のとき,

$$P(-c \leq Z \leq c) = 0.95, \quad P(Y \leq d) = 0.95$$

を満たす c, d の値はそれぞれ

$$c \approx 1.96, \quad d \approx 3.84$$

であった. このとき

$$c^2 \approx 1.96^2 = 3.8416 \approx 3.84 \approx d$$

となる. これが必然である理由を説明せよ.

In [43]:
```
(196//100)^2 ▷ float
```
Out[43]: 3.8416

解答例1: 実際には正確に $c^2 = d$ となっている. なぜならば, $z > 0$ を $z = \sqrt{y}$ とおくと,

$$\begin{aligned} P(-c \leq Z \leq c) &= \int_{-c}^{c} \frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz = 2\int_0^c \frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz \\ &= 2\int_0^{c^2} \frac{e^{-y/2}}{\sqrt{2\pi}}\frac{1}{2}y^{-1/2}\,dy = \int_0^{c^2} \frac{e^{-y/2}y^{1/2-1}}{2^{1/2}\Gamma(1/2)}\,dy = \\ &= P(Y \leq c^2). \end{aligned}$$

つまり $P(-c \leq Z \leq c) = P(y \leq c^2)$ なので, $P(-c \leq Z \leq c) = 1-\alpha$ となる c の二乗と $P(Y \leq d) = 1-\alpha$ となる d は一致する.

解答終

解答例2: $Z^2 \sim \mathrm{Chisq}(1)$ となるのであった. ゆえに

$$P(-c \leq Z \leq c) = P(Z^2 \leq c^2) = P(Y \leq c^2).$$

これより, $P(-c \leq Z \leq c) = 1-\alpha$ となる c の二乗と $P(Y \leq d) = 1-\alpha$ となる d は一致することがわかる.

解答終

6 ベータ分布の一様乱数生成の繰り返しによる解釈

$\alpha, \beta > 0$ と仮定する. ベータ分布は 0 から 1 のあいだに分布する連続確率分布でその確率密度函数は

$$p(t|\alpha,\beta) = \frac{t^{\alpha-1}(1-t)^{\beta-1}}{B(\alpha,\beta)} \quad (0 < t < 1)$$

と書けるのであった. 別のノートで我々はこの確率密度函数を天下り的に与えて, この分布のある種の極限でガンマ分布が得られることや, ガンマ分布からベータ分布を作れることなどを示した. ガンマ分布は正規分布がらみの確率分布なので, この事実は, ベータ分布が正規分布モデルの統計学で大活躍する確率分布であることを示唆し, 実際にそうなっている.

以下ではこれとは別のベータ分布の解釈を紹介する. それによって, ベータ分布を二項分布や負の二項分布と繋げることに成功する. ベータ分布は二項分布モデルの統計学や負の二項分布モデルの統計学でも役に立つ.

6.1 一様分布のサイズnの標本分布の順序統計量

独立な確率変数達 $T_1, T_2, \ldots, T_n$ のそれぞれは一様分布 $\mathrm{Uniform}(0,1)$ に従っていると仮定する. $T_1, \ldots, T_n$ の中で下から k 番目に小さなものを $T_{(k)}$ と書く.

$T_{(k)}$ も確率変数になることに注意せよ. $T_{(k)}$ に対応する乱数は次の手続きで作れる:

(1) $\mathrm{rand}()$ を n 回実行してその結果を $T_1, \ldots, T_n$ とする. (2) $T_1, \ldots, T_n$ の中で下から k 番目に小さな値を $T_{(k)}$ とする.

これによって 0 から 1 のあいだの数値 $T_{(k)}$ で以上の手続きによってランダムに与えられるものが得られる.

$T_{(k)}$ を **一様分布のサイズ n の標本分布の順序統計量** と呼ぶ.

6.2 一様分布のサイズnの標本分布の順序統計量が従う分布

一様分布のサイズ n の標本分布の順序統計量 $T_{(k)}$ が従う分布はベータ分布 $\mathrm{Beta}(k, n-k+1)$ になる:

$$T_{(k)} \sim \mathrm{Beta}(k-1, n-k).$$

このようにパラメータが正の整数のすべてのベータ分布は一様分布の標本分布の順序統計量の分布として得られる.

証明1: $0 < t < t + dt < 1$ と $1, 2, \ldots, n$ を並び替えたもの $i_1, i_2, \ldots, i_n$ について,

$$P(T_{i_1}, \ldots, T_{i_{k-1}} < t < T_{i_k} < t + dt < T_{i_{k+1}}, \ldots, T_{i_n}) = t^{k-1}\, dt\, (1 - (t + dt))^{n-k}.$$

さらに, $1, 2, \ldots, n$ を $k-1$ 個の $i_1, \ldots, i_{k-1}$ と1個の i_k と $n-k$ 個の $i_{k+1}, \ldots, i_n$ の3グループへの分割の仕方の個数は

$$\frac{n!}{(k-1)!1!(n-k)!} = \frac{\Gamma(n+1)}{\Gamma(k)\Gamma(n-k+1)} = \frac{1}{B(k, n-k+1)}$$

と書ける. これと上の確率をかけて, $dt \to 0$ のときの dt より高次の微小量を無視したものが $T_{(k)}$ が従う分布の密度函数 $\times dt$ になる. (順序統計量 $T_{(k)}$ は $T_1, \ldots, T_n$ の中で k 番目に小さなものであった.) ゆえに, $T_{(k)}$ が従う分布の確率密度函数は

$$p(t|n, k) = \frac{t^{k-1}(1-t)^{n-k}}{B(k, n-k+1)} \quad (0 < t < 1)$$

になる. これはベータ分布 $\mathrm{Beta}(k, n-k+1)$ の確率密度函数である. これで $T_{(k)}$ が従う分布はベータ分布 $\mathrm{Beta}(k, n-k+1)$ になることがわかった.

証明終

証明2: 一様分布 $\mathrm{Uniform}(0, 1)$ の累積分布函数と確率密度函数をそれぞれ $F(t) = t$ $(0 < t < 1)$, $p(t) = F'(t) = 1$ $(0 < t < 1)$ と書き, 一様分布のサイズ n の標本分布の順序統計量 $T_{(k)}$ の累積分布函数を $G(t)$ と書く:

$$G(t) = P(T_{(k)} \leq t).$$

$T_{(k)} \leq t$ が成立することは, $T_1, \ldots, T_n$ の中の k 個以上が t 以下になることと同値なので,

$$G(t) = \sum_{j=k}^{n} \binom{n}{j} F(t)^j (1 - F(t))^{n-j}.$$

これの導函数が $T_{(k)}$ の確率密度函数になる:

$$\begin{aligned}
G'(t) &= \sum_{j \geq k} j \binom{n}{j} F(t)^{j-1} F'(t)(1 - F(t))^{n-j} - \sum_{j \geq k} (n-j) \binom{n}{j} F(t)^j F'(t)(1 - F(t))^{n-j-1} \\
&= \sum_{j \geq k} n \binom{n-1}{j-1} F(t)^{j-1} F'(t)(1 - F(t))^{n-j} - \sum_{j \geq k} n \binom{n-1}{j} F(t)^j F'(t)(1 - F(t))^{n-j-1} \\
&= \sum_{j \geq k} n \binom{n-1}{j-1} F(t)^{j-1} F'(t)(1 - F(t))^{n-j} - \sum_{j \geq k+1} n \binom{n-1}{j-1} F(t)^j F'(t)(1 - F(t))^{n-j} \\
&= n \binom{n-1}{j-1} F(t)^{k-1} F'(t)(1 - F(t))^{n-k} \\
&= \frac{n!}{(k-1)!(n-k)!} F(t)^{k-1} F'(t)(1 - F(t))^{n-k} \\
&= \frac{1}{B(k, n-k+1)} F(t)^{k-1} F'(t)(1 - F(t))^{n-k} \\
&= \frac{1}{B(k, n-k+1)} t^{k-1}(1-t)^{n-k}.
\end{aligned}$$

これはベータ分布 $\mathrm{Beta}(k, n-k+1)$ の確率密度函数である. これで $T_{(k)}$ が従う分布はベータ分布 $\mathrm{Beta}(k, n-k+1)$ になることがわかった.

証明終

注意: 上の $G'(t)$ の計算は最後の等号を除けば任意の函数 $F(t)$ について成立している. そのことを使えば一様分布に限らない場合にも順序統計量の確率密度函数を計算できる.

6.3 一様乱数生成の繰り返しからベータ分布が出て来ることの数値的確認

実際にコンピュータで n 個の一様乱数の組を大量に発生させて, 各々の中で k 番目に小さな値の分布がベータ分布になることを確認してみよう.

In [44]:

```
function sim_uniformbeta(n; L = 10^6)
    T = Matrix{Float64}(undef, n, L) # 順序統計量の乱数を保存する2次元行列
    tmp = [Vector{Float64}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for j in 1:L
        t = rand!(tmp[Threads.threadid()]) # 一様乱数をn個生成
        T[:, j] .= sort!(t) # 小さい順序にソートして順序統計量の乱数を作る
    end
    T
end

function plot_sim_uniformbeta(T::Matrix; ran = axes(T, 1), L = 10^6, kwargs...)
    n = size(T, 1)
    PP = []
    for k in ran
        P = stephist(T[k,:]; norm=true, alpha=true, bin=[0; 0.01:0.02:0.99; 1], label="")
        plot!(; title="n=$n, k=$k", xtick=0:0.1:1)
        plot!(Beta(k, n-k+1), -0.01, 1.01; ls=:dash, label="")
        push!(PP, P)
    end
    plot(PP...; kwargs...)
end

function plot_sim_uniformbeta(n::Integer; ran = 1:n, L = 10^6, kwargs...)
    T = sim_uniformbeta(n; L)
    plot_sim_uniformbeta(T; ran, L, kwargs...)
end
```

Out[44]: plot_sim_uniformbeta (generic function with 2 methods)

In [45]:

```
plot_sim_uniformbeta(9; size=(800, 500), layout=(3, 3), tickfontsize=6)
```

Out[45]:

乱数の大量発生によるシミュレーションの結果のステップヒストグラムと理論的な分布 $\mathrm{Beta}(k, n-k+1)$ の確率密度函数をプロットした破線がぴったり一致している.

6.4 問題: 0からθのあいだの一様分布の上限θの不偏推定量

$a < b$ と仮定する. a から b のあいだの **一様分布**

$$\mathrm{Uniform}(a, b)$$

が確率密度函数

$$p(x|a,b) = \begin{cases} 1/(b-a) & (a < x < b) \\ 0 & (\text{otherwise}) \end{cases}$$

によって定義される. $a < x < b$ における不等号は $\leq$ でもよい.

$\theta > 0$ であるとし, 独立な確率変数達 $X_1, \ldots, X_n$ の各々は 0 から θ のあいだの一様分布に従っていると仮定する. $X_1, \ldots, X_n$ の中で k 番目に小さなものを $X_{(k)}$ と書く. 例えば $X_{(n)} = \max(X_1, \ldots, X_n)$ となる. 定数 c_k で次を満たすものを求めよ:

$$E[c_k X_{(k)}] = \theta.$$

このとき $c_k X_{(k)}$ はパラメータ θ の **不偏推定量** (unbiases estimator)であるという.

解答例: $T_i = X_i/\theta$, $T_{(k)} = X_{(k)}/\theta$ とおくと, $T_i \sim \mathrm{Uniform}(0,1)$ なので, $T_{(k)} \sim \mathrm{Beta}(k, n-k+1)$ となる. ゆえに, $X_{(k)} = \theta T_{(k)}$ と $\mathrm{Beta}(k, n-k+1)$ の平均が $k/(n+1)$ であることより,

$$E[X_{(k)}] = \theta E[T_{(k)}] = \frac{k}{n+1}\theta.$$

したがって,

$$c_k = \frac{n+1}{k}$$

とおくと, $E[c_k X_{(k)}] = \theta$ が成立する.

解答終

注意: 例えば,

$$E\left[\frac{n+1}{n}\max(X_1, \ldots, X_n)\right] = \theta.$$

$X_1, \ldots, X_n$ の最大値を $(n+1)/n$ 倍して少し大きくすると, 一様分布 $\mathrm{Uniform}(0, \theta)$ の上限 θ の不偏推定量が得られる.

In [46]:

```
θ = 10
dist = Uniform(0, θ)
n = 10
tmp = zeros(n)
θ̂ = [(n+1)/n*maximum(rand!(dist, tmp)) for _ in 1:10^6]
@show mean(θ̂)
@show std(θ̂)
@show std(n/(n+1)*θ̂; mean=θ)

histogram(θ̂; norm=true, alpha=0.3, bin=100,
    label="unbiased estimator")
vline!([θ]; label="true θ")
plot!(; legend=:topleft, xtick=0:20)
```

```
mean(θ̂) = 9.99945302861342
std(θ̂) = 0.9123861475572801
std((n / (n + 1)) * θ̂; mean = θ) = 1.2309855372059995
```

Out[46]:

In [47]:
```
[
    [mean((n+1)/k * Beta(k, n-k+1)) for k in 1:n];;
    [std((n+1)/k * Beta(k, n-k+1)) for k in 1:n];;
]
```

Out[47]:
```
10×2 Matrix{Float64}:
 1.0  0.912871
 1.0  0.612372
 1.0  0.471405
 1.0  0.381881
 1.0  0.316228
 1.0  0.263523
 1.0  0.218218
 1.0  0.176777
 1.0  0.136083
 1.0  0.0912871
```

6.5 二項分布の累積分布函数のベータ分布の累積分布函数表示

$0 < p < 1$ のとき, 前節の証明2より,

$$P(T_{(k)} \leq p) = \sum_{j=k}^{n} \binom{n}{j} p^j (1-p)^{n-j}.$$

この右辺は二項分布 $\mathrm{Binomial}(n, p)$ において k 以上になる確率に一致している.

さらに, $T_{(k)}$ が従う分布が $\mathrm{Beta}(k, n-k+1)$ だったので, その確率はベータ分布 $\mathrm{Beta}(k, n-k+1)$ において p 以下になる確率に一致することになる.

これで, 二項分布とベータ分布の累積分布函数の関係が得られた:

$$\begin{aligned} &K \sim \mathrm{Binomial}(n, p), \quad T \sim \mathrm{Beta}(k, n-k+1) \\ &\implies P(K \geq k) = P(T \leq p), \quad P(K \leq k-1) = 1 - P(T \leq p). \end{aligned}$$

k を $k+1$ に置き換えると,

$$\begin{aligned} &K \sim \mathrm{Binomial}(n, p), \quad T \sim \mathrm{Beta}(k+1, n-k) \\ &\implies P(K \geq k+1) = P(T \leq p), \quad P(K \leq k) = 1 - P(T \leq p). \end{aligned}$$

これらの公式の利点は右辺のベータ分布側の確率の計算を基本特殊函数ライブラリによって効率よく可能なことである. 左辺の二項分布における確率は n が大きいときには平均して $n/2$ 個の数値を足し上げる計算が必要になってしまう.

注意: 前節の証明1の結果($T_{(k)}$ の確率密度函数の形)を認めれば, 証明2における $G'(t)$ に関する少し面倒な計算を省略して,

$$P(T_{(k)} \leq p) = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)}$$

が得られるので, 平易に得られる

$$P(T_{(k)} \leq p) = \sum_{j=k}^{n} \binom{n}{j} p^j (1-p)^{n-j}$$

と合わせて,

$$\sum_{j=k}^{n} \binom{n}{j} p^j (1-p)^{n-j} = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)}$$

が得られる. このことからも上の方で述べた二項分布の累積分布函数がベータ分布の累積分布函数で書けるという結果が得られる.

注意: 以上の結果はBernoulli試行を一様乱数の生成の繰り返しに持ち上げることによって得られたと考えられる. 0 と 1 の間の一様乱数の値が p 以下のときに値 1 を生成し, それ以外のときに 0 を生成するようにすれば, 一様乱数の生成の繰り返しからBernoulli試行が得られる.

そのときに, n 回分の一様乱数の生成結果の中で k 番目に小さな乱数のなす分布がベータ分布になっていることから, 二項分布とベータ分布の関係が自然に導かれた.

In [48]:
```
function plot_cdfbinbeta(n, p)
    bin = Binomial(n, p)
    μ, σ = mean(bin), std(bin)
    beta(k) = Beta(k, n-k+1)
    k = range(max(1, μ-4σ), min(n, μ+4σ), 1000)
    plot(k, k -> ccdf(bin, k-1); label="ccdf(Binomial(n,p),k-1)")
    plot!(k, k -> cdf(beta(floor(k)), p); ls=:dash, label="cdf(Beta(⌊k⌋,n-⌊k⌋+1)),p)")
    plot!(; legend=:topright, xlabel="k")
    title!("n=$n, p=$p")
end
```

Out[48]: plot_cdfbinbeta (generic function with 1 method)

In [49]:
```
plot(plot_cdfbinbeta(20, 0.2), plot_cdfbinbeta(20, 0.4); size=(800, 250), bottommargin=4Plots.
```

Out[49]:

6.6 二項分布とベータ分布の関係のClopper-PearsonのP値函数への応用

二項分布モデルに基く比率 p (成功確率パラメータと呼んで来た)の **Clopper-Pearsonの信頼区間** を与えるP値函数(以下 **Clopper-PearsonのP値函数** と呼ぶ)の効率的な計算に, 前節で述べた二項分布とベータ分布の関係が実際に使われている.

Clopper-PearsonのP値函数は二項分布モデル $\mathrm{Binomial}(n,p)$ と「n 回中 k 回成功した」というデータの整合性を測る函数である. Clopper-PearsonのP値函数は以下のように定義される.

二項分布 $\mathrm{Binomial}(n,p)$ の累積分布函数を, 確率変数 $K \sim \mathrm{Binomial}(n,p)$ を使って

$$F(k) = P(K \le k) = \sum_{j=0}^{k} \binom{n}{j} p^j (1-p)^{n-j}$$

と定義する. Clopper-PearsonのP値函数 $\mathrm{pvalue}_{\mathrm{CP}}(k|n,p)$ を次のように定める:

$$\mathrm{pvalue}_{\mathrm{CP}}(k|n,p) = \min(1, 2F(k), 2(1-F(k-1)) = \min(1, 2P(K \le k), 2P(K \ge k)).$$

$\min(P(K \le k), P(K \ge k))$ は二項分布におけるデータの値 k 以上に端側に偏る確率を意味し, その2倍がClopper-PearsonのP値である. P値が小さいほど, 二項分布モデル $\mathrm{Binomial}(n,p)$ と「n 回中 k 回成功した」というデータの整合性が低いと考える.

二項分布の累積分布函数はベータ分布の累積分布函数で書けるのであった. 確率変数 $T_k \sim \mathrm{Beta}(k, n-k+1)$ を使えば次が成立している:

$$F(k) = P(K \le k) = 1 - P(T_{k+1} \le p), \quad 1 - F(k-1) = P(K \ge k) = P(T_k \le p).$$

そして, 右辺の計算で必要な $P(T_k \le p)$ は次のように書ける:

$$P(T_k \le p) = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1}.$$

これの右辺は, 基本特殊函数の1つである正則化された不完全ベータ函数になっており, コンピュータで効率的に計算できる.

注意: $0 < \alpha < 1$ のとき, 「n 回中 k 回成功した」というデータが与える信頼度 $1-\alpha$ のClopper-Pearsonの信頼区間は $\mathrm{pvalue}_{\mathrm{CP}}(k|n,p) \ge \alpha$ を満たすパラメータ p の範囲と定義される:

$$\mathrm{CI}_{\mathrm{CP}}(k|n) = \{\, p \in [0,1] \mid \mathrm{pvalue}_{\mathrm{CP}}(k|n,p) \ge \alpha \,\}.$$

この辺りの話は後で詳しく説明することになる.

6.7 おまけ: 二項分布とベータ分布の関係のBayes統計への応用

この一連のノートではBayes統計についての詳しい説明はしないつもりでいるのだが, 以下では **用語の詳しい定義を一切せずに** 二項分布とベータ分布の関係のBayes統計について説明してみたい. (以下の部分は理解できなくてもよい.)

二項分布モデルで事前分布として共役事前分布 $\mathrm{Beta}(a, b)$ を採用したとき, 「n 回中 k 回成功した」(例: 「n 任中 k 人の病気が治った」, 「n 人中 k 人が商品を購入してくれた」「n 個中 k 個が不良品だった」)というデータが得られたときの成功確率 p に関する事後分布は $\mathrm{Beta}(a + k, b + (n - k))$ になる.

特に形式的にimproper共役事前分布 $\mathrm{Beta}(0, 1)$ や $\mathrm{Beta}(1, 0)$ を採用したときの事後分布は前節や前々節で出て来た $\mathrm{Beta}(k, n - k + 1)$ や $\mathrm{Beta}(k + 1, n - k)$ になる.

そして, 二項分布の累積分布函数とベータ分布の累積分布函数の関係は以下のようになっていた:

$$K \sim \mathrm{Binomial}(n, p_0)$$

のとき,

$$\begin{aligned} T \sim \mathrm{Beta}(k, n - k + 1) &\implies P(K \geq k) = P(T \leq p_0), \\ T \sim \mathrm{Beta}(k + 1, n - k) &\implies P(K \leq k) = P(T \geq p_0). \end{aligned}$$

これは, 以下を意味している:

- 事前分布 $\mathrm{Beta}(0, 1)$ に関する事後分布において「$p \leq p_0$」という仮説が成立する確率は二項分布モデルにおける「$p \leq p_0$」という仮説の片側検定のP値にぴったり等しい.
- 事前分布 $\mathrm{Beta}(1, 0)$ に関する事後分布において「$p \geq p_0$」という仮説が成立する確率は二項分布モデルにおける「$p \geq p_0$」という仮説の片側検定のP値にぴったり等しい.

このように, 特定のケースではP値とBayes統計の事後分布において仮説が成立する確率がぴったり等しくなる. 事前分布を例えば平坦事前分布 $\mathrm{Beta}(1, 1)$ に変えても, $(0, 1)$ や $(1, 0)$ と $(1, 1)$ の違いは小さいので, n が数十程度で差は小さくなる. 「$p \leq p_0$」や「$p \geq p_0$」のような仮説の片側検定については, 通常のP値を使う統計学と事前事後分布を使うBayes統計学は数学的にはほぼ同じものだと思ってよい. (注意: ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ は α, β がともに正の場合にのみ定義されているので, $\mathrm{Beta}(0, 1)$ や $\mathrm{Beta}(1, 0)$ は確率分布としては定義されていない. しかし, Bayes法はその場合にも有効になっている. $\mathrm{Beta}(0, 1)$ や $\mathrm{Beta}(1, 0)$ はimproper事前分布と呼ばれている.)

この事実から, **「Bayes統計は通常の統計学とは異なる主義思想哲学に基く. そしてP値の使用には害がある. P値を使う統計学は捨ててBayes統計を使うべきである」のような主張をすることによって, Bayes統計を宣伝している人達が完全に間違っている** ことがわかる.

数学的には同じデータから常に同じ結果が出ることがわかっているのに, 主義思想哲学によって結果が変わると主張することは論理的に誤りである. 事前事後分布ではなくP値を使った場合にも, 数学的に同等だという理由で事前事後分布の場合と同じような解釈をP値についても許すならば論理的に整合性は取れているが, 「Bayes統計は通常の統計学とは異なる主義思想哲学に基く」のような主張をする人達はそういう論理的に整合性のある議論を決してしない.

以上の結果から, 「怪しげな事前分布を使うのでBayes統計は信用でいない」という主張も完全に間違っていることもわかる. 通常のP値を使った統計学に価値があるならば, 事前分布を適当に選べばそれと完全に同じことをできるBayes統計にも価値があるということは明らかである. むしろ, 事前分布という数学的選択肢が増えたおかげで, それなしにはできなかったこともできるようになると予想され, その予想は実際に正しい.

各種の技術には長所と短所の両方があることが普通である. 方法Aを方法Bに切り替える場合には, それで失われる方法A側の相対的長所とそれによって得られる方法Bの側の相対的長所のどちらを選ぶかになることが多い. 所謂「トレードオフ」の問題になる. P値を使う方法を使うか, Bayes統計の方法を使うかの問題も典型的なトレードオフの問題に過ぎない. そこに「主義思想哲学」を持ち込む必要はない.

6.7.1 平坦事前分布とimproper事前分布Beta(0,1)の事後分布の比較

上で述べたように次が成立しているのであった.

- 事前分布 $\mathrm{Beta}(0, 1)$ に関する事後分布において「$p \leq p_0$」という仮説が成立する確率は二項分布モデルにおける「$p \leq p_0$」という仮説の片側検定のP値にぴったり等しい.

次の節の「おまけ関連問題」で扱うように, 事前分布 $\mathrm{Beta}(\alpha, \beta)$ のもとでデータ「n 回中 k 回成功」が得られたときの事後分布は $\mathrm{Beta}(\alpha + k, \beta + n - k)$ になる. 以下では事前分布が $\mathrm{Beta}(0, 1)$ の場合の事後分布 $\mathrm{Beta}(k, n - k + 1)$ と平坦事前分布 $\mathrm{Beta}(1, 1)$ の事後分布 $\mathrm{Beta}(k + 1, n - k + 1)$ の確率密度函数を同時プロットして比較してみる. k と $n - k$ が大きな場合には差は小さい.

In [50]:
```
function plot_2betas(n, k; α=1, β=1, kwargs...)
    plot(Beta(k, n-k+1); label="Beta(k, n-k+1)")
    plot!(Beta(k+α, n-k+β); label="Beta(k+$α,n-k+$β)", ls=:dash)
    plot!(; xtick=0:0.1:1)
    title!("n=$n, k=$k"; kwargs...)
end
```

Out[50]: plot_2betas (generic function with 1 method)

In [51]:
```
plot(plot_2betas(20, 5), plot_2betas(40, 10), plot_2betas(80, 20);
    size=(800, 200), layout=(1, 3), tickfontsize=7)
```

Out[51]:

このように k, $n-k$ が大きくなると, 事後分布への事前分布のパラメータの違いの影響は小さくなって行く.

6.8 おまけ関連問題: ベータ分布は二項分布の共役事前分布である

現実から「n 回中 k 回成功」の型のデータが得られる状況を考える(「n 人中 k 人が病気になった」「n 人中 k 人が商品を購入した」など).

そのデータと成功確率 p の二項分布 $\mathrm{Binomial}(n,p)$ の整合性がどうなっているかを見たいとする. その確率質量函数を次のように書く:

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k}$$

Bayes統計では, 成功確率パラメータ p に関する確率密度函数 $\varphi(p)$ で与えられる確率分布(**事前分布** (prior)と呼ぶ)をさらに与えて, 統計モデル内部では次のようになっていると考える:

1. 最初に成功確率パラメータの値 p が事前分布に従ってランダムに決まる.
2. その p の値を採用した二項分布 $\mathrm{Binomial}(n,p)$ に従って k がランダムに決まる.

この統計モデル内における k と p の同時分布は $P(k|n,p)\varphi(p)$ によって与えられ, モデル内での k,p のすべての組み合わせに関する確率の総和が 1 になることは次のようにして確認される:

$$\int_0^1 \underbrace{\sum_{k=0}^{n} P(k|n,p)}_{=1} \varphi(p)\,dp = \int_0^1 \varphi(p)\,dp = 1.$$

現実のデータを取得における成功確率 p に対応する値はこのモデル内設定のように確率分布しておらず, 定数かもしれない. 例えば, S市の住人をランダムに選んだときのBMI(=体重/身長の2乗)の値が25以上である確率は, S市におけるBMIが25以上である人達の割合に一致し, 確率分布しているわけではない. そういう状況であっても, 統計モデル内では成功確率パラメータ p がランダムに決まっていると考えてもよい. (その理由を一般的な主義思想哲学に求めることはよく見る典型的な誤りである. Bayes統計に限らず一般的に, 統計モデルとして現実にぴったり一致する可能性があるものを選ぶ必要はない.)

「n 人中 k 人成功」の型のデータが得られたとき, データとモデルを比較するために, その統計モデル内でデータと同じ数値が生成されたという条件で制限された **条件付き確率分布** (条件付き確率については後で説明したい)を考える. 統計モデルと現実から得たデータを比較するために, 数学的フィクションであるモデル内で現実から得たデータと同じ数値がランダムに生成されたという状況を考えるのである. Bayes法とは異なる最尤法の予測分布もそのような条件付き確率分布になっているので, そのような条件付き確率分布を考えるか否かはBayes統計であるか否かとは無関係である.

そのモデル内条件付き確率分布におけるパラメータ p の分布は **事後分布** (posterior)と呼ばれる. 条件付き確率の定義より, 「n 回中 k 回成功」というデータに関する事後分布 $\varphi(p|n,k)$ の確率密度函数は次の形になる:

$$\varphi(p|n,k) = \frac{P(k|n,p)\varphi(p)}{\int_0^1 P(k|n,p)\varphi(p)\,dp}.$$

事後分布は統計モデルとデータの整合性を見るために使われる. 例えば, 事後分布に従う確率変数 P について $P \leq 0.3$ となる確率が小さいならば, その統計モデル+仮説「成功確率は 0.3 以下である」の組み合わせとデータとの整合性は低いと考える. 注意するべきことは「成功確率は 0.3 以下である」という仮説単体とデータの整合性を見ているのではなく, 統計モデル+仮説とデータの整合性しかわからないということである. 統計モデルを適切に与えないと仮説とデータの整合性について信頼できる結果は得られない.

問題: 以上の設定のもとで, 事前分布がベータ分布 $\mathrm{Beta}(a, b)$ で与えられたとき, 「n 回中 k 回成功」というデータに関する事後分布はベータ分布 $\mathrm{Beta}(a+k, b+n-k)$ になることを示せ.

解答例: 事前分布がベータ分布 $\mathrm{Beta}(a, b)$ で与えられているという設定は事前分布 $\varphi(p)$ が次のように書けているということを意味する:

$$\varphi(p) = \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)}.$$

このとき,

$$C = \binom{n}{k} \Big/ B(a,b)$$

とおくと,

$$\begin{aligned} P(k|n,p)\varphi(p) &= C\,p^k(1-p)^{n-k} \cdot p^{a-1}(1-p)^{b-1} = C\,p^{a+k-1}(1-p)^{b+n-k-1} \\ \int_0^1 P(k|n,p)\varphi(p)\,dp &= C\int_0^1 p^{a+k-1}(1-p)^{b+n-k-1}\,dp = C\,B(a+k, b+n-k). \end{aligned}$$

ゆえに, 事後分布 $\varphi(p|n,k)$ は次のようになる:

$$\varphi(p|n,k) = \frac{P(k|n,p)\varphi(p)}{\int_0^1 P(k|n,p)\varphi(p)\,dp} = \frac{C\,p^{a+k-1}(1-p)^{b+n-k-1}}{C\,B(a+k,b+n-k)} = \frac{p^{a+k-1}(1-p)^{b+n-k-1}}{B(a+k,b+n-k)}.$$

これは事後分布がベータ分布 $\mathrm{Beta}(a+k, b+n-k)$ になることを意味している.

解答終

In [52]:
```
function plot_betaposterior(a, b, n, k; kwargs...)
    beta = Beta(a+k, b+n-k)
    plot(beta, -0.01, 1.01; label="", xtick=0:0.1:1)
    title!("posterior: Beta(a+k, b+n-k)\n(a,b)=($a,$b), n=$n, k=$k")
    plot!(; kwargs...)
end
```

Out[52]:
```
plot_betaposterior (generic function with 1 method)
```

6.8.1 様々な事前分布の事後分布のプロット

事後分布 $\mathrm{Beta}(a+k, b+n-k)$ を色々プロットしてみよう.

様々な事前分布について, データが「n 回中3割成功」の場合の事後分布を $n = 0, 10, 100, 1000$ の場合にプロットしてみる. ベイズ法ではデータサイズ n が 0 の場合(データが取得される前の状況)も扱える.

以下は事前分布が一様分布 $\mathrm{Beta}(1,1) = \mathrm{Uniform}(0,1)$ の場合.

In [53]:
```
a, b = 1, 1
plot(plot_betaposterior(a, b, 0, 0),
    plot_betaposterior(a, b, 10, 3);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[53]:

In [54]:
```
a, b = 1, 1
plot(plot_betaposterior(a, b, 100, 30),
    plot_betaposterior(a, b, 1000, 300);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[54]:

以下は事前分布がJeffreys事前分布 $\mathrm{Beta}(0.5, 0.5)$ の場合だが, 上の一様事前分布の場合とほとんど同じになっている.

In [55]:
```
a, b = 0.5, 0.5
plot(plot_betaposterior(a, b, 0, 0; ylim=(-0.3, 3.3)),
    plot_betaposterior(a, b, 10, 3);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[55]:

In [56]:
```
a, b = 0.5, 0.5
plot(plot_betaposterior(a, b, 100, 30),
    plot_betaposterior(a, b, 1000, 300);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[56]:

以下では違いを出すために事前分布を $\mathrm{Beta}(18, 2)$ にしてみた.

この事前分布を採用するとデータがない状態では「p は 0.9 に近い」という仮説と(事前分布を含めての)統計モデルの整合性が高いということになる. しかし, データが「n 回中3割成功」の形をしている場合には n を大きくすると, 「p は 0.3 に近い」という仮説と統計モデル+データの整合性が高くなる.

In [57]:
```
a, b = 18, 2
plot(plot_betaposterior(a, b, 0, 0),
    plot_betaposterior(a, b, 10, 3);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[57]:

In [58]:
```
a, b = 18, 2
plot(plot_betaposterior(a, b, 100, 30),
    plot_betaposterior(a, b, 1000, 300);
    size=(800, 200), layout=(1, 2), topmargin=4Plots.mm)
```

Out[58]:

6.9 負の二項分布の累積分布函数のベータ分布の累積分布函数表示

独立な確率変数達 $T_1, T_2, T_3, \ldots$ のそれぞれは一様分布 $\mathrm{Uniform}(0,1)$ に従っていると仮定する. その最初の n 個 $T_1, \ldots, T_n$ の中で下から k 番目に小さなものをこの節では $T_{(n,k)}$ と書くことにする. $T_{(n,k)}$ はベータ分布に従うのであった:

$$T_{(n,k)} \sim \mathrm{Beta}(k, n-k+1).$$

$0 < p < 1$ と仮定する. このとき, $T_i \leq p$ ならば $X_i = 1$ とし, そうでないならば $X_i = 0$ とすることによって, 成功確率 p の Bernoulli分布に従う独立な確率変数達 $X_1, X_2, X_3, \ldots$ が得られる. これはBernoulli試行そのものである.

Bernoulli試行を 1 がちょうど k 回出るまで続けたときの試行回数 N (これは確率変数になる)が n 以下になる確率は負の二項分布の確率質量函数を使って次のように書ける:

$$P(N \leq n) = \sum_{j \leq n} \binom{j-1}{j-k} p^k (1-p)^{j-k}.$$

この確率は $T_1, \ldots, T_n$ の中に p 以下のものが k 個以上存在する確率に一致し, さらにその確率は $T_1, \ldots, T_n$ の中で k 番目に小さな値 $T_{(n,k)}$ が p 以下になる確率に一致する.

以上より以下が成立することがわかる:

$$\begin{aligned} &N - k \sim \mathrm{NegativeBinomial}(k, p), \quad T \sim \mathrm{Beta}(k, n-k+1) \\ &\implies P(N \leq n) = P(T \leq p), \quad P(N > n) = 1 - P(T \leq p). \end{aligned}$$

In [59]:
```julia
function plot_cdfnegbinbeta(α, p)
    negbin = NegativeBinomial(α, p)
    μ, σ = mean(negbin), std(negbin)
    beta(m) = Beta(α, m+1)
    m = range(max(0, μ-4σ), μ+4σ, 1000)
    plot(m, m -> cdf(negbin, m); label="cdf(NegBin(α,p),m)")
    plot!(m, m -> cdf(beta(floor(m)), p); ls=:dash, label="cdf(Beta(α, ⌊m⌋+1), p)")
    plot!(; legend=:bottomright, xlabel="m")
    title!("α=$α, p=$p")
end
```

Out[59]: plot_cdfnegbinbeta (generic function with 1 method)

In [60]:
```julia
plot(plot_cdfnegbinbeta(5, 0.7), plot_cdfnegbinbeta(5, 0.3);
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[60]:

6.10 負の二項分布とベータ分布の関係の非整数パラメータケース

前節の議論では k は正の整数でなければいけない. 以下では正の整数 k を正の実数 α に置き換えても同様の結果が成立することを前々節の証明2の方法で示そう.

$M \sim \mathrm{NegativeBinomial}(\alpha, p)$ であるとし,

$$H(p) = P(M \le m) = \sum_{i \le m} \binom{\alpha+i-1}{i} p^\alpha (1-p)^i$$

とおく. このとき,

$$\begin{aligned}
\frac{\partial}{\partial p}(p^\alpha(1-p)^i) &= \alpha t^{\alpha-1}(1-t)^i - i t^\alpha (1-p)^{i-1} \\
&= \alpha t^{\alpha-1}(1-t)^i - i(t^{\alpha-1}(-(1-t)+1)(1-t)^{i-1} \\
&= \alpha t^{\alpha-1}(1-t)^i + i t^{\alpha-1}(1-t)^i - i t^{\alpha-1}(1-t)^{i-1} \\
&= (\alpha+i) t^{\alpha-1}(1-t)^i - i t^{\alpha-1}(1-t)^{i-1}
\end{aligned}$$

より

$$\begin{aligned}
H'(t) &= \sum_{i \le m}(\alpha+i)\binom{\alpha+i-1}{i} t^{\alpha-1}(1-t)^i - \sum_{i \le m} i \binom{\alpha+i-1}{i} t^{\alpha-1}(1-t)^{i-1} \\
&= \sum_{i \le m}(i+1)\binom{\alpha+i}{i+1} t^{\alpha-1}(1-t)^i - \sum_{i \le m} i \binom{\alpha+i-1}{i} p^{\alpha-1}(1-p)^{i-1} \\
&= \sum_{i \le m+1} i \binom{\alpha+i-1}{i} t^{\alpha-1}(1-t)^{i-1} - \sum_{i \le m} i \binom{\alpha+i-1}{i} t^{\alpha-1}(1-t)^{i-1} \\
&= (m+1)\binom{\alpha+m}{m+1} t^{\alpha-1}(1-t)^m \\
&= -\frac{\Gamma(\alpha+m+1)}{\Gamma(\alpha)\Gamma(m+1)} t^{\alpha-1}(1-t)^m \\
&= -\frac{1}{B(\alpha, m+1)} t^{\alpha-1}(1-t)^m.
\end{aligned}$$

これはベータ分布 $\mathrm{Beta}(\alpha, m+1)$ の確率密度函数の形をしている. $H(0) = 0$ より,

$$P(M \le m) = H(p) = \int_0^p H'(t)\,dt = \frac{\int_0^p t^{\alpha-1}(1-t)^m\,dt}{B(\alpha, m+1)}.$$

以上より, 整数とは限らない $\alpha > 0$ についても以下が成立することがわかった:

$$\begin{aligned} &M \sim \mathrm{NegativeBinomial}(\alpha, p), \quad T \sim \mathrm{Beta}(\alpha, m+1) \\ &\implies P(M \le m) = P(T \le p), \quad P(M > m) = 1 - P(T \le p). \end{aligned}$$

注意: 負の二項分布の累積分布函数 $P(M \le m)$ の計算を定義通りに行うと, $m+1$ 個の数値の和を取る計算をすることになる. しかし, 以上の結果を使って, $P(M \le m)$ の計算をベータ分布の累積分布函数 $P(T \le p)$ の値として計算すれば, 基本特殊函数ライブラリの中に含まれている正則化された(不完全)ガンマ函数を使って効率的に計算できる.

In [61]:
```
cdf(NegativeBinomial(3.14159, 0.7), 2), cdf(Beta(3.14159, 2+1), 0.7)
```

Out[61]:
```
(0.8243946882545398, 0.8243946882545398)
```

In [62]:
```
plot(plot_cdfnegbinbeta(3.14159, 0.7),plot_cdfnegbinbeta(3.14159, 0.3);
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[62]:

6.11 おまけ: 二項分布と負の二項分布のP値のBayes統計的解釈への応用

二項分布の場合と同様に, 負の二項分布の場合にもベータ分布は共役事前分布になっている.

以下では二項分布と負の二項分布とベータ分布の関係から, 二項分布モデルおよび負の二項分布モデルの片側検定のP値がBayes統計的な解釈を持つことを説明しよう. ただし, P値や検定などに関する解説はここでは省略し, 結果だけを簡潔に記す.

ベータ分布の累積分布函数と二項分布・負の二項分布の累積分布函数達の関係より,

$$\begin{aligned} \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)} &= \sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} = \sum_{j=k}^{n} \binom{j-1}{k-1} p^k (1-p)^{j-k}, \\ \frac{\int_p^1 t^{k}(1-t)^{n-k-1}\,dt}{B(k+1, n-k)} &= \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i} = \sum_{j=n+1}^{\infty} \binom{j-1}{(k+1)-1} p^{k+1} (1-p)^{j-(k+1)}, \\ \frac{\int_p^1 t^{k-1}(1-t)^{n-k-1}\,dt}{B(k, n-k)} &= \sum_{i=0}^{k-1} \binom{n-1}{i} p^i (1-p)^{(n-1)-i} = \sum_{j=n}^{\infty} \binom{j-1}{k-1} p^k (1-p)^{j-k}. \end{aligned}$$

すなわち,

$$\begin{aligned} \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p) &= \mathrm{ccdf}(\mathrm{Binomial}(n, p), k-1) \\ &= \mathrm{cdf}(\mathrm{NegativeBinomial}(k, p), n-k), \end{aligned}$$

$$\begin{aligned} \mathrm{ccdf}(\mathrm{Beta}(k+1, n-k), p) &= \mathrm{cdf}(\mathrm{Binomial}(n, p), k) \\ &= \mathrm{ccdf}(\mathrm{NegativeBinomial}(k+1, p), n-(k+1)), \end{aligned}$$

$$\begin{aligned} \mathrm{ccdf}(\mathrm{Beta}(k, n-k), p) &= \mathrm{cdf}(\mathrm{Binomial}(n-1, p), k-1) \\ &= \mathrm{ccdf}(\mathrm{NegativeBinomial}(k, p), (n-1)-k). \end{aligned}$$

これは「試行回数 n, 成功回数 k」というデータについて, 以下のようになることを含む:

$$
\begin{aligned}
&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ での仮説 } p \le p_0 \text{ の片側検定のP値}) \\
&= (\text{負の二項分布 } \mathrm{NegativeBinomial}(k,p) \text{ での仮説 } p \le p_0 \text{ の片側検定のP値}) \\
&= (\text{ベータ分布 } \mathrm{Beta}(k, n-k+1) \text{ 内で仮説 } p \le p_0 \text{ が成立する確率})
\end{aligned}
$$

$$
\begin{aligned}
&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ での仮説 } p \ge p_0 \text{ の片側検定のP値}) \\
&= (\text{ベータ分布 } \mathrm{Beta}(k+1, n-k) \text{ 内で仮説 } p \ge p_0 \text{ が成立する確率})
\end{aligned}
$$

$$
\begin{aligned}
&(\text{負の二項分布 } \mathrm{NegativeBinomial}(k,p) \text{ での仮説 } p \ge p_0 \text{ の片側検定のP値}) \\
&= (\text{ベータ分布 } \mathrm{Beta}(k, n-k) \text{ 内で仮説 } p \ge p_0 \text{ が成立する確率}).
\end{aligned}
$$

これは以下が成立していることを意味している:

(1) 二項分布 $\mathrm{Binomial}(n,p)$ と負の二項分布 $\mathrm{NegativeBinomial}(k,p)$ のどちらにおいても, 仮説 $p \le p_0$ の片側検定のP値は, improper事前分布 $\mathrm{Beta}(0,1)$ から定まる事後分布内でその仮説が成立する確率に等しい.

(2) 二項分布 $\mathrm{Binomial}(n,p)$ での仮説 $p \ge p_0$ の片側検定のP値は, improper事前分布 $\mathrm{Beta}(1,0)$ から定まる事後分布内でその仮説が成立する確率に等しい.

(3) 負の二項分布 $\mathrm{NegativeBinomial}(k,p)$ での仮説 $p \ge p_0$ の片側検定のP値は, improper事前分布 $\mathrm{Beta}(0,0)$ から定まる事後分布内で仮説 $p \ge p_0$ が成立する確率に等しい.

このとき, (2)と(3)が一致していないことに注意せよ.

In [63]:
```
rd(x) = round(x; digits=10)
n, k, p = 10, 1:9, 0.2
```

Out[63]:
```
(10, 1:9, 0.2)
```

In [64]:
```
[
    @. cdf(Beta(k, n-k+1), p);;
    @. ccdf(Binomial(n, p), k-1);;
    @. cdf(NegativeBinomial(k, p), n-k);;
] .▷ rd
```

Out[64]:
```
9×3 Matrix{Float64}:
 0.892626     0.892626     0.892626
 0.62419      0.62419      0.62419
 0.3222       0.3222       0.3222
 0.120874     0.120874     0.120874
 0.0327935    0.0327935    0.0327935
 0.00636938   0.00636938   0.00636938
 0.000864358  0.000864358  0.000864358
 7.79264e-5   7.79264e-5   7.79264e-5
 4.1984e-6    4.1984e-6    4.1984e-6
```

In [65]:
```
[
    @. ccdf(Beta(k+1, n-k), p);;
    @. cdf(Binomial(n, p), k);;
    @. ccdf(NegativeBinomial(k+1, p), n-(k+1));;
] .▷ rd
```

Out[65]:
```
9×3 Matrix{Float64}:
 0.37581   0.37581   0.37581
 0.6778    0.6778    0.6778
 0.879126  0.879126  0.879126
 0.967207  0.967207  0.967207
 0.993631  0.993631  0.993631
 0.999136  0.999136  0.999136
 0.999922  0.999922  0.999922
 0.999996  0.999996  0.999996
 1.0       1.0       1.0
```

In [66]:
```
[
    @. ccdf(Beta(k, n-k), p);;
    @. cdf(Binomial(n-1, p), k-1);;
    @. ccdf(NegativeBinomial(k, p), (n-1)-k);;
] .▷ rd
```

Out[66]:
```
9×3 Matrix{Float64}:
 0.134218  0.134218  0.134218
 0.436208  0.436208  0.436208
 0.738198  0.738198  0.738198
 0.914358  0.914358  0.914358
 0.980419  0.980419  0.980419
 0.996934  0.996934  0.996934
 0.999686  0.999686  0.999686
 0.999981  0.999981  0.999981
 0.999999  0.999999  0.999999
```

6.12 二項分布の累積分布函数と負の二項分布の累積分布函数の関係(まとめ)

以上によって, 以下が示された:

$$K \sim \mathrm{Binomial}(n,p), \quad T \sim \mathrm{Beta}(k, n-k+1), \quad M = N-k \sim \mathrm{NegativeBinomial}(k,p)$$

のとき,

$$\begin{aligned} P(K \geq k) &= P(T \leq p) = P(M \leq n-k) = P(N \leq n), \\ P(K < k) &= P(T > p) = P(M > n-k) = P(N > n). \end{aligned}$$

これを具体的な式で書くと,

$$\begin{aligned} \sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} &= \frac{\int_0^p t^{k-1}(1-t)^{n-k}}{B(k, n-k+1)} \\ &= \sum_{i=0}^{n-k} \binom{k+i-1}{i} p^k (1-p)^i = \sum_{j=k}^{n} \binom{j-1}{j-k} p^k (1-p)^{j-k}, \\ \sum_{i=0}^{k-1} \binom{n}{i} p^i (1-p)^{n-i} &= \frac{\int_p^1 t^{k-1}(1-t)^{n-k}}{B(k, n-k+1)} \\ &= \sum_{i=n-k+1}^{\infty} \binom{k+i-1}{i} p^k (1-p)^i = \sum_{j=n+1}^{\infty} \binom{j-1}{j-k} p^k (1-p)^{j-k}. \end{aligned}$$

これら2つの公式それぞれの中の2番目の等号は k が整数でなくても, k が正の実数で $n-k$ が0以上の整数ならば成立している.

In [67]:
```
k, p = 3.14, 0.5772
mycdf_negbin(k, p, m) = sum(mybinomial(k+i-1,i)*p^k*(1-p)^i for i in 0:m)
[(m, mycdf_negbin(k, p, m), cdf(Beta(k, m+1), p)) for m in 0:10]
```

Out[67]:
```
11-element Vector{Tuple{Int64, Float64, Float64}}:
 (0, 0.17805929618323893, 0.1780592961832391)
 (1, 0.4144493933217376, 0.4144493933217378)
 (2, 0.6213370607769629, 0.6213370607769628)
 (3, 0.7712059353810816, 0.7712059353810814)
 (4, 0.8684705352614055, 0.8684705352614053)
 (5, 0.9271948544617901, 0.9271948544617898)
 (6, 0.9608790456560383, 0.960879045656038)
 (7, 0.9794746055099702, 0.9794746055099699)
 (8, 0.9894399474401325, 0.9894399474401322)
 (9, 0.9946551341921691, 0.9946551341921688)
 (10, 0.997331981076105, 0.9973319810761047)
```

In [68]:
```
function plot_cdf_bin_beta_nb(n, p; xtick=0:n)
    bin = Binomial(n, p)
    beta(k) = Beta(k, n-k+1)
    nb(k) = NegativeBinomial(k, p)
    μ, σ = mean(bin), std(bin)
    k = range(max(1, μ-4σ), μ+5σ, 1000)
    plot(k, k -> ccdf(bin, k-1); label="ccdf(Bin(n,p),k-1)")
    plot!(k, k -> cdf(beta(floor(k)), p); ls=:dash, label="cdf(Beta(k,n-⌊k⌋+1),p)")
    plot!(k, k -> cdf(nb(floor(k)), n - floor(k)); ls=:dashdot, label="cdf(NegBin(⌊k⌋,p),n-⌊k⌋
    plot!(; xlabel="k", xtick)
    title!("n=$n, p=$p")
end
```

Out[68]: plot_cdf_bin_beta_nb (generic function with 1 method)

In [69]:
```
plot(plot_cdf_bin_beta_nb(10, 0.3), plot_cdf_bin_beta_nb(30, 0.3);
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[69]:

In [70]:
```
plot(plot_cdf_bin_beta_nb(100, 0.3; xtick=0:5:100), plot_cdf_bin_beta_nb(300, 0.3; xtick=0:10:
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[70]:

6.13 Poisson分布の累積分布函数とガンマの累積分布函数の関係(再掲, まとめ)

二項分布と負の二項分布の連続時間極限でそれぞれPoisson分布とガンマ分布が得られるのであった. ゆえに, 前節でまとめた結果は以下の連続時間版の結果の離散時間版だとみなされる:

$$K \sim \mathrm{Poisson}(\lambda\tau), \quad T \sim \mathrm{Gamma}(\alpha = \kappa, \theta = 1/\lambda)$$

のとき,

$$P(K \geq \kappa) = P(T \leq \tau), \quad P(K < \kappa) = P(T > \tau).$$

これを具体的な式で書くと,

$$\sum_{j=\kappa}^{\infty} \frac{e^{-\lambda\tau}(\lambda\tau)^j}{j!} = \frac{\int_0^\tau e^{-\lambda t} t^{\kappa-1}\,dt}{\lambda^{-\kappa}\Gamma(\kappa)},$$
$$\sum_{j=0}^{\kappa-1} \frac{e^{-\lambda\tau}(\lambda\tau)^j}{j!} = \frac{\int_\tau^\infty e^{-\lambda t} t^{\kappa-1}\,dt}{\lambda^{-\kappa}\Gamma(\kappa)}.$$

In [71]:
```
function plot_cdf_poi_gam(λ, τ; xtick=0:round(Int, λ*τ+10√(λ*τ)))
    poi = Poisson(λ*τ)
    gam(κ) = Gamma(κ, 1/λ)
    μ, σ = mean(poi), std(poi)
    κ = range(max(1, μ-4σ), μ+5σ, 1000)
    plot(κ, κ -> ccdf(poi, κ-1); label="ccdf(Poisson(λτ),κ-1)")
    plot!(κ, κ -> cdf(gam(floor(κ)), τ); label="cdf(Gamma(⌊κ⌋,1/λ),τ)", ls=:dash)
    plot!(; xlabel="κ", xtick)
    title!("λ=$λ, τ=$τ")
end
```

Out[71]: plot_cdf_poi_gam (generic function with 1 method)

In [72]:
```
plot(plot_cdf_poi_gam(1, 1), plot_cdf_poi_gam(1, 3; xtick=0:5:100);
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[72]:

In [73]:
```
plot(plot_cdf_poi_gam(3, 3; xtick=0:5:1000), plot_cdf_poi_gam(3, 10; xtick=0:20:1000);
    size=(800, 250), bottommargin=4Plots.mm)
```

Out[73]:

7 ベータ二項分布

我々は以下が成立することを知っている:

(a) 期待値パラメータ λ がガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ に従う確率変数になっているようなPoisson分布 $\mathrm{Poisson}(\lambda)$ に従う確率変数が従う分布は負の二項分布 $\mathrm{NegativeBinomial}(\alpha, 1/(1+\theta))$ になる.

そしてさらに

(1) 二項分布 $\mathrm{Binomial}(n, \lambda/n)$ は $n \to \infty$ でPoisson分布 $\mathrm{Poisson}(\lambda)$ に収束する.

(2) $T_n \sim \mathrm{Beta}(\alpha, n+b)$ のとき $\theta n T_n$ が従う分布は $n \to \infty$ でガンマ分布 $\Gamma(\alpha, \theta)$ に収束する.

以上の結果から, まだ埋まっていない穴を埋めるとベータ二項分布が自然に得られる. 結果的に以下が成立することになる:

(b) 二項分布における成功確率パラメータ p がベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従う確率変数になっているような二項分布 $\mathrm{Binomial}(n, p)$ に従う確率変数が従う分布は **ベータ二項分布** になる.

(3) ベータ二項分布の連続時間極限として, 負の二項分布が得られる. しかも, その極限は(1), (2)と整合的である.

7.1 ベータ二項分布の定義

次の確率質量函数で定義される有限離散分布を **ベータ二項分布** (Beta-binomial disttribution)と呼び,

$$\mathrm{BetaBinomial}(n, \alpha, \beta)$$

と表す:

$$P(k|n, \alpha, \beta) = \binom{n}{k} \frac{B(\alpha + k, \beta + n - k)}{B(\alpha, \beta)} \quad (k = 0, 1, \ldots, n)$$

次が成立していることに注意せよ:

$$P(k|n, \alpha, \beta) = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)} \, dp.$$

この式は二項分布 $\mathrm{Binomial}(n, p)$ の成功確率パラメータ p がベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従って確率的に揺らいでいる場合の分布を表す確率質量函数になっている.

ベータ二項分布の乱数は以下の手続きで生成可能である:

1. ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従って乱数 p を生成する.
2. その p を採用した二項分布 $\mathrm{Binomial}(n, p)$ に従って乱数 k を生成する.

このように k を大量に生成したときの k の分布がベータ二項分布になる.

以下はその確認である.

In [74]:

```
function plot_bb(n, α, β; L = 10^6, kwargs...)
    # Monte Carlo シミュレーション
    P = rand(Beta(α, β), L) # ベータ分布の乱数を大量に生成
    K = @. rand(Binomial(n, P)) # 各p∈Pごとに二項分布Binomial(n,p)で乱数を生成

    # 比較のための同時プロット
    kmin, kmax = round.(Int, quantile.(BetaBinomial(n, α, β), (0.001, 0.999)))
    x = range(kmin-0.6, kmax+0.6, 1000)
    histogram(K; norm=true, alpha=0.3, bin=kmin-0.5:kmax+0.5, label="Monte Carlo")
    plot!(x, x -> pdf(BetaBinomial(n, α, β), round(Int, x)); label="BetaBin(n,α,β)", lw=1.5)
    plot!(x, x -> pdf(Binomial(n, α/(α+β)), round(Int, x)); label="Bin(n,α/(α+β))", ls=:dash)
    title!("n=$n, α=$α, β=$β")
    plot!(; kwargs...)
end

function plot_beta(α, β)
    plot(Beta(α, β); label="", title="Beta(α=$α, β=$β)", xtick=0:0.1:1)
end
```

Out[74]:

```
plot_beta (generic function with 1 method)
```

In [75]:

```
plot(plot_bb(20, 3, 7), plot_bb(20, 9, 21); size=(800, 250))
```

Out[75]:

In [76]:
```
plot(plot_bb(20, 30, 70), plot_bb(20, 90, 210); size=(800, 250))
```

Out[76]:

ベータ分布で p をランダムに生成して, その p を採用した二項分布で k をランダムに生成することを繰り返した Monte Carloシミュレーション(mc sim)の結果と, ベータ二項分布の確率質量函数のプロット($\mathrm{BetaBin}(\alpha, \beta)$)がぴったり一致している. 破線は二項分布のグラフである.

$\alpha/(\alpha+\beta)$ を保ったまま α, β を大きくすると分布 $\mathrm{BetaBinomial}(n, \alpha, \beta)$ は 分布 $\mathrm{Binomial}(n, \alpha/(\alpha+\beta)$ に近付く. そうなる理由は $\alpha/(\alpha+\beta)$ を保ったまま α, β を大きくすると, 分布 $\mathrm{Beta}(\alpha, \beta)$ が $\alpha/(\alpha+\beta)$ に集中するようになるからである.

In [77]:
```
plot(plot_beta(3, 7), plot_beta(9, 21); size=(800, 250))
```

Out[77]:

In [78]:
```
plot(plot_beta(30, 70), plot_beta(90, 210); size=(800, 250))
```

Out[78]:

7.2 ベータ二項分布の「成功確率が確率的に揺らいでいる二項分布」という解釈

ある母集団に含まれる医師を m 人選び, $i = 1, 2, \ldots, m$ のそれぞれについて i 番目の医師に n_i 人の患者を方法Aで治療してもらい, 治療に成功した人数 k_i を記録したとする. (本当は対照群を設ける必要があるが, 説明の単純化のためにここでは扱わない.)

このとき n_i 人中の治療成功数 k_i 達のデータの1つの単純なモデル化は, 医師によらずに方法Aによる共通の治療成功確率 p が決まっていて, 成功回数 k_i は共通の成功確率 p を持つ二項分布 $\mathrm{Binomial}(n_i, p)$ に従ってランダムに決まっていると想定することである.

統計モデルとして現実には確実に一致しないことが明らかな確率分布を選ぶことが多い. (しかし, 目的のために十分な精度で現実の状況をうまく把握できれば統計モデルの設定は成功したことになる.)

例えば, 上の単純な統計モデルの前提である「医師によらずに治療成功確率は同じである」という仮定は非常に疑わしいかもしれない. その場合には, 上の統計モデルの採用は失敗に終わる可能性が高い. 医師達のあいだで方法Aによる治療技術が違っている可能性を考慮する必要があるだろう.

そのような場合には, 「医師達に共通の成功確率 p が決まっている」という設定を, 「医師ごとに成功確率は違っていて, 各医師固有の成功確率はある確率分布にしたがってばらついている」という設定に緩めれば良さそうである.

そのような統計モデルの有力候補の1つがベータ二項分布である. ベータ二項分布における成功回数 k は以下のように生成されるのであった:

1. ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従って乱数 p を生成する.
2. その p を成功確率として採用した二項分布 $\mathrm{Binomial}(n, p)$ に従って乱数 k を生成する.

ベータ分布で乱数 p を生成するところが, 選んだ医師の治療成功確率 p がベータ分布に従ってランダムに決まることを記述していると解釈可能である. 現実の医師達についてはそう単純ではないことは確実だが, このようなモデル設定を試せることも知っていると, 統計分析の成功可能性が高まるだろう.

上で見たように, α, β が大きくなると, ベータ分布がランダムに生成する成功確率 p のばらつきの幅が小さくなり, ベータ二項分布はただの二項分布に近付く. このような場合にはよりシンプルな二項分布モデルを採用した方が合理的な可能性も出て来る.

7.3 ベータ二項分布での確率の総和が 1 になることの確認

ベータ二項分布の確率質量函数の次の表示を使う:

$$P(k|n, \alpha, \beta) = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \, \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)} \, dp.$$

これを使うと,

$$\begin{aligned}\sum_{k=0}^{n} P(k|n, \alpha, \beta) &= \int_0^1 \underbrace{\sum_{k=0}^{n} \binom{n}{k} p^k (1-p)^{n-k}}_{=1} \, \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)} \, dp \\ &= \int_0^1 \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)} \, dp = 1.\end{aligned}$$

一般にパラメータ θ を持つ確率質量函数 $P(x|\theta)$ と θ に関するパラメータ η を持つ確率密度函数 $p(\theta|\eta)$ に対して,

$$\tilde{P}(x|\eta) = \int P(x|\theta) p(\theta|\eta) \, d\theta$$

によって, パラメータ η を持つ確率質量函数 $\tilde{P}(x|\eta)$ を構成できる. 確率の総和が 1 になることは上の計算と同様で以下のようにして示される:

$$\sum_x \tilde{P}(x|\eta) = \int \underbrace{\sum_x P(x|\theta)}_{=1} \, p(\theta|\eta) \, d\theta = \int p(\theta|\eta) \, d\theta = 1.$$

確率質量函数 $P(x|\theta)$ を確率密度函数 $p(x|\theta)$ で置き換えた場合も同様である.

7.4 問題: ベータ二項分布の期待値と分散

ベータ二項分布 $\mathrm{Beta}(n, \alpha, \beta)$ の期待値と分散が以下のようになることを示せ: $K \sim \mathrm{Beta}(n, \alpha, \beta)$ とすると,

$$E[K] = \frac{n\alpha}{\alpha + \beta}, \quad \mathrm{var}(K) = \frac{n\alpha\beta(\alpha + \beta + n)}{(\alpha + \beta)^2(\alpha + \beta + 1)}.$$

注意: これらを二項分布と比較するために仮に $p = \alpha/(\alpha + \beta)$ とおくと,

$$E[K] = np, \quad \mathrm{var}(K) = np(1-p)\left(1 + \frac{n-1}{\alpha+\beta+1}\right).$$

期待値の方は二項分布と同じ形になったが, **分散の方は** 二項分布の場合の分散 $np(1-p)$ よりも $1+(n-1)/(\alpha+\beta+1)$ 倍だけ **大きくなっている**. 大雑把に言うと, $\alpha+\beta$ が n よりずっと大きい場合にはベータ二項分布は二項分布でよく近似されるようになる.

解答例: $K \sim \mathrm{Binomial}(n,p)$ について

$$\begin{aligned}
E[K(K-1)\cdots(K-m+1)] &= \sum_k k(k-1)\cdots(k-m+1)\frac{n!}{k!(n-k)!}p^k(1-p)^{n-k} \\
&= n(n-1)\cdots(n-m+1)p^m \underbrace{\sum_k \frac{(n-m)!}{(k-m)!(n-k)!}p^{k-m}(1-p)^{n-k}}_{=(p+(1-p))^{n-m}=1} \\
&= n(n-1)\cdots(n-m+1)p^m.
\end{aligned}$$

$P \sim \mathrm{Beta}(\alpha,\beta)$ について

$$\begin{aligned}
E[P^m] &= \int_0^1 p^m \frac{p^{\alpha-1}(1-p)^{\beta-1}\,dp}{B(\alpha,\beta)} = \frac{\int_0^1 p^{\alpha+m-1}(1-p)^{\beta-1}\,dp}{B(\alpha,\beta)} \\
&= \frac{B(\alpha+m,\beta)}{B(\alpha,\beta)} = \frac{\Gamma(\alpha+m)\Gamma(\beta)}{\Gamma(\alpha+\beta+m)}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \\
&= \frac{\alpha(\alpha+1)\cdots(\alpha+m-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+m-1)}.
\end{aligned}$$

ゆえに, $K \sim \mathrm{BetaBinomial}(n,\alpha,\beta)$ について,

$$\begin{aligned}
E[K(K-1)\cdots(K-m+1)] &= \sum_k k(k-1)\cdots(k-m+1)\int_0^1 \binom{n}{k}p^k(1-p)^{n-1}\frac{p^{\alpha-1}(1-p)^{\beta-1}\,dp}{B(\alpha,\beta)} \\
&= \int_0^1 \left(\sum_k k(k-1)\cdots(k-m+1)\binom{n}{k}p^k(1-p)^{n-1}\right)\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= \int_0^1 n(n-1)\cdots(n-m+1)p^m\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= n(n-1)\cdots(n-m+1)\int_0^1 p^m\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= n(n-1)\cdots(n-m+1)\frac{\alpha(\alpha+1)\cdots(\alpha+m-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+m-1)}.
\end{aligned}$$

特に,

$$E[K] = \frac{n\alpha}{\alpha+\beta}, \quad E[K(K-1)] = \frac{n(n-1)\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}.$$

ゆえに

$$E[K] - E[K]^2 = E[K](1-E[K]) = \frac{n\alpha}{\alpha+\beta}\left(1-\frac{n\alpha}{\alpha+\beta}\right) = \frac{n\alpha(\alpha+\beta-n\alpha)}{(\alpha+\beta)^2}$$

なので

$$\begin{aligned}
\mathrm{var}(K) &= E[K^2] - E[K]^2 = E[K(K-1)] + E[K] - E[K]^2 \\
&= \frac{n(n-1)\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)} + \frac{n\alpha(\alpha+\beta-n\alpha)}{(\alpha+\beta)^2} \\
&= \frac{n\alpha((n-1)(\alpha+1)(\alpha+\beta)+(\alpha+\beta-n\alpha)(\alpha+\beta+1))}{(\alpha+\beta)^2(\alpha+\beta+1)} \\
&= \frac{n\alpha(\alpha\beta+\beta^2+n\beta)}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{n\alpha\beta(\alpha+\beta+n)}{(\alpha+\beta)^2(\alpha+\beta+1)}.
\end{aligned}$$

解答終

In [79]:
```
@vars a b n
var"E[K]" = n*a/(a+b)
var"E[K(K-1)]" = n*(n-1)*a*(a+1)/((a+b)*(a+b+1))
var"var(K)" = var"E[K(K-1)]" + var"E[K]" - var"E[K]"^2 ▷ factor
```

Out[79]:

$$\frac{abn\left(a+b+n\right)}{\left(a+b\right)^{2}\left(a+b+1\right)}$$

7.5 問題: ベータ二項分布 = 負の超幾何分布

ベータ二項分布の確率質量函数

$$P(k|n,\alpha,\beta) = \binom{n}{k}\frac{B(\alpha+k,\beta+n-k)}{B(\alpha,\beta)} \quad (k=0,1,\ldots,n)$$

は以下のように書き換えられることを示せ:

$$\begin{aligned} P(k|n,\alpha,\beta) &= \binom{\alpha+k-1}{k}\binom{\beta+(n-k)-1}{n-k}\Big/\binom{\alpha+\beta+n-1}{n} \\ &= \binom{-\alpha}{k}\binom{-\beta}{n-k}\Big/\binom{-\alpha-\beta}{n}. \end{aligned}$$

注意: この形の確率質量函数は別に定義される有限離散分布のパラメータ $s,f,n\in\mathbb{Z}_{\geq 0}$, $n\leq s+f$ に関する **超幾何分布** $\mathrm{Hypergeometric}(s,f,n)$ の 確率質量函数

$$k\mapsto \binom{s}{k}\binom{f}{n-k}\Big/\binom{s+f}{n}.$$

における s,f に負の値 $-\alpha,-\beta$ を形式的に代入したものに一致しているので, ベータ二項分布は **負の超幾何分布** と呼ばれることがある.

注意: 上に書いた超幾何分布の確率は, s 個の赤い球と f 個の白い球が入っている袋から n 個をまとめて無作為に取り出したとき, その中にちょうど k 個の赤い球が含まれる確率になっている(s はsuccessの, f はfailureの頭文字). 実際, 分母の $\binom{n+m}{n}$ は $s+f$ 個から n 個を選ぶときの組み合わせの数であり, 分子は s 個の赤い球から k 個を選び, f 個の白い球から $n-k$ 個選ぶ組み合わせの個数であり, 分子を分母で割れば, n 個中ちょうど k 個が赤い球である確率になる. 超幾何分布自体は「超幾何」という難しそうな形容詞が付いているが, 初等的に理解できる確率分布である.

注意: 負の二項分布の確率質量函数は

$$P(m|\alpha,p) = \binom{\alpha+m-1}{m}p^{\alpha}(1-p)^{m} = (-1)^{m}\binom{-\alpha}{m}p^{\alpha}(1-p)^{m}$$

と書けるのであった. 二項分布と負の二項分布の関係は超幾何分布と負の超幾何分布=ベータ二項分布の関係に似ている.

解答例:

$$\begin{aligned} P(k|n,\alpha,\beta) &= \binom{n}{k}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+n)}\frac{\Gamma(\alpha+k)}{\Gamma(\alpha)}\frac{\Gamma(\beta+n-k)}{\Gamma(\beta)} \\ &= \frac{n!}{k!(n-k)!}\frac{\alpha(\alpha+1)\cdots(\alpha+k-1)\cdot\beta(\beta+1)\cdots(\beta+n-k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+n-1)} \\ &= \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)}{k!}\frac{\beta(\beta+1)\cdots(\beta+n-k-1)}{(n-k)!} \\ &\times \frac{n!}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+n-1)} \\ &= \binom{\alpha+k-1}{k}\binom{\beta+(n-k)-1}{n-k}\Big/\binom{\alpha+\beta+n-1}{n} \end{aligned}$$

さらに,

$$\begin{aligned} (-1)^{k}\binom{-a}{k} &= (-1)^{k}\frac{(-a)(-a-1)\cdots(-a-k+1)}{k!} \\ &= \frac{a(a+1)\cdots(a+k-1)}{k!} = \binom{a+k-1}{k} \end{aligned}$$

なので,

$$P(k|n,\alpha,\beta) = \binom{-\alpha}{k}\binom{-\beta}{n-k}\Big/\binom{-\alpha-\beta}{n}.$$

解説終

7.6 超幾何分布, 二項分布, ベータ二項分布の統一的な理解

この節の内容は本質的に [Pólya urn model (https://www.google.com/search?q=P%C3%B3lya+urn+model)](https://www.google.com/search?q=P%C3%B3lya+urn+model) の話である. この節では α と β は 0 以上の整数であり, $\alpha+\beta>0$ であるとし, δ は -1 以上の整数であると仮定する.

最初の状態では袋の中に赤い球が α 個, 白い球が β 個入っているとし, 以下の操作を n 回繰り返そう:

1. 袋の中から玉を無作為に取り出し, その色を記録する.
2. その玉が赤いならば袋の中に赤い球を $\delta+1$ 個戻し(袋の中の赤い球は δ 個増える($\delta=-1$ ならば 1 個減る)), 白いならば袋の中に白い球を $\delta+1$ 個戻す(袋の中の白い球は δ 個増える($\delta=-1$ ならば 1 個減る)).

このとき, n 回中 k 回赤い球が取り出される確率は次のように書ける:

$$P(k|n,\alpha,\beta,\delta) = \binom{n}{k}\frac{\alpha(\alpha+\delta)\cdots(\alpha+(k-1)\delta)\cdot\beta(\beta+\delta)\cdots(\beta+(n-k-1)\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)\cdots(\alpha+\beta+(n-1)\delta)} \quad (k=0,1,\dots,n).$$

注意: $\delta=-1$ のときであっても, $k>\alpha$ ならば

$$\alpha(\alpha+\delta)\cdots(\alpha+(k-1)\delta) = \alpha(\alpha-1)\cdots(\alpha-\alpha)\cdots(\alpha-(k-1)) = 0$$

となるので $p(k|n,\alpha,\beta,-1)=0$ となり, 同様にして $n-k>\beta$ ならば $p(k|n,\alpha,\beta,-1)=0$ となることがわかる. これより, $\delta=-1$ であっても $p(k|n,\alpha,\beta,-1)$ $(k=0,1,\dots,n)$ が負の値にならないことがわかる.

注意: $\delta>0$ ならば分子分母を $\delta^k\delta^{n-k}=\delta^n$ で割ることによって形式的に δ の部分を 1 にでき, α,β を 0 以上の正の有理数にできるので, $\delta=1$ の場合が本質的だと感が得られる. そして, その場合の極限として, α,β が 0 以上の任意の実数で $\delta=1$ の場合も理解できる.

注意: α, β を球の個数とみなす解釈を止めて, α,β,δ は任意の 0 以上の実数で $\alpha+\beta>0$ を満たすという設定にしても, 上の $p(k|n,\alpha,\beta,\delta)$ の定義式は確率質量函数を与える. その場合には, 最初の状態ではランダムに赤い玉が確率 $\alpha/(\alpha+\beta)$ で, 白い球が確率 $\beta/(\alpha+\beta)$ で出て来て, 赤い玉が出たらその次に赤い玉の出る確率が $(\alpha+\delta)/(\alpha+\beta+\delta)$ に増加し, 白い玉が出たらその次に白い玉の出る確率が $(\beta+\delta)/(\alpha+\beta+\delta)$ に増加し, その後も同様とすると考えればよい. このような設定の場合にも, 上の注意で述べたのと同じ理由で, $\delta>0$ の場合は $\delta=1$ の場合に帰着される. 数学的な設定としてなら, 袋の中に非整数個の赤い球と白い球が入っており, 取り出される確率がそれぞれの個数に比例すると考えてもよいだろう.

袋に取り出した玉を戻さない非復元抽出の場合 ($\delta=-1$): n 回中 k 回赤い球が取り出される確率の分布は超幾何分布 $\mathrm{Hypergeometric}(\alpha,\beta,n)$ になる:

$$\begin{aligned}
P(k|n,\alpha,\beta,-1) &= \binom{n}{k}\frac{\alpha(\alpha-1)\cdots(\alpha-(k-1))\cdot\beta(\beta-1)\cdots(\beta-(n-k-1)}{(\alpha+\beta)(\alpha+\beta-1)\cdots(\alpha+\beta-(n-1))} \\
&= \binom{\alpha}{k}\binom{\beta}{n-k}\Big/\binom{\alpha+\beta}{n}.
\end{aligned}$$

これを超幾何分布 $\mathrm{Hypergeometric}(\alpha,\beta,n)$ の定義だと考えてよい.

袋に取り出した玉をそのまま玉を戻す復元抽出の場合 ($\delta=0$): n 回中 k 回赤い球が取り出される確率の分布は二項分布 $\mathrm{Binomial}(n,\alpha/(\alpha+\beta))$ になる:

$$P(k|n,\alpha,\beta,0) = \binom{n}{k}\frac{\alpha^k\beta^{n-k}}{(\alpha+\beta)^n} = \binom{n}{k}\left(\frac{\alpha}{\alpha+\beta}\right)^k\left(1-\frac{\alpha}{\alpha+\beta}\right)^{n-k}.$$

袋に取り出した玉と同じ色の玉を2個戻して袋の中のその色の玉が1個増えるようにした場合 ($\delta=0$): n 回中 k 回赤い球が取り出される確率の分布はベータ二項分布 $\mathrm{BetaBinomial}(n,\alpha,\beta)$ になる:

$$\begin{aligned}
P(k|n,\alpha,\beta,1) &= \binom{n}{k}\frac{\alpha(\alpha+1)\cdots(\alpha+(k-1))\cdot\beta(\beta+1)\cdots(\beta+(n-k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+(n-1))} \\
&= \binom{n}{k}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+n)}\frac{\Gamma(\alpha+k)}{\Gamma(\alpha)}\frac{\Gamma(\beta+n-k)}{\Gamma(\beta)} \\
&= \binom{n}{k}\frac{B(\alpha+k,\beta+n-k)}{B(\alpha,\beta)}.
\end{aligned}$$

袋の中の赤い球の個数と白い球の個数の初期値である α, β が n よりも非常に大きい場合には上の3つの場合の確率はほぼ同じになる.

ベータ二項分布の場合には, 赤い球を引けた人はその次に赤い球を引く確率が少し上がる. この意味でベータ二項分布は "The rich get richer and the poor get poorer. (https://www.google.com/search?q=the+rich+get+richer+and+the+poor+get+poorer)" (「金持ちはより金持ちに、貧乏人はより貧乏に」)という状況のモデル化になっていることがわかる.

In [80]:
```
function plot_hg_bin_bb(n, α, β; xtick=0:n)
    @assert n < α + β "n must be less than α + β."
    hg  = Hypergeometric(α, β, n)
    bin = Binomial(n, α/(α+β))
    bb  = BetaBinomial(n, α, β)
    μ, σ = mean(bb), std(bb)
    x = range(max(-0.5, μ-4σ), min(n+0.5, μ+4σ), 1000)
    plot(; xtick)
    plot!(x, x -> pdf(hg,  round(Int, x)); label="Hypergeom")
    plot!(x, x -> pdf(bin, round(Int, x)); label="Binomial", ls=:dash)
    plot!(x, x -> pdf(bb,  round(Int, x)); label="BetaBin",  ls=:dashdot)
    title!("n=$n, α=$α, β=$β")
end
```

Out[80]: plot_hg_bin_bb (generic function with 1 method)

以下は $\alpha/(\alpha+\beta) = 1/4$ を保ったまま α, β を大きくして行った場合の超幾何分布, 二項分布, ベータ二項分布の比較である. α, β を大きくして行くと, 3つの分布は一致して行く(直観的には当然そうなるべき結果).

In [81]:
```
plot(plot_hg_bin_bb(10, 3, 9), plot_hg_bin_bb(10, 10, 30); size=(800, 250))
```

Out[81]:

In [82]:
```
plot(plot_hg_bin_bb(10, 30, 90), plot_hg_bin_bb(10, 100, 300); size=(800, 250))
```

Out[82]:

7.7 問題: 超幾何分布, 二項分布, ベータ二項分布の期待値と分散の統一的な公式

前節の確率質量函数 $P(k|n, \alpha, \beta, \delta)$ を持つ確率変数 K について次が成立することを示せ:

$$E[K] = \frac{n\alpha}{\alpha+\beta}, \quad \mathrm{var}(K) = \frac{\alpha\beta n(\alpha+\beta+n\delta)}{(\alpha+\beta)^2(\alpha+\beta+\delta)}.$$

注意: 特にこれの $\delta = -1$, $\alpha = s$, $\beta = f$ の場合より, 超幾何分布 $\mathrm{Hypergeometric}(s, f, n)$ の期待値と分散が次のようになることがわかる: 確率変数 $K \sim \mathrm{Hypergeometric}(s, f, n)$ について

$$E[K] = \frac{ns}{s+f}, \quad \mathrm{var}(K) = \frac{sfn(s+f-n)}{(s+f)^2(s+f-1)}.$$

解答例: 問題を一般化して, 前節の確率質量函数

$$P(k|n,\alpha,\beta,\delta) = \binom{n}{k}\frac{\alpha(\alpha+\delta)\cdots(\alpha+(k-1)\delta)\cdot\beta(\beta+\delta)\cdots(\beta+(n-k-1)\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)\cdots(\alpha+\beta+(n-1)\delta)}.$$

を持つ確率変数 K について考える. このとき,

$$\begin{aligned}
&E[K(K-1)\cdots(K-m+1)] \\
&= \sum_k k(k-1)\cdots(k-m+1)\binom{n}{k}\frac{\alpha(\alpha+\delta)\cdots(\alpha+(k-1)\delta)\cdot\beta(\beta+\delta)\cdots(\beta+(n-k-1)\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)\cdots(\alpha+\beta+(n-1)\delta)} \\
&= n(n-1)\cdots(n-m+1)\frac{\alpha(\alpha+\delta)\cdots(\alpha+(m-1)\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)\cdots(\alpha+\beta+(m-1)\delta)} \\
&\quad\times\underbrace{\sum_k\binom{n-m}{k-m}\frac{(\alpha+m\delta)(\alpha+(m+1)\delta)\cdots(\alpha+(k-1)\delta)\cdot\beta(\beta+\delta)\cdots(\beta+(n-k-1)\delta)}{(\alpha+\beta+m)(\alpha+\beta+(m+1)\delta)\cdots(\alpha+\beta+(n-1)\delta)}}_{=\sum_k P(k|n-m,\alpha+m,\beta)=1} \\
&= n(n-1)\cdots(n-m+1)\frac{\alpha(\alpha+\delta)\cdots(\alpha+(m-1)\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)\cdots(\alpha+\beta+(m-1)\delta)}.
\end{aligned}$$

例えば,

$$E[K] = \frac{n\alpha}{\alpha+\beta}, \quad E[K(K-1)] = \frac{n(n-1)\alpha(\alpha+\delta)}{(\alpha+\beta)(\alpha+\beta+\delta)}.$$

これを使って少し計算すると,

$$\mathrm{var}(K) = E[K^2] - E[K]^2 = E[K(K-1)] + E[K] - E[K]^2 = \frac{\alpha\beta n(\alpha+\beta+n\delta)}{(\alpha+\beta)^2(\alpha+\beta+\delta)}$$

となることがわかる(以下のセルでのコンピュータによる代数計算も参照せよ).

解答終

In [83]:

```
@vars a b d n
var"E[K]" = n*a/(a+b)
var"E[K(K-1)]" = n*(n-1)*a*(a+d)/((a+b)*(a+b+d))
var"var(K)" = var"E[K(K-1)]" + var"E[K]" - var"E[K]"^2 ▷ factor
```

Out[83]:

$$\frac{abn\,(a+b+dn)}{(a+b)^2\,(a+b+d)}$$

7.8 問題: ベータ二項分布の極限として負の二項分布が得られる

$\alpha,\theta>0$ であるとし, ベータ二項分布 $\mathrm{BetaBinomial}(L,\alpha,\beta)$ の確率質量函数

$$P(m|L,\alpha,\beta) = \binom{L}{m}\frac{B(\alpha+m,\beta+L-m)}{B(\alpha,\beta)} \quad (m=0,1,\ldots,L)$$

は, $\beta = L/\theta$ とおいて, $L\to\infty$ とすると,

$$P(m|L,\alpha,L/\theta) \to \binom{\alpha+m-1}{m}\left(\frac{1}{1+\theta}\right)^{\alpha}\left(1-\frac{1}{1+\theta}\right)^{m}$$

と負の二項分布 $\mathrm{NegativeBinomial}(\alpha,1/(1+\theta))$ の確率質量函数に収束することを示せ.

解答例1: $n\to\infty$ で $n^{\alpha}B(\alpha,n+b)\to\Gamma(\alpha)$ となることを使うと, $L\to\infty$ のとき,

$$
\begin{aligned}
P(m|L,\alpha,L/\theta) &= \binom{L}{m}\frac{B(\alpha+m,L/\theta+L-m)}{B(\alpha,L/\theta)} \\
&= \frac{L(L-1)\cdots(L-m+1)}{m!}\frac{(L/\theta)^\alpha}{((1+\theta)/\theta)L)^{\alpha+m}} \\
&\times \frac{(((1+\theta)/\theta)L)^{\alpha+m}B(\alpha+m,((1+\theta)/\theta)L-m)}{(L/\theta)^\alpha B(\alpha,L/\theta)} \\
&= \underbrace{\frac{L(L-1)\cdots(L-m+1)}{L^m}}_{\to 1} \\
&\times \underbrace{\frac{(((1+\theta)/\theta)L)^{\alpha+m}B(\alpha+m,((1+\theta)/\theta)L-m)}{m!\,(L/\theta)^\alpha B(\alpha,L/\theta)}}_{\to\Gamma(\alpha+m)/(m!\Gamma(\alpha))}\left(\frac{1}{1+\theta}\right)^\alpha\left(\frac{\theta}{1+\theta}\right)^m \\
&\to \frac{\Gamma(\alpha+m)}{m!\Gamma(\alpha)}\left(\frac{1}{1+\theta}\right)^\alpha\left(\frac{\theta}{1+\theta}\right)^m \\
&= \binom{\alpha+m-1}{m}\left(\frac{1}{1+\theta}\right)^\alpha\left(1-\frac{1}{1+\theta}\right)^m.
\end{aligned}
$$

解答終

解答例2: せっかくなので, 積を取って積分してベータ二項分布の確率質量函数を作る前の二項分布の確率質量函数とベータ分布の確率密度函数の連続時間極限との整合性も確認してしまおう.

ベータ分布 $\mathrm{Beta}(\alpha,\beta)$ の確率密度函数 $\times dp$ については,

$$
\beta = L/\theta, \quad p = \lambda/L, \quad dp = d\lambda/L
$$

とおくと, $L\to\infty$ のとき以下が成立している:

$$
\begin{aligned}
\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp &= \frac{(\lambda/L)^{\alpha-1}(1-\lambda/L)^{L/\theta-1}}{B(\alpha,L/\theta)}\frac{d\lambda}{L} \\
&= \frac{\lambda^{\alpha-1}(1-\lambda/L)^{L/\theta-1}}{L^\alpha B(\alpha,L/\theta)}\,d\lambda \\
&= \frac{1}{\theta^\alpha\underbrace{(L/\theta)^\alpha B(\alpha,L/\theta)}_{\to\Gamma(\alpha)}}\underbrace{\left(1-\frac{\lambda}{L}\right)^{L/\theta-1}}_{\to\exp(-\lambda/\theta)}\lambda^{\alpha-1}\,d\lambda \\
&\to \frac{1}{\theta^\alpha\Gamma(\alpha)}e^{-\lambda/\theta}\lambda^{\alpha-1}\,d\lambda.
\end{aligned}
$$

$L\to\infty$ のとき, ベータ分布はこのように λ に関するガンマ分布 $\mathrm{Gamma}(\alpha,\theta)$ に近付く. p は 0 と 1 のあいだを動くが, $\lambda = Lp$ なので $L\to\infty$ の極限で λ は正の実数全体を動くことになる.

二項分布 $\mathrm{Binomial}(n,p)$ の確率質量函数については,

$$
n = L, \quad p = \lambda/L, \quad k = m
$$

とおくと, $L\to\infty$ のとき以下が成立している:

$$
\begin{aligned}
\binom{n}{k}p^k(1-p)^{n-k} &= \binom{L}{m}\left(\frac{\lambda}{L}\right)^m\left(1-\frac{\lambda}{L}\right)^{L-m} \\
&= \underbrace{\frac{L(L-1)\cdots(L-m+1)}{L^m}}_{\to 1}\underbrace{\left(1-\frac{\lambda}{L}\right)^{L-m}}_{\to\exp(-\lambda)}\frac{\lambda^m}{m!} \\
&\to e^{-\lambda}\frac{\lambda^m}{m!}.
\end{aligned}
$$

$L\to\infty$ のとき, 二項分布はこのように m に関するPoisson分布 $\mathrm{Poisson}(\lambda)$ に近付く.

したがって, ベータ二項分布 $\mathrm{Beta}(L,\alpha,L/\theta)$ の確率質量函数

$$
P(m|L,\alpha,L/\theta) = \int_0^1\binom{L}{m}p^k(1-p)^{L-m}\,\frac{p^{\alpha-1}(1-p)^{L/\theta-1}}{B(\alpha,L/\theta)}\,dp
$$

について, $p=\lambda/L$ とおくことによって, $L\to\infty$ で以下が成立することがわかる:

$$
\begin{aligned}
P(m|L,\alpha,\beta) &\to \int_0^\infty e^{-\lambda}\frac{\lambda^m}{m!}\frac{e^{-\lambda/\theta}\lambda^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}\,d\lambda \\
&= \frac{1}{m!\theta^\alpha\Gamma(\alpha)}\int_0^\infty e^{-((1+\theta)/\theta)\lambda}\lambda^{\alpha+m-1}\,d\lambda \\
&= \frac{1}{m!\theta^\alpha\Gamma(\alpha)}\left(\frac{\theta}{1+\theta}\right)^{\alpha+m}\Gamma(\alpha+m) \\
&= \frac{\Gamma(\alpha+m)}{m!\Gamma(\alpha)}\left(\frac{1}{1+\theta}\right)^{\alpha}\left(\frac{\theta}{1+\theta}\right)^{m} \\
&= \binom{\alpha+m-1}{m}\left(\frac{1}{1+\theta}\right)^{\alpha}\left(1-\frac{1}{1+\theta}\right)^{m}.
\end{aligned}
$$

$L \to \infty$ のとき, ベータ二項分布はこのように m に関する負の二項分布 $\mathrm{NegativeBinomial}(\alpha, 1/(1+\theta))$ に近付く.

解答終

In [84]:
```
function plot_bbnb(L, α, θ; kwargs...)
    bb = BetaBinomial(L, α, L/θ)
    nb = NegativeBinomial(α, 1/(1+θ))
    μ, σ = mean(nb), std(nb)
    m = range(max(-1, μ-4σ), μ+4σ, 1000)
    plot(m, m -> pdf(bb, round(m)); label="BetaBin(L,α,L/θ)")
    plot!(m, m -> pdf(nb, round(m)); label="NegBin(α,1/(1+θ))", ls=:dash)
    title!("L=$L, α=$α, θ=$θ, p=1/(1+θ)=1/$(1+θ)")
    plot!(; kwargs...)
end
```

Out[84]: plot_bbnb (generic function with 1 method)

In [85]:
```
plot(plot_bbnb.((10, 30), 3, 2)...; size=(800, 250))
```

Out[85]:

In [86]:
```
plot(plot_bbnb.((100, 300), 3, 2)...; size=(800, 250))
```

Out[86]:

7.9 問題: 二項分布とベータ二項分布のモーメント母函数

(1) 二項分布のモーメント母函数: 次が成立することを示せ:

$$K \sim \mathrm{Binomial}(n,p) \implies E[e^{tK}] = (1-(1-e^t)p)^n.$$

(2) ベータ二項分布のモーメント母函数が本質的にGaussの超幾何函数になっていることを示せ. すなわち, 次が成立することを示せ:

$$K \sim \mathrm{BetaBonomial}(n,\alpha,\beta) \implies E[e^{tK}] = {}_2F_1(\,-n,\alpha;\alpha+\beta;1-e^t\,).$$

ここで, ${}_2F_1(a,b;c;z)$ はGaussの超幾何函数で次のように定義される:

$$ {}_2F_1(a,b;c;z) = \frac{1}{B(b,c-b)}\int_0^1 (1-zt)^{-a}t^{b-1}(1-t)^{c-b-1}\,dt. \quad (\mathrm{Re}\,c > \mathrm{Re}\,b > 0)$$

解答例: (1) $K \sim \mathrm{Binomial}(n,p)$ のとき, 二項定理より,

$$E[e^{tK}] = \sum_{k=0}^{n}\binom{n}{k}e^{tk}p^k(1-p)^{n-k} = (e^tp+1-p)^n = (1-(1-e^t)p)^n.$$

(2) $K \sim \mathrm{BetaBonomial}(n,\alpha,\beta)$ と仮定する. 負の二項分布の確率質量函数の

$$P(k|n,\alpha,\beta) = \int_0^1 \binom{n}{k}p^k(1-p)^{n-k}\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp$$

という表示を使うと, (1)の計算より,

$$\begin{aligned}
E[e^{tK}] &= \sum_{k=0}^{n} e^{tk}\int_0^1 \binom{n}{k}p^k(1-p)^{n-k}\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= \int_0^1 \left(\sum_{k=0}^{n}\binom{n}{k}e^{tk}p^k(1-p)^{n-k}\right)\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= \int_0^1 (1-(1-e^t)p)^n\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp \\
&= \frac{1}{B(\alpha,\beta)}\int_0^1 (1-(1-e^t)p)^n p^{\alpha-1}(1-p)^{\beta-1}\,dp \\
&= {}_2F_1(\,-n,\alpha;\alpha+\beta;1-e^t\,).
\end{aligned}$$

解答終

注意: 二項分布もベータ分布も統計学における最も基本的な確率分布である. それらを組み合わせて作ったベータ二項分布から, モーメント母函数の形式でGaussの超幾何函数の積分表示式が自然に出て来ることはちょっと面白い. 超幾何函数のような特殊函数は主に微分方程式との関連で自然に現れて来ることが多いのだが, この場合には微分方程式と全く無関係に超幾何函数が自然に出て来ている. 統計学によく出て来る特殊函数達に関する話題は「微分方程式抜きの特殊函数論」というような趣がある.

文献: 超幾何函数から作られる確率分布については以下の文献を参照せよ:

- 清水良一. 一般超幾何分布. 統計数理研究所彙報, 第16巻, 第2号, 147-165, 1968-12. [link (https://ismrepo.ism.ac.jp/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=32361&item_no=1&page_id=13&block_id=
- Shibuya, Masaaki and Shimizu, Ryoichi. Classification of the Generalized Hypergeometric family of distributions. Keio Science and Technology Reports 34 (1), 1-38, 1981-09. [Google Scholar (https://scholar.google.co.jp/scholar?cluster=14585775532723000776)]
- Shibuya, Masaaki and Shimizu, Ryoichi. The generalized hypergeometric family of distributions. Annals of the Institute of Statistical Mathematics, volume 33, pages 177–190 (1981). [Google Scholar (https://scholar.google.co.jp/scholar?cluster=5536931452160381057)]

8 ベータ負の二項分布とベータプライム分布

この節ではベータ負の二項分布 $\mathrm{BetaNegativeBinomial}(k,\alpha,\beta)$ という離散分布を定義し, その連続時間極限としてベータプライム分布が現れることを説明する.

8.1 ベータ負の二項分布の定義

$k,\alpha,\beta > 0$ と仮定する. $m = 0,1,2,\ldots$ に関する確率質量函数

$$P(m|k,\alpha,\beta) = \binom{k+m-1}{m}\frac{B(\alpha+k,\beta+m)}{B(\alpha,\beta)}$$

で定義される離散分布を **ベータ負の二項分布** と呼び,

$$\mathrm{BetaNegativeBinomial}(k,\alpha,\beta)$$

と書く. ベータ負の二項分布の確率質量函数は次のようにも表される:

$$P(m|k,\alpha,\beta) = \int_0^1 \binom{k+m-1}{m} p^k (1-p)^m \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp.$$

8.2 問題: ベータ負の二項分布の期待値と分散

$M \sim \mathrm{BetaNegativeBinomial}(k,\alpha,\beta)$ のとき, 次が成立することを示せ:

$$E[M] = \frac{k\beta}{\alpha-1}, \quad \mathrm{var}(M) = \frac{\alpha(\alpha+k-1)\beta(\alpha+\beta-1)}{(\alpha-1)^2(\alpha-2)}.$$

解答例: 負の二項分布に従う確率変数 $M \sim \mathrm{NegativeBinomial}(k,p)$ の函数 $f(M)$ の期待値を $E[M|k,p]$ と書くとき,

$$E[M(M-1)\cdots(M-i+1)|k,p] = k(k+1)\cdots(k+i-1)\frac{(1-p)^i}{p^i}$$

が成立するのであった. ベータ負の二項分布はパラメータ p がベータ分布 $\mathrm{Beta}(\alpha,\beta)$ に従っている場合の負の二項分布に等しいので, $M \sim \mathrm{BetaNegativeBinomial}(k,\alpha,\beta)$ のとき,

$$E[f(M)] = \int_0^1 E[f(M)|k,p]\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp$$

が成立している. これと上を合わせると以下を得る:

$$\begin{aligned}
&E[M(M-1)\cdots(M-i+1)] \\
&= k(k+1)\cdots(k+i-1)\frac{B(\alpha-i,\beta+i)}{B(\alpha,\beta)} \\
&= k(k+1)\cdots(k+i-1)\frac{\Gamma(\alpha-i)\Gamma(\beta+i)}{\Gamma(\alpha+\beta)}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \\
&= k(k+1)\cdots(k+i-1)\frac{\beta(\beta+1)\cdots(\beta+i-1)}{(\alpha-1)(\alpha-2)\cdots(\alpha-i)}.
\end{aligned}$$

特に,

$$E[M] = \frac{k\beta}{\alpha-1}, \quad E[M(M-1)] = \frac{k(k+1)\beta(\beta+1)}{(\alpha-1)(\alpha-2)}.$$

ゆえに,

$$\begin{aligned}
\mathrm{var}(M) &= E[M(M-1)] + E[M] - E[M]^2 \\
&= \frac{k(k+1)\beta(\beta+1)}{(\alpha-1)(\alpha-2)} + \frac{k\beta}{\alpha-1} - \frac{k^2\beta^2}{(\alpha-1)^2} \\
&= \frac{k\beta}{(\alpha-1)^2(\alpha-2)}((k+1)(\beta+1)(\alpha-1) + (\alpha-1)(\alpha-2) - k\beta(\alpha-1)) \\
&= \frac{k\beta}{(\alpha-1)^2(\alpha-2)}((\alpha-1)^2 + (k+\beta)(\alpha-1) + k\beta) \\
&= \frac{k\beta}{(\alpha-1)^2(\alpha-2)}(\alpha-1+k)(\alpha-1+\beta) = \frac{k(k+\alpha-1)\beta(\alpha+\beta-1)}{(\alpha-1)^2(\alpha-2)}.
\end{aligned}$$

解答終

In [87]:
```
@vars k α β
var"E[M]" = k*β/(α-1)
var"E[M(M-1)]" = k*(k+1)*β*(β+1)/((α-1)*(α-2))
var"var(M)" = var"E[M(M-1)]" + var"E[M]" - var"E[M]"^2 ▷ factor
```

Out[87]: $$\frac{k\beta\,(k+\alpha-1)\,(\alpha+\beta-1)}{(\alpha-2)\,(\alpha-1)^2}$$

8.3 負の二項分布とベータ負の二項分布の期待値と分散の比較

負の二項分布と期待値と分散を比較しよう. $M' \sim \mathrm{NegativeBinomial}(k,p)$ のとき,

$$E[M'] = \frac{k(1-p)}{p}, \quad \mathrm{var}(M') = \frac{k(1-p)}{p^2}.$$

これに,

$$p = \frac{\alpha - 1}{\alpha + \beta - 1}, \quad 1 - p = \frac{\beta}{\alpha + \beta - 1}$$

を代入すると,

$$E[M'] = \frac{k\beta}{\alpha - 1} = E[M], \quad \mathrm{var}(M') = \frac{k\beta(\alpha + \beta - 1)}{(\alpha - 1)^2}.$$

ゆえに, $k > 0$ より,

$$E[M] = \frac{\alpha + k - 1}{\alpha - 2} E[M'] > \frac{\alpha - 1}{\alpha - 2} E[M'] > E[M'].$$

ベータ負の二項分布の期待値と負の二項分布の期待値を上のようにして合わせると, ベータ負の二項分布の分散は負の二項分布の分散よりも大きくなる.

8.4 負の二項分布の連続時間極限の復習

任意に実数 b を取り,

$$b + m = Lt, \quad p = \frac{\lambda}{L}$$

とおいて, $L \to \infty$ とすると,

$$\begin{aligned}
\binom{k+m-1}{m} p^k (1-p)^m &= \frac{1}{(k+m)B(k, m+1)} p^k (1-p)^m \, dm \\
&= \frac{1}{(Lt + k - b)\, B(k, Lt - b + 1)} \frac{\lambda^k}{L^k} \left(1 - \frac{\lambda}{L}\right)^{Lt-b} L \, dt \\
&= \frac{Lt}{Lt + k - b} \frac{(Lt^{k-1})}{(Lt)^k B(k, Lt - b + 1)} \frac{\lambda^k}{L^{k-1}} \left(1 - \frac{\lambda}{L}\right)^{Lt-b} dt \\
&= \frac{Lt}{Lt - b} \frac{\lambda^k}{(Lt)^k B(k, Lt - b + 1)} \left(1 - \frac{\lambda}{L}\right)^{Lt-b} t^{k-1} \, dt \\
&\to \frac{\lambda^k}{\Gamma(k)} e^{-\lambda t} t^{k-1} \, dt.
\end{aligned}$$

すなわち, $L \to \infty$ のとき,

$$\frac{b + \mathrm{NegativeBinomial}(k, \lambda/L)}{L} \to \mathrm{Gamma}(k, 1/\lambda).$$

注意: $\mathrm{NegativeBinomial}(k, p)$ の期待値は $k(1-p)/p$ なので, $k + \mathrm{NegativeBinomial}(k, p)$ の期待値は k/p になる. 一方, $\mathrm{Gamma}(k, 1/\lambda)$ の期待値は k/λ である. ゆえに $(k + \mathrm{NegativeBinomial}(k, \lambda/L))/L$ の期待値は $\mathrm{Gamma}(k, 1/\lambda)$ の期待値は一致する. だから, 上の連続時間極限では $b = k$ とすると収束先と期待値を一致させたままで極限を取ることができる.

注意: $\mathrm{NegativeBinomial}(k, p)$ の分散は $k(1-p)/p^2$ なので, $(k + \mathrm{NegativeBinomial}(k, \lambda/L))/L$ の分散は $(k/\lambda^2)(1 - \lambda/L)$ になる. 一方, $\mathrm{Gamma}(k, 1/\lambda)$ の分散は k/λ^2 になるので, 上の連続極限の過程は分散が単調増加しながらの収束になっている.

In [88]:
```
k, λ, L = 3, 4, 40
@show mean((k + NegativeBinomial(k, λ/L))/L), mean(Gamma(k, 1/λ))
@show var((k + NegativeBinomial(k, λ/L))/L), var(Gamma(k, 1/λ));
```

```
(mean((k + NegativeBinomial(k, λ / L)) / L), mean(Gamma(k, 1 / λ))) = (0.75, 0.75)
(var((k + NegativeBinomial(k, λ / L)) / L), var(Gamma(k, 1 / λ))) = (0.16875, 0.1875)
```

8.5 ベータ分布の極限の復習

$$p = \frac{\lambda}{L}, \quad \beta = \frac{L}{\theta}$$

とおいて, $L \to \infty$ とすると,

$$
\begin{aligned}
\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}\,dp &= \frac{L^\alpha}{\theta^\alpha (L/\theta)^\alpha B(\alpha, L/\theta)}\frac{\lambda^{\alpha-1}}{L^{\alpha-1}}\left(1-\frac{\lambda}{L}\right)^{L/\theta-1}\frac{d\lambda}{L} \\
&= \frac{1}{\theta^\alpha (L/\theta)^\alpha B(\alpha, L/\theta)}\left(1-\frac{\lambda}{L}\right)^{L/\theta-1}\lambda^{\alpha-1}\,d\lambda \\
&\to \frac{1}{\theta^\alpha\Gamma(\alpha)}e^{-\lambda/\theta}\lambda^{\alpha-1}\,d\lambda.
\end{aligned}
$$

すなわち, $L \to \infty$ のとき,

$$
L\,\mathrm{Beta}(\alpha, L/\theta) \to \mathrm{Gamma}(\alpha,\theta).
$$

注意: $\mathrm{Beta}(\alpha,\beta)$ の期待値は $\alpha/(\alpha+\beta)$ なので $L\,\mathrm{Beta}(\alpha, L/\theta)$ の期待値は $\alpha\theta/(1+\alpha\theta/L)$ になる. これは $L\to\infty$ で $\mathrm{Gamma}(\alpha,\theta)$ の期待値 $\alpha\theta$ に単調増加しながら収束する.

注意: $\mathrm{Beta}(\alpha,\beta)$ の分散は $\alpha\beta/((\alpha+\beta)^2(\alpha+\beta+1))$ なので $L\,\mathrm{Beta}(\alpha, L/\theta)$ の分散は $\alpha\theta^2/((1+\alpha\theta/L)^2(1+(\alpha+1)\theta/L))$ になる. これは $L\to\infty$ で $\mathrm{Gamma}(\alpha,\theta)$ の分散 $\alpha\theta^2$ に単調増加しながら収束する.

In [89]:
```
α, θ, L = 3, 4, 2^10
@show mean(L*Beta(α, L/θ)), mean(Gamma(α, θ))
@show var(L*Beta(α, L/θ)), var(Gamma(α, θ));
```

```
(mean(L * Beta(α, L / θ)), mean(Gamma(α, θ))) = (11.86100386100386, 12.0)
(var(L * Beta(α, L / θ)), var(Gamma(α, θ))) = (46.17301746568156, 48.0)
```

8.6 ベータ負の二項分布の連続極限

任意に実数 b を取り,

$$
b+m = Lt, \quad p = \frac{\lambda}{L}, \quad \beta = \frac{L}{\theta}
$$

とおく. このとき,

$$
\beta + m = \frac{L}{\theta} + Lt - b = L\left(\frac{1}{\theta}+t\right) - b = L\frac{1+\theta t}{\theta} - b
$$

直接的に極限を取ると,

$$
\begin{aligned}
&\binom{k+m-1}{m}\frac{B(\alpha+k,\beta+m)}{B(\alpha,\beta)}\,dm \\
&= \frac{1}{(k+m)B(k,m+1)}\frac{B(\alpha+k,\beta+m)}{B(\alpha,\beta)}\,dm \\
&= \frac{1}{(Lt-b+k)B(k,Lt-b+1)}\frac{B(\alpha+k, L/\theta+Lt-b)}{B(\alpha,L/\theta)}\,dm \\
&= \frac{Lt}{Lt-b+k}\frac{(Lt)^{k-1}}{(Lt)^k B(k,Lt-b+1)}\frac{(L(1+\theta t)/\theta)^{\alpha+k}B(\alpha+k, L(1+\theta t)/\theta - b)}{(L/\theta)^\alpha B(\alpha, L/\theta)}\frac{(L/\theta)^\alpha}{(L(1+\theta t)/\theta)^{\alpha+k}}L\,dt \\
&= \frac{Lt}{Lt-b+k}\frac{(L(1+\theta t)/\theta)^{\alpha+k}B(\alpha+k, L(1+\theta t)/\theta - b)}{(Lt)^k B(k,Lt-b+1)(L/\theta)^\alpha B(\alpha,L/\theta)}\frac{(\theta t)^k}{(1+\theta t)^{k+\alpha}}\frac{dt}{t} \\
&\to \frac{\Gamma(\alpha+k)}{\Gamma(k)\Gamma(\alpha)}\frac{(\theta t)^k}{(1+\theta t)^{k+\alpha}}\frac{dt}{t} = \frac{1}{B(k,\alpha)}\frac{(\theta t)^k}{(1+\theta t)^{k+\alpha}}\frac{dt}{t}.
\end{aligned}
$$

これは, 負の二項分布に従う確率変数

$$
M \sim \mathrm{BetaNegativeBinomial}(k,\alpha,L/\theta)
$$

から定まる確率変数

$$
T = \frac{b+M}{L}
$$

が $L\to\infty$ の極限で

$$
\theta T \sim \mathrm{BetaPrime}(k,\alpha), \quad \text{i.e.} \quad T \sim \frac{1}{\theta}\mathrm{BetaPrime}(k,\alpha)
$$

を満たすことを意味する. すなわち, $L\to\infty$ のとき,

$$\frac{b + \mathrm{BetaNegativeBinomial}(k, \alpha, L/\theta)}{L} \to \frac{1}{\theta}\mathrm{BetaPrime}(k, \alpha).$$

負の二項分布とベータ分布の極限を取った後に積分すると,

$$\begin{aligned}
&\int_0^\infty \frac{\lambda^k}{\Gamma(k)} e^{-\lambda t} t^{k-1} \frac{1}{\theta^\alpha \Gamma(\alpha)} e^{-\lambda/\theta} \lambda^{\alpha-1}\, d\lambda \\
&= \frac{t^{k-1}}{\Gamma(k)\theta^\alpha\Gamma(\alpha)} \int_0^\infty e^{-\lambda(1+\theta t)/\theta} \lambda^{\alpha+k-1}\, d\lambda \\
&= \frac{t^{k-1}}{\Gamma(k)\theta^\alpha\Gamma(\alpha)} \frac{\theta^{\alpha+k}\Gamma(\alpha+k)}{(1+\theta t)^{\alpha+k}} = \frac{1}{B(k,\alpha)} \frac{(\theta t)^k}{(1+\theta t)^{k+\alpha}} \frac{dt}{t}.
\end{aligned}$$

当然だが, 上と同じ結果が得られる.

注意: $\mathrm{BetaNegativeBinomial}(k, \alpha, \beta)$ の期待値は $k\beta/(\alpha-1)$ なので, $\mathrm{BetaNegativeBinomial}(k, \alpha, L/\theta)/L$ の期待値は $k/(\theta(\alpha-1))$ になる. 一方, $\mathrm{BetaPrime}(k, \alpha)$ の期待値は $k/(\alpha-1)$ なので $\mathrm{BetaPrime}(k, \alpha)/\theta$ の期待値も $k/(\theta(\alpha-1))$ になる. ゆえに $b = 0$ とすれば, 上の連続時間極限を期待値を収束先に一致させた状態での極限にできる.

注意: $\mathrm{BetaNegativeBinomial}(k, \alpha, \beta)$ の分散は $k(\alpha+k-1)\beta(\alpha+\beta-1)/((\alpha-1)^2(\alpha-2))$ なので, $\mathrm{BetaNegativeBinomial}(k, \alpha, L/\theta)/L$ の分散は

$$\frac{k(k+\alpha-1)}{\theta^2(\alpha-1)^2(\alpha-2)}\left(1 + \frac{(\alpha-1)\theta}{L}\right)$$

になる. 一方, $\mathrm{BetaPrime}(k, \alpha)/\theta$ の分散は

$$\frac{k(k+\alpha-1)}{\theta^2(\alpha-1)^2(\alpha-2)}$$

である. ゆえに上の連続時間極限は分散が収束先の値に単調減少しながら収束する.

In [90]:

```
mean_betanegbin(k, α, β) = k*β/(α - 1)
var_betanegbin(k, α, β) = k*(α+k-1)*β*(α+β-1)/((α-1)^2*(α-2))
std_betanegbin(k, α, β) = √var_betanegbin(k, α, β)

k, α, θ, L = 3, 4, 2, 100
@show mean_betanegbin(k, α, L/θ)/L, mean(BetaPrime(k, α)/θ)
@show var_betanegbin(k, α, L/θ)/L^2, var(BetaPrime(k, α)/θ);
```

```
(mean_betanegbin(k, α, L / θ) / L, mean(BetaPrime(k, α) / θ)) = (0.5, 0.5)
(var_betanegbin(k, α, L / θ) / L ^ 2, var(BetaPrime(k, α) / θ)) = (0.265, 0.25)
```

In [91]:

```
function pdf_betanegbin(x, k, α, β)
    m = round(x)
    m < 0 && return 0.0
    exp(loggamma(k+m) - loggamma(m+1) - loggamma(k) +
        logbeta(α+k, β+m) - logbeta(α, β))
end

@show k, α, β = 3, 8, 12
@show mmax = 10^6
@show sum(pdf_betanegbin(m, k, α, β) for m in 0:mmax)
@show sum(m*pdf_betanegbin(m, k, α, β) for m in 0:mmax)
@show μ = mean_betanegbin(k, α, β)
@show sum((m - μ)^2*pdf_betanegbin(m, k, α, β) for m in 0:mmax)
@show var_betanegbin(k, α, β)
@show √sum((m - μ)^2*pdf_betanegbin(m, k, α, β) for m in 0:mmax)
@show σ = std_betanegbin(k, α, β)
plot(x -> pdf_betanegbin(x, k, α, β), -1, μ+4σ; label="")
title!("BetaNegBin(k=$k, α=$α, β=$β)")
```

```
(k, α, β) = (3, 8, 12) = (3, 8, 12)
mmax = 10 ^ 6 = 1000000
sum((pdf_betanegbin(m, k, α, β) for m = 0:mmax)) = 0.9999999999999972
sum((m * pdf_betanegbin(m, k, α, β) for m = 0:mmax)) = 5.142857142857044
μ = mean_betanegbin(k, α, β) = 5.142857142857143
sum(((m - μ) ^ 2 * pdf_betanegbin(m, k, α, β) for m = 0:mmax)) = 23.265306122447598
var_betanegbin(k, α, β) = 23.26530612244898
√(sum(((m - μ) ^ 2 * pdf_betanegbin(m, k, α, β) for m = 0:mmax))) = 4.823412290323895
σ = std_betanegbin(k, α, β) = 4.823412290324038
```

Out[91]:

In [92]:

```
function plot_bnb_bp(k, α, θ, L;
        bp = BetaPrime(k, α)/θ, μ_bp = mean(bp), σ_bp = std(bp),
        tlim = (max(-1/L, μ_bp - 4σ_bp), μ_bp + 4σ_bp))
    β = L/θ
    μ_bnb, σ_bnb = mean_betanegbin(k, α, β), std_betanegbin(k, α, β)
    @show μ_bnb/L, μ_bp
    @show σ_bnb/L, σ_bp
    plot(t -> pdf_betanegbin(L*t, k, α, β)*L, tlim...;
        label="BetaNegBin(k,α,L/θ)/L")
    plot!(bp, tlim...; label="BetaPrime(k,α)/θ")
    title!("k=$k, α=$α, θ=$θ, L=$L")
end
```

Out[92]: plot_bnb_bp (generic function with 1 method)

In [93]:
```
plot(plot_bnb_bp.(2, 3, 10, (10, 30, 100, 300); tlim=(-0.1, 0.7))...;
    size=(800, 500), layout=(2, 2))
```

```
(μ_bnb / L, μ_bp) = (0.1, 0.1)
(σ_bnb / L, σ_bp) = (0.2449489742783178, 0.14142135623730953)
(μ_bnb / L, μ_bp) = (0.1, 0.1)
(σ_bnb / L, σ_bp) = (0.18257418583505539, 0.14142135623730953)
(μ_bnb / L, μ_bp) = (0.1, 0.1)
(σ_bnb / L, σ_bp) = (0.15491933384829668, 0.14142135623730953)
(μ_bnb / L, μ_bp) = (0.1, 0.1)
(σ_bnb / L, σ_bp) = (0.1460593486680443, 0.14142135623730953)
```

Out[93]:

8.7 ベータ負の二項分布の解釈 (Pólyaの壺)

$k = 1, 2, 3, \ldots$ であると仮定する.

このとき, ベータ負の二項分布の確率質量函数は次のように書き直される:

$$P(m|k, \alpha, \beta) = \binom{k+m-1}{m} \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)\cdot\beta(\beta+1)\cdots(\beta+m-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+k+m-1)}.$$

次のように当たりとはずれが出るくじを引き続ける状況を考える. 当たりが出る確率 p は2つの正の値のパラメータ達 (α, β) から $p = \alpha/(\alpha+\beta)$ で決まっている. そして, くじを引くごとにそれらのパラメータの値は次のように変化して行く:

(1) 当たりが出たら, パラメータの値を $(\alpha, \beta) \mapsto (\alpha+1, \beta)$ と更新する. (これによって次に当たりが出る確率は $(\alpha+1)/(\alpha+\beta+1)$ に上昇する.

(2) はずれが出たら, パラメータの値を $(\alpha, \beta) \mapsto (\alpha, \beta+1)$ と更新する. (これによって次にはずれが出る確率は $(\beta+1)/(\alpha+\beta+1)$ に上昇する.

上の式を見ると, ベータ負の二項分布はこのようなくじを引き続けて, 当たりが k 回出るまでに出たはずれの回数 m の分布になっていることがわかる. ちょうどこれの二項分布の場合の類似がベータ二項分布の解釈になることはすでに説明した.

注意: 上のくじの設定は次のようにも解釈される. 壺の中に当たりとはずれの玉がそれぞれ α 個, β 個入っており, 壺から無作為に当たりの玉が取り出されたら壺に当たりの玉を2個戻し, はずれの玉が取り出されたら壺にはずれの玉を2個戻す. これを繰り返す. この設定の壺を **Pólyaの壺** (Pólya's urn) と呼んだりする. この設定では α, β は整数でなければいけないが, 非整数の場合にもそれらが正の値であれば離散確率分布としてwell-definedになる.

8.8 Pólyaの壺の確率分布の定義

$\alpha, \beta > 0$ であると仮定する. 試行回数 n の **Pólyaの壺の確率分布** は次の確率質量函数によって定義される: $x_i = 1, 0$ でかつ $x_1 + \cdots + x_n = k$ のとき,

$$P(x_1, \dots, x_n | n, \alpha, \beta) = \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)\cdot\beta(\beta+1)\cdots(\beta+n-k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+n-1)}.$$

1 は当たりを 0 ははずれを意味し, $x_1 + \cdots + x_n = k$ は x_i 達の中に含まれる 1 の個数(当たりの個数)を意味する. 試行回数 n のPólyaの壺の確率分布を

$$\mathrm{Pólya}(n, \alpha, \beta)$$

と書くことにする.

Pólyaの壺での確率が実際に上のように表されることは以下のように考えればわかる.

例えば, $n = 5$ のPólyaの壺の出目が $(1, 0, 1, 1, 0)$ になる確率は

$$\frac{\alpha}{\alpha+\beta}\frac{\beta}{\alpha+\beta+1}\frac{\alpha+1}{\alpha+\beta+2}\frac{\alpha+2}{\alpha+\beta+3}\frac{\beta+1}{\alpha+\beta+4}$$

になり, 出目が $(0, 1, 1, 0, 1)$ になる確率は

$$\frac{\beta}{\alpha+\beta}\frac{\alpha}{\alpha+\beta+1}\frac{\alpha+1}{\alpha+\beta+2}\frac{\beta+1}{\alpha+\beta+3}\frac{\alpha+2}{\alpha+\beta+4}$$

になる. これらは共通の分子と分母を持つので互いに等しい.

この例から一般に n 回の試行で壺から当たりが出た回数が同じであれば, それらが出た順序に確率がよらず, 試行回数 n のPólyaの壺の確率分布の確率質量函数の定義通りの式で表されることがわかる.

Pólyaの壺の試行の $i+j-1$ 回目で i 回目の当たりが出たとき, 壺のパラメータ値は $(\alpha+i-1, \beta+j-1)$ の形をしており, 確率の計算では $(\alpha+i-1)/(\alpha+\beta+i+j-2)$ がかけられる. $i+j-1$ 回目で j 回目のはずれが出たとき, 壺のパラメータ値は $(\alpha+i-1, \beta+j-1)$ の形をしており, 確率の計算では $(\beta+j-1)/(\alpha+\beta+i+j-2)$ がかけられる. したがって, $x_1, \dots, x_n$ の中に当たりが k 個含まれているとき, n 回の試行で $(x_1, \dots, x_n)$ が生じる確率は, 分子に $\alpha+i-1$ ($i = 1, 2, \dots, k$) と $\beta+j-1$ ($j = 1, 2, \dots, n-k$)が並び, 分母に $\alpha+\beta+r-1$ ($r = 1, 2, \dots, n$) を並べてできる商の値になる.

試行回数 n のPólyaの壺の確率分布の確率質量函数は以下のように表される:

$$\begin{aligned}
P(x_1, \dots, x_n | n, \alpha, \beta) &= \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)\cdot\beta(\beta+1)\cdots(\beta+n-k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+n-1)} \\
&= \frac{\Gamma(\alpha+k)}{\Gamma(\alpha)}\frac{\Gamma(\beta+n-k)}{\Gamma(\beta)}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+n)} \\
&= \frac{B(\alpha+k, \beta+n-k)}{B(\alpha, \beta)} \\
&= \int_0^1 p^k(1-p)^{n-k}\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)}\,dp \\
&= \int_0^1 \prod_{i=1}^n \left(p^{x_i}(1-p)^{1-x_i}\right)\frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)}\,dp.
\end{aligned}$$

ここで $\prod_{i=1}^n \left(p^{x_i}(1-p)^{1-x_i}\right)$ は通常のBernoulli試行の確率質量函数なので, Pólyaの壺の確率分布に従う試行は成功確率 p がベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従って確率的に揺らいでいるようなBernoulli試行になっていることがわかった.

注意: この結果は, Pólyaの壺の確率分布が共役事前分布 $p \sim \mathrm{Beta}(\alpha, \beta)$ に関するBernoulli試行 $\mathrm{Bernoulli}(p)^n$ のBayes版の統計モデルそのものになっていることを意味している. Bernoulli試行と一様乱数生成の繰り返しとベータ分布とPólyaの壺の関係については以下のリンク先も参照せよ:

- https://twitter.com/genkuroki/status/1529044220887310336 (https://twitter.com/genkuroki/status/1529044220887310336)

In [ ]:

```
1
```

標本分布について

- 黒木玄

- 2022-04-11〜2022-07-13

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

このノートはかなり長大なものになったが, 内容的には資料として役に立てるための視覚化の部分が多く, 実質的な分量はそう多くはない. 他であまり見ないグラフも載せたつもりなので, 全体に目を通して頂ければとてもうれしい.

目次

- 1 標本分布とその使い方
  - 1.1 同時確率質量函数と同時確率密度函数
  - 1.2 確率変数の独立性の定義
    - 1.2.1 独立な確率変数達の同時確率質量函数
    - 1.2.2 独立な確率変数達の同時確率密度函数
    - 1.2.3 独立性に関する大雑把なまとめ
    - 1.2.4 分散を0に近付けたときの正規分布について
  - 1.3 独立同分布(i.i.d., iid)の定義
  - 1.4 標本分布の定義
  - 1.5 確率分布の積とn乗の定義
  - 1.6 重要: 分布Dの標本分布の主な使用用途 (統計モデル, 統計量)
  - 1.7 試行回数nのBernoulli試行の分布はBernoulli分布の標本分布
  - 1.8 二項分布による推定の確率的揺らぎの記述
  - 1.9 必修問題: 大阪都構想に関する住民投票の結果について
    - 1.9.1 WolframAlphaによる計算の例
    - 1.9.2 Julia言語による計算の例
    - 1.9.3 Clopper-Pearsonの信頼区間とそれを与えるP値
    - 1.9.4 信頼区間よりも情報量が大きなP値函数のプロット
    - 1.9.5 Sterneの信頼区間とそれを与えるP値函数
    - 1.9.6 Sterneの信頼区間を与えるP値函数の実装例
  - 1.10 対ごとに独立であっても全体が独立であるとは限らない
  - 1.11 確率変数の独立性の現実における解釈に関する重大な注意
- 2 確率変数達の共分散と相関係数と無相関性
  - 2.1 確率変数達の共分散と相関係数の定義
  - 2.2 確率変数達の対ごとの無相関性の定義
  - 2.3 問題: 確率変数の相関係数の計算例
  - 2.4 問題: Cauchy-Schwarzの不等式
  - 2.5 問題: 等確率有限離散分布の相関係数と cos θ の関係
  - 2.6 問題: 相関係数の計算
  - 2.7 問題: 共分散が0に近くても相関係数が0から遠い場合がある
  - 2.8 問題: 独立ならば無相関である (実質1行で解ける)
  - 2.9 問題: 無相関でも独立とは限らない
  - 2.10 問題: 対ごとに無相関な確率変数達の和の分散はそれぞれの分散の和になる
  - 2.11 問題(メタアナリシスの出発点): 共通の期待値と異なる分散を持つ確率変数の荷重平均
  - 2.12 問題: 二項分布と負の二項分布の平均と分散のBernoulli分布と幾何分布の場合への帰着
  - 2.13 問題: 番号が異なる確率変数達が対ごとに無相関なときの確率変数の和の共分散
  - 2.14 分散共分散行列とその半正定値性
- 3 標本(サンプル, データ)の平均と分散と共分散と相関係数
  - 3.1 標本平均の定義

- 3.2 問題: 対ごとに無相関な確率変数達の標本平均の分散
- 3.3 標本分散と不偏分散の定義
- 3.4 不偏推定量について: 不偏分散の定義ではどうして n ではなく n-1 で割るか
- 3.5 データの共分散の定義
- 3.6 問題: 標本平均達の共分散
- 3.7 問題: 不偏共分散の定義ではどうして n ではなく n-1 で割るか
- 3.8 データの相関係数の定義 (以上の定義のまとめにもなっている)
- 4 最小二乗法による線形回帰
  - 4.1 問題: 最小二乗法による線形回帰の公式の導出
    - 4.1.1 解答例1: 単回帰の公式の素朴な導出
    - 4.1.2 解答例2: データの5つの要約値だけで平均二乗残差を表示
    - 4.1.3 コンピュータによる最小二乗法の計算例
  - 4.2 必修問題: 最小二乗法の計算例 (Anscombe's quartet)
    - 4.2.1 WolframAlphaでAnscombeの例1を扱う方法
    - 4.2.2 Julia言語でAnscombeの例を扱う方法
  - 4.3 Galton (1886)が描いた2つの回帰直線
    - 4.3.1 Galtonのグラフの再現
    - 4.3.2 グラフ中の楕円と2変量正規分布モデルや回帰直線の関係
    - 4.3.3 楕円と回帰直線の関係の視覚化
- 5 モーメントとキュムラントと歪度と尖度
  - 5.1 モーメントとその母函数と特性函数とキュムラント母函数の定義
  - 5.2 特性函数による期待値の表示
  - 5.3 問題: 分布のアフィン変換のキュムラント
  - 5.4 問題: 標準正規分布のモーメント母函数と特性函数とキュムラント母函数
  - 5.5 確率変数の標準化と標準化キュムラントと歪度と尖度
  - 5.6 問題: 正規分布の歪度と尖度
  - 5.7 問題: 一様分布のキュムラント母函数と歪度と尖度
  - 5.8 問題: 独立な確率変数達の和のモーメント母函数と特性函数とキュムラント母函数
  - 5.9 問題: ガンマ分布のキュムラント母函数と歪度と尖度
  - 5.10 問題: 二項分布のキュムラント母函数と歪度と尖度
- 6 独立同分布な確率変数達の不偏分散の分散
  - 6.1 期待値0, 分散1の場合への帰着
  - 6.2 標本平均と不偏分散の共分散の計算
  - 6.3 不偏分散の分散の計算
  - 6.4 歪度と尖度に関する不等式
  - 6.5 問題: 不偏分散と不偏補正されていない標本分散の平均二乗誤差の比較
- 7 標本分布における標本平均と不偏分散の同時分布の視覚化
  - 7.1 正規分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.2 一様分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.3 単峰型のガンマ分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.4 指数分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.5 非対称なベータ分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.6 非対称な2つ山の混合正規分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.7 対称なBernoulli分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.8 非対称なBernoulli分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.9 Poisson分布の標本分布における $\bar{X}$, S^2 の同時分布
  - 7.10 対数正規分布の標本分布における $\bar{X}$, S^2 の同時分布
- 8 正規分布の標本分布の場合
  - 8.1 準備: $\sqrt{n}\,\bar{x}$ を含む正規直交座標系
  - 8.2 準備: n - 2 次元単位球面上の微小面積要素
  - 8.3 正規分布の標本分布における $\bar{X}$, S^2 の同時分布の計算
  - 8.4 正規分布の標本分布から自然にt分布に従う確率変数が得られること
  - 8.5 正規分布モデルの使用に関する重要な注意: t 分布の使いどころ
  - 8.6 必修問題: データの t 値の計算例
    - 8.6.1 WolframAlphaによる t 値の計算例
    - 8.6.2 Julia言語による t 値の計算例
    - 8.6.3 問題用の擬似データの生成の仕方
- 9 標本分布における T の分布の視覚化
  - 9.1 正規分布の標本分布における T の分布
  - 9.2 一様分布の標本分布における T の分布
  - 9.3 単峰型のガンマ分布の標本分布における T の分布
  - 9.4 指数分布の標本分布における T の分布
  - 9.5 非対称なベータ分布の標本分布における T の分布
  - 9.6 非対称な2つ山の混合正規分布の標本分布における T の分布
  - 9.7 対称なBernoulli分布の標本分布における T の分布
  - 9.8 非対称なBernoulli分布の標本分布における T の分布

9.9 Poisson分布の標本分布における T の分布
9.10 対数正規分布の標本分布における T の分布

In [1]:
```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using LinearAlgebra
using Optim
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:
```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

1 標本分布とその使い方

1.1 同時確率質量函数と同時確率密度函数

確率変数達 $X_1, \ldots, X_n$ が同時確率質量函数 $P(x_1, \ldots, x_n)$ を持つとは, $P(x_1, \ldots, x_n) \geq 0$ でかつ, $P(x_1, \ldots, x_n)$ の総和が 1 であり,

$$E[f(X_1, \ldots, X_n)] = \sum_{x_1} \cdots \sum_{x_n} f(x_1, \ldots, x_n) P(x_1, \ldots, x_n)$$

が成立することである. このとき, X_i 単独の確率質量函数 $P(x_i)$ は

$$P(x_i) = \sum_{x_1} \cdots \widehat{\sum_{x_i}} \cdots \sum_{x_n} P(x_1, \cdots, x_i, \ldots, x_n)$$

になる. ここで, $\widehat{\sum_{x_i}}$ は $\sum_{x_i}$ を除くという意味である. なぜならば,

$$\begin{aligned} E[f(X_i)] &= \sum_{x_1} \cdots \sum_{x_i} \cdots \sum_{x_n} f(x_i) P(x_1, \cdots, x_i, \ldots, x_n) \\ &= \sum_{x_i} f(x_1) \left(\sum_{x_1} \cdots \widehat{\sum_{x_i}} \cdots \sum_{x_n} P(x_1, \cdots, x_i, \ldots, x_n) \right). \end{aligned}$$

注意: $i \neq j$ のとき $P(x_i)$ と $P(x_j)$ は一般に異なる確率質量函数になることに注意せよ. 区別を明瞭にするためには $P(x_i)$ を $P_i(x_i)$ のように書くべきであるが, 統計学の世界での慣習に従ってこのような記号法上の省略を行った. これが気に入らない人は自分でノートを作るときに, 区別されるべきものに異なる記号を割り振るようにすればよいだろう. 以下についても同様である.

確率変数達 $X_1, \ldots, X_n$ が同時確率密度函数 $p(x_1, \ldots, x_n)$ を持つとは, $p(x_1, \ldots, x_n) \geq 0$ でかつ, $p(x_1, \ldots, x_n)$ の積分が 1 になり,

$$E[f(X_1,\ldots,X_n)] = \int\cdots\int f(x_1,\ldots,x_n)p(x_1,\ldots,x_n)\,dx_1\cdots dx_n$$

が成立することである. ここで $\int$ は定積分を表す. このとき, X_i 単独の確率密度函数 $p(x_i)$ は

$$p(x_i) = \int\cdots\widehat{\int}\cdots\int p(x_1,\cdots,x_i,\ldots,x_n)\,dx_1\cdots\widehat{dx_i}\cdots dx_n$$

になる. ここで, $\widehat{\int}$, $\widehat{dx_i}$ はそれらを除くことを意味する. なぜならば,

$$\begin{aligned}E[f(X_i)] &= \int\cdots\int\cdots\int f(x_i)p(x_1,\cdots,x_i,\ldots,x_n)\,dx_1\cdots dx_i\cdots dx_n \\ &= \int f(x_i)\left(\int\cdots\widehat{\int}\cdots\int p(x_1,\cdots,x_i,\ldots,x_n)\,dx_1\cdots\widehat{dx_i}\cdots dx_n\right)dx_i.\end{aligned}$$

1.2 確率変数の独立性の定義

確率変数達 $X_1,\ldots,X_n$ が与えられており, それらの函数の期待値 $E[f(X_1,\ldots,X_n)]$ が定義されているとする. このとき, $X_1,\ldots,X_n$ が **独立** (independent)であるとは,

$$E[f_1(X_1)\cdots f_n(X_n)] = E[f_1(X_1)]\cdots E[f_n(X_n)] \tag{1}$$

のように X_i ごと函数達 $f_i(X_i)$ の積の期待値が各々の $f_i(X_i)$ の期待値の積になることだと定める.

注意: 厳密には函数 f_i 達を動かす範囲を確定させる必要があるが, このノートではそのようなことにこだわらずに解説することにする. 厳密な理論の展開のためには測度論的確率論の知識が必要になるが, そのような方向はこのノートの目標とは異なる. 測度論的確率論に興味がある人は別の文献を参照して欲しい. ただし, 測度論的確率論の理解と統計学の理解は別の話題になってしまうことには注意して欲しい. 測度論的確率論と統計学では興味の方向が異なる.

1.2.1 独立な確率変数達の同時確率質量函数

確率変数達 $X_1,\ldots,X_n$ が同時確率質量函数 $P(x_1,\ldots,x_n)$ を持つとき, X_i 単独の確率質量函数を $P_i(x_i)$ と書くならば, 確率変数達 $X_1,\ldots,X_n$ が独立であることと,

$$P(x_1,\ldots,x_n) = P_1(x_1)\cdots P_n(x_n) \tag{2}$$

が成立することは同値である. すなわち, $(X_1,\ldots,X_n)$ の函数の期待値が次のように表されることと同値である:

$$E[f(X_1,\ldots,X_n)] = \sum_{x1}\cdots\sum_{x_n} f(x_1,\ldots,x_n)P_1(x_1)\cdots P_n(x_n)$$

注意: ここでは気分を変えて, $P_i(x_i)$ をずぼらに $P(x_i)$ と書く流儀をやめてみた.

証明: (2)が成立しているならば,

$$\begin{aligned}E[f_1(X_1)\cdots f_n(X_n)] &= \sum_{x1}\cdots\sum_{x_n} f_1(x_1)\cdots f_n(x_n)P_1(x_1)\cdots P_n(x_n) \\ &= \sum_{x1} f_1(x_1)P_1(x_1)\cdots\sum_{x_n} f_n(x_n)P_n(x_1) \\ &= E[f_1(X_1)]\cdots E[f_n(X_n)]\end{aligned}$$

と(1)が成立する.

逆に(1)が成立しているならば, $f_i(x_i)$ として $x_i = a_i$ のときにのみ 1 でそれ以外のときに 0 になる函数を取ると,

$$P(a_1,\ldots,a_n) = E[f_1(X_1)\cdots f_n(X_n)] = E[f_1(X_1)]\cdots E[f_n(X_n)] = P_1(a_1)\cdots P_n(a_n)$$

なので, (2)が成立する.

1.2.2 独立な確率変数達の同時確率密度函数

確率変数達 $X_1,\ldots,X_n$ が同時確率密度函数 $p(x_1,\ldots,x_n)$ を持つとき, X_i 単独の確率密度函数を $p_i(x_i)$ と書くならば, 確率変数達 $X_1,\ldots,X_n$ が独立であることと,

$$p(x_1,\ldots,x_n) = p_1(x_1)\cdots p_n(x_n) \tag{3}$$

が成立することは同値である. すなわち, $(X_1,\ldots,X_n)$ の函数の期待値が次のように表されることと同値である:

$$E[f(X_1,\dots,X_n)] = \int\cdots\int f(x_1,\dots,x_n)p_1(x_1)\cdots p_n(x_n)\,dx_1\cdots dx_n.$$

ここで $\int\cdots dx_i$ は適切な範囲での定積分を表す.

注意: ここでは気分を変えて, $p_i(x_i)$ をずぼらに $p(x_i)$ と書く流儀をやめてみた.

(雑な)証明: (3)が成立しているならば,

$$\begin{aligned}
E[f_1(X_1)\cdots f_n(X_n)] &= \int\cdots\int f_1(x_1)\cdots f_n(x_n)p_1(x_1)\cdots p_n(x_n)\,dx_1\cdots dx_n \\
&= \int f_1(x_1)p_1(x_1)\,dx_1\cdots\int f_n(x_n)p_n(x_n)\,dx_n \\
&= E[f_1(X_1)]\cdots E[f_n(X_n)]
\end{aligned}$$

と(1)が成立する.

簡単のため, 密度函数達 $p(x_1,\dots,p_n)$, $p_i(x_i)$ は連続函数になっていると仮定する(応用上は概ねこれで十分). このとき,

$$f_i(x_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\exp\left(-\frac{(x_i-a_i)^2}{2\sigma_i^2}\right)$$

とおくと, $\sigma_i \searrow 0$ のとき

$$\begin{aligned}
E[f_i(X_i)] &= \int f_i(x_i)p_i(x_i)\,dx_i \to p_i(a_i), \\
E[f_1(X_1)\cdots f_n(X_n)] &= \int\cdots\int f_1(x_1)\cdots f_n(x_n)p(x_1,\dots,x_n)\,dx_1\cdots dx_n \to p(a_1,\dots,a_n)
\end{aligned}$$

となるので, 逆に(1)が成立しているならば

$$p(a_1,\dots,a_n) = p_1(a_1)\cdots p_n(a_n)$$

と(3)が成立する.

確率密度函数と確率質量函数が混ざっている場合も同様である.

1.2.3 独立性に関する大雑把なまとめ

大雑把に誤解を恐れずに言うと, 確率が積になるときに独立だという.

実際の計算では, 積の期待値が期待値の積になるという形式で確率変数の独立性を使う.

In [3]:

```
# 以下で使う図の準備

var"正規分布は分散が小さくなると一点に集中して行く." = plot()
for σ in (1.0, 0.5, 0.25, 0.1)
    plot!(Normal(0, σ); label="σ = $σ")
end
title!("Normal(μ=0, σ)");
```

1.2.4 分散を0に近付けたときの正規分布について

X_σ を平均 μ, 分散 σ^2 の正規分布に従う確率変数とすると, 有界な連続函数 $f(x)$ について

$$\lim_{\sigma\searrow 0} E[f(X_\sigma)] = f(\mu)$$

となる. 証明が知りたい人は次のリンク先の「総和核」に関する解説(特にGauss核の説明)を参照せよ:

- [12 Fourier解析 - 総和核](https://nbviewer.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb#%E7%B7%8F%E5%92%8C%E6%

大雑把に言えば, これは $\sigma \searrow 0$ の極限で正規分布が一点 $x=\mu$ に集中することを意味している.

In [4]:
```
var"正規分布は分散が小さくなると一点に集中して行く."
```

Out[4]:

$E[f(X_0)] = f(\mu)$ を満たす確率変数 X_0 は実質定数 μ に等しく, 実質定数に等しい確率変数が従う分布を **Dirac分布** と呼ぶことがある.

1.3 独立同分布(i.i.d., iid)の定義

確率変数達 $X_1, \ldots, X_n$ が **独立同分布** (independent and identically distributed, **i.i.d.**, **iid**) であるとは, それらが独立でかつ X_i 達が従う分布が互いにすべて等しいことであると定める.

1.4 標本分布の定義

n 個の確率変数達 $X_1, \ldots, X_n$ は独立同分布であり, 各 X_i は共通の確率分布 D に従うと仮定する:

$$X_1, \ldots, X_n \sim D \quad (\text{independent})$$

このとき, $X_1, \ldots, X_n$ を分布 D の **標本** または **サンプル** (sample)と呼び, $X_1, \ldots, X_n$ の同時確率分布を **分布 D のサイズ n の標本分布** (distribution of samples of size n from distribution D)と呼ぶことにする.

注意: 独立同分布な確率変数達 $X_1, \ldots, X_n$ (確率変数は数ではなく函数であった)そのものではなく, それらの実現値 $x_1, \ldots, x_n$ (函数としての確率変数の値達)を標本もしくはサンプルと呼ぶことがある. 1つひとつの値 x_i をサンプルと呼ぶのではなく, 数の列 $x_1, \ldots, x_n$ をサンプルと呼ぶことに注意せよ. その辺が紛らわしい場合には数の列 $x_1, \ldots, x_n$ を **データ** (data)と呼ぶことがある. $X_1, \ldots, X_n$ もデータと呼ぶことがある. この辺の用語の使い方はかなりイーカゲンになり易いので注意して欲しい.

確率変数 X_i 達が同一の確率密度函数 $p(x_i)$ を持つとき, $X_1, \ldots, X_n$ の同時確率密度函数が

$$p(x_1, \ldots, x_n) = p(x_1) \cdots p(x_n)$$

であることと, $X_1, \ldots, X_n$ の同時確率分布が確率密度函数 $p(x)$ が定める連続分布の標本分布であることは同値である.

確率変数 X_i 達が同一の確率質量函数 $P(x_i)$ を持つとき, $X_1, \ldots, X_n$ の同時確率質量函数が

$$P(x_1, \ldots, x_n) = P(x_1) \cdots P(x_n)$$

であることと, $X_1, \ldots, X_n$ の同時確率分布が確率質量函数 $P(x)$ が定める離散分布の標本分布であることは同値である.

1.5 確率分布の積とn乗の定義

$X_1, \ldots, X_n$ が独立な確率変数達であり, X_i が従う分布が D_i であるとき, $(X_1, \ldots, X_n)$ が従う分布を

$$D_1 \times \cdots \times D_n$$

と書き, 分布 $D_1, \ldots, D_n$ の **積** (product, 直積, direct product)と呼ぶ:

$$(X_1, \ldots, X_n) \sim D_1 \times \cdots \times D_n \iff X_i \sim D_i \text{ for } i = 1, \ldots, n \text{ and } X_1, \ldots, X_n \text{ are independent.}$$

例えば, 分布 D_i が確率密度函数 $p_i(x_i)$ を持つとき, 積分布 $D_1 \times \cdots \times D_n$ は確率密度函数

$$p(x_1, \ldots, x_n) = p_1(x_1) \cdots p(x_n)$$

によって定義される多変量連続分布になる. 確率密度函数が積の形になるので, 分布の積と呼ぶ.

D_i がすべて同一の分布 D であるとき, 積 $D_1 \times \cdots \times D_n$ を D^n と書き, 分布 D の **べき乗** (冪乗, power)もしくは **累乗** と呼ぶ. 前節で説明したように分布 D の n 乗 D^n は分布 D のサイズ n の標本分布と呼ばれる:

$$(X_1, \ldots, X_n) \sim D^n \iff (X_1, \ldots, X_n) \text{ is a sample of size } n \text{ from the distribution D.}$$

例: 試行回数 n, 成功確率 p のBernoulli試行の分布は成功確率 p のBernoulli分布の n 乗になる. 以下の節も参照せよ.

1.6 重要: 分布Dの標本分布の主な使用用途 (統計モデル, 統計量)

確率分布 D のサイズ n の標本分布は, 現実に得られる長さ n の数値列 $x_1, \ldots, x_n$ の形式のデータの生成法則のモデル化としてよく使われている.

そのとき, 確率分布 D やその標本分布 D^n は **統計モデル** と呼ばれる.

ただし, 分布 D のサイズ n の標本分布 D^n は各々が分布 D に従う独立同分布な n 個の確率変数達 $X_1, \ldots, X_n$ の同時確率分布のことだと定義されていたので, データ $x_1, \ldots, x_n$ の生成法則が独立同分布という強い条件を近似的に満たしていなければ, そのようなモデル化は妥当ではなくなる. (独立同分布の仮定を緩めて統計モデルを作ることもよくある. しかし, 独立同分布の易しい場合についての理解抜きにその仮定を外した場合について理解することは無理である.)

使用している統計モデルの妥当性の判断をシンプルな一般論に従って完遂することは不可能である. 応用先の専門知識を徹底的に用いて, 個々の目的に合わせて, 統計モデルの適切さをそのユーザー側が独自に見積もる必要がある.

統計モデルとしての標本分布 D^n とそれに従う独立同分布な確率変数達 $X_1, \ldots, X_n$ が与えられたとき, $X_1, \ldots, X_n$ の函数 $f(X_1, \ldots, X_n)$ (これも確率変数になる)を **統計量** (statistic)と呼ぶことがある.

例えば, 標本の平均

$$\bar{X} = \mathrm{mean}(X_1, \ldots, X_n) = \frac{1}{n} \sum_{i=1}^{n} X_i$$

は統計量の典型的な例になっている. これ以外にも, 標本の分散, 標本の中央値, 標本の四分位数, 標本の最大値や最小値などはすべて以上の意味での統計量とみなされる.

統計モデルに従う確率変数達 $X_1, \ldots, X_n$ は現実で得られるデータ $x_1, \ldots, x_n$ の生成法則のモデル化になっているので, 統計量 $f(X_1, \ldots, X_n)$ には現実から得たデータの数値 $x_1, \ldots, x_n$ を代入して, 数値 $f(x_1, \ldots, x_n)$ を計算できる.

例えば, 上の標本平均の定義式中の X_i に現実から得た数値 x_i を代入すれば, データの数値の標本平均

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

が得られる.

現実世界と統計モデルの比較は様々な統計モデル内での確率変数としての統計量 $f(X_1, \ldots, X_n)$ 達と現実世界で得たデータから得られるそれに対応する数値　$f(x_1, \ldots, x_n)$ 達を比較することによって行われる.

例えば, 統計モデル内の確率変数としての標本平均 $\bar{X}$ の振る舞い方と, 現実のデータの標本平均 $\bar{x}$ (これは数値になる)を比較することによって, 平均に関する統計分析が行われる.

統計モデルを現実から得たデータ(もしくはその生成法則)に適合させるためにモデルのパラメータを調節することを **パラメータの推定** (estimating parameters)という.

例えばモデルの平均パラメータ μ はデータの標本平均で推定することが多い.

標本平均は統計量(統計モデル内での確率変数)でもあった. モデルのパラメータ θ の推定に使われる統計量をよく $\hat{\theta}$ のように書いて, パラメータ θ の **推定量** (estimator)と呼ぶ. 平均を意味するパラメータ μ を標本平均で推定する場合には $\hat{\mu} = \bar{X}$ と書いたりする.

例: データ $x_1, \ldots, x_n$ がコンピュータで $\mathrm{rand}()$ 函数を n 回繰り返して得られた乱数列とするとき, そのデータの生成法則は一様分布 $\mathrm{Uniform}(0, 1)$ のサイズ n の標本分布に近似的に従っているとみなされる. **例終**

例(基本!): S市の中学3年生男子の総人数は N 人であるとし, その身長の値全体を $a_1, a_2, \ldots, a_N$ と書く. 等確率 $1/N$ で a_i 達のどれかの値が選ばれるという条件で定義される有限離散分布を Q と書き, S市の中3男子の身長の分布と呼ぶことにする. (全数調査前であるという理由で現時点で分布 Q は未知であると仮定する.)

$1, 2, \ldots, N$ から n 個の数 $i_1, \ldots, i_n$ を無作為に選んで, それに対応する N 人中 n 人分の中3男子の身長を測定してデータ $(x_1, \ldots, x_N) = (a_{i_1}, \ldots, a_{i_N})$ が得られたとする. このとき, データ $x_1, \ldots, x_N$ の生成法則はS市の中3男子の身長の分布 Q のサイズ n の標本分布になっていると考えられる. (確率的揺らぎは無作為抽出の段階で生じている.)

未知の分布 Q についての推定・推測・推論をデータ $x_1, \ldots, x_N$ を用いて行いたい.

そのときに, 未知の分布 Q のモデル化として平均 μ と分散 σ^2 をパラメータに持つ正規分布を設定し, データ $x_1, \ldots, x_N$ の生成法則のモデル化として, 正規分布のサイズ n の標本分布 $\mathrm{Normal}(\mu, \sigma)^n$ を採用することが考えらえる. この場合には正規分布モデルを統計モデルとして採用したことになる.

モデルのパラメータ μ, σ はデータ $x_1, \ldots, x_N$ を使って推定されることになる.

未知であるS市の中3男子の身長の分布 Q (よく **真の分布** と呼ばれる)が正規分布からかけ離れた形をしていると, 正規分布モデルの採用は妥当ではなくなる. 真の分布 Q が十分に正規分布に近いことが十分に確からしい場合には正規分布モデルの採用は適切になる.

統計モデル内で確率変数としての標本平均 $\bar{X}$ が従う分布とデータから得た数値としての標本平均 $\bar{x}$ の関係を調べて, 平均を意味するモデルのパラメータ μ について何が言えるかを考えることが平均に関する統計分析の出発点になる.

例終

例: 当たりが常に同じ確率で出続けると考えられるルーレットを n 回まわして, 当たりなら 1 を記録し, 外れなら 0 を記録することによって得られる $1, 0$ で構成された長さ n の数列 $x_1, \ldots, x_n$ の生成法則は, 未知の成功確率 p を持つ試行回数 n のBernoulli試行の確率分布 $\mathrm{Bernoulli}(p)^n$ (これはBernoulli分布の標本分布に等しい)に近似的に従うと考えられる.
$(X_1, \ldots, X_n) \sim \mathrm{Bernoulli}(p)^n$ のとき, 確率変数 $K = X_1 + \cdots + X_n$ は「n 回のBernoulli試行で 1 が出た回数」を意味する統計量になっている. 統計量 K はBernoulli試行モデル内部では二項分布に従う. 現実世界で「n 回中 k 回成功した」の形式のデータが得られたならば, 二項分布に従う統計量 K と現実世界で得たデータの数値 k を比較することができる. 以下の節ではそのための準備を行う. **例終**

1.7 試行回数nのBernoulli試行の分布はBernoulli分布の標本分布

試行回数 n, 成功確率 p のBernoulli試行の確率質量函数は

$$P(x_1, \ldots, x_n) = p^{x_1 + \cdots + x_n}(1-p)^{n-(x_1+\cdots+x_n)} = \prod_{i=1}^{n}(p^{x_i}(1-p)^{1-x_i}) \quad (x_i = 1, 0)$$

であった. これは成功確率 p のBernoulli分布の確率質量函数

$$P(x_i) = p^{x_i}(1-p)^{1-x_i} \quad (x_i = 1, 0)$$

の積になっているので, 試行回数 n, 成功確率 p のBernoulli試行の確率分布は, 成功確率 p のBernoulli分布のサイズ n の標本分布になっている.

未知の確率 p で当たりが出るルーレットを n 回まわしたときの, 長さ n の当たりと外れからなる列をそのルーレットの出目のサイズ n の **標本** (**サンプル**, sample)と呼ぶ.

その意味でのルーレットの出目のサンプルの確率的揺らぎは, Bernoulli分布の標本分布(すなわちBernoulli試行の分布)でモデル化される.

例えば, 未知の確率 p で当たりが出るルーレットを $n = 1000$ 回まわしてサンプル(データ)を取得したら, 1000 回中当たりが 308 回で外れが 692 回ならば, そのルーレットで当たりが出る確率は3割程度だろうと推定できる. 実際にはルーレットを 1000 回まわし直すたびに当たりの回数は確率的に揺らぐので, 推定結果も確率的に揺らぐことになる. そのような揺らぎを数学的にモデル化するために標本分布は使用される.

言葉使いに関する重要な注意: 「標本」「サンプル」は一般に複数の数値の集まりになる. 上のルーレットの場合には当たりには 1 を対応させ, 外れを 0 に対応させると, サンプルは 1 と 0 からなる長さ n の列になる. 一つひとつの数値をサンプルと呼ぶのではなく, 複数の数値の集まりをサンプルと呼ぶ. この専門用語的言葉遣いは日常用語的なサンプルという言葉の使い方からはずれているので注意が必要である. この辺の言葉遣いで誤解を防ぎたい場合には「データ」と呼ぶこともある.

モデルと現実の区別: 「標本」「サンプル」「データ」という用語は統計モデル内での確率変数列についても使われることがある. 現実から得た数値列 $x_1, \ldots, x_n$ と統計モデル内での確率変数列 $X_1, \ldots, X_n$ は厳密に区別されなければいけない. しかし, 慣習的な言葉遣いはその区別を難しくするスタイルになっているので注意が必要である.

1.8 二項分布による推定の確率的揺らぎの記述

前節の設定を引き継ぐ.

Bernoulli分布のサイズ n の標本分布における 1 の個数の分布は二項分布になるのであった. 仮に

> n 回中 k 回の当たりが出たときに, 未知である当たりが出る確率は k/n に近いだろうと推定

することにしたときの, k/n の確率的な揺らぎは二項分布によって計算できる.

K は二項分布 $\mathrm{Binomial}(n,p)$ に従う確率変数であるとし, 確率変数 $\hat{p}$ を $\hat{p} = K/n$ と定める. この確率変数 $\hat{p} = K/n$ は上のルールで定めた未知の確率 p の推定の仕方の数学的記述になっている. このとき, $\hat{p}$ はパラメータ p の **推定量** (estimator)であるという. 確率変数 $\hat{p}$ は「n 回中 k 回当たりが出た」というデータに対応する確率変数 K の函数になっている.

$\hat{p} = K/n$ の期待値と分散は, 二項分布の期待値と分散がそれぞれ np と $np(1-p)$ であることより, 以下のように計算される:

$$
\begin{aligned}
E[\hat{p}] &= E[K/n] = \frac{E[K]}{n} = \frac{np}{n} = p, \\
\mathrm{var}(\hat{p}) &= \mathrm{var}(K/n) = \frac{\mathrm{var}(K)}{n^2} = \frac{np(1-p)}{n^2} = \frac{p(1-p)}{n}.
\end{aligned}
$$

例えば, $n = 1000$, $p = 0.3$ のとき, $\hat{p}$ の標準偏差は

$$
\mathrm{std}(\hat{p}) = \sqrt{\frac{p(1-p)}{n}} = \sqrt{\frac{0.3 \cdot 0.7}{1000}} \approx 1.45\%.
$$

モデル内の設定では大雑把に推定結果はこの2倍の $\pm 3\%$ 程度揺らぐと考えらえる.

未知の p を使わずに $\mathrm{SE} = \mathrm{std}(\hat{p})$ の値を推定するためには, その式の中の p に p の推定量 $\hat{p}$ を代入して得られる公式

$$
\widehat{SE} = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}
$$

を使えばよいだろう. ($\mathrm{SE} = \mathrm{std}(\hat{p})$ は **標準誤差** (standard error)と呼ばれる. $\widehat{SE}$ は標準誤差の推定量である. $\widehat{SE}$ のことも標準誤差と呼ぶことがある. この辺の言葉遣いもイーカゲンな場合が多いので注意が必要である. 何をどう呼ぶかよりも, それが正確には何を意味しているかが重要である.

このような統計分析の結果が現実において信頼できるかどうかは, Bernoulli分布の標本分布によるモデル化の現実における妥当性に依存する. モデルが現実において妥当である証拠が全然無ければ, このような推測結果も信頼できないことになる. モデルの現実における妥当性の証拠の提示は統計モデルのユーザー側が独自に行う必要がある.

In [5]:

```
n, p = 1000, 0.3
dist = Binomial(n, p)
L = 10^6
K = rand(dist, L)
p̂ = K/n
histogram(p̂; norm=true, alpha=0.3, label="p̂ = K/n", bin=100)
vline!([p]; label="p=$p", xlabel="p̂", xtick=0.30-0.12:0.03:0.30+0.12)
title!("n = $n, p = $p")
```

Out[5]:

確かにランダムに決まる $\hat{p} = K/n$ の値は, 推定先の値である $p = 0.30 = 30\%$ を中心に大雑把に $\pm 3\%$ 程度の範囲に分布している. (グラフを見ればわかるように小さな確率で $\pm 3\%$ よりも大きく外れることもあることに注意せよ.)

現実の統計分析では, データを上のグラフの例のように100万回取り直したりできないので, このようなグラフを描くことはできない. たった1つの「n 回中 k 回成功」(もしくは「n 人の中 k 人」)の型のデータだけを使って判断を下さなければいけなくなる.

真っ暗闇の中を手探りで進むような感じになる.

1.9 必修問題: 大阪都構想に関する住民投票の結果について

2015年と2020年の大阪都構想に関する住民投票の結果は

- 2015年: 賛成: 694,844 (49.6%)　反対: 705,585 (50.4%)
- 2020年: 賛成: 675,829 (49.4%)　反対: 692,996 (50.6%)

であった([検索 (https://www.google.com/search?q=%E5%A4%A7%E9%98%AA%E9%83%BD%E6%A7%8B%E6%83%B3++%E4%BD%8F%E6%B0%91%E6%8A%95%E7%A5%A](https://www.google.com/search?q=%E5%A4%A7%E9%98%AA%E9%83%BD%E6%A7%8B%E6%83%B3++%E4%BD%8F%E6%B0%91%E6%8A%95%E7%A5%A)

どちらでも僅差で反対派が勝利した. パーセントの数値を見ると大変な僅差であったようにも見える. この数値に二項分布モデルを適用したらどうなるかを計算するのがこの問題の内容である. (注意・警告: ただし, 二項分布モデルの適用がこの場合に妥当であるかどうかは度外視する. 妥当であるか否かは目的によって異なる.)

確率変数 K は二項分布 $\mathrm{Binomial}(n,p)$ に従うと仮定する.

(1) $n = 694844 + 705585 = 1400429$, $p = 0.5$ のとき, 確率 $P(K \leq 694844)$ の2倍の値を求めよ.

(2) $n = 675829 + 692996 = 1368825$, $p = 0.5$ のとき, 確率 $P(K \leq 675829)$ の2倍の値を求めよ.

(3) $n = 694844 + 705585 = 1400429$ のとき, 以下を求めよ:

- $P(K \geq 694844) = 2.5\%$ になるようなパラメータ p の値 p_L,
- $P(K \leq 694844) = 2.5\%$ になるようなパラメータ p の値 p_U.

(4) $n = 675829 + 692996 = 1368825$ のとき, 以下を求めよ:

- $P(K \geq 675829) = 2.5\%$ になるようなパラメータ p の値 p_L,
- $P(K \leq 675829) = 2.5\%$ になるようなパラメータ p の値 p_U.

p_L, p_U の L, U は lower, upper の頭文字のつもりである.

確率やパラメータの数値は有効桁4桁まで求めよ. $0.00000000000000000001234$ のように 0 を沢山含む表示は見難いので,

$$1.234 \times 10^{20} = 0.\underbrace{0000000000000000000}_{20}1234$$

のように書かずに,

$$1.234\mathrm{E}{-20} = 1.234\mathrm{e}{-20}$$

のように書くこと. $1.234\mathrm{E}{-20}$, $1.234\mathrm{e}{-20}$ のどちらでもよい.

ヒント: $K \sim \mathrm{Binomia}(n,p)$ のとき,

$$\begin{aligned} P(K \geq k) &= 1 - \mathrm{cdf}(\mathrm{Binomial}(n,p), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p), \\ P(K \leq k) &= \mathrm{cdf}(\mathrm{Binomial}(n,p), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p). \end{aligned}$$

これらの公式から, 与えられた n と k について $P(K \geq k) = \alpha/2$ を満たす $p = p_L$ と $P(K \leq k) = \alpha/2$ を満たす $p = p_U$ はそれぞれ次のように表されることがわかる:

$$p_L = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2), \quad p_U = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2).$$

WolframAlphaでは Beta は `BetaDistribution` と書く.

注意: この問題の内容を一般化するだけで **検定** (統計的仮説検定, statistical hypothesis testing)や **信頼区間** (confidence interval)の一般論が得られる. (1), (2)はP値(モデルのデータへの適合度の指標の一種)を求める問題になっており, (3), (4)は 95% 信頼区間 $[p_L, p_U]$ を求める問題になっている(ただし両方Clopper-Pearsonの信頼区間の場合).

解答例:

(1) $P(K \leq 694844) \approx 1.130\mathrm{e}{-19}$

(2) $P(K \leq 675829) \approx 9.687\mathrm{e}{-49}$

(3) $[p_L, p_U] \approx [0.4953, 0.4970]$

(4) $[p_L, p_U] \approx [0.4929, 0.4946]$

解答終

1.9.1 WolframAlphaによる計算の例

以下の例は一部分が間違っている可能性がある. 鵜呑みせずに自分で正しい方法を採用しているかどうかを調べてから使って欲しい. (3), (4)については検算も重要なので確認しておくこと.

(1) [2 cdf(BinomialDistribution(694844 + 705585, 0.5), 694844) (https://www.wolframalpha.com/input?i=2+cdf%28BinomialDistribution%28694844+%2B+705585%2C+0.5%29%2C+694844%29)](https://www.wolframalpha.com/input?i=2+cdf%28BinomialDistribution%28694844+%2B+705585%2C+0.5%29%2C+694844%29)

(2) 2 cdf(BinomialDistribution(675829 + 692996, 0.5), 675829) (https://www.wolframalpha.com/input?i=2+cdf%28BinomialDistribution%28675829+%2B+692996%2C+0.5%29%2C+675829%29)

(3) p_L: quantile(BetaDistribution(694844, 705585+1), 0.025) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28694844%2C+705585%2B1%29%2C+0.025%29), もしくは InverseBetaRegularized(0.025, 694844, 705585 + 1) (https://www.wolframalpha.com/input?i=InverseBetaRegularized%280.025%2C+694844%2C+705585+%2B+1%29&lang=ja). 検算: 1 - cdf(BinomialDistribution(694844 + 705585, 0.49533667058), 694843) (https://www.wolframalpha.com/input?i=1+-+cdf%28BinomialDistribution%28694844+%2B+705585%2C+0.49533667058%29%2C+694843%29&lang=ja)

p_U: quantile(BetaDistribution(694844+1, 705585), 0.975) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28694844%2B1%2C+705585%29%2C+0.975%29), もしくは InverseBetaRegularized(0.975, 694844+1, 705585) (https://www.wolframalpha.com/input?i=InverseBetaRegularized%280.975%2C+694844%2B1%2C+705585%29). 検算: cdf(BinomialDistribution(694844 + 705585, 0.496993553), 694844) (https://www.wolframalpha.com/input?i=cdf%28BinomialDistribution%28694844+%2B+705585%2C+0.496993553%29%2C+694844%29&lang=ja)

(4) p_L: quantile(BetaDistribution(675829, 692996 + 1), 0.025) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28675829%2C+692996+%2B+1%29%2C+0.025%29), もしくは InverseBetaRegularized(0.025, 675829, 692996 + 1) (https://www.wolframalpha.com/input?i=InverseBetaRegularized%280.025%2C+675829%2C+692996+%2B+1%29). 検算: 1 - cdf(BinomialDistribution(675829 + 692996, 0.49289139359), 675828) (https://www.wolframalpha.com/input?i=1+-+cdf%28BinomialDistribution%28675829+%2B+692996%2C+0.49289139359%29%2C+675828%29)

p_U: quantile(BetaDistribution(675829+1, 692996), 0.975) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28675829%2B1%2C+692996%29%2C+0.975%29), もしくは InverseBetaRegularized(0.975, 675829+1, 692996) (https://www.wolframalpha.com/input?i=InverseBetaRegularized%280.975%2C+675829%2B1%2C+692996%29). 検算: cdf(BinomialDistribution(675829 + 692996, 0.494567220), 675829) (https://www.wolframalpha.com/input?i=cdf%28BinomialDistribution%28675829+%2B+692996%2C+0.494567220%29%2C+675829%29&lang=ja)

用語の復習:

- cdf = cumulative distribution function = 累積分布函数 $F(x) = P(X \leq x)$
- quantile function = 分位点函数 = 累積分布函数の逆函数 $Q(p) = F^{-1}(p)$

確率分布に関するこれらの函数はコンピュータ上でも実装されており, 便利に使える. 確率分布をどのように使っているかがよくわからないブラックボックス的な統計ソフトのアウトプットを眺めるのではなく, 確率分布をどのように使ったかが分かるような道具を使うようにすると, 理解を深め易くなる.

1.9.2 Julia言語による計算の例

In [6]:
```
# 確率関係の数値計算は対数値を使わないと多くの場合に失敗する.
# 次は二項分布における確率質量函数の対数である.
logP(n, p, k) = logabsbinomial(n, k)[1] + k*log(p) + (n-k)*log(1-p)
```

Out[6]:
```
logP (generic function with 1 method)
```

In [7]:
```
# (1)
@show 2exp(logsumexp(logP(694844 + 705585, 0.5, k) for k in 0:694844))
@show 2cdf(Binomial(694844 + 705585, 0.5), 694844);
```

```
2 * exp(logsumexp((logP(694844 + 705585, 0.5, k) for k = 0:694844))) = 1.130422573100686e-19
2 * cdf(Binomial(694844 + 705585, 0.5), 694844) = 1.130422573435063e-19
```

In [8]:
```
# (2)
@show 2exp(logsumexp(logP(675829 + 692996, 0.5, k) for k in 0:675829))
@show 2cdf(Binomial(675829 + 692996, 0.5), 675829);
```

```
2 * exp(logsumexp((logP(675829 + 692996, 0.5, k) for k = 0:675829))) = 9.687442124914083e-49
2 * cdf(Binomial(675829 + 692996, 0.5), 675829) = 9.687442131513948e-49
```

In [9]:
```
# (3)
f(t) = ccdf(Binomial(694844 + 705585, t), 694844-1) - 0.025
g(t) =  cdf(Binomial(694844 + 705585, t), 694844)   - 0.025
@show p_L = find_zero(f, (0, 1))
@show p_U = find_zero(g, (0, 1));
```

```
p_L = find_zero(f, (0, 1)) = 0.4953366705819138
p_U = find_zero(g, (0, 1)) = 0.4969935526649652
```

In [10]:
```
# (3)
@show n = 694844 + 705585
@show k = 694844
@show α = 0.05
@show p_L = quantile(Beta(k, n-k+1), α/2)
@show p_U = quantile(Beta(k+1, n-k), 1 - α/2);
```

```
n = 694844 + 705585 = 1400429
k = 694844 = 694844
α = 0.05 = 0.05
p_L = quantile(Beta(k, (n - k) + 1), α / 2) = 0.4953366705819138
p_U = quantile(Beta(k + 1, n - k), 1 - α / 2) = 0.49699355266496514
```

In [11]:
```
# (3)
@show n = 694844 + 705585
@show k = 694844
@show α = 0.05
@show p_L = beta_inc_inv(k, n-k+1, α/2)[1]
@show p_U = beta_inc_inv(k+1, n-k, 1 - α/2)[1];
```

```
n = 694844 + 705585 = 1400429
k = 694844 = 694844
α = 0.05 = 0.05
p_L = (beta_inc_inv(k, (n - k) + 1, α / 2))[1] = 0.4953366705819138
p_U = (beta_inc_inv(k + 1, n - k, 1 - α / 2))[1] = 0.49699355266496514
```

In [12]:
```
# (4)
f(t) = ccdf(Binomial(675829 + 692996, t), 675829-1) - 0.025
g(t) =  cdf(Binomial(675829 + 692996, t), 675829)   - 0.025
@show p_L = find_zero(f, (0, 1))
@show p_U = find_zero(g, (0, 1));
```

```
p_L = find_zero(f, (0, 1)) = 0.49289139358630474
p_U = find_zero(g, (0, 1)) = 0.4945672196912542
```

In [13]:
```
# (4)
@show n = 675829 + 692996
@show k = 675829
@show α = 0.05
@show p_L = quantile(Beta(k, n-k+1), α/2)
@show p_U = quantile(Beta(k+1, n-k), 1 - α/2);
```

```
n = 675829 + 692996 = 1368825
k = 675829 = 675829
α = 0.05 = 0.05
p_L = quantile(Beta(k, (n - k) + 1), α / 2) = 0.49289139358630474
p_U = quantile(Beta(k + 1, n - k), 1 - α / 2) = 0.4945672196912543
```

In [14]:
```
# (4)
@show n = 675829 + 692996
@show k = 675829
@show α = 0.05
@show p_L = beta_inc_inv(k, n-k+1, α/2)[1]
@show p_U = beta_inc_inv(k+1, n-k, 1 - α/2)[1];
```

```
n = 675829 + 692996 = 1368825
k = 675829 = 675829
α = 0.05 = 0.05
p_L = (beta_inc_inv(k, (n - k) + 1, α / 2))[1] = 0.49289139358630474
p_U = (beta_inc_inv(k + 1, n - k, 1 - α / 2))[1] = 0.4945672196912543
```

1.9.3 Clopper-Pearsonの信頼区間とそれを与えるP値

(3)と(4)で計算した値から得られる区間 $[p_L, p_U]$ は **母比率に関する信頼度95%のClopper-Pearsonの信頼区間** として統計学ユーザーのあいだでよく知られている.

(1)と(2)での2倍する前の確率は **片側検定のP値** になっている. 2倍後の値は **両側検定のP値** (通常はこちらを使う)の一種になっており, Clopper-Pearsonの信頼区間を与える.

P値は採用した統計モデルとデータの整合性の指標である(P値が小さければ整合性が低い). (1)と(2)で求めたP値の値は極めて小さいということは, 成功確率 $p = 0.5$ の二項分布モデルと2015年と2020年の大阪都構想に関する住民投票の結果の整合性が極めて低いということを意味している. 2015年と2020年の大阪都構想に関する住民投票の結果については, 成功確率 $p = 0.5$ の二項分布モデルは捨て去る必要があると考えられる.

(3)と(4)で求めた区間 $[p_L, p_U]$ はデータから計算されるP値が 5% 以上になるパラメータ p の値全体の集合になっている. すなわち, P値に関する 5% の閾値(**有意水準** と呼ばれる)で整合性が低すぎるという理由で捨て去られずにすむ p の値全体が信頼度 $100\% - 5\% = 95\%$ の信頼区間になっている. このパターンは一般の場合にもそのまま通用する.

注意・警告: この 5% の閾値の設定は非常に普及しているが, 5% の閾値の設定に科学的に合理的な根拠があるわけではない. そのような閾値に頼る推論は科学的に不合理な推論になるので注意を要する. 非常に残念なことに, 統計学をかじってしまったせいで, 自分自身は科学的だと信じならが, そのような非科学的な推論をしてしまう人達がたくさんいる.

Clopper-Pearsonの信頼区間の効率的計算には, 二項分布の累積分布函数はベータ分布の累積分布函数で書けることが使われる: $K \sim \mathrm{Binomial}(n, p)$ のとき,

$$\begin{aligned} P(K \geq k) &= 1 - \mathrm{cdf}(\mathrm{Binomial}(n,p), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p), \\ P(K \leq k) &= \mathrm{cdf}(\mathrm{Binomial}(n,p), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p). \end{aligned}$$

これらの公式から, 与えられた n と k について $P(K \geq k) = \alpha/2$ を満たす $p = p_L$ と $P(K \leq k) = \alpha/2$ を満たす $p = p_U$ はそれぞれ次のように表されることがわかる:

$$p_L = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2), \quad p_U = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2).$$

ベータ分布の累積分布函数は **正則化された不完全ベータ函数** (regularized incomplete Beta function)になっている:

$$P(T \leq \theta) = I_\theta(\alpha, \beta) = \frac{\int_0^\theta t^{\alpha-1}(1-t)^{\beta-1}\,dt}{B(\alpha, \beta)} \quad \text{if} \quad T \sim \mathrm{Beta}(\alpha, \beta).$$

このことから, `quantile` 函数の代わりに $\theta \mapsto p = I_\theta(\alpha, \beta)$ の逆函数を直接使ってもよい:

$$p_L = \mathrm{beta_inc_inv}(k, n-k+1, \alpha/2)[1], \quad p_U = \mathrm{beta_inc_inv}(k+1, n-k, 1-\alpha/2)[1].$$

Julia言語の `SpecialFunctions.jl` では正則化された不完全ベータ函数とその逆函数はそれぞれ `using SpecialFunctions` した後に `p = beta_inc(α, β, θ)[1]` と `θ = beta_inc_inv(α, β, p)[1]` で使える.

Wolfram言語では `BetaRegularized` (https://reference.wolfram.com/language/ref/BetaRegularized.html) と `InverseBetaRegularized` (https://reference.wolfram.com/language/ref/InverseBetaRegularized.html) を使う.

Clopper-Pearsonの信頼区間を使うことのメリットは, 二項分布の累積分布函数を和で計算してからパラメータに関する方程式を解くという面倒な手続きを経由せずに, 基本特殊函数の1つである正則化された不完全ベータ函数の逆函数に帰着して効率的に計算できることである. 別の信頼区間の定義の仕方との比較でデメリットもある. Sterneの信頼区間(Sterne (1954) (https://www.jstor.org/stable/2333026))との相対比較では, Clopper-Pearsonの信頼区間の方が無駄に広くなってしまう場合が多い. np と $n(1-p)$ の両方が十分に大きな場合にはどちらを使っても実践的に意味のある差は出ない.

In [15]:

```
n, k = 16, 7

P1 = plot(p -> 1 - cdf(Binomial(n, p), k-1), 0, 1;
    label="1 - cdf(Binomial(n, p), k-1)", legend=:bottomright)
plot!(p -> cdf(Beta(k, n-k+1), p), 0, 1;
    label="cdf(Beta(k, n-k+1), p)", ls=:dash)
title!("n = $n,  k = $k"; xlabel="p")

P2 = plot(p -> cdf(Binomial(n, p), k), 0, 1;
    label="cdf(Binomial(n, p), k)", legend=:topright)
plot!(p -> 1 - cdf(Beta(k+1, n-k), p), 0, 1;
    label="1 - cdf(Beta(k+1, n-k), p)", ls=:dash)
title!("n = $n,  k = $k"; xlabel="p")

plot(P1, P2; size = (800, 250), bottommargin=4Plots.mm)
```

Out[15]:

上のグラフでは $1 - \mathrm{cdf}(\mathrm{Binomial}(n, p), k-1)$ と $\mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p)$ がぴったり一致し, $\mathrm{cdf}(\mathrm{Binomial}(n, p), k)$ と $1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p)$ がぴったり一致し, 区別がつかなくなっている. ここで $\mathrm{cdf}(D, x)$ は分布 D の累積分布函数を表す:

$$X \sim D \implies \mathrm{cdf}(D, x) = P(X \leq x).$$

1.9.4 信頼区間よりも情報量が大きなP値函数のプロット

二項分布モデルにおいて n, k が与えられたときに, パラメータ p に対してP値を対応させる函数を **P値函数** (p-value function)と呼ぶ. P値函数の値が有意水準 α 以上の p 全体の集合が信頼度 $1 - \alpha$ の信頼区間になる. この意味でP値函数はすべての信頼度に関する信頼区間の情報をすべて持っており, 適当な条件の下ではすべての信頼度に関する信頼区間が与えられていればそこからP値函数を逆に作れる. この意味でP値函数と信頼区間達は表裏一体の関係になっている.

以下では上の問題の場合についてのP値函数をプロットしてみよう.

In [16]:

```julia
function plot_pvalue_function!(pvalue_func, n, k; label="", kwargs...)
    p̂ = k/n
    SE = √(p̂*(1 - p̂)/n)
    ps = range(p̂ - 4SE, p̂ + 4SE, 1000)
    plot!(ps, p -> pvalue_func(n, k, p); label, kwargs...)
end

function plot_pvalue_function(pvalue_func, n, k; label="", kwargs...)
    plot()
    plot_pvalue_function!(pvalue_func, n, k; label)
    title!("n = $n, k = $k")
    plot!(; ytick=0:0.1:1, xlabel="parameter p", ylabel="p-value")
    plot!(; kwargs...)
end

function pvalue_clopper_pearson(dist::DiscreteUnivariateDistribution, x)
    min(1, 2cdf(dist, x), 2ccdf(dist, x-1))
end
pvalue_clopper_pearson(n, k, p) = pvalue_clopper_pearson(Binomial(n, p), k)

# (3)
P1 = plot_pvalue_function(pvalue_clopper_pearson, 694844 + 705585, 694844;
    label="Clopper-Pearson", xtick=0:0.0005:1)

# (4)
P2 = plot_pvalue_function(pvalue_clopper_pearson, 675829 + 692996, 675829;
    label="Clopper-Pearson", xtick=0:0.0005:1)

plot(P1, P2; size=(800, 250),
    leftmargin=4Plots.mm, bottommargin=4Plots.mm, rightmargin=4Plots.mm)
```

Out[16]:

データ x を与えられているとき(上の場合には「n 人中 k 人が賛成」というデータを与えられているとき), 統計モデルのパラメータ θ に(上の場合には p に)対してP値を対応させる函数をP値函数と呼ぶ. 上のグラフは大阪都構想に関する住民投票の結果データとし, 統計モデルを二項分布にしたときのP値函数のプロットである.

P値が高いパラメータ値ほど与えられたデータに統計モデルがよくフィット(fit, 適合)している. P値が低い場合にはその逆になる. あまりにも低いP値を持つパラメータ値については, 統計モデルとそのパラメータ値の組が与えられたデータに全然適合していないことになる.

そのとき重要なことは, P値が低いパラメータ値について, そのパラメータ値のみについてデータに適合していないと考えるのではなく, 統計モデルとそのパラメータ値の組がデータにフィットしていないと考えなければいけないことである.

例えば, 上のグラフを見れば, グラフに描いた部分の横軸の目盛りをはみ出しているパラメータ値 $p = 0.5$ のP値はおそろしく低い. 二項分布モデル内で $p = 0.5$ は賛成派の比率がちょうど 50% であることを意味している. $p = 0.5$ のP値がおそろしく低いことを理由に「賛成派の比率が 50% であるという仮説は否定された」のように単純に考えてはいけない. そうではなく, 「賛成派の比率が 50% であることを意味するパラメータ値の二項分布モデルの現実における妥当性は疑わしい」と考えなければいけない. 二項分布モデルも現実における妥当性を疑う対象に入れる必要がある.

この手のことが大学学部生向けの教科書に書かれていないせいで, 大学で統計学の講義を受講した人達の多くが統計学の最も基本的な部分をひどく誤解したまま単位を取得しているように思われる.

誤解を修正するためには次の講義動画がお勧めである:

- 京都大学大学院医学研究科 聴講コース 臨床研究者のための生物統計学「仮説検定とP値の誤解」佐藤 俊哉 医学研究科教授. [YouTube (https://youtu.be/vz9cZnB1d1c)]

信頼区間に関する解説は40:22以降にある.

その講義で扱われているP値に関するASA声明の翻訳とその翻訳の経緯の解説を以下の場所で読める:

- 統計的有意性とP値に関するASA声明 [pdf (https://www.biometrics.gr.jp/news/all/ASA.pdf)]
- 佐藤俊哉, ASA声明と疫学研究におけるP値, 計量生物学, 2018年38巻2号, pp. 109-115. [link (https://www.jstage.jst.go.jp/article/jjb/38/2/38_109/_article/-char/ja/)]

1.9.5 Sterneの信頼区間とそれを与えるP値函数

Sterneの信頼区間を与えるP値函数の定義は, 二項分布の確率質量函数を

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

と書くとき,

$$\mathrm{pvalue}_{\mathrm{Sterne}}(k|n,p) = \sum_{j \text{ with } P(j|n,p) \leq P(k|n,p)} P(j|n,p)$$

と $P(k|n,p)$ 以下となるような $P(j|n,p)$ 達の和として定義される. すなわち, 二項分布 $\mathrm{Binomial}(n,p)$ においてその値が生じる確率がデータの数値 k が生じる確率以下になる確率としてSterneの信頼区間を与えるP値は定義される.

そして, 与えられた n, k について, Sterneの信頼区間はこのP値が α 以上になるパラメータ p の範囲として定義される. (実はその定義だと区間になるとは限らない場合が稀にあるので, その場合には適当に定義を訂正することになる.)

1.9.6 Sterneの信頼区間を与えるP値函数の実装例

以下はSterneの信頼区間を与えるP値函数の実装例である.

Clopper-Pearsonの信頼区間を与えるP値函数の実装(実質1行!)と比較すると相当に複雑になっている.

そして, 実装の仕方によって計算効率に大きな違いが生じていることにも注目せよ.

In [17]:

```julia
x ≲ y = x < y || x ≈ y

# Naive implementation is terribly slow.
function pvalue_sterne_naive(dist::DiscreteUnivariateDistribution, x; xmax = 10^6)
    Px = pdf(dist, x)
    Px == 0 && return Px
    ymin, maxdist = minimum(dist), maximum(dist)
    ymax = maxdist == Inf ? xmax : maxdist
    sum(pdf(dist, y) for y in ymin:ymax if 0 < pdf(dist, y) ≲ Px; init = 0.0)
end
pvalue_sterne_naive(n, k, p) = pvalue_sterne_naive(Binomial(n, p), k)

# Second implementation is very slow.
function pvalue_sterne_old(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    Px == 0 && return Px
    distmin, distmax = extrema(dist)
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = m + 1
        while !(pdf(dist, y) ≲ Px)
            y += 1
        end
        cdf(dist, x) + ccdf(dist, y-1)
    else # k > m
        y = m - 1
        while !(pdf(dist, y) ≲ Px)
            y -= 1
        end
        cdf(dist, y) + ccdf(dist, x-1)
    end
end
pvalue_sterne_old(n, k, p) = pvalue_sterne_old(Binomial(n, p), k)

### The following implementation is efficient.

_pdf_le(x, (dist, y)) =  pdf(dist, x) ≲ y

function _search_boundary(f, x0, Δx, param)
    x = x0
    if f(x, param)
        while f(x - Δx, param) x -= Δx end
    else
        x += Δx
        while !f(x, param) x += Δx end
    end
    x
end

function pvalue_sterne(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    Px == 0 && return Px
    Px == 1 && return Px
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = _search_boundary(_pdf_le, 2m - x, 1, (dist, Px))
        cdf(dist, x) + ccdf(dist, y-1)
    else # x > m
        y = _search_boundary(_pdf_le, 2m - x, -1, (dist, Px))
        cdf(dist, y) + ccdf(dist, x-1)
    end
end
pvalue_sterne(n, k, p) = pvalue_sterne(Binomial(n, p), k)
```

Out[17]:

```
pvalue_sterne (generic function with 2 methods)
```

In [18]:
```
n = 10
k = -1:11
p = 0.4
a = @time pvalue_sterne_naive.(n, k, p)
b = @time pvalue_sterne_old.(n, k, p)
c = @time pvalue_sterne.(n, k, p)
d = @time pvalue_clopper_pearson.(n, k, p)
@show a ≈ b ≈ c
[a b c d]
```

```
  0.140221 seconds (441.18 k allocations: 24.341 MiB, 99.54% compilation time)
  0.099670 seconds (252.88 k allocations: 13.405 MiB, 99.52% compilation time)
  0.089010 seconds (202.95 k allocations: 10.804 MiB, 99.48% compilation time)
  0.041052 seconds (133.10 k allocations: 7.211 MiB, 98.88% compilation time)
a ≈ b ≈ c = true
```

Out[18]:
```
13×4 Matrix{Float64}:
 0.0          0.0          0.0          0.0
 0.00772434   0.00772434   0.00772434   0.0120932
 0.058652     0.058652     0.058652     0.0927148
 0.333528     0.333528     0.333528     0.33458
 0.749177     0.749177     0.749177     0.764561
 1.0          1.0          1.0          1.0
 0.534186     0.534186     0.534186     0.733793
 0.212596     0.212596     0.212596     0.332477
 0.101119     0.101119     0.101119     0.109524
 0.0183412    0.0183412    0.0183412    0.0245891
 0.00167772   0.00167772   0.00167772   0.00335544
 0.000104858  0.000104858  0.000104858  0.000209715
 0.0          0.0          0.0          0.0
```

In [19]:
```
# (3)の場合に
# pvalue_sterne_naive は pvalue_sterne_old よりも数百倍遅く,
# pvalue_sterne_old は pvalue_sterne よりも数百倍遅く,
# pvalue_sterne は pvalue_clopper_pearson よりも少し遅い.
n = 694844 + 705585
k = 694844
a = @btime pvalue_sterne_naive($n, $k, 0.5)
b = @btime pvalue_sterne_old($n, $k, 0.5)
c = @btime pvalue_sterne($n, $k, 0.5)
d = @btime pvalue_clopper_pearson($n, $k, 0.5)
@show a ≈ b ≈ c ≈ d
a, b, c, d
```

```
  139.354 ms (0 allocations: 0 bytes)
  566.800 μs (8 allocations: 1.88 KiB)
  1.310 μs (8 allocations: 1.88 KiB)
  817.073 ns (8 allocations: 1.88 KiB)
a ≈ b ≈ c ≈ d = true
```

Out[19]:
```
(1.1304225731963725e-19, 1.130422573435063e-19, 1.130422573435063e-19, 1.130422573435063e-19)
```

In [20]:
```
# 極端な場合
n = 694844 + 705585
k = 600000
b = @btime pvalue_sterne_old($n, $k, 0.5)
c = @btime pvalue_sterne($n, $k, 0.5)
d = @btime pvalue_clopper_pearson($n, $k, 0.5)
b, c, d
```

```
  102.553 ns (0 allocations: 0 bytes)
  100.319 ns (0 allocations: 0 bytes)
  179.124 ns (0 allocations: 0 bytes)
```

Out[20]:
```
(0.0, 0.0, 0.0)
```

In [21]:
```
# この場合には pvalue_sterne_naive はさらに遅い.
n = 100000
k = 49500:50500
a = @time pvalue_sterne_naive.(n, k, 0.5)
b = @time pvalue_sterne_old.(n, k, 0.5)
c = @time pvalue_sterne.(n, k, 0.5)
d = @time pvalue_clopper_pearson.(n, k, 0.5)
@show a ≈ b ≈ c ≈ d;
```

```
 11.429182 seconds (3 allocations: 8.094 KiB)
  0.028661 seconds (8.00 k allocations: 1.839 MiB)
  0.001661 seconds (8.00 k allocations: 1.839 MiB)
  0.001319 seconds (8.01 k allocations: 1.841 MiB)
a ≈ b ≈ c ≈ d = true
```

In [22]:
```
# 以上の実装は超幾何分布でも使える.
dist = Hypergeometric(9, 9, 9)
k = -1:10
a = @time pvalue_sterne_naive.(dist, k)
b = @time pvalue_sterne_old.(dist, k)
c = @time pvalue_sterne.(dist, k)
d = @time pvalue_clopper_pearson.(dist, k)
@show a ≈ b ≈ c ≈ d
[a b c d]
```

```
  0.114350 seconds (360.91 k allocations: 19.790 MiB, 10.38% gc time, 82.45% compilation time)
  0.047664 seconds (166.45 k allocations: 8.987 MiB, 96.80% compilation time)
  0.055169 seconds (154.17 k allocations: 8.274 MiB, 99.27% compilation time)
  0.037542 seconds (171.76 k allocations: 8.863 MiB, 98.93% compilation time)
a ≈ b ≈ c ≈ d = true
```

Out[22]:
```
12×4 Matrix{Float64}:
 0.0          0.0          0.0          0.0
 4.11353e-5   4.11353e-5   4.11353e-5   4.11353e-5
 0.0033731    0.0033731    0.0033731    0.0033731
 0.0566845    0.0566845    0.0566845    0.0566845
 0.346935     0.346935     0.346935     0.346935
 1.0          1.0          1.0          1.0
 1.0          1.0          1.0          1.0
 0.346935     0.346935     0.346935     0.346935
 0.0566845    0.0566845    0.0566845    0.0566845
 0.0033731    0.0033731    0.0033731    0.0033731
 4.11353e-5   4.11353e-5   4.11353e-5   4.11353e-5
 0.0          0.0          0.0          0.0
```

In [23]:
```
# 以上の実装は無限離散分布のPoisson分布でも使える.
dist = Poisson(4)
k = -1:10
a = @time pvalue_sterne_naive.(dist, k)
b = @time pvalue_sterne_old.(dist, k)
c = @time pvalue_sterne.(dist, k)
d = @time pvalue_clopper_pearson.(dist, k)
@show a ≈ b ≈ c
[a b c d]
```

```
  0.586314 seconds (434.29 k allocations: 24.231 MiB, 2.44% gc time, 20.48% compilation time)
  0.052354 seconds (225.50 k allocations: 12.065 MiB, 94.51% compilation time)
  0.050064 seconds (156.00 k allocations: 8.388 MiB, 97.21% compilation time)
  0.038139 seconds (150.01 k allocations: 8.137 MiB, 98.82% compilation time)
a ≈ b ≈ c = true
```

Out[23]:
```
12×4 Matrix{Float64}:
 0.0          0.0          0.0          0.0
 0.0396791    0.0396791    0.0396791    0.0366313
 0.202252     0.202252     0.202252     0.183156
 0.452973     0.452973     0.452973     0.476207
 1.0          1.0          1.0          0.86694
 1.0          1.0          1.0          1.0
 0.609266     0.609266     0.609266     0.742326
 0.306448     0.306448     0.306448     0.429739
 0.12899      0.12899      0.12899      0.221348
 0.0694493    0.0694493    0.0694493    0.102267
 0.0213634    0.0213634    0.0213634    0.0427269
 0.00813224   0.00813224   0.00813224   0.0162645
```

In [24]:

```
# (3)
P1 = plot_pvalue_function(pvalue_sterne, 694844 + 705585, 694844;
    label="Sterne", xtick=0:0.0005:1)

# (4)
P2 = plot_pvalue_function(pvalue_sterne, 675829 + 692996, 675829;
    label="Sterne", xtick=0:0.0005:1)

plot(P1, P2; size=(800, 250), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[24]:

In [25]:

```
# Clopper-Pearsonの信頼区間を与えるP値函数とSterneの信頼区間を与えるP値函数の比較
# (3)の場合にはほぼぴったり一致している.
n, k = 694844 + 705585, 694844
plot(title="p-value functions for n = $n, k = $k", ytick=0:0.1:1)
plot_pvalue_function!(pvalue_sterne, n, k; label="Sterne")
plot_pvalue_function!(pvalue_clopper_pearson, n, k; label="Clopper-Pearson", ls=:dash)
```

Out[25]:

このように数値的にぴったり一致する場合にはClopper-Pearsonの信頼区間を与えるP値函数とSterneの信頼区間のどちらを使うか悩む必要はないだろう. どちらを使っても(ほぼ)同じ結果が得られる.

In [26]:

```
# Clopper-Pearsonの信頼区間を与えるP値函数とSterneの信頼区間を与えるP値函数の比較
# n = 50, k = 15 の場合には違いが見える.
n, k = 50, 15
plot(title="p-value functions for n = $n, k = $k", ytick=0:0.1:1)
plot_pvalue_function!(pvalue_sterne, n, k; label="Sterne")
plot_pvalue_function!(pvalue_clopper_pearson, n, k; label="Clopper-Pearson", ls=:dash)
```

Out[26]:

Sterneの信頼区間を与えるP値函数の値はClopper-Pearsonの信頼区間を与えるP値函数の値よりも小さくなりことが多い. (常にそうなるわけではない.) その結果, 対応する信頼区間もSterneの側が狭くなってくれることが多い.

1.10 対ごとに独立であっても全体が独立であるとは限らない

確率変数達 X, Y, Z の互いに異なる任意の2つが独立であっても, X, Y, Z の全体が独立でない場合があることを具体的な例を作ることによって証明しよう.

確率変数 X, Y, Z の同時確率質量函数が $P(x, y, z)$ であるとき, X 単独の確率質量函数は $P(x) = \sum_{y,z} P(x, y, z)$ になり, (X, Y) の同時確率質量函数は $P(x, y) = \sum_z P(x, y, z)$ になる. 他の場合も同様である.

注意: 変数名で異なる函数を区別するスタイルを採用したので記号法の運用時に注意すること. このスタイルでは $P(x, y)$ と $P(x, z)$ は異な函数になる. 値を代入する場合には $P(x = 1, y = 1)$ や $P(x = 1, z = 1)$ のように書いて区別できるようにする.

$P(x, y, z)$ で $P(x, y) = P(x)P(y)$, $P(x, z) = P(x)P(z)$, $P(y, z) = P(y)P(z)$ を満たすが $P(x, y, z) \neq P(x)P(y)P(z)$ となるものを具体的に構成すればよい.

例えば以下のような確率質量函数 $P(x, y, z)$ の例を作ることができる:

	$z = 1$		$z = 0$	
	$y = 1$	$y = 0$	$y = 1$	$y = 0$
$x = 1$	$P(1, 1, 1) = 1/4$	$P(1, 0, 1) = 0$	$P(1, 1, 0) = 0$	$P(1, 0, 0) = 1/4$
$x = 0$	$P(0, 1, 1) = 0$	$P(0, 0, 1) = 1/4$	$P(0, 1, 0) = 1/4$	$P(0, 0, 0) = 0$

x, y, z はそれぞれ $1, 0$ を動くとする. このとき, $P(x, y)$, $P(x, z)$, $P(y, z)$ の値はすべて $1/4$ になり, $P(x)$, $P(y)$, $P(z)$ の値はすべて $1/2$ になることがわかる. たとえば, $P(x = 1, y = 1) = P(1, 1, 1) + P(1, 1, 0) = 1/4 + 0 = 1/4$ であり, $P(x = 1) = \sum_{y,z} P(1, y, z) = 1/4 + 0 + 0 + 1/4 = 1/2$.

だから, $P(x, y) = P(x)P(y)$, $P(x, z) = P(x)P(z)$, $P(y, z) = P(y)P(z)$ が成立するが, $P(x, y, z) \neq 1/8 = P(x)P(y)P(z)$ となっている.

このとき, X, Y, Z が同時確率質量函数 $P(x, y, z)$ を持つとすると, そのうちの任意の異なる2つは独立だが, X, Y, Z の全体は独立ではない.

1.11 確率変数の独立性の現実における解釈に関する重大な注意

上の例は具体的には次のような状況だと解釈可能である.

(1) $P(x) = 1/2$ の解釈: $X = 1$ は薬Aを与えたことを, $X = 0$ は薬Bを与えたことを意味する. 全員に確率 $1/2$ で薬Aまたは薬Bを与えた.

$x = 1 =$ 薬A	$P(x = 1) = 1/2$
$x = 0 =$ 薬B	$P(x = 0) = 1/2$

(2) $P(y) = 1/2$ の解釈: $Y = 1$ は薬に効果があったことを, $Y = 0$ は薬に効果が無かったことを意味する. 全体で見たら, $1/2$ の確率で薬には効果があった.

$y = 1 =$ 効果有	$y = 0 =$ 効果無
$P(y = 1) = 1/2$	$P(y = 0) = 1/2$

(3) $P(z) = 1/2$ の解釈: $Z = 1$ は女性であることをを, $Z = 0$ は男性であることを意味する. 女性である確率と男性である確率は $1/2$ だった.

$z = 1 =$ 女性	$z = 0 =$ 男性
$P(z = 1) = 1/2$	$P(z = 0) = 1/2$

(4) 男女の区別をやめると, 薬Aも薬Bも効果がある確率は半々であり, 薬Aと薬Bのどちらを与えたかと効果があったかどうかは独立である. 男女を合わせた($z = 1, 0$ の場合の和を取って得られる)確率質量函数 $P(x, y)$ の表

	$y = 1 =$ 効果有	$y = 0 =$ 効果無
$x = 1 =$ 薬A	$P(x = 1, y = 1) = 1/4$	$P(x = 1, y = 0) = 1/4$
$x = 0 =$ 薬B	$P(x = 0, y = 1) = 1/4$	$P(x = 0, y = 0) = 1/4$

は$x = 1$ の薬Aの場合も $x = 0$ の薬Bの場合も男女の区別をやめると薬Aと薬Bで効果に変わりがないことを意味している.

(5) 薬Aと薬Bのどちらを与えたかと男女のどちらであるかは独立である. そのことは効果の有無を意味する $y = 1, 0$ について和を取って得られる確率質量函数 $P(x, z)$ の表

	$z = 1 =$ 女性	$z = 0 =$ 男性
$x = 1 =$ 薬A	$P(x = 1, z = 1) = 1/4$	$P(x = 1, z = 0) = 1/4$
$x = 0 =$ 薬B	$P(x = 0, z = 1) = 1/4$	$P(x = 0, z = 0) = 1/4$

からわかる.

(6) 薬Aと薬Bのどちらを与えたかを無視すると, 男女のどちらであるかと薬の効果の有無は独立である. そのことは薬Aと薬Bのどちらを与えたかを意味する $x = 1, 0$ について和を取って得られる確率質量函数 $P(y, z)$ の表

	$y = 1 =$ 効果有	$y = 0 =$ 効果無
$z = 1 =$ 女性	$P(y = 1, z = 1) = 1/4$	$P(y = 0, z = 1) = 1/4$
$z = 0 =$ 男性	$P(y = 1, z = 0) = 1/4$	$P(y = 0, z = 0) = 1/4$

からわかる.

(7) しかし, 男女を区別すると全然違う結果が見えて来る. 薬Aは女性だけに効果があり, 薬Bは男性だけに効果がある. $z = 1$ の女性の場合に制限した確率質量函数 $P(x, y, z)$ $P(x, y, z)$ の表

	$z = 1 =$ 女性	
	$y = 1 =$ 効果有	$y = 0 =$ 効果無
$x = 1 =$ 薬A	$P(1, 1, 1) = 1/4$	$P(1, 0, 1) = 0$
$x = 0 =$ 薬B	$P(0, 1, 1) = 0$	$P(0, 0, 1) = 1/4$

より, $x = 1$ の薬Aの場合には $y = 1$ の効果有の確率が正であるが, $x = 0$ の薬Bの場合には $y = 1$ の効果有の確率が0になっている. $z = 0$ の男性の場合に制限した確率質量函数 $P(x, y, z)$ の表

	$z = 0 =$ 男性	
	$y = 1 =$ 効果有	$y = 0 =$ 効果無
$x = 1 =$ 薬A	$P(1, 1, 0) = 0$	$P(1, 0, 0) = 1/4$
$x = 0 =$ 薬B	$P(0, 1, 0) = 1/4$	$P(0, 0, 0) = 0$

より, $x = 1$ の薬Aの場合には $y = 1$ の効果有の確率が0で効果がないが, $x = 0$ の薬Bの場合には $y = 1$ の効果有の確率が正になっている.

このように現実世界において確率変数達が独立か否かは重大な意味を持ち得る. ある重要な条件(上の場合には女性か男性か)を無視して, 「XとYは独立である」(どちらの薬を与えても効果は同じである)と結論すると大変なことになってしまうかもしれない. XとYも, XとZも, YとZも独立であっても, XとYとZの全体は独立でないかもしれない.

2 確率変数達の共分散と相関係数と無相関性

分散と共分散と相関係数と無相関性に関する理論は線形代数における有限実ベクトル空間における内積の理論と本質的に同じものであると考えられる. 分散, 共分散, 相関係数, 無相関性はそれぞれベクトルのノルムの2乗, 内積, ベクトルのあいだの角度 θ に関する $\cos\theta$, ベクトルの直交に対応している.

確率変数	線形代数
分散	ノルムの2乗
共分散	内積
相関係数	$\cos\theta$
無相関	直交

2.1 確率変数達の共分散と相関係数の定義

確率函数 X, Y の期待値をそれぞれ $\mu_X = E[X]$, $\mu_Y = E[Y]$ と書くことにする.

確率変数 X, Y の **共分散** (covariance) $\sigma_{XY} = \mathrm{cov}(X, Y)$ を次のように定義する:

$$\sigma_{XY} = \mathrm{cov}(X, Y) = E[(X - \mu_X)(Y - \mu_Y)].$$

$X = Y$ のときこれは X の分散になる. 分散の場合と同様に

$$\begin{aligned}\mathrm{cov}(X, Y) &= E[(X - \mu_X)(Y - \mu_Y)] \\ &= E[XY] - \mu_X E[Y] - E[X]\mu_Y + \mu_X\mu_Y \\ &= E[XY] - E[X]E[Y] - E[X]E[Y] + E[X]E[Y] \\ &= E[XY] - E[X]E[Y].\end{aligned}$$

$X = Y$ の場合から $\mathrm{var}(X) = E[X^2] - E[X]^2$ が得られる.

線形代数の言葉を使えば, 共分散 $\sigma_{XY} = \mathrm{cov}(X, Y)$ は内積に対応しており, 分散 $\sigma_X^2 = \mathrm{var}(X)$, $\sigma_Y^2 = \mathrm{var}(Y)$ はノルム(ベクトルの長さ)の二乗に対応しており, 標準偏差 $\sigma_X = \mathrm{std}(X)$, $\sigma_Y = \mathrm{std}(Y)$ はノルム(ベクトルの長さ)に対応している.

さらに確率変数 X, Y の **相関係数** (correlation coefficient) $\rho_{XY} = \mathrm{cov}(X, Y)$ はベクトルのあいだのなす角度 θ に対する $\cos\theta$ の対応物として次のように定義される:

$$\rho_{XY} = \mathrm{cor}(X, Y) = \frac{\mathrm{cov}(X, Y)}{\mathrm{std}(X)\,\mathrm{std}(Y)} = \frac{\sigma_{XY}}{\sigma_X\sigma_Y}.$$

相関係数という言葉を見たら $\cos\theta$ を想像し, 分散, 標準偏差, 共分散という言葉を見たら, ベクトルの長さの2乗, ベクトルの長さ, ベクトルの内積を想像すればよい. $X - \mu_X$ や $Y - \mu_Y$ がベクトルに対応している.

非常に重要な注意・警告: 以上で定義したのは確率変数達の共分散と相関係数であり, 後で定義する標本(サンプル, データ)の共分散と相関係数とは異なる数学的対象になるので注意が必要である. この点は平均と分散の場合と同様である. 我々は複雑な現実世界に立ち向かうための道具として数学的に統計モデルを設定し, 統計モデル内の住人として確率変数を考える. そのとき, 確率変数達の平均(期待値), 分散, 共分散, 相関係数を計算することは, 統計モデル内の確率変数の特徴を調べていることになる. 統計モデルを数学的フィクションである現実世界のデータと比較することは以下のように行われる:

1. 数学的フィクションである統計モデル内に住んでる現実世界のデータのモデル化になっている確率変数の特徴を調べる.
2. 現実世界で取得したデータの特徴を調べる.
3. 統計モデル内の確率変数の特徴と現実世界で取得したデータの特徴を比較してみる.

上で定義したの確率変数達の共分散と相関係数は1の段階で使用される. 2の段階で使われる道具については後で説明する.

数学的な注意: 統計学的な位置付けとして, 確率変数達の平均, 分散, 共分散, 相関係数などと, 標本(サンプル, データ)平均, 分散, 共分散, 相関係数は異なる対象だとみなす必要があることはすぐ上で説明した. しかし, 純粋に数学的性質を調べる場合には, 後者は前者の特別な場合になっていることから, 前者の数学的性質を調べれば後者の数学的性質もわかるという仕組みになっている. 初学者が混乱しがちになる原因の1つは, 数学的に鋭い初学者が確率変数達の扱いとデータ(サンプル, 標本)の取り扱いが数学的に似ていることに気付いてしまうせいで, それらの位置付けを混同してしまいがちなことにもあるように思われる.

2.2 確率変数達の対ごとの無相関性の定義

X, Y の相関係数 $\rho_{XY} = \mathrm{cor}(X, Y)$ が 0 になるとき, 確率変数達 X, Y は **無相関** であるという. X と Y が無相関であることは $X - \mu_X$ と $Y - \mu_Y$ が「直交している」と考えてもよい.

注意: X, Y の相関係数 $\rho_{XY} = \mathrm{cor}(X, Y)$ が **近似的に** 0 **になっている場合** には($\cos\theta$ に対応する量が近似的に 0 になっている場合には), 確率変数達 X, Y は **近似的に無相関** であると考える. そのとき注意するべきことは共分散 $\sigma_{XY} = \mathrm{cov}(X, Y)$ が近似的に 0 になっていても(内積に当たる量が近似的に 0 になっていても), 相関係数が 0 からほど遠い値になっていて, 近似的に無相関とは言えない場合があることである. 下にある関連問題の節の内容を参照せよ.

確率変数達 $X_1, \ldots, X_n$ が **対ごとに無相関** であるとは, そのうちの互いに異なる任意の2つが無相関であることだと定める. $\mu_i = E[X_i]$, $\sigma_i^2 = \mathrm{var}(X_i)$ のとき, $X_1, \ldots, X_n$ が対ごとに無相関であることは,

$$\mathrm{cov}(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)] = \sigma_i^2 \delta_{ij}$$

と書けることと同値である. ここで δ_{ij} は $i = j$ の場合にのみ 1 になり, それ以外の場合に 0 になるKroneckerのデルタである.

2.3 問題: 確率変数の相関係数の計算例

確率変数の組 (X, Y) は確率 $0.1, 0.2, 0.3, 0.4$ でそれぞれ $(10, -50), (-4, -10), (0, 20), (2, 0)$ という値になると仮定する:

probability	0.1	0.2	0.3	0.4
X	10	-4	0	2
Y	-50	-10	20	0

確率変数 X, Y の相関係数を小数点以下第2桁まで求めよ.

解答例: まず X, Y の期待値を求める:

$$\begin{aligned}\mu_X &= E[X] = 10 \cdot 0.1 + (-4) \cdot 0.2 + 0 \cdot 0.3 + 2 \cdot 0.4 = 1, \\ \mu_Y &= E[Y] = (-50) \cdot 0.1 + (-10) \cdot 0.2 + 20 \cdot 0.3 + 0 \cdot 0.4 = -1.\end{aligned}$$

ゆえに期待値を X, Y から引いた値の分布は次の表のようになる:

probability	0.1	0.2	0.3	0.4
$X - \mu_X$	9	-5	-1	1
$Y - \mu_Y$	-49	-9	21	1

この表を用いて確率変数達としての X, Y の分散と共分散を求める:

$$\begin{aligned}\mathrm{var}(X) &= 9^2 \cdot 0.1 + (-5)^2 \cdot 0.2 + (-1)^2 \cdot 0.3 + 1^2 \cdot 0.4 = 13.8, \\ \mathrm{var}(Y) &= (-49)^2 \cdot 0.1 + (-9)^2 \cdot 0.2 + 21^2 \cdot 0.3 + 1^2 \cdot 0.4 = 389, \\ \mathrm{cov}(X, Y) &= 9(-49) \cdot 0.1 + (-5)(-9) \cdot 0.2 + (-1)21 \cdot 0.3 + 1 \cdot 1 \cdot 0.4 = -41.\end{aligned}$$

相関係数を求めよう:

$$\mathrm{cor}(X, Y) = \frac{\mathrm{cov}(X, Y)}{\sqrt{\mathrm{var}(X)}\sqrt{\mathrm{var}(Y)}} \approx -0.56.$$

解答終

In [27]:
```
@show E_X = 10*0.1 + (-4)*0.2 + 0*0.3 + 2*0.4
@show E_Y = (-50)*0.1 + (-10)*0.2 + 20*0.3 + 0*0.4
```

```
E_X = 10 * 0.1 + -4 * 0.2 + 0 * 0.3 + 2 * 0.4 = 1.0
E_Y = -50 * 0.1 + -10 * 0.2 + 20 * 0.3 + 0 * 0.4 = -1.0
```

Out[27]:
```
-1.0
```

In [28]:
```
@show var_X = 9^2*0.1 + (-5)^2*0.2 + (-1)^2*0.3 + 1^2*0.4
@show var_Y = (-49)^2*0.1 + (-9)^2*0.2 + 21^2*0.3 + 1^2*0.4
@show cov_XY = 9*(-49)*0.1 + (-5)*(-9)*0.2 + (-1)*21*0.3 + 1*1*0.4;
```

```
var_X = 9 ^ 2 * 0.1 + (-5) ^ 2 * 0.2 + (-1) ^ 2 * 0.3 + 1 ^ 2 * 0.4 = 13.8
var_Y = (-49) ^ 2 * 0.1 + (-9) ^ 2 * 0.2 + 21 ^ 2 * 0.3 + 1 ^ 2 * 0.4 = 389.0
cov_XY = 9 * -49 * 0.1 + -5 * -9 * 0.2 + -1 * 21 * 0.3 + 1 * 1 * 0.4 = -41.0
```

In [29]:
```
@show cor_XY = cov_XY/(√var_X * √var_Y);
```

```
cor_XY = cov_XY / (√var_X * √var_Y) = -0.5595894432479921
```

In [30]:

```
X = [10, -4, 0, 2]
Y = [-50, -10, 20, 0]
P = [0.1, 0.2, 0.3, 0.4]
@show E_X = sum(X[i]*P[i] for i in eachindex(P))
@show E_Y = sum(Y[i]*P[i] for i in eachindex(P))
@show var_X = sum((X[i] - E_X)^2*P[i] for i in eachindex(P))
@show var_Y = sum((Y[i] - E_Y)^2*P[i] for i in eachindex(P))
@show cov_XY = sum((X[i] - E_X) * (Y[i] - E_Y) * P[i] for i in eachindex(P))
@show cor_XY = cov_XY/(√var_X * √var_Y);
```

```
E_X = sum((X[i] * P[i] for i = eachindex(P))) = 1.0
E_Y = sum((Y[i] * P[i] for i = eachindex(P))) = -1.0
var_X = sum(((X[i] - E_X) ^ 2 * P[i] for i = eachindex(P))) = 13.8
var_Y = sum(((Y[i] - E_Y) ^ 2 * P[i] for i = eachindex(P))) = 389.0
cov_XY = sum(((X[i] - E_X) * (Y[i] - E_Y) * P[i] for i = eachindex(P))) = -41.0
cor_XY = cov_XY / (√var_X * √var_Y) = -0.5595894432479921
```

2.4 問題: Cauchy-Schwarzの不等式

確率変数達 X, Y について, 以下の不等式を示せ:

$$|\operatorname{cov}(X,Y)| \leq \operatorname{std}(X)\operatorname{std}(Y).$$

簡単のため X, Y が同時確率質量函数を持つと仮定してよい.

注意: Cauchy-Schwarzの不等式より, $\operatorname{std}(X)\operatorname{std}(Y) \neq 0$ のとき, 相関係数

$$\rho_{XY} = \operatorname{cor}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\operatorname{std}(X)\operatorname{std}(Y)}$$

が -1 以上 1 以下になることがわかる. これより, 0 以上 π 以下の θ を

$$\rho_{XY} = \operatorname{cor}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\operatorname{std}(X)\operatorname{std}(Y)} = \cos\theta$$

によって定めることができる. この θ はベクトルのあいだの角度の確率変数の場合の類似物になっている.

解答例1: 右辺の2乗から左辺の2乗を引いた結果が 0 以上になること, すなわち, 次を示せばよい:

$$\operatorname{var}(X)\operatorname{var}(Y) - \operatorname{cov}(X,Y)^2 \geq 0. \tag{$*$}$$

確率変数の分散や共分散は確率変数から定数を引いても変わらないので, X, Y からそれらの期待値を引いたものを考えることによって, X, Y の期待値は 0 であると仮定してよい. このとき,

$$\operatorname{var}(X) = E[X^2], \quad \operatorname{var}(Y) = E[Y^2], \quad \operatorname{cov}(X,Y) = E[XY].$$

X, Y の同時確率質量函数を $P(x,y)$ と書く:

$$E[f(X,Y)] = \sum_{x,y} f(x,y)P(x,y).$$

このとき,

$$E[f(X,Y)]E[g(X,Y)] = \sum_{x,y,x',y'} f(x,y)g(x',y')P(x,y)P(x',y').$$

以上の準備のもとで,

$$\begin{aligned}
2\operatorname{var}(X)\operatorname{var}(Y) &= E[X^2]E[Y^2] + E[Y^2]E[X^2] \\
&= \sum_{x,y,x',y'} (x^2y'^2 + y^2x'^2)P(x,y)P(x',y'), \\
2\operatorname{cov}(X,Y)^2 &= 2E[XY]E[XY] = \sum_{x,y,x',y'} 2xyx'y'P(x,y)P(x',y') \\
&= \sum_{x,y,x',y'} 2xy'yx'P(x,y)P(x',y').
\end{aligned}$$

ゆえに,

$$
\begin{aligned}
2\operatorname{var}(X)\operatorname{var}(Y) - 2\operatorname{cov}(X,Y)^2 &= \sum_{x,y,x',y'} (x^2y'^2 + y^2x'^2 - 2xy'yx')P(x,y)P(x',y') \\
&= \sum_{x,y,x',y'} (xy' - yx')^2 P(x,y)P(x',y') \geq 0.
\end{aligned}
$$

これより(∗)が成立することがわかる.

解答終

上の解答例でやっていることは本質的に (X,Y) と独立で同分布な確率変数 (X',Y') に対応する離散変数 x',y' を導入したことである. そのことに気付けば, 同時確率質量函数の存在に頼らない一般的な証明法も得られる.

解答例2(より一般的な場合): 右辺の2乗から左辺の2乗を引いた結果が 0 以上になること, すなわち, 次を示せばよい:

$$
\operatorname{var}(X)\operatorname{var}(Y) - \operatorname{cov}(X,Y)^2 \geq 0. \tag{$*$}
$$

確率変数の分散や共分散は確率変数から定数を引いても変わらないので, X, Y からそれらの期待値を引いたものを考えることによって, X, Y の期待値は 0 であると仮定してよい. このとき,

$$
\operatorname{var}(X) = E[X^2], \quad \operatorname{var}(Y) = E[Y^2], \quad \operatorname{cov}(X,Y) = E[XY].
$$

さらに, (X',Y') は (X,Y) と同じ分布を持つ (X,Y) と独立な確率変数であるとする. このとき, 次の公式を使用できる:

$$
E[f(X,Y)] = E[f(X',Y')], \quad E[f(X,Y)g(X',Y')] = E[f(X,Y)]E[g(X',Y')].
$$

以上の準備のもとで,

$$
\begin{aligned}
&2\operatorname{var}(X)\operatorname{var}(Y) = 2E[X^2]E[Y^2] = E[X^2]E[Y'^2] + E[Y^2]E[X'^2] = E[X^2Y'^2 + Y^2X'^2], \\
&2\operatorname{cov}(X,Y)^2 = 2E[XY]^2 = 2E[XY]E[X'Y'] = E[2XY'YX'].
\end{aligned}
$$

ゆえに,

$$
\begin{aligned}
2\operatorname{var}(X)\operatorname{var}(Y) - 2\operatorname{cov}(X,Y)^2 &= E[X^2Y'^2 + Y^2X'^2] - E[2XY'YX'] \\
&= E[X^2Y'^2 + Y^2X'^2 - 2XY'YX'] \\
&= E[(XY' - X'Y)^2] \geq 0.
\end{aligned}
$$

これより(∗)が成立することがわかる.

解答終

注意: 線形代数の教科書におけるCauchy-Schwarzの不等式の証明と以上で紹介した証明を比較してみよ. もしも比較した線形代数の教科書の証明が以上で紹介した証明と全く違っているならば, その教科書の文脈において上で紹介した証明の方法が使えないかどうかを考えてみよ. 参照にした教科書に上で紹介した証明と全く異なる証明が載っているならば, 上で証明した確率変数に関するCauchy-Schwarzの不等式をその教科書の方法で証明してみよ.

2.5 問題: 等確率有限離散分布の相関係数と cos θ の関係

$a_i, b_i \in \mathbb{R}$ であるとし, 確率変数の組 (X,Y) は同じ確率 $1/n$ で $(a_1,b_1),\dots,(a_n,b_n)$ のどれかの値になるものと仮定する. この仮定は

$$
E[f(X,Y)] = \frac{1}{n}\sum_{i=1}^{n} f(a_i,b_i)
$$

の成立と同値である(確率変数の組が定義されていることはその函数の期待値が定義されていることと同じことだと考えてよい). さらに,

$$
\mu_X = E[X] = \frac{1}{n}\sum_{i=1}^{n} a_i, \quad \mu_Y = E[Y] = \frac{1}{n}\sum_{i=1}^{n} b_i
$$

とおき, ベクトル v_X, v_Y を次のように定める:

$$
v_X = \frac{1}{\sqrt{n}}\begin{bmatrix} a_1 - \mu_X \\ \vdots \\ a_n - \mu_X \end{bmatrix}, \quad v_Y = \frac{1}{\sqrt{n}}\begin{bmatrix} b_1 - \mu_Y \\ \vdots \\ b_n - \mu_y \end{bmatrix}
$$

これらの内積(対応する成分の積の和)を (v_X, v_Y) と書き, ノルム(成分の2乗の和の平方根)を $\|v_X\|, \|v_Y\|$ と書き, v_X, v_Y のあいだの角度を θ と書くことにし, $\|v_X\| > 0$, $\|v_Y\| > 0$ であると仮定する. このとき, X, Y の相関係数 $\mathrm{cor}(X, Y)$ について次が成立することを示せ:

$$\mathrm{cor}(X, Y) = \frac{(v_X, v_Y)}{\|v_X\|\|v_Y\|} = \cos\theta.$$

注意: 相関係数とベクトルのあいだの角度の関係についてはこの場合をイメージしておけばよい.

解答例: 内積とノルムの定義より,

$$\begin{aligned}
\|v_X\|^2 &= \frac{1}{n}\sum_{i=1}^{n}(a_i - \mu_X)^2 = E[(X - \mu_X)^2] = \mathrm{var}(X), \\
\|v_Y\|^2 &= \frac{1}{n}\sum_{i=1}^{n}(b_i - \mu_Y)^2 = E[(Y - \mu_Y)^2] = \mathrm{var}(Y), \\
(v_X, v_Y) &= \frac{1}{n}\sum_{i=1}^{n}(a_i - \mu_X)(b_i - \mu_Y) = E[(X - \mu_X)(Y - \mu_Y)] = \mathrm{cov}(X, Y).
\end{aligned}$$

ゆえに,

$$\mathrm{cor}(X, Y) = \frac{\mathrm{cov}(X, Y)}{\sqrt{\mathrm{var}(X)}\sqrt{\mathrm{var}(Y)}} = \frac{(v_X, v_Y)}{\|v_X\|\|v_Y\|} = \cos\theta.$$

解答終

2.6 問題: 相関係数の計算

確率変数達 X, Y は無相関であるとし, それぞれの期待値と分散はどちらも 0 と 1 であるとし, $a, b, c, d \in \mathbb{R}$, $a^2 + b^2 > 0$, $c^2 + d^2 > 0$ であると仮定する. 確率変数 A, B を次のように定める:

$$A = aX + bY, \quad B = cX + dY.$$

確率変数達 A, B の相関係数を求めよ.

解答例: X, Y の期待値はどちらも 0 なのでそれらの一次結合である A, B の期待値も 0 になる. さらに, X, Y の分散はどちらも 1 でそれらは無相関だと仮定したので,

$$\begin{aligned}
&\mathrm{var}(A) = a^2\,\mathrm{var}(X) + b^2\,\mathrm{var}(Y) = a^2 + b^2, \\
&\mathrm{var}(B) = c^2\,\mathrm{var}(X) + d^2\,\mathrm{var}(Y) = c^2 + d^2, \\
&\mathrm{cov}(A, B) = E[AB] = E[acX^2 + (ad + bc)XY + bdY^2] \\
&= ac\,\mathrm{var}(X) + (ad + bc)\,\mathrm{cov}(X, Y) + bd\,\mathrm{var}(Y) = ac + bd.
\end{aligned}$$

ゆえに

$$\mathrm{cor}(A, B) = \frac{\mathrm{cov}(A, B)}{\sqrt{\mathrm{var}(A)}\sqrt{\mathrm{var}(B)}} = \frac{ac + bd}{\sqrt{a^2 + b^2}\sqrt{c^2 + d^2}}.$$

解答終

注意: ベクトル v_A, v_B を

$$v_A = \begin{bmatrix} a \\ b \end{bmatrix}, \quad v_B = \begin{bmatrix} c \\ d \end{bmatrix}$$

と定め, これらの標準内積(対応する成分の積の和)を (v_A, v_B) と書き, それらのノルム(すべての成分の2乗の和の平方根)を $\|v_A\|, \|v_B\|$ と書き, それらの間の角度を θ と書くとき,

$$\mathrm{cor}(A, B) = \frac{(v_A, v_B)}{\|v_A\|\|v_B\}} = \cos\theta$$

が成立している. 上の問題の状況において, 確率変数 A, B の相関係数が 1 に近いことはベクトル v_A, v_B の向きがほぼ同じであることと同値であり, A, B の相関係数が -1 に近いことは v_A, v_B の向きがほぼ正反対であることに同値であり, A, B が無相関に近いことは v_A, v_B がほぼ直交していることと同値である.

2.7 問題: 共分散が0に近くても相関係数が0から遠い場合がある

平均が 0 の具体的な確率変数達 X, Y の組で, 共分散 $\sigma_{XY} = \mathrm{cov}(X, Y) = E[XY]$ は 0 に近い値だが, 相関係数 $\rho_{XY} = \mathrm{cov}(X, Y) = \mathrm{cov}(X, Y)/(\mathrm{std}(X)\,\mathrm{std}(Y))$ が 0 から遠い値になっているものを構成せよ.

解答例: $\varepsilon, \delta > 0$ であるとし, 確率変数の組 (X, Y) は同じ確率 $1/3$ で $(\varepsilon + \delta, \varepsilon)$, $(-\varepsilon, \delta)$, $(-\delta, -\varepsilon - \delta)$ のどれかの値になるものとする. このとき, X, Y の平均はそれぞれ

$$E[X] = \frac{(\varepsilon + \delta) + (-\varepsilon) + (-\delta)}{3} = 0, \quad E[Y] = \frac{\varepsilon + \delta + (-\varepsilon - \delta)}{3} = 0$$

となる. X と Y の分散と共分散と相関係数は以下のようになる:

$$\begin{aligned}
\mathrm{var}(X) &= E[X^2] = \frac{(\varepsilon + \delta)^2 + (-\varepsilon)^2 + (-\delta)^2}{3} = \frac{2(\varepsilon^2 + \varepsilon\delta + \delta^2)}{3}, \\
\mathrm{var}(Y) &= E[Y^2] = \frac{\varepsilon^2 + \delta^2 + (-\varepsilon - \delta)^2}{3} = \frac{2(\varepsilon^2 + \varepsilon\delta + \delta^2)}{3}, \\
\mathrm{cov}(X, Y) &= E[XY] = \frac{(\varepsilon + \delta)\varepsilon + (-\varepsilon)\delta + (-\delta)(-\varepsilon - \delta)}{3} = \frac{(\varepsilon^2 + \varepsilon\delta + \delta^2)}{3}, \\
\mathrm{cor}(X, Y) &= \frac{\mathrm{cov}(X, Y)}{\sqrt{\mathrm{var}(X)}\sqrt{\mathrm{var}(Y)}} = \frac{1}{2}.
\end{aligned}$$

$\varepsilon, \delta > 0$ の値が小さいとき, X, Y の共分散は小さくなるが, 相関係数は $1/2$ で全然 0 に近くない. **解答終**

注意: 3次元ベクトル v_X, v_Y を以下のように定める:

$$v_X = \frac{1}{\sqrt{3}} \begin{bmatrix} \varepsilon + \delta \\ -\varepsilon \\ -\delta \end{bmatrix}, \quad v_Y = \frac{1}{\sqrt{3}} \begin{bmatrix} \varepsilon \\ \delta \\ -\varepsilon - \delta \end{bmatrix}.$$

このとき, これらの標準内積(対応する成分の積の和)を (v_X, v_Y) と書き, それらのノルム(すべての成分の2乗の和の平方根)を $\|v_X\|, \|v_Y\|$ と書き, それらの間の角度を θ と書くとき,

$$\begin{aligned}
&\mathrm{var}(X) = \|v_X\|^2, \quad \mathrm{var}(Y) = \|v_Y\|^2, \quad \mathrm{cov}(X, Y) = (v_X, v_Y), \\
&\mathrm{cor}(X, Y) = \frac{(v_X, v_Y)}{\|v_X\|\|v_Y\|} = \cos\theta
\end{aligned}$$

が成立していることに注意せよ.

2.8 問題: 独立ならば無相関である (実質1行で解ける)

確率変数達 X, Y が独立ならば確率変数達 X, Y は無相関であることを示せ.

解答例: X, Y は独立なので $E[f(X)g(Y)] = E[f(X)]E[g(Y)]$ となる. ゆえに, $\mu_X = E[X]$, $\mu_Y = E[Y]$ とおくと,

$$E[(X - \mu_X)(Y - \mu_Y)] = E[X - \mu_X]E[Y - \mu_Y] = (E[X] - \mu_X)(E[Y] - \mu_Y) = 0 \cdot 0 = 0.$$

解答終

2.9 問題: 無相関でも独立とは限らない

確率変数 X, Y で無相関だが独立でないものを具体的に構成せよ.

解答例1: 確率質量函数 $P(x, y)$ を次の表の通りに定める:

	$y = 1$	$y = 2$	$y = 3$	
$x = 1$	$P(1, 1) = 0$	$P(1, 2) = 1/4$	$P(1, 3) = 0$	$P(x = 1) = 1/4$
$x = 2$	$P(2, 1) = 1/4$	$P(2, 2) = 0$	$P(2, 3) = 1/4$	$P(x = 2) = 2/4$
$x = 3$	$P(3, 1) = 0$	$P(3, 2) = 1/4$	$P(3, 3) = 0$	$P(x = 3) = 1/4$
	$P(y = 1) = 1/4$	$P(y = 2) = 2/4$	$P(y = 3) = 1/4$	

確率変数 X, Y は同時確率質量函数 $P(x, y)$ を持つとする. このとき,

$$E[X] = E[Y] = 1 \cdot (1/4) + 2 \cdot (2/4) + 3 \cdot (1/4) = 2.$$

であり, 確率が 0 でない (x, y) については x または y が X と Y の期待値 2 になるので, 共分散は 0 になる:

$$
\begin{aligned}
\mathrm{cov}(X,Y) = E[(X-2)(Y-2)] &= \sum_{x,y}(x-2)(y-2)P(x,y) \\
&= (2-2)(1-2)(1/4) + (1-2)(2-2)(1/4) \\
&\quad + (3-2)(2-2)(1/4) + (2-2)(3-2)(1/4) = 0.
\end{aligned}
$$

これで X, Y は無相関であることがわかった.

しかし, たとえば $P(1,1) = 0$ と $P(x=1)P(y=1) = (1/4)(1/4)$ が等しくないので X, Y は独立ではない.

解答終

解答例2: 確率質量函数 $P(x,y)$ を次の表の通りに定めても上と同様に, X, Y は無相関だが独立ではないことを示せる:

	$y=1$	$y=2$	$y=3$	
$x=1$	$P(1,1)=1/8$	$P(1,2)=1/8$	$P(1,3)=1/8$	$P(x=1)=3/8$
$x=2$	$P(2,1)=1/8$	$P(2,2)=0$	$P(2,3)=1/8$	$P(x=2)=2/8$
$x=3$	$P(3,1)=1/8$	$P(3,2)=1/8$	$P(3,3)=1/8$	$P(x=3)=3/8$
	$P(y=1)=3/8$	$P(y=2)=2/8$	$P(y=3)=3/8$	

解答終

注意: 3×3 の表をすべて $1/9$ で埋めてしまうと, X, Y は独立になってしまう.

解答例3: $\mathbb{R}^2$ における単位円盤上の一様分布の確率密度函数を次のように定める:

$$
p(x,y) = \begin{cases} 1/\pi & (x^2+y^2 \leq 1) \\ 0 & (x^2+y^2 > 1) \end{cases}
$$

これを同時確率密度函数として持つ確率変数 X, Y を考える:

$$
E[f(X,Y)] = \iint_{\mathbb{R}^2} f(x,y)p(x,y)\,dx\,dy.
$$

単位円盤の対称性から $E[X] = E[Y] = 0$ となることがわかる. (具体的に積分を計算しても易しい.) それらの共分散は

$$
\mathrm{cov}(X,Y) = E[XY] = \frac{1}{\pi}\iint_{x^2+y^2 \leq 1} xy\,dx\,dy
$$

と書けるが, $xy \geq 0$ の部分の積分と $xy \leq 0$ の部分の積分が単位円盤の対称性より互いに打ち消しあって $\mathrm{cov}(X,Y) = 0$ となることがわかる. X 単独の密度函数は

$$
p(x) = \frac{1}{\pi}\int_{-\sqrt{1-x^2}}^{\sqrt{1-x^2}} dy = \frac{2}{\pi}\sqrt{1-x^2}
$$

となり, 同様にして $p(y) = (2/\pi)\sqrt{1-y^2}$ となるが, $-1 < x, y < 1$ かつ $x^2+y^2 > 1$ のとき, $p(x,y) = 0$ となるが, $p(x)p(y) \neq 0$ となるので, それらは等しくない. ゆえに X, Y は独立ではない.

解答終

この解答例3については, 下の方でプロットした (X,Y) の分布のサンプルの散布図も参照して欲しい. 散布図を見れば直観的にどうなっているかを把握しやすい.

In [31]:

```
n = 10^4
XY = [(r = √rand(); t = 2π*rand(); (r*cos(t), r*sin(t))) for _ in 1:n]
X, Y = first.(XY), last.(XY)
@show cov(X, Y) cor(X, Y)
P1 = scatter(X, Y; msc=:auto, ms=2, alpha=0.7, label="", xlabel="x", ylabel="y")
P2 = histogram(X; norm=true, alpha=0.3, bin=41, label="X")
plot!(x -> 2/π*√(1 - x^2), -1, 1; label="", lw=2)
P3 = histogram(X; norm=true, alpha=0.3, bin=41, label="Y")
plot!(y -> 2/π*√(1 - y^2), -1, 1; label="", lw=2)
plot(P1, P2, P3; size=(800, 400), layout=@layout [a [b; c]])
```

```
cov(X, Y) = -0.0005083072927225841
cor(X, Y) = -0.0020186738360558273
```

Out[31]:

単位円盤上の一様分布は「無相関だが独立ではない場合」の例になっている.

この「無相関だが独立ではない場合」の例のサンプルに含まれる点の個数を減らすと以下のように見える.

In [32]:

```
n = 100
XY = [(r = √rand(); t = 2π*rand(); (r*cos(t), r*sin(t))) for _ in 1:n]
X, Y = first.(XY), last.(XY)
@show cov(X, Y) cor(X, Y)
P1 = scatter(X, Y; msc=:auto, label="", xlabel="x", ylabel="y")
plot!(; xlim=(-1.05, 1.05), ylim=(-1.05, 1.05))
plot!(; size=(400, 400))
```

```
cov(X, Y) = 0.011653127995250636
cor(X, Y) = 0.046497931398409825
```

Out[32]:

上の場合において X, Y はどちらも **半円分布** (semicircle distribution)に従っているが, 独立ではない. X, Y の分布が以上と同じ半円分布のときと X, Y が独立になるならば, (X, Y) の分布のサンプルは以下のようになる.

In [33]:

```
n = 10^4
sc = Semicircle(1)
X, Y = rand(sc, n), rand(sc, n)
@show cov(X, Y) cor(X, Y)
P1 = scatter(X, Y; msc=:auto, ms=2, alpha=0.7, label="", xlabel="x", ylabel="y")
P2 = histogram(X; norm=true, alpha=0.3, bin=41, label="X")
plot!(x -> 2/π*√(1 - x^2), -1, 1; label="", lw=2)
P3 = histogram(X; norm=true, alpha=0.3, bin=41, label="Y")
plot!(y -> 2/π*√(1 - y^2), -1, 1; label="", lw=2)
plot(P1, P2, P3; size=(800, 400), layout=@layout [a [b; c]])
```

```
cov(X, Y) = 0.004473060988369731
cor(X, Y) = 0.0179942903137646
```

Out[33]:

これは X, Y が独立な場合であり, ゆえに無相関にもなっている.

サンプルが含む点の個数を減らすと以下のように見える.

In [34]:

```
n = 100
sc = Semicircle(1)
X, Y = rand(sc, n), rand(sc, n)
@show cov(X, Y)
P1 = scatter(X, Y; msc=:auto, label="", xlabel="x", ylabel="y")
plot!(; xlim=(-1.05, 1.05), ylim=(-1.05, 1.05))
plot!(; size=(400, 400))
```

```
cov(X, Y) = 0.0030840296722668315
```

Out[34]:

2.10 問題: 対ごとに無相関な確率変数達の和の分散はそれぞれの分散の和になる

$X_1, \ldots, X_n$ は対ごとにも無相関とは限らない(ゆえに独立とは限らない)確率変数達であるとする. このとき, 期待値を取る操作の線形性より,

$$E[X_1 + \cdots + X_n] = E[X_1] + \cdots + E[X_n]$$

となる. 確率変数達の和の期待値は, 独立性や無相関性が成立していなくても, それぞれの期待値の和になる.

対ごとの無相関性を仮定すると, 分散についても簡明な結果を得ることができる.

$X_1, \ldots, X_n$ が **対ごとに無相関な確率変数達** ならば(特に独立な確率変数ならば), 次が成立することを示せ:

$$\mathrm{var}(X_1 + \cdots + X_n) = \mathrm{var}(X_1) + \cdots + \mathrm{var}(X_n).$$

この結果は今後空気のごとく使われる.

ヒント: 互いに直交するベクトル達 $v_1, \ldots, v_n$ について, 内積を $(\ ,\ )$ と書くとき, $(v_i, v_j) = \delta_{ij}\|v_i\|^2$ が成立することを使えば, $\|v_1 + \cdots + v_n\|^2 = \|v_1\|^2 + \cdots + \|v_n\|^2$ を示せることと本質的に同じ. この結果はPythagorasの定理(平面の場合は三平方の定理)そのものである.

解答例: $\mu_i = E[X_i]$ とおくと,

$$E\left[\sum_{i=1}^n X_i\right] = \sum_{i=1}^n E[X_i] = \sum_{i=1}^n \mu_i$$

が成立しており, さらに, $\sigma_i^2 = \mathrm{var}(X_i) = E[(X_i - \mu_i)^2]$ とおくと, $X_1, \ldots, X_n$ が対ごとに無相関であることより,

$$E[(X_i-\mu_i)(X_j-\mu_j)] = \mathrm{cov}(X_i, X_j) = \sigma_i^2\delta_{ij}$$

が成立しているので(δ_{ij} は $i=j$ の場合にのみ 1 でそれ以外のとき 0), 一般に

$$\left(\sum_{i=1}^n a_i\right)^2 = \sum_{i=1}^n a_i \sum_{i=j}^n a_j = \sum_{i,j=1}^n a_i a_j$$

と計算できることを使うと,

$$\begin{aligned}
\mathrm{var}\left(\sum_{i=1}^n X_i\right) &= E\left[\left(\sum_{i=1}^n X_i - \sum_{i=1}^n \mu_i\right)^2\right] = E\left[\left(\sum_{i=1}^n (X_i-\mu_i)\right)^2\right] \\
&= E\left[\sum_{i,j=1}^n (X_i-\mu_i)(X_j-\mu_j)\right] = \sum_{i,j=1}^n E\left[(X_i-\mu_i)(X_j-\mu_j)\right] \\
&= \sum_{i,j=1}^n \sigma_i^2\delta_{ij} = \sum_{i=1}^n \sigma_i^2 = \sum_{i=1}^n \mathrm{var}(X_i).
\end{aligned}$$

解答終

2.11 問題(メタアナリシスの出発点): 共通の期待値と異なる分散を持つ確率変数の荷重平均

$X_1, \ldots, X_n$ は対ごとに無相関な確率変数達であるとし, X_i 達は共通の期待値 μ と共通でない分散 $\sigma_i^2 > 0$ を持つと仮定する:

$$E[X_1] = \cdots = E[X_n] = \mu, \quad E[X_i] = \sigma_i^2 \quad (i = 1, \ldots, n)$$

このとき, 共通の期待値 μ の推定量 $\hat{\mu}$ を荷重 w_i による荷重平均

$$\hat{\mu} = \sum_{i=1}^n w_i X_i, \quad \sum_{i=1}^n w_i = 1$$

の形で定義したい. 条件 $\sum_{i=1}^n w_i = 1$ より, 不偏性 $E[\hat{\mu}] = \mu$ がただちに導かれる. このとき, $\hat{\mu}$ の分散 $\mathrm{var}(\hat{\mu})$ を最小化する荷重 w_i は

$$w_i = \frac{1/\sigma_i^2}{\sum_{i=1}^n 1/\sigma_i^2}, \quad \text{i.e.} \quad \hat{\mu} = \frac{\sum_{i=1}^n X_i/\sigma_i^2}{\sum_{i=1}^n 1/\sigma_i^2}$$

になることを示せ. さらに, このとき

$$\mathrm{var}(\hat{\mu}) = \frac{1}{\sum_{i=1}^n 1/\sigma_i^2} < \min(\sigma_1^2, \ldots, \sigma_n^2)$$

となることも示せ.

注意: この結果はことなる分散を持つ複数の推定量をうまくまとめることによって, 分散がより小さな推定量を作れることを意味している. これは同一の量に関する異なる研究の結果をまとめるメタアナリシスの出発点になる.

解答例1: まず $\hat{\mu}$ の分散の式を整理しよう:

$$\mathrm{var}(\hat{\mu}) = E\left[\left(\sum_{i=1}^n w_i X_i - \mu\right)^2\right] = E\left[\left(\sum_{i=1}^n w_i (X_i - \mu)\right)^2\right] = \sum_{i=1}^n w_i^2 \sigma_i^2.$$

2つめの等号で $\sum_{i=1}^n w_i = 1$ であることを使い, 3つめの等号で $w_i X_i$ 達が対ごとに無相関であることを使った(一般に対ごとに無相関な確率変数の和の分散がそれぞれの分散の和になるのであった). $w_n = 1 - \sum_{i=1}^{n-1} w_i$ とおいて, $w_1, \ldots, w_{n-1}$ を独立変数とみなすとき, それらによる $\mathrm{var}(\hat{\mu})$ の偏微分が 0 になるという方程式の解が求める w_i 達になる. $w_n = 1 - \sum_{i=1}^{n-1} w_i$ に注意すれば, $i = 1, \ldots, n-1$ について

$$0 = \frac{1}{2}\frac{\partial}{\partial w_i}\mathrm{var}(\hat{\mu}) = w_i\sigma_i^2 - \left(1 - \sum_{i=1}^{n-1} w_i\right)\sigma_i^2 = w_i\sigma_i^2 - w_n\sigma_i^2$$

なので, この方程式の解について, $w_i\sigma_i$ 達は互いにすべて等しくなり, w_i 達の総和は 1 になる. そのとき, 以下のようになることがわかる:

$$w_i = \frac{1/\sigma_i^2}{\sum_{i=1}^n 1/\sigma_i^2}, \quad \mathrm{var}(\hat{\mu}) = \sum_{i=1}^n w_i^2\sigma_i^2 = \frac{1}{\sum_{i=1}^n 1/\sigma_i^2} < \frac{1}{1/\sigma_i^2} = \sigma_i^2.$$

解答終

答えが分かっているので, 以下のように平方完成の方法によって偏微分を使わずにも証明できる.

解答例2: まず,

$$A = \sum_{i=1}^{n} 1/\sigma_i^2$$

とおいて, $\sum_{i=1}^{n} w_i = 1$ の仮定のもとで, $\hat{\mu}$ の分散の式を整理しよう:

$$\begin{aligned}
\mathrm{var}(\hat{\mu}) &= E\left[\left(\sum_{i=1}^{n} w_i X_i - \mu\right)^2\right] = E\left[\left(\sum_{i=1}^{n} w_i (X_i - \mu)\right)^2\right] \\
&= E\left[\sum_{i=1}^{n} w_i^2 (X_i - \mu)^2 + \sum_{i \neq j} w_i w_j (X_i - \mu)(X_j - \mu)\right] \\
&= \sum_{i=1}^{n} w_i^2 E[(X_i - \mu)^2] + \sum_{i \neq j} w_i w_j E[(X_i - \mu)(X_j - \mu)] \\
&= \sum_{i=1}^{n} w_i^2 E[(X_i - \mu)^2] = \sum_{i=1}^{n} w_i^2 \sigma_i^2 = \sum_{i=1}^{n} \sigma_i^2 \left(w_i - \frac{1/\sigma_i^2}{A} + \frac{1/\sigma_i^2}{A}\right)^2 \\
&= \sum_{i=1}^{n} \sigma_i^2 \left(\left(w_i - \frac{1/\sigma_i^2}{A}\right)^2 - 2\frac{1/\sigma_i^2}{A}\left(w_i - \frac{1/\sigma_i^2}{A}\right) + \frac{(1/\sigma_i^2)^2}{A^2}\right) \\
&= \sum_{i=1}^{n} \sigma_i^2 \left(w_i - \frac{1/\sigma_i^2}{A}\right)^2 + \frac{1}{A}.
\end{aligned}$$

2つめの等号で $\sum_{i=1}^{n} w_i = 1$ であることを使い, 5つめの等号で X_i 達が対ごとに無相関であること($i \neq j$ ならば $E[(X_i - \mu)(X_j - \mu)]$)を使い, 最後の等号で $\sum_{i=1}^{n} w_i = 1$ と $\sum_{i=1}^{n} 1/\sigma_i^2 = A$ を使った.

このとき, $\sum_{i=1}^{n} (1/\sigma_i^2)/A = 1$ に注意すれば, $\mathrm{var}(\hat{\mu})$ は $\sum_{i=1}^{n} w_i = 1$ という条件のもとで $w_i = (1/\sigma_i^2)/A$ のときに最小になり, 最小値は $1/A$ になることがわかる. そして, さらに,

$$\frac{1}{A} = \frac{1}{\sum_{i=1}^{n} 1/\sigma_i^2} < \frac{1}{1/\sigma_i^2} = \sigma_i^2.$$

解答終

以下の図も参照:

原点からの距離は誤差の大きさ,
直交は無相関を意味する.

一般に $E[X_1] = \cdots = E[X_n] = \mu$, $E[(X_i-\mu)(X_j-\mu)] = \delta_{ij}\sigma_i^2$ のとき,
$\sum_{i=1}^n w_i = 1$ のときの $\hat{\mu} = \sum_{i=1}^n w_i X_i$ の分散は $w_i = \frac{1/\sigma_i^2}{\sum_{j=1}^n 1/\sigma_j^2}$ のとき
最小になり, その最小値は $\frac{1}{\sum_{j=1}^n 1/\sigma_j^2} < \sigma_i^2$ になる.

2.12 問題: 二項分布と負の二項分布の平均と分散のBernoulli分布と幾何分布の場合への帰着

Bernoulli分布 $\mathrm{Bernoulli}(p)$ の平均と分散がそれぞれ $p, p(1-p)$ であることと, 幾何分布 $\mathrm{Geometric}(p)$ の平均と分散がそれぞれ $(1-p)/p$, $(1-p)/p^2$ であることを認めて, 二項分布 $\mathrm{Binomial}(n,p)$ と負の二項分布 $\mathrm{NegativeBinomial}(k,p)$ の平均と分散を平易な計算で求めてみよ. 以下を示せ:

(1) $K \sim \mathrm{Binomial}(n,p)$ ならば $E[K] = np$, $\mathrm{var}(K) = np(1-p)$.

(2) $M \sim \mathrm{NegativeBinomial}(k,p)$ ならば $E[M] = k(1-p)/p$, $\mathrm{var}(M) = k(1-p)/p^2$.

解答例:

二項分布は試行回数 n の成功確率 p のBernoulli試行で生成された 1 と 0 からなる長さ n の数列中に含まれる 1 の個数の分布であった. Bernoulli試行の確率質量函数は

$$P(x_1,\ldots,x_n) = P(x_1)\cdots P(x_n), \quad P(x_i) = p^{x_i}(1-p)^{1-x_i} \quad (x_i = 1, 0)$$

とBernoulli分布の確率質量函数 $P(x_i)$ の **積で書ける** のであった. この事実はBernoulli分布に従う確率変数達 $X_1,\ldots,X_n$ が独立であることを意味する. そして, Bernoulli試行で生成された 1 と 0 からなる長さ n の数列中に含まれる 1 の個数を意味する確率変数は $K = \sum_{i=1}^n X_i$ と書ける. このことから,

$$E[K] = \sum_{i=1}^n E[X_i] = \sum_{i=1}^n p = np,$$
$$\mathrm{var}(K) = \sum_{i=1}^n \mathrm{var}(X_i) = \sum_{i=1}^n p(1-p) = np(1-p).$$

となることがわかる.

幾何分布は成功確率 p のBernoulli試行を 1 が1つ出るまで続けたときに出た 0 の個数の分布であった. $M_1, \ldots, M_k$ はそれぞれが成功確率 p の幾何分布に従う独立な確率変数であるとする. このとき, $M = \sum_{i=1}^k M_i$ はBernoulli試行を 1 が k 回出るまで続けたときに 0 が出た個数に等しい. M_i は $i-1$ 番目の 1 から i 番目の 1 が出るまでに出た 0 の個数を意味する確率変数だと解釈される. このことは, M が負の二項分布 $\mathrm{NegativeBinomial}(k, p)$ に従う確率変数になることを意味する. このことから,

$$
\begin{aligned}
E[M] &= \sum_{i=1}^k E[M_i] = \sum_{i=1}^k \frac{1-p}{p} = \frac{k(1-p)}{p}, \\
\mathrm{var}(M) &= \sum_{i=1}^k \mathrm{var}(M_i) = \sum_{i=1}^k \frac{1-p}{p^2} = \frac{k(1-p)}{p^2}.
\end{aligned}
$$

となることがわかる.

解答終

注意: 計算が大幅に簡単になった!

2.13 問題: 番号が異なる確率変数達が対ごとに無相関なときの確率変数の和の共分散

確率変数達 $X_1, Y_1, \ldots, X_n, Y_n$ について次が成立していると仮定する:

$$
\mathrm{cov}(X_i, Y_j) = \delta_{ij} \mathrm{cov}(X_i, Y_i).
$$

このとき, 次が成立することを示せ:

$$
\mathrm{cov}(X_1 + \cdots + X_n, Y_1 + \cdots + Y_n) = \mathrm{cov}(X_1, Y_1) + \cdots + \mathrm{cov}(X_n, Y_n).
$$

ヒント: ベクトル達 $u_1, v_2, \ldots, u_n, v_n$ の中の2つの異なる添え字を持つ u_i と v_j が互いに直交するならば, 内積 $(\ ,\ )$ について

$$
(u_1 + \cdots + u_n, v_1 + \cdots + v_n) = (u_1, v_1) + \cdots + (u_n, v_n)
$$

が成立することと本質的に同じことである.

解答例:

記号の簡単のため $A_i = X_i - E[X_i]$, $B_i = Y_i - E[Y_i]$ とおく. このとき, $E[A_i] = E[B_i] = 0$ より,

$$
\mathrm{var}(A_i) = E[A_i^2], \quad \mathrm{var}(B_i) = E[B_i^2], \quad \mathrm{cov}(A_i, B_j) = E[A_i B_j].
$$

A_i, B_i 達について上の問題を解けばよい. 問題の仮定より, A_i, B_i 達について次が成立している:

$$
E[A_i B_j] = \delta_{ij} E[A_i B_i].
$$

ゆえに,

$$
E\left[\left(\sum_{i=1}^n A_i\right)\left(\sum_{j=1}^n B_j\right)\right] = \sum_{i,j=1}^n E[A_i B_j] = \sum_{i,j=1}^n \delta_{ij} E[A_i B_i] = \sum_{i=1}^n E[A_i B_i].
$$

これは

$$
\mathrm{cov}(A_1 + \cdots + A_n, B_1 + \cdots + B_n) = \mathrm{cov}(A_1, B_1) + \cdots + \mathrm{cov}(A_n, B_n).
$$

の成立を意味する.

解答終

2.14 分散共分散行列とその半正定値性

n 個の確率変数を縦に並べてできるベクトル X と実数を成分に持つベクトル a を考える:

$$
X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}, \quad a = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \in \mathbb{R}^n.
$$

このとき, X, a の転置をそれぞれ X, a^T と書くと,

$$
a^T X = X^T a = \sum_{i=1}^n a_i X_i
$$

なので

$$a^T XX^T a = \left(\sum_{i=1}^n a_i X_i \right)^2 \geq 0.$$

ゆえに, 期待値を取る操作の線形性と単調性より,

$$0 \leq E[a^T XX^T a] = a^T E[XX^T] a$$

でかつ, XX^T は $X_i X_j$ を (i,j) 成分とする $n \times n$ 行列になるので, $E[XX^T]$ は $E[X_i X_j]$ を (i,j) 成分とする $n \times n$ の実対称行列になる. 上の計算より, その実対称行列の固有値はすべて 0 以上になることがわかる.

ここで $E[X_i] = 0$ と仮定する. このとき, $E[XX^T]$ は i 番目の対角成分が X_i の分散 $\mathrm{var}(X_i)$ で, $i \neq j$ のときの (i,j) 成分が共分散 $\mathrm{cov}(X_i, X_j)$ であるような $n \times n$ の実対称行列になる.

このとき, 行列 $E[XX^T]$ を **分散共分散行列** と呼ぶ.

分散共分散行列の固有値はすべて 0 以上になる. この結果を分散共分散行列の **半正定値性** と呼ぶことにする. 特に分散共分散行列式は 0 以上になる.

3 標本(サンプル, データ)の平均と分散と共分散と相関係数

3.1 標本平均の定義

n 個の数値 $x_1, \ldots, x_n$ で構成されたデータ(標本, サンプル)に対して,

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + \cdots + x_n}{n}$$

をその **標本平均** (sample mean)と呼ぶ.

統計モデルを設定するとは現実世界で得られるデータ(標本, サンプル)の生じ方を確率分布の言葉でモデル化することである. そのとき, 統計モデル内部では, データ(標本, サンプル)を, n 個の数値ではなく, n 個の確率変数達 $X_1, \ldots, X_n$ でモデル化することになる. その場合には, 確率変数達の加法平均

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i = \frac{X_1 + \cdots + X_n}{n}$$

をその **標本平均** (sample mean)と呼ぶ. X_i 達が確率変数であることより, それらの標本平均 $\bar{X}$ も確率変数になることに注意せよ. この文脈での標本平均 $\bar{X}$ は確率モデル内部における確率変数になる.

統計学入門の文脈では, 標本のモデル化とみなされるn 個の確率変数達 $X_1, \ldots, X_n$ は独立同分布だと仮定することが多い.

以下では, X_i 達は同じ期待値と分散を持ち, 対ごとに無相関であると仮定する(この条件は独立同分布よりも弱い).

注意・警告: 標本のモデル化としての確率変数達の **標本平均と確率変数の平均(期待値)を明瞭に区別しなければいけない**. 各確率変数 X_i の期待値 $E[X_i]$ や確率変数としての標本平均 $\bar{X}$ の期待値

$$E[\bar{X}] = E\left[\frac{1}{n} \sum_{i=1}^n X_i \right] = \frac{1}{n} \sum_{i=1}^n E\left[X_i\right] = \mu$$

と標本平均 $\bar{X}$ を明瞭に区別することが必要になる.

統計モデル内での標本のモデル化としての確率変数達 $X_1, \ldots, X_n$ の標本平均 $\bar{X}$ は確率変数なので, 数学的には函数とみなされ, 直観的にはランダムに値が決まる変数だとみなされるものになる. それに対して, 標本平均 $\bar{X}$ の期待値 $E[\bar{X}]$ は単なる数値になる. このように, 確率変数としての標本平均 $\bar{X}$ とその期待値 $E[\bar{X}]$ は全く異なる数学的対象になる.

注意: 上で期待値を取る操作 $E[\,]$ の線形性を用いて, 標本平均の期待値が X_i 達の共通の期待値 μ に一致すること

$$E[\bar{X}] = \mu$$

も示されていることに注意せよ. これは標本平均を取る操作が平均の **不偏推定量** を作る操作になっていることも意味している. この点については後で詳しく説明する.

3.2 問題: 対ごとに無相関な確率変数達の標本平均の分散

同じ期待値 μ と分散 σ^2 を持つ対ごとに無相関な n 個の確率変数達 $X_1, \ldots, X_n$ の標本分散 $\bar{X}$ について次が成立していることを示せ:

$$\mathrm{var}(\bar{X}) = \frac{\sigma^2}{n}$$

注意: この問題で扱っている確率変数としての $\bar{X}$ の分散と次の節で説明する標本のモデル化 $X_1, \ldots, X_n$ の分散は異なる数学的対象であることに注意せよ.

注意: $\mathrm{std}(\bar{X}) = \sigma/\sqrt{n}$ は **標準誤差** (standard error)と呼ばれ, SE と書かれることがある. 未知の真の標準誤差の推定量をも標準誤差と呼ぶことがあるが, このノートではそのスタイルを採用せずに, 未知の真の標準誤差の推定量を「標準誤差の推定量」と丁寧に呼ぶことにする.

解答例: 対ごとに無相関な確率変数達の和の分散はそれぞれの確率変数の分散の和になるのであった. X_i 達が対ごとに無相関であるという仮定より, X_i/n 達も対ごとに無相関になる. そして,

$$\bar{X} = \sum_{i=1}^{n} \frac{X_i}{n}$$

であることより,

$$\mathrm{var}(\bar{X}) = \sum_{i=1}^{n} \mathrm{var}\left(\frac{X_i}{n}\right) = \sum_{i=1}^{n} \frac{\mathrm{var}(X_i)}{n^2} = \sum_{i=1}^{n} \frac{\sigma^2}{n^2} = \frac{\sigma^2}{n}.$$

解答終

3.3 標本分散と不偏分散の定義

n 個の数値 $x_1, \ldots, x_n$ で構成されたデータ(標本, サンプル)の標本平均を

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + \cdots + x_n}{n}$$

と書くとき,

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

を **標本分散** (sample variance, 標本の無補正分散)と呼ぶ. **慣習的には** 補正無し標本分散が使用されることは少なく, 次のように定義される **不偏分散** (unbiased variance, 標本の不偏補正分散)を使うことが多い:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

どうして n ではなく $n-1$ で割るかについては後で説明する. このとき,

$$\overline{x^2} = \frac{1}{n}\sum_{i=1}^{n} x_i^2$$

とおくと,

$$\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sum_{i=1}^{n} x_i^2 - 2\bar{x}\sum_{i=1}^{n} x_i + n\bar{x}^2 = n\overline{x^2} - 2n\bar{x}^2 + n\bar{x}^2 = n\left(\overline{x^2} - \bar{x}^2\right)$$

なので,

$$s^2 = \frac{n}{n-1}\left(\overline{x^2} - \bar{x}^2\right).$$

$X_1, \ldots, X_n$ は標本のモデル化とみなされる n 個の確率変数達であるとする. X_i 達は同じ期待値と分散を持ち, 対ごとに無相関であると仮定し, それらの標本平均を

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i = \frac{X_1 + \cdots + X_n}{n}$$

と書く. このとき, $X_1, \ldots, X_n$ の **不偏分散** S^2 を次のように定義する:

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2.$$

このとき,

$$\overline{X^2} = \frac{1}{n}\sum_{i=1}^{n} x_i^2$$

とおくと,

$$\sum_{i=1}^{n}(X_i - \bar{X})^2 = \sum_{i=1}^{n} X_i^2 - 2\bar{X}\sum_{i=1}^{n} X_i + n\bar{X}^2 = n\overline{X^2} - 2n\bar{X}^2 + n\bar{X}^2 = n\left(\overline{X^2} - \bar{X}^2\right)$$

なので,

$$S^2 = \frac{n}{n-1}\left(\overline{X^2} - \bar{X}^2\right).$$

この場合には, 不偏分散 S^2 は確率変数になることに注意せよ. この S^2 と単なる数値になる $\bar{X}$ の確率変数としての分散 $\mathrm{var}(\bar{X})$ は異なる数学的対象になるので混乱しないようにして欲しい.

3.4 不偏推定量について: 不偏分散の定義ではどうして n ではなく n-1 で割るか

$X_1, \ldots, X_n$ に共通の期待値と分散をそれぞれ μ, σ^2 と書くことにする. $X_1, \ldots, X_n$ を現実で得られる標本のモデル化とみなす. そのとき, 統計モデル内での μ, σ^2 の値は未知の真の平均と分散の値であって欲しい. この設定では未知の μ と σ^2 を標本から推定する方法が欲しくなる.

未知の μ の推定量として優れているのは標本平均である. 標本のモデル化とみなされる $X_1, \ldots, X_n$ の標本平均 $\bar{X}$ についてはその期待値 $E[\bar{X}]$ が未知の真の平均値 μ に等しくなる. この性質を標本平均の **不偏性** という. 未知の値 θ の推定量を $E[T] = \theta$ を満たす確率変数 T とするとき, T は θ の **不偏推定量** (unbiased estimator) と呼ぶ.

次に, 不偏分散の定義でどうして n ではなく $n-1$ で割るかを理解するために, $n-1$ で割る前の量の期待値を計算してみよう.

$X_1, \ldots, X_n$ 達の標本平均 $\bar{X}$ の期待値と分散がそれぞれ次のようになることはわかっている:

$$E[\bar{X}] = \mu, \quad \mathrm{var}(\bar{X}) = E[(\bar{X} - \mu)^2] = \frac{\sigma^2}{n}.$$

そして, 以下が成立している:

$$\begin{aligned}
\sum_{i=1}^{n}(X_i - \bar{X})^2 &= \sum_{i=1}^{n}((X_i - \mu) - (\bar{X} - \mu))^2 \\
&= \sum_{i=1}^{n}\left((X_i - \mu)^2 - 2(X_i - \mu)(\bar{X} - \mu) + (\bar{X} - \mu)^2)^2\right) \\
&= \sum_{i=1}^{n}(X_i - \mu)^2 - 2\left(\sum_{i=1}^{n}(X_i - \mu)\right)(\bar{X} - \mu) + n(\bar{X} - \mu)^2)^2 \\
&= \sum_{i=1}^{n}(X_i - \mu)^2 - 2n(\bar{X} - \mu)^2 + n(\bar{X} - \mu)^2)^2 \\
&= \sum_{i=1}^{n}(X_i - \mu)^2 - n(\bar{X} - \mu)^2.
\end{aligned}$$

4つめの等号で $\sum_{i=1}^{n} X_i = n\bar{X}$ を用いた. ゆえに,

$$E\left[\sum_{i=1}^{n}(X_i - \bar{X})^2\right] = \sum_{i=1}^{n} E[(X_i - \mu)^2] - nE[(\bar{X} - \mu)^2] = n\sigma^2 - n\frac{\sigma^2}{n} = (n-1)\sigma^2.$$

これより, σ^2 の不偏推定量を作るためには $\sum_{i=1}^{n}(X_i - \bar{X})^2$ を $n-1$ で割らなければいけないことがわかる. 実際にそのように定義した σ^2 の不偏推定量が不偏分散 S^2 である:

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2, \quad E[S^2] = \sigma^2.$$

このように, $\sum_{i=1}^{n}(X_i - \bar{X})^2$ の確率変数としての分散を計算して $(n-1)\sigma^2$ になることを確認すれば, n ではなく $n-1$ で割る理由は明瞭である.

注意: 不偏分散の平方根 $\sqrt{S^2}$ は標準偏差 σ の不偏推定量になって **いない** ことに注意せよ. 推定量の不偏性は座標に依存する.

標準正規分布の場合に不偏分散と補正されていない標本分散の分布を同時プロットしてみよう.

In [35]:
```
n = 10
L = 10^5
X = rand(Normal(), n, L)
S² = var.(eachcol(X))
BV = (n-1)/n * S²
stephist(S²; norm=true, label="unbiased", c=1)
vline!([mean(S²)]; label="E[unbiased]", c=:blue)
stephist!(BV; norm=true, label="biased", c=2)
vline!([mean(BV)]; label="E[biased]", c=:red)
vline!([1]; label="true variance", c=:black, ls=:dash)
```

Out[35]:

確かに不偏分散(unbiased variance)の分布の平均値は真の値に一致しているが, 補正されていない標本分散(biased)の平均値は真の値よりも少し小さくなっている.

さらにこのグラフから, 不偏分散の場合も非補正標本分散の場合も確率密度函数は真の値よりも小さな値で最大になることにも注意せよ.

注意: 標準正規分布のサイズ n の標本分布において, 不偏分散の $n-1$ 倍は自由度 $n-1$ のχ²分布 $\mathrm{Chisq}(n-1)$ に従うことを示せる. 分布 $\mathrm{Chisq}(n-1)$ の期待値は $n-1$ なので, その $n-1$ 分の 1 の期待値は 1 になる.

注意: 標準誤差の推定量 として

$$\widehat{\mathrm{SE}} = \sqrt{\frac{S^2}{n}} = \sqrt{\frac{1}{n(n-1)}\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

を採用しよう. このとき,

$$E\left[\widehat{\mathrm{SE}}^2\right] = \frac{E[S^2]}{n} = \frac{\sigma^2}{n} = \mathrm{var}(\bar{X}) = \mathrm{SE}^2$$

なので, $\widehat{\mathrm{SE}}^2$ は標準誤差の2乗 $\mathrm{SE}^2 = \mathrm{var}(\bar{X}) = \sigma^2/n$ の不偏推定量になっている. ただし, $\widehat{\mathrm{SE}}$ 自身は標準誤差 SE の不偏推定量になって **いない** ことに注意せよ. 推定量の不偏性は座標に依存する.

注意: 不偏推定量の方が不偏でない推定量より常に優れていると考えるのは誤りである. 推定量に不偏性の条件を課すと平均二乗誤差が大きくなってしまうことが多い(不偏分散についても実際にそうなることを後で示す). 不偏推定量を使うか否かはそうすることのメリットとデメリットの両方を考慮に入れて決定するべきである. ただし, 上で紹介した不偏分散については害は相当に小さく, 慣習的に非常によく使われている.

注意: $X_1, \ldots, X_n$ は標本のモデル化とみなされる独立同分布な確率変数達であるとし, X_i 達に共通の期待値と分散をそれぞれ μ, σ^2 と書くことにする. このとき, その標本平均 $\bar{X} = (1/n)\sum_{i=1}^{n} X_i$ の期待値と分散がそれぞれ

$$E[\bar{X}] = \mu, \quad \mathrm{var}(\bar{X}) = \frac{\sigma^2}{n}$$

となり, 不偏分散 $S^2 = (1/(n-1))\sum_{i=1}^{n}(X_i - \bar{X})^2$ の期待値が

$$E[S^2] = \sigma^2$$

となることまではすでにわかっている. この一連のノートではほぼ常に与えられた確率変数の期待値と分散を同時に計算して来た. ここでも, 確率変数としての不偏分散の分散も計算するべきである. そして, さらに標本平均と不偏分散の共分散も計算するべきである. しかし, そのためには X_i 達共通の分布の **歪度** (わいど, skewness)と **尖度** (せんど, kurtosis)の定義を先にしておいた方がよい. 次の節で歪度と尖度を定義する.

3.5 データの共分散の定義

n 個の数値の対 $(x_1, y_1), \dots, (x_n, y_n)$ で構成されたデータ(標本, サンプル)について, x_i, y_i 達各々の標本平均を

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

と書くとき,

$$\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

を **標本共分散** (sample covariance, 標本の無補正共分散)と呼ぶ. **慣習的には** 補正無しの標本共分散が使用されることは少なく, 次のように定義される **不偏共分散** (unbiased variance, 標本の不偏補正共分散)を使うことが多い:

$$s_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}).$$

n ではなく $n-1$ で割る理由は不偏分散の場合と同様である. 後で詳しく説明する. このとき,

$$\overline{xy} = \frac{1}{n}\sum_{i=1}^{n} x_i y_i$$

とおくと,

$$\begin{aligned}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) &= \sum_{i=1}^{n} x_i y_i - \bar{x}\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i\,\bar{y} + n\bar{x}\bar{y} \\ &= n\overline{xy} - n\bar{x}\bar{y} - n\bar{x}\bar{y} + n\bar{x}\bar{y} = n\left(\overline{xy} - \bar{x}\bar{y}\right)\end{aligned}$$

なので,

$$s_{xy} = \frac{n}{n-1}\left(\overline{xy} - n\bar{x}\bar{y}\right).$$

$(X_1, Y_1), \dots, (X_n, Y_n)$ は上の型のデータのモデル化とみなされる n 個の確率変数対達であるとする. X_i 達は対ごとに無相関で共通の同じ期待値 μ_X と分散 σ_X^2 を持ち, Y_i 達も対ごとに無相関で共通の同じ期待値 μ_Y と分散 σ_Y^2 を持ち, X_i と Y_i の共分散は i によらない共通の値 σ_{XY} になっているとし, 異なる番号 i, j について X_i と Y_j は無相関であると仮定する. これらの仮定は次のように式で書ける:

$$\begin{aligned}&E[X_i] = \mu_X, \quad E[Y_i] = \mu_Y \\ &E[(X_i - \mu_X)(X_j - \mu_X)] = \delta_{ij}\sigma_X^2, \\ &E[(Y_i - \mu_Y)(Y_j - \mu_Y)] = \delta_{ij}\sigma_Y^2, \\ &E[(X_i - \mu_X)(Y_j - \mu_Y)] = \delta_{ij}\sigma_{XY}.\end{aligned}$$

それらの X_i 達と Y_i 達それぞれの標本平均を次のように書くことにする:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i.$$

このとき, $(X_1, Y_1), \dots, (X_n, Y_n)$ の **不偏共分散** S_{XY} を次のように定義する:

$$S_{XY} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})$$

このとき,

$$\overline{XY} = \frac{1}{n}\sum_{i=1}^{n} X_i Y_i$$

とおくと,

$$
\begin{aligned}
\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y}) &= \sum_{i=1}^{n}X_iY_i - \bar{X}\sum_{i=1}^{n}Y_i - \sum_{i=1}^{n}X_i\bar{Y} + n\bar{X}\bar{Y} \\
&= n\overline{XY} - n\bar{X}\bar{Y} - n\bar{X}\bar{Y} + n\bar{X}\bar{Y} = n\left(\overline{XY} - \bar{X}\bar{Y}\right)
\end{aligned}
$$

なので,

$$
s_{XY} = \frac{n}{n-1}\left(\overline{XY} - n\bar{X}\bar{Y}\right).
$$

この場合には, 不偏分散 S_{XY} は確率変数になることに注意せよ. この S_{XY} と単なる数値になる $\bar{X}, \bar{Y}$ の確率変数としての共分散 $\mathrm{cov}(\bar{X}, \bar{Y})$ は異なる数学的対象になるので混乱しないようにして欲しい.

3.6 問題: 標本平均達の共分散

前節の設定のもとで次が成立することを示せ:

$$
\mathrm{cov}(\bar{X}, \bar{Y}) = \frac{\sigma_{XY}}{n}.
$$

解答例: 標本平均の分散の計算とほぼ同じになる.

番号が異なる確率変数達が対ごとに無相関なときの確率変数達の和の共分散は各々の対の共分散の和になるのであった. これを確率変数の和達

$$
\bar{X} = \sum_{i=1}^{n}\frac{X_i}{n}, \quad \bar{Y} = \sum_{i=1}^{n}\frac{Y_i}{n}
$$

に適用すると,

$$
\mathrm{cov}(\bar{X}, \bar{Y}) = \sum_{i=1}^{n}\mathrm{cov}\left(\frac{X_i}{n}, \frac{Y_i}{n}\right) = \sum_{i=1}^{n}\frac{\mathrm{cov}(X_i, Y_i)}{n^2} = \sum_{i=1}^{n}\frac{\sigma_{XY}}{n^2} = \frac{\sigma_{XY}}{n}.
$$

解答終

3.7 問題: 不偏共分散の定義ではどうして n ではなく n-1 で割るか

前々の設定のもとで, 不偏共分散 S_{XY} が X_i, Y_i の共分散 σ_{XY} の不偏推定量になっていることを示せ. すなわち次が成立することを示せ:

$$
E[S_{XY}] = \sigma_{XY}.
$$

注意: これが不偏共分散の定義で n ではなく $n-1$ で割る理由である.

解答例: 不偏分散の場合の議論とほぼ同じ.

$\bar{X}, \bar{Y}$ の期待値と共分散がそれぞれ次のようになることはわかっている:

$$
E[\bar{X}] = \mu_X, \quad E[\bar{Y}] = \mu_Y, \quad \mathrm{cov}(\bar{X}, \bar{Y}) = E[(\bar{X}-\mu_Y)(\bar{Y}-\mu_Y)] = \frac{\sigma_{XY}}{n}.
$$

そして, 以下が成立している:

$$
\begin{aligned}
\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y}) &= \sum_{i=1}^{n}((X_i-\mu_X)-(\bar{X}-\mu_X))((Y_i-\mu_Y)-(\bar{Y}-\mu_Y)) \\
&= \sum_{i=1}^{n}\big((X_i-\mu_X)(Y_i-\mu_Y)-(X_i-\mu_X)(\bar{Y}-\mu_Y) \\
&\qquad -(\bar{X}-\mu_X)(Y_i-\mu_Y)+(\bar{X}-\mu_X)(\bar{Y}-\mu_Y)\big) \\
&= \sum_{i=1}^{n}(X_i-\mu_X)(Y_i-\mu_Y)-\left(\sum_{i=1}^{n}(X_i-\mu_X)\right)(\bar{Y}-\mu_Y) \\
&-(\bar{X}-\mu_X)\left(\sum_{i=1}^{n}(Y_i-\mu_Y)\right)+n(\bar{X}-\mu_X)(\bar{Y}-\mu_Y) \\
&= \sum_{i=1}^{n}(X_i-\mu_X)(Y_i-\mu_Y)-n(\bar{X}-\mu_X)(\bar{Y}-\mu_Y) \\
&-n(\bar{X}-\mu_X)(\bar{Y}-\mu_Y)+n(\bar{X}-\mu_X)(\bar{Y}-\mu_Y) \\
&= \sum_{i=1}^{n}(X_i-\mu_X)(Y_i-\mu_Y)-n(\bar{X}-\mu_X)(\bar{Y}-\mu_Y).
\end{aligned}
$$

4つめの等号で $\sum_{i=1}^{n}X_i=n\bar{X}$, $\sum_{i=1}^{n}Y_i=n\bar{Y}$ を用いた. ゆえに,

$$
\begin{aligned}
E\left[\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})\right] &= \sum_{i=1}^{n}E[(X_i-\mu_X)(Y_i-\mu_Y)]-nE[(\bar{X}-\mu_X)(\bar{Y}-\mu_Y)] \\
&= n\sigma_{XY}-n\frac{\sigma_{XY}}{n}=(n-1)\sigma_{XY}.
\end{aligned}
$$

ゆえに

$$
S_{XY}=\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})
$$

より

$$
E[S_{XY}]=\sigma_{XY}.
$$

解答終

3.8 データの相関係数の定義 (以上の定義のまとめにもなっている)

データの **相関係数** (correlation coefficient of sample, Pearsonの相関係数)を定義しよう.

数値データの場合の共分散の定義の設定のもとで, x_i, y_i 達の標本平均, 不偏分散達と不偏共分散を以下のように書くことにする:

$$
\begin{aligned}
&\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i, \quad \bar{y}=\frac{1}{n}\sum_{i=1}^{n}y_i, \\
&s_x^2=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2, \quad s_y^2=\frac{1}{n-1}\sum_{i=1}^{n}(y_i-\bar{y})^2, \\
&s_{xy}=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y}).
\end{aligned}
$$

s_x^2, s_y^2 の平方根をそれぞれ s_x, s_y と書く. x_i 達と y_i 達の **相関係数** r_{xy} を次のように定める:

$$
r_{xy}=\frac{s_{xy}}{s_x s_y}=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}.
$$

この r_{xy} は数値になる.

データのモデル化である確率変数達の共分散の定義の設定のもとで, X_i, Y_i 達の標本平均, 不偏分散達と不偏共分散を以下のように書くことにする:

$$
\begin{aligned}
&\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad \bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i, \\
&S_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2, \quad S_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \bar{Y})^2, \\
&S_{XY} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y}).
\end{aligned}
$$

S_X^2, S_Y^2 の平方根をそれぞれ S_X, S_Y と書く. X_i 達と Y_i 達の **相関係数** R_{XY} を次のように定める:

$$
R_{XY} = \frac{S_{XY}}{S_X S_Y} = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}.
$$

この R_{XY} は確率変数になる. (確率変数とは数学的には「その函数の期待値が定義されている変数」のことであり, 直観的には値がランダムに決まる変数だと思える.)

4 最小二乗法による線形回帰

最小二乗法による単純な線形回帰について説明しよう. 以下, $x_1, \ldots, x_n$ のうち2つは互いに異なると仮定する.

与えられたデータ $(x_1, y_1), \ldots, (x_n, y_n)$ における y_i 達の値達を x_i 達の値の共通の一次函数 $\alpha + \beta x_i$ (α, β は i によらない)で近似することを考える. そのとき,

$$
\varepsilon_i = y_i - (\alpha + \beta x_i)
$$

を **残差** と呼ぶ. このとき, **平均二乗残差**

$$
f(\alpha, \beta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\alpha + \beta x_i))^2
$$

を最小化することを **最小二乗法** と呼び, 最小二乗法で求めた係数 α, β の値を **回帰係数** と呼び, $\hat{\alpha}, \hat{\beta}$ と書き, それに対応する一次函数

$$
y = \hat{\alpha} + \hat{\beta} x
$$

を **回帰直線** (regression line)と呼ぶ. このとき, x を **説明変数** や **独立変数** などと呼び, y を **目的変数**, **従属変数**, **反応変数** などと呼ぶ.

これは一次函数によるデータの近似になっているので, **線形回帰** (ordinary least squares, OLS, linear regression)と呼ばれる. この場合には説明変数が x の1つだけなので **単回帰** (simple regression, simple linear regression)と呼ぶこともある. 説明変数が多変数になった場合には **重回帰** (multiple regression)と呼ぶ.

以下ではさらに, **最小平均二乗残差** (もしくは単に **平均二乗残差**)の値を次のように書くことにする:

$$
\hat{\sigma}^2 = f(\hat{\alpha}, \hat{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (\hat{\alpha} + \hat{\beta} x_i))^2
$$

注意: 最小二乗法は説明変数 x と目的変数 y について対称ではないので, それらの変数の立場を逆転して作った回帰直線は元の回帰直線に一致しない. 下の方で紹介するGalton (1886)の回帰直線の節も参照せよ.

4.1 問題: 最小二乗法による線形回帰の公式の導出

データ x_i の標本平均と不偏分散をそれぞれ $\bar{x}$, s_x^2 と書き, y_i の標本平均と不偏分散をそれぞれ $\bar{y}$, s_y^2 と書き, x_i と y_i の不偏共分散を s_{xy} と書くとき, 以上の $\hat{\alpha}, \hat{\beta}, \hat{\sigma}^2$ は次のように表されることを示せ:

$$
\hat{\beta} = \frac{s_{xy}}{s_x^2}, \quad \hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}, \quad \hat{\sigma}^2 = \frac{n-1}{n}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}.
$$

4.1.1 解答例1: 単回帰の公式の素朴な導出

解答例1: 複雑な計算をシンプルに記述するために以下のようにおく:

$$
\overline{x^2} = \frac{1}{n}\sum_{i=1}^{n} x_i^2, \quad \overline{y^2} = \frac{1}{n}\sum_{i=1}^{n} y_i^2, \quad \overline{xy} = \frac{1}{n}\sum_{i=1}^{n} x_i y_i.
$$

このとき,

$$s_x^2 = \frac{n}{n-1}\left(\overline{x^2} - \bar{x}^2\right), \quad s_y^2 = \frac{n}{n-1}\left(\overline{y^2} - \bar{y}^2\right), \quad s_{xy} = \frac{n}{n-1}\left(\overline{xy} - \bar{x}\bar{y}\right).$$

平均二乗残差 $f(\alpha, \beta)$ を最小化する $\alpha = \hat{\alpha}$, $\beta = \hat{\beta}$ は f の導函数を 0 する. ゆえに, $\hat{\alpha}, \hat{\beta}$ は次の方程式の解である:

$$\begin{aligned}
0 &= \frac{1}{2}\frac{\partial f}{\partial \alpha} = \frac{1}{n}\sum_{i=1}^{n}(\alpha + \beta x_i - y_i) = \alpha + \bar{x}\beta - \bar{y}, \\
0 &= \frac{1}{2}\frac{\partial f}{\partial \beta} = \frac{1}{n}\sum_{i=1}^{n} x_i(\alpha + \beta x_i - y_i) = \bar{x}\alpha + \overline{x^2}\beta - \overline{xy}.
\end{aligned}$$

後者から前者に $\bar{x}$ をかけたものを引くと

$$\left(\overline{x^2} - \bar{x}^2\right)\beta = \overline{xy} - \bar{x}\bar{y}.$$

これの解は

$$\hat{\beta} = \frac{\overline{xy} - \bar{x}\bar{y}}{\overline{x^2} - \bar{x}^2} = \frac{s_{xy}}{s_x^2}.$$

上の方程式の前者より, 次によって $\hat{\alpha}$ も求めることができる:

$$\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$

このとき

$$\begin{aligned}
\hat{\sigma}^2 &= f(\hat{\alpha}, \hat{\beta}) = f(\bar{y} - \hat{\beta}\bar{x}, \hat{\beta}) \\
&= \frac{1}{n}\sum_{i=1}^{n}(y_i - (\bar{y} - \hat{\beta}\bar{x} + \beta x_i))^2 = \frac{1}{n}\sum_{i=1}^{n}((y_i - \bar{y}) - \hat{\beta}(x_i - \bar{x}))^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left((y_i - \bar{y})^2 - 2\hat{\beta}(x_i - \bar{x})(y_i - \bar{y}) + \hat{\beta}^2(x_i - \bar{x})^2\right) \\
&= \frac{n-1}{n}\left(s_y^2 - 2\hat{\beta}s_{xy} + \hat{\beta}^2 s_x^2\right) = \frac{n-1}{n}\left(s_y^2 - 2\frac{s_{xy}}{s_x^2}s_{xy} + \frac{s_{xy}^2}{(s_x^2)^2}s_x^2\right) \\
&= \frac{n-1}{n}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}.
\end{aligned}$$

解答終

注意: 上の解答例ではわざと素朴な方法で計算してみた.

4.1.2 解答例2: データの5つの要約値だけで平均二乗残差を表示

解答例2: 平均二乗残差 $f(\alpha, \beta)$ の和の二乗の内側に $\bar{y} - (\alpha + \beta\bar{x})$ を引いて足して, 適当に整理して, 展開して, 整理すると以下のようになる:

$$\begin{aligned}
f(\alpha, \beta) &= \frac{1}{n}\sum_{i=1}^{n}(y_i - (\alpha + \beta x_i) - (\bar{y} - (\alpha + \beta\bar{x})) + (\bar{y} - (\alpha + \beta\bar{x})))^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}((y_i - \bar{y}) - \beta(x_i - \bar{x}) + (\bar{y} - (\alpha + \beta\bar{x})))^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\big((y_i - \bar{y})^2 + \beta^2(x_i - \bar{x})^2 + 2\beta(x_i - \bar{x})(y_i - \bar{y}) + (\bar{y} - (\alpha + \beta\bar{x}))^2 \\
&\qquad + 2\underbrace{(y_i - \bar{y})}_{\text{sum}=0}(\bar{y} - \alpha + \beta\bar{x})) - 2\beta\underbrace{(x_i - \bar{x})}_{\text{sum}=0}(\bar{y} - (\alpha + \beta\bar{x}))\big) \\
&= \frac{n-1}{n}\left(s_y^2 + \beta^2 s_x^2 - 2\beta s_{xy}\right) + (\bar{y} - (\alpha + \beta\bar{x}))^2 \\
&= \frac{n-1}{n}\left(s_x^2\left(\beta - \frac{s_{xy}}{s_x^2}\right)^2 + \frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}\right) + (\bar{y} - (\alpha + \beta\bar{x}))^2.
\end{aligned}$$

これより, 平均二乗残差 $f(\alpha, \beta)$ は, β, α がそれぞれ

$$\hat{\beta} = \frac{s_{xy}}{s_x^2}, \quad \hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$$

に等しいときに, 最小値

$$\hat{\sigma}^2 = \frac{n-1}{n}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}$$

になることがわかる.

解答終

注意: 上の計算では最初に, 回帰係数 α, β の函数 $f(\alpha, \beta)$ (平均二乗残差)が $2n$ 個の数値で構成されているデータ $(x_1, y_1), \ldots, (x_n, y_n)$ を要約している標本平均と分散共分散の5つの数値 $\bar{x}, \bar{y}, s_x^2, s_y^2, s_{xy}^2$ だけで記述されることを示した. 特に **回帰直線と最小平均二乗残差の値はそれらの5つの要約値だけで決まってしまう.** データのそれら5つの値への要約が不適切な場合には最小二乗法を使うべきではない.

注意: 以上とはまったく別の **線形代数** における **直交射影** で最小二乗法を理解することもできる. 最小二乗法は数学的には直交射影の理論そのものだと言ってよい. この点に関する詳しい解説は別の機会に行いたい. 以上では最も素朴な最小二乗法の扱い方をした.

注意: 最小二乗法は **正規分布モデルの最尤法の一種** だとみなされる. これも非常に重要な視点である. この点についての解説も別の機会に行いたい. 最小二乗法を与える正規分布モデルの確率密度函数は

$$\begin{aligned} p(y_1, \ldots, y_n | x_1, \ldots, x_n, \alpha, \beta, \sigma) &= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - (\alpha + \beta x_i))^2 \right) \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{n}{2\sigma^2} \left[\frac{n-1}{n} \left(s_x^2 \left(\beta - \frac{s_{xy}}{s_x^2} \right)^2 + \frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2} \right) + (\bar{y} - (\alpha + \beta\bar{x}))^2 \right] \right) \end{aligned}$$

であり, $\bar{x}, \bar{y}, s_x^2, s_y^2, s_{xy}^2$ はこの統計モデルの **十分統計量** になっている. この統計モデルは以下のようにも記述される:

$$y_i - (\alpha + \beta x_i) \sim \mathrm{Norma}(0, \sigma) \quad (i = 1, \ldots, n)$$

4.1.3 コンピュータによる最小二乗法の計算例

WolframAlpha (https://www.wolframalpha.com/)では次のように入力すれば単回帰を実行できる.

[実行 (https://www.wolframalpha.com/input?i=linear+fit+%2811%2C22%29%2C+%2812%2C20%29%2C+%2813%2C25%29%2C+%2814%2C+19%29%2C+%2815%2C+24%2 → 7.35238 + 1.17143 x

```
linear fit (11,22), (12,20), (13,25), (14, 19), (15, 24), (16, 29)
```

不偏分散と不偏共分散も以下のように入力すれば計算できる.

[実行 (https://www.wolframalpha.com/input?i=variance+%2811%2C+12%2C+13%2C+14%2C+15%2C+16%29)] → 3.5

```
variance (11, 12, 13, 14, 15, 16)
```

[実行 (https://www.wolframalpha.com/input?i=covariance+%2811%2C+12%2C+13%2C+14%2C+15%2C+16%29%2C+%2822%2C+20%2C+25%2C+19%2C+24%2C+29%29)] → 4.1

```
covariance (11, 12, 13, 14, 15, 16), (22, 20, 25, 19, 24, 29)
```

他にも便利な道具があるので, 自分で色々試してみて研究しておくとよい.

以下のコードはJulia言語 (https://julialang.org/).

In [36]:
```
[ones(6) [11, 12, 13, 14, 15, 16]] \ [22, 20, 25, 19, 24, 29]
```

Out[36]:
```
2-element Vector{Float64}:
 7.352380952380989
 1.171428571428569
```

In [37]:
```
var([11, 12, 13, 14, 15, 16])
```
Out[37]: 3.5

In [38]:
```
cov([11, 12, 13, 14, 15, 16], [22, 20, 25, 19, 24, 29])
```
Out[38]: 4.1

不偏分散と不偏共分散については以下のように素朴に計算してもよい. (素朴な計算も一度はやってみるべきである.)

In [39]:
```
n = 6
```
Out[39]: 6

In [40]:
```
# xの標本平均
a = (11 + 12 + 13 + 14 + 15 + 16)/n
```
Out[40]: 13.5

In [41]:
```
# xの不偏分散
((11 - a)^2 + (12 - a)^2 + (13 - a)^2 + (14 - a)^2 + (15 - a)^2 + (16 - a)^2)/(n - 1)
```
Out[41]: 3.5

次の公式を使った方が入力が楽になる:

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}\left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^2\right).$$

In [42]:
```
(11^2 + 12^2 + 13^2 + 14^2 + 15^2 + 16^2 - n*a^2)/(n - 1) # こちらの方が入力が楽
```
Out[42]: 3.5

In [43]:
```
# yの標本平均
b = (22 + 20 + 25 + 19 + 24 + 29)/n
```
Out[43]: 23.166666666666668

標本分散についても次の公式を使って計算してみよう:

$$s_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_u - \bar{y}) = \frac{1}{n-1}\left(\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}\right).$$

In [44]:
```
# xとyの不偏共分散
(11*22 + 12*20 + 13*25 + 14*19 + 15*24 + 16*29 - n*a*b)/(n - 1)
```
Out[44]: 4.1

グラフもプロットしてみよう. さらに色々な計算方法も試してみよう.

In [45]:
```
# プロット用函数
function plot_ols(x, y, α̂, β̂; xlim=nothing,
        title="data and regression line", kwargs...)
    a, b = extrema(x)
    isnothing(xlim) && (xlim = (a - 0.1(b-a), b + 0.1(b-a)))
    scatter(x, y; label="", msw=0)
    plot!(x -> α̂ + β̂*x, xlim...; label="", lw=2)
    plot!(; xlabel="x", ylabel="y", msw=0)
    plot!(; size=(400, 300))
    plot!(; title, kwargs...)
end
```
Out[45]: plot_ols (generic function with 1 method)

In [46]:
```
# 「よろしくみなさん」に擬似乱数のシードを設定(テスト用)
Random.seed!(4649373)
```

Out[46]:
```
TaskLocalRNG()
```

In [47]:
```
# create test data
n = 100
α₀, β₀, σ₀ = 10, 2, 4
x = rand(Normal(10, 2), n)
noise = rand(Normal(0, σ₀), n)
y = @. α₀ + β₀*x + noise
@show α₀ β₀ σ₀
println()

# ordinary least squares
@show x̄ = sum(x[i] for i in 1:n)/n
@show ȳ = sum(y[i] for i in 1:n)/n
@show sx² = sum((x[i] - x̄)^2 for i in 1:n)/(n - 1)
@show sy² = sum((y[i] - ȳ)^2 for i in 1:n)/(n - 1)
@show sxy = sum((x[i] - x̄)*(y[i] - ȳ) for i in 1:n)/(n - 1)
println()
β̂ = sxy/sx²
α̂ = ȳ - β̂*x̄
σ̂ = √((n - 1)/n*(sx²*sy² - sxy^2)/sx²)
@show α̂ β̂ σ̂

plot_ols(x, y, α̂, β̂)
```

```
α₀ = 10
β₀ = 2
σ₀ = 4

x̄ = sum((x[i] for i = 1:n)) / n = 9.930093374143759
ȳ = sum((y[i] for i = 1:n)) / n = 29.76287823674706
sx² = sum(((x[i] - x̄) ^ 2 for i = 1:n)) / (n - 1) = 3.9902128888981885
sy² = sum(((y[i] - ȳ) ^ 2 for i = 1:n)) / (n - 1) = 37.32614282138912
sxy = sum(((x[i] - x̄) * (y[i] - ȳ) for i = 1:n)) / (n - 1) = 8.920847575776838

α̂ = 7.562346117161635
β̂ = 2.235682111246986
σ̂ = 4.148269982085684
```

Out[47]:

不偏分散と不偏共分散は次のようなコードでも計算できる.

In [48]:
```
@show sx² = var(x)
@show sy² = var(y)
@show sxy = cov(x, y);
```

```
sx² = var(x) = 3.990212888898188
sy² = var(y) = 37.3261428213891
sxy = cov(x, y) = 8.920847575776842
```

以下のように線形代数的な実装にすればシンプルなコードになる.

In [49]:

```
# ordinary least squares
X = [ones(n) x] # design matrix
α̂, β̂ = X \ y
σ̂ = norm(y - (α̂ .+ β̂*x))/√n
@show α̂ β̂ σ̂

plot_ols(x, y, α̂, β̂)
```

```
α̂ = 7.562346117161626
β̂ = 2.235682111246988
σ̂ = 4.148269982085681
```

Out[49]:

正規分布モデルの最尤法でも同じことをできる.

In [50]:

```
# negative log likelihood
function f(x, y, α, β, t)
    -sum(logpdf(Normal(0, exp(t)), y[i] - (α + β*x[i])) for i in eachindex(x))
end

# minimize negative log likelihood = maximize likelihood
o = optimize(((α, β, t),) -> f(x, y, α, β, t), zeros(3), LBFGS())
α̂, β̂, t̂ = o.minimizer
σ̂ = exp(t̂)
@show α̂ β̂ σ̂

plot_ols(x, y, α̂, β̂)
```

```
α̂ = 7.562346115104934
β̂ = 2.235682111439474 3
σ̂ = 4.148269982255
```

Out[50]:

4.2 必修問題: 最小二乗法の計算例 (Anscombe's quartet)

以下に示した $(x_1, y_1), \ldots, (x_n, y_n)$ 型の4つのデータのそれぞれについて,

- x_i 達の標本平均 $\bar{x}$
- y_i 達の標本平均 $\bar{y}$
- x_i 達の不偏分散 s_x^2,
- y_i 達の不偏分散 s_y^2,
- x_i 達と y_i 達の不偏共分散 s_{xy},
- 回帰直線 $y = \hat{\alpha} + \hat{\beta}x$ の係数 $\hat{\alpha}$, $\hat{\beta}$ と最小平均二乗残差の平方根 $\hat{\sigma}$

を求めよ. それぞれについて小数点以下第1桁まで求めよ.

1. (10, 8.04), (8, 6.95), (13, 7.58), (9, 8.81), (11, 8.33), (14, 9.96), (6, 7.24), (4, 4.26), (12, 10.84), (7, 4.82), (5, 5.68)
2. (10, 9.14), (8, 8.14), (13, 8.74), (9, 8.77), (11, 9.26), (14, 8.1), (6, 6.13), (4, 3.1), (12, 9.13), (7, 7.26), (5, 4.74)
3. (10, 7.46), (8, 6.77), (13, 12.74), (9, 7.11), (11, 7.81), (14, 8.84), (6, 6.08), (4, 5.39), (12, 8.15), (7, 6.42), (5, 5.73)
4. (8, 6.58), (8, 5.76), (8, 7.71), (8, 8.84), (8, 8.47), (8, 7.04), (8, 5.25), (19, 12.5), (8, 5.56), (8, 7.91), (8, 6.89)

注意: 実はこれは有名なアンスコムの例 (https://ja.wikipedia.org/wiki/%E3%82%A2%E3%83%B3%E3%82%B9%E3%82%B3%E3%83%A0%E3%81%AE%E4%BE%8B)である. この例の教訓はデータの散布図をプロットすることの重要性である. 回帰直線を求める統計分析は2,3,4のデータについては適切ではない.

コピー&ペースト用: Anscombe's quartet

データ1

- x: 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
- y: 8.04, 6.95, 7.58, 8.81, 8.33, 9.96, 7.24, 4.26, 10.84, 4.82, 5.68

データ2

- x: 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
- y: 9.14, 8.14, 8.74, 8.77, 9.26, 8.1, 6.13, 3.1, 9.13, 7.26, 4.74

データ3

- x: 10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5
- y: 7.46, 6.77, 12.74, 7.11, 7.81, 8.84, 6.08, 5.39, 8.15, 6.42, 5.73

データ4

- x: 8, 8, 8, 8, 8, 8, 8, 19, 8, 8, 8
- y: 6.58, 5.76, 7.71, 8.84, 8.47, 7.04, 5.25, 12.5, 5.56, 7.91, 6.89

解答例:

データ1:

$$\begin{aligned}
&\bar{x} \approx 9.0, \quad \bar{y} \approx 7.5, \\
&s_x^2 \approx 11.0, \quad s_y^2 \approx 4.1, \quad s_{xy} \approx 5.5 \\
&\hat{\alpha} \approx 3.0, \quad \hat{\beta} \approx 0.5, \quad \hat{\sigma} \approx 1.1
\end{aligned}$$

データ2:

$$\begin{aligned}
&\bar{x} \approx 9.0, \quad \bar{y} \approx 7.5, \\
&s_x^2 \approx 11.0, \quad s_y^2 \approx 4.1, \quad s_{xy} \approx 5.5 \\
&\hat{\alpha} \approx 3.0, \quad \hat{\beta} \approx 0.5, \quad \hat{\sigma} \approx 1.1
\end{aligned}$$

データ3:

$$\begin{aligned}
&\bar{x} \approx 9.0, \quad \bar{y} \approx 7.5, \\
&s_x^2 \approx 11.0, \quad s_y^2 \approx 4.1, \quad s_{xy} \approx 5.5 \\
&\hat{\alpha} \approx 3.0, \quad \hat{\beta} \approx 0.5, \quad \hat{\sigma} \approx 1.1
\end{aligned}$$

データ4:

$$\begin{aligned}
&\bar{x} \approx 9.0, \quad \bar{y} \approx 7.5, \\
&s_x^2 \approx 11.0, \quad s_y^2 \approx 4.1, \quad s_{xy} \approx 5.5 \\
&\hat{\alpha} \approx 3.0, \quad \hat{\beta} \approx 0.5, \quad \hat{\sigma} \approx 1.1
\end{aligned}$$

解答終

注意: 小数点以下第1桁までの数値は4つのデータについてすべて一致している! データの散布図を描くと, 回帰直線を描くことが適切なのは最初のデータだけで, 残りの3つのデータについては不適切なことがわかる. グラフは下の方にある.

4.2.1 WolframAlphaでAnscombeの例1を扱う方法

データ1に関するWolframAlphaの使用例: xの標本平均 (https://www.wolframalpha.com/input?i=mean+10%2C+8%2C+13%2C+9%2C+11%2C+14%2C+6%2C+4%2C+12%2C+7%2C+5&lang=ja), yの標本平均 (https://www.wolframalpha.com/input?i=mean+8.04%2C+6.95%2C+7.58%2C+8.81%2C+8.33%2C+9.96%2C+7.24%2C+4.26%2C+10.84%2C+4.82%2C+5.68), xの不偏分散 (https://www.wolframalpha.com/input?i=var+10%2C+8%2C+13%2C+9%2C+11%2C+14%2C+6%2C+4%2C+12%2C+7%2C+5), yの不偏分散 (https://www.wolframalpha.com/input?i=var+8.04%2C+6.95%2C+7.58%2C+8.81%2C+8.33%2C+9.96%2C+7.24%2C+4.26%2C+10.84%2C+4.82%2C+5.68), x,yの不偏共分散 (https://www.wolframalpha.com/input?i=covar+%2810%2C+8%2C+13%2C+9%2C+11%2C+14%2C+6%2C+4%2C+12%2C+7%2C+5%29%2C+%288.04%2C+6.95%2C+7
回帰直線 (https://www.wolframalpha.com/input?i=linear+fit+%2810%2C+8.04%29%2C+%288%2C+6.95%29%2C+%2813%2C+7.58%29%2C+%289%2C+8.81%29%2C+%2811%2
最小平均二乗残差の平方根 (https://www.wolframalpha.com/input?i=sqrt%28%28n-1%29%2Fn*%28a*c+-+b%5E2%29%2Fa%29+where+n%3D11%2C+a%3D11%2C+b%3D5.5%2C+c%3D4.127)

4.2.2 Julia言語でAnscombeの例を扱う方法

RDatatsets.jl パッケージを入れればその中にAnscombeの例が入っている.

In [51]:
```
using RDatasets
anscombe = dataset("datasets", "anscombe")
```

Out[51]: 11 rows × 8 columns

	X1	X2	X3	X4	Y1	Y2	Y3	Y4
	Int64	Int64	Int64	Int64	Float64	Float64	Float64	Float64
1	10	10	10	8	8.04	9.14	7.46	6.58
2	8	8	8	8	6.95	8.14	6.77	5.76
3	13	13	13	8	7.58	8.74	12.74	7.71
4	9	9	9	8	8.81	8.77	7.11	8.84
5	11	11	11	8	8.33	9.26	7.81	8.47
6	14	14	14	8	9.96	8.1	8.84	7.04
7	6	6	6	8	7.24	6.13	6.08	5.25
8	4	4	4	19	4.26	3.1	5.39	12.5
9	12	12	12	8	10.84	9.13	8.15	5.56
10	7	7	7	8	4.82	7.26	6.42	7.91
11	5	5	5	8	5.68	4.74	5.73	6.89

In [52]:
```
# コピー＆ペーストに利用するための表示
data1 = collect(zip(anscombe.X1, anscombe.Y1))
data2 = collect(zip(anscombe.X2, anscombe.Y2))
data3 = collect(zip(anscombe.X3, anscombe.Y3))
data4 = collect(zip(anscombe.X4, anscombe.Y4))
@show data1 data2 data3 data4;
```

```
data1 = [(10, 8.04), (8, 6.95), (13, 7.58), (9, 8.81), (11, 8.33), (14, 9.96), (6, 7.24), (4, 4.
26), (12, 10.84), (7, 4.82), (5, 5.68)]
data2 = [(10, 9.14), (8, 8.14), (13, 8.74), (9, 8.77), (11, 9.26), (14, 8.1), (6, 6.13), (4, 3.
1), (12, 9.13), (7, 7.26), (5, 4.74)]
data3 = [(10, 7.46), (8, 6.77), (13, 12.74), (9, 7.11), (11, 7.81), (14, 8.84), (6, 6.08), (4,
5.39), (12, 8.15), (7, 6.42), (5, 5.73)]
data4 = [(8, 6.58), (8, 5.76), (8, 7.71), (8, 8.84), (8, 8.47), (8, 7.04), (8, 5.25), (19, 12.
5), (8, 5.56), (8, 7.91), (8, 6.89)]
```

In [53]:
```
# コピー＆ペーストに利用するための表示
@show anscombe.X1 anscombe.Y1
@show anscombe.X2 anscombe.Y2
@show anscombe.X3 anscombe.Y3
@show anscombe.X4 anscombe.Y4;
```

```
anscombe.X1 = [10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5]
anscombe.Y1 = [8.04, 6.95, 7.58, 8.81, 8.33, 9.96, 7.24, 4.26, 10.84, 4.82, 5.68]
anscombe.X2 = [10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5]
anscombe.Y2 = [9.14, 8.14, 8.74, 8.77, 9.26, 8.1, 6.13, 3.1, 9.13, 7.26, 4.74]
anscombe.X3 = [10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5]
anscombe.Y3 = [7.46, 6.77, 12.74, 7.11, 7.81, 8.84, 6.08, 5.39, 8.15, 6.42, 5.73]
anscombe.X4 = [8, 8, 8, 8, 8, 8, 8, 19, 8, 8, 8]
anscombe.Y4 = [6.58, 5.76, 7.71, 8.84, 8.47, 7.04, 5.25, 12.5, 5.56, 7.91, 6.89]
```

In [54]:
```
function solve_anscombe(x, y;
        title="", xlim=(3, 20), ylim=(2, 14), size=(250, 225))
    @show x̄ = mean(x)
    @show ȳ = mean(y)
    @show sx² = var(x)
    @show sy² = var(y)
    @show sxy = cov(x, y)
    β̂ = sxy/sx²
    α̂ = ȳ - β̂*x̄
    σ̂ = √((n - 1)/n*(sx²*sy² - sxy^2)/sx²)
    @show α̂ β̂ σ̂

    plot_ols(x, y, α̂, β̂; title, xlim, ylim, size)
end
```

Out[54]:
```
solve_anscombe (generic function with 1 method)
```

In [55]:
```
PA1 = solve_anscombe(anscombe.X1, anscombe.Y1, title="Anscombe 1")
```

```
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500909090909093
sx² = var(x) = 11.0
sy² = var(y) = 4.127269090909091
sxy = cov(x, y) = 5.501
α̂ = 3.0000909090909103
β̂ = 0.5000909090909091
σ̂ = 1.1672644558967773
```

Out[55]:

In [56]:
```
PA2 = solve_anscombe(anscombe.X2, anscombe.Y2, title="Anscombe 2")
```

```
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500909090909091
sx² = var(x) = 11.0
sy² = var(y) = 4.127629090909091
sxy = cov(x, y) = 5.5
α̂ = 3.000909090909091
β̂ = 0.5
σ̂ = 1.167841085079644
```

Out[56]:

In [57]:
```
PA3 = solve_anscombe(anscombe.X3, anscombe.Y3, title="Anscombe 3")
```

```
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500000000000001
sx² = var(x) = 11.0
sy² = var(y) = 4.12262
sxy = cov(x, y) = 5.497
α̂ = 3.0024545454545466
β̂ = 0.49972727272727274
σ̂ = 1.1669888559879227
```

Out[57]:

In [58]:
```
PA4 = solve_anscombe(anscombe.X4, anscombe.Y4, title="Anscombe 4")
```

```
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.50090909090909
sx² = var(x) = 11.0
sy² = var(y) = 4.12324909090909
sxy = cov(x, y) = 5.499
α̂ = 3.0017272727272726
β̂ = 0.49990909090909086
σ̂ = 1.1664075231238866
```

Out[58]:

In [59]:
```
plot(PA1, PA2, PA3, PA4; size=(500, 450), layout=(2,2))
```

Out[59]:

アンスコムの例では1以外のデータについて単純に最小二乗法で回帰直線を引くことは不適切である.

さらに, データを要約せずにその全体の様子を見るためにグラフを描いてみるとことが非常に大事なこともこの例からわかる.

4.3 Galton (1886)が描いた2つの回帰直線

下の方にある図は次の論文の5ページ目からの転載である:

- Galton, Francis. Regression towards mediocrity in hereditary stature. Journal of the Anthropological Institute, Vol. 5 (1886), 246-263. [link (https://galton.org/bib/JournalItem.aspx_action=view_id=157)]

この論文の著者の Francis Galton (https://en.wikipedia.org/wiki/Francis_Galton) (フランシス・ゴルトン (https://ja.wikipedia.org/wiki/%E3%83%95%E3%83%A9%E3%83%B3%E3%82%B7%E3%82%B9%E3%83%BB%E3%82%B4%E3%
は「平均への回帰」の記述や「相関係数」の導入で有名である.

このグラフにおける縦軸は親の身長の「平均値」であり, 横軸は成人した子供の身長である(単位はインチ). 親の身長の「平均値」の正確な意味は論文の2ページ目で以下のように説明されている:

> My data consisted of th e heights of 930 adult children and of their respective parentages, 205 in number. In every case I transmuted the female statures to their corresponding male equivalents and used them in their transmuted form, so that no objection grounded on the sexual difference of stature need be raised when I speak of averages. The factor I used was 1.08, which is equivalent to adding a little less than one-twelfth to each female height.

翻訳: 私のデータは930人の成人した子供とその親205人の身長である. どの場合でも, 女性の身長を対応する男性の身長に変換し, 変換された状態で使用した. それによって, 平均値について語るときに, 身長の性差に基づく異論を唱える必要を無くした. 使用した係数は1.08であり, これは各々の女性の身長にその12分の1弱を足すことに相当する.

上のグラフに描かれた楕円は, データの分布が多変量正規分布に近いことを表していると考えてよい. さらに, 親と子の身長のあいだに相関関係があることがわかる.

直線 ON が説明変数を親の身長, 目的変数を子の身長としたときの回帰直線であり, 直線 OM は2つの変数の立場を逆転させて作った回帰直線である. **2つの回帰直線は互いに一致しないし, 楕円の長軸とも一致しない.**

点 O はデータの平均値(数式で書くと $(\bar{x}, \bar{y})$ もしくは $(\bar{y}, \bar{x})$)であり, 回帰直線 ON が通る点 N は楕円と横軸に平行な直線が接する点になっており, 回帰直線 OM が通る点 M は楕円と縦軸に平行な直線が接する点になっている. (そうなる理由を自分で考えることは数学に関するちょっとした練習になるので, 興味がある人は自分で考えてみるとよい. 少し下の方に解答例がある.)

4.3.1 Galtonのグラフの再現

以下のセルのデータは上の論文の3ページ目にある Table I である. (数値の入力間違いが残っている可能性があるので, 原論文を参照して各自確認すること. 私自身は入力間違いはないと思っているが, それは私がそう思っているだけに過ぎない. このようなことはよくある.)

In [60]:

```
galton = [
    0  0  0  0  0  0  0  0  0  0  0  1  3  0
    0  0  0  0  0  0  0  1  2  1  2  7  2  4
    0  0  0  0  1  3  4  3  5 10  4  9  2  2
    1  0  1  0  1  1  3 12 18 14  7  4  3  3
    0  0  1 16  4 17 27 20 33 25 20 11  4  5
    1  0  7 11 16 25 31 34 48 21 18  4  3  0
    0  3  5 14 15 36 38 28 38 19 11  4  0  0
    0  3  3  5  2 17 17 14 13  4  0  0  0  0
    1  0  9  5  7 11 11  7  7  5  2  1  0  0
    1  1  4  4  1  5  5  0  2  0  0  0  0  0
    1  0  2  4  1  2  2  1  1  0  0  0  0  0
]
# 上の行列の縦軸は親の身長であり，行列の下ほど身長は小さくなる．
# 各種函数の仕様に合わせるために，行列の縦方向を逆転させて扱う．
galton = reverse(galton; dims=1)
# 行列の横軸が子の身長(単位はインチ)
galton_x = 61.2:74.2
# 行列の縦軸が親の身長(単位はインチ)
galton_y = 63.5:73.5

# データのヒートマップによる可視化
heatmap(galton_x, galton_y, galton; size=(320, 220))
```

Out[60]:

以下のセルでは楕円の長軸と短軸と2本の回帰直線をプロットしている.

In [61]:

```
# 適当にランダマイズしてデータを作成
Random.seed!(4649373)
data = hcat(([x-0.5+rand(), y-0.5+rand()]
        for (i, y) in enumerate(galton_y)
            for (j, x) in enumerate(galton_x)
                for _ in 1:galton[i,j])...)

# 多変量正規分布の最尤法
mvnormal = fit_mle(MvNormal, data)

# 楕円の長軸と短軸を求める
μ = mean(mvnormal) # 楕円の中心 [x̄, ȳ]
Σ = cov(mvnormal)  # 分散共分散行列 ((n-2)/n)[sx² sxy; sxy sy²]
d, U = eigen(Σ)    # U[:,j] = Σの固有値 d[j] の固有ベクトル
a = [-U[1,1]/U[2,1], -U[1,2]/U[2,2]] # 長軸・短軸の傾き
@show Σ μ a
g1(x) = a[1]*(x - μ[1]) + μ[2] # 長軸
g2(x) = a[2]*(x - μ[1]) + μ[2] # 短軸

# 最小二乗法による線形回帰
x, y = data[1,:], data[2,:]
X = [x.^0 x]
Y = [y.^0 y]
@show β = X\y # OLS(x, y)の回帰係数
@show γ = Y\x # OLS(y, x)の回帰係数
f1(x) = β[1] + β[2]*x # OLS(x, y)の回帰直線
f2(y) = γ[1] + γ[2]*y # OLS(y, x)の回帰直線

# グラフをプロット
widenlim((a, b)) = (c = 0.05(b-a); (a-c, b+c))
xlim = widenlim(extrema(galton_x))
ylim = widenlim(extrema(galton_y))
xs = range(xlim..., 1000)
ys = range(ylim..., 1000)

plot(legend=:topleft)
scatter!(x, y; label="data", msw=0, alpha=0.7)
contour!(xs, ys, (x,y) → pdf(mvnormal, [x,y]); colorbar=false, ls=:dot)
plot!(xs, g1.(xs); label="major axis", color=:blue)
plot!(xs, g2.(xs); label="minor axis", color=:blue, ls=:dash)
plot!(f2.(ys), ys; label="OLS(y, x)", color=:red)
plot!(xs, f1.(xs); label="OLS(x, y)", color=:red, ls=:dash)
plot!(; xlim, ylim, aspectratio=1, xtick=0:100, ytick=0:100)
plot!(; size=(560, 440))
title!("Ellipse axes and regression lines of Galton (1886)")
```

```
Σ = [6.496979569005935 2.0836915591542615; 2.0836915591542615 3.350202984520065]
μ = [68.09572980291252, 68.30018786422937]
a = [0.4979680162348105, -2.008161101512316]
β = X \ y = [46.460732175978194, 0.320716963478627]
γ = Y \ x = [25.61575900743164, 0.6219597943116172]
```

Out[61]:

注意・警告: 2つの回帰直線は楕円の長軸に一致しない. この点を誤解している人もいるようなので注意すること. 目分量で大雑把に回帰直線を手で引くときには, 楕円の長軸に線を引きたくなるかもしれないが, 誤りになる. 回帰直線は楕円と横軸または縦軸に平行な直線が接する点を通る直線になる.

4.3.2 グラフ中の楕円と2変量正規分布モデルや回帰直線の関係

上の計算での多変量正規分布の最尤法では, 次の確率密度函数で定まる2変量正規分布の標本分布を統計モデルとして採用している:

$$p(x,y|\mu,\Sigma) = \frac{1}{\sqrt{\det(2\pi\Sigma)}}\exp\left(-\frac{1}{2}\left[\,x-\mu_x,\ y-\mu_y\,\right]\Sigma^{-1}\begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix}\right).$$

ここで $\mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \in \mathbb{R}^2$ で Σ は固有値が正の 2×2 の実対称行列である. 独立なパラメータは5つある. この統計モデルのデータ $(x_1,y_1),\ldots,(x_n,y_n)$ に関する最尤法はその5つのパラメータを次のように決定する:

$$\hat{\mu} = \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}, \quad \hat{\Sigma} = \frac{n-1}{n}\begin{bmatrix} s_x^2 & s_{xy} \\ s_{xy} & s_y^2 \end{bmatrix}.$$

単回帰の結果を決定したデータの5つの要約値 $\bar{x},\bar{y},s_x^2,s_{xy},s_y^2$ が2変量正規分布モデルの最尤法の解も決定する.

$\hat{\Sigma}$ の式の右辺の $(n-1)/n$ は分散と共分散の推定値として不偏分散と不偏共分散を採用したので必要になった因子である. 分散と共分散の推定値として $n-1$ ではなく n で割った方を採用すればその因子はなくなる. このように正規分布モデルの最尤法の解の公式と不偏分散・不偏共分散は相性は悪い. 慣習的に使用されている不偏分散と不偏共分散にはこのようなトレードオフもある. (どのような場合であっても都合のよい流儀は存在しない.)

以上の統計モデルと楕円の関係について説明しよう.

分散共分散行列 Σ を $\Sigma = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$ と書くと,

$$\Sigma^{-1} = \frac{1}{ac-b^2}\begin{bmatrix} c & -b \\ -b & a \end{bmatrix}$$

なので

$$\left[\,x-\mu_x,\ y-\mu_y\,\right]\Sigma^{-1}\begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix} = \frac{c(x-\mu_x)^2 - 2b(x-\mu_x)(y-\mu_y) +}{ac-b^2}.$$

前節で描いたグラフ中の楕円はこれが一定の値になるという条件で定義された図形である.

以下, 計算を簡単に記述するために $\mu_x = \mu_y = 0$ と仮定する. このとき, その楕円の方程式は

$$ay^2 - 2bxy + cx^2 = \text{constant}.$$

になる. この条件を保ったまま (x, y) を微小に動かすと,

$$2ay\,dy - 2bx\,dy - 2by\,dx + 2cx\,dx = 0$$

すなわち

$$(ay - bx)\,dy - (by - cx)\,dx = 0$$

が得られる(これは点 (x, y) を通る楕円の接線の方程式だとみなされる).

点 (x, y) が楕円と y 軸と平行な直線の接点のとき, そのような接点 (x, y) を楕円上に乗せたままで微小に動かすと x 座標は高次の微小量しか変化しないので $dx = 0$ が得られ, $y = (b/a)x$ が得られる. これは, $\Sigma = \hat{\Sigma}$ の場合には $a = ((n-1)/n)s_x^2$, $b = ((n-1)/n)s_{xy}$ なので $y = (s_{xy}/s_x^2)x$ を意味する. その式で定義される直線の傾き s_{xy}/s_x^2 は説明変数が x で目的変数が y のときの単回帰直線の傾き $\widetilde{\beta}$ に一致している.

これで, 説明変数が x で目的変数が y のときの単回帰直線が楕円と y 軸に平行な直線の接点を通ることも証明された.

注意: このことから, 楕円の内側を通る y 軸に平行な直線 ℓ について, 説明変数が x で目的変数が y のときの単回帰直線と直線 ℓ の交点と楕円と直線 ℓ の2つの交点の距離が等しくなることもわかる. なぜならば, 変換 $(x, y) \mapsto (x, y - (b/a)x)$ によって, 回帰直線 $y = (b/a)x$ が楕円と y 軸に平行な直線の接点を通るという条件も保たれるので($\mu_x = \mu_y = 0$ と仮定している), その変換でうつした先の楕円の軸の片方は x 軸に一致し, うつした先の楕円が x 軸について線対称になることがわかるからである. 下の方で描いたグラフを見よ.

注意: 以上の数学的議論を追い切れない人は前節で描いたグラフやGolton氏自身が描いたグラフをよく見て, 単回帰直線が楕円と横軸(もしくは縦軸)と平行な直線の接点を通っていることを目で確認せよ. 数学的な議論が追い切れない場合であっても, 結論をグラフを見て確認できるならば, 記憶に残って役に立つ知識になることが多い.

4.3.3 楕円と回帰直線の関係の視覚化

In [62]:

```
f(x, y, h) = 6y^2 - 4x*y + 3x^2 - h
g(x, h, s) = (2x + s*√(4x^2 - 6*(3x^2 - h)))/6
h = f(1.5, 0.5, 0)
x = range(-2, 2, 200)
y = range(-1.5, 1.5, 200)
plot(; aspectratio=1, size=(400, 300))
plot!(x, x -> x/3; label="regression line", legend=:topleft)
vline!([-1.5, 1.5]; label="", ls=:dash)
k = -0.4
plot!([k, k], [g(k, h, -1), g(k, h, 1)]; label="", c=3)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2, 2); label="", c=3)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2, 2); label="", c=3)
k = 0.8
plot!([k, k], [g(k, h, -1), g(k, h, 1)]; label="", c=4)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2 + 0.02, 2); label="", c=4)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2 - 0.02, 2); label="", c=4)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2 + 0.02, 2); label="", c=4)
plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2 - 0.02, 2); label="", c=4)
contour!(x, y, (x, y) -> f(x, y, h); colorbar=false, levels=[0])
plot!(; xlabel="x", ylabel="y")
```

Out[62]:

確かに, 楕円と縦軸に平行な直線の接点を通る直線(=回帰直線)が楕円を上下の長さについて二等分していることがこの図を見ればわかる.

In [63]:

```
f(x, y, h) = 6y^2 - 4x*y + 3x^2 - h
g(x, h, s) = (2x + s*√(4x^2 - 6*(3x^2 - h)))/6
h = f(1.5, 0.5, 0)
x = range(-2, 2, 200)
y = range(-1.5, 1.5, 200)
anim = @animate for a in [reverse(range(-1, 1/3, 40)); range(-1, 1/3, 40)]
    plot(; aspectratio=1, size=(400, 320))
    plot!(x, x -> x/3+a*x; label="", legend=:topleft)
    vline!([-1.5, 1.5]; label="", ls=:dash)
    k = -0.4
    plot!([k, k], [g(k, h, -1)+a*k, g(k, h, 1)+a*k]; label="", c=3)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2+a*k, 2); label="", c=3)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2+a*k, 2); label="", c=3)
    k = 0.8
    plot!([k, k], [g(k, h, -1)+a*k, g(k, h, 1)+a*k]; label="", c=4)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2+a*k + 0.02, 2); label="", c=4)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h, -1))/2+a*k - 0.02, 2); label="", c=4)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2+a*k + 0.02, 2); label="", c=4)
    plot!([k-0.05, k+0.05], fill((k/3 + g(k, h,  1))/2+a*k - 0.02, 2); label="", c=4)
    contour!(x, y, (x, y) -> f(x, y-a*x, h); colorbar=false, levels=[0])
    plot!(; xlabel="x", ylabel="y")
    plot!(; xtick=-2:0.5:2, ytick=-1.5:0.5:1.5)
    title!("ellipse and regression line")
end
gif(anim, "images/ellipse_and_regressionline.gif")
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\ellipse_and_regressionline.gif
@ Plots D:\.julia\packages\Plots\mnkn4\src\animation.jl:114
```

Out[63]:

5 モーメントとキュムラントと歪度と尖度

5.1 モーメントとその母函数と特性函数とキュムラント母函数の定義

確率変数 X と $m = 0, 1, 2, \ldots$ について

$$\mu_m(X) = E[X^m]$$

を X の m 次の **モーメント**(moment, 積率) と呼び,

$$M_X(t) = E[e^{tX}] = E\left[\sum_{m=0}^{\infty} X^m \frac{t^m}{m!}\right] = \sum_{m=0}^{\infty} E[X^m]\frac{t^m}{m!} = \sum_{m=0}^{\infty} \mu_m(X)\frac{t^m}{m!}$$

を **モーメント母函数** (moment generating function, mgf)と呼ぶ.

X が従う確率分布の名前が Dist のとき, これらを **分布** Dist **のモーメントとモーメント母函数** と呼ぶ. 以下も同様である.

モーメント母函数の定義で t を $it = \sqrt{-1}\,t$ で置き換えたもの

$$
\begin{aligned}
\varphi_X(t) = E[e^{itX}] &= E\left[\sum_{m=0}^{\infty} i^m X^m \frac{t^m}{m!}\right] \\
&= \sum_{m=0}^{\infty} i^m E[X^m] \frac{t^m}{m!} = \sum_{m=0}^{\infty} i^m \mu_m(X) \frac{t^m}{m!}
\end{aligned}
$$

を **特性函数** (characteristic function)と呼ぶ. 特性函数を扱う場合には $i = \sqrt{-1}$ としたいので, i を番号の意味で使わないように気を付ける必要がある.

モーメント母函数だけではなく, 特性函数もモーメント達の母函数になっている.

モーメント母函数の対数

$$
K_X(t) = \log M(t) = \log E[e^{tX}] = \sum_{m=1}^{\infty} \kappa_m(X) \frac{t^m}{m!} = \sum_{m=1}^{\infty} \kappa_m \frac{t^m}{m!}
$$

を **キュムラント母函数** (cumulant generating function, cgf) と呼び, その展開係数 $\kappa_m = \kappa_m(X)$ を X の m 次のキュムラントと呼ぶ.

注意: 取り得る値が実数になる確率変数 X について $|e^{itX}| = 1$ となるので, $E[e^{itX}]$ は常に絶対収束しており, 特性函数は常にうまく定義されている. それに対して e^{tX} の値は巨大になる可能性があり, $E[e^{tX}]$ が収束しない場合が出て来る. モーメント母函数やキュムラント母函数の取り扱いではこの点に注意する必要がある.

注意: モーメント母函数とキュムラント母函数はそれぞれ物理での統計力学における **分配函数** と **自由エネルギー** (もしくは **Massieu函数**)の確率論的な類似物になっている. ただし, 逆温度 β について $t = -\beta$ とおく必要がある. 逆に言えば, モーメント母函数とキュムラントの表示における $\beta = -t$ の逆数は絶対温度の確率論的類似物になっていることになる.

5.2 特性函数による期待値の表示

X が確率密度函数 $p(x)$ を持つとき, 函数 $f(x)$ のFourier変換を

$$
\hat{f}(t) = \int_{-\infty}^{\infty} f(x) e^{-itx}\, dx
$$

と書くと, $f(t)$ がそう悪くない函数ならば逆Fourier変換によってもとの函数に戻せる:

$$
f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(t) e^{itx}\, dt.
$$

Fourier解析の基礎については次のリンク先を参照せよ(逆Fourier変換に関する結果はこのノート内では認めて使ってよい):

- [12 Fourier解析 (https://nbviewer.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb)](https://nbviewer.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb)

ゆえに, x に確率変数 X を代入して両辺の期待値を取り, 期待値を取る操作と積分を交換すると,

$$
E[f(X)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(t) E[e^{itX}]\, dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(t) \varphi_X(t)\, dt.
$$

ここで $\varphi_X(t) = E[e^{itX}]$ は X の特性函数である.

確率変数 X が従う分布は様々な函数 $f(x)$ に関する期待値 $E[f(X)]$ から決まるので, $E[f(X)]$ が X の特性函数 $\varphi_X(t)$ を用いて表せたということは, **X の特性函数 $\varphi_X(t)$ から X が従う分布が唯一つに決まる** ことを意味している.

さらに, 分布Aの特性函数が分布Bの特性函数で近似されていれば, 分布Aにおける期待値が分布Bにおける期待値で近似されることもわかる. これは **分布の近似を特性函数の近似で確認できる** ことを意味する.

モーメント母函数 $M_X(t) = E[e^{tX}]$ が t の函数として, 定義域を自然に複素数まで拡張できているとき(正確には解析接続できていれば), $\varphi_X(t) = M_X(it)$ が成立する. キュムラント母函数はモーメント母函数の対数である. これらより, **モーメント母函数やキュムラント母函数からも分布が唯一つに決まる** ことがわかる.

5.3 問題: 分布のアフィン変換のキュムラント

確率変数 X と $a, b \in \mathbb{R}$, $a \neq 0$ について, $aX + b$ のモーメント母函数とキュムラント母函数とキュムラントが X のそれらで次のように表されることを示せ:

$$
\begin{aligned}
&M_{aX+b}(t) = e^{bt} M_X(at), \quad K_{aX+b}(t) = K_X(at) + bt, \\
&\kappa_1(aX+b) = a\kappa_1(X) + b, \quad \kappa_m(aX+b) = a^m \kappa_m(X) \quad (m = 2, 3, 4, \dots)
\end{aligned}
$$

注意: キュムラントの変換公式は非常に単純な形になる. $\kappa_1(aX+b) = a\kappa_1(X)+b$ は $\kappa_1(X) = E[X]$ だったので当然である. 2 次以上のキュムラントは a^m 倍されるだけになる. モーメント母函数の対数を取ってキュムラント母函数を定義し, その展開によってキュムラントを定義することにはこのような利点がある. キュムラント母函数を定義することには, 物理の統計力学で分配函数の対数を取って自由エネルギーを定義することと同様の利点がある.

注意: この結果は空気のごとく使われる.

解答例: $aX+b$ のモーメント母函数を X のモーメント母函数であらわそう:

$$M_{aX+b}(t) = E[e^{t(aX+b)}] = e^{bt}E[e^{atX}] = e^{bt}M_X(at).$$

ゆえに, $aX+b$ のキュムラントは次の形になる:

$$K_{aX+b}(t) = \log M_{aX+b}(t) = \log(e^{bt}M_X(at)) = K_X(at)+bt.$$

$X \mapsto aX+b$ によってキュムラント母函数は $K_X(t) \mapsto K_X(at)+bt$ に似た形式で変換される.

X のキュムラント $\kappa_m(X)$ は次のようにキュムラント母函数を展開することによって定義されるのであった:

$$K_X(t) = \sum_{m=1}^{\infty} \kappa_m(X)\frac{t^m}{m!}.$$

$K_{aX+b}(t)$ の展開結果は

$$\begin{aligned} K_{aX+b}(t) = K_X(at)+bt &= \sum_{m=0}^{\infty} \kappa_m(X)\frac{(at)^m}{m!} + bt \\ &= (a\kappa_1(X)+b)t + \sum_{m=2}^{\infty} a^m\kappa_m(X)\frac{t^m}{m!} \end{aligned}$$

になるので,

$$\kappa_1(aX+b) = a\kappa_1(X)+b, \quad \kappa_m(aX+b) = a^m\kappa_m(X) \quad (m=2,3,4,\ldots)$$

となることがわかる.

解答終

5.4 問題: 標準正規分布のモーメント母函数と特性函数とキュムラント母函数

標準正規分布に従う確率変数 Z のモーメント母函数と特性函数とキュムラント母函数が次のようになることを示せ:

$$M_Z(t) = e^{t^2/2}, \quad \varphi_Z(t) = M_Z(it) = e^{-t^2/2}, \quad K_Z(t) = \log M_Z(t) = \frac{t^2}{2}.$$

この結果は中心極限定理の証明で使われる.

解答例: $Z \sim \mathrm{Normal}(0,1)$ と仮定する. このとき,

$$tz - \frac{z^2}{2} = -\frac{z^2-2tz}{2} = -\frac{(z-t)^2-t^2}{2} = -\frac{(z-t)^2}{2} + \frac{t^2}{2}$$

なので,

$$\begin{aligned} M_Z(t) = E[e^{tZ}] &= \int_{-\infty}^{\infty} e^{tz}\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} e^{-(z-t)^2/2+t^2/2}\,dz \\ &= \frac{e^{t^2/2}}{\sqrt{2\pi}}\int_{-\infty}^{\infty} e^{-w^2/2}\,dw = e^{t^2/2}. \end{aligned}$$

4つめの等号で $z = w+t$ とおいた. ゆえに,

$$\varphi_Z(t) = E[e^{itZ}] = M_Z(it) = e^{-t^2/2}, \quad K_Z(t) = \log M_Z(t) = \frac{t^2}{2}.$$

解答終

注意: 標準正規分布のキュムラント母函数は $t^2/2$ というたったの一項だけになってしまう. 標準でない正規分布のキュムラント母函数は t について1次と2次の項だけになる. 1次の項の係数は分布の期待値で, $t^2/2$ の係数は分散になる. 実際, 平均 μ, 分散 σ^2 を持つ確率変数 X について,

$$
\begin{aligned}
K_X(t) &= \log E[e^{tX}] = \log E[e^{t(X-\mu)+t\mu}] \\
&= t\mu + \log\left(1 + E[X-\mu]t + E[(X-\mu)^2]\frac{t^2}{2} + O(t^3)\right) \\
&= t\mu + \log\left(1 + \sigma^2\frac{t^2}{2} + O(t^3)\right) = t\mu + \sigma^2\frac{t^2}{2} + O(t^3).
\end{aligned}
$$

そして, $\sigma Z + \mu \sim \mathrm{Normal}(\mu, \sigma)$ であり,

$$
M_{\sigma Z+\mu}(t) = E[e^{t(\sigma Z+\mu)}] = e^{\mu t}E[e^{t\sigma Z}] = e^{\mu t}M_Z(\sigma t) = e^{\mu t + \sigma^2 t^2/2}.
$$

ゆえに

$$
K_{\sigma Z+\mu}(t) = \log M_{\sigma Z+\mu}(t) = \mu t + \sigma^2\frac{t^2}{2}.
$$

他の分布のキュムラント母函数を計算したときに出て来る t について3次以上の項はその分布が正規分布からどれだけ離れているかを表している.

5.5 確率変数の標準化と標準化キュムラントと歪度と尖度

確率変数 X は確率変数であるとし, $\mu = E[X]$, $\sigma = \sqrt{E[(X-\mu)^2]}$ とおく. このとき,

$$
Z = \frac{X-\mu}{\sigma}
$$

を確率変数の **標準化** (standardization)と呼ぶ. Z の期待値と分散はそれぞれ 0 と 1 になる.

X の標準化のモーメントやキュムラントおよびそれらの母函数をそれぞれ **標準化モーメント**, **標準化キュムラント** と呼び, それぞれを $\bar{\mu}_m(X)$, $\bar{\kappa}_m(X)$ と表す. 詳しくは以下の通り:

$$
\begin{aligned}
&\bar{\mu}_m(X) = \mu_m(Z) = E\left[\left(\frac{X-\mu}{\sigma}\right)^m\right], \\
&M_Z(t) = E\left[\exp\left(t\frac{X-\mu}{\sigma}\right)\right] = \sum_{m=0}^{\infty}\bar{\mu}_m(X)\frac{t^m}{m!} = 1 + \frac{t^2}{2} + \bar{\mu}_3(X)\frac{t^3}{3!} + \bar{\mu}_4(X)\frac{t^4}{4!} + \cdots, \\
&K_Z(t) = \log M_Z(t) = \sum_{m=1}^{\infty}\bar{\kappa}_m(X)\frac{t^m}{m!} = \frac{t^2}{2} + \bar{\kappa}_3(X)\frac{t^3}{3!} + \bar{\kappa}_4(X)\frac{t^4}{4!} + \cdots.
\end{aligned}
$$

$\bar{\kappa}_3(X)$ と $\bar{\kappa}_4(X)$ は次のように表される:

$$
\bar{\kappa}_3(X) = \bar{\mu}_3(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right], \quad \bar{\kappa}_4(X) = \bar{\mu}_4(X) - 3 = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] - 3.
$$

このことは, $\log(1+a) = a - a^2/2 + O(a^3)$ を使って以下のようにして確認される:

$$
\begin{aligned}
\log\left(1 + \frac{t^2}{2} + \bar{\mu}_3(X)\frac{t^3}{3!} + \bar{\mu}_4(X)\frac{t^4}{4!} + O(t^5)\right) &= \frac{t^2}{2} + \bar{\mu}_3(X)\frac{t^3}{3!} + \bar{\mu}_4(X)\frac{t^4}{4!} - \frac{1}{2}\left(\frac{t^2}{2}\right)^2 + O(t^5) \\
&= \frac{t^2}{2} + \bar{\mu}_3(X)\frac{t^3}{3!} + (\bar{\mu}_4(X) - 3)\frac{t^4}{4!} + O(t^5).
\end{aligned}
$$

$\bar{\kappa}_3(X)$ を X もしくは X が従う分布の **歪度** (わいど, skewness) と呼び, $\bar{\kappa}_4(X)$ を **尖度** (せんど, kurtosis)と呼び, 次のようにも書くことにする:

$$
\mathrm{skewness}(X) = \bar{\kappa}_3(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right], \quad \mathrm{kurtosis}(X) = \bar{\kappa}_4(X) = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] - 3.
$$

歪度は左右の非対称性の尺度であり, 尖度は分布の尖り具合が正規分布とどれだけ違うかの尺度になっている.

X が正規分布に従う確率変数の場合にはその標準化 $Z = (X-\mu)/\sigma$ は標準正規分布に従う確率変数になるので, その標準化キュムラント達は $\bar{\kappa}_2(X) = 1$, $\bar{\kappa}_m(X) = 0$ $(m \neq 0)$ となる. 2次の標準化キュムラントは常に 1 になるが, 3 次以上の標準化キュムラントは X が正規分布でなければ 0 でなくなる.

このことから, 3 次以上の標準化キュムラントは分布が正規分布からどれだけ離れているかを表していると考えられる. それらのうち最初の2つが上で定義した歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ になっている.

分布の歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ は分布がどれだけ正規分布から離れているかを表す最も基本的な量である.

注意: $\bar{\kappa}_4(X) = \bar{\mu}_4(X) - 3$ ではなく, 3 を引く前の $\bar{\mu}_4(X)$ を尖度と定義する流儀もあるが, このノートでは正規分布の扱いでキュムラントが非常に便利なことを重視したいので, 3 を引いた側の $\bar{\kappa}_4$ を尖度の定義として採用する. 3 を引いた方の $\bar{\kappa}_4(X) = \bar{\mu}_4(X) - 3$ は **過剰尖度** (かじょうせんど, **excess kurtosis**)と呼ばれることも多い. 正規分布の尖度を($\bar{\kappa}_4$ の方の 0 ではなく) $\bar{\mu}_4$ の方の 3 とするときに, そこからどれだけ分布の尖り具合が増したかを $\bar{\kappa}(X)$ が表していることを「過剰」と表現している.

In [64]:
```
@vars t μ3 μ4 μ5 κ3 κ4 κ5
Mt = 1 + t^2/2 + μ3*t^3/6 + μ4*t^4/24 + μ5*t^5
expr = series(log(Mt), t)
```

Out[64]:
$$\frac{t^2}{2} + \frac{t^3\mu_3}{6} + t^4\left(\frac{\mu_4}{24} - \frac{1}{8}\right) + t^5\left(-\frac{\mu_3}{12} + \mu_5\right) + O\left(t^6\right)$$

5.6 問題: 正規分布の歪度と尖度

X が正規分布に従う確率変数であるとき, その歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ が以下のようになることを示せ:

$$\bar{\kappa}_3(X) = \bar{\kappa}_4(X) = 0.$$

解答例: X が正規分布 $\mathrm{Normal}(\mu, \sigma)$ に従う確率変数のとき, その標準化 $Z = (X - \mu)/\sigma$ は標準正規分布に従うので, $Z = (X - \mu)/\sigma$ のキュムラント母函数は $t^2/2$ になり, その 3 次以上の項がすべて 0 になる. ゆえに $\bar{\kappa}_m(X) = 0$ ($m = 3, 4, 5, \ldots$). 特に $\bar{\kappa}_3(X) = \bar{\kappa}_4(X) = 0$.

解答終

5.7 問題: 一様分布のキュムラント母函数と歪度と尖度

一様分布 $\mathrm{Uniform}(0, 1)$ に従う確率変数 X について, そのキュムラント母函数 $K_X(t) = \log E[e^{tX}]$ と歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ が以下のようになることを示せ:

$$K_X(t) = \log\frac{e^t - 1}{t}, \quad \bar{\kappa}_3(X) = 0, \quad \bar{\kappa}_4(X) = -\frac{6}{5}.$$

一様分布は歪度が 0 でかつ尖度がかなり小さな連続分布の例になっている.

解答例: モーメント母函数とキュムラント母函数は以下のように計算される:

$$E[e^{tX}] = \int_0^1 e^{tx}\,dx = \left[\frac{e^{tx}}{t}\right]_0^1 = \frac{e^t - 1}{t},$$
$$K_X(t) = \log\frac{e^t - 1}{t} = \frac{1}{2}t + \frac{1}{12}\frac{t^2}{2} - \frac{1}{120}\frac{t^4}{4!} + O(t^5).$$

ゆえに, $\mu = 1/2$, $\sigma = \sqrt{1/12}$, $Z = (X - \mu)/\sigma$ とおくと,

$$K_Z(t) = K_X\left(\frac{t}{\sigma}\right) - \frac{\mu}{\sigma}t = \frac{t^2}{2} - \frac{6}{5}\frac{t^4}{4!} + O(t^5).$$

ゆえに $\bar{\kappa}_3(X) = 0$, $\bar{\kappa}_4 = -6/5$.

解答終

In [65]:
```
@vars t
K_X = log((exp(t) - 1)/t)
κ = [limit(diff(K_X, t, m) , t, 0) for m in 1:4]
```

Out[65]:
$$\begin{bmatrix} \frac{1}{2} \\ \frac{1}{12} \\ 0 \\ -\frac{1}{120} \end{bmatrix}$$

In [66]:
```
μ, σ = κ[1], √κ[2]
κ = [limit(diff(K_X(t⇒t/σ) - μ/σ*t, t, m) , t, 0) for m in 1:4]
```

Out[66]:
$$\begin{bmatrix} 0 \\ 1 \\ 0 \\ -\frac{6}{5} \end{bmatrix}$$

5.8 問題: 独立な確率変数達の和のモーメント母函数と特性函数とキュムラント母函数

独立な確率変数達 $X_1, \dots, X_n$ の和のモーメント母函数と特性函数とキュムラント母函数が次のように表されることを示せ:

$$\begin{aligned} M_{X_1+\cdots+X_n}(t) &= M_{X_1}(t)\cdots M_{X_n}(t), \\ \varphi_{X_1+\cdots+X_n}(t) &= \varphi_{X_1}(t)\cdots \varphi_{X_n}(t), \\ K_{X_1+\cdots+X_n}(t) &= K_{X_1}(t)+\cdots+K_{X_n}(t). \end{aligned}$$

注意: この結果は空気のごとく使われる.

解答例: 独立な確率変数達 $X_1, \dots, X_n$ について

$$E[f_1(X_1)\cdots f_n(X_n)] = E[f_1(X_1)]\cdots E[f_n(X_n)]$$

が成立することより,

$$\begin{aligned} M_{X_1+\cdots+X_n}(t) &= E[e^{t(X_1+\cdots+X_n)}] = E[e^{tX_1}\cdots e^{tX_n}] = M_{X_1}(t)\cdots M_{X_n}(t), \\ \varphi_{X_1+\cdots+X_n}(t) &= E[e^{it(X_1+\cdots+X_n)}] = E[e^{itX_1}\cdots e^{itX_n}] = \varphi_{X_1}(t)\cdots \varphi_{X_n}(t). \end{aligned}$$

ゆえに

$$\begin{aligned} K_{X_1+\cdots+X_n}(t) &= \log M_{X_1+\cdots+X_n}(t) = \log(M_{X_1}(t)\cdots M_{X_n}(t)) \\ &= \log M_{X_1}(t)+\cdots+\log M_{X_n}(t) = K_{X_1}(t)+\cdots+K_{X_n}(t). \end{aligned}$$

解答終

5.9 問題: ガンマ分布のキュムラント母函数と歪度と尖度

ガンマ分布 $\mathrm{Gamma}(\alpha,\theta)$ に従う確率変数 X について, そのキュムラント母函数 $K_X(t) = \log E[e^{tX}]$ と歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ が以下のようになることを示せ:

$$K_X(t) = -\alpha\log(1-\theta t), \quad \bar{\kappa}_3(X) = \frac{2}{\sqrt{\alpha}}, \quad \bar{\kappa}_3(X) = \frac{6}{\alpha}.$$

解答例: $t < 1/\theta$ と仮定する. このとき,

$$\mu = \alpha\theta, \quad \sigma = \sqrt{\alpha}\,\theta, \quad Z = \frac{X-\mu}{\sigma}$$

とおくと,

$$\begin{aligned} E[e^{tX}] &= \int_0^\infty e^{tx}\frac{e^{-x/\theta}x^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}\,dx = \frac{1}{\theta^\alpha\Gamma(\alpha)}\int_0^\infty e^{-((1-\theta t)/\theta)x}x^{\alpha-1}\,dx \\ &= \frac{1}{\theta^\alpha\Gamma(\alpha)}\frac{\theta^\alpha}{(1-\theta t)^\alpha}\Gamma(\alpha) = \frac{1}{(1-\theta t)^\alpha}, \\ K_X(t) &= -\alpha\log(1-\theta t) \\ &= \alpha\theta t + \alpha\theta^2\frac{t^2}{2} + \alpha\theta^3\frac{t^3}{3} + \alpha\theta^4\frac{t^4}{4} + O(t^5) \\ &= \mu t + \sigma^2\frac{t^2}{2} + \frac{2\sigma^3}{\sqrt{\alpha}}\frac{t^3}{3!} + \frac{6\sigma^4}{\alpha}\frac{t^4}{4!} + O(t^5), \\ K_Z(t) &= K_X\left(\frac{t}{\sigma}\right) - \frac{\mu}{\sigma}t = \frac{t^2}{2} + \frac{2}{\sqrt{\alpha}}\frac{t^3}{3!} + \frac{6}{\alpha}\frac{t^4}{4!} + O(t^5). \end{aligned}$$

ゆえに,

$$\bar{\kappa}_3(X) = \frac{2}{\sqrt{\alpha}}, \quad \bar{\kappa}_3(X) = \frac{6}{\alpha}.$$

解答終

In [67]:
```
@vars t
@vars α θ positive=true
K_X = -α*log(1 - θ*t)
κ = [diff(K_X, t, m)(t⇒0).simplify().factor() for m in 1:4]
```

Out[67]:

$$\begin{bmatrix} \alpha\theta \\ \alpha\theta^2 \\ 2\alpha\theta^3 \\ 6\alpha\theta^4 \end{bmatrix}$$

In [68]:
```
μ, σ = κ[1], √κ[2]
[diff(K_X(t⇒t/σ) - μ/σ*t, t, m)(t⇒0).simplify().factor() for m in 1:4]
```

Out[68]:

$$\begin{bmatrix} 0 \\ 1 \\ \frac{2}{\sqrt{\alpha}} \\ \frac{6}{\alpha} \end{bmatrix}$$

5.10 問題: 二項分布のキュムラント母函数と歪度と尖度

二項分布 $\mathrm{Binomial}(n,p)$ に従う確率変数 X について, そのキュムラント母函数 $K_X(t) = \log E[e^{tX}]$ と歪度 $\bar{\kappa}_3(X)$ と尖度 $\bar{\kappa}_4(X)$ が以下のようになることを示せ:

$$K_X(t) = n\log(1+p(e^t-1)), \quad \bar{\kappa}_3(X) = \frac{1-2p}{\sqrt{np(1-p)}}, \quad \bar{\kappa}_4(X) = \frac{1-6p(1-p)}{np(1-p)}.$$

注意: $n=1, p=1/2$ のとき $\bar{\kappa}_4(X) = -2$ となる. 実はその場合の分布 $\mathrm{Bernoulli}(1/2)$ が確率分布全体の中で最小の尖度を持つ分布になっている.

解答例: $X_1, \dots, X_n$ をそれぞれがBernoulli分布 $\mathrm{Binomial}(p)$ に従う独立同分布な確率変数達であるとする. このとき, 二項分布に従う確率変数は $X = \sum_{i=1}^n X_i$ と作れる. ゆえに

$$K_X(t) = \sum_{i=1}^n K_{X_i}(t) = nK_{X_1}(t).$$

$X_1 \sim \mathrm{Bernoulli}(p)$ について,

$$\begin{aligned} &E[e^{tX_1}] = e^{t1}p + e^{t0}(1-p) = 1 + p(e^t-1), \\ &K_{X_1}(t) = \log(1+p(e^t-1)), \\ &K_X(t) = n\log(1+p(e^t-1)) \end{aligned}$$

なので,

$$\mu = np, \quad \sigma = \sqrt{np(1-p)}$$

とおくと,

$$\begin{aligned} &K_X(t) = n\log(1+p(e^t-1)) \\ &= npt + np(1-p)\frac{t^2}{2} + np(1-p)(1-2p)\frac{t^3}{3!} + np(1-p)(6p^2-6p+1)\frac{t^4}{4!} + O(t^5) \\ &= \mu t + \sigma^2\frac{t^2}{2} + \sigma^2(1-2p)\frac{t^3}{3!} + \sigma^2(1-6p(1-p))\frac{t^4}{4!} + O(t^5), \\ &K_{(X-\mu)/\sigma}(t) = K_X\left(\frac{t}{\sigma}\right) - \frac{\mu}{\sigma}t = \frac{t^2}{2} + \frac{1-2p}{\sigma}\frac{t^3}{3!} + \frac{1-6p(1-p)}{\sigma^2}\frac{t^4}{4!} + O(t^5). \end{aligned}$$

ゆえに

$$\bar{\kappa}_3(X) = \frac{1-2p}{\sigma} = \frac{1-2p}{\sqrt{np(1-p)}}, \quad \bar{\kappa}_4(X) = \frac{1-6p(1-p)}{\sigma^2} = \frac{1-6p(1-p)}{np(1-p)}.$$

解答終

In [69]:
```
kurtosis(Bernoulli(0.5))
```

Out[69]: -2.0

In [70]:
```
@vars n p t
K_X = n * log(1 + p*(exp(t) - 1))
κ = [diff(K_X, t, m)(t⇒0).simplify().factor() for m in 1:4]
```

Out[70]:

$$\begin{bmatrix} np \\ -np\,(p-1) \\ np\,(p-1)\,(2p-1) \\ -np\,(p-1)\,(6p^2-6p+1) \end{bmatrix}$$

In [71]:
```
μ, σ = κ[1], √κ[2]
[diff(K_X(t⇒t/σ) - μ/σ*t, t, m)(t⇒0).simplify().factor() for m in 1:4]
```

Out[71]:

$$\begin{bmatrix} 0 \\ 1 \\ -\dfrac{2p-1}{\sqrt{-np\,(p-1)}} \\ -\dfrac{6p^2-6p+1}{np\,(p-1)} \end{bmatrix}$$

6 独立同分布な確率変数達の不偏分散の分散

以下の節においては, $X_1, \ldots, X_n$ は独立同分布な確率変数達であるとし, それらの標本分散と不偏分散を以下のように書くことにする:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad S_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2.$$

さらに, X_i の期待値と分散をそれぞれ μ, σ^2 と書くことにする.

このとき, 以下が成立することはすでに示した:

$$E[\bar{X}] = \mu, \quad E[S_X^2] = \sigma^2, \quad \mathrm{var}(X) = E[(\bar{X}-\mu)^2] = \frac{\sigma^2}{n}.$$

さらに以下では, X_i の **歪度** (わいど, skewness)と **尖度** (せんど, kurtosis)をそれぞれ $\bar{\kappa}_3 = \bar{\kappa}_3(X_i)$, $\bar{\kappa}_4 = \bar{\kappa}_4(X_i)$ と書くことにする:

$$\bar{\kappa}_3 = E\left[\left(\frac{X_i-\mu}{\sigma}\right)^3\right], \quad \bar{\kappa}_4 = E\left[\left(\frac{X_i-\mu}{\sigma}\right)^4\right] - 3.$$

このとき, 標本平均と不偏分散の共分散と不偏分散の分散は X_i の歪度 $\bar{\kappa}_3$ と尖度 $\bar{\kappa}_4$ を使って次のように書ける:

$$\mathrm{cov}(\bar{X}, S_X^2) = \sigma^3\frac{\bar{\kappa}_3}{n}, \quad \mathrm{var}(S_X^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right).$$

以下ではこの結果を証明する.

6.1 期待値0, 分散1の場合への帰着

$Z = (X-\mu)/\sigma$ とおくと, Z の期待値と分散はそれぞれ 0 と 1 になる.

$Z_i = (X_i-\mu)/\sigma$ とおくと, $Z_1, \ldots, Z_n$ はそれぞれが Z と同じ分布に従う独立同分布な確率変数達になる. そして, $X_i = \sigma Z_i + \mu$ より,

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i, \quad S_Z^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Z_i - \bar{Z})^2$$

とおくと,

$$\bar{X} = \sigma\bar{Z} + \mu, \quad S_X^2 = \sigma^2 S_Z^2$$

なので,

$$\mathrm{cov}(\bar{X}, S_X^2) = \sigma^3 \,\mathrm{cov}(\bar{Z}, S_Z^2), \quad \mathrm{var}(S_X^2) = \sigma^4 \,\mathrm{var}(S_Z^2).$$

これより, 標本分散と不偏分散の分散と共分散の計算は $\mu = 0, \sigma = 1$ の場合に帰着する.

6.2 標本平均と不偏分散の共分散の計算

標本平均と不偏分散の共分散は X_i の歪度 $\bar{\kappa}_3$ を使って次のように書ける:

$$\mathrm{cov}(\bar{X}, S_X^2) = \frac{\bar{\kappa}_3}{n}\sigma^3.$$

証明: $\mu = 0, \sigma = 1$ と仮定する. このとき, $\mu = 0$ という仮定より,

$$\mathrm{cov}(\bar{X}, S_X^2) = E[\bar{X}(S_X^2 - \sigma^2)] = E[\bar{X}S_X^2] - \sigma^2 \underbrace{E[\bar{X}]}_{=0} = E[\bar{X}S_X^2].$$

そして,

$$\begin{aligned}
&n\bar{X}^2 = \frac{1}{n}\sum_{i=1}^n X_i^2 + \frac{2}{n}\sum_{1\le i<j\le n} X_iX_j, \\
&(n-1)S_X^2 = \sum_{i=1}^n (X_i - \bar{X})^2 = \sum_{i=1}^n (X_i^2 - 2\bar{X}X_i + \bar{X}^2) = \sum_{i=1}^n X_i^2 - n\bar{X}^2 \\
&= \sum_{i=1}^n X_i^2 - \frac{1}{n}\sum_{i=1}^n X_i^2 - \frac{2}{n}\sum_{1\le i<j\le n} X_iX_j = \frac{n-1}{n}\sum_{i=1}^n X_i^2 - \frac{2}{n}\sum_{1\le i<j\le n} X_iX_j, \\
&\therefore \quad S_X^2 = \frac{1}{n}\sum_{i=1}^n X_i^2 - \frac{2}{n(n-1)}\sum_{1\le i<j\le n} X_iX_j.
\end{aligned}$$

ゆえに

$$\bar{X}S_X^2 = \frac{1}{n}\sum_{i=1}^n X_i \left(\frac{1}{n}\sum_{i=1}^n X_i^2 - \frac{2}{n(n-1)}\sum_{1\le i<j\le n} X_iX_j\right).$$

$E[X_i] = \mu = 0$ という仮定と X_i 達の独立性より, i, j, k がすべて等しいとき以外には $E[X_iX_jX_k] = 0$ となり, $\mu = 0, \sigma = 1$ という仮定より $E[X_i^3] = \bar{\kappa}_3$ となるので,

$$\mathrm{cov}(\bar{X}, S_X^2) = E[\bar{X}S_X^2] = \frac{1}{n^2}\sum_{i=1}^n E[X_i^3] = \frac{\bar{\kappa}_3}{n}.$$

$\mu = 0, \sigma = 1$ と仮定しない一般の場合には, これが σ^3 倍されて,

$$\mathrm{cov}(\bar{X}, S_X^2) = \sigma^3\frac{\bar{\kappa}_3}{n}.$$

証明終

6.3 不偏分散の分散の計算

不偏分散の分散は X_i の尖度 $\bar{\kappa}_4$ を使って次のように書ける:

$$\mathrm{var}(S_X^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right).$$

証明: $\mu = 0, \sigma = 1$ と仮定する. このとき, $E[S_X^2] = \sigma^2 = 1$ より,

$$\mathrm{var}(S_X^2) = E[(S_X^2)^2] - E[S_X^2]^2 = E[(S_X^2)^2] - 1.$$

そして, 前節の計算より

$$S_X^2 = \frac{1}{n}\sum_{i=1}^n X_i^2 - \frac{2}{n(n-1)}\sum_{1\le i<j\le n} X_iX_j$$

なので,

$$
\begin{aligned}
(S_X^2)^2 = \frac{1}{n^2}\left(\sum_{i=1}^n X_i^4 + 2\sum_{1\le i<j\le n} X_i^2 X_j^2\right) \\
- \frac{2}{n^2(n-1)}\sum_{i=1}^n \sum_{1\le j<k\le n} X_i^2 X_j X_k \\
+ \frac{4}{n^2(n-1)^2}\sum_{1\le i<j\le n}\sum_{1\le k<l\le n} X_i X_j X_k X_l.
\end{aligned}
$$

$\mu = 0, \sigma = 1$ でかつ $X_1, \ldots, X_n$ は独立であるという仮定と, $E[X_i] = 0$, $E[X_i^2] = 1$, $E[X_i^4] = \bar{\kappa}_4 + 3$ であることを使うと,

$$
\begin{aligned}
E\left[\sum_{i=1}^n X_i^4\right] &= n(\bar{\kappa}_4 + 3), \\
E\left[2\sum_{1\le i<j\le n} X_i^2 X_j^2\right] &= 2\frac{n(n-1)}{2} = n(n-1), \\
E\left[\sum_{1\le j<k\le n} X_i^2 X_j X_k\right] &= 0, \\
E\left[\sum_{1\le i<j\le n}\sum_{1\le k<l\le n} X_i X_j X_k X_l\right] &= \frac{n(n-1)}{2}.
\end{aligned}
$$

最後の等式は $i = k, j = l$ の場合のみが和に寄与することを使った. ゆえに

$$
\begin{aligned}
\mathrm{var}(S_X^2) &= \frac{n(\bar{\kappa}_4 + 3) + n(n-1)}{n^2} + \frac{4}{n^2(n-1)^2}\frac{n(n-1)}{2} - 1 \\
&= \frac{\bar{\kappa}_4 + n + 2}{n} + \frac{2}{n(n-1)} - 1 \\
&= \frac{\bar{\kappa}_4 + n + 2}{n} + \frac{2}{n-1} - \frac{2}{n} - \frac{n}{n} \\
&= \frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}.
\end{aligned}
$$

$\mu = 0, \sigma = 1$ と仮定しない一般の場合には, これが σ^4 倍されて,

$$
\mathrm{var}(S_X^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right).
$$

証明終

6.4 歪度と尖度に関する不等式

以上の結果を用いて, 歪度 $\bar{\kappa}_3$ と尖度 $\bar{\kappa}_4$ のあいだには

$$
\bar{\kappa}_4 \ge \bar{\kappa}_3^2 - 2 \ge -2
$$

が成立することを示そう. 標本平均 $\bar{X}$ と不偏分散 S_X^2 の分散共分散行列

$$
\begin{bmatrix} \mathrm{var}(\bar{X}) & \mathrm{cov}(\bar{X}, S_X^2) \\ \mathrm{cov}(S_X^2, \bar{X}) & \mathrm{var}(S_X^2) \end{bmatrix} = \begin{bmatrix} \sigma^2/n & \sigma^3\bar{\kappa}_3/n \\ \sigma^3\bar{\kappa}_3/n & \sigma^4(\bar{\kappa}_4/n + 2/(n-1)) \end{bmatrix}
$$

は半正定値になる. 特にその行列式は 0 以上になる:

$$
0 \le \frac{\sigma^2}{n}\sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right) - \left(\frac{\sigma^3\bar{\kappa}_3}{n}\right)^2 = \frac{\sigma^6}{n^2}\left(\bar{\kappa}_4 + \frac{2}{1-1/n} - \bar{\kappa}_3^2\right).
$$

ゆえに,

$$
\bar{\kappa}_4 \ge \bar{\kappa}_3^2 - \frac{2}{1-1/n}.
$$

$n \to \infty$ とすることによって,

$$
\bar{\kappa}_4 \ge \bar{\kappa}_3^2 - 2.
$$

を得る.

注意: 上の不等式は **Karl Pearson の不等式** と呼ばれることがある.

- Pearson, Karl. IX. Mathematical contributions to the theory of evolution.—XIX. Second supplement to a memoir on skew variation. Philosophical Transactions of the Royal Society of London A, 216 (546): 429–457. [doi (https://doi.org/10.1098/rsta.1916.0009)]

注意: 実は上で紹介した歪度と尖度に関する不等式の証明はひどく遠回りになっている. 分散共分散行列の半正定値性(非負性)に帰着するのであれば, 確率変数 X について $X-\mu$ と $(X-\mu)^2$ の分散共分散行列を求めれみればよい.

$$E[X]=\mu,\quad E[(X-\mu)^2]=\sigma^2,\quad E[(X-\mu)^3]=\sigma^3\bar{\kappa}_3,\quad E[(X-\mu)^4]=\sigma^4(\bar{\kappa}_4+3)$$

のとき,

$$\begin{aligned}
&E[X-\mu]=0,\quad E[(X-\mu)^2]=\sigma^2,\\
&\mathrm{var}(X-\mu)=\sigma^2,\\
&\mathrm{cov}(X-\mu,(X-\mu)^2)=E[(X-\mu)^3]=\sigma^3\bar{\kappa}_3,\\
&\mathrm{var}((X-\mu)^2)=E[(X-\mu)^4]-E[(X-\mu)^2]^2=\sigma^4(\bar{\kappa}_4+3)-\sigma^4=\sigma^4(\bar{\kappa}_4+2).
\end{aligned}$$

これより, $X-\mu$ と $(X-\mu)^2$ の分散共分散行列

$$\begin{bmatrix}\sigma^2 & \sigma^3\bar{\kappa}_3\\ \sigma^3\bar{\kappa}_3 & \sigma^4(\bar{\kappa}_4+2)\end{bmatrix}=\begin{bmatrix}\sigma & 0\\ 0 & \sigma^2\end{bmatrix}\begin{bmatrix}1 & \bar{\kappa}_3\\ \bar{\kappa}_3 & \bar{\kappa}_4+2\end{bmatrix}\begin{bmatrix}\sigma & 0\\ 0 & \sigma^2\end{bmatrix}$$

は半正定値なので, 特にその行列式は 0 以上になり,

$$\bar{\kappa}_4+2\geq\bar{\kappa}_3^3$$

という不等式を得る.

6.5 問題: 不偏分散と不偏補正されていない標本分散の平均二乗誤差の比較

不偏補正されていない($n-1$ ではなく $n-1$ で割ってできる)標本分散

$$\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2=\frac{n-1}{n}S_X^2$$

と推定先の値 σ^2 の差の二乗の期待値(平均二乗誤差)が次のようになることを示せ:

$$E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right]=\left(\frac{n-1}{n}\right)^2\mathrm{var}(S_X^2)+\frac{\sigma^4}{n^2}.$$

これより, 次の公式が導かれることを示せ:

$$\mathrm{var}(S_X^2)-E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right]=\frac{\sigma^4}{n^2}\left(\left(2-\frac{1}{n}\right)\bar{\kappa}_4+3+\frac{2}{n-1}\right).$$

これより, $\bar{\kappa}_4\geq-3/2$ ならば不偏分散の平均二乗誤差の方が不偏補正されていない標本平均の平均二乗誤差よりも大きくなることがわかる.

注意: $\bar{\kappa}_4\geq\bar{\kappa}_3^2-2\geq-2$ なので多くの場合に $\bar{\kappa}_4>-3/2$ となる. 例えば正規分布では $\bar{\kappa}_4=0$ なので,

$$\mathrm{var}(S_X^2)-E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right]=\frac{\sigma^4}{n^2}\frac{3n-1}{n-1}$$

となり, 不偏分散の平均二乗誤差は補正されていない標本分散より大きくなる. 不偏分散の採用にはこのようなトレードオフも存在するが, この害は小さいと考えられる.

解答例: $(n-1)/n-1=-1/n$ と $E[S_X^2]=\sigma^2$, $E[(S_X^2-\sigma^2)^2]=\mathrm{var}(S_X^2)$ より,

$$
\begin{aligned}
E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right] &= E\left[\left(\frac{n-1}{n}(S_X^2-\sigma^2)-\frac{\sigma^2}{n}\right)^2\right] \\
&= E\left[\left(\frac{n-1}{n}\right)^2(S_X^2-\sigma^2)^2-2\frac{\sigma^2}{n}\frac{n-1}{n}(S_X^2-\sigma^2)+\frac{\sigma^4}{n^2}\right] \\
&= \left(\frac{n-1}{n}\right)^2\mathrm{var}(S_X^2)+\frac{\sigma^4}{n^2}.
\end{aligned}
$$

これより,

$$
\mathrm{var}(S_X^2)-E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right]=\frac{2n-1}{n^2}\mathrm{var}(S_X^2)-\frac{\sigma^4}{n^2}.
$$

これに $\mathrm{var}(S_X^2)=\sigma^4\left(\bar{\kappa}_4/n+2/(n-1)\right)$ を代入すると,

$$
\begin{aligned}
\mathrm{var}(S_X^2)-E\left[\left(\frac{n-1}{n}S_X^2-\sigma^2\right)^2\right] &= \frac{2n-1}{n^2}\sigma^4\left(\frac{\bar{\kappa}_4}{n}+\frac{2}{n-1}\right)-\frac{\sigma^4}{n^2} \\
&= \frac{\sigma^4}{n^2}\left(\frac{2n-1}{n}\bar{\kappa}_4+\frac{4n-2}{n-1}-1\right)=\frac{\sigma^4}{n^2}\left(\frac{2n-1}{n}\bar{\kappa}_4+\frac{3n-1}{n-1}\right) \\
&= \frac{\sigma^4}{n^2}\left(\left(2-\frac{1}{n}\right)\bar{\kappa}_4+3+\frac{2}{n-1}\right).
\end{aligned}
$$

解答終

In [72]:

```
function print_varS²_minus_MSEV²(dist, n; L=10^6)
    @show dist
    @show μ = mean(dist)
    @show σ = std(dist)
    @show κ = kurtosis(dist)
    @show n
    @show L = 10^6
    X = rand(dist, n, L)
    S² = var.(eachcol(X))
    V² = var.(eachcol(X); corrected=false)
    var_S² = mean((s² - σ^2)^2 for s² in S²)
    MSE_V² = mean((v² - σ^2)^2 for v² in V²)
    theoretical_val = σ^4/n^2*((2n-1)/n * κ + (3n-1)/(n-1))
    theoretical_value = theoretical_val
    @show theoretical_val
    @show var_S² - MSE_V²
    @show theoretical_value / σ^4
    @show (var_S² - MSE_V²) / σ^4
    nothing
end
```

Out[72]: print_varS²_minus_MSEV² (generic function with 1 method)

In [73]:

```
print_varS²_minus_MSEV²(Normal(1, 0.5), 5)
```

```
dist = Normal{Float64}(μ=1.0, σ=0.5)
μ = mean(dist) = 1.0
σ = std(dist) = 0.5
κ = kurtosis(dist) = 0.0
n = 5
L = 10 ^ 6 = 1000000
theoretical_val = 0.00875
var_S² - MSE_V² = 0.008711816245110061
theoretical_value / σ ^ 4 = 0.14
(var_S² - MSE_V²) / σ ^ 4 = 0.13938905992176098
```

正規分布の場合に確かに理論通りに不偏分散の平均二乗誤差の方が非補正標本分散の平均二乗誤差よりも大きくなっている.

In [74]:
```
print_varS²_minus_MSEV²(Bernoulli(0.5), 5)
```

```
dist = Bernoulli{Float64}(p=0.5)
μ = mean(dist) = 0.5
σ = std(dist) = 0.5
κ = kurtosis(dist) = -2.0
n = 5
L = 10 ^ 6 = 1000000
theoretical_val = -0.0002500000000000002
var_S² - MSE_V² = -0.00025235920018789205
theoretical_value / σ ^ 4 = -0.0040000000000000036
(var_S² - MSE_V²) / σ ^ 4 = -0.004037747203006273
```

これが尖度が最小の場合. 不偏分散の平均二乗誤差の方を確かに小さくできているが, ほんの少しだけである.

7 標本分布における標本平均と不偏分散の同時分布の視覚化

以下の節においては, $X_1, \ldots, X_n$ は独立同分布な確率変数達であるとし, それらの標本分散と不偏分散を以下のように書くことにする:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2.$$

さらに, 各 X_i の期待値と分散と歪度と尖度をそれぞれ $\mu, \sigma^2, \bar{\kappa}_3, \bar{\kappa}_4$ と書くことにする. このとき, 標本平均 $\bar{X}$ と不偏分散 S^2 の同時分布の期待値と分散・共分散は以下のようになるのであった:

$$\begin{aligned}
&E[\bar{X}] = \mu, \quad E[S^2] = \sigma^2, \\
&\mathrm{var}(\bar{X}) = \frac{\sigma^2}{n}, \quad \mathrm{cov}(\bar{X}, S^2) = \sigma^3\frac{\bar{\kappa}_3}{n}, \quad \mathrm{var}(S^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right).
\end{aligned}$$

これを, 期待値ベクトルと分散共分散行列の形式で書くと,

$$\begin{aligned}
&\begin{bmatrix} E[\bar{X}] \\ E[S^2] \end{bmatrix} = \begin{bmatrix} \mu \\ \sigma^2 \end{bmatrix}, \\
&\begin{bmatrix} \mathrm{var}(\bar{X}) & \mathrm{cov}(\bar{X}, S^2) \\ \mathrm{cov}(S^2, \bar{X}) & \mathrm{var}(S^2) \end{bmatrix} = \frac{1}{n}\begin{bmatrix} \sigma^2 & \sigma^3\bar{\kappa}_3 \\ \sigma^3\bar{\kappa}_3 & \sigma^4\left(\bar{\kappa}_4 + \frac{2}{1-1/n}\right) \end{bmatrix} \\
&\qquad = \frac{1}{n}\begin{bmatrix} \sigma & 0 \\ 0 & \sigma^2 \end{bmatrix}\begin{bmatrix} 1 & \bar{\kappa}_3 \\ \bar{\kappa}_3 & \bar{\kappa}_4 + \frac{2}{1-1/n} \end{bmatrix}\begin{bmatrix} \sigma & 0 \\ 0 & \sigma^2 \end{bmatrix}.
\end{aligned}$$

以下では X_i の従う分布が色々な場合に $\bar{X}, S^2$ の同時分布を視覚化してみる. その結果を見ると, 一般に以下が成立していそうなことがわかる:

- 標本平均 $\bar{X}$ と不偏分散 S^2 の組 $(\bar{X}, S^2)$ の平面上での分布は, n が十分に大きければ (μ, σ^2) を中心とする楕円状の小さな領域内に集中する分布(実は多変量正規分布で近似される)になり(中心極限定理!), $n \to \infty$ で (μ, σ^2) に集中するようになる(大数の法則!).
- もとの分布が対称ならば $(\bar{X}, S^2)$ の分布も左右対称になり, もとの分布が非対称ならば $(\bar{X}, S^2)$ の分布も左右非対称になり, n が大きなとき斜めに傾いた楕円状の領域内に集中する分布になる.

In [75]:
```
myskewness(dist) = skewness(dist)
myskewness(dist::MixtureModel) = _myskewness(dist)
function _myskewness(dist)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> ((x-μ)/σ)^3*pdf(dist, x), extrema(dist)...)[1]
end

mykurtosis(dist) = kurtosis(dist)
mykurtosis(dist::MixtureModel) = _mykurtosis(dist)
function _mykurtosis(dist)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> ((x-μ)/σ)^4*pdf(dist, x), extrema(dist)...)[1] - 3
end
```

Out[75]:
```
_mykurtosis (generic function with 1 method)
```

In [76]:
```
skewness(Gamma(2, 3)), _myskewness(Gamma(2,3))
```
Out[76]: (1.414213562373095, 1.4142135623731023)

In [77]:
```
kurtosis(Gamma(2, 3)), _mykurtosis(Gamma(2,3))
```
Out[77]: (3.0, 2.999999999999999)

In [78]:

```julia
function plot_dist(dist; m = 4, xlim = :auto, kwargs...)
    if xlim isa Symbol || length(xlim) != 2
        μ, σ = mean(dist), std(dist)
        a = max(minimum(dist), μ - m*σ)
        b = min(maximum(dist), μ + m*σ)
        xmin, xmax = a - 0.1(b - a), b + 0.1(b - a)
    else
        xmin, xmax = xlim
    end
    if dist isa DiscreteUnivariateDistribution
        x = range(round(xmin) - 1, round(xmax) + 1, 1000)
        plot(x, x -> pdf(dist, round(x)); label="", xlim, kwargs...)
    else
        x = range(xmin, xmax, 1000)
        plot(x, x -> pdf(dist, x); label="", xlim, kwargs...)
    end
end

function plot_X̄_and_SX²(dist; n = 10, L = 10^4,
        sk = myskewness(dist), ku = mykurtosis(dist),
        size = (400, 350), legend = :topleft, alpha = 0.4, kwargs...)
    μ, σ = mean(dist), std(dist)
    Σ = (1/n)*[σ^2 σ^3*sk; σ^3*sk σ^4*(ku+2/(1-1/n))]
    λ, U = eigen(Σ)
    h = quantile(Chisq(2), 0.95)
    f(t) = μ + √(λ[1]*h)*cos(t)*U[1,1] + √(λ[2]*h)*sin(t)*U[1,2]
    g(t) = σ^2 + √(λ[1]*h)*cos(t)*U[2,1] + √(λ[2]*h)*sin(t)*U[2,2]

    X̄ = Vector{Float64}(undef, L)
    S² = similar(X̄)
    tmp = [Vector{eltype(dist)}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L
        X = rand!(dist, tmp[Threads.threadid()])
        X̄[i] = mean(X)
        S²[i] = var(X)
    end
    scatter(X̄, S²; ms=1.5, msw=0, alpha, label="")
    scatter!([μ], [σ^2]; msc=:auto, label="(μ, σ²)")
    x = range(extrema(X̄)..., 200)
    y = range(extrema(S²)..., 200)
    t = range(0, 2π, 1000)
    plot!(f.(t), g.(t); label="", levels=[h], c=2)
    plot!(; xlabel="x̄", ylabel="s²")
    plot!(; size, legend, kwargs...)
end

function plot_X̄_and_SX²_2x2(dist;
        m = 4, distxlim = :auto, distylim = :auto,
        ns = (10, 40, 160, 640), size=(700, 700), kwargs...)
    μ, σ², sk, ku = mean(dist), var(dist), myskewness(dist), mykurtosis(dist)
    println(dist)
    @show μ σ²
    println("skewness = ", sk)
    println("kurtosis = ", ku)
    flush(stdout)

    D = plot_dist(dist; m, xlim=distxlim, ylim=distylim, size=(300, 200))
    display(D)

    PP = []
    for n in ns
        P = plot_X̄_and_SX²(dist; n, sk, ku, kwargs..., title="n = $n")
        push!(PP, P)
    end
    plot(PP...; size, layout=(2,2),
        titlefontsize=12, guidefontsize=10, tickfontsize=8)
end
```

Out[78]:

```
plot_X̄_and_SX²_2x2 (generic function with 1 method)
```

7.1 正規分布の標本分布における X̄, S² の同時分布

In [79]:
```
plot_X̄_and_SX²_2x2(Normal(); xlim=(-1.1, 1.1), ylim=(-0.1, 3.0))
```

```
Normal{Float64}(μ=0.0, σ=1.0)
μ = 0.0
σ² = 1.0
skewness = 0.0
kurtosis = 0.0
```

Out[79]:

7.2 一様分布の標本分布における X̄, S² の同時分布

In [80]:
```
plot_X̄_and_SX²_2x2(Uniform(); xlim=(0.2, 0.8), ylim=(0.01, 0.18))
```

```
Uniform{Float64}(a=0.0, b=1.0)
μ = 0.5
σ² = 0.08333333333333333
skewness = 0.0
kurtosis = -1.2
```

Out[80]:

7.3 単峰型のガンマ分布の標本分布における X̄, S² の同時分布

In [81]:
```
plot_X̄_and_SX²_2x2(Gamma(3, 4); xlim=(5, 20), ylim=(0, 200))
```

```
Gamma{Float64}(α=3.0, θ=4.0)
μ = 12.0
σ² = 48.0
skewness = 1.1547005383792517
kurtosis = 2.0
```

Out[81]:

7.4 指数分布の標本分布における X̄, S² の同時分布

In [82]:
```
plot_X̄_and_SX²_2x2(Exponential(); xlim=(0.2, 2.3), ylim=(-0.2, 7))
```

```
Exponential{Float64}(θ=1.0)
μ = 1.0
σ² = 1.0
skewness = 2.0
kurtosis = 6.0
```

Out[82]:

7.5 非対称なベータ分布の標本分布における X̄, S² の同時分布

In [83]:

```
plot_X̄_and_SX²_2x2(Beta(0.1, 0.9); xlim=(-0.02, 0.4), ylim=(-0.05, 0.2),
    distxlim=(-0.05, 1.05), distylim=(-0.5, 10))
```

```
Beta{Float64}(α=0.1, β=0.9)
μ = 0.1
σ² = 0.045000000000000005
skewness = 2.514157444218836
kurtosis = 5.611111111111112
```

Out[83]:

7.6 非対称な2つ山の混合正規分布の標本分布における $\bar{X}$, S^2 の同時分布

In [84]:
```
plot_X̄_and_SX²_2x2(MixtureModel([Normal(), Normal(20)], [0.95, 0.05]);
    xlim=(-1, 8), ylim=(-2, 120))
```

```
MixtureModel{Normal{Float64}}(K = 2)
components[1] (prior = 0.9500): Normal{Float64}(μ=0.0, σ=1.0)
components[2] (prior = 0.0500): Normal{Float64}(μ=20.0, σ=1.0)

μ = 1.0
σ² = 20.0
skewness = 3.823676241524649
kurtosis = 13.58499999999097
```

Out[84]:

7.7 対称なBernoulli分布の標本分布における X̄, S² の同時分布

In [85]:
```
plot_X̄_and_SX²_2x2(Bernoulli(0.5); xlim=(-0.02, 1.02), ylim=(-0.01, 0.30))
```

```
Bernoulli{Float64}(p=0.5)
μ = 0.5
σ² = 0.25
skewness = 0.0
kurtosis = -2.0
```

Out[85]:

7.8 非対称なBernoulli分布の標本分布における X̄, S² の同時分布

In [86]:
```
plot_X̄_and_SX²_2x2(Bernoulli(0.1);
    xlim=(-0.02, 0.6), ylim=(-0.01, 0.3), distxlim=(-1, 2))
```

```
Bernoulli{Float64}(p=0.1)
μ = 0.1
σ² = 0.09000000000000001
skewness = 2.6666666666666665
kurtosis = 5.111111111111109
```

Out[86]:

7.9 Poisson分布の標本分布における X̄, S² の同時分布

In [87]:
```
plot_X̄_and_SX²_2x2(Poisson(0.5); xlim=(-0.02, 1.2), ylim=(-0.04, 2.0))
```

```
Poisson{Float64}(λ=0.5)
μ = 0.5
σ² = 0.5
skewness = 1.414213562373095
kurtosis = 2.0
```

Out[87]:

7.10 対数正規分布の標本分布における X̄, S² の同時分布

In [88]:
```
plot_X̄_and_SX²_2x2(LogNormal(); xlim=(-0.2, 5), ylim=(-2, 50))
```

```
LogNormal{Float64}(μ=0.0, σ=1.0)
μ = 1.6487212707001282
σ² = 4.670774270471604
skewness = 6.184877138632554
kurtosis = 110.9363921763115
```

Out[88]:

対数正規分布では $(\bar{X}, S^2)$ の分布の正規分布近似の精度をよくするためには相当に n を大きくする必要がある.

8 正規分布の標本分布の場合

この節では, $X_1, \ldots, X_n$ はそれぞれが正規分布 $\mathrm{Normal}(\mu, \sigma)$ に従う独立な確率変数達であると仮定する:

$$(X_1, \ldots, X_n) \sim \mathrm{Normal}(\mu, \sigma)^n.$$

さらに, それらの標本分散と不偏分散を

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

と書く. このとき, 正規分布の標本分布の特殊性によって, $\bar{X}$ と S^2 は独立になり, $\bar{X}$ は平均 μ, 分散 σ^2/n の正規分布に従い,

$$Y = \frac{(n-1)S^2}{\sigma^2} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

は自由度 $\nu = n-1$ のχ²分布に従う:

$$\left(\bar{X}, \frac{(n-1)S^2}{\sigma^2}\right) \sim \mathrm{Normal}\bigl(\mu, \sigma/\sqrt{n}\bigr) \times \mathrm{Chisq}(n-1).$$

以下ではこの結果を証明する.

8.1 準備: √n x̄ を含む正規直交座標系

$\mathbb{R}^n$ を縦ベクトルの空間とみなし, その標準座標系を $x_1, \ldots, x_n$ と書く. このとき, $\mathbb{R}^n$ の任意の正規直交基底 $v_1, \ldots, v_n$ について

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \sum_{i=1}^{n} y_i v_i$$

で定義される $\mathbb{R}^n$ の座標系 $y_1, \ldots, y_n$ をその直交基底が定める $\mathbb{R}^n$ の **正規直交座標系** と呼ぶ. このとき, ベクトル $a, b \in \mathbb{R}^n$ の標準内積を (a, b) と書く, それに対応するノルムを $\|a\| = \sqrt{(a,a)}$ と書くと,

$$y_i = (v_i, x), \quad \|x\|^2 = \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i^2$$

などが成立している.

以下では $v_n \in \mathbb{R}^n$ は

$$v_n = \frac{1}{\sqrt{n}}\begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

と定められていると仮定し, $\bar{x}$ を次のように定める:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

このとき, $\|v_n\|^2 = (1^2 + \cdots + 1^2)/n = 1$ なので v_n は長さ 1 の単位ベクトルになり,

$$(v_n, x) = \frac{x_1 + \cdots + x_n}{\sqrt{n}} = \sqrt{n}\,\bar{x}.$$

さらに, v_n を含む $\mathbb{R}^n$ の任意の正規直交基底 $v_1, \ldots, v_{n-1}, v_n$ を取り, それが定める正規直交座標系を $y_1, \ldots, y_{n-1}, y_n$ と書くことにする. このとき, $y_n = (v_n, x) = \sqrt{n}\,\bar{x}$ なので, $y_1, \ldots, y_{n-1}, y_n$ は $y_n = \sqrt{n}\,\bar{x}$ の正規直交座標系への拡張になっている.

例: v_n を含む正規直交基底が任意に取られているせいで議論が抽象的過ぎるように感じる人達は, 以下の $v_1, \ldots, v_{n-1}$ 上の v_n を合わせたものがそのような正規直交基底の具体例になっていることを確認して, それらを使用すればよい:

$$v_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_2 = \frac{1}{\sqrt{6}}\begin{bmatrix} 1 \\ 1 \\ -2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots, \quad v_{n-1} = \frac{1}{\sqrt{n(n-1)}}\begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \\ -(n-1) \end{bmatrix}.$$

これらが内積について $(v_i, v_j) = \delta_{ij}$ を満たしていること(正規直交基底であること)は, この定義式を眺めて暗算すればすぐに確かめられる. **例終**

ベクトル v が

$$v = \begin{bmatrix} x_1 - \mu \\ \vdots \\ x_n - \mu \end{bmatrix} = x - \sqrt{n}\,\mu v_n$$

と定義されているとき,

$$(v_i, v) = (v_i, x) = y_i \quad (i = 1, \dots, n-1), \quad (v_n, v) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (x_i - \mu) = \sqrt{n}(\bar{x} - \mu)$$

なので,

$$\|v\|^2 = \sum_{i=1}^{n} (x_i - \mu)^2 = \sum_{i=1}^{n-1} y_i^2 + n(\bar{x} - \mu)^2.$$

さらに不偏分散関係でよく出て来る公式

$$\sum_{i=1}^{n} (x_i - \bar{x})^2 = \sum_{i=1}^{n} ((x_i - \mu) - (\bar{x} - \mu))^2 = \sum_{i=1}^{n} (x_i - \mu)^2 - n(\bar{x} - \mu)^2$$

より,

$$\sum_{i=1}^{n} (x_i - \mu)^2 = \sum_{i=1}^{n} (x_i - \bar{x})^2 + n(\bar{x} - \mu)^2, \quad \sum_{i=1}^{n} (x_i - \bar{x})^2 = \sum_{i=1}^{n-1} y_i^2$$

であることもわかる. これらの公式は後で正規分布の確率密度函数達の積に適用される.

8.2 準備: n - 2 次元単位球面上の微小面積要素

前節の設定をそのまま引き継ぐ.

$\mathbb{R}^n$ における $\mathbb{R}v_n$ の直交補空間上の正規直交座標系として $y_1, \dots, y_{n-1}$ を採用できる. その直交補空間内での $n-2$ 単位球面 $S^{n-2} : y_1^2 + \cdots + y_{n-1}^2 = 1$ 上の積分で使用可能な座標系 $\theta = (\theta_1, \dots, \theta_{n-2})$ を任意に取って, その座標系での球面上の微小面積要素を $d\theta$ と略して書くことにする. このとき, $n-2$ 次元単位球面 S^{n-2} の面積 A_{n-2} は

$$A_{n-2} = \int_{S^{n-2}} d\theta = \frac{2\pi^{(n-1)/2}}{\Gamma((n-1)/2)}$$

と表される.

例: 具体的に座標系を取っていないせいで議論が抽象的過ぎると感じる人は, 以下の座標系を利用すればよい:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} \cos\theta_1 \cos\theta_2 \cos\theta_3 \cdots \cos\theta_{n-2} \\ \sin\theta_1 \cos\theta_2 \cos\theta_3 \cdots \cos\theta_{n-2} \\ \sin\theta_2 \cos\theta_3 \cdots \cos\theta_{n-2} \\ \sin\theta_3 \cdots \cos\theta_{n-2} \\ \ddots \quad \vdots \\ \sin\theta_{n-2} \end{bmatrix} \qquad \begin{pmatrix} -\pi \le \theta_1 \le \pi, \\ -\frac{\pi}{2} \le \theta_2, \dots, \theta_{n-2} \le \frac{\pi}{2} \end{pmatrix}$$

この座標系は「Gauss積分, ガンマ函数, ベータ函数」のノートの「$n-1$ 次元単位球面上の座標を使った面積の計算」の節で紹介したものと本質的に同じものである. この座標系で微小面積要素は次のように表される:

$$d\theta = \cos^0\theta_1 \cos^1\theta_2 \cos^2\theta_3 \cdots \cos^{n-3}\theta_{n-2}\, d\theta_1\, d\theta_2\, d\theta_3 \cdots d\theta_{n-2}.$$

しかし, この例で紹介した複雑な式は後で使用されない. **例終**

このとき, さらに,

$$r = \sqrt{y_1^2 + \cdots + y_{n-1}^2}$$

とおくと, $\mathbb{R}^n$ における積分のための座標として $(\bar{x}, r, \theta)$ を取れて, その微小体積要素 dV は

$$dV = dy_n\, r^{n-2}\, dr\, d\theta = \sqrt{n}\, r^{n-2}\, d\bar{x}\, dr\, d\theta$$

と書ける. さらに, 不偏分散 S^2 に対応する変数 $u = s^2$ を

$$u = s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}\sum_{i=1}^{n-1} y_i^2$$

と定めると, $r = \sqrt{(n-1)u}$, $dr = \frac{\sqrt{n-1}}{2}u^{-1/2}\,du$ なので, これを上の式に代入して,

$$dV = \frac{1}{2}\sqrt{n}\,(n-1)^{(n-1)/2}\,u^{(n-1)/2-1}\,d\bar{x}\,du\,d\theta.$$

これの右辺の定数因子の形は重要ではない.

8.3 正規分布の標本分布における X̄, S² の同時分布の計算

前節までの設定をそのまま引き継ぐ.

このとき, $(X_1, \ldots, X_n) \sim \mathrm{Normal}(\mu, \sigma)^n$ の同時密度函数は以上で容易した座標系で

$$\sum_{i=1}^{n}(x_i - \mu)^2 = \sum_{i=1}^{n}(x_i - \bar{x})^2 + n(\bar{x} - \mu)^2 = (n-1)u + n(\bar{x} - \mu)^2$$

を代入すれば

$$\mathrm{const.}\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2\right) = \mathrm{const.}\exp\left(-\frac{(n-1)u}{2\sigma^2} - \frac{n(\bar{x} - \mu)^2}{2\sigma^2}\right)$$

と表される. 確率変数 S^2 に対応する積分変数を u と書くことにしたのであった. ゆえに,

$$\begin{aligned}
&E[f(\bar{X}, S^2)] \\
&= \mathrm{const.}\int_{S^{n-2}}\left(\int_0^\infty\left(\int_{\mathbb{R}} f(\bar{x}, u)\exp\left(-\frac{(n-1)u}{2\sigma^2} - \frac{n(\bar{x} - \mu)^2}{2\sigma^2}\right)u^{(n-1)/2-1}\,d\bar{x}\right)du\right)d\theta \\
&= \mathrm{const.}\int_0^\infty\left(\int_{\mathbb{R}} f(\bar{x}, u)\exp\left(-\frac{n(\bar{x} - \mu)^2}{2\sigma^2}\right)\exp\left(-\frac{(n-1)u}{2\sigma^2}\right)u^{(n-1)/2-1}\,d\bar{x}\right)du.
\end{aligned}$$

const. の部分は $f = 1$ の場合にこれが 1 になることから決定される. この式から S^2 に対応する通常の変数を $u > 0$ と書くことにすれば, $\bar{X}, S^2$ の同時確率密度函数は次の形になることがわかる:

$$\mathrm{const.}\exp\left(-\frac{n(\bar{x} - \mu)^2}{2\sigma^2}\right)\exp\left(-\frac{(n-1)u}{2\sigma^2}\right)u^{(n-1)/2-1}.$$

これは $\bar{X}$ と S^2 が独立であり,

$$\begin{aligned}
&\bar{X} \sim \mathrm{Normal}\left(\mu, \sqrt{\frac{\sigma^2}{n}}\right), \\
&S^2 \sim \mathrm{Gamma}\left(\frac{n-1}{2}, \frac{2\sigma^2}{n-1}\right) = \frac{\sigma^2\,\mathrm{Chisq}(n-1)}{n-1}
\end{aligned}$$

となっていることを意味している. これらは次と同値である:

$$\frac{\bar{X} - \mu}{\sqrt{\sigma^2/n}} \sim \mathrm{Normal}(0, 1), \quad \frac{(n-1)S^2}{\sigma^2} \sim \mathrm{Chisq}(n-1).$$

注意: 以上の議論では定数因子 const. の部分は重要でないので, その具体的な式を書かずに最後まで結論を出した. しかし, その部分について数学的な興味を持っている読者もいると思われるので, 以下では定数因子部分を正確に計算してみよう.

まず, 結論から, $\bar{X}, S^2$ それぞれの確率密度函数は以下の形になることがわかる(S^2 に対応する通常の変数を u と書く):

$$\begin{aligned}
&p(\bar{x}|n, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2/n}}\exp\left(-\frac{n(\bar{x} - \mu)^2}{2\sigma^2}\right), \\
&p(u = s^2|n, \mu, \sigma) = \frac{1}{(2\sigma^2/(n-1))^{(n-1)/2}\Gamma((n-1)/2)}\exp\left(-\frac{(n-1)u}{2\sigma^2}\right)u^{(n-1)/2-1}.
\end{aligned}$$

定数因子部分を略さずに上の計算を書くと以下のようになる:

$$
\begin{aligned}
&E[f(\bar{X}, S^2)] \\
&= \frac{1}{(2\pi\sigma^2)^{n/2}} \frac{1}{2} \sqrt{n}\, (n-1)^{(n-1)/2} \\
&\times \int_{S^{n-2}} \left(\int_0^\infty \left(\int_{\mathbb{R}} f(\bar{x}, u) \exp\left(-\frac{(n-1)u}{2\sigma^2} - \frac{n(\bar{x}-\mu)^2}{2\sigma^2} \right) u^{(n-1)/2-1}\, d\bar{x} \right) du \right) d\theta \\
&= \frac{1}{(2\pi\sigma^2)^{n/2}} \frac{1}{2} \sqrt{n}\, (n-1)^{(n-1)/2} \frac{2\pi^{(n-1)/2}}{\Gamma((n-1)/2)} \\
&\times \mathrm{const.} \int_0^\infty \left(\int_{\mathbb{R}} f(\bar{x}, u) \exp\left(-\frac{n(\bar{x}-\mu)^2}{2\sigma^2} \right) \exp\left(-\frac{(n-1)u}{2\sigma^2} \right) u^{(n-1)/2-1}\, d\bar{x} \right) du.
\end{aligned}
$$

そして,

$$
\begin{aligned}
&\frac{1}{(2\pi\sigma^2)^{n/2}} \frac{1}{2} \sqrt{n}\, (n-1)^{(n-1)/2} \frac{2\pi^{(n-1)/2}}{\Gamma((n-1)/2)} \\
&= \frac{1}{\sqrt{2\pi\sigma^2}\, (2\pi\sigma^2)^{(n-1)/2}} \sqrt{n}\, (n-1)^{(n-1)/2} \frac{\pi^{(n-1)/2}}{\Gamma((n-1)/2)} \\
&= \frac{1}{\sqrt{2\pi\sigma^2/n}} \times \frac{1}{(2\sigma^2/(n-1))^{(n-1)/2} \Gamma((n-1)/2)}
\end{aligned}
$$

なので, つじつまが合っている.

8.4 正規分布の標本分布から自然にt分布に従う確率変数が得られること

一般に, Z, Y が独立な確率変数達で, Z は標準正規分布に従い, Y が自由度 ν のχ²分布に従うとき, $T = Z/\sqrt{Y/\nu}$ は自由度 ν の t 分布に従うのであった.

この結果を標本平均 $\bar{X}$ の標準化

$$
Z = \frac{\bar{X} - \mu}{\sqrt{\sigma^2/n}} \sim \mathrm{Normal}(0, 1)
$$

とこれと独立に自由度 $\nu = n-1$ のχ²分布に従う

$$
Y = \frac{(n-1)S^2}{\sigma^2} \sim \mathrm{Chisq}(n-1)
$$

に適用すると, $Y/(n-1) = S^2/\sigma^2$ なので,

$$
T = \frac{Z}{\sqrt{Y/(n-1)}} = \frac{\bar{X} - \mu}{\sqrt{\sigma^2/n}} \sqrt{\frac{\sigma^2}{S^2}} = \frac{\bar{X} - \mu}{\sqrt{S^2/n}}
$$

が自由度 $n-1$ の t 分布に従うことがわかる:

$$
T = \frac{\bar{X} - \mu}{\sqrt{S^2/n}} \sim \mathrm{TDist}(\nu).
$$

注意: この T は Z が含む分散パラメータ σ^2 を標本の不偏分散 S^2 に置き換えた形をしており, Z と違って T は σ^2 を使わずに書けている. T は μ のみを含み σ^2 を含まないので, 分散 σ^2 の設定によらない μ に関する結果を得たいときに, この T は使える. 実際, この T は, 正規分布モデルによって, 平均 μ に関する区間推定や検定を σ^2 の存在に煩わされずに行うために使用される.

8.5 正規分布モデルの使用に関する重要な注意: t 分布の使いどころ

以上では, 正規分布のサイズ n の標本分布においては, 標本平均の標準化の式に含まれる分散のパラメータ σ^2 を不偏分散 S^2 で置き換えると, 自由度 $n-1$ の t 分布に従う確率変数 T が得られることを長くて精密な計算によって示すことができた.

しかし, 次の疑問は重要である:

> このようにがんばって t 分布に従う統計量 T を正規分布から作っても, 現実世界のデータがぴったり正規分布に従っていることはありそうに思えない. 現実世界におけるデータは近似的にも全然正規分布に従っていない場合が多いだろう. t 分布は現実世界における統計分析に役に立つのだろうか?

この疑問への答えを得るためには中心極限定理について学ぶ必要がある. (グラフのプロットの形ではすでに中心極限定理の例をすでに我々は目で確認しているが, 中心極限定理についは後で詳しく説明する.)

以上で構成した t 分布の自由度は $n-1$ だったので, 標本サイズ n が大きな場合にはその t 分布は標準正規分布で近似される. 実は正規分布の標本分布に限らず, 適当な緩い条件を満たす任意の確率分布の標本分布について, 中心極限定理より, T と同じ式で定義された統計量が標準正規分布で近似されることを示せる. そのような場合には t 分布を使うことに実践的に大して意味がないが, t 分布を使用しても害はない(正規分布以外の標本分布で使っても害がない).

一方, 左右が対称でおとなしめの確率分布の標本分布については, かなり小さな n であっても中心極限定理がよく効いてくれて, 標本平均 $\bar{X}$ の標準化 Z の分布の標準正規分布による近似の誤差が小さくなる. しかし, 実践的な状況では真の分散の値は不明なので, それを不偏分散で置き換える近似が必要になる. その近似は式の上では以上で構成した T を使うのと同じことになる.

その場合の T はもとの分布が正規分布ではないので t 分布には従わない. しかし, 標準正規分布でよく近似された分布の分散を適当に確率的に揺らがせることによって得られた分布に従うことになる. だから, その分布をどんぶり勘定で t 分布で近似してやれば統計分析の安全性を高めることができるだろうと考えられる. t 分布を使用することにはこういう意味がある.

注意・警告: 以上で説明したように, データが正規分布に従っていなくても, 中心極限定理による標本平均の分布の正規分布近似がうまく行っているという条件が成立していれば, t 分布を使った統計分析は実践的に意味を持ち得る. しかし, 極めて多くの解説で「正規分布の仮定が成立していない場合には t 分布を使った統計分析法を使ってはいけない」と解説されている. それは誤解である. 他の解説を読むときには注意して欲しい.

8.6 必修問題: データの t 値の計算例

次のサイズ $n=10$ のデータを $x_1,\ldots,x_n$ と書く:

- データ: 172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7

このデータと $\mu=170.0$ について以下を計算せよ:

(1) データの標本平均 $\bar{x}$

(2) データの不偏分散 s^2

(3) データの t 値 $t=(\bar{x}-\mu)/\sqrt{s^2/n}$

(4) 自由度 $n-1$ の t 分布に従う確率変数 T に関する $\text{p-value}=2(1-P(T\geq|t|))$. (この値をP値と呼ぶ.)

P値が小数点以下第3桁まで合っていれば正解とする.

解答例:

$$\bar{x}\approx 173.43,\quad s^2\approx 19.5779,\quad t\approx 2.45138,\quad \text{p-value}\approx 0.037.$$

解答終

8.6.1 WolframAlphaによる t 値の計算例

(1) mean 172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7 (https://www.wolframalpha.com/input?i=mean+172.5%2C+174.0%2C+181.0%2C+175.0%2C+180.7%2C+169.3%2C+170.6%2C+172.9%2C+169.6%2C+168.7) → 173.43

(2) var 172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7 (https://www.wolframalpha.com/input?i=var+172.5%2C+174.0%2C+181.0%2C+175.0%2C+180.7%2C+169.3%2C+170.6%2C+172.9%2C+169.6%2C+168.7) → 19.5779

(3) T = (173.43 - 170) / sqrt(19.5779/10) (https://www.wolframalpha.com/input?i=%28173.43+-+170%29+%2F+sqrt%2819.5779%2F10%29&lang=ja) → 2.45138

(4) P値 = 2(1 - cdf(StudentTDistribution(9), 2.45138)) (https://www.wolframalpha.com/input?i=2%281+-+cdf%28StudentTDistribution%289%29%2C+2.45138%29%29)

データの簡易要約: 172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7 (https://www.wolframalpha.com/input?i=172.5%2C+174.0%2C+181.0%2C+175.0%2C+180.7%2C+169.3%2C+170.6%2C+172.9%2C+169.6%2C+168.7)

8.6.2 Julia言語による t 値の計算例

In [89]:

```
x = [172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7]
n = length(x)
μ = 170.0
@show x̄ = mean(x)
@show s² = var(x)
@show t = (x̄ - μ)/√(s²/n)
@show pvalue = 2(1 - cdf(TDist(n-1), abs(t)))

ecdf_x = ecdf(x)
P1 = plot(; legend=:topleft)
scatter!(x, fill(-0.1, n), label="data", c=1)
plot!(x -> ecdf_x(x), 165, 185; label="ecdf of data", c=1)
vline!([x̄]; label="x̄", lw=1.5, c=2)

xlim = (-5, 5)
P2 = plot(; legend=:topleft)
plot!(TDist(n-1), xlim...; label="TDist($(n-1))")
vline!([t]; label="t-value", c=2)
vline!([-t]; label="", ls=:dash, c=2)
plot!(TDist(n-1), first(xlim), -abs(t);
    fill=0, c=1, fc=:red, fa=0.5, label="$(round(100pvalue; digits=2))%")
plot!(TDist(n-1), abs(t), last(xlim);
    fill=0, c=1, fc=:red, fa=0.5, label="")

plot(P1, P2; size=(800, 250))
```

```
x̄ = mean(x) = 173.43
s² = var(x) = 19.57788888888888
t = (x̄ - μ) / √(s² / n) = 2.4513831147919563
pvalue = 2 * (1 - cdf(TDist(n - 1), abs(t))) = 0.03667341466610541
```

Out[89]:

P値が約 3.67% であることの意味は上の右図を見ればわかる. 正規分布のサイズ $n = 10$ の標本分布における T 統計量は自由度 $n - 1 = 9$ の t 分布に従うのであった. 仮に与えられたデータの真の分布が正規分布に近ければ(現実の日本人の成人男性の身長の分布もかなり正規分布に近い形をしている), 計算した t 値がどれだけ外れた値になっているかは, 自由度 $n - 1 = 9$ の t 分布を使ってP値(=データから得られる数値以上に極端な値が統計モデル内で生成される確率)を計算すればわかる. このようなことは後で検定と信頼区間に関する別のノートで詳しく解説する予定である.

8.6.3 問題用の擬似データの生成の仕方

問題用の擬似データは Julia v1.7.2 による以下のコードで生成した:

Input:

```
@show VERSION
using Random
Random.seed!(4649373009)
X = round.(rand(Normal(170, 6), 10); digits=1)
@show X;
```

Output:

```
VERSION = v"1.7.2"
X = [172.5, 174.0, 181.0, 175.0, 180.7, 169.3, 170.6, 172.9, 169.6, 168.7]
```

これは日本人の成人男性の身長の擬似データのつもりである. 擬似乱数のシードを意図的にいじって偏ったデータが生成されるようにしてある. 平均 170, 標準偏差 6 の正規分布からであっても, 3.67% 程度の確率でこれ以上に偏ったデータが生成されることが上の計算で分かったことになる.

9 標本分布における T の分布の視覚化

以下に示すグラフ群を見れば, 正規分布以外の標本分布において T 統計量が従う分布がどれだけ自由度 $\nu = n - 1$ の t 分布からずれるかを確認できる. n を大きくすればずれが小さくなって行くことが確認できる.

$T = (\bar{X} - \mu)/\sqrt{S^2/n}$ だけではなく, $Z = (\bar{X} - \mu)/\sqrt{\sigma^2/n}$ (これの期待値と分散はそれぞれ 0 と 1 になる)と, S^2/σ^2 (これの期待値は 1 になる)の分布も横に並べてプロットした. n を大きくすると, T と Z の分布は正規分布の場合と同じ分布に一致して行くが, S^2/σ^2 の分布の場合には $n \to \infty$ で正規分布の場合もそうでない場合も 1 に台を持つデルタ分布に収束するが, 区間推定に利用できるな近似的な一致は成立していない. このように, 正規分布モデルは平均の区間推定については頑健だが, 分散の区間推定については全然そうではない.

In [90]:

```julia
function plot_X̄_S²_T(dist; n = 10, L = 10^6, kwargs...)
    μ, σ = mean(dist), std(dist)

    Z = Vector{Float64}(undef, L)
    Y = similar(Z)
    T = similar(Z)
    tmp = [Vector{eltype(dist)}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L
        X = rand!(dist, tmp[Threads.threadid()])
        X̄ = mean(X)
        S² = var(X)
        Z[i] = (X̄ - μ)/(σ/√n)
        Y[i] = S²/σ^2
        T[i] = (X̄ - μ)/√(S²/n)
    end

    normal = Normal()
    P1 = plot()
    if dist isa DiscreteUnivariateDistribution
        ecdf_Z = ecdf(Z)
        plot!(; legend=:topleft)
        plot!(x -> ecdf_Z(x), -4, 5; label="√n(X̄-μ)/σ")
        plot!(x -> cdf(normal, x); label="Normal()", ls=:dash)
        title!("cdf for n = $n")
    else
        stephist!(Z; norm=true, label="√n(X̄-μ)/σ")
        plot!(normal; label="Normal()", ls=:dash)
        plot!(; xlim=(-4, 5))
        title!("pdf for n = $n")
    end

    gamma = Chisq(n - 1)/(n - 1)
    xlim = (-0.2, 3.1)
    P2 = plot()
    if dist isa DiscreteUnivariateDistribution
        ecdf_Y = ecdf(Y)
        plot!(; legend=:bottomright)
        plot!(x -> ecdf_Y(x), xlim...; label="S²/σ²")
        plot!(x -> cdf(gamma, x); label="Chisq(ν)/ν", ls=:dash)
        title!("cdf for ν = $(n-1)")
    else
        stephist!(Y; norm=true, label="S²/σ²")
        plot!(gamma, max(0, 1-8/√(n-1)), 1+8/√(n-1); label="Chisq(ν)/ν", ls=:dash)
        plot!(; xlim)
        title!("pdf for ν = $(n-1)")
    end

    tdist = TDist(n - 1)
    P3 = plot()
    if dist isa DiscreteUnivariateDistribution
        ecdf_T = ecdf(T[.!isnan.(T)])
        plot!(; legend=:topleft)
        plot!(x -> ecdf_T(x), -4, 5; label="T")
        plot!(x -> cdf(tdist, x); label="TDist(ν)", ls=:dash)
        title!("cdf for ν = $(n-1)")
    else
        stephist!(T; norm=true, label="T")
        plot!(tdist; label="TDist(ν)", ls=:dash)
        plot!(; xlim=(-4, 5))
        title!("pdf for ν = $(n-1)")
    end

    plot(P1, P2, P3; size = (800, 180), layout=(1, 3))
    plot!(; legendfontsize=8)
end

function plot_X̄_S²_T_4x3(dist; ns = (10, 40, 160, 640),
        m = 4, distxlim = :auto, distylim = :auto,
        size = (800, 750), kwargs...)
    μ, σ², sk, ku = mean(dist), var(dist), myskewness(dist), mykurtosis(dist)
    println(dist)
    @show μ σ²
    println("skewness = ", sk)
    println("kurtosis = ", ku)
    flush(stdout)

    D = plot_dist(dist; m, xlim=distxlim, ylim=distylim, size=(300, 180))
```

```
    display(D)

    PP = []
    for n in ns
        P = plot_X̄_S²_T(dist; n, kwargs...)
        push!(PP, P)
    end
    plot(PP...; size, layout=(4,1),
        titlefontsize=10, guidefontsize=8, tickfontsize=6)
end
```

Out[90]: plot_X̄_S²_T_4x3 (generic function with 1 method)

9.1 正規分布の標本分布における T の分布

In [91]:
```
plot_X̄_S²_T_4x3(Normal(2, 3))
```

```
Normal{Float64}(μ=2.0, σ=3.0)
μ = 2.0
σ² = 9.0
skewness = 0.0
kurtosis = 0.0
```

Out[91]:

9.2 一様分布の標本分布における T の分布

In [92]:
```
plot_X̄_S²_T_4x3(Uniform())
```

```
Uniform{Float64}(a=0.0, b=1.0)
μ = 0.5
σ² = 0.08333333333333333
skewness = 0.0
kurtosis = -1.2
```

Out[92]:

9.3 単峰型のガンマ分布の標本分布における T の分布

In [93]:
```
plot_X̄_S²_T_4x3(Gamma(3, 4))
```

```
Gamma{Float64}(α=3.0, θ=4.0)
μ = 12.0
σ² = 48.0
skewness = 1.1547005383792517
kurtosis = 2.0
```

Out[93]:

9.4 指数分布の標本分布における T の分布

In [94]:
```
plot_X̄_S²_T_4x3(Exponential())
```

```
Exponential{Float64}(θ=1.0)
μ = 1.0
σ² = 1.0
skewness = 2.0
kurtosis = 6.0
```

Out[94]:

9.5 非対称なベータ分布の標本分布における T の分布

In [95]:

```
plot_X̄_S²_T_4x3(Beta(0.1, 0.9); distxlim=(-0.05, 1.05), distylim=(-0.5, 10))
```

```
Beta{Float64}(α=0.1, β=0.9)
μ = 0.1
σ² = 0.045000000000000005
skewness = 2.514157444218836
kurtosis = 5.611111111111112
```

Out[95]:

9.6 非対称な2つ山の混合正規分布の標本分布における T の分布

In [96]:
```
plot_X̄_S²_T_4x3(MixtureModel([Normal(), Normal(20)], [0.95, 0.05]))
```

```
MixtureModel{Normal{Float64}}(K = 2)
components[1] (prior = 0.9500): Normal{Float64}(μ=0.0, σ=1.0)
components[2] (prior = 0.0500): Normal{Float64}(μ=20.0, σ=1.0)

μ = 1.0
σ² = 20.0
skewness = 3.823676241524649
kurtosis = 13.58499999999097
```

Out[96]:

9.7 対称なBernoulli分布の標本分布における T の分布

In [97]:
```
plot_X̄_S²_T_4x3(Bernoulli(0.5))
```

```
Bernoulli{Float64}(p=0.5)
μ = 0.5
σ² = 0.25
skewness = 0.0
kurtosis = -2.0
```

Out[97]:

9.8 非対称なBernoulli分布の標本分布における T の分布

In [98]:
```
plot_X̄_S²_T_4x3(Bernoulli(0.1))
```

```
Bernoulli{Float64}(p=0.1)
μ = 0.1
σ² = 0.09000000000000001
skewness = 2.6666666666666665
kurtosis = 5.111111111111109
```

Out[98]:

9.9 Poisson分布の標本分布における T の分布

In [99]:
```
plot_X̄_S²_T_4x3(Poisson(0.5))
```

```
Poisson{Float64}(λ=0.5)
μ = 0.5
σ² = 0.5
skewness = 1.414213562373095
kurtosis = 2.0
```

Out[99]:

9.10 対数正規分布の標本分布における T の分布

In [100]:
```
plot_X̄_S²_T_4x3(LogNormal())
```

```
LogNormal{Float64}(μ=0.0, σ=1.0)
μ = 1.6487212707001282
σ² = 4.670774270471604
skewness = 6.184877138632554
kurtosis = 110.9363921763115
```

Out[100]:

In [ ]:
```
```

大数の法則と中心極限定理

- 黒木玄

- 2022-04-11〜2022-05-31

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 大数の法則
  - 1.1 Markovの不等式
  - 1.2 Chebyshevの不等式
  - 1.3 大数の弱法則
  - 1.4 不偏分散の $n \to \infty$ での挙動
  - 1.5 大数の強法則
    - 1.5.1 Borel-Cantelliの補題
    - 1.5.2 4次以下のモーメントの存在を仮定した場合の大数の強法則
  - 1.6 大数の強法則に関する文献
  - 1.7 問題: 大数の法則に関する誤解
  - 1.8 大数の法則のデモンストレーション
    - 1.8.1 Bernoulli試行の大数の法則
    - 1.8.2 正規分布のサンプルの大数の法則
    - 1.8.3 ガンマ分布のサンプルの大数の法則
    - 1.8.4 大数の法則が成立しない場合: Cauchy分布
    - 1.8.5 大数の法則が成立しない場合: Pólyaの壺
  - 1.9 $E[X_i] = 0$ のときの和 $X_1 + X_2 + \cdots + X_n$ の分布の様子 (中心極限定理に向けて)
- 2 二項分布の中心極限定理
  - 2.1 二項分布の中心極限定理の内容
  - 2.2 二項分布の中心極限定理のプロット
  - 2.3 必修重要問題: 二項分布の正規分布近似を使った確率の近似計算1
    - 2.3.1 WolframAlphaによる計算1
    - 2.3.2 Julia言語による計算1
    - 2.3.3 Julia言語による視覚化1
  - 2.4 必修重要問題: 二項分布の正規分布近似を使った確率の近似計算2
    - 2.4.1 WolframAlphaによる計算
    - 2.4.2 Julia言語による計算2
    - 2.4.3 Julia言語による視覚化2
  - 2.5 連続性補正とmid-P補正
  - 2.6 問題: 二項分布の中心極限定理を使った極限の計算問題
  - 2.7 二項分布の中心極限定理の証明の方針
  - 2.8 (1) Stirlingの公式を使った二項分布の確率質量函数の近似
  - 2.9 注意: Kullback-Leibler情報量とSanovの定理との関係
  - 2.10 (2) 二項分布の確率質量函数から正規分布の密度函数が出て来ること
- 3 中心極限定理
  - 3.1 中心極限定理のラフな説明
  - 3.2 中心極限定理の特性函数を使った証明
  - 3.3 中心極限定理の収束の速さと歪度
  - 3.4 中心極限定理のキュムラント母函数を使った証明
  - 3.5 中心極限定理の収束の速さと歪度と尖度
  - 3.6 中心極限定理のTaylorの定理のみを使う証明
    - 3.6.1 多重積分の書き方

3.6.2 積分剰余項型のTaylorの定理
3.6.3 積分剰余項の別の表示
3.6.4 微分剰余項型のTaylorの定理
3.6.5 Taylorの定理を使った中心極限定理の証明
3.6.6 中心極限定理の収束の速さと歪度と尖度(再)
3.7 問題: 中心極限定理の収束の様子のグラフ
4 デルタ法
4.1 1次のデルタ法と2次のデルタ法
4.2 おまけ: 2次のデルタ法使用時の近似分散と歪度と尖度に関する不等式
4.3 問題: 2次のデルタ法の不偏分散の平方根の期待値の近似への応用
4.4 問題: 1次のデルタ法の正規分布近似への応用
4.5 問題: 1次のデルタ法による分布の逆数の近似
4.6 例: 自由度が大きなχ²分布の逆数の近似
4.7 問題: デルタ法による分布の対数の近似
4.8 例: 自由度が大きなχ²分布の対数の近似
4.9 問題: ベータ分布の正規分布近似
4.10 問題: ベータ分布の正規分布近似の直接的証明
5 各種の確率分布の正規分布近似
5.1 二項分布の正規分布近似
5.2 負の二項分布の正規分布近似
5.3 Poisson分布の正規分布近似
5.4 ガンマ分布の正規分布近似
5.5 χ²分布の正規分布近似
5.6 逆ガンマ分布の正規分布近似
5.7 ベータ分布の正規分布近似
5.8 t 分布の正規分布近似
5.9 ベータプライム分布の正規分布近似
5.10 F 分布の正規分布近似
5.11 超幾何分布の正規分布近似
5.12 ベータ二項分布の正規分布近似
5.13 ベータ負の二項分布の正規分布近似

大数の法則と中心極限定理のイメージ

$\mu = E[X_i]$, $\sigma = \sqrt{E[(X_i-\mu)^2]}$ とおく。

$X_1, X_2, X_3, \ldots$ は独立同分布確率変数列であるとし、$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ とおく。

平均 μ, 標準偏差 $\frac{\sigma}{\sqrt{n}} \to 0$

サイズ n の標本の平均 $\overline{X}_n$ の分布の $n \to \infty$ での様子：

← $\overline{X}_n \approx \mu$ に集中する。 ← 大数の法則

$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ の分布は $n \to \infty$ で $\mu = E[X_i]$ に集中する。

$\overline{X}_n$ と μ の差を $\sqrt{n}$ 倍して拡大して見るとこうなる：$Y_n = \sqrt{n}(\overline{X}_n - \mu) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(X_i - \mu)$.

平均 0, 標準偏差 σ

← 中心極限定理

$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ と μ の差の大きさはだいたい $\frac{\sigma}{\sqrt{n}}$ の大きさであり、

差を $\sqrt{n}$ 倍拡大して分布を見ると標準偏差 σ の正規分布になっている。

さらに σ 分の1倍するとこうなる：$Z_n = \frac{\sqrt{n}}{\sigma}(\overline{X}_n - \mu) = \frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\frac{X_i - \mu}{\sigma}$

平均 0, 標準偏差 1

← 中心極限定理

以上のように、大数の法則によって1点に集中する標本平均の分布を

$\sqrt{n}$ 倍（もしくは $\sigma/\sqrt{n}$ 分の1倍）することによってより精密に見ると中心極限定理が得られる。

In [1]:
```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:
```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:
```julia
x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 4) - 3
```

Out[3]:
```
mykurtosis (generic function with 2 methods)
```

1 大数の法則

大数の法則 (law of large numbers, LLN)とは, 大雑把に言えば, 独立同分布確率変数列 $X_1, X_2, \ldots$ の最初の n 個の標本平均を $\bar{X}_n = (1/n)\sum_{i=1}^n X_i$ と書くと, $\bar{X}_n$ が X_i 達の共通の期待値に収束するという結果のことである. (より正確に言うと, その「収束」の仕方に強弱があって, それぞれについて「大数の強法則」と呼んだり, 「大数の弱法則」と呼んだりする. しかし, 応用上, 多くの場合にその違いに神経質になる必要はない.)

以下では大数の法則について解説する.

1.1 Markovの不等式

確率変数 X と任意の $a > 0$ について

$$P(|X| \geq a) \leq \frac{1}{a}E[|X|].$$

証明: 函数 $1_{|x|\geq a}(x)$ を

$$1_{|x|\geq a}(x) = \begin{cases} 1 & (|x| \geq a) \\ 0 & (|x| < a) \end{cases}$$

とおくと, $|x| \geq a$ と $1 \leq |x|/a$ は同値なので常に $1_{|x|\geq a}(x) \leq |x|/a$ となる. ゆえに期待値を取る操作の単調性と線形性より,

$$P(|X| \geq a) = E[1_{|x|\geq a}(x)] \leq E\left[\frac{|X|}{a}\right] = \frac{1}{a}E[|X|].$$

証明終

注意: Markovの不等式, Chebyshevの不等式, Jensenの不等式, Gibbsの情報不等式などについては「Bernoulli試行と関連確率分布」のノートですでに説明していたのであった.

1.2 Chebyshevの不等式

Markovの不等式で X を $(X-\mu)^2$ ($\mu = E[X]$) で置き換えた場合をChebyshevの不等式と呼ぶ.

確率変数 X について $E[|X|] < \infty$ が成立しているならば, X の期待値 $\mu = E[X]$ がうまく定義されている. そのとき, X の分散を $\sigma^2 = E[(X-\mu)^2]$ と書くと, 任意の $\varepsilon > 0$ について

$$P(|X-\mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2}.$$

証明: 前節のMarkovの不等式を X, a がそれぞれ $(X-\mu)^2$, ε^2 の場合に適用すると,

$$P(|X-\mu| \geq \varepsilon) = P((X-\mu)^2 \geq \varepsilon^2) \leq \frac{1}{\varepsilon^2}E[(X-\mu)^2] = \frac{\sigma^2}{\varepsilon^2}.$$

証明終

1.3 大数の弱法則

$X_1, X_2, X_3, \ldots$ は独立同分布確率変数の列であるとし, 共通の期待値 $\mu = E[X_i]$ が定義されており, 共通の分散 $\sigma^2 = E[(X_i-\mu)^2]$ は有限の値であると仮定する. このとき, 最初の n 個の標本平均を

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^n X_i$$

と書くと, 任意の $\varepsilon > 0$ について,

$$\lim_{n\to\infty} P(|\bar{X}_n - \mu| \geq \varepsilon) = 0.$$

注意: この結論は

$$\lim_{n\to\infty} P(|\bar{X}_n - \mu| < \varepsilon) = 1$$

と同値なので標本平均 $\bar{X}_n$ の分布が $n \to \infty$ で μ の近く(任意に小さな $\varepsilon > 0$ に関する μ から距離 ε 未満の範囲)に集中して行くことを意味している. このとき $\bar{X}_n$ は μ に **確率収束** するというが, 標本平均 $\bar{X}_n$ の分布が $n \to \infty$ で μ の近くに集中して行くことを理解していれば, 確率変数の収束に関する細かいことは応用上あまり神経質になる必要はない. だから, このノートでは「大数の弱法則」と「大数の強法則」の区別に神経質にならずに単に「大数の法則」ということが多い.

証明: 標本平均 $\bar{X}_n$ の期待値と分散は以下のようになるのであった:

$$\begin{aligned}
E[\bar{X}_n] &= \frac{1}{n}\sum_{i=1}^n E[X_i] = \frac{1}{n}\sum_{i=1}^n \mu = \mu, \\
(\bar{X}_n - \mu)^2 &= \left(\frac{1}{n}\sum_{i=1}^n (X_i - \mu)\right)^2 = \frac{1}{n^2}\sum_{i,j=1}^n (X_i-\mu)(X_j-\mu), \\
\mathrm{var}(\bar{X}_n) &= E[(\bar{X}_n-\mu)^2] = \frac{1}{n^2}\sum_{i,j=1}^n E[(X_i-\mu)(X_j-\mu)] = \frac{1}{n^2}\sum_{i,j=1}^n \delta_{ij}\sigma^2 = \frac{\sigma^2}{n}.
\end{aligned}$$

最後から2つめの等号で X_i 達が独立(特に無相関)であることを使った. ゆえに, Chebyshevの不等式を X, μ, σ^2 がそれぞれ $\bar{X}_n$, μ, $\mathrm{var}(\bar{X}_n) = \sigma^2/n$ の場合に適用すると, 任意の $\varepsilon > 0$ について,

$$P(|\bar{X}_n - \mu| \geq \varepsilon) \leq \frac{\sigma^2/n}{\varepsilon^2} \to 0 \quad \text{as } n \to \infty.$$

証明終

1.4 不偏分散の $n \to \infty$ での挙動

$X_1, X_2, X_3, \ldots$ は独立同分布確率変数の列であるとし, その最初の n 個の標本平均と不偏分散をそれぞれ次のように書くことにする:

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X}_n)^2.$$

以下では $\mu = E[X_i]$ がうまく定義されており, $\sigma^2 = E[(X_i-\mu)^2] < \infty$, $E[|X_i - \mu|^3] < \infty$, $E[(X_i-\mu)^4] < \infty$ となっていると仮定する.

「標本分布について」のノートで以下を示していたのであった. X_i 達が共通に従う分布の歪度(わいど, skewness)と尖度(せんど, kurtosis)をそれぞれ

$$\bar{\kappa}_3 = E\left[\left(\frac{X_i-\mu}{\sigma}\right)^3\right], \quad \bar{\kappa}_4 = E\left[\left(\frac{X_i-\mu}{\sigma}\right)^4\right] - 3$$

と書くと,

$$\begin{aligned}
&E[\bar{X}_n] = \mu, \quad E[S_n^2] = \sigma^2, \\
&\mathrm{var}(\bar{X}_n) = \frac{\sigma^2}{n}, \quad \mathrm{cov}(\bar{X}_n, S_n^2) = \sigma^3\frac{\bar{\kappa}_3}{n}, \quad \mathrm{var}(S_n^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right).
\end{aligned}$$

ゆえに, Chebyshevの不等式を X, μ, σ^2 がそれぞれ S_n^2, $E[S_n^2] = \sigma^2$, $\mathrm{var}(S_n^2) = \sigma^4(\bar{\kappa}_4/n + 2/(n-1))$ の場合に適用すると, 任意の $\varepsilon > 0$ について,

$$P(|S_n^2 - \sigma^2| \geq \varepsilon) \leq \frac{\sigma^4}{\varepsilon^2}\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right) \to 0 \quad \text{as } n \to \infty.$$

このように, 不偏分散 S_n^2 についても大数の弱法則が成立しており, S_n^2 の分布は $n \to \infty$ で σ^2 の近くに集中して行く.

1.5 大数の強法則

この節の内容は無理して読まなくてもよい. 定義していない用語や記号法が登場する. 大数の強法則の取り扱いには測度論的確率論が必要になるが, この一連のノート群では測度論を避ける方針なので, 大数の強法則について詳しく説明することはできない. 以下で紹介する議論は「よく見る議論」のコピー&ペーストに過ぎない. 必要な用語の定義もされていないし, 内容的にも雑である. しかし, 大数の強法則の証明の雰囲気を伝えたいと思ったので, このような説明をしてみることにした.

確率空間とは σ-algebra (= σ-field) $\mathscr{F}$ と確率測度 $\mathscr{F} \ni A \mapsto P(A) \in \mathbb{R}_{\geq 0}$ が与えられた集合 Ω のことである. 以下では確率空間 Ω 上の点を ω と表す. Ω 上の実数値可測函数を確率変数と呼ぶ.

1.5.1 Borel-Cantelliの補題

確率空間 Ω において次が成立する:

$$\sum_{k=1}^{\infty} P(A_k) < \infty \implies P\left(\bigcap_{n=1}^{\infty}\bigcup_{k\geq n} A_k\right) = 0.$$

証明: $\sum_{k=1}^{\infty} P(A_k) < \infty$ と仮定し, $B_n = \bigcup_{k\geq n} A_k$ とおく. $P\left(\bigcap_{n=1}^{\infty} B_n\right) = 0$ を示せばよい. $\sum_{k=1}^{\infty} P(A_k) < \infty$ より,

$$P(B_b) \leq \sum_{k\geq n} P(A_k) = \sum_{k=1}^{\infty} P(A_k) - \sum_{k=1}^{n-1} P(A_k) \to 0 \text{as } n \to \infty.$$

$B_1 \supset B_2 \supset B_3 \supset \cdots$ なので

$$P\left(\bigcap_{n=1}^{\infty} B_n\right) = \lim_{n\to\infty} P(B_n) = 0.$$

証明終

1.5.2 4次以下のモーメントの存在を仮定した場合の大数の強法則

$X_1, X_2, X_3, \ldots$ は独立同分布確率変数の列であるとし, その最初の n 個の標本平均を $\bar{X}_n = (1/n)\sum_{i=1}^n X_i$ と書くことにする. さらに, $\mu = E[X_i]$ がうまく定義されており, $\sigma^2 = E[(X_i-\mu)^2] < \infty$, $\mu_4 = E[(X_i-\mu)^4] < \infty$ となっていると仮定する. このとき,

$$P\left(\lim_{n\to\infty} \bar{X}_n = \mu\right) = 1.$$

注意: これは標本平均が μ に確率 1 で収束することを意味している.

証明: 大数の弱法則の場合と同様に補事象の側の同値な主張を示す. すなわち, 確率変数(可測函数)としての標本平均 $\bar{X}_n$ が $n \to \infty$ で μ に収束しない確率が 0 であることを示そう.

確率変数(可測函数)としての最初の n 個の標本平均 $\bar{X}_n$ の ω における値 $\bar{X}_n(\omega)$ が $n \to \infty$ で μ に収束しないという条件は以下のように書き換えられる. すなわち,

$$A_k(\varepsilon) = \{\, \omega \mid |\bar{X}_k(\omega) - \mu| \geq \varepsilon \,\}$$

とおくと, 以下が成立している:

$\bar{X}_n(\omega)$ は $n \to \infty$ で μ に収束しない.
$\iff$ ある $\varepsilon > 0$ が存在して, 任意の n について, ある $k \geq n$ で $|\bar{X}_k(\omega) - \mu| \geq \varepsilon$ を満たすものが存在する.
$\iff$ ある $\varepsilon > 0$ が存在して, $\omega \in \bigcap_{n=1}^{\infty} \bigcup_{k\geq n} A_k(\varepsilon)$.
$\iff$ $\omega \in \bigcup_{L=1}^{\infty} \bigcap_{n=1}^{\infty} \bigcup_{k\geq n} A_k(1/L)$.

この条件の書き換えによって, 任意の $\varepsilon > 0$ について,

$$\sum_{k=1}^{\infty} P(A_k(\varepsilon)) = \sum_{k=1}^{\infty} P(|\bar{X}_k - \mu| \geq \varepsilon) < \infty$$

を示せれば, Borel-Cantelliの補題より $P\left(\bigcap_{n=1}^{\infty} \bigcup_{k\geq n} A_k(\varepsilon)\right) = 0$ が得られ, それより

$$P\left(\bigcup_{L=1}^{\infty} \bigcap_{n=1}^{\infty} \bigcup_{k\geq n} A_k(1/L)\right) \leq \sum_{L=1}^{\infty} P\left(\bigcap_{n=1}^{\infty} \bigcup_{k\geq n} A_k(1/L)\right) = 0$$

が導かれ, $\bar{X}_n$ が $n \to \infty$ で μ に収束しない確率が 0 であることが示されることがわかる.

必要ならば X_i を $X_i - \mu$ で置き換えることによって, $\mu = 0$ と仮定しても一般性が失われないので, $\mu = 0$ と仮定する.

$\varepsilon > 0$ を任意に取る. このとき,

$$P(|\bar{X}_n| \geq \varepsilon) = P\left(\frac{\bar{X}_n^4}{\varepsilon^4} \geq 1\right) = E\left[1_{\bar{X}_n^4/\varepsilon^4 \geq 1}(\bar{X}_n)\right] \leq E\left[\frac{\bar{X}_n^4}{\varepsilon^4}\right] = \frac{1}{\varepsilon^4} E\left[\bar{X}_n^4\right],$$

でかつ

$$\begin{aligned} E\left[\bar{X}_n^4\right] &= \frac{1}{n^4} \sum_{i,j,k,l=1}^{n} E[X_i X_j X_k X_l] = \frac{1}{n^4}\left(nE[X_1^4] + \underbrace{\binom{4}{2}\binom{n}{2}}_{=3n(n-1)} E[X_1^2]^2\right) \\ &= \frac{\mu_4}{n^3} + \frac{3(n-1)\sigma^4}{n^3} \leq \frac{\mu_4 + 3\sigma^4}{n^2} \end{aligned}$$

なので,

$$\sum_{k=1}^{\infty} P(A_k) = \sum_{n=1}^{\infty} P(|\bar{X}_n| \geq \varepsilon) \leq \frac{\mu_4 + 3\sigma^4}{\varepsilon^4} \sum_{n=1}^{\infty} \frac{1}{n^2} < \infty.$$

これで示すべきことが示された.

証明終

注意: 要するに, Borel-Cantelliの補題から, 任意の $\varepsilon > 0$ について,

$$\sum_{n=1}^{\infty} P(|\bar{X}_n - \mu| \geq \varepsilon) < \infty$$

を示せれば, 大数の強法則が証明される. 大数の弱法則の証明で示したことは

$$P(|\bar{X}_n - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2}\frac{1}{n}$$

に過ぎず, $\sum_{n=1} 1/n = \infty$ と調和級数の和は無限大になるので, 大数の弱法則の証明の議論だけで大数の強法則を示すことはできない.

注意: $\zeta(s) = \sum_{n=1}^{\infty} 1/n^s$ はRiemannのゼータ函数と呼ばれ, $s = 1$ では調和級数になって無限大に発散するが, $s > 1$ ならば有限の値に収束する. $\zeta(2) = \sum_{n=1}^{\infty} 1/n^2$ を求めよという問題は **Basel問題** (バーゼル問題)と呼ばれ, 18世紀の超大数学者のEulerが解いた. 答えは $\pi^2/6$ になる. その逆数の $6/\pi^2 = 0.6079271\cdots$ は2つの整数の互いに素になる(最大公約数が 1 になる)「確率」に等しい.

In [4]:
```
6/π^2
```

Out[4]: 0.6079271018540267

In [5]:
```
"""
エラトステネスの篩(sieve of Eratosthenes)によって,
以下の条件を満たす長さ `N` のビットベクトル `IsPrime` を作成する:

>`IsPrime = make_isprime(N)` のとき `IsPrime[p]` は `p` が素数なら真になり, 素数でないなら偽に
"""
function make_isprime(N)
    IsPrime = trues(N)
    # 1は素数ではない
    IsPrime[1] = false
    @inbounds for p in 2:isqrt(N)
        if IsPrime[p]
            for i in 2p:p:N
                # 素数の非自明な倍数は素数ではない
                IsPrime[i] = false
            end
        end
    end
    IsPrime
end

"""
N以下の2つの正の整数が互いに素になる確率を篩法で求める.
"""
function prob_mutually_prime(N)
    IsPrime = make_isprime(N)
    C = trues(N, N)
    @inbounds for p in 2:N
        if IsPrime[p]
            for i in p:p:N
                for j in i:p:N
                    # 共通の素因数 p を持つ i, j を除外する
                    C[i, j] = false
                    C[j, i] = false
                end
            end
        end
    end
    c = count(C)
    c/N^2
end
```

Out[5]: prob_mutually_prime

In [6]:
```
@doc make_isprime
```

Out[6]: エラトステネスの篩(sieve of Eratosthenes)によって, 以下の条件を満たす長さ \texttt{N} のビットベクトル \texttt{IsPrime} を作成する:

```
\begin{quote}
\texttt{IsPrime = make\_isprime(N)} のとき \texttt{IsPrime[p]} は \texttt{p} が素数なら真になり, 素数でないなら偽になる.

\end{quote}
```

In [7]:
```
IsPrime = make_isprime(100)
primes = findall(IsPrime)
@show primes;
```

```
primes = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 8
3, 89, 97]
```

In [8]:
```
# 1億以下の素数のリストの作成も一瞬で終わる.
@time make_isprime(10^8);
```

```
  0.555804 seconds (3 allocations: 11.921 MiB, 3.60% gc time)
```

In [9]:
```
@time prob_mutually_prime(10000)
```

```
  0.068892 seconds (5 allocations: 11.922 MiB)
```

Out[9]:
```
0.60794971
```

In [10]:
```
@time prob_mutually_prime(100000)
```

```
 18.603973 seconds (5 allocations: 1.164 GiB, 0.07% gc time)
```

Out[10]:
```
0.6079301507
```

10万以下の2つの正の整数が互いに素な確率を求めてみたが, 確かに理論値の $1/\zeta(2) = 0.6079271\cdots$ に近い値になっている.

1.6 大数の強法則に関する文献

大数の強法則は前節で採用した条件よりもずっと弱い条件のもとで成立している. そのようなことに興味がある人達は以下の文献を参照せよ.

- N. Etemadi, An elementary proof of the strong law of large numbers, Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete volume 55, pages 119–122 (1981) [Springer (https://link.springer.com/article/10.1007/BF01013465#citeas)] [pdf (https://people.math.umass.edu/~yaoli/ptrl.pdf)]
- N. Lizia, A simple proof of the strong law of large numbers with rates, Bulletin of the Australian Mathematical Society, Volume 97, Issue 3, June 2018, pp. 513 - 517. [doi (https://doi.org/10.1017/S0004972718000059)] [pdf (http://www.im.ufrj.br/nuno/SLLN.pdf)]

$X_1, X_2, X_3, \ldots$ が同分布な確率変数の列であるとし, 共通の期待値 $\mu = E[X_i]$ のwell-definedness ($E[|X_i|] < \infty$)を仮定する.

前者の文献では, $X_1, X_2, X_3, \ldots$ に関する(独立性よりもずっと弱い)対ごとの独立性の仮定のもとで大数の強法則を証明している(分散の有限性も仮定していない).

後者の文献では, $X_1, X_2, X_3, \ldots$ の独立性と分散の有限性($\sigma^2 = \mathrm{var}(X_i) < \infty$)のもとで, 収束の速さが分かるような大数の強法則の証明を与えている.

1.7 問題: 大数の法則に関する誤解

当たりとはずれが毎回独立ににどちらも 50% の確率で出るルーレットを回し続ける状況を考える. 当たりが出たら, 胴元から1万円もらい, はずれが出たら1万円支払うというルールになっているとする. この賭け事に参加しているある人がこう言った:

> はずれの方が沢山出続けてたくさん負けてしまった! でも, 確率論の大数の法則によって, これからは当たりの方が出易くなって, 負けは帳消しになるだろう.

この発言が間違っていることについて説明せよ.

解答例: 大数の法則は, はずれが沢山出た後に当たりの方が出易くなったり, 当たりが沢山出たあとにはずれが出易くなることによって成立するのではない.

例えば, いきなり 10 回続けてはずれが出てしまったとする.

その後の90回分では, 当たりとはずれが両方とも45回出たとする. そのとき, トータルでの当たりが出たの割合は $55/100 = 55\%$ で 50% に近付く.

はずれ10回後の990回分でも, 当たりとはずれがどちらも同じ回数の495回出たとする. そのとき, トータルでの当たりが出た割合は $505/1000 = 50.5\%$ でさらに 50% に近付く.

そのとき, 最初に10回外れを引いて10万円負けていたその人は, 大数の法則が予想する結果に近付いているにもかかわらず, 1000回の勝負で最初に負けた10万円を取り戻せていない.

大数の法則は単に公平なギャンブルを続けると勝ちの回数の割合と負けの回数の割合が 50% に近付くことを主張しているだけで, 沢山負けた後に勝ち易くなったり, その逆が起こることを何も主張していない.

解答終

他にも沢山の正しい解答例を作れると思います.

1.8 大数の法則のデモンストレーション

In [11]:

```
function plot_law_of_large_numbers(dist, N;
        randfunc = rand, μ = mean(dist), kwargs...)
    X = randfunc(dist, N)
    X̄ = cumsum(X) ./ (1:N)
    plot(1:N, X̄; label="")
    hline!([μ]; label="")
    plot!(; kwargs...)
end

function plot_law_of_large_numbers_6x3(dist, N;
        randfunc = rand, μ = mean(dist), seed = 4649363, title = "", kwargs...)
    Random.seed!(seed)
    PP = []
    for _ in 1:18
        P = plot_law_of_large_numbers(dist, N; randfunc, μ, kwargs...)
        push!(PP, P)
    end
    P0 = plot(; title,
        framestyle=nothing, showaxis=false, ticks=false, margin=0Plots.mm)
    layout = @layout [a{0.005h}; grid(6, 3)]
    plot(P0, PP...; size=(800, 800), layout,
        titlefontsize=12, guidefontsize=8, tickfontsize=6)
end
```

Out[11]: plot_law_of_large_numbers_6x3 (generic function with 1 method)

1.8.1 Bernoulli試行の大数の法則

In [12]:

```
bar(x → pdf(Bernoulli(0.3), x), 0:1; alpha=0.5, label="",
    title="Bernoulli(0.3)", xtick=0:1, ytick=0:0.1:1)
```

Out[12]:

In [13]:

```
plot_law_of_large_numbers_6x3(Bernoulli(0.3), 2^12;
    title="Law of large numbers for Bernoulli(0.3)", ylim=(0.1, 0.5))
```

Out[13]:

1.8.2 正規分布のサンプルの大数の法則

In [14]:
```
plot(Normal(2, 3); label="", title="Normal(2, 3)")
```

Out[14]:

In [15]:
```
plot_law_of_large_numbers_6x3(Normal(2, 3), 2^12;
    title="Law of large numbers for Normal(2, 3)", ylim=(1, 3))
```

Out[15]:

1.8.3 ガンマ分布のサンプルの大数の法則

In [16]:
```
plot(Gamma(2, 3); label="", title="Gamma(2, 3)")
```

Out[16]:

In [17]:
```
plot_law_of_large_numbers_6x3(Gamma(2, 3), 2^12;
    title="Law of large numbers for Gamma(2, 3)", ylim=(5, 7))
```

Out[17]:

1.8.4 大数の法則が成立しない場合: Cauchy分布

確率密度函数が

$$p(x) = \frac{1}{\pi}\frac{1}{1+x^2}$$

のCauchy分布 $\mathrm{Cauchy}()$ に従う確率変数 X については

$$E[|X|] = \frac{2}{\pi}\int_0^\infty \frac{x}{1+x^2}\,dx = \infty$$

なので, 期待値 $E[X]$ が定義されない. Cauchy分布の場合には大数の法則は成立しない.

In [18]:
```
plot(Cauchy(), -10, 10; label="", title="Cauchy()")
```

Out[18]:

In [19]:

```
plot_law_of_large_numbers_6x3(Cauchy(), 2^12; μ = 0.0,
    title="Law of large numbers for Cauchy() does not hold", ylim=(-10, 10))
```

Out[19]:

注意: $X_1, X_2, \ldots, X_n$ がCauchy分布のサイズ n の標本であるとき, その標本平均 $\bar{X}$ もCauchy分布に従うことを示せる. そのことはCauchy分布の特性函数が

$$\varphi(t) = E[e^{itX}] = e^{-|t|} \quad (X \sim \mathrm{Cauchy}())$$

となることを認めれば容易に確認できる(この公式のよくある証明では複素解析を使う). 実際, これを認めると,

$$E[e^{it\bar{X}}] = \phi\left(\frac{t}{n}\right)^n = (e^{-|t/n|})^n = e^{-|t|} = \varphi(t) = E[e^{itX}].$$

$\bar{X}$ と $X \sim \mathrm{Cauchy}()$ の特性函数が等しいので $\bar{X} \sim \mathrm{Cauchy}()$.

In [20]:
```
function plot_cauchy(n; dist=Cauchy(), L=10^6, xlim=(-10, 10), kwargs...)
    X̄ = Vector{Float64}(undef, L)
    tmp = [Vector{Float64}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L
        X = rand!(dist, tmp[Threads.threadid()])
        X̄[i] = mean(X)
    end
    a, b = quantile.(Ref(X̄), (0.001, 0.999))
    stephist(X̄; bin=a:0.1:b, norm=true, label="X̄")
    plot!(Cauchy(), xlim...; label="Cauchy()", ls=:dash)
    title!("Sample means of n = $n size samples of Cauchy()")
    plot!(; xlim, kwargs...)
end
```

Out[20]: plot_cauchy (generic function with 1 method)

In [21]:
```
plot_cauchy(10)
```

Out[21]:

1.8.5 大数の法則が成立しない場合: Pólyaの壺

$\alpha, \beta > 0$ であると仮定する. パラメータ (α, β) から 1 の出る確率が $\alpha/(\alpha+\beta)$ と, 0 が出る確率が $\beta/(\alpha+\beta)$ と決まっているとする. そして, 1 が出たら, パラメータが $(\alpha, \beta) \mapsto (\alpha+1, \beta)$ と更新され, 0 が出たら, パラメータが $(\alpha, \beta) \mapsto (\alpha, \beta+1)$ と更新されると仮定する. このようにして $1, 0$ の列 $X_1, X_2, X_3, \ldots$ がランダムに得られる試行を **Pólyaの壺** の試行と呼ぶことにする.

Pólyaの壺では 1 が出るとその次に 1 が出る確率は上昇し, 0 が出るとその次に 0 が出る確率は上昇する. **Pólyaの壺** の試行では独立性の仮定が満たされていない. 実際, **Pólyaの壺** の試行の n 回分の出目が $(x_1, \ldots, x_n) \in \{1, 0\}^n$ になる確率は, $x_1, \ldots, x_n$ の中に 1 がちょうど k 個含まれているとき,

$$P(x_1, \ldots, x_n | \alpha, \beta) = \binom{n}{k} \frac{\alpha(\alpha+1)\cdots(\alpha+k-1)\ \beta(\beta+1)\cdots(\beta+n-k-1)}{(\alpha+\beta)(\alpha+\beta+1)\cdots(\alpha+\beta+n-1)}$$

と書ける. (「確率分布達の解釈」のノートに詳しい説明がある.)

しかし, $n-1$ 回目までの出目に条件を付けないとき, n 番目に 1 が出る確率は n によらず $\alpha/(\alpha+\beta)$ のまま一定であることを示せる. すなわち, 各 X_n が従う確率分布は n によらずにどれも成功確率 $\alpha/(\alpha+\beta)$ のBernoulli分布になる. **Pólyaの壺** の試行では同分布の仮定は満たされている.

Pólyaの壺 の試行で得られる $1, 0$ の列 $X_1, X_2, X_3, \ldots$ については大数の法則は成立しない. その n 番目までの標本平均 $\bar{X}_n$ はベータ二項分布 $\mathrm{BetaBinomial}(n, \alpha, \beta)$ に従い, n が大きなときに $\bar{X}_n$ はベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に近似的に従う. そうなることは,

$$P(x_1, \ldots, x_n | \alpha, \beta) = \int_0^1 \prod_{i=1}^n \left(p^{x_i}(1-p)^{1-x_i}\right) \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha, \beta)}\, dp$$

なので, **Pólyaの壺** の試行が成功確率 p がベータ分布 $\mathrm{Beta}(\alpha, \beta)$ に従ってランダムに決まっているようなBernoulli試行であることから分かる.

In [22]:

```
# pがベータ分布に従ってランダムに決まっているベルヌーイ試行

plot_law_of_large_numbers_6x3(Beta(4, 6), 2^10;
    randfunc = (dist, n) → (p = rand(dist); rand(Bernoulli(p), n)),
    title="Pólya's urn trials for α=4, β=6", ylim=(-0.02, 1.02), ytick=0:0.2:1)
```

Out[22]:

In [23]:

```julia
# 富むものがさらに富み，貧しいものがさらに貧しくなるポリアの壺試行

function rand_polyaurn(beta::Beta, n)
    α, β = Distributions.params(beta)
    X = Vector{Int8}(undef, n)
    for i in 1:n
        x = rand(Bernoulli(α/(α+β)))
        X[i] = x
        if x == 1
            α += 1
        else
            β += 1
        end
    end
    X
end

plot_law_of_large_numbers_6x3(Beta(4, 6), 2^10;
    randfunc = rand_polyaurn, title="Pólya's urn trials for α=4, β=6",
    ylim=(-0.02, 1.02), ytick=0:0.2:1)
```

Out[23]:

In [24]:
```
function plot_polyaurn(n, α, β, bin=:auto; L=10^6, kwargs...)
    X̄ = Vector{Float64}(undef, L)
    tmp = [Vector{Int}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L
        p = rand(Beta(α, β))
        X = rand!(Bernoulli(p), tmp[Threads.threadid()])
        X̄[i] = mean(X)
    end
    stephist(X̄; norm=true, bin, label="X̄")
    plot!(Beta(α, β), 0, 1; label="Beta(α, β)", ls=:dash)
    title!("Pólya's urn for n=$n, α=$α, β=$β")
    plot!(; xtick=0:0.1:1, kwargs...)
end
```

Out[24]: plot_polyaurn (generic function with 2 methods)

In [25]:
```
plot(plot_polyaurn.((10, 30), 4, 6, (0:0.1:1, 0:0.1:1))...; size=(800, 250))
```

Out[25]:

In [26]:
```
plot(plot_polyaurn.((100, 300), 4, 6, (0:0.01:1, 0:0.01:1))...; size=(800, 250))
```

Out[26]:

1.9 $E[X_i] = 0$ のときの和 $X_1 + X_2 + \cdots + X_n$ の分布の様子 (中心極限定理に向けて)

共通の期待値 0 と分散 σ^2 を持つ独立同分布確率変数列 $X_1, X_2, X_3, \ldots$ を沢山生成して, $X_1 + X_2 + \cdots + X_n$ の動きを同時プロットしてみよう. $X_1 + X_2 + \cdots + X_n$ の動きを **ランダムウォーク** (乱歩)と呼ぶことにする.

ランダムウォークは大雑把に $\pm 2\sigma\sqrt{n}$ の範囲に分布する.

この事実を多くのグラフで視覚的に確認した人達は中心極限定理に限りなく近付いている.

$X_1 + X_2 + \cdots + X_n$ がちらばる速さは $\sqrt{n}$ に比例するので, $(X_1 + X_2 + \cdots + X_n)/\sqrt{n}$ は一定の範囲に分布すると考えられる. その一定の範囲内の分布が正規分布で近似されるというのが中心極限定理の内容である.

In [27]:

```
function plot_randomwalk!(dist, N;
        randfunc = rand, μ = mean(dist), c=1, lw=0.3, alpha=0.3, kwargs...)
    X = randfunc(dist - μ, N)
    X̄ = cumsum(X)
    plot!(1:N, X̄; label="", c, lw, alpha, kwargs...)
end

function plot_randomwalks(dist, N;
        randfunc = rand, μ = mean(dist), σ = std(dist),
        seed = 4649363,
        title = "Random walks for $(distname(dist)) - $μ",
        kwargs...)
    Random.seed!(seed)
    P = plot(; title, )
    for _ in 1:100
        P = plot_randomwalk!(dist, N; randfunc, μ, kwargs...)
    end
    x = range(0, N, 1000)
    y = @. 2σ * √x
    plot!(x,  y; c=2, label="±2σ√n")
    plot!(x, -y; c=2, label="")
    plot!(; xlabel="n", ylabel="X₁+X₂+⋯+Xₙ")
    plot!(; legend=:topleft, size=(500, 320))
end
```

Out[27]: plot_randomwalks (generic function with 1 method)

In [28]:

```
plot_randomwalks(Bernoulli(0.3), 2^12)
```

Out[28]:

In [29]:

```
plot_randomwalks(Normal(2, 3), 2^12)
```

Out[29]:

In [30]:
```
plot_randomwalks(Gamma(2, 3), 2^12)
```

Out[30]:

In [31]:
```
plot_randomwalks(Cauchy(), 2^10; μ=0, σ=10, ylim=(-1000, 1000), alpha=0.7)
```

Out[31]:

In [32]:
```
plot_randomwalks(Beta(4, 6), 2^10;
    μ = 0, σ = 1,
    randfunc = (dist, n) → (p = rand(dist); rand(Bernoulli(p) - mean(dist), n)),
    title="Random walks for Pólya's urn with α=4, β=6", alpha=0.7)
```

Out[32]:

2 二項分布の中心極限定理

2.1 二項分布の中心極限定理の内容

二項分布 $\mathrm{Binomial}(n,p)$ が正規分布 $\mathrm{Normal}\left(np, \sqrt{np(1-p)}\right)$ で近似されることを **二項分布の中心極限定理** と呼ぶことにしよう. (ただしその近似の精度をよくするためには np と $n(1-p)$ をある程度以上に大きくする必要がある. 「大きくする」と言っても粗い近似でよければ3から5程度でも十分な場合がある.)

二項分布の中心極限定理には以下のような様々な言い換えがある. $0 < p < 1$ と仮定し, n は十分に大きいと仮定する. そのとき,

(1) $K \sim \mathrm{Binomial}(n,p)$ ならば K の分布は平均 np, 分散 $np(1-p)$ の正規分布で近似される.

(2) $K \sim \mathrm{Binomial}(n,p)$ ならば $\dfrac{K}{n}$ の分布は平均 p, 分散 $p(1-p)/n$ の正規分布で近似される.

(3) $K \sim \mathrm{Binomial}(n,p)$ ならば $\dfrac{K-np}{\sqrt{n}}$ の分布は平均 0, 分散 $p(1-p)$ の正規分布で近似される.

(4) $K \sim \mathrm{Binomial}(n,p)$ ならば $\dfrac{K-np}{\sqrt{np(1-p)}}$ の分布は標準正規分布で近似される.

例えば, (1)の結果は, $K \sim \mathrm{Binomial}(n,p)$, $X \sim \mathrm{Normal}\left(np, \sqrt{np(1-p)}\right)$ のとき,

$$P(K \le a) \approx P(X \le a)$$

という近似を使えることを意味する. (2),(3),(4)についても同様のことが言える. これを使うと, 二項分布における確率を対応する正規分布の確率で置き換えて近似計算できる. 二項分布に限らず, 正規分布で近似できる確率分布であれば, どのような分布であっても確率の計算を正規分布という一種類の分布における確率で近似的に計算できる.

例えば, (4)の結果は, $K \sim \mathrm{Binomial}(n,p)$ のとき,

$$E\left[f\left(\frac{K-np}{\sqrt{np(1-p)}}\right)\right] = \sum_{k=0}^{n} f\left(\frac{k-np}{\sqrt{np(1-p)}}\right)\binom{n}{k}p^k(1-p)^{n-k}$$

が, $n \to \infty$ で, $Z \sim \mathrm{Normal}(0,1)$ に関する

$$E[f(Z)] = \int_{-\infty}^{\infty} f(z)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz.$$

に収束することと同じことだと考えてよい. (3)についても同様のことが言える.

2.2 二項分布の中心極限定理のプロット

以下は, 試行回数 n, 成功確率 p の二項分布(期待値と分散はそれぞれ np, $np(1-p)$ になる)の確率質量函数と平均 np, 分散 $np(1-p)$ を持つ正規分布の確率密度函数の同時プロットである. それらを見れば, 証明しなくても, 二項分布の中心極限定理が成立していることは明らかだろう. (証明する前にコンピュータでグラフを確認するべき!)

注意: 以下のグラフを見れば, 二項分布の正規分布近似の精度を高めるためには, p が小さいほど n を大きくする必要があることがわかる. (p が 1 に近い場合にも n を大きくする必要がある.)

In [33]:
```
function plot_binomial_clt(n, p, s = 1; c = 4.5)
    μ, σ = n*p, √(n*p*(1-p))
    xlim = (μ-c*σ, μ+c*σ)
    plot(x -> mypdf(Binomial(n, p), x), xlim...; label="Binomial(n,p)")
    plot!(Normal(μ, σ), xlim...; label="Normal(μ,σ)", lw=2)
    # plot!(x -> mypdf(Poisson(μ), x), xlim...; label="Poisson(μ)", ls=:dash)
    title!("n = $n, p = $p, μ=np, σ²=np(1-p)")
    plot!(; xtick=0:s:n)
end
```

Out[33]:
```
plot_binomial_clt (generic function with 2 methods)
```

In [34]:
```
plot(plot_binomial_clt(10, 0.3), plot_binomial_clt(10, 0.1); size=(800, 250))
```

Out[34]:

In [35]:
```
plot(plot_binomial_clt(40, 0.3, 2), plot_binomial_clt(40, 0.1); size=(800, 250))
```

Out[35]:

In [36]:
```
plot(plot_binomial_clt(160, 0.3, 5), plot_binomial_clt(160, 0.1, 2); size=(800, 250))
```

Out[36]:

2.3 必修重要問題: 二項分布の正規分布近似を使った確率の近似計算1

確率変数 K は二項分布 $\mathrm{Binomial}(100, 1/2)$ に従っていると仮定する.

(1) 確率 $P(K \leq 40)$ を小数点以下第5桁まで求めよ.

(2) 二項分布 $\mathrm{Binomial}(100, 1/2)$ と同じ期待値と分散を持つ正規分布に従う確率変数 X について確率 $P(X \leq 40)$ を小数点以下第5桁まで求めよ.

解答例:

(1) $K \sim \mathrm{Binomial}(100, 1/2)$ のとき,

$$P(K \leq 40) \approx 0.02844$$

(2) 一般に $\mathrm{Binomial}(n, p)$ の平均と分散はそれぞれ $\mu = np$, $\sigma^2 = np(1-p)$ なので, $n = 100$, $p = 1/2$ の場合は $\mu = 50$, $\sigma^2 = 25$, $\sigma = 5$ なので, $X \sim \mathrm{Normal}(50, 5)$ で

$$P(X \leq 40) \approx 0.02275$$

解答終

2.3.1 WolframAlphaによる計算1

(1) [cdf(BinomialDistribution(n, p), 40) where n=100, p=0.5 (https://www.wolframalpha.com/input?i=cdf%28BinomialDistribution%28n%2C+p%29%2C+40%29+where+n%3D100%2C+p%3D0.5)](https://www.wolframalpha.com/input?i=cdf%28BinomialDistribution%28n%2C+p%29%2C+40%29+where+n%3D100%2C+p%3D0.5)

(2) [cdf(NormalDistribution(50, 5), 40) (https://www.wolframalpha.com/input?i=cdf%28NormalDistribution%2850%2C+5%29%2C+40%29)](https://www.wolframalpha.com/input?i=cdf%28NormalDistribution%2850%2C+5%29%2C+40%29)

2.3.2 Julia言語による計算1

In [37]:
```
n, p = 100, 1/2
```

Out[37]:
```
(100, 0.5)
```

In [38]:
```
# (1)
p_B = cdf(Binomial(n, p), 40)
```

Out[38]:
```
0.02844396682049039
```

In [39]:
```
# (2)
q_N = cdf(Normal(n*p, √(n*p*(1-p))), 40)
```

Out[39]:
```
0.022750131948179205
```

2.3.3 Julia言語による視覚化1

In [40]:
```
n, p = 100, 1/2
```

Out[40]:
```
(100, 0.5)
```

In [41]:
```
# (1)
plot(x -> mypdf(Binomial(n, p), x), 30, 70; label="", c=1)
vline!([n*p]; label="np", c=1)
plot!(x -> mypdf(Binomial(n, p), x), 30, 40.5;
    label="", fillrange=0, c=1, alpha=0.5)
prob = cdf(Binomial(n, p), 40)
annotate!(35, 0.01, text(@sprintf("%0.5f", prob), 10, :blue))
vline!([40]; label="", c=:gray)
title!("Binomial(n = $n, p = $p)")
```

Out[41]:

In [42]:

```
# (2)
plot(x -> mypdf(Normal(n*p, √(n*p*(1-p))), x), 30, 70; label="", c=:blue, ls=:dash)
vline!([n*p]; label="np", ls=:dashdot, c=:blue)
plot!(x -> mypdf(Normal(n*p, √(n*p*(1-p))), x), 30, 40;
    label="", fillrange=0, c=:blue, alpha=0.5)
prob = cdf(Normal(n*p, √(n*p*(1-p))), 40)
annotate!(35, 0.01, text(@sprintf("%0.5f", prob), 10, :blue))
vline!([40]; label="", c=:gray)
title!("Normal(μ = $n×$p, σ = √$(n*p*(1-p)))")
```

Out[42]:

In [43]:

```
# (1) and (2)
plot(x -> mypdf(Binomial(n, p), x), 30, 70; label="", c=1)
vline!([n*p]; label="np", c=1)

plot!(x -> mypdf(Normal(n*p, √(n*p*(1-p))), x), 30, 70;
    label="", c=:blue, ls=:dash)
#vline!([n*p]; label="np", ls=:dashdot, c=:blue)

vline!([40]; label="", c=:gray)
title!("Binomial(n = $n, p = $p) and normal approx")
```

Out[43]:

2.4 必修重要問題: 二項分布の正規分布近似を使った確率の近似計算2

$0 < p < 1$ であると仮定し, K は二項分布 $\mathrm{Binomial}(100, p)$ に従う確率変数であるとする.

(1) $P(K \leq 40) = 0.025$ を満たす p の値 p_U を小数点以下第4桁まで求めよ.

(2) $P(K \geq 40) = 0.025$ を満たす p の値 p_L を小数点以下第4桁まで求めよ.

$\hat{p} = 40/100 = 0.4$ とおく. X は K と同じ平均 np と推定された分散 $n\hat{p}(1-\hat{p})$ を持つ正規分布に従う確率変数であるとする.

(3) $P(X \leq 40) = 0.025$ を満たす p の値 q_U を小数点以下第4桁まで求めよ.

(4) $P(X \geq 40) = 0.025$ を満たす p の値 q_L を小数点以下第4桁まで求めよ.

注意: これらによって計算される区間 $[p_L, p_U]$ と $[q_L, q_U]$ はデータ「100 回中 40 回当たり」というデータから得られる p の 95% 信頼区間と呼ばれる. 前者は **Clopper-Pearsonの信頼区間** と, 後者は **Waldの信頼区間** と呼ばれる.

解答例: $n = 100$, $k = 40$ とおく.

累積分布函数を cdf と書き, その逆函数の分位点函数を quantile と書くことにする: X が分布 D に従う確率変数であるとき,

$$\begin{aligned}
&\mathrm{cdf}(D, x) = P(X \le x), \\
&\mathrm{quantile}(D, a) = (\text{the value } x \text{ satisfying } P(X \le x) = a).
\end{aligned}$$

以上の記号法のもとで,

$$\begin{aligned}
F_K(p) &= P(K \le k) = \mathrm{cdf}(\mathrm{Binomial}(n, p), k) \\
&= 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p), \\
G_K(p) &= P(K \ge k) = 1 - \mathrm{cdf}(\mathrm{Binomial}(n, p), k-1) \\
&= \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p)
\end{aligned}$$

について,

$$\begin{aligned}
F_K^{-1}(\alpha/2) &= \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2), \\
G_K^{-1}(\alpha/2) &= \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2).
\end{aligned}$$

これらを $n = 100$, $k = 40$, $\alpha = 0.05$ に適用すると,

(1) $p_U \approx 0.5028$

(2) $p_L \approx 0.3033$

$X \sim \mathrm{Normal}(np, \sigma)$ のとき,

$$\begin{aligned}
F_X(p) &= P(X \le k) = \mathrm{cdf}(\mathrm{Normal}(np, \sigma), k) \\
&= \mathrm{cdf}(\mathrm{Normal}(0, 1), (k - np)/\sigma), \\
G_X(p) &= P(X \ge k) = 1 - \mathrm{cdf}(\mathrm{Normal}(np, \sigma), k) \\
&= 1 - \mathrm{cdf}(\mathrm{Normal}(0, 1), (k - np)/\sigma)
\end{aligned}$$

について

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0, 1), 1 - \alpha/2)$$

とおくと, $\mathrm{quantile}(\mathrm{Normal}(0, 1), \alpha/2) = -z_{\alpha/2}$ なので,

$$\begin{aligned}
F_X^{-1}(\alpha/2) &= \frac{k - \sigma\,\mathrm{quantile}(\mathrm{Normal}(0, 1), \alpha/2)}{n} = \frac{k + z_{\alpha/2}\sigma}{n} \\
G_X^{-1}(1-\alpha/2) &= \frac{k - \sigma\,\mathrm{quantile}(\mathrm{Normal}(0, 1), 1-\alpha/2)}{n} = \frac{k - z_{\alpha/2}\sigma}{n}
\end{aligned}$$

$n = 100$, $\hat{p} = 0.4$ のとき,

$$\hat{\sigma}^2 = n\hat{p}(1 - \hat{p}) = 24.$$

であり, 上の公式を $n = 100$, $k = 40$, $\sigma = \hat{\sigma}$ の場合に使うと,

(3) $q_U \approx 0.4960$

(4) $q_L \approx 0.3040$

解答終

2.4.1 WolframAlphaによる計算

(1) quantile(BetaDistribution(41, 60), 0.975) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%2841%2C+60%29%2C+0.975%29)

(1)検算 cdf(BinomialDistribution(100, 0.502791), 40) (https://www.wolframalpha.com/input?i=cdf%28BinomialDistribution%28100%2C+0.502791%29%2C+40%29)

(2) quantile(BetaDistribution(40, 61), 0.025) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%2840%2C+61%29%2C+0.025%29)

(2)検算 1 - cdf(BinomialDistribution(100, 0.303295), 39) (https://www.wolframalpha.com/input?i=1+-+cdf%28BinomialDistribution%28100%2C+0.303295%29%2C+39%29)

quantile(NormalDistribution(0, 1), 0.975) (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C+1%29%2C+0.975%29) → 1.95996

(3) (40 + 1.95996 * sqrt(24))/100 (https://www.wolframalpha.com/input?i=%2840+%2B+1.95996+*+sqrt%2824%29%29%2F100)

(3)検算 cdf(NormalDistribution(49.60180, sqrt(24)), 40) (https://www.wolframalpha.com/input?i=cdf%28NormalDistribution%2849.60180%2C+sqrt%2824%29%29%2C+40%29)

(4) (40 - 1.95996 * sqrt(24))/100 (https://www.wolframalpha.com/input?i=%2840+-+1.95996+*+sqrt%2824%29%29%2F100)

(4)検算 1 - cdf(NormalDistribution(30.39820, sqrt(24)), 40) (https://www.wolframalpha.com/input?i=1+-+cdf%28NormalDistribution%2830.39820%2C+sqrt%2824%29%29%2C+40%29)

2.4.2 Julia言語による計算2

記号法: `cdf(d, x)` は分布 `d` の累積分布函数で `ccdf(d, x) = 1 - cdf(d, x)`.

`cdf(Binomial(n, p), k) == ccdf(Beta(k+1, n-k), p)` の確認:

In [44]:
```
n, p = 8, 0.3
[
    [cdf(Binomial(n, p), k) for k in 0:n-1]'
    [ccdf(Beta(k+1, n-k), p) for k in 0:n-1]'
]
```

Out[44]:
```
2×8 Matrix{Float64}:
 0.057648  0.255298  0.551774  0.805896  0.942032  0.988708  0.99871  0.999934
 0.057648  0.255298  0.551774  0.805896  0.942032  0.988708  0.99871  0.999934
```

`ccdf(Binomial(n, p), k-1) == cdf(Beta(k, n-k+1), p)` の確認:

In [45]:
```
n, p = 8, 0.3
[
    [ccdf(Binomial(n, p), k-1) for k in 1:n]'
    [cdf(Beta(k, n-k+1), p) for k in 1:n]'
]
```

Out[45]:
```
2×8 Matrix{Float64}:
 0.942352  0.744702  0.448226  0.194104  0.0579676  0.0112922  0.00129033  6.561e-5
 0.942352  0.744702  0.448226  0.194104  0.0579676  0.0112922  0.00129033  6.561e-5
```

In [46]:
```
n, k = 100, 40
α = 0.05
```

Out[46]:
```
0.05
```

In [47]:
```
# (1)
p_U = quantile(Beta(k+1, n-k), 1 - α/2)
```

Out[47]:
```
0.5027908495776652
```

In [48]:
```
# (1)検算
cdf(Binomial(n, p_U), k)
```

Out[48]:
```
0.02499999999999997
```

In [49]:
```
# (2)
p_L = quantile(Beta(k, n-k+1), α/2)
```

Out[49]:
```
0.3032947687028774
```

In [50]:
```
# (2)検算
ccdf(Binomial(n, p_L), k-1)
```

Out[50]:
```
0.025000000000000005
```

In [51]:
```
p̂ = k/n
```

Out[51]:
```
0.4
```

In [52]:
```
σ̂² = n*p̂*(1-p̂)
```
Out[52]: 24.0

In [53]:
```
σ̂ = √σ̂²
```
Out[53]: 4.898979485566356

In [54]:
```
# z_{α/2}
z = quantile(Normal(0, 1), 1 - α/2)
```
Out[54]: 1.9599639845400576

In [55]:
```
# (3)
q_U = (k + z*σ̂)/n
```
Out[55]: 0.4960182335271064

In [56]:
```
# (3)検算
cdf(Normal(n*q_U, σ̂), k)
```
Out[56]: 0.024999999999999776

In [57]:
```
# (4)
q_L = (k - z*σ̂)/n
```
Out[57]: 0.3039817664728936

In [58]:
```
# (4)検算
1 - cdf(Normal(n*q_L, σ̂), k)
```
Out[58]: 0.0249999999999998

2.4.3 Julia言語による視覚化2

In [59]:
```
n, k, α = 100, 40, 0.05
p̂ = k/n
σ̂² = n*p̂*(1-p̂)
σ̂ = √σ̂²

p_U = quantile(Beta(k+1, n-k), 1 - α/2)
p_L = quantile(Beta(k, n-k+1), α/2)
z = quantile(Normal(0, 1), 1 - α/2)
q_U = (k + z*σ̂)/n
q_L = (k - z*σ̂)/n;
```

In [60]:

```
# (1)
plot(x -> mypdf(Binomial(n, p_U), x), 10, 70; label="", c=1)
vline!([n*p_U]; label="n p_U", c=1)
plot!(x -> mypdf(Binomial(n, p_U), x), 10, 40.5;
    label="", fillrange=0, c=1, alpha=0.5)
annotate!(35, 0.01, text("2.5%", 10, :blue))
vline!([40]; label="", c=:gray)
title!("Binomial(n = $n, p_U = $(@sprintf "%.4f" p_U))")
plot!(; size=(600, 250))
```

Out[60]:

In [61]:

```
# (2)
plot(x -> mypdf(Binomial(n, p_L), x), 10, 70; label="", c=2)
vline!([n*p_L]; label="n p_L", c=2)
plot!(x -> mypdf(Binomial(n, p_L), x), 39.5, 70;
    label="", fillrange=0, c=2, alpha=0.5)
annotate!(45, 0.01, text("2.5%", 10, :red))
vline!([40]; label="", c=:gray)
title!("Binomial(n = $n, p_L = $(@sprintf "%.4f" p_L))")
plot!(; size=(600, 250))
```

Out[61]:

In [62]:

```
# (1) and (2)
plot(x -> mypdf(Binomial(n, p_U), x), 10, 70; label="", c=1)
vline!([n*p_U]; label="n p_U", c=1)
plot!(x -> mypdf(Binomial(n, p_U), x), 10, 40.5;
    label="", fillrange=0, c=1, alpha=0.5)
annotate!(35, 0.01, text("2.5%", 10, :blue))

plot!(x -> mypdf(Binomial(n, p_L), x), 10, 70; label="", c=2)
vline!([n*p_L]; label="n p_L", c=2)
plot!(x -> mypdf(Binomial(n, p_L), x), 39.5, 70;
    label="", fillrange=0, c=2, alpha=0.5)
annotate!(45, 0.01, text("2.5%", 10, :red))

vline!([40]; label="", c=:gray)
title!("[p_L, p_U] = [$(@sprintf "%.4f" p_L), $(@sprintf "%.4f" p_U)]")
plot!(; size=(600, 250))
```

Out[62]:

In [63]:

```
# (3)
plot(x -> mypdf(Normal(n*q_U, σ̂), x), 10, 70; label="", c=:blue, ls=:dash)
vline!([n*q_U]; label="n q_U", ls=:dashdot, c=:blue)
plot!(x -> mypdf(Normal(n*q_U, σ̂), x), 10, 40;
    label="", fillrange=0, c=:blue, alpha=0.5)
annotate!(35, 0.01, text("2.5%", 10, :blue))
vline!([40]; label="", c=:gray)
title!("Normal(μ = $(n)×$(@sprintf "%.4f" q_L), σ = √($n·$p̂(1-$p̂))")
plot!(; size=(600, 250))
```

Out[63]:

In [64]:

```
# (4)
plot(x -> mypdf(Normal(n*q_L, p̂), x), 10, 70; label="", c=:red, ls=:dash)
vline!([n*q_L]; label="n q_L", ls=:dashdot, c=:red)
plot!(x -> mypdf(Normal(n*q_L, p̂), x), 40, 70;
    label="", fillrange=0, c=:red, alpha=0.5)
annotate!(45, 0.01, text("2.5%", 10, :red))
vline!([40]; label="", c=:gray)
title!("Normal(μ = $(n)×$(@sprintf "%.4f" q_L), σ = √($n⋅$p̂(1-$p̂))")
plot!(; size=(600, 250))
```

Out[64]:

In [65]:

```
# (3) and (4)
plot(x -> mypdf(Normal(n*q_U, p̂), x), 10, 70; label="", c=:blue, ls=:dash)
vline!([n*q_U]; label="n q_U", ls=:dashdot, c=:blue)
plot!(x -> mypdf(Normal(n*q_U, p̂), x), 10, 40;
    label="", fillrange=0, c=:blue, alpha=0.5)
annotate!(35, 0.01, text("2.5%", 10, :blue))

plot!(x -> mypdf(Normal(n*q_L, p̂), x), 10, 70; label="", c=:red, ls=:dash)
vline!([n*q_L]; label="n q_L", ls=:dashdot, c=:red)
plot!(x -> mypdf(Normal(n*q_L, p̂), x), 40, 70;
    label="", fillrange=0, c=:red, alpha=0.5)
annotate!(45, 0.01, text("2.5%", 10, :red))

vline!([40]; label="", c=:gray)
title!("[q_L, q_U] = [$(@sprintf "%.4f" q_L), $(@sprintf "%.4f" q_U)]")
plot!(; size=(600, 250))
```

Out[65]:

In [66]:

```julia
# (1)(2)(3)(4)
plot(x -> mypdf(Binomial(n, p_U), x), 10, 70; label="", c=1)
vline!([n*p_U]; label="n p_U", c=1)

plot!(x -> mypdf(Binomial(n, p_L), x), 10, 70; label="", c=2)
vline!([n*p_L]; label="n p_L", c=2)

plot!(x -> mypdf(Normal(n*q_U, ô), x), 10, 70; label="", c=:blue, ls=:dash)
vline!([n*q_U]; label="n q_U", ls=:dashdot, c=:blue)

plot!(x -> mypdf(Normal(n*q_L, ô), x), 10, 70; label="", c=:red, ls=:dash)
vline!([n*q_L]; label="n q_L", ls=:dashdot, c=:red)

vline!([40]; label="", c=:gray)
title!("[p_L, p_U] = [$(@sprintf "%.4f" p_L), $(@sprintf "%.4f" p_U)]"*
    ",  [q_L, q_U] = [$(@sprintf "%.4f" q_L), $(@sprintf "%.4f" q_U)]")
plot!(; size=(600, 250))
```

Out[66]:

次節で使う視覚化のための函数

In [67]:

```julia
function plot_bin_and_normal_cdfs(;
        n=20, p = 0.4, k = 5, k_normal = k, midp = false,
        xlim = (-0.5, 16), annpos = (2, 0.05), midpannpos = (2, 0.07),
        xtick = 0:20, kwargs...
    )
    P1 = plot(x -> mypdf(Binomial(n, p), x), xlim...; label="", c=1)
    if midp
        plot!(x -> mypdf(Binomial(n, p), x), first(xlim), k;
                label="", fillrange=0, c=1, alpha=0.5)
        prob1 = (cdf(Binomial(n, p), k-1) + cdf(Binomial(n, p), k))/2
        annotate!(midpannpos..., text("mid-P value", 9, :blue))
    else
        plot!(x -> mypdf(Binomial(n, p), x), first(xlim), k+0.5;
                label="", fillrange=0, c=1, alpha=0.5)
        prob1 = cdf(Binomial(n, p), k)
    end
    annotate!(annpos..., text(@sprintf("%0.5f", prob1), 10, :blue))
    vline!([k]; label="x = $k", c=:gray)
    title!("Binomial(n = $n, p = $p)")
    plot!(; xtick)

    P2 = plot(x -> mypdf(Normal(n*p, √(n*p*(1-p))), x), xlim...;
        label="", c=:blue, ls=:dash)
    plot!(x -> mypdf(Normal(n*p, √(n*p*(1-p))), x), first(xlim), k_normal;
        label="", fillrange=0, c=:blue, alpha=0.5)
    prob2 = cdf(Normal(n*p, √(n*p*(1-p))), k_normal)
    annotate!(annpos..., text(@sprintf("%0.5f", prob2), 10, :blue))
    vline!([k_normal]; label="x = $k_normal", c=:gray)
    title!("Normal(μ = $(n*p), σ = √$(n*p*(1-p)))")
    plot!(; xtick)

    plot(P1, P2; size=(800, 250))
end
```

Out[67]: plot_bin_and_normal_cdfs (generic function with 1 method)

In [68]:

```
pval_exact(k, n, p0) = cdf(Binomial(n, p0), k)
pval_normal(k, n, p0) = cdf(Normal(n*p0, √(n*p0*(1-p0))), k)
pval_conti(k, n, p0) = cdf(Normal(n*p0, √(n*p0*(1-p0))), k+0.5)
pval_midp(k, n, p0) = (cdf(Binomial(n, p0), k-1) + cdf(Binomial(n, p0), k))/2

function prob_alphaerror(pvalfunc, n, p0, α)
    null = Binomial(n, p0)
    sum(pdf(null, k) for k in support(null)
            if pvalfunc(k, n, p0) ≲ α; init = 0.0)
end

function _plot_alphaerrors(n, p0, α, tick)
    p_exact  = prob_alphaerror.(pval_exact, n, p0, α)
    p_normal = prob_alphaerror.(pval_normal, n, p0, α)
    p_conti  = prob_alphaerror.(pval_conti, n, p0, α)
    p_midp   = prob_alphaerror.(pval_midp, n, p0, α)
    plot(; legend=:topleft)
    plot!(α, p_exact; label="exact")
    plot!(α, p_normal; label="normal", ls=:dash)
    plot!(α, p_conti; label="conti. corr.", ls=:dot, lw=1.5)
    plot!(α, p_midp; label="mid-P", ls=:dashdot)
    plot!(α, α; label="", c=:black, ls=:dot)
    xtick = ytick = tick
    plot!(; xtick, ytick, xrotation=90)
    a, b = extrema(α)
    xlim = ylim = (a-0.02(b-a), b+0.02(b-a))
    plot!(; xlim, ylim)
    title!("n = $n, p₀ = $p0")
end

function plot_alphaerrors(n, p0)
    α =
    P1 = _plot_alphaerrors(n, p0, 0:0.001:1, 0:0.1:1)
    P2 = _plot_alphaerrors(n, p0, 0:0.0001:0.1, 0:0.01:1)
    plot(P1, P2; size=(640, 320), layout=(1, 2))
end
```

Out[68]: plot_alphaerrors (generic function with 1 method)

2.5 連続性補正とmid-P補正

まずは次の図を見て欲しい.

In [69]:

```
plot_bin_and_normal_cdfs()
```

Out[69]:

この図の左側と右側は, それぞれ, $n = 20$, $p = 0.4$, $k = 5$ に関する

$$\mathrm{cdf}(\mathrm{Binomial}(n, p), k) \approx 0.12560,$$
$$\mathrm{cdf}(\mathrm{Normal}(np, \sqrt{np(1-p)}), k) \approx 0.08545,$$

の計算を視覚化したものである. ここで cdf は累積分布函数 (cumulative distribution function) を意味する.

後者は前者の正規分布近似による計算であるが, 前者の値 ≈ 0.12560 より小さな値 ≈ 0.08545 になってしまっている. この近似をより正確にするためにはどうすればよいだろうか?

図を見ればわかるように, 正規分布側で 5 以下になる確率ではなく, 5.5 以下になる確率を計算すれば良さそうである. 実際に計算してみると次のようになる.

In [70]:
```
plot_bin_and_normal_cdfs(; k_normal = 5.5)
```

Out[70]:

正規分布近似の側で 5.5 以下になる確率を計算するようにしたら, 二項分布の側で 5 以下である確率の値 ≈ 0.12560 にかなり近い値 ≈ 0.12692 が得られた.

正規分布による近似の側(より一般的には離散分布の連続分布による近似側)をこのように補正することを **連続性補正** (continuity correction)と呼ぶ.

逆に二項分布の側での値を 5 以下になる確率と 4 以下になる確率の平均値で求めるようにすると, 正規分布近似側の値に近付く. 離散分布の側でこのように補正することを **mid-P 補正** (mid-P correction)と呼ぶ. 次の図を見よ.

In [71]:
```
plot_bin_and_normal_cdfs(; midp = true)
```

Out[71]:

mid-P補正は近似である連続分布の側に近似されている離散分布側の値を近付ける補正になっている. 近似される側を近似する側に近付ける補正を使うことは無意味だと感じる人もいるかもしれないが必ずしもそうではないことを説明しよう.

二項分布 $\mathrm{Binomial}(n, p_0)$ において, k 以下になる確率を次のように書き, データ k に関する仮説 $p \geq p_0$ の片側検定の正確な **P 値** (P-value) と呼ぶことにする:

$$\mathrm{pval}_{\mathrm{exact}}(k|n, p_0) = \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k).$$

これの正規分布近似版とその連続補正版とmid-P補正版をそれぞれ次のように書くことにする:

$$\begin{aligned}
\mathrm{pval}_{\mathrm{normal}}(k|n, p_0) &= \mathrm{cdf}(\mathrm{Normal}(np_0, \sqrt{np_0(1-p_0)}), k), \\
\mathrm{pval}_{\mathrm{conti}}(k|n, p_0) &= \mathrm{cdf}(\mathrm{Normal}(np_0, \sqrt{np_0(1-p_0)}), k+0.5), \\
\mathrm{pval}_{\text{mid-P}}(k|n, p_0) &= \frac{1}{2}(\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k-1) + \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k)).
\end{aligned}$$

確率変数 K が $\mathrm{Binomial}(n, p_0)$ に従うとき, データを K としたときのこれらのP値の値が $0 < \alpha < 1$ について α 以下になる確率がどうなるかを計算してみよう. その確率は **第一種の過誤の確率** もしくは **アルファエラー** と呼ばれ, α に近いことが期待される値とみなされる(その理由は後で検定論を学べばわかる).

正確なP値については, 第一種の過誤の確率は確実に α 以下になるが, 大部分の場合に α より真に小さな値になり, 場合によってはかなり小さな値になる(この性質を **保守的** (conservative)と形容することがある). それの良い近似である連続補正された正規分布近似版も保守的である. それらに対して, 正規分布近似によるP値とmid-P補正によるP値の第一種の過誤の確率は α 前後の値になり, 平均的には正確なP値よりも α の値に近くなる.

第一種の過誤の確率を確実に α 以下にしたければ正確なP値が優れているが, 第一種の過誤の確率が α に近い方が良いことにすると正規分布近似によるP値とmid-P補正によるP値の方が優れていることになる.

このようにP値のユーザーが何を欲しているかによってどのP値の採用が適切であるかは変わる.

以下のグラフは横軸を α としたときのP値達の第一種の過誤の確率を同時プロットしたものである.

In [72]:
```
plot_alphaerrors(10, 0.6)
```

Out[72]:

In [73]:
```
plot_alphaerrors(20, 0.6)
```

Out[73]:

In [74]:
```
plot_alphaerrors(40, 0.6)
```

Out[74]:

2.6 問題: 二項分布の中心極限定理を使った極限の計算問題

$0 < p < 1$ と $m = 0, 1, 2, \ldots$ について以下の極限を求めよ:

$$\lim_{n\to\infty} \sum_{k=0}^{n} \left(\frac{k - np}{\sqrt{np(1-p)}} \right)^{2m} \binom{n}{k} p^k (1-p)^{n-k}.$$

解答例: 二項分布の中心極限定理より,

$$\begin{aligned}
&\lim_{n\to\infty} \sum_{k=0}^{n} \left(\frac{k - np}{\sqrt{np(1-p)}} \right)^{2m} \binom{n}{k} p^k (1-p)^{n-k} \\
&= \int_{-\infty}^{\infty} \left(\frac{x}{\sqrt{p(1-p)}} \right)^{2m} \frac{1}{\sqrt{2\pi p(1-p)}} \exp\left(-\frac{x^2}{2p(1-p)} \right) dx \\
&= \int_{-\infty}^{\infty} z^{2m} \frac{1}{\sqrt{2\pi}} e^{-z^2/2}\, dz = \frac{2}{\sqrt{2\pi}} \int_0^{\infty} e^{-z^2/2} z^{2m}\, dz \\
&= \frac{1}{\sqrt{2\pi}} \int_0^{\infty} e^{-y/2} y^{m+1/2-1}\, dy = \frac{1}{\sqrt{2\pi}} 2^{m+1/2} \Gamma(m+1/2) \\
&= \frac{1}{\sqrt{2\pi}} 2^{m+1/2} \frac{2m-1}{2} \cdots \frac{3}{2} \frac{1}{2} \Gamma(1/2) = 1 \cdot 3 \cdots (2m-1).
\end{aligned}$$

最初の等号で二項分布の中心極限定理を使い, 2番目の等号で $x = \sqrt{p(1-p)}\, z$ とおき, 4番目の等号で $z = \sqrt{y}$ とおいた. 5番目の等号では $\int_0^{\infty} e^{-y/\theta} y^{\alpha-1}\, dy = \theta^{\alpha}\Gamma(\alpha)$ を使った.

解答終

上の問題の結果を数値的に確認してみよう.

In [75]:

```
g(n, p, m, k) = ((k - n*p)/√(n*p*(1-p)))^(2m)
p = 0.4
@show p
for m in 0:5
    println()
    @show m
    @show n = 10^(m+1)
    @show prod(1:2:2m-1)
    @show sum(g(n, p, m, k)*pdf(Binomial(n, p), k) for k in 0:n)
end;
```

```
p = 0.4

m = 0
n = 10 ^ (m + 1) = 10
prod(1:2:2m - 1) = 1
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 1.000000000000001

m = 1
n = 10 ^ (m + 1) = 100
prod(1:2:2m - 1) = 1
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 1.0000000000000095

m = 2
n = 10 ^ (m + 1) = 1000
prod(1:2:2m - 1) = 3
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 2.998166666666703

m = 3
n = 10 ^ (m + 1) = 10000
prod(1:2:2m - 1) = 15
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 14.997416790272869

m = 4
n = 10 ^ (m + 1) = 100000
prod(1:2:2m - 1) = 105
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 104.99661670588064

m = 5
n = 10 ^ (m + 1) = 1000000
prod(1:2:2m - 1) = 945
sum((g(n, p, m, k) * pdf(Binomial(n, p), k) for k = 0:n)) = 944.9952750033706
```

2.7 二項分布の中心極限定理の証明の方針

二項分布の中心極限定理を以下の2段階で示す.

記号法に関する準備: $0 < p < 1$ は固定するが, n, k は動かす. k は n ごとに決まっていると仮定する. 混乱を完全に防ぐには k を k_n のように書いた方がよいが, 記号が煩雑になって書くのが面倒になるので, 単に k と書くことにする. $n \to \infty$ で 0 に収束する量を $o(1)$ と書く. より一般に a_n で割った結果が $n \to \infty$ で 0 に収束する量を $o(a_n)$ と書く. 例えば $o\left(1/\sqrt{n}\right)$ は $\sqrt{n}$ をかけた後であっても $n \to \infty$ で 0 に収束する量になる. この記号法のもとで, Stirlingの公式は次のように書ける:

$$n! = n^n e^{-n} \sqrt{2\pi n}\,(1 + o(1)).$$

(1) n ごとに決まっている k は $n \to \infty$ で

$$\frac{k}{n} = p + o(1)$$

という条件を満たしていると仮定する. このとき, 二項分布の確率質量函数

$$P(k|n,p) = \frac{n!}{k!(n-k)!} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

における二項係数の中の $n!$, $k!$, $(n-k)!$ に階乗に関するStirlingの公式による近似を適用すると次の近似式が得られる:

$$P(k|n,p) = \frac{1}{\sqrt{2\pi np(1-p)}} \left(\left(\frac{k/n}{p}\right) \left(\frac{1-k/n}{1-p}\right) \right)^{-n} (1 + o(1)).$$

途中の段階としてこの形に整理しておくと計算の見通しが良くなる. (しかもこの形には重要な意味がある. その点については後述する.)

(2) 固定された実数 x について, 上の(1)での近似式を示すために使った仮定 $k/n = p + o(1)$ よりも強い条件

$$\frac{k}{n} = p + \frac{x}{\sqrt{n}} + o(n^{-1/2})$$

を仮定すると, 次が得られる:

$$P(k|n,p) = \underbrace{\frac{1}{\sqrt{2\pi p(1-p)}} \exp\left(-\frac{x^2}{2p(1-p)}\right)}_{\text{pdf of normal distribution}} \underbrace{\frac{1}{\sqrt{n}}}_{dx/dk} (1 + o(1)).$$

これより,

$$K_n \sim \mathrm{Binomial}(n,p), \quad X_n = \frac{K_n - np}{\sqrt{n}}, \quad X_\infty \sim \mathrm{Normal}(0, p(1-p))$$

ならば

$$\lim_{n\to\infty} E[f(X_n)] \to E[f(X_\infty)]$$

となることがわかる.

この結果は, n が大きなとき, $K_n \sim \mathrm{Binomial}(n,p)$ に対する $X_n = (K_n - np)/\sqrt{n}$ が従う分布が $\mathrm{Normal}(0, p(1-p))$ で近似されることを意味している($p(1-p)$ は分布 $\mathrm{Bernoulli}(p)$ の分散に等しい). この結果を二項分布の中心極限定理と呼ぶ.

2.8 (1) Stirlingの公式を使った二項分布の確率質量函数の近似

n ごとに決まっている k は $n \to \infty$ で

$$\frac{k}{n} = p + o(1) \quad \left(\Longleftrightarrow\ 1 - \frac{k}{n} = 1 - p + o(1) \right)$$

という条件を満たしていると仮定する. $0 < p < 1$ と仮定していたので, $n \to \infty$ のとき $k \to \infty$ だけではなく, $n - k \to \infty$ も成立している. ゆえに $n, k, n-k$ について次のStirlingの公式が成立している:

$$\begin{aligned}
n! &= n^n e^{-n} \sqrt{2\pi n}\,(1 + o(1)), \\
k! &= k^k e^{-k} \sqrt{2\pi k}\,(1 + o(1)), \\
(n-k)! &= (n-k)^{n-k} e^{-(n-k)} \sqrt{2\pi(n-k)}\,(1 + o(1)).
\end{aligned}$$

これを代入すると

$$
\begin{aligned}
\frac{n!}{k!(n-k)!} &= \frac{n^n e^{-n}\sqrt{2\pi n}}{k^k e^{-k}\sqrt{2\pi k}\,(n-k)^{n-k}e^{-(n-k)}\sqrt{2\pi(n-k)}}(1+o(1)) \\
&= \frac{1}{(k/n)^k(1-k/n)^{n-k}\sqrt{2\pi n(k/n)(1-k/n)}}(1+o(1)) \\
&= \frac{1}{(k/n)^k(1-k/n)^{n-k}\sqrt{2\pi np(1-p)}}(1+o(1))
\end{aligned}
$$

2つめの等号で分子の e^{-n} と分母の $e^{-k}e^{-(n-k)}$ がキャンセルさせ, 分子分母を n^k と n^{n-k} と $\sqrt{2\pi n}$ で割った. 3つめの等号で $k/n = p + o(1)$, $1 - k/n = 1 - p + o(1)$ を使った. これを二項分布の確率質量函数

$$
P(k|n,p) = \frac{n!}{k!(n-k)!}p^k(1-p)^{n-k} \quad (k = 0, 1, \ldots, n)
$$

に適用すると,

$$
\begin{aligned}
P(k|n,p) &= \frac{1}{\sqrt{2\pi np(1-p)}}\left(\frac{p}{k/n}\right)^k\left(\frac{1-p}{1-k/n}\right)^{n-k}(1+o(1)) \\
&= \frac{1}{\sqrt{2\pi np(1-p)}}\left(\left(\frac{k/n}{p}\right)^{k/n}\left(\frac{1-k/n}{1-p}\right)^{1-k/n}\right)^{-n}(1+o(1)).
\end{aligned}
$$

2.9 注意: Kullback-Leibler情報量とSanovの定理との関係

前節でわざわざ最後の形に変形した理由は, 対数を取ると

$$
\log\left(\left(\left(\frac{k/n}{p}\right)^{k/n}\left(\frac{1-k/n}{1-p}\right)^{1-k/n}\right)^{-n}\right) = -n\left(\frac{k}{n}\log\frac{k/n}{p} + \left(1 - \frac{k}{n}\log\frac{1-k/n}{1-p}\right)\right)
$$

となることを分かり易くするためである. これを $-n$ で割った結果は

$$
D(Q||P) = \sum_{i=1}^{r} Q(i)\log\frac{Q(i)}{P(i)} \quad \left(P(i), Q(i) > 0,\ \sum_{i=1}^{r} P(i) = \sum_{i=1}^{r} Q(i) = 1\right)
$$

の特別な場合になっている($r = 2$, $P(1) = p$, $P(2) = 1 - p$, $Q(1) = k/n$, $Q(2) = 1 - k/n$). この $D(Q||P)$ は **Kullback-Leibler情報量** (KL情報量)と呼ばれ, そして, 上の計算結果は $r = 2$ の場合のKL情報量のSanovの証明にもなっている. Sanovの定理の易しい解説が以下の場所にある:

- Kullback-Leibler情報量とSanovの定理 (https://genkuroki.github.io/documents/20160616KullbackLeibler.pdf)

Sanovの定理の内容を要約すると, Kullback-Leibler情報量 $D(Q||P)$ は確率分布 P に従う乱数の繰り返し生成において確率分布 Q がどれだけ出て来難いかを表しているという主張になる. 上の二項分布の場合には, Q に対応する k/n が生じる確率が $\exp(-nD(Q||P) + O(\log n))$ の形をしていることが示されていることになるので(分母の $\sqrt{2\pi np(1-p)}$ は $\exp(O(\log n))$ になる), KL情報量 $D(Q||P)$ が大きいほど k/n が生じる確率は急激に小さくなる.

Sanovの定理は統計学でのモデル選択における情報量規準の考え方の基礎になっている. KL情報量は確率論では **大偏差原理** に出て来る. KL情報量は物理的には統計力学的な相対エントロピーの -1 倍になる. このような分野を超えた広がりについては, **赤池情報量規準** (**AIC**)で有名な赤池弘次氏による1980年の2つの論説も参照せよ:

- 赤池弘次, エントロピーとモデルの尤度(<講座>物理学周辺の確率統計), 日本物理学会誌, 1980年第35巻7号, pp. 608-614. link (https://www.jstage.jst.go.jp/article/butsuri1946/35/7/35_7_608/_article/-char/ja/)
- 赤池弘次, 統計的推論のパラダイムの変遷について, 統計数理研究所彙報, 1980年第27巻第1号, pp. 5-12. link (https://ismrepo.ism.ac.jp/index.php?active_action=repository_view_main_item_detail&page_id=13&block_id=21&item_id=32568&item_no=1)

赤池弘次さんは前者のp.612に「筆者によって導入されたこの統計量はAIC(an information criterionの略記*)と呼ばれ」と書いているが, それに対して会誌編集委員会が「一般には Akaike's information criterionの略と解されている」と脚注を付けている. このことを確認するだけでもダウンロードする価値があるだろう.

上で紹介した計算は二項分布の確率質量函数にStirlingの公式を適用する単純な計算に過ぎないのだが, その先には分野を超えた非常に面白い世界が広がっている.

2.10 (2) 二項分布の確率質量函数から正規分布の密度函数が出て来ること

n ごとに決まっている k は $n \to \infty$ で

$$\frac{k}{n} = p + o(1)$$

という条件を満たしているならば

$$P(k|n,p) = \frac{1}{\sqrt{2\pi np(1-p)}} \left(\left(\frac{k/n}{p} \right)^{k/n} \left(\frac{1-k/n}{1-p} \right)^{1-k/n} \right)^{-n} (1+o(1))$$

となることを上で示した. 以下では $k/n = p + o(1)$ よりも強い条件

$$\frac{k-np}{\sqrt{n}} = x + \varepsilon_n, \quad \lim_{n\to\infty} \varepsilon_n = 0$$

を仮定する. このとき,

$$\frac{k}{n} = p + \frac{x+\varepsilon_n}{\sqrt{n}}, \quad 1 - \frac{k}{n} = 1 - p - \frac{x+\varepsilon_n}{\sqrt{n}}.$$

以上の仮定のもとで, $\log(1+t) = t - t^2/2 + t^3/3 - \cdots$ を使うと,

$$\begin{aligned}
\log \left(\frac{k/n}{p} \right)^{k/n} &= \frac{k}{n} \log \frac{k/n}{p} = \left(p + \frac{x+\varepsilon_n}{\sqrt{n}} \right) \log \left(1 + \frac{x+\varepsilon_n}{\sqrt{n}\, p} \right) \\
&= \left(p + \frac{x+\varepsilon_n}{\sqrt{n}} \right) \left(\frac{x+\varepsilon_n}{\sqrt{n}\, p} - \frac{x^2}{2np^2} + o(1/n) \right) \\
&= \frac{x+\varepsilon_n}{\sqrt{n}} - \frac{x^2}{2np} + \frac{x^2}{np^2} + o(1/n) \\
&= \frac{x+\varepsilon_n}{\sqrt{n}} + \frac{x^2}{2np} + o(1/n), \\
\log \left(1 - \frac{k/n}{1-p} \right)^{1-k/n} &= \left(1 - \frac{k}{n} \right) \log \frac{k/n - 1}{p} = \left(1 - p - \frac{x+\varepsilon_n}{\sqrt{n}} \right) \log \left(1 - \frac{x+\varepsilon_n}{\sqrt{n}\,(1-p)} \right) \\
&= \left(1 - p - \frac{x+\varepsilon_n}{\sqrt{n}} \right) \left(-\frac{x+\varepsilon_n}{\sqrt{n}\,(1-p)} - \frac{x^2}{2n(1-p)^2} + o(1/n) \right) \\
&= -\frac{x+\varepsilon_n}{\sqrt{n}} - \frac{x^2}{2n(1-p)} + \frac{x^2}{n(1-p)^2} + o(1/n) \\
&= -\frac{x+\varepsilon_n}{\sqrt{n}} + \frac{x^2}{2n(1-p)} + o(1/n).
\end{aligned}$$

ゆえに,

$$\begin{aligned}
&\log \left(\left(\frac{k/n}{p} \right)^{k/n} \left(\frac{1-k/n}{1-p} \right)^{1-k/n} \right)^{-n} \\
&= -n \left(\log \left(\frac{k/n}{p} \right)^{k/n} + \log \left(1 - \frac{k/n}{1-p} \right)^{1-k/n} \right) \\
&= -\frac{x^2}{2p} - \frac{x^2}{2(1-p)} + o(1) = -\frac{x^2}{2p(1-p)} + o(1).
\end{aligned}$$

したがって,

$$P(k|n,p) = \frac{1}{\sqrt{2\pi p(1-p)}} \exp \left(-\frac{x^2}{2p(1-p)} \right) \frac{1}{\sqrt{n}} \, (1+o(1)).$$

この公式の $1/\sqrt{n}$ より前の部分は平均 0, 分散 $p(1-p)$ の正規分布の確率密度函数であり, $1/\sqrt{n}$ の因子は
$k = np + \sqrt{n}\,(x+\varepsilon_n)$ から形式的に得られる $dk = \sqrt{n}\,dx$ の $\sqrt{n}$ の因子とキャンセルすると考えると自然に見える.

これより, $K_n \sim \mathrm{Binomial}(n,p)$, X_\infty \sim \op{Normal}(0, p(1-p)) のとき,

$$X_n = \frac{K_n - np}{\sqrt{n}}, \quad x_n(k) = \frac{k-np}{\sqrt{n}}, \quad \Delta x_n = \frac{1}{\sqrt{n}}$$

とおくと, 有界な連続函数 $f(x)$ について, $n \to \infty$ のとき

$$
\begin{aligned}
E[f(X_n)] &= \sum_{k=0}^{n} f(x_n(k)) P(k|n,p) \\
&= \sum_{k=0}^{n} f(x_n(k)) \frac{1}{\sqrt{2\pi p(1-p)}} \exp\left(-\frac{x_n(k)^2}{2p(1-p)}\right) \Delta x_n \, (1+o(1)) \\
&= \int_{-\infty}^{\infty} f(x) \frac{1}{\sqrt{2\pi p(1-p)}} \exp\left(-\frac{x^2}{2p(1-p)}\right) dx = E[f(X_\infty)].
\end{aligned}
$$

以上の議論は厳密には少しギャップがあるのだが, 二項分布に関して中心極限定理が成立する理由の本質は十分に分かる内容になっている.

3 中心極限定理

中心極限定理 (central limit theorem)は「中心に収束する極限定理」というような意味ではなく、**「確率論における中心的な極限定理」** という意味である.

注意: 特性函数, モーメント母函数, キュムラント母函数, 歪度(わいど), 尖度(せんど)などについては「標本分布について」のノートに詳しい解説がある.

3.1 中心極限定理のラフな説明

$X_1, \ldots, X_n$ は各々が期待値 μ, 分散 σ^2 を持つ分布に従う n 個の独立同分布確率変数達であるとし, n は十分に大きいと仮定する.

中心極限定理は以下のように同値な言い方が色々ある:

(1) それらの和 $X_1 + \cdots + X_n$ が従う分布は期待値 $n\mu$, 分散 $n\sigma^2$ の正規分布で近似される:

$$
\sum_{i=1}^{n} X_i \sim \mathrm{Normal}\bigl(n\mu, \sqrt{n}\,\sigma\bigr) \quad \text{approximately.}
$$

(2) それらの加法平均 $\bar{X}_n = (X_1 + \cdots + X_n)/n$ が従う分布は期待値 μ, 分散 σ^2/n の正規分布で近似される:

$$
\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i \sim \mathrm{Normal}\bigl(\mu, \sigma/\sqrt{n}\bigr) \quad \text{approximately.}
$$

(3) $\sqrt{n}\,(\bar{X}_n - \mu)$ が従う分布は期待値 0, 分散 σ^2 の正規分布で近似される:

$$
\sqrt{n}\,(\bar{X}_n - \mu) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(X_i - \mu) \sim \mathrm{Normal}(0, \sigma) \quad \text{approximately.}
$$

(4) 次の Z_n が従う分布は標準正規分布で近似される:

$$
Z_n = \frac{\sqrt{n}\,(\bar{X}_n - \mu)}{\sigma} = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\frac{X_i - \mu}{\sigma} \sim \mathrm{Normal}(0, 1) \quad \text{approximately.}
$$

これらはどれも便利であり, 今後, 自由に使われることになるだろう.

注意・警告: 統計学に中心極限定理を応用する場合には, $n \to \infty$ とできる数学的に理想的な状況における中心極限定理ではなく, 有限の固定された n における近似として中心極限定理は使われる. だから, 場合ごとにその近似の誤差が問題になる. この問題は目的ごとに異なる許容される誤差との兼ね合いの問題になるので, きれいな一般論で解決できる話ではなく, 中心極限定理のユーザー自身が自分自身の目的に合わせて誤差が大きくなるリスクをどこまで許容するかを決定する必要がある. そのためには, 中心極限定理による近似の精度がどのような場合に悪くなりそうかについて前もって理解しておく必要がある.

補足: 分布 D_n が分布 D_∞ で近似されるとは, 分布 D_n に従う確率変数 X_n と分布 D_∞ に従う確率変数 X_∞ と適切なクラスに含まれる任意の函数 $f(x)$ について

$$
\lim_{n\to\infty} E[f(X_n)] = E[f(X_\infty)]
$$

が成立することだと, 大雑把に定義しておく. (函数 $f(x)$ としてどのような函数を許すかについて詳細な説明をするためにはさらなる数学的道具立てが必要でかる, 雑に扱っても害はほとんどない場合にもなっているので, このノートで説明しない. 所謂「分布収束」「法収束」の話になる.) 統計学で必要な確率論の計算は期待値を取る操作経由で可能なのでこのような定義にしておいても困ることはない.

補足の補足: 数学的詳細を理解できそうもないと感じる人であっても, 中心極限定理の良いユーザーになることは可能である. そのためには, 正規分布で近似したい分布のグラフとそれを近似すると期待される正規分布のグラフを重ねて描く作業を十分に沢山行えばよい. 数学的詳細を理解可能な人であっても, そういう視覚化による理解をサボってしまうと, 有限の n でどのように正規分布による近似の精度が悪くなるかについての感覚が身に付かなくなってしまう. 数学的詳細を理解していなくても, 中心極限定理の利用でどのようなリスクが発生するかを具体例を沢山見ることによって把握していれば, その人は十分に良い中心極限定理ユーザーになれるだろう.

3.2 中心極限定理の特性函数を使った証明

中心極限定理: $X_1, X_2, X_3, \ldots$ が独立同分布な確率変数の列であるとき, $\mu = E[X_k]$ が定義されていて, $\sigma^2 = \mathrm{var}(X_k) = E[(X_k - \mu)^2] < \infty$ でかつ, $E[|X_k - \mu|^3] < \infty$ となっていると仮定する. このとき,

$$\bar{X}_n = \frac{1}{n}\sum_{k=1}^{n} X_k, \quad Z_n = \frac{\sqrt{n}\,(\bar{X}_n - \mu)}{\sigma}$$

とおくと, $n \to \infty$ で Z_n の分布は標準正規分布に近付く.

証明: X_k の標準化を $Y_k = (X_k - \mu)/\sigma$ と書くことにする. $Y_1, Y_2, \ldots$ も独立同分布になり, $E[Y_k] = 0, \quad E[Y_k^2] = 1$ が成立している. ゆえに Y_k の特性函数 $\varphi(t)$ は k によらず,

$$\varphi(t) = E[e^{itY_k}] = 1 + iE[Y_k]t - E[Y_k^2]\frac{t^2}{2} + O(t^3) = 1 - \frac{t^2}{2} + O(t^3)$$

という形になる. そして,

$$\frac{1}{\sqrt{n}}\sum_{k=1}^{n} Y_k = \frac{\sqrt{n}}{\sigma}\frac{1}{n}\sum_{k=1}^{n}(X_k - \mu) = \frac{\sqrt{n}}{\sigma}(\bar{X}_n - \mu) = Z_n$$

なので, Z_n の特性函数は, $n \to \infty$ で

$$\begin{aligned}\varphi_{Z_n}(t) &= E\left[\exp\left(it\frac{1}{\sqrt{n}}\sum_{k=1}^{n} Y_k\right)\right] = E\left[\prod_{k=1}^{n}\exp\left(i\frac{t}{\sqrt{n}}Y_k\right)\right] \\ &= \prod_{k=1}^{n} E\left[\exp\left(i\frac{t}{\sqrt{n}}Y_k\right)\right] = \varphi\left(\frac{t}{\sqrt{n}}\right)^n = \left(1 - \frac{t^2}{2n} + O(n^{-3/2})\right)^n \to e^{-t^2/2}.\end{aligned}$$

と標準正規分布の特性函数 $e^{-t^2/2}$ に収束する. (3つめの等号で $Y_1, \ldots, Y_n$ の独立性を使った.)

ゆえに Z_n の分布は $n \to \infty$ で標準正規分布に近付く.

証明終

注意: $f(x)$ のFourier変換を

$$\hat{f}\,(t) = \int_{-\infty}^{\infty} f(x)e^{-itx}\,dx$$

と定義するとき, 逆Fourier変換によって,

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}\,(t)e^{itx}\,dt$$

が成立しているので, 確率変数 X について $f(X)$ の期待値は, X の特性函数 $\varphi_X(t) = E[e^{itX}]$ を使って,

$$E[f(X)] = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}\,(x)\varphi_X(t)\,dt$$

と書ける(積分と期待値を取る操作を交換できる). このことから, 確率変数 X の特性函数と確率変数 Y の特性函数が近ければ, $f(X)$ と $f(Y)$ の期待値も近くなる. このことから, 分布の近似は特性函数の近似で扱えることがわかる.

3.3 中心極限定理の収束の速さと歪度

前節の証明より, 独立同分布確率変数列 $X_1, X_2, \ldots$ に対して,

$$\mu = E[X_k], \quad \sigma = \sqrt{E[(X - \mu)^2]}, \quad Y_k = \frac{X_k - \mu}{\sigma}, \quad \varphi(t) = E[e^{itY_k}]$$

とおいたときの, $n \to \infty$ での $\varphi(t/\sqrt{n})^n \to e^{-t^2/2}$ という収束の速さを調べれば, 中心極限定理による正規分布への収束の速さがわかる.

$$\bar{\mu}_3 = E[Y_k^3] = E\left[\left(\frac{X_k - \mu}{\sigma}\right)^3\right]$$

とおくと,

$$\varphi\left(\frac{t}{\sqrt{n}}\right) = E[e^{itY_k}] = 1 - \frac{t^2}{2n} - i\bar{\mu}_3\frac{t^3}{6n\sqrt{n}} + O(n^{-2})$$

なので,

$$\log\varphi\left(\frac{t}{\sqrt{n}}\right)^n = n\log\left(1 - \frac{t^2}{2n} - i\bar{\mu}_3\frac{t^3}{6n\sqrt{n}} + O(n^{-2})\right) = -\frac{t^2}{2} - i\bar{\mu}_3\frac{t^3}{6\sqrt{n}} + O(n^{-1}).$$

これは $n \to \infty$ での $\log\varphi\big(t/\sqrt{n}\big)^n \to -t^2/2$ の収束の速さは, $Y_k = (X_k - \mu)/\sigma$ の3次のモーメント $\bar{\mu}_3$ の絶対値の大きさで大体決まっていることがわかる. $\bar{\mu}_3$ の絶対値が小さいほど収束が速く, 大きいほど収束が遅い.

$\bar{\mu}_3$ は X_k の分布の期待値 μ を中心とする非対称性の σ によって適切に正規化した尺度になっている. $\bar{\mu}_3$ は $Y_k = (X_k - \mu)/\sigma$ の3次のキュムラントにも一致している:

$$K_{Y_k}(t) = \log E[e^{tY_k}] = \log\left(1 + \frac{t^2}{2} + \bar{\mu}_3\frac{t^3}{3!} + O(t^4)\right) = \frac{t^2}{2} + \bar{\mu}_3\frac{t^3}{3!} + O(t^4).$$

ここでの $t^3/3!$ の係数 $\bar{\kappa}_3 = \bar{\mu}_3$ は X_k の **歪度** (skewness) と呼ばれるのであった.

3.4 中心極限定理のキュムラント母函数を使った証明

中心極限定理: $X_1, X_2, X_3, \ldots$ が独立同分布な確率変数の列であるとき, $\mu = E[X_k]$ が定義されていて, $\sigma^2 = \mathrm{var}(X_k) = E[(X_k - \mu)^2] < \infty$ でかつ, $E[|X_k - \mu|^3] < \infty$ となっており, さらに各 X_k のキュムラント母函数がうまく定義されているとする. このとき,

$$\bar{X}_n = \frac{1}{n}\sum_{k=1}^n X_k, \quad Z_n = \frac{\sqrt{n}\,(\bar{X}_n - \mu)}{\sigma}$$

とおくと, $n \to \infty$ で Z_n の分布は標準正規分布に近付く.

証明: X_k の標準化を $Y_k = (X_k - \mu)/\sigma$ と書くことにする. $Y_1, Y_2, \ldots$ も独立同分布になり, $E[Y_k] = 0, \quad E[Y_k^2] = 1$ が成立している. ゆえに Y_k のキュムラント母函数 $K(t)$ は k によらず,

$$K(t) = K_{Y_k}(t) = \frac{t^2}{2} + O(t^3)$$

という形になる. そして,

$$\frac{1}{\sqrt{n}}\sum_{k=1}^n Y_k = \frac{\sqrt{n}}{\sigma}\frac{1}{n}\sum_{k=1}^n (X_k - \mu) = \frac{\sqrt{n}}{\sigma}(\bar{X}_n - \mu) = Z_n$$

なので, Z_n のキュムラント母函数は, $n \to \infty$ で

$$\begin{aligned} K_{Z_n}(t) &= K_{Y_1/\sqrt{n} + \cdots + Y_n/\sqrt{n}}(t) = K_{Y_1}\left(\frac{t}{\sqrt{n}}\right) + \cdots + K_{Y_1}\left(\frac{t}{\sqrt{n}}\right) \\ &= nK\left(\frac{t}{\sqrt{n}}\right) = n\left(\frac{t^2}{2n} + O(n^{-3/2})\right) = \frac{t^2}{2} + O(n^{-1/2})) \to \frac{t^2}{2} \end{aligned}$$

と標準正規分布のキュムラント母函数 $t^2/2$ に収束する.

ゆえに Z_n の分布は $n \to \infty$ で標準正規分布に近付く.

証明終

注意: このようにキュムラント母函数を使うと証明が非常にシンプルになる.

3.5 中心極限定理の収束の速さと歪度と尖度

前節の証明より, 独立同分布確率変数列 $X_1, X_2, \ldots$ に対して,

$$\mu = E[X_k], \quad \sigma = \sqrt{E[(X-\mu)^2]}, \quad K(t) = K_{(X_k-\mu)/\sigma}(t) = \log E\left[\left(\frac{X_k-\mu}{\sigma}\right)\right]$$

とおいたときの, $n \to \infty$ での $nK(t/\sqrt{n}) \to t^2/2$ という収束の速さを調べれば, 中心極限定理による正規分布への収束の速さがわかる. X_k の歪度(わいど) $\bar{\kappa}_3$ と尖度(せんど) $\bar{\kappa}_4$ は

$$\bar{\kappa}_3 = E\left[\left(\frac{X_k-\mu}{\sigma}\right)^3\right], \quad \bar{\kappa}_4 = E\left[\left(\frac{X_k-\mu}{\sigma}\right)^4\right] - 3$$

と表され, X_k の標準化のキュムラント母函数 $K(t)$ の展開の係数になっているのであった:

$$K(t) = \frac{t^2}{2} + \bar{\kappa}_3 \frac{t^3}{3!} + \bar{\kappa}_4 \frac{t^4}{4!} + O(t^5).$$

このとき,

$$nK\left(\frac{t}{\sqrt{n}}\right) = n\left(\frac{t^2}{2n} + \bar{\kappa}_3 \frac{t^3}{3!\,n^{3/2}} + \bar{\kappa}_4 \frac{t^4}{4!\,n^2} + O(n^{-5/2})\right) = \frac{t^2}{2} + \bar{\kappa}_3 \frac{t^3}{3!\,\sqrt{n}} + \bar{\kappa}_4 \frac{t^4}{4!\,n} + O(n^{-3/2})$$

これが $t^2/2$ に収束する速さは $\bar{\kappa}_3 \neq 0$ ならば $O(n^{-1/2})$ のオーダーになり, 歪度 $\bar{\kappa}_3$ の絶対値が大きいほど遅くなる. そして, $\bar{\kappa}_3 = 0$ ならば $O(n^{-1})$ のオーダーでの収束になり, 尖度 $\bar{\kappa}_4$ の絶対値が大きいほど収束は遅くなる.

3.6 中心極限定理のTaylorの定理のみを使う証明

以上では中心極限定理を特性函数やキュムラント母函数を使って証明する方法を(論理的には大雑把に)紹介した.

その方法は本質的にFourier解析に依存している.

実はFourier解析に依存せずに, 本質的にTaylorの定理(Taylor展開の剰余項版)しか使わない初等的な証明も存在する.

しかし, 読者の中にはTaylorの定理も証明をよく知らない人もいるかもしれないので, 以下では丁寧にTaylorの定理の証明の概略から説明して行くことにする.

3.6.1 多重積分の書き方

以下では多重積分での括弧を略して書くために, 次の式の右辺を左辺のように書くことにする:

$$\int_a^b dx\, f(x) = \int_a^b f(x)\, dx.$$

このように書くと, 次の右辺のように括弧が沢山必要になる多重積分の式を左辺のようにシンプルに書くことができる:

$$\int_a^x dx_1 \int_a^{x_1} dx_2 \int_a^{x_2} dx_3\, g(x_3) = \int_a^x \left(\int_a^{x_1} \left(\int_a^{x_2} g(x_3)\, dx_3\right) dx_2\right) dx_1$$

しかも, この式の右辺の書き方だと, 積分変数の x_i 達の指定が式の最後の方でされているので, $\int_a^{x_i}$ の部分の解釈を確定させるためには式を最後の方まで全部見る必要が生じてしまう. 左辺の書き方だとそのような問題が生じない.

3.6.2 積分剰余項型のTaylorの定理

以下では次の結果を示そう:

$$R_4(x,a) = \int_a^x dx_1 \int_a^{x_1} dx_2 \int_a^{x_2} dx_3 \int_a^{x_3} dx_4\, f^{(4)}(x_4)$$

とおくと,

$$f(x) = f(a) + f(a)(x-a) + f''(a)\frac{(x-a)^2}{2} + f'''(a)\frac{(x-a)^3}{3!} + R_4(x,a).$$

これを示すためには4階の導函数 $f^{(4)}(x)$ を4回不定積分すればよい. すなわち, 「微分して不定積分するときに適切に積分定数を決めてやればもとの函数に戻る」という結果

$$g(x) = g(a) + \int_a^x dx'\, g'(x')$$

を4回使うと,

$$
\begin{aligned}
f'''(x_3) &= f'''(a) + \int_a^{x_3} dx_4\, f^{(4)}(x_4), \\
f''(x_2) &= f''(a) + \int_a^{x_2} dx_3\, f'''(x_3) \\
&= f''(a) + f'''(a)(x_2 - a) + \int_a^{x_2} dx_3 \int_a^{x_3} dx_4\, f^{(4)}(x_4) \\
f'(x_1) &= f'(a) + \int_a^{x_1} dx_2\, f''(x_2) \\
&= f'(a) + f''(a)(x_1 - a) + f'''(a)\frac{(x_1 - a)^2}{2} \\
&+ \int_a^{x_1} dx_2 \int_a^{x_2} dx_3 \int_a^{x_3} dx_4\, f^{(4)}(x_4) \\
f(x) &= f(a) + \int_a^{x} dx_1\, f'(x_1) \\
&= f(a) + f'(a)(x - a) + f''(a)\frac{(x - a)^2}{2} + f'''(a)\frac{(x - a)^3}{3!} \\
&+ \underbrace{\int_a^{x} dx_1 \int_a^{x_1} dx_2 \int_a^{x_2} dx_3 \int_a^{x_3} dx_4\, f^{(4)}(x_4)}_{=R_4(x,a)}.
\end{aligned}
$$

不定積分を繰り返す過程で,

$$
\int_a^x dx'\, \frac{(x' - a)^k}{k!} = \frac{(x - a)^{k+1}}{(k+1)!}
$$

となることから, 次の形の式が得られた:

$$
f(a) + f'(a)(x - a) + f''(a)\frac{(x - a)^2}{2} + f'''(a)\frac{(x - a)^3}{3!}
$$

以上とまったく同様にして次を示すことができる:

$$
R_n(x, a) = \int_a^x dx_1 \int_a^{x_1} dx_2 \cdots \int_a^{x_{n-1}} dx_n\, f^{(n)}(x_n)
$$

とおくと,

$$
f(x) = \sum_{k=0}^{n-1} f^{(k)}(a)\frac{(x - a)^k}{k!} + R_n(x, a).
$$

この結果を **積分剰余項型のTaylorの定理** と呼ぶことにする.

もしかしたら, 大学1年生のときの講義で剰余項の形が別のTaylorの定理について複雑な証明を聴いた人が多いかもしれないが, Taylorの定理の本質は「沢山微分して沢山積分すればもとの函数に戻る」ということに過ぎず, 直観的にはほとんど明らかな結果に過ぎない. 明らかな結果について「わけがわからないがとにかく成立する定理」だと認識してしまうことは数学を理解するときに避けるべきことである.

3.6.3 積分剰余項の別の表示

剰余項 $R_n(x, t)$ は次の式で定義されると考えてよい:

$$
R_n(x, t) = f(x) - \sum_{k=0}^{n-1} f^{(k)}(t)\frac{(x - t)^k}{k!}.
$$

このとき, $R_n(x, x) = 0$ でかつ, 両辺を t で偏微分すると,

$$
\begin{aligned}
\frac{\partial}{\partial t} R_n(x, t) &= \sum_{k=1}^{n-1} f^{(k)}(t)\frac{(x - t)^{k-1}}{(k-1)!} - \sum_{k=0}^{n-1} f^{(k+1)}(t)\frac{(x - t)^k}{k!} \\
&= \sum_{k=1}^{n-1} f^{(k)}(t)\frac{(x - t)^{k-1}}{(k-1)!} - \sum_{k=1}^{n} f^{(k)}(t)\frac{(x - t)^{k-1}}{(k-1)!} \\
&= -f^{(n)}(t)\frac{(x - t)^{n-1}}{(n-1)!}.
\end{aligned}
$$

この偏導函数を t について x から a まで積分することによって(a から x まで積分して -1 倍することと同じ), 次が得られる:

$$R_n(x,a) = \int_a^x f^{(n)}(t)\frac{(x-t)^{n-1}}{(n-1)!}\,dt.$$

以上によっても積分剰余項型のTaylorの定理がシンプルに証明されたことになる.

Taylorの定理の証明は易しい!

注意: $h = x-a$, $t = a+sh$ とおくと, $R_n(x,a)$ は次のようにも表される:

$$R_n(a+h,a) = h^n\int_0^1 f^{(n)}(a+sh)\frac{(1-s)^{n-1}}{(n-1)!}\,ds.$$

すなわち,

$$f(a+h) = \sum_{k=0}^{n-1} f^{(k)}(a)\frac{h^k}{k!} + h^n\int_0^1 f^{(n)}(a+sh)\frac{(1-s)^{n-1}}{(n-1)!}\,ds.$$

後でTaylorの定理をこの形で使うことになる.

注意: 上で求めた積分剰余項の公式と前節で求めた公式の関係. 前節では

$$R_n(x,a) = \int_a^x dx_1\int_a^{x_1} dx_2\cdots\int_a^{x_{n-1}} dx_n\, f^{(n)}(x_n)$$

を示した. 簡単のため $a \leq x$ と仮定する($a \geq x$ の場合も同様である). そのとき, すぐ上の式の右辺の積分の範囲は

$$a \leq x_n \leq x_{n-1} \leq x_{n-2} \leq \cdots \leq x_2 \leq x_1 \leq x$$

なので, x_n による積分を一番外側に出すと次のように書き直される:

$$R_n(x,a) = \int_a^x f^{(n)}(x_n)\left(\int_{x_n}^x dx_1\int_{x_n}^{x_1} dx_2\cdots\int_{x_n}^{x_{n-2}} dx_{n-1}\right)dx_n.$$

$\int_{x_n}^x dx_1\int_{x_n}^{x_1} dx_2\cdots\int_{x_n}^{x_{n-2}} dx_{n-1}$ の部分は 1 の $n-1$ 回の不定積分なので $(x-x_n)^{n-1}/(n-1)!$ に等しい:

$$\int_{x_n}^x dx_1\int_{x_n}^{x_1} dx_2\cdots\int_{x_n}^{x_{n-2}} dx_{n-1} = \frac{(x-x_n)^{n-1}}{(n-1)!}.$$

これより, 前節で求めた積分剰余項の公式から上で求めた公式が得られることがわかった.

3.6.4 微分剰余項型のTaylorの定理

$f(t)$ は C^n 級であると仮定する.

簡単のため $a \leq x$ と仮定する($a \geq x$ の場合も同様である). a 以上 x 以下の t についての $f^{(n)}(t)$ の最小値と最大値をそれぞれ $f^{(n)}(\alpha)$, $f^{(n)}(\beta)$ ($a \leq \alpha, \beta \leq x$) と書くと,

$$f^{(n)}(\alpha)\frac{(x-a)^n}{n!} \leq R_n(x,a) = \int_a^x f^{(n)}(t)\frac{(x-t)^{n-1}}{(n-1)!}\,dt \leq f^{(n)}(\beta)\frac{(x-a)^n}{n!}.$$

ゆえに, 中間値の定理より, α と β のあいだのある実数 ξ が存在して,

$$f^{(n)}(\xi)\frac{(x-a)^n}{n!} = R_n(x,a).$$

これで積分を使わずに微分だけを使った剰余項の表示が得られた.

3.6.5 Taylorの定理を使った中心極限定理の証明

$X_1, X_2, X_3, \ldots$ は同分布独立確率変数列であるとし, それらの共通の期待値 $\mu = E[X_i]$ は 0 であるとし, 共通の分散 $\sigma^2 = E[(X_i-\mu)^2]$ は 1 であるとし, $E[|X_i|^m] < \infty$ ($m = 3,4,5$)であると仮定し, X_i の歪度と尖度を次のように書くことにする:

$$\bar{\kappa}_3 = E[X_i^3], \quad \bar{\kappa}_4 = E[X_i^4] - 3.$$

このとき, Z を標準正規分布に従う確率変数であるとし,

$$Z_n = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_i.$$

とおくと, 有限区間の外で 0 になる C^5 級函数 $f(z)$ に対して, $m = 3, 4, 5$ に関する $|f^{(m)}(z)|$ の最大値を M_m と書くと, $n \to \infty$ において次が成立している:

$$\begin{aligned}
&|E[f(Z_n)] - E[f(Z)]| \\
&\leq \frac{M_3}{3!\sqrt{n}}|\bar{\kappa}_3| + \frac{M_4}{4!\,n}|\bar{\kappa}_4| + \frac{M_5}{5!\,n\sqrt{n}}(E[|X_i|^5] + E[|Z|^5]).
\end{aligned}$$

特に次が成立している:

$$\lim_{n\to\infty} E[f(Z_n)] = E[f(Z)].$$

注意: 必要ならば X_i を $(X_i - \mu)/\sigma$ で置き換えることによって, 上のように仮定しても一般性が失われないことに注意せよ.

証明:

Step 1. $Y_1, Y_2, Y_3, \ldots$ は $X_1, X_2, X_3, \ldots$ とは別の独立同分布確率変数列であるとし, X_i, Y_i の全体は独立でかつ $E[Y_i] = 0$, $E[{Y_i}^2] = 1$, $E[|Y_i|^3] < \infty$ を満たしていると仮定する.

W_n, $Z_n^{(k)}$, $A_n^{(k)}$ を次のように定める:

$$\begin{aligned}
W_n &= \frac{1}{\sqrt{n}} \sum_{i=1}^{n} Y_i, \\
Z_n^{(k)} &= \frac{1}{\sqrt{n}}(X_1 + \cdots + X_k + Y_{k+1} + \cdots + Y_n), \\
A_n^{(k)} &= \frac{1}{\sqrt{n}}(X_1 + \cdots + X_{k-1} + Y_{k+1} + \cdots + Y_n).
\end{aligned}$$

このとき,

$$Z_n = Z_n^{(n)}, \quad W_n = Z_n^{(0)}, \quad Z_n^{(k)} = A_n^{(k)} + \frac{X_k}{\sqrt{n}}, \quad Z_n^{(k-1)} = A_n^{(k)} + \frac{Y_k}{\sqrt{n}}.$$

以下では, 大雑把に言うと, $Z_n^{(k)}$ の分布と $Z_n^{(k-1)}$ の分布の違いは小さいことを示す.

まず, $G(a, h)$ を次のように定める:

$$G(a, h) = \int_0^1 f^{(5)}(a + sh)\frac{(1-s)^4}{4!}\,ds.$$

$m = 3, 4, 5$ に対する $|f^{(m)}(z)|$ の最大値を M_m と書くと,

$$|G(a, h)| \leq M_5 \int_0^1 \frac{(1-s)^4}{4!}\,ds = \frac{M_5}{5!}.$$

$f(z)$ にTaylorの定理を適用すると,

$$f(a+h) = f(a) + hf'(a) + \frac{h^2}{2}f''(a) + \frac{h^3}{3!}f'''(a) + \frac{h^4}{4!}f^{(4)}(a) + h^3 G(a, h).$$

これを $a = A_n^{(k)}$, $h = X_k/\sqrt{n}, Y_k/\sqrt{n}$ に適用すると,

$$\begin{aligned}
f(Z_n^{(k)}) &= f(A_n^{(k)}) + \frac{X_k}{\sqrt{n}}f'(A_n^{(k)}) + \frac{X_k^2}{2n}f''(A_n^{(k)}) \\
&\quad + \frac{X_k^3}{3!\,n\sqrt{n}}f'''(A_n^{(k)}) + \frac{X_k^4}{4!\,n^2}f^{(4)}(A_n^{(k)}) + \frac{X_k^5}{n^2\sqrt{n}}G\left(A_n^{(k)}, \frac{X_k}{\sqrt{n}}\right), \\
f(Z_n^{(k-1)}) &= f(A_n^{(k)}) + \frac{Y_k}{\sqrt{n}}f'(A_n^{(k)}) + \frac{Y_k^2}{2n}f''(A_n^{(k)}) \\
&\quad + \frac{Y_k^3}{3!\,n\sqrt{n}}f'''(A_n^{(k)}) + \frac{Y_k^4}{4!\,n^2}f^{(4)}(A_n^{(k)}) + \frac{Y_k^5}{n^2\sqrt{n}}G\left(A_n^{(k)}, \frac{Y_k}{\sqrt{n}}\right).
\end{aligned}$$

$A_n^{(k)}, X_k, Y_k$ は独立で, $E[X_k] = E[Y_k] = 0$, $E[X_k^2] = E[{Y_k}^2] = 1$ であることより,

$$
\begin{aligned}
&E[f(Z_n^{(k)})] - E[f(Z_n^{(k-1)})] \\
&= \frac{E[f'''(A_n^{(k)})]}{3!\,n\sqrt{n}}(E[X_k^3] - E[Y_k^3]) + \frac{E[f^{(4)}(A_n^{(k)})]}{4!\,n^2}(E[X_k^4] - E[Y_k^4]) \\
&+ \frac{1}{n^2\sqrt{n}}\left(E\left[X_k^5 G\left(A_n^{(k)}, \frac{Y_k}{\sqrt{n}}\right)\right] + E\left[Y_k^5 G\left(A_n^{(k)}, \frac{Y_k}{\sqrt{n}}\right)\right]\right)
\end{aligned}
$$

$m = 3, 4, 5$ に対する $|f^{(m)}(z)|$ の最大値を M_m と書くことにしてあったので,

$$
\begin{aligned}
&\left|E[f(Z_n^{(k)})] - E[f(Z_n^{(k-1)})]\right| \\
&\le \frac{M_3}{3!\,n\sqrt{n}}|E[X_k^3] - E[Y_k^3]| + \frac{M_4}{4!\,n^2}|E[X_k^4] - E[Y_k^4]| \\
&+ \frac{M_5}{5!\,n^2\sqrt{n}}(E[|X_k|^5] + E[|Y_k|^5]).
\end{aligned}
$$

Step 2. X_k, Y_k の分布が k によらないことに注意しながら, これを $k = 1, 2, \ldots, n$ について足し上げることによって次が得られる:

$$
\begin{aligned}
&|E[f(Z_n)] - E[f(W_n)]| \\
&\le \frac{M_3}{3!\,\sqrt{n}}|E[X_1^3] - E[Y_1^3]| + \frac{M_4}{4!\,n}|E[X_1^4] - E[Y_1^4]| \\
&+ \frac{M_5}{5!\,n\sqrt{n}}(E[|X_1|^5] + E[|Y_1|^5]).
\end{aligned}
$$

Step 3. $Y_1, Y_2, Y_3, \ldots$ の各々が標準正規分布に従っている場合. そのとき, $\sum_{i=1}^n Y_i$ は平均 0, 分散 n の正規分布に従うので, $W_n = (1/\sqrt{n})\sum_{i=1}^n Y_i$ は標準正規分布に従う. ゆえに $E[f(W_n)] = E[f(Z)]$. さらにこのとき, $E[Y_1^3] = 0$, $E[Y_1^4] = 3$ となっていることにも注意せよ. ゆえに, この場合に Step 2 の結果を適用すると次が得られる:

$$
\begin{aligned}
&|E[f(Z_n)] - E[f(Z)]| \\
&\le \frac{M_3}{3!\,\sqrt{n}}|\bar{\kappa}_3| + \frac{M_4}{4!\,n}|\bar{\kappa}_4| + \frac{M_5}{5!\,n\sqrt{n}}(E[|X_i|^5] + E[|Z|^5]).
\end{aligned}
$$

これが示したい不等式であった.

証明終

注意: Z を標準正規分布に従う確率変数とするとき,

$$
E[|Z|^m] = \frac{\Gamma((m+1)/2)}{\sqrt{2\pi}} = \begin{cases} 1 \cdot 3 \cdots (2k-1) & (m = 2k \in 2\mathbb{Z}_{\ge 0} + 1) \\ 2^{k+1}k!/\sqrt{2\pi} & (m = 2k+1 \in 2\mathbb{Z}_{\ge 0}) \end{cases}
$$

In [76]:
```
[(2quadgk(z -> z^(2k)*pdf(Normal(), z), 0, Inf)[1],
        2^(k+1/2)*gamma(k+1/2)/√(2π), prod(1:2:2k-1)) for k in 0:4]
```

Out[76]:
```
5-element Vector{Tuple{Float64, Float64, Int64}}:
 (0.9999999999999998, 1.0000000000000002, 1)
 (1.0000000000000009, 1.0000000000000002, 1)
 (3.000000000000005, 3.0000000000000004, 3)
 (15.00000000000029, 15.000000000000004, 15)
 (105.00000000000988, 105.00000000000001, 105)
```

In [77]:
```
[(2√(2π)*quadgk(z -> z^(2k+1)*pdf(Normal(), z), 0, Inf)[1],
        2^(k+1)*gamma(k+1), 2^(k+1)*factorial(k)) for k in 0:4]
```

Out[77]:
```
5-element Vector{Tuple{Float64, Float64, Int64}}:
 (1.9999999999999993, 2.0, 2)
 (4.0, 4.0, 4)
 (16.0000000000001, 16.0, 16)
 (96.00000000000463, 96.0, 96)
 (767.9999999999999, 768.0, 768)
```

3.6.6 中心極限定理の収束の速さと歪度と尖度(再)

前節で示した不等式

$$
\begin{aligned}
&|E[f(Z_n)] - E[f(Z)]| \\
&\le \frac{M_3}{3!\,\sqrt{n}}|\bar{\kappa}_3| + \frac{M_4}{4!\,n}|\bar{\kappa}_4| + \frac{M_5}{5!\,n\sqrt{n}}(E[|X_i|^5] + E[|Z|^5]).
\end{aligned}
$$

より, 再び, X_i 達に共通の歪度 $\bar{\kappa}_3$ と尖度 $\bar{\kappa}_4$ の大きさが中心極限定理の収束の速さにどのように関係しているかがわかる.

$\bar{\kappa}_3 = 0$ ならば中心極限定理の収束の速さのオーダーは $O(1/\sqrt{n})$ から $O(1/n)$ に速くなる. $\bar{\kappa}_3 = 0$ の場合には $\bar{\kappa}_4$ が 0 に近いと中心極限定理の収束はさらに速くなる.

3.7 問題: 中心極限定理の収束の様子のグラフ

中心極限定理による正規分布への収束の様子をコンピュータでグラフを描いて確認せよ. 収束が速い場合($n = 10$ ですでに正規分布に十分に近い場合)と遅い場合の両方の例を作れ.

注意: 特に収束が遅い場合の様子を確認することが重要である. 中心極限定理は統計学において空気のごとく使用されるが, 数学的には収束が遅い場合があるので, 現実の分析で中心極限定理を使うことには注意を要する. しかし, 単に注意を要することを知っているだけでは役に立たない. 中心極限定理による収束が遅い場合について具体的な例を知っていれば, 危険を察知し易くなるだろう.

解答例は以下の通り.

In [78]:

```
# 中心極限定理による収束が速い場合1
# distを左右対称な分布でかつ「おとなしめなもの」とする
dist = Uniform(0, 1)
@show skewness(dist) kurtosis(dist)
plot(dist, -0.5, 1.5; label="")
title!("Uniform(0, 1)")
```

```
skewness(dist) = 0.0
kurtosis(dist) = -1.2
```

Out[78]:

In [79]:

```
# 分布distの独立同分布確率変数達(n個)の実現値(要するに乱数)を大量に(L個)生成
n = 10
L = 10^6
Xs = rand(dist, n, L) # Xs の s は X = (X_1, … , X_n) 達意味(複数という意味)
```

Out[79]:

```
10×1000000 Matrix{Float64}:
 0.672547   0.0848108  0.524284  0.33867     …  0.0986632  0.705264  0.683651  0.764941
 0.337707   0.306371   0.582008  0.324429       0.681252   0.153969  0.680763  0.342673
 0.395861   0.178901   0.979898  0.366622       0.437717   0.640549  0.569048  0.7454
 0.0274189  0.252751   0.49977   0.677686       0.4238     0.836334  0.850832  0.283192
 0.390767   0.0892799  0.33393   0.436294       0.663443   0.274744  0.455814  0.672315
 0.190572   0.150144   0.988904  0.482095    …  0.870442   0.228277  0.556281  0.733509
 0.515534   0.710163   0.722886  0.725216       0.540596   0.111925  0.478272  0.13938
 0.977059   0.106152   0.312564  0.750852       0.232703   0.39544   0.677336  0.210737
 0.989893   0.227767   0.748116  0.177032       0.174567   0.632077  0.585337  0.0599564
 0.306079   0.944226   0.966805  0.00705588     0.940918   0.268038  0.301074  0.0494107
```

In [80]:

```
# Z_n = √n(X̄_n - μ)/σ を大量に計算
μ = mean(dist)
σ = std(dist)
Zns = [√n*(mean(X) - μ)/σ for X in eachcol(Xs)] # Zns はZ_n達という意味
first(Zns, 5)
```

Out[80]:

```
5-element Vector{Float64}:
 -0.2153235809234101
 -2.1354980522614544
  1.817524904240514
 -0.7822010477082603
 -0.6391880610739005
```

In [81]:

```
# Z_n達のヒストグラムと標準正規分布の密度函数を比較
histogram(Zns; norm=true, alpha=0.3, bin=100, label="Zₙ")
plot!(Normal(0, 1); label="Normal(0,1)", lw=2)
title!("n = $n")
```

Out[81]:

In [82]:

```julia
# 以上の手続きを函数化
# さらに
#   * メモリアロケーションの節約と並列化による最適化
#   * 離散分布の場合にも対応
#   * 歪度と尖度を混合モデルの場合にも表示

function plot_central_limit_theorem(dist, n;
        L=10^6,
        μ = mean(dist),
        σ = std(dist),
        a = max(minimum(dist), μ - 5σ),
        b = min(maximum(dist), μ + 5σ),
        disttitle = distname(dist),
        bin = 100,
        kwargs...
    )
    println("skewness(dist) = ", myskewness(dist))
    println("kurtosis(dist) = ", mykurtosis(dist))

    # 分布distをプロット
    if dist isa DiscreteUnivariateDistribution
        P1 = bar(round(Int, a):round(Int, b), x -> pdf(dist, x), ; alpha=0.3, label="")
    else
        P1 = plot(x -> pdf(dist, x), a, b; label="")
    end
    title!(disttitle)

    # 分布distの独立同分布確率変数達(n個)の実現値(要するに乱数)を大量に(L個)生成し,
    # Z_n = √n(X̄_n - μ)/σ を大量に計算することを並列化で実行
    # このような計算には並列化が非常に有効である
    Zns = Vector{Float64}(undef, L)
    tmp = [Vector{Float64}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L # Threads.@threads マクロで並列化
        X = rand!(dist, tmp[Threads.threadid()]) # rand!を使ってアロケーションを節約
        Zns[i] = √n*(mean(X) - μ)/σ
    end

    # Z達のヒストグラムと標準正規分布の密度函数を比較
    if dist isa DiscreteUnivariateDistribution
        summin = max(n*minimum(dist), round(n*μ - 6√n*σ))
        summax = min(n*maximum(dist), round(n*μ + 6√n*σ))
        sumran = summin-0.5:summax+0.5
        bin = @. √n*(sumran/n - μ)/σ
        P2 = histogram(Zns; norm=true, alpha=0.3, bin, label="Zₙ")
    else
        P2 = histogram(Zns; norm=true, alpha=0.3, bin, label="Zₙ")
    end
    plot!(Normal(0,1); label="Normal(0,1)", lw=2)
    title!("n = $n")

    plot(P1, P2; size=(800, 250), kwargs...)
end
```

Out[82]: plot_central_limit_theorem (generic function with 1 method)

In [83]:

```julia
# 中心極限定理による収束が速い場合1の再現
plot_central_limit_theorem(Uniform(0, 1), 10; a=-0.5, b=1.5)
```

```
skewness(dist) = 0.0
kurtosis(dist) = -1.2
```

Out[83]:

In [84]:
```
# 中心極限定理による収束が速い場合2
plot_central_limit_theorem(Beta(0.5, 0.5), 10; a=0.01, b=0.99)
```

```
skewness(dist) = 0.0
kurtosis(dist) = -1.5
```

Out[84]:

In [85]:
```
# 中心極限定理による収束が速い場合3
plot_central_limit_theorem(Bernoulli(0.5), 10)
```

```
skewness(dist) = 0.0
kurtosis(dist) = -2.0
```

Out[85]:

In [86]:
```
# 中心極限定理による収束が速い場合4
plot_central_limit_theorem(Laplace(), 10)
```

```
skewness(dist) = 0.0
kurtosis(dist) = 3.0
```

Out[86]:

In [87]:
```
# 左右非対称な分布の場合1
plot_central_limit_theorem(Gamma(3, 1), 10)
```

```
skewness(dist) = 1.1547005383792517
kurtosis(dist) = 2.0
```

Out[87]:

In [88]:
```
# 左右非対称な分布の場合2
plot_central_limit_theorem(Bernoulli(0.25), 10)
```

```
skewness(dist) = 1.1547005383792517
kurtosis(dist) = -0.666666666666667
```

Out[88]:

In [89]:
```
# 左右非対称な分布の場合3
plot_central_limit_theorem(Poisson(1), 10)
```

```
skewness(dist) = 1.0
kurtosis(dist) = 1.0
```

Out[89]:

In [90]:
```
# 中心極限定理による収束が遅い場合1
# 左右の非対称性が大きな分布を試してみる
# 指数分布は左右の非対称性が大きな分布になっている
plot_central_limit_theorem(Exponential(), 10)
```

```
skewness(dist) = 2.0
kurtosis(dist) = 6.0
```

Out[90]:

In [91]:
```
plot_central_limit_theorem(Exponential(), 40)
```

```
skewness(dist) = 2.0
kurtosis(dist) = 6.0
```

Out[91]:

In [92]:
```
plot_central_limit_theorem(Exponential(), 160)
```

```
skewness(dist) = 2.0
kurtosis(dist) = 6.0
```

Out[92]:

In [93]:
```
# 中心極限定理による収束が遅い場合2
# 自由度1のχ² 分は左右の非対称性が指数分布よりも大きな分布になっている
plot_central_limit_theorem(Chisq(1), 10; a=0.05, b=6)
```

```
skewness(dist) = 2.8284271247461903
kurtosis(dist) = 12.0
```

Out[93]:

In [94]:
```
plot_central_limit_theorem(Chisq(1), 40; a=0.05, b=6)
```

```
skewness(dist) = 2.8284271247461903
kurtosis(dist) = 12.0
```

Out[94]:

In [95]:
```
plot_central_limit_theorem(Chisq(1), 160; a=0.05, b=6)
```

```
skewness(dist) = 2.8284271247461903
kurtosis(dist) = 12.0
```

Out[95]:

以下で使う, 分布 `MixtureModel([Normal(0, 1), Normal(20, 1)], [0.95, 0.05])` の確率密度函数は次の形になる:

$$p(x) = 0.95\,\frac{e^{-x^2/2}}{\sqrt{2\pi}} + 0.05\,\frac{e^{-(x-20)^2/2}}{\sqrt{2\pi}}.$$

これは標準正規分布に割合が 5% の極端な外れ値を付け加えた分布になっている. 何らかの原因で分布に極端な外れ値が混ざっている場合には中心極限定理を使うときに注意を要する.

このように確率密度函数が複数の正規分布の確率密度函数の一次結合になっている分布は **混合正規分布** (mixture normal distribution)と呼ばれている.

In [96]:

```
# 中心極限定理による収束が遅い場合3
# 以下のような分布distも左右の非対称性が大きな分布に分類される
dist = MixtureModel([Normal(0, 1), Normal(20, 1)], [0.95, 0.05])
disttitle = "MixtureModel([Normal(0, 1), Normal(20, 1)], [0.95, 0.05])"
titlefontsize = 9
a, b = -4, 24
plot_central_limit_theorem(dist, 10; a, b, disttitle, titlefontsize)
```

```
skewness(dist) = 3.8236762415246486
kurtosis(dist) = 13.584999999990963
```

Out[96]:

In [97]:

```
plot_central_limit_theorem(dist, 40; a, b, disttitle, titlefontsize)
```

```
skewness(dist) = 3.8236762415246486
kurtosis(dist) = 13.584999999990963
```

Out[97]:

In [98]:
```
plot_central_limit_theorem(dist, 160; a, b, disttitle, titlefontsize)
```

```
skewness(dist) = 3.8236762415246486
kurtosis(dist) = 13.584999999990963
```

Out[98]:

In [99]:
```
# 中心極限定理による収束が遅い場合4
# 対数正規分布は左右の非対称性が非常に大きな分布である
# 右側の裾が太く，外れ値が出やすい
plot_central_limit_theorem(LogNormal(), 10; bin=range(-3, 9, 100))
```

```
skewness(dist) = 6.184877138632554
kurtosis(dist) = 110.9363921763115
```

Out[99]:

In [100]:
```
plot_central_limit_theorem(LogNormal(), 40; bin=range(-3, 9, 100))
```

```
skewness(dist) = 6.184877138632554
kurtosis(dist) = 110.9363921763115
```

Out[100]:

In [101]:
```
plot_central_limit_theorem(LogNormal(), 160; bin=range(-3, 9, 100))
```

```
skewness(dist) = 6.184877138632554
kurtosis(dist) = 110.9363921763115
```

Out[101]:

4 デルタ法

4.1 1次のデルタ法と2次のデルタ法

X は平均 μ, 分散 σ^2 を持つ確率変数であるとし, 滑らかな函数 $f(x)$ にそれを代入して得られる確率変数 $Y = f(X)$ について考える.

x が μ に十分近ければ, $y = f(x)$ はその $x = \mu$ での接線

$$y = f(\mu) + f'(\mu)(x - \mu)$$

で近似される. σ が十分小さければ X の分布の大部分は μ の近くに集まり, 確率変数 $Y = f(X)$ は確率変数

$$Y_1 = f(\mu) + f'(\mu)(X - \mu)$$

で近似される. ゆえに

$$\begin{aligned} &E[Y] \approx E[Y_1] = f(\mu) + f'(\mu)E[X - \mu] = f(\mu), \\ &\mathrm{var}(Y) \approx \mathrm{var}(Y_1) = f'(\mu)^2\sigma^2 \end{aligned}$$

という近似が使える. これを **1次のデルタ法** と呼ぶことにする.

接線よりも精密に2次までの項を加えた

$$y = f(\mu) + f'(\mu)(x - \mu) + \frac{f''(\mu)}{2}(x - \mu)^2$$

は $y = f(x)$ のさらに良い近似になっている. ゆえに, 確率変数 $Y = f(X)$ は, 上の Y_1 よりも,

$$Y_2 = f(\mu) + f'(\mu)(X - \mu) + \frac{f''(\mu)}{2}(X - \mu)^2$$

でより良く近似される. この近似を使えば, $E[Y] \approx E[Y_1] = f(\mu)$ よりも精度が高い

$$E[Y] \approx E[Y_2] = f(\mu) + \frac{f''(\mu)}{2}E[(x - \mu)^2] = f(\mu) + \frac{f''(\mu)\sigma^2}{2}$$

という近似が得られる. これを **2次のデルタ法** と呼ぶことにする.

4.2 おまけ: 2次のデルタ法使用時の近似分散と歪度と尖度に関する不等式

確率変数 X の期待値を μ, 分散を σ^2, 歪度を $\bar{\kappa}_3 = E[((X-\mu)/\sigma)^3]$, 尖度を $\bar{\kappa}_4 = E[((X-\mu)/\sigma)^4] - 3$ と表し, 任意の実数 a, b, c に対して,

$$Y_2 = a + b(X-\mu) + c(X-\mu)^2$$

とおく. このとき, $E[Y_2] = a + c\sigma^2$ となることは易しい. Y_2 の分散は以下のようにして求められる:

$$\begin{aligned}
\mathrm{var}(Y_2) &= E[(Y_2 - (a+c\sigma^2))^2] = E[(-c\sigma^2 + b(X-\mu) + c(X-\mu)^2)^2] \\
&= E[c^2\sigma^4 - 2bc\sigma^2(X-\mu) + (b^2 - 2c^2\sigma^2)(X-\mu)^2 + 2bc(X-\mu)^3 + c^2(X-\mu)^4] \\
&= c^2\sigma^4 + (b^2 - 2c^2\sigma^2)\sigma^2 + 2bc\sigma^3\bar{\kappa}_3 + c^2\sigma^4(\bar{\kappa}_4 + 3) \\
&= b^2\sigma^2 + 2bc\sigma^3\bar{\kappa}_3 + c^2\sigma^4(\bar{\kappa}_4 + 2) \\
&= \begin{bmatrix} b\sigma & c\sigma^2 \end{bmatrix} \begin{bmatrix} 1 & \bar{\kappa}_3 \\ \bar{\kappa}_3 & \bar{\kappa}_4 + 2 \end{bmatrix} \begin{bmatrix} b\sigma \\ c\sigma^2 \end{bmatrix}.
\end{aligned}$$

これが任意の実数 b, c について 0 以上になることから, この結果に現れた 2×2 行列の行列式は 0 以上になることがわかるので, 次の不等式が得られる:

$$\bar{\kappa}_4 + 2 \geq \bar{\kappa}_3^2.$$

「標本分布について」のノートで標本平均と不偏分散の分散共分散行列の非負性を用いて示した「歪度と尖度に関する不等式」が再び証明された.

注意: 以上の証明は本質的に $X-\mu$ と $(X-\mu)^2$ の分散共分散行列が半正定値になることを使う証明と本質的に同じである.

4.3 問題: 2次のデルタ法の不偏分散の平方根の期待値の近似への応用

$X_1, X_2, \ldots, X_n$ を独立同分布な確率変数達であるとし, それらの標本平均と不偏分散を

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i, \quad S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

と書き, X_i 達共通の期待値を μ, 共通の分散を σ^2 と書き, 共通の尖度を $\bar{\kappa}_4 = E[((X_i-\mu)/\sigma)^4] - 3$ と書く. このとき, S^2 の期待値と分散は

$$E[S^2] = \sigma^2, \quad \mathrm{var}(S^2) = \sigma^4\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right)$$

となるのであった. これに2次のデルタ法を適用すると, $S = \sqrt{S^2}$ の期待値の σ からのずれが近似的に

$$E[S] \approx \sigma - \frac{\mathrm{var}(S^2)}{8\sigma^3} = \sigma\left(1 - \frac{1}{8}\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right)\right)$$

のように表されることを確認せよ.

解答例: $f(x) = \sqrt{x}$ とおくと,

$$f'(x) = \frac{1}{2}x^{-1/2}, \quad f''(x) = -\frac{1}{4}x^{-3/2}, \quad f(\sigma^2) = \sigma, \quad f''(\sigma^2) = -\frac{1}{4\sigma^3}$$

なので, $X = S^2$ に2次のデルタ法を適用すると,

$$\begin{aligned}
E[S] &\approx f(\sigma^2) + \frac{f''(\sigma^2)\,\mathrm{var}(S^2)}{2} \\
&= \sigma - \frac{\mathrm{var}(S^2)}{8\sigma^3} = \sigma\left(1 - \frac{1}{8}\left(\frac{\bar{\kappa}_4}{n} + \frac{2}{n-1}\right)\right).
\end{aligned}$$

解答終

注意: 不偏分散 S^2 は分散 σ^2 の不偏推定量だが, その平方根 $S = \sqrt{S^2}$ は標準偏差 σ の不偏推定量にならないことが上の問題からわかる. $\mathrm{var}(S^2) \geq 0$ なので $E[S]$ の値は σ よりも少し小さくなる.

注意: これを理由に $S = \sqrt{S^2}$ は標準偏差の推定量として不適切であると考えては **いけない**. 推定量が不偏であること自体に特別に優先度が高い価値があるわけではない. 推定法に不偏性の条件を課すと誤差が増えてしまう場合がある.

In [102]:
```
function check_deltamethod_for_S(dist, n; L=10^7)
    σ = std(dist)
    ku = mykurtosis(dist)
    @show dist
    println("kurtosis = ", ku)
    @show n

    S = Vector{Float64}(undef, L)
    tmp = [Vector{Float64}(undef, n) for _ in 1:Threads.nthreads()]
    Threads.@threads for i in 1:L
        X = rand!(dist, tmp[Threads.threadid()])
        S[i] = std(X)
    end
    println("approximation = ", σ*(1 - (1/8)*(ku/n + 2/(n-1))))
    println("mean(S) =       ", mean(S))
    println("σ =             ", σ)
end
```

Out[102]:
```
check_deltamethod_for_S (generic function with 1 method)
```

In [103]:
```
check_deltamethod_for_S(Normal(2, 3), 10)
```

```
dist = Normal{Float64}(μ=2.0, σ=3.0)
kurtosis = 0.0
n = 10
approximation = 2.9166666666666665
mean(S) =       2.9181035910662665
σ =             3.0
```

In [104]:
```
check_deltamethod_for_S(Uniform(), 10)
```

```
dist = Uniform{Float64}(a=0.0, b=1.0)
kurtosis = -1.2
n = 10
approximation = 0.28498650787499025
mean(S) =       0.28475298814557065
σ =             0.28867513459481287
```

In [105]:
```
check_deltamethod_for_S(Gamma(4, 3), 10)
```

```
dist = Gamma{Float64}(α=4.0, θ=3.0)
kurtosis = 1.5
n = 10
approximation = 5.720833333333333
mean(S) =       5.757022288766159
σ =             6.0
```

In [106]:
```
check_deltamethod_for_S(Bernoulli(0.4), 10)
```

```
dist = Bernoulli{Float64}(p=0.4)
kurtosis = -1.833333333333333
n = 10
approximation = 0.4875165001955964
mean(S) =       0.4859214468305372
σ =             0.4898979485566356
```

In [107]:
```
check_deltamethod_for_S(LogNormal(), 10; L=10^6)
```

```
dist = LogNormal{Float64}(μ=0.0, σ=1.0)
kurtosis = 110.9363921763115
n = 10
approximation = -0.895778896909659
mean(S) =       1.7171498580585198
σ =             2.1611974158950877
```

In [108]:
```
check_deltamethod_for_S(LogNormal(), 100; L=10^6)
```

```
dist = LogNormal{Float64}(μ=0.0, σ=1.0)
kurtosis = 110.9363921763115
n = 100
approximation = 1.856045541537819
mean(S) =       2.045538958724601
σ =             2.1611974158950877
```

In [109]:
```
check_deltamethod_for_S(LogNormal(), 1000; L=10^6)
```

```
dist = LogNormal{Float64}(μ=0.0, σ=1.0)
kurtosis = 110.9363921763115
n = 1000
approximation = 2.1306871451883986
mean(S) =       2.140192566082289
σ =             2.1611974158950877
```

4.4 問題: 1次のデルタ法の正規分布近似への応用

確率変数 X が近似的に正規分布 $\mathrm{Normal}(\mu, \sigma)$ に従っているとき, 滑らかな函数 $f(x)$ に対して, σ が十分に小さければ, 確率変数 $Y = f(X)$ は近似的に正規分布 $\mathrm{Normal}(f(\mu), |f'(\mu)|\sigma)$ に従うことを示せ.

解答例: x が μ に十分近ければ $y = f(x)$ は $x = \mu$ における接線 $y = f(\mu) + f'(\mu)(x - \mu)$ で近似される. σ が十分に小さければ確率変数 X の分布の大部分は μ の十分近くに留まる. 以上の状況のもとで, 確率変数 $Y = f(X)$ は確率変数 $Y_1 = f(\mu) + f'(\mu)(X - \mu)$ で近似される. X が正規分布 $\mathrm{Normal}(\mu, \sigma)$ に近似的に従っていると仮定しているので, そのアフィン変換である Y_1 も近似的に正規分布に従い, その期待値と分散は

$$E[Y_1] = f(\mu) + f'(\mu)E[X - \mu] \approx f(\mu), \quad \mathrm{var}(Y_1) = f'(\mu)^2 \,\mathrm{var}(X) \approx f'(\mu)^2\sigma^2$$

となる. 以上をまとめると, Y が近似的に正規分布 $\mathrm{Normal}(f(\mu), |f'(\mu)|\sigma)$ に従うことがわかる.

解答終

注意: デルタ法を使うと, 正規分布で近似される確率変数から正規分布に近似的に従う別の確率変数を作れる. 中心極限定理による正規分布近似はデルタ法を使えば大幅に拡張される. 以上では一変数函数 $f(x)$ に正規分布に近似的に従う確率変数 X を代入する場合のみを扱ったが, 多変数函数に複数の正規分布に近似的に従う確率変数達を代入する場合も同じように一次近似で考えればよい.

4.5 問題: 1次のデルタ法による分布の逆数の近似

正値の確率変数 X は正規分布 $\mathrm{Normal}(\mu, \sigma)$ に近似的に従っていると仮定し, σ/μ は十分に小さいと仮定する. このとき, $Y = 1/X$ が正規分布 $\mathrm{Normal}(1/\mu, \sigma/\mu^2)$ に近似的に従うことを示せ.

解答例: $f(x) = 1/x$ とおくと,

$$f(x) = \frac{1/\mu}{1 + (x/\mu - 1)} = \frac{1}{\mu}\left(1 - \left(\frac{x}{\mu} - 1\right) + \left(\frac{x}{\mu} - 1\right)^2 - \cdots\right)$$

なので, $y = f(x)$ は x/μ が 1 に近ければ, その $x = \mu$ での接線

$$y = \frac{1}{\mu}\left(1 - \left(\frac{x}{\mu} - 1\right)\right) = \frac{1}{\mu} - \frac{x - \mu}{\mu^2}$$

で近似される. σ/μ が十分に小さければ X/μ の分布の大部分は 1 の近くに留まる. ゆえに $Y = 1/X$ の分布は $Y_1 = 1/\mu - (X - \mu)/\mu^2$ の分布で近似される. このことから, $Y = 1/X$ が平均 $1/\mu$, 分散 σ^2/μ^4 の正規分布に近似的に従うことがわかる.

解答終

4.6 例: 自由度が大きなχ²分布の逆数の近似

例えば, $X \sim \mathrm{Chisq}(\nu)$ のとき, $E[X] = \nu$, $\mathrm{var}(X) = 2\nu$ であり, ν が大きなとき X は近似的に平均 $\mu = \nu$, 分散 $\sigma^2 = 2\nu$ の正規分布に近似的に従い, ν が大きなとき $\sigma/\mu = \sqrt{2/\nu}$ の値は小さくなる. ゆえに上の問題の結果より, $1/X$ は平均 $1/\mu = 1/\nu$, 分散 $\sigma^2/\mu^4 = 2/\nu^3$ の正規分布に近似的に従う.

注意: $X \sim \mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$ のとき, $1/X \sim \mathrm{InverseGamma}(\nu/2, 1/2)$ なので,

$$\begin{aligned} E[1/X] &= \frac{1/2}{\nu/2 - 1} = \frac{1}{\nu - 2} \approx \frac{1}{\nu}, \\ \mathrm{var}(1/X) &= \frac{(1/2)^2}{(\nu/2 - 1)^2(\nu/2 - 2)} = \frac{2}{(\nu - 2)^2(\nu - 4)} \approx \frac{2}{\nu^3}. \end{aligned}$$

In [110]:
```
InverseChisq(ν) = InverseGamma(ν/2, 1/2)

function plot_invchisq(ν)
    xlim = (max(0, 1/ν - 4√(2/ν^3)), 1/ν + 6√(2/ν^3))
    plot(InverseChisq(ν), xlim...; label="InverseChisq(ν=$ν)")
    plot!(Normal(1/ν, √(2/ν^3)); label="Normal(1/ν, √(2/ν³))", ls=:dash)
    plot!(; legend=:outertop)
end
```

Out[110]: plot_invchisq (generic function with 1 method)

In [111]:
```
plot(plot_invchisq.((10, 40))...; size=(800, 250), layout=(1, 2))
```

Out[111]:

In [112]:
```
plot(plot_invchisq.((160, 640))...; size=(800, 250), layout=(1, 2))
```

Out[112]:

4.7 問題: デルタ法による分布の対数の近似

正値の確率変数 X は正規分布 $\mathrm{Normal}(\mu, \sigma)$ に近似的に従っていると仮定し, σ/μ は十分に小さいと仮定する. このとき, $Y = \log X$ が正規分布 $\mathrm{Normal}(\log \mu, \sigma/\mu)$ に近似的に従うことを示せ.

解答例: $f(x) = \log x$ とおくと,

$$f(x) = \log \nu \left(1 + \left(\frac{x}{\mu} - 1\right)\right) = \log \mu + \left(\frac{x}{\mu} - 1\right) - \frac{1}{2}\left(\frac{x}{\mu} - 1\right)^2 + \cdots$$

なので, $y = f(x)$ は x/μ が 1 に近ければ, その $x = \mu$ での接線

$$y = \log \mu + \left(\frac{x}{\mu} - 1\right) = \log \mu + \frac{x - \mu}{\mu}$$

で近似される. σ/μ が十分に小さければ X/μ の分布の大部分は 1 の近くに留まる. ゆえに $Y = \log X$ の分布は $Y_1 = \log \mu + (X - \mu)/\mu$ の分布で近似される. このことから, $Y = \log X$ が平均 $\log \mu$, 分散 σ^2/μ^2 の正規分布に近似的に従うことがわかる.

解答終

注意: 正値の確率変数の分布は対数を取ることによって左右の非対称性が緩和されて, 正規分布近似がより正確になる場合がある (次の節の例を見よ). 分布の非対称性を補正して正規分布近似をより正確にする工夫は色々研究されている.

4.8 例: 自由度が大きなχ^2分布の対数の近似

例えば, $X \sim \mathrm{Chisq}(\nu)$ のとき, $E[X] = \nu$, $\mathrm{var}(X) = 2\nu$ であり, ν が大きなとき X は近似的に平均 $\mu = \nu$, 分散 $\sigma^2 = 2\nu$ の正規分布に近似的に従い, ν が大きなとき $\sigma/\mu = \sqrt{2/\nu}$ の値は小さくなる. ゆえに上の問題の結果より, $\log X$ は平均 $\log \mu = \log \nu$, 分散 $\sigma^2/\mu^2 = 2/\nu$ の正規分布に近似的に従う.

この場合には余り大きくない自由度 ν におけるχ²分布の左右の非対称性が対数を取ることによって緩和され, 正規分布による近似の精度が高まる.

In [113]:

```
pdf_logchisq(ν, y) = pdf(Chisq(ν), exp(y))*exp(y)

function plot_logchisq(ν)
    xlim = (max(0, ν - 4√(2ν)), ν + 4√(2ν))
    P0 = plot(Chisq(ν), xlim...; label="Chisq(ν=$ν)")
    plot!(Normal(ν, √(2ν)); label="Normal(ν, √(2ν))", ls=:dash)
    plot!(; legend=:outertop)

    xlim = (log(ν) - 5√(2/ν), log(ν) + 4√(2/ν))
    P1 = plot(y -> pdf(Chisq(ν), exp(y))*exp(y), xlim...; label="LogChisq(ν=$ν)")
    plot!(Normal(log(ν), √(2/ν)); label="Normal(log ν, √(2/ν))", ls=:dash)
    plot!(; legend=:outertop)

    plot(P0, P1; size=(800, 250))
end
```

Out[113]: plot_logchisq (generic function with 1 method)

In [114]:

```
plot_logchisq(10)
```

Out[114]:

In [115]:

```
plot_logchisq(20)
```

Out[115]:

In [116]:
```
plot_logchisq(40)
```

Out[116]:

In [117]:
```
plot_logchisq(80)
```

Out[117]:

4.9 問題: ベータ分布の正規分布近似

確率変数達 X, Y は独立であり, $X \sim \mathrm{Gamma}(\alpha, 1)$, $Y \sim \mathrm{Gamma}(\beta, 1)$ を満たしていると仮定する. このとき $T = X/(X+Y)$ とおくと, $T \sim \mathrm{Beta}(\alpha, \beta)$ となるのであった. α, β が大きなとき, X, Y の分布はそれぞれ $\mathrm{Normal}(\alpha, \sqrt{\alpha})$, $\mathrm{Normal}(\beta, \sqrt{\beta})$ に近似的に従う. 2変数函数 $f(x, y) = x/(x+y)$ に1次のデルタ法を適用して, α, β が大きなとき, $T = X/(X+Y) \sim \mathrm{Beta}(\alpha, \beta)$ が

$$\mathrm{Normal}\left(\frac{\alpha}{\alpha+\beta}, \sqrt{\frac{\alpha\beta}{(\alpha+\beta)^3}}\right)$$

に近似的に従うことを示せ.

注意: $T \sim \mathrm{Beta}(\alpha, \beta)$ の正確な期待値と分散は次のようになる:

$$E[T] = \frac{\alpha}{\alpha+\beta}, \quad \mathrm{var}(T) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}.$$

解答例: $f(x, y) = x/(x+y) = 1 - y/(x+y)$ を偏微分すると,

$$f_x(x, y) = \frac{y}{(x+y)^2}, \quad f_y(x, y) = \frac{-x}{(x+y)^2},$$
$$f_{xx}(x, y) = \frac{-2y}{(x+y)^3}, \quad f_{xy}(x, y) = \frac{x-y}{(x+y)^3}, \quad f_{yy}(x, y) = \frac{2x}{(x+y)^3}$$

なので,

$$
\begin{aligned}
T &= \frac{X}{X+Y} = f(X,Y) \\
&= f(\alpha,\beta) + f_x(\alpha,\beta)(X-\alpha) + f_y(\alpha,\beta)(Y-\beta) \\
&+ \frac{1}{2}\left(f_{xx}(\alpha,\beta)(X-\alpha)^2 + 2f_{xy}(\alpha,\beta)(X-\alpha)(Y-\beta) + f_{yy}(\alpha,\beta)(Y-\beta)^2\right) + \cdots \\
&= \frac{\alpha}{\alpha+\beta} + \frac{\beta}{(\alpha+\beta)^2}(X-\alpha) + \frac{-\alpha}{(\alpha+\beta)^2}(Y-\beta) \\
&+ \frac{-\beta}{(\alpha+\beta)^3}(X-\alpha)^2 + \frac{\alpha-\beta}{(\alpha+\beta)^3}(X-\alpha)(Y-\beta) + \frac{\alpha}{(\alpha+\beta)^3}(Y-\beta)^2 + \cdots \\
&= \frac{\alpha}{\alpha+\beta} + \frac{\beta}{\alpha+\beta}\frac{X-\alpha}{\alpha+\beta} + \frac{-\alpha}{\alpha+\beta}\frac{Y-\beta}{\alpha+\beta} \\
&+ \frac{-\beta}{\alpha+\beta}\left(\frac{X-\alpha}{\alpha+\beta}\right)^2 + \frac{\alpha-\beta}{\alpha+\beta}\frac{X-\alpha}{\alpha+\beta}\frac{Y-\beta}{\alpha+\beta} + \frac{\alpha}{\alpha+\beta}\left(\frac{Y-\beta}{\alpha+\beta}\right)^2 + \cdots.
\end{aligned}
$$

ゆえに, α, β の両方が大きいならば(より正確に言えば $\alpha = N\mu$, $\beta = N(1-\mu$, $0 < \mu < 1$ で N が大きいならば),

$$
\frac{\mathrm{std}(X)}{\alpha+\beta} = \frac{\sqrt{\alpha}}{\alpha+\beta}, \quad \frac{\mathrm{std}(Y)}{\alpha+\beta} = \frac{\sqrt{\beta}}{\alpha+\beta}
$$

がどちらも小さくなり, 1次近似

$$
T \approx \frac{\alpha}{\alpha+\beta} + \frac{\beta}{(\alpha+\beta)^2}(X-\alpha) + \frac{-\alpha}{(\alpha+\beta)^2}(Y-\beta)
$$

が有効になり,

$$
\begin{aligned}
E[T] &\approx \frac{\alpha}{\alpha+\beta}, \\
\mathrm{var}(T) &\approx \frac{\beta^4\,\mathrm{var}(X)}{(\alpha+\beta)^4} + \frac{\alpha^2\,\mathrm{var}(Y)}{(\alpha+\beta)^4} = \frac{\beta^2\alpha + \alpha^2\beta}{(\alpha+\beta)^4} = \frac{\alpha\beta}{(\alpha+\beta)^3}.
\end{aligned}
$$

さらにそのとき, X, Y の分布は正規分布で近似されるので, T の分布も正規分布で近似される.

解答終

In [118]:

```
@vars x y
expr = x/(x+y)
[
    expr
    diff(expr, x).simplify()
    diff(expr, y).simplify()
    ;;
    diff(expr, x, x).simplify()
    diff(expr, x, y).simplify()
    diff(expr, y, y).simplify()
]
```

Out[118]:

$$
\begin{bmatrix}
\frac{x}{x+y} & -\frac{2y}{(x+y)^3} \\
\frac{y}{(x+y)^2} & \frac{x-y}{(x+y)^3} \\
-\frac{x}{(x+y)^2} & \frac{2x}{(x+y)^3}
\end{bmatrix}
$$

注意: $X_1, X_2, \ldots, X_n$ が独立同分布な確率変数達で $X_i \sim \mathrm{Gamma}(\alpha/n, \theta)$ のとき, それらの和もガンマ分布に $X_1 + X_2 + \cdots + X_n \sim \mathrm{Gamma}(\alpha, \theta)$ と従う. この事実と中心極限定理から, α が大きなときのガンマ分布 $\mathrm{Gamma}(\alpha, \theta)$ が正規分布 $\mathrm{Normal}(\alpha, \sqrt{\alpha\theta^2})$ で近似されることが導かれる.

ゆえに, 上の問題の結果は, パラメータの値が大きなときのベータ分布が正規分布で近似されることが, ガンマ分布に関する中心極限定理の帰結になっていることを示していることになる.

ベータ分布の確率密度函数の形から直接的に正規分布近似を導くことも易しい(次の節の問題を見よ).

注意終

In [119]:
```
function plot_gammanormal(α, θ)
    m, s = α*θ, √(α*θ^2)
    xlim = (m - 4s, m + 4s)
    plot(Gamma(α, θ), xlim...; label="Gamma(α,θ)")
    plot!(Normal(m, s); label="normal approx", ls=:dash)
    #plot!(; legend=:outertop)
    title!("α = $α, θ = $θ")
end
```

Out[119]: plot_gammanormal (generic function with 1 method)

In [120]:
```
plot(plot_gammanormal(5, 20), plot_gammanormal(20, 5); size=(800, 250))
```

Out[120]:

In [121]:
```
plot(plot_gammanormal(80, 1.25), plot_gammanormal(320, 0.3125); size=(800, 250))
```

Out[121]:

In [122]:
```
function plot_betanormal(α, β)
    m, s = α/(α+β), √(α*β/(α+β)^3)
    xlim = (m - 4s, m + 4s)
    plot(Beta(α, β), xlim...; label="Beta(α,β)")
    plot!(Normal(m, s); label="normal approx", ls=:dash)
    #plot!(; legend=:outertop)
    title!("α = $α, β = $β")
end
```

Out[122]: plot_betanormal (generic function with 1 method)

In [123]:
```
[3, 7] .* (4 .^ (0:3))'
```

Out[123]:
```
2×4 Matrix{Int64}:
 3  12   48  192
 7  28  112  448
```

In [124]:
```
plot(plot_betanormal(3, 7), plot_betanormal(12, 28); size=(800, 250))
```

Out[124]:

In [125]:
```
plot(plot_betanormal(48, 112), plot_betanormal(192, 448); size=(800, 250))
```

Out[125]:

4.10 問題: ベータ分布の正規分布近似の直接的証明

α, β が大きなときにベータ分布 $\mathrm{Beta}(\alpha, \beta)$ が正規分布で近似されることを, その確率密度函数の具体形を使って直接証明せよ.

解答例: $0 < \mu < 1$ と仮定し,

$$\alpha = N\mu, \quad \beta = N(1-\mu), \quad t = \mu + \frac{x}{\sqrt{N}}$$

とおいて, N を大きくすると, Stirlingの公式より,

$$\begin{aligned}
\frac{1}{B(\alpha,\beta)} &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} = \frac{\Gamma(N)}{\Gamma(N\mu)\Gamma(N(1-\mu))} \\
&\sim \frac{N\mu\, N(1-\mu)}{N} \frac{N^N e^{-N}\sqrt{2\pi N}}{(N\mu)^{N\mu} e^{-N\mu}\sqrt{2\pi N\mu}\,(N(1-\mu))^{N(1-\mu)} e^{-N(1-\mu)}\sqrt{2\pi N(1-\mu)}} \\
&= \frac{N\mu(1-\mu)}{\mu^{N\mu}\sqrt{2\pi N\mu}\,(1-\mu)^{N(1-\mu)}\sqrt{1-\mu}} \\
&= \frac{1}{\sqrt{2\pi\mu(1-\mu)}} \frac{\sqrt{N}}{\mu^{N\mu-1}(1-\mu)^{N(1-\mu)-1}}.
\end{aligned}$$

そして

$$
\begin{aligned}
&N\mu \log\left(1+\frac{x}{\sqrt{N}\,\mu}\right)\\
&= N\mu\left(\frac{x}{\sqrt{N}\,\mu}-\frac{x^2}{2N\mu^2}+O(N^{-3/2})\right)\\
&= \sqrt{N}\,x-\frac{x^2}{2\mu}+O(N^{-1/2}),\\
&N(1-\mu)\log\left(1-\frac{x}{\sqrt{N}(1-\mu)}\right)\\
&= N(1-\mu)\left(-\frac{x}{\sqrt{N}(1-\mu)}-\frac{x^2}{2N(1-\mu)^2}+O(N^{-3/2})\right)\\
&= -\sqrt{N}\,x-\frac{x^2}{2(1-\mu)}+O(N^{-1/2}),\\
&\frac{x^2}{2\mu}+\frac{x^2}{2(1-\mu)}=\frac{x^2}{2\mu(1-\mu)}
\end{aligned}
$$

より, 以下が成立することがわかる:

$$
\begin{aligned}
&t^{\alpha-1}(1-t)^{\beta-1}\,dt\\
&=\left(\mu+\frac{x}{\sqrt{N}}\right)^{N\mu-1}\left(1-\mu-\frac{x}{\sqrt{N}}\right)^{N(1-\mu)-1}\frac{dx}{\sqrt{N}}\\
&=\mu^{N\mu-1}(1-\mu)^{N(1-\mu)-1}\left(1+\frac{x}{\sqrt{N}\,\mu}\right)^{N\mu-1}\left(1-\frac{x}{\sqrt{N}(1-\mu)}\right)^{N(1-\mu)-1}\frac{dx}{\sqrt{N}}\\
&=\frac{\mu^{N\mu-1}(1-\mu)^{N(1-\mu)-1}}{\sqrt{N}}\exp\left(-\frac{x^2}{2\mu(1-\mu)}\right)(1+O(n^{-1/2}))\,dx.
\end{aligned}
$$

以上の結果より,

$$
\frac{t^{\alpha-1}(1-t)^{\beta-1}\,dt}{B(\alpha,\beta)}\,dt=\frac{1}{\sqrt{2\pi\mu(1-\mu)}}\exp\left(-\frac{x^2}{2\mu(1-\mu)}\right)(1+O(n^{-1/2}))\,dx.
$$

この公式はベータ分布が上で導入した x 座標では平均 0, 分散 $\mu(1-\mu)$ の正規分布で近似されることを意味している.

解答終

注意: $\alpha=N\mu$, $\beta=N(1-\mu)$, $T\sim\mathrm{Beta}(\alpha,\beta)$ のとき,

$$
E[T]=\frac{\alpha}{\alpha+\beta}=\mu,\quad \mathrm{var}(T)=\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}=\frac{\mu(1-\mu)}{N+1}
$$

なので, $X=\sqrt{N}(T-\mu)$ とおくと, $N\to\infty$ のとき,

$$
E[X]=0,\quad \mathrm{var}(X)=N\,\mathrm{var}(T)=\frac{N}{N+1}\mu(1-\mu)\to\mu(1-\mu).
$$

この $\mu(1-\mu)$ が上の解答例の中で登場した正規分布の分散である.

5 各種の確率分布の正規分布近似

以下に挙げるもの以外にも正規分布で近似される確率分布はたくさんある. 各自, 研究してノートにまとめておくとよいだろう. 具体的な数値例をコンピュータで作っておくと, どのようなときにどれだけ近似がうまく行くかについての「土地勘」が得られる.

In [126]:

```
function plot_normalapprox(dist; title=distname(dist), kwargs...)
    μ, σ = mean(dist), std(dist)
    normal = Normal(μ, σ)
    xlim = (μ - 4.5σ, μ + 4.5σ)
    plot(x -> mypdf(dist, x), xlim...; label="")
    plot!(normal, xlim...; label="normal approx", ls=:dash)
    plot!(; title, kwargs...)
end
```

Out[126]:

```
plot_normalapprox (generic function with 1 method)
```

In [127]:

```julia
# 2022-05-13: Distributions.jl で BetaNegativeBinomial(k, α, β)
# がまだ実装されていないので自前で必要最小限の実装を行う.

module O

using Distributions
using SpecialFunctions

struct BetaNegativeBinomial{T} <: DiscreteUnivariateDistribution
    k::T
    α::T
    β::T
end

function logpdf_betanegbin(x, k, α, β)
    m = round(x)
    m < 0 && return -Inf
    (loggamma(k+m) - loggamma(m+1) - loggamma(k) + logbeta(α+k, β+m) - logbeta(α, β))
end

function Distributions.logpdf(dist::BetaNegativeBinomial, x)
    (; k, α, β) = dist
    logpdf_betanegbin(x, k, α, β)
end

function Distributions.mean(dist::BetaNegativeBinomial)
    (; k, α, β) = dist
    k*β/(α - 1)
end

function Distributions.var(dist::BetaNegativeBinomial)
    (; k, α, β) = dist
    k*(α+k-1)*β*(α+β-1)/((α-1)^2*(α-2))
end

end
```

Out[127]: Main.O

5.1 二項分布の正規分布近似

$0 < p < 1$ でありかつ,

$$(X_1, \ldots, X_n) \sim \mathrm{Bernoulli}(p)^n = \mathrm{Binomial}(1, p)^n$$

であるとき,

$$K_n = X_1 + \cdots + X_n \sim \mathrm{Binomial}(n, p)$$

であり, 中心極限定理によって独立同分布な確率変数達の和の分布が正規分布で近似されることより, n が大きなとき, K_n の分布は正規分布で近似される.

In [128]:

```julia
plot_normalapprox(Binomial(50, 0.2))
```

Out[128]:

5.2 負の二項分布の正規分布近似

$0 < p < 1$ でありかつ,

$$(Y_1, \ldots, Y_k) \sim \mathrm{Geometric}(p) = \mathrm{NegativeBinomial}(1, p)^k$$

であるとき,

$$N_k = Y_1 + \cdots + Y_k \sim \mathrm{NegativeBinomial}(k, p)$$

であり, 中心極限定理によって独立同分布な確率変数達の和の分布が正規分布で近似されることより, k が大きなとき, N_k の分布は正規分布で近似される.

In [129]:
```
plot_normalapprox(NegativeBinomial(50, 0.7))
```

Out[129]:

5.3 Poisson分布の正規分布近似

$\lambda > 0$ でありかつ,

$$(X_1, \ldots, X_n) \sim \mathrm{Poisson}(\lambda/n)^n$$

であるとき,

$$K = X_1 + \cdots + X_n \sim \mathrm{Poisson}(\lambda)$$

であり, 中心極限定理によって独立同分布な確率変数達の和の分布が正規分布で近似されることより, λ が大きなとき, K の分布は正規分布で近似される.

In [130]:
```
plot_normalapprox(Poisson(30))
```

Out[130]:

5.4 ガンマ分布の正規分布近似

$\alpha > 0, \theta > 0$ でありかつ,

$$(Y_1, \ldots, Y_n) \sim \mathrm{Gamma}(\alpha/n, \theta)$$

であるとき,

$$T = Y_1 + \cdots + Y_n \sim \mathrm{Gamma}(\alpha, \theta)$$

であり, 中心極限定理によって独立同分布な確率変数達の和の分布が正規分布で近似されることより, α が大きなとき, T の分布は正規分布で近似される.

In [131]:
```
plot_normalapprox(Gamma(50, 0.1))
```

Out[131]:

5.5 χ²分布の正規分布近似

$\nu > 0$ でありかつ,

$$(Y_1, \ldots, Y_n) \sim \mathrm{Chisq}(\nu/n) = \mathrm{Gamma}(\nu/(2n), 2)$$

であるとき,

$$Y = Y_1 + \cdots + Y_n \sim \mathrm{Chisq}(\nu) = \mathrm{Gamma}(\nu/2, 2)$$

であり, 中心極限定理によって独立同分布な確率変数達の和の分布が正規分布で近似されることより, ν が大きなとき, Y の分布は正規分布で近似される.

In [132]:
```
plot_normalapprox(Chisq(100))
```

Out[132]:

5.6 逆ガンマ分布の正規分布近似

$\alpha > 0$ が大きなとき, 逆ガンマ分布 $\mathrm{InverseGamma}(\alpha, \theta)$ は正規分布で近似される.

In [133]:
```
plot_normalapprox(InverseGamma(100, 100); title="InverseGamma(100, 100)")
```

Out[133]:

5.7 ベータ分布の正規分布近似

すでに上で示したように, $\alpha, \beta > 0$ が大きなとき, ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ は正規分布で近似される.

In [134]:
```
plot_normalapprox(Beta(15, 35))
```

Out[134]:

5.8 t 分布の正規分布近似

「Gauss積分, ガンマ函数, ベータ函数」のノートですでに示したように, $\nu > 0$ が大きなとき, t 分布 $\mathrm{TDist}(\nu)$ は標準正規分布で近似される.

In [135]:
```
plot_normalapprox(TDist(20))
```

Out[135]:

5.9 ベータプライム分布の正規分布近似

$\alpha, \beta > 0$ が大きなとき, ベータプライム分布 $\mathrm{BetaPrime}(\alpha, \beta)$ は正規分布で近似される. (証明に興味がある人は自分で証明してみよ. ベータプライム分布は F 分布のスケール変換に等しい.)

In [136]:
```
plot_normalapprox(BetaPrime(100, 200))
```

Out[136]:

5.10 F 分布の正規分布近似

$\nu_1, \nu_2 > 0$ が大きなとき, F 分布 $\mathrm{FDist}(\nu_1, \nu_2) = (\nu_2/\nu_1)\,\mathrm{BetaPrime}(\nu_1/2, \nu_2/2)$ は正規分布で近似される.

In [137]:
```
plot_normalapprox(FDist(200, 400))
```

Out[137]:

5.11 超幾何分布の正規分布近似

s, f, n が大きなとき, 超幾何分布 $\mathrm{Hypergeometric}(s, f, n)$ は正規分布で近似される.

In [138]:
```
plot_normalapprox(Hypergeometric(40, 60, 80))
```

Out[138]:

5.12 ベータ二項分布の正規分布近似

n, α, β が大きなとき, ベータ二項分布 $\mathrm{BetaBinomial}(n, \alpha, \beta)$ は正規分布で近似される.

In [139]:
```
plot_normalapprox(BetaBinomial(30, 30, 70))
```

Out[139]:

5.13 ベータ負の二項分布の正規分布近似

k, α, β が大きなとき, ベータ二項分布 $\mathrm{BetaNegativeBinomial}(k, \alpha, \beta)$ は正規分布で近似される.

In [140]:

```
plot_normalapprox(O.BetaNegativeBinomial(100, 300, 100);
    title="BetaNegativeBinomial(100, 200, 400)")
```

Out[140]:

In [ ]:

```
```

条件付き確率分布, 尤度, 推定, 記述統計

- 黒木玄

- 2022-05-22〜2022-05-31, 2022-07-22

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 条件付き確率分布
  - 1.1 離散確率分布の条件付き確率分布
  - 1.2 条件付き期待値の基本性質(離散分布版)
  - 1.3 離散分布の条件付き確率分布の簡単な例
  - 1.4 問題: サイコロの出目の数だけコインを投げる場合
  - 1.5 問題: 離散分布の条件付き確率分布として二項分布が得られること
  - 1.6 離散分布の場合のBayesの定理
  - 1.7 2×2の分割表での条件付き確率分布(偽陽性率, 偽陰性率)
  - 1.8 必修の易しい計算問題: 有病率によって偽陽性率と偽陰性率がどのように変化するか
  - 1.9 有病率によって偽陽性率と偽陰性率がどのように変化するかの視覚化
  - 1.10 おまけの問題: モンティ・ホール問題 (Monty Hall problem)
  - 1.11 連続確率分布の条件付き確率分布
  - 1.12 条件付き期待値の基本性質(連続分布版)
  - 1.13 連続分布版のBayesの定理
  - 1.14 問題: 連続分布の条件付き確率分布の例 (分散が固定された正規分布モデルのBayes統計)
  - 1.15 分散が固定された正規分布モデルのBayes統計の事後分布の視覚化
  - 1.16 問題: 2変量の正規分布とその条件付き確率分布の例
  - 1.17 2変量の正規分布とその条件付き確率分布の例の視覚化
- 2 2×2の分割表の分布
  - 2.1 4つのPoisson分布の積
  - 2.2 四項分布
  - 2.3 2つの二項分布の積
  - 2.4 Fisherの非心超幾何分布
  - 2.5 Fisherの非心超幾何分布の正規分布近似の視覚化
  - 2.6 四項分布は4つのPoisson分布の積の条件付き確率分布
  - 2.7 2つの二項分布の積は四項分布は4つの条件付き確率分布
  - 2.8 Fisherの非心超幾何分布は2つの二項分布の積の条件付き確率分布
  - 2.9 独立性条件
  - 2.10 Pearsonのχ^2統計量
- 3 尤度 (ゆうど)と推定
  - 3.1 尤度に関するまとめ
  - 3.2 尤度函数と尤度の定義と諸注意
  - 3.3 例: Bernoulli試行の尤度函数と最尤法
  - 3.4 Bernoulli試行モデルの尤度函数の視覚化
  - 3.5 問題: 尤度が高くても全然もっともらしくない例(オーバーフィッティングの例)
  - 3.6 問題: 正規分布の標本分布モデルの尤度函数と最尤法
  - 3.7 正規分布の標本分布モデルの尤度函数の視覚化
  - 3.8 問題: 最小二乗法がある種の正規分布モデルの最尤法に一致すること
  - 3.9 最尤法の観点から見た十分統計量
    - 3.9.1 Bernoulli試行モデルの十分統計量
    - 3.9.2 正規分布の標本分布モデルの十分統計量
    - 3.9.3 最小二乗法を与える正規分布モデルの尤度函数を記述する統計量

- 4 記述統計 (要約統計)
  - 4.1 n 個の 1 と 0 からなるデータ $x_1, \ldots, x_n$ の要約
  - 4.2 より一般の離散的なデータの要約
  - 4.3 n 個の実数からなるデータ $x_1, \ldots, x_n$ の要約
  - 4.4 n 個の実数の対からなるデータ $(x_1, y_1), \ldots, (x_n, y_n)$ の要約
  - 4.5 Anscombeの例(アンスコムの例, 再)
  - 4.6 問題: DataSaurusの例
    - 4.6.1 Julia言語を使う解答例
    - 4.6.2 Julia言語の使い方
    - 4.6.3 インターネット上の各種計算サイトを使う方法

In [1]:

```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using LinearAlgebra
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:

```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```
safemul(x, y) = x == 0 ? x : x*y
safediv(x, y) = x == 0 ? x : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

1 条件付き確率分布

我々は複雑な現実世界に立ち向かうために, 各種の確率分布を数学的モデル(統計モデル)として採用して, モデルとデータを比較することによって, 現実世界の様子を推測しようとする.

そのときに, 統計モデル内部に現実世界で得たデータの数値を持ち込んで利用する方法として,

- 条件付き確率分布を考えること. 統計モデル(数学的フィクション)内部で現実世界から得たデータと同じ数値が生成された場合に制限した条件付き確率分布を考える.
- 尤度を考えること. 統計モデル(数学的フィクション)内部で現実世界から得たデータと同じ数値が生成される確率またはその密度を考える.

の2つが特に基本的である.

以下では前者の条件付き確率分布の概念について一般的に説明する.

1.1 離散確率分布の条件付き確率分布

離散集合 $\mathscr{X}$ (大文字のグザイ)の中を動く離散変数 x とその確率質量函数 $P(x)$ によって, 離散確率分布 D が与えられているとする. (注意: x は $(x_1, \ldots, x_n)$ のように多変数を1つにまとめて書いたものかもしれない. その場合には Ξ は $\mathbb{R}^n$ の離散部分集合であると考える.)

集合 Ξ 上の函数 $y = f(x)$ が与えられたとき, 確率変数 $Y = f(X)$ が得られる. この Y の函数の期待値は

$$E[g(Y)] = \sum_{x \in \mathscr{X}} g(f(x))P(x)$$

と定義される. (確率変数とはその函数の期待値が定義されているような変数であると言ってよい.)

数値 y について, $P(Y = y) > 0$ であると仮定する.

条件 $Y = y$ (もしくは条件 $f(x) = y$)が定める **条件付き確率分布** (conditional probability distribution) $D|_{Y=y}$ を以下のように定める:

(1) 条件付き確率分布 $D|_{Y=y}$ は離散集合 $\mathscr{X}_y = \{\, x \in \mathscr{X} \mid f(x) = y \,\}$ 上の確率分布である.

(2) 条件付き確率分布 $D|_{Y=y}$ は確率質量函数

$$P(x|y) = \frac{P(x)}{P(Y = y)}, \quad P(Y = y) = \sum_{f(x)=y} P(x) = \sum_{x \in \mathscr{X}_y} P(x)$$

によって定義される. このとき,

$$\sum_{x \in \mathscr{X}_y} P(x|y) = \frac{\sum_{x \in \mathscr{X}_y} P(x)}{P(Y = y)} = 1$$

なので確かにこの函数は「確率の総和が1になる」という条件を満たしており, 確率質量函数になっている.

条件 $Y = y$ という条件のもとでの変数 x に対応する確率変数 X と y の函数 $f(X, y)$ の **条件付き期待値** を次のように定める:

$$E[f(X, Y)|Y = y] = E[f(X, y)|y] = \sum_x f(x, y)p(x|y).$$

これと同じ記号法を一般のパラメータ θ 付きの確率質量函数 $P(x|\theta)$ の場合にも使うことにする:

$$E[f(X, \theta)|\theta] = \sum_x f(x, \theta)P(x|\theta).$$

1.2 条件付き期待値の基本性質(離散分布版)

条件付き期待値 $g(y) = E[f(X, y)|y]$ は y の函数とみなせるので, 確率変数 $g(Y) = E[f(X, Y)|Y]$ が得られる. このとき次が成立することを示せ:

$$E[g(Y)] = E[E[f(X, Y)|Y]] = E[f(X, Y)].$$

解答例: 定義通りに計算しよう:

$$
\begin{aligned}
E[f(X,Y)|Y] = E[g(Y)] &= \sum_y g(y)P(y) \\
&= \sum_y E[f(X,y)|y]P(y) \\
&= \sum_y \left(\sum_x f(x,y)P(x|y) \right) P(y) \\
&= \sum_{x,y} f(x,y)P(x|y)P(y) \\
&= \sum_{x,y} f(x,y)P(x,y) = E[f(X,Y)].
\end{aligned}
$$

要するに x, y の同時確率質量函数が $P(x,y) = P(x|y)P(y)$ と書けることから, この問題の結論が出て来る.

解答終

注意: 測度論的確率論ではこれを一般化して条件付き期待値を定義する(Radon–Nikodýmの定理の話になる).

注意: 個人的な意見では以上のようなシンプルで初等的な例さえ知らずに, いきなり測度論的確率論を勉強しようとするのは不健全な勉強の仕方で好ましくないと思う. 測度論が無用な事柄について多くの意味のある具体例を知った後でそれらを測度論を使って整理し直すようにした方が測度論的確率論についてもスムーズに理解できように思われる.

1.3 離散分布の条件付き確率分布の簡単な例

正二十面体のサイコロの20個の面のうち 6 つの面には1と書いてあり, 5つの面には2と書いてあり, 4つの面には3と書いてあり, 3つの面には4と書いてあり, 残りの2つの面には5と書いてあるとする. そのサイコロをふったときにすべての面は等確率で出るとする. このサイコロの出目の確率分布は

$$
P(1) = \frac{6}{20}, \quad P(2) = \frac{5}{20}, \quad P(3) = \frac{4}{20}, \quad P(4) = \frac{3}{20}, \quad P(5) = \frac{2}{20}
$$

によって定まる集合 $\mathscr{X} = \{1,2,3,4,5\}$ 上の有限離散分布によってモデル化される.

$\mathscr{X}$ 上の函数 $f(x)$ を x が偶数ならば $f(x) = 0$, x が奇数ならば $f(x) = 1$ と定める.

このとき, 確率変数 $Y = f(X)$ に関する条件 $Y = y$ が定める条件付き確率分布を求めてみよう.

この場合の $\mathscr{X}_y = \{\, x \in \mathscr{X} \mid f(x) = y \,\}$ は次のようになる:

$$
\mathscr{X}_0 = \{2,4\}, \quad \mathscr{X}_1 = \{1,2,3\}.
$$

さらに,

$$
P(Y=0) = P(2) + P(4) = \frac{8}{20}, \quad P(Y=1) = P(1) + P(3) + P(5) = \frac{12}{20}
$$

となる. ゆえに, $P(x|0)$ $(x \in \mathscr{X}_0)$ は

$$
\begin{aligned}
P(2|0) &= \frac{P(2)}{P(Y=0)} = \frac{5/20}{8/20} = \frac{5}{8}, \\
P(4|0) &= \frac{P(4)}{P(Y=0)} = \frac{3/20}{8/20} = \frac{3}{8}
\end{aligned}
$$

となり, $P(x|1)$ $(x \in \mathscr{X}_1)$ は

$$
\begin{aligned}
P(1|0) &= \frac{P(1)}{P(Y=1)} = \frac{6/20}{12/20} = \frac{6}{12}, \\
P(2|0) &= \frac{P(3)}{P(Y=1)} = \frac{4/20}{12/20} = \frac{4}{12}, \\
P(3|0) &= \frac{P(5)}{P(Y=1)} = \frac{2/20}{12/20} = \frac{2}{12}.
\end{aligned}
$$

となる. 偶数の目は2,4の2通りで20面のうち8面に2,4と書かれており, そのうちの5面が2で残りの3面が4なので, 偶数の目に制限した場合に2の目が出る条件付き確率は $P(2|0) = 5/8$ になり, 4の目が出る条件付き確率は $P(4|0) = 3/8$ になる. 奇数の目の場合も同様である.

1.4 問題: サイコロの出目の数だけコインを投げる場合

1から6の目が等確率で出るサイコロをふって出た目の数だけコインを投げて表になった枚数のみが報告されるとする. ただし, 各々のコインを投げて表になる確率はちょうど $1/2$ であると仮定し, サイコロの出目は秘密のままになっているとする. 表になったコインの枚数が $k = 0, 1, \ldots, 6$ だと報告されたときの, サイコロの出目の条件付き確率分布の表を作成し, 以下に答えよ:

(1) 表になったコインの枚数は 2 枚だと報告されたとき, サイコロの出目が 5 である条件付き確率をどうなるか?

(2) 表になったコインの枚数は 5 枚だと報告されたとき, サイコロの出目が 5, 6 である条件付き確率はどちらが大きいか?

解答例: サイコロの出目を X とし, X 枚のコインを投げて表になった枚数を Y と書くと,

$$P(X=x, Y=y) = \frac{1}{6}\binom{x}{y}\frac{1}{2^x} \quad (x = 1, 2, \ldots, 6,\ y = 0, 1, \ldots, x)$$

$$P(Y=y) = \sum_{x=y}^{6} P(X=x, Y=y) = \frac{1}{6}\sum_{x=y}^{6}\binom{x}{y}\frac{1}{2^x}$$

$$P(X=x|Y=y) = \frac{P(X=x, Y=y)}{P(Y=y)} = \frac{\binom{x}{y}(1/2^x)}{\sum_{x=y}^{6}\binom{x}{y}(1/2^x)}.$$

これを $y = 0, 1, \ldots, 6$ と $x = y, y+1, \ldots, 6$ について地道計算するとその結果は次のようになる:

	$Y=0$	$Y=1$	$Y=2$	$Y=3$	$Y=4$	$Y=5$	$Y=6$
$X=1$	$32/127$	$16/60$	$0/99$	$0/16$	$0/29$	$0/4$	0
$X=2$	$16/127$	$16/60$	$16/99$	$0/16$	$0/29$	$0/4$	0
$X=3$	$8/127$	$12/60$	$24/99$	$2/16$	$0/29$	$0/4$	0
$X=4$	$4/127$	$8/60$	$24/99$	$4/16$	$4/29$	$0/4$	0
$X=5$	$2/127$	$5/60$	$20/99$	$5/16$	$10/29$	$1/4$	0
$X=6$	$1/127$	$3/60$	$15/99$	$5/16$	$15/29$	$3/4$	1

(1) $Y = 2$ のときの $X = 5$ となる条件付き確率は $20/99$ である.

(2) $Y = 5$ のときの $X = 5, 6$ となる条件付き確率はそれぞれ $1/4, 3/4$ なので, $X = 6$ となる条件付き確率の方が大きい.

解答終

(2)を直確認: (2)は実は $P(X=5, Y=5)$ と $P(X=6, Y=5)$ を比較するだけでよい. なぜならば

$$P(X=5|Y=5) = \frac{P(X=5, Y=5)}{P(Y=5)}, \quad P(X=6|Y=5) = \frac{P(X=6, Y=5)}{P(Y=5)}$$

の分母は共通なので, 分子の大小を比較すれば十分だからである.

$$6P(X=5, Y=5) = \binom{5}{5}\frac{1}{2^5} = \frac{1}{32}, \quad 6P(X=6, Y=5) = \binom{6}{5}\frac{1}{2^6} = \frac{6}{64} = \frac{3}{32}$$

なので, 後者の方が3倍大きい. ゆえに, $Y = 5$ のときの $X = 6$ となる条件付き確率は $X = 5$ となる条件付き確率よりも3倍大きい.

注意: この(2)の結果は, 「表になったコインの枚数は5枚である」という情報が得られたとき, 秘密のままになっているサイコロの出目は 5 ではなく 6 であった可能性が高いことを意味している(確率的に3倍). 新たな情報が得られるたびにまだ見えていない部分がどうなっているかを推測する問題の解決は確率が絡むゲームでは必須である. しかもこのような計算をリアルタイムで処理し続けることは結構大変であり, 強者に勝つためには十分な修練が必要になることが多い.

注意・警告: 非常に残念なことに, このような確率的推測について, 「逆確率」「ベイズ統計」「主観確率」などの用語を持ち出して何か深淵なことをやっているかのように見せる解説が普及してしまっている. 上の解答例を見ればわかるように, 単に条件付き確率分布を計算すればよいだけなので, それらの用語を使った議論は不要である. 条件付き確率の定義を知っていれ定義に忠実に計算できれば「ベイズの定理」さえ不要である. 簡単なことを哲学的にもしくは思想的に深淵であるかのように説明している解説は有害なので相手にしないように注意して欲しい.

In [4]:
```
function condprob_dicecoins(x, y)
    num = binomial(x, y)//2^x
    den = sum(binomial(x, y)//2^x for x in y:6)
    num/den
end
```

Out[4]: condprob_dicecoins (generic function with 1 method)

In [5]:
```
CP = condprob_dicecoins.(1:6, (0:6)')
```

Out[5]:
```
6×7 Matrix{Rational{Int64}}:
 32//127  4//15   0//1   0//1    0//1   0//1  0//1
 16//127  4//15  16//99  0//1    0//1   0//1  0//1
  8//127  1//5    8//33  1//8    0//1   0//1  0//1
  4//127  2//15   8//33  1//4    4//29  0//1  0//1
  2//127  1//12  20//99  5//16  10//29  1//4  0//1
  1//127  1//20   5//33  5//16  15//29  3//4  1//1
```

In [6]:
```
heatmap(0:6, 1:6, CP; xlabel="y", ylabel="x")
```

Out[6]:

1.5 問題: 離散分布の条件付き確率分布として二項分布が得られること

2つの0以上の整数の組 (x, y) 全体の集合 $\mathscr{X} = \mathbb{Z}_{\geq 0}^2 = \{\,(x, y) \mid x, y = 0, 1, 2, \ldots\,\}$ 上の確率質量函数を次のようにPoisson分布の確率質量函数の積で定める($\lambda, \mu > 0$ と仮定する):

$$P(x, y) = e^{-(\lambda+\mu)} \frac{\lambda^x}{x!} \frac{\mu^y}{y!} \quad (x, y = 0, 1, 2, \ldots)$$

集合 $\mathscr{X}$ 上の函数 $f(x, y)$ を $f(x, y) = x + y$ と定める. $N = f(X, Y) = X + Y$ によって確率変数 N を定めることができる(ここで X, Y はいつものように変数 x, y に対応する確率変数である). 確率変数 N の函数の期待値は次のように表される:

$$E[g(N)] = \sum_{(x,y)\in\mathscr{X}} g(x + y) P(x, y).$$

n が 0 以上の整数であるとする. 以下を示せ:

(1) N は平均 $\lambda + \mu$ のPoisson分布に従う.

(2) この場合には, 条件 $N = n$ が定める条件付き確率分布は本質的に二項分布になる.

解答例: $\mathscr{X}_n = \{\,(x, y) \in \mathscr{X} \mid f(x, y) = x + y = n\,\}$, $P(N = n)$, $P(x, y|n) = P(x, y)/P(N = n)$ ($(x, y \in \mathscr{X}_n)$ を順番に計算して行こう. まず,

$$\mathscr{X}_n = \{\,(x, n - x) \mid x = 0, 1, 2, \ldots, n\}$$

であることはすぐにわかる. 次に,

$$\begin{aligned} P(N = n) &= \sum_{(x,y)\in\mathscr{X}_n} P(x, y) = \sum_{x=0}^{n} e^{-(\lambda+\mu)} \frac{\lambda^x}{x!} \frac{\mu^{n-x}}{(n-x)!} \\ &= e^{-(\lambda+\mu)} \frac{1}{n!} \sum_{x=0}^{n} \frac{n!}{x!(n-x)!} \lambda^x \mu^{n-x} = e^{-(\lambda+\mu)} \frac{(\lambda+\mu)^n}{n!}. \end{aligned}$$

これは確率変数 N が平均 $\lambda + \mu$ のPoisson分布 $\mathrm{Poisson}(\lambda + \mu)$ に従うことを意味し, (1)が示された. したがって, $(x, n - x) \in \mathscr{X}_n$ のとき,

$$\begin{aligned} P(x, n - x|n) &= \frac{P(x, n - x)}{P(N = n)} = \frac{e^{-(\lambda+\mu)}(\lambda^x/x!)(\mu^{n-x}/(n-x)!)}{e^{-(\lambda+\mu)}(\lambda+\mu)^n/n!} \\ &= \frac{n!}{x!(n-x)!} \left(\frac{\lambda}{\lambda+\mu}\right)^x \left(\frac{\mu}{\lambda+\mu}\right)^{n-x} \\ &= \frac{n!}{x!(n-x)!} \left(\frac{\lambda}{\lambda+\mu}\right)^x \left(1 - \frac{\lambda}{\lambda+\mu}\right)^{n-x}. \end{aligned}$$

これは条件 $N=n$ が定める条件付き確率分布が本質的に二項分布 $\mathrm{Binomial}(n, \lambda/(\lambda+\mu))$ であることを意味している. これで(2)も示された.

解答終

1.6 離散分布の場合のBayesの定理

条件付き確率分布の概念を理解していれば, Bayesの定理(ベイズの定理)については忘れてもよい. しかし, Bayesの定理という用語が使われる頻度は高いので念のために説明しておくことにする.

離散集合 $\mathscr{X}=\{(x,y)\}$ 上の離散確率分布が確率質量函数 $P(x,y)$ によって定義されているとする. 以下では $P(X=x)$ や $P(Y=y)$ を以下のように書くことにする:

$$P(x)=P(X=x)=\sum_y P(x,y), \quad P(y)=P(Y=y)=\sum_x P(x,y).$$

ここで, 前者の式の $\sum_y$ の y は各 x ごとに $P(x,y)$ が 0 にならないような y 全体を動き, 後者の式の $\sum_x$ の x は各 y ごとに $P(x,y)$ が 0 にならないような x 全体を動く.

記号法上の注意: 慣習に従って $P(x)$ と $P(y)$ の違いを独立変数名 x, y の違いで区別するという記号法を採用している. これは簡潔に書くための手抜きであり, $P_X(x)$ や $P_Y(y)$ のように書けば独立変数名の違いで区別する必要はなくなる. 以下では簡潔さを重視するスタイルを採用したので混乱しないように注意して欲しい. $P(x|y)$ と $P(y|x)$ の違いも独立変数名 x, y の違いで区別することになる.

このとき, y に対応する確率変数 Y によって $Y=y$ という条件で定義される条件付き確率分布の確率質量函数

$$P(x|y)=\frac{P(x,y)}{P(y)}, \quad P(y)=\sum_x P(x,y) \tag{1}$$

だけではなく, x に対応する確率変数 X によって $X=x$ という条件で定義される条件付き確率分布の確率質量函数

$$P(y|x)=\frac{P(x,y)}{P(x)}, \quad P(x)=\sum_y P(x,y) \tag{2}$$

が定義される. このとき

$$\begin{aligned} &P(x,y)=P(x|y)P(y)=P(y|x)P(x), \\ &P(x)=\sum_y P(x,y)=\sum_y P(x|y)P(y), \\ &P(y)=\sum_x P(x,y)=\sum_x P(y|x)P(x) \end{aligned} \tag{3}$$

なので,

$$\begin{aligned} P(y|x)&=\frac{P(x|y)P(y)}{\sum_y P(x|y)P(y)}, \\ P(x|y)&=\frac{P(y|x)P(x)}{\sum_x P(y|x)P(x)}. \end{aligned} \tag{B}$$

この結果(B)を **Bayesの定理** (ベイズの定理)と呼ぶ.

しかし, 以上の議論を見ればわかるように, Bayesの定理は条件付き確率分布の定義を書き直したものに過ぎず, 価値ある新しい考え方が得られるような結果ではない. しかし, 具体的もしくは理論的な計算で条件付き確率の概念が(B)の形式で使われることがあることは知っておいて損がないと思われるので, このように紹介することにした.

注意: 多くの初学者はBayesの定理を「与えられた公式」として直接使う計算を直観的に理解できないことをやってしまっているという理由でよく間違ってしまうようだ. そのようになってしまいそうな人は無理して(B)の公式を直接使おうとせずに, (1),(2)(および(3))まで戻って地道に計算するようにした方がよいだろう.

注意: 以上のようにBayesの定理は条件付き確率の定義から自明に導かれる無理して使う必要がない公式に過ぎない. そのような自明でつまらない結果について, 「逆確率」「主観確率」のような用語を用いて特別な価値があるかのように説明するスタイルが伝統的になってしまっているので惑わされないように注意が必要である. Bayesの定理は, 条件付き確率の定義を書き直しただけの, 「逆確率」「主観確率」のような用語と無関係に一般的かつ普遍的に成立している自明な数学的定理に過ぎない.

注意: 所謂Bayes統計についても条件付き確率分布の概念まで戻って考えればBayesの定理を使わずに理解可能である.

1.7 2×2の分割表での条件付き確率分布(偽陽性率, 偽陰性率)

病気Dに罹っているいるかどうかに関するある検査法を使うと,

- 病気Dに罹っている人は 75% の確率で陽性だと判定でき,
- 病気Dに罹っていない人は 95% の確率で陰性と判定できるもの

と仮定する. このとき, この検査法の **感度** は 75% であり, **特異度** は 95% であるという.

この検査を病気Dに罹っている確率が p の人(**有病率** p の人)に適用する状況について考える. このとき, その人が病気Dに罹っているか否かと検査結果が陽性か陰性かで分類することによって, 次のような 2×2 の確率の表を作れる:

	病気有	病気無	
陽性	$0.75p$	$0.05(1-p)$	$0.05+0.70p$
陰性	$0.25p$	$0.95(1-p)$	$0.95-0.70p$
	p	$1-p$	1

下段の $p, 1-p$ や右端の $0.05+0.70p, 0.95-0.70p$ は確率の縦もしくは横の合計である. それらをよく **マージン** (margin, 周辺確率)と呼ぶ.

この確率の2×2の表の部分に対応する確率質量函数は以下のように書ける:

$$\begin{aligned} P(\text{陽性}, \text{病気有}) &= 0.75p, & P(\text{陽性}, \text{病気無}) &= 0.05(1-p), \\ P(\text{陰性}, \text{病気有}) &= 0.25p, & P(\text{陰性}, \text{病気無}) &= 0.95(1-p). \end{aligned}$$

さらに, マージンとして以下の確率も上の表にすでに書き込まれている:

$$\begin{aligned} P(\text{病気有}) &= P(\text{陽性}, \text{病気有}) + P(\text{陰性}, \text{病気有}) = p, \\ P(\text{病気無}) &= P(\text{陽性}, \text{病気無}) + P(\text{陰性}, \text{病気無}) = 1-p, \\ P(\text{陽性}) &= P(\text{陽性}, \text{病気有}) + P(\text{陽性}, \text{病気無}) = 0.05+0.70p, \\ P(\text{陰性}) &= P(\text{陰性}, \text{病気有}) + P(\text{陰性}, \text{病気無}) = 0.95-0.70p. \end{aligned}$$

上の表から, 陽性または陰性であるという条件によって定まる病気の有無に関する条件付き確率が以下のように計算される:

$$\begin{aligned} (\text{陽性的中率}) &:= P(\text{病気有}|\text{陽性}) = \frac{P(\text{陽性}, \text{病気有})}{P(\text{陽性})} = \frac{0.75p}{0.05+0.70p}, \\ (\text{偽陽性率}) &:= P(\text{病気無}|\text{陽性}) = \frac{P(\text{陽性}, \text{病気無})}{P(\text{陽性})} = \frac{0.05(1-p)}{0.05+0.70p}, \\ (\text{偽陰性率}) &:= P(\text{病気有}|\text{陰性}) = \frac{P(\text{陰性}, \text{病気有})}{P(\text{陰性})} = \frac{0.25p}{0.95-0.70p}, \\ (\text{陰性的中率}) &:= P(\text{病気無}|\text{陰性}) = \frac{P(\text{陰性}, \text{病気無})}{P(\text{陰性})} = \frac{0.95(1-p)}{0.95-0.70p}. \end{aligned}$$

これらの式は, 私には見難く, 上の確率の表を直接見た方がわかり易いように感じられる. しかし, 議論の内容を正確に把握できるようにするためにあえて式も書いてみた.

注意: 伝統的には以上の内容はBayesの定理の応用として説明されることが多い. しかし, 上の説明ではBayesの定理は一切使用する必要がなかった. このことからもBayesの定理が必須ではないことがわかる. 最後に得られた陽性的中立, 偽陽性率, 偽陰性率, 陰性的中率の公式達はちょうどBayesの定理の形をしている. そして, すぐ上でも述べたように見易い結果ではない. 実際の計算ではそれらのBayesの定理を使っても得られる公式を使うよりも, 確率の表を地道に書いて, 表を見ながら直接計算した方が間違う可能性も減り, 直観的な意味も理解し易いだろう.

以上を踏まえて次の問題を解け.

1.8 必修の易しい計算問題: 有病率によって偽陽性率と偽陰性率がどのように変化するか

前節の状況において, 有病率 p が $5\%, 10\%, 20\%, 40\%, 80\%$ の場合の偽陽性率と偽陰性率を求めよ.

解答例: 前節で求めた偽陽性率と偽陰性率の公式に $p = 0.05, 0.1, 0.2, 0.4, 0.8$ を代入すれば求まるが, 以下では地道に確率の表を書いて求めてみよう. (前節で求めた公式を使った人は以下の地道なやり方を採用しても手間がそう増えず, 状況をより把握し易くなっていることを確認して欲しい.)

前節の確率の表を引用しよう:

	病気有	病気無	
陽性	$0.75p$	$0.05(1-p)$	$0.05+0.70p$
陰性	$0.25p$	$0.95(1-p)$	$0.95-0.70p$
	p	$1-p$	1

$p = 0.05$ の場合:

	病気有	病気無	
陽性	3.75%	4.75%	8.5%
陰性	1.25%	90.25%	91.5%
	5%	95%	100%

$p = 0.1$ の場合:

	病気有	病気無	
陽性	7.5%	4.5%	12%
陰性	2.5%	85.5%	88%
	10%	90%	100%

$p = 0.2$ の場合:

	病気有	病気無	
陽性	15%	4%	19%
陰性	5%	76%	81%
	20%	80%	100%

$p = 0.4$ の場合:

	病気有	病気無	
陽性	30%	3%	33%
陰性	10%	57%	67%
	40%	60%	100%

$p = 0.8$ の場合:

	病気有	病気無	
陽性	60%	1%	61%
陰性	20%	19%	39%
	80%	20%	100%

以上の表を眺めると, 偽陽性率(陽性になった場合の病気無の条件付き確率)と偽陰性率(陰性になった場合の病気有の条件付き確率)は以下の表のようになることがわかる:

有病率	偽陽性率	偽陰性率
5%	$4.75/8.5 \approx 56\%$	$1.25/91.5 \approx 1.4\%$
10%	$4.5/12 \approx 38\%$	$2.5/88 \approx 2.8\%$
20%	$4/19 \approx 21\%$	$5/81 \approx 6.2\%$
30%	$3/33 \approx 9.1\%$	$10/67 \approx 15\%$
40%	$1/61 \approx 1.6\%$	$20/39 \approx 51\%$

有病率が低いと偽陽性率が高くなり, 有病率が高いと偽陰性率が高くなる.

解答終

In [7]:
```julia
# 有病率 = prevalence rate
# 感度 = sensitivity
# 特異度 = specificity
# 偽陽性 = false-positive
# 偽陰性 = false-negative
function falsepositive_falsenegative(prevalence_rate, sensitivity, specificity)
    a = positive_ill     = sensitivity       * prevalence_rate
    b = negative_ill     = (1 - sensitivity) * prevalence_rate
    c = positive_healthy = (1 - specificity) * (1 - prevalence_rate)
    d = negative_healthy = specificity       * (1 - prevalence_rate)
    positive = positive_ill + positive_healthy
    negative = negative_ill + negative_healthy
    falsepositive = positive_healthy / positive
    falsenegative = negative_ill     / negative
    falsepositive, falsenegative
end
```

Out[7]:
```
falsepositive_falsenegative (generic function with 1 method)
```

In [8]:
```julia
falsepositive_falsenegative.([0.05, 0.1, 0.2, 0.4, 0.8], 0.75, 0.95) .|>
x -> round.(x; digits=4)
```

Out[8]:
```
5-element Vector{Tuple{Float64, Float64}}:
 (0.5588, 0.0137)
 (0.375, 0.0284)
 (0.2105, 0.0617)
 (0.0909, 0.1493)
 (0.0164, 0.5128)
```

1.9 有病率によって偽陽性率と偽陰性率がどのように変化するかの視覚化

In [9]:
```julia
function plot_falsepositive(prevalence_rate, sensitivity, specificity;
        xsize = 20, ysize = xsize, ms = 5, size = (480, 300), kwargs...)
    N = xsize * ysize
    a = positive_besick  = sensitivity       * prevalence_rate
    b = negative_besick  = (1 - sensitivity) * prevalence_rate
    c = positive_notsick = (1 - specificity) * (1 - prevalence_rate)
    d = negative_notsick = specificity       * (1 - prevalence_rate)
    A = round(Int, a*N)
    B = round(Int, b*N)
    C = round(Int, c*N)
    D = round(Int, d*N)
    xsize_besick  = (A + B + ysize - 1) ÷ ysize
    xsize_notsick = (C + D + ysize - 1) ÷ ysize
    TP = [(mod(i, xsize_besick), -(i ÷ xsize_besick)) for i in 0:A-1]
    FN = [(mod(i, xsize_besick), -(i ÷ xsize_besick)) for i in A:A+B-1]
    FP = [(xsize_besick + mod(i, xsize_notsick), -(i ÷ xsize_notsick)) for i in 0:C-1]
    TN = [(xsize_besick + mod(i, xsize_notsick), -(i ÷ xsize_notsick)) for i in C:C+D-1]
    P = plot(; frame=false, axis=false, tick=false, margin=0Plots.mm)
    scatter!(TP; m=:utriangle, msc=:red, mc=:white, label="true-positive", ms)
    scatter!(FN; m=:diamond, msc=:red, mc=:white, label="false-negative", ms, msw=3)
    scatter!(FP; m=:dtriangle, msc=:blue, mc=:white, label="false-positive", ms, msw=3)
    scatter!(TN; m=:square, msc=:blue, mc=:white, label="true-negative", ms)
    plot!(; legend = :outertopright)
    plot!(; xlim = (-0.5, xsize+0.5), ylim = (-ysize-0.5, 0.5))
    plot!(; size, kwargs...)
 end
```

Out[9]:
```
plot_falsepositive (generic function with 1 method)
```

以下は, 感度(病気有の人が検査陽性になる確率)が 75% で特異度(病気無の人が検査陽性になる確率)が 95% の場合.

赤色は病気有を, 青色は病気無を意味する. すなわち, 赤い上向き三角と赤いひし形は病気Dに罹っている人で, 青の下向き三角と青の正方形は病気Dに罹っていない人である.

三角は検査陽性を意味し, 四角は検査陰性を意味する. すなわち, 赤い上向き三角と青の下向き三角は検査で陽性になった人達で, それ以外のひし形と正方形は陰性になった人達である.

青い下向き三角 は病気Dに罹っていないのに検査で陽性になってしまったので **偽陽性** (**false-positive**). 目立つように太くしてある. (青い正方形は病気有でかつ検査陽性なので真陽性 (true-negative).)

赤いひし形 は病気Dに罹っているのに検査で陰性になってしまったので **偽陰性** (**false-negative**). これも目立つように太くしてある. (青い正方形は病気無でかつ検査陰性なので真陰性 (true-negative).)

以下の図を見れば, 有病率(赤色の割合=赤い上向き三角と赤いひし形を合わせた割合)を変えたときにどのように偽陽性と偽陰性の割合が変化するかが分かる.

In [10]:
```
plot_falsepositive(0.05, 0.75, 0.95)
```

Out[10]:

In [11]:
```
plot_falsepositive(0.10, 0.75, 0.95)
```

Out[11]:

In [12]:
```
plot_falsepositive(0.20, 0.75, 0.95)
```

Out[12]:

In [13]:
```
plot_falsepositive(0.40, 0.75, 0.95)
```

Out[13]:

In [14]:
```
plot_falsepositive(0.80, 0.75, 0.95)
```

Out[14]:

感度(病気有の人が検査陽性になる確率)を 75% から 95% に上げて, 特異度(病気無の人が検査陽性になる確率)が 75% に下がった場合. 当たり前のことだが, 偽陽性(青い下向き三角)が大幅に増える. 検査対象の有病率を 20% まで上げても, 検査陽性者のうち半分以上が病気無になってしまう.

In [15]:
```
plot_falsepositive(0.05, 0.95, 0.75)
```

Out[15]:

In [16]:
```
plot_falsepositive(0.10, 0.95, 0.75)
```

Out[16]:

In [17]:
```
plot_falsepositive(0.20, 0.95, 0.75)
```

Out[17]:

In [18]:
```
plot_falsepositive(0.40, 0.95, 0.75)
```

Out[18]:

In [19]:
```
plot_falsepositive(0.80, 0.95, 0.75)
```

Out[19]:

1.10 おまけの問題: モンティ・ホール問題 (Monty Hall problem)

以下のようなゲームを考える:

(1) 親は $1, 2, 3$ の中から番号 X を無作為に選んで秘密にしておく.

(2) あなたは, $1, 2, 3$ の中から番号 1 を選んでそれを親に告げる.

(3) 親は $1, 2, 3$ の中から 1 と X 以外の番号 Y を無作為に選んで, あなたに告げる.

(4) あなたは $1, 2, 3$ の中から $1, Y$ を除いて残った1つの番号 Z を選ぶ.

あなたが最後に選んだ番号 Z が秘密の番号 X に一致する確率を求めよ.

注意: (2)で 1 を選ばずに $1, 2, 3$ のどれかを無作為に選んでも結果は同じになる.

代表的誤答例: $Z = X$ となる確率は $1/2$ である. **誤答終**

解答例1: すべての場合の確率を書き下す方法. 確率 $P(X = x, Y = y)$ 達を表にすると以下のようになる:

	$Y=1$	$Y=2$	$Y=3$	
$X=1$	0	$1/6$	$1/6$	$1/3$
$X=2$	0	0	$1/3$	$1/3$
$X=3$	0	$1/3$	0	$1/3$
	0	$1/2$	$1/2$	

Y として 1 と X 以外の番号を選ぶので, $Y=1$ の確率は 0 になる. $X=1$ となる確率は $1/3$ であり, そのとき, 1 と $X=1$ 以外の番号は $2,3$ の2つで, それらを無作為に選ぶのでそれぞれに確率は $1/6$ になる. $X=2$ の場合には常に $Y=3$ となり, $X=3$ の場合には常に $Y=2$ となり, それぞれの確率は $1/3$ になる. これを表にすると上のようになる.

- $X=1, Y=2$ のとき $Z=3$ となり, $Z \neq X$ となる.
- $X=1, Y=3$ のとき $Z=2$ となり, $Z \neq X$ となる.
- $X=2, Y=3$ のとき $Z=2$ となり, $Z=X$ となる.
- $X=3, Y=2$ のとき $Z=3$ となり, $Z=X$ となる.

ゆえに $P(Z=X)=P(X=2,Y=3)+P(X=3,Y=2)=1/3+1/3=2/3$.

解答終

解答例2: あなたが最後に選んだ番号 Z が X に一致しないことと, あなたが最初に選んだ番号 1 が X に一致することは同値であり, $X=1$ となる確率は $1/3$ である. ゆえに, その補事象である $Z=X$ となる確率は $2/3$ である.

要するに, 最初に選んだ数字が運悪く X に一致していなければ, 最後に選び直して得た Z は X に一致する. たったこれだけの話である.

解答終

注意: 単に確率を計算すればよいだけの問題である. 「主観確率」「ベイズ確率」のような用語を持ち出す説明を読むときには, 本当は易しい話を無駄に難しく考えて, おかしな考え方に陥っていることを疑った方がよい. 歴史の偶然のせいで世間的に広まってしまった考え方の多くが間違っている.

1.11 連続確率分布の条件付き確率分布

連続変数 (x,y) の確率密度函数 $p(x,y)$ が定める連続分布 D を考え, 変数 y に対応する確率変数を Y と書く:

$$E[f(Y)]=\iint f(y)p(x,y)\,dx\,dy=\int f(y)\left(\int p(x,y)\,dx\right)dy.$$

これより, 確率変数 Y が従う分布は確率密度函数

$$p(y)=\int p(x,y)\,dx$$

を持つことがわかる. 条件 $Y=y$ が定める x に関する **条件付き確率分布** の密度函数 $p(x|y)$ を次のように定める:

$$p(x|y)=\frac{p(x,y)}{p(y)}.$$

このとき,

$$\int p(x|y)\,dx=\frac{\int p(x,y)\,dx}{p(y)}=1$$

より, $p(x|y)$ がパラメータ y を持つ x の確率密度函数になっていることがわかる.

条件 $Y=y$ という条件のもとでの変数 x に対応する確率変数 X と y 付きの函数 $f(X,y)$ の **条件付き期待値** を次のように定める:

$$E[f(X,Y)|Y=y]=E[f(X,y)|y]=\int f(x,y)p(x|y)\,dx.$$

これと同じ記号法を一般のパラメータ θ 付きの確率密度函数 $p(x|\theta)$ の場合にも使うことにする:

$$E[f(X,\theta)|\theta]=\int f(x,\theta)p(x|\theta)\,dx.$$

注意: 条件付き確率分布の密度函数の定義について.

$p(x|y)$ の定義が確率密度の比になっていることがわかりにくいと感じる人は微小な dx, dy をかけて確率とみなせるようにし，$p(x|y)\,dx$ を

$$p(x|y)\,dx = \frac{p(x,y)\,dx\,dy}{p(y)\,dy}$$

によって定義していると思っても良い，

$$\frac{(\text{図の } dx\,dy \text{ に } (x,y) \text{ が含まれる確率})}{(\text{図の } dy \text{ に } y \text{ が含まれる確率})} = \frac{p(x,y)\,dx\,dy}{p(y)\,dy}$$

$$\| \qquad\qquad\qquad \|$$

$$\left(\begin{array}{l} y \text{ が } dy \text{ に含まれるという条件が定める} \\ \text{条件付き確率分布で} \\ x \text{ が } dx \text{ に含まれる確率} \end{array}\right) = p(x|y)\,dx$$

1.12 条件付き期待値の基本性質(連続分布版)

条件付き期待値 $g(y) = E[f(X,y)|y]$ は y の函数とみなせるので, 確率変数 $E[f(X,Y)|Y] = g(Y)$ が得られる. このとき次が成立することを示せ:

$$E[g(Y)] = E[E[f(X,Y)|Y]] = E[f(X,Y)].$$

解答例: 定義通りに計算しよう:

$$\begin{aligned} E[f(X,Y)|Y] = E[g(Y)] &= \int g(y)p(y)\,dy \\ &= \int E[f(X,y)|y]p(y)\,dy \\ &= \int \left(\int f(x,y)p(x|y)\,dx \right) p(y)\,dy \\ &= \iint f(x,y)p(x|y)p(y)\,dx\,dy \\ &= \iint f(x,y)p(x,y)\,dx\,dy = E[f(X,Y)]. \end{aligned}$$

要するに x, y の同時確率密度函数が $p(x,y) = p(x|y)p(y)$ と書けることから, この問題の結論が出て来る.

解答終

注意: 測度論的確率論ではこれを一般化して条件付き期待値を定義する.

1.13 連続分布版のBayesの定理

以下では $\int \cdots dx$ は定積分を意味するとする.

連続変数 (x,y) の確率密度函数 $P(x,y)$ によって連続確率分布が定義されているとする. 以下では, 変数 x, y に対応する確率変数達 X, Y それぞれ単独の確率密度函数を $p(x), p(y)$ と書くことにする:

$$p(x) = \int p(x,y)\,dy \quad p(y) = \int p(x,y)\,dx.$$

確率密度函数 $p(x)$,p(y)のそれぞれが定めるx,y$ の分布を **周辺分布** (marginal distribution)と呼ぶことがある.

このとき, y に対応する確率変数 Y によって $Y = y$ という条件で定義される条件付き確率分布の確率密度函数

$$p(x|y) = \frac{p(x,y)}{p(y)}, \quad p(y) = \int p(x,y)\,dx \tag{1}$$

だけではなく, x に対応する確率変数 X によって $X = x$ という条件で定義される条件付き確率分布の確率密度函数

$$p(y|x) = \frac{p(x,y)}{p(x)}, \quad p(x) = \int P(x,y)\,dy \tag{2}$$

が定義される. このとき

$$\begin{aligned} p(x,y) &= p(x|y)p(y) = p(y|x)p(x), \\ p(x) &= \int p(x,y)\,dy = \int p(x|y)p(y)\,dy, \\ p(y) &= \int p(x,y)\,dx = \int p(y|x)p(x)\,dx \end{aligned} \tag{3}$$

なので,

$$\begin{aligned} p(y|x) &= \frac{p(x|y)p(y)}{\int p(x|y)p(y)\,dy}, \\ p(x|y) &= \frac{p(y|x)p(x)}{\int p(y|x)p(x)\,dx}. \end{aligned} \tag{B}$$

この結果(B)を **Bayesの定理** (ベイズの定理)と呼ぶ.

しかし, 以上の議論を見ればわかるように, Bayesの定理は条件付き確率分布の定義を書き直したものに過ぎず, 価値ある新しい考え方が得られるような結果ではない. しかし, 具体的もしくは理論的な計算で条件付き確率の概念が(B)の形式で使われることがあることは知っておいて損がないと思われるので, このように紹介することにした.

記号法上の注意1: 慣習に従って $p(x)$ と $p(y)$ の違いを独立変数名 x, y の違いで区別するという記号法を採用した. これは簡潔に書くための手抜きであり, $p_X(x)$ や $p_Y(y)$ のように書けば独立変数名の違いで区別する必要はなくなる. 以上では簡潔さを重視するスタイルを採用したので混乱しないように注意して欲しい. $p(x|y)$ と $p(y|x)$ の違いも独立変数名 x, y の違いで区別することになった.

記号法上の注意2: $p(y|x) = \dfrac{p(x|y)p(y)}{\int p(x|y)p(y)\,dy}$ における分子の y は独立変数だが, 分母の y は積分変数であり, 異なる意味を持っている. 違うものを同じ記号で書いている. $\int p(x|y)p(y)\,dy$ は $\int p(x|y')p(y')\,dy'$ と書いた方が混乱を防ぎ易かったかもしれない. しかし, そのようにすると, 変数名を y の1個から y, y' の2個に増やす必要があり, 式の煩雑さが増すので, あえて同じ記号で書くことにした. 混乱しないように注意して欲しい.

注意: Bayes統計ではパラメータ θ 付きの確率密度函数 $p(x|\theta)$ の他にパラメータ θ の確率密度函数 $p(\theta)$ も用意して, x と θ の同時確率密度函数を $p(x,\theta) = p(x|\theta)p(\theta)$ と定義する. この状況のもとでは, $X = x$ という条件で定義される θ の条件付き確率分布 $p(\theta|x)$ は次のように表される:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(\theta)\,d\theta}.$$

Bayes統計の文脈では, $p(\theta)$ は **事前分布** (prior)と呼ばれ, $p(x|\theta)$ は **事後分布** (posterior)と呼ばれる. Bayes統計では新たな道具として事前分布 $p(\theta)$ を利用できるお陰で, それ抜きでは困難な統計分析も可能になる. ただし, Bayes統計がそういう優れた性質を持っている理由は「主観確率」や「ベイズ主義」のような概念抜きに語ることができる. この点に関する詳しい説明は次の教科書にある:

- 渡辺澄夫, ベイズ統計の理論と方法, 2012年, コロナ社, 226ページ.

さらに, 次の論説も参考になる:

- 赤池弘次, 統計的推論のパラダイムの変遷について, 統計数理研究所彙報, 1980年第27巻第1号, pp. 5-12. link (https://ismrepo.ism.ac.jp/index.php?active_action=repository_view_main_item_detail&page_id=13&block_id=21&item_id=32568&item_no=1)

1.14 問題: 連続分布の条件付き確率分布の例 (分散が固定された正規分布モデルのBayes統計)

この節の式には細かい誤りが多数含まれている可能性がある. 誤りを見付けたら自分で訂正すること. 誤りがなかったら, 拍手喝采して欲しい.

$n, \rho > 0$ を任意に取って固定する. $x, \mu \in \mathbb{R}$ の同時確率密度函数

$$p(x,\mu|n,\rho) = \frac{e^{-n(x-\mu)^2/2}}{\sqrt{2\pi/n}}\,\frac{e^{-\mu^2/(2\rho^2)}}{\sqrt{2\pi\rho^2}}.$$

によって定義される $(x,\mu) \in \mathbb{R}^2$ の確率分布を考える. その分布において, 変数 x に対応する確率変数 X に関する条件 $X = x$ で定義された μ に関する条件付き確率分布の密度函数が

$$p(\mu|x,n,\rho) = \frac{1}{\sqrt{2\pi\rho^2/(1+n\rho^2)}}\exp\left(-\frac{1+n\rho^2}{2\rho^2}\left(\mu - \frac{n\rho^2 x}{1+n\rho^2}\right)^2\right).$$

になることを示せ. これは平均と分散がそれぞれ

$$\frac{n\rho^2}{1+n\rho^2}x, \quad \frac{\rho^2}{1+n\rho^2}$$

の正規分布である.

注意: 特に, $n \to \infty$ とすると, 条件 $X = x$ で定義された μ の条件付き確率分布の期待値は x に収束し, 分散は 0 に収束するので, μ の条件付き確率分布は x の近くに集中するようになる.

注意: この問題の例は正規分布モデル $p(x|n,\mu) = e^{-n(x-\mu)^2/2}/\sqrt{2\pi/n}$ と事前分布 $p(\mu|\rho) = e^{-\mu^2/(2\rho^2)}/\sqrt{2\pi\rho^2}$ に関するBayes統計におけるデータの数値 x が定める事後分布 $p(\mu|x,n,\rho)$ を求める計算になっている. x は標本平均に, n はサンプルサイズに対応しており, $p(x|n,\mu)$ の形は分散 1 の正規分布のサイズ n 標本分布のおいて標本平均の分散が $1/n$ になることを表している. だから, この場合には固定された分散 1 を持つ正規分布の標本分布を統計モデルとして採用している場合になっていると考えて欲しい. $n = 0$ のとき事後分布が事前分布に一致することにも注意せよ.

注意: 条件付き確率分布の概念はBayes統計に限らず統計学における最も基本的かつ重要な確率論の道具であるが, 特にBayes統計では条件付き確率分布を使った推論が系統的に利用される.

解答例: X 単独の密度函数 $p(x|\rho)$ を求めよう.

$$\begin{aligned} n(x-\mu)^2 + \frac{\mu^2}{\rho^2} &= \frac{1+n\rho^2}{\rho^2}\mu^2 - 2nx\mu + nx^2 \\ &= \frac{1+n\rho^2}{\rho^2}\left(\mu - \frac{n\rho^2 x}{1+n\rho^2}\right)^2 + \frac{nx^2}{1+n\rho^2} \end{aligned} \tag{$*$}$$

より,

$$\begin{aligned} p(x|n,\rho) &= \int_{\mathbb{R}} p(x,\mu|n,\rho)\,d\mu \\ &= \frac{\exp(-nx^2/(2(1+n\rho^2)))}{\sqrt{2\pi/n}\,\sqrt{2\pi\rho^2}}\int_{\mathbb{R}}\exp\left(-\frac{1+n\rho^2}{2\rho^2}\left(\mu - \frac{n\rho^2 x}{1+n\rho^2}\right)^2\right)d\mu \\ &= \frac{\exp(-nx^2/(2(1+n\rho^2)))}{\sqrt{2\pi/n}\,\sqrt{2\pi\rho^2}}\sqrt{\frac{2\pi\rho^2}{1+n\rho^2}} \\ &= \frac{1}{\sqrt{2\pi(1+n\rho^2)/n}}\exp\left(-\frac{x^2}{2(1+n\rho^2)/n}\right). \end{aligned}$$

このように X 単独の分布は平均 0, 分散 $(1+\rho^2)/n$ の正規分布になる.

ゆえに, 条件 $X = x$ で定義される μ の分布の確率密度函数 $p(\mu|x,n,\rho)$ は以下のようになる. すなわち,

$$n(x-\mu)^2 + \frac{\mu^2}{\rho^2} - \frac{nx^2}{1+n\rho^2} = \frac{1+n\rho^2}{\rho^2}\left(\mu - \frac{n\rho^2 x}{1+n\rho^2}\right)^2$$

を使うと,

$$p(\mu|x,n,\rho) = \frac{p(x,\mu|n,\rho)}{p(x|n,\rho)} = \frac{1}{\sqrt{2\pi\rho^2/(1+n\rho^2)}}\exp\left(-\frac{1+n\rho^2}{2\rho^2}\left(\mu - \frac{n\rho^2 x}{1+n\rho^2}\right)^2\right).$$

解答終

注意: 本当は最初に公式($*$)を得た時点ですべての答えが出ていたと考えられる. 以上では丁寧だが無駄な詳細を含む計算を納得感を高めるためにあえてやってみせたが, 実践的な計算においてはそのように考えるべきである. 以下においても同様に無駄な詳細を含む計算をやってみせる場合があるので注意して欲しい. (要するにこのノートの計算の仕方を実践的な計算で真似しないで欲しいということ.)

In [20]:

```
@vars n ρ positive=true
@vars x μ

expr1 = n*(x - μ)^2 + μ^2/ρ^2
expr2 = (1+n*ρ^2)/ρ^2 * (μ - n*ρ^2/(1+n*ρ^2)*x)^2
expr3 = (expr1 - expr2).factor()
[
    expr1
    expr2
    expr3
]
```

Out[20]:

$$\left[\begin{array}{c} n(x-\mu)^2+\frac{\mu^2}{\rho^2} \\ \frac{(n\rho^2+1)\left(-\frac{nx\rho^2}{n\rho^2+1}+\mu\right)^2}{\rho^2} \\ \frac{nx^2}{n\rho^2+1} \end{array}\right]$$

1.15 分散が固定された正規分布モデルのBayes統計の事後分布の視覚化

$\mu_0 = 4$, $\rho = 1/2$ とおく.

平均 $\mu_0 = 4$, 分散 1 の正規分布のサイズ $N = 2^{10}$ の標本 $X_1, X_2, \ldots, X_N$ をランダムに生成し, $n = 0, 1, 2, 4, 8, \ldots, 1024 = 2^{10}$ について標本平均 $x = \bar{X}_n = (X_1 + \cdots + X_n)/n$ を求め, 前節で求めた μ の事後分布 $p(\mu|x, n, \rho)$ をプロットしてみよう.

n を大きくすると μ の事後分布はサンプルを生成した分布の平均値 μ_0 に集中して行く.

これはBayes統計の漸近論的な基礎付けの最も簡単な場合になっている.

In [21]:

```julia
Random.seed!(4649373) # よろしくみなさん

μ₀ = 4
ρ = 1/2
N = 2^10
X = rand(Normal(μ₀, 1), N) # 標本生成
PP = []
prior = Normal(0, ρ)
P = plot(prior, μ₀-6, μ₀+4.5; label="prior", xtick=-10:10)
push!(PP, P)
for k in 0:10
    n = 2^k
    x = mean(@view X[1:n])
    μ_posterior = n*ρ^2*x/(1 + n*ρ^2)
    σ_posterior = √(ρ^2/(1+n*ρ^2))
    posterior = Normal(μ_posterior, σ_posterior)
    P = plot(posterior, μ₀-6, μ₀+4.5; label="n=$n", xtick=-10:10)
    push!(PP, P)
end
plot(PP...; size=(800, 500), layout=(4, 3))
```

Out[21]:

1.16 問題: 2変量の正規分布とその条件付き確率分布の例

$$\Sigma = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

は固有値がすべて正の実対称行列であるとし, $\mu = (\mu_x, \mu_y) \in \mathbb{R}^2$ であるとする. このとき,

$$\Sigma^{-1} = \frac{1}{ac - b^2} \begin{bmatrix} c & -b \\ -b & a \end{bmatrix}$$

であり, $(x, y) \in \mathbb{R}^2$ の確率密度函数を

$$\begin{aligned} p(x, y) &= \frac{1}{\det(2\pi\Sigma)^{1/2}} \exp\left(-\frac{1}{2} [x - \mu_x,\ y - \mu_y] \Sigma^{-1} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix} \right) \\ &= \frac{1}{\sqrt{(2\pi)^2 (ac - b^2)}} \exp\left(-\frac{a(y - \mu_y)^2 - 2b(x - \mu_x)(y - \mu_y) + c(x - \mu_x)^2}{2(ac - b^2)} \right) \end{aligned}$$

と定めることができる. これが $\iint_{\mathbb{R}^2} p(x, y)\, dx\, dy = 1$ を満たすことは, Σ を直交行列で対角すれば示せる. (詳細は略す. 以下では認めて使ってよい.) この確率密度函数が定める確率分布を **2変量正規分布** と呼び,

$$\mathrm{MvNormal}(\mu, \Sigma)$$

と表すことにする. μ を平均(もしくは期待値)と呼び, Σ を **分散共分散行列** (variance-covariance matrix)と呼ぶ. (この記号法は, 2変量の場合に限定せずに, 任意の **多変量正規分布** (multivariate normal distribution)にも使用することにする.)

この場合には条件 $X = x$ が定める y に関する条件付き確率分布が, 平均と分散がそれぞれ

$$\mu = (b/a)(x - \mu_x) + \mu_y, \quad \sigma^2 = (ac - b^2)/a$$

の正規分布になることを示せ.

解答例: 上の2変量正規分布において条件 $X = x$ が定める y に関する条件付き確率分布を求めよう.

$$\begin{aligned}
&a(y-\mu_y)^2 - 2b(x-\mu_x)(y-\mu_y) + c(x-\mu_x)^2 \\
&= a\left(y - \mu_y - \frac{b}{a}(x-\mu_x)\right)^2 - \frac{b^2}{a}(x-\mu_x)^2 + c(x-\mu_x)^2 \\
&= a\left(y - \mu_y - \frac{b}{a}(x-\mu_x)\right)^2 + \frac{ac-b^2}{a}(x-\mu_x)^2.
\end{aligned}$$

ゆえに,

$$\begin{aligned}
&\sqrt{(2\pi)^2(ac-b^2)}\, p(x) \\
&= \int_{\mathbb{R}} \exp\left(-\frac{a\big(y-\mu_y-(b/a)(x-\mu_x)\big)^2 + ((ac-b^2)/a)(x-\mu_x)^2}{2(ac-b^2)}\right) dy \\
&= \sqrt{\frac{2\pi(ac-b^2)}{a}} \exp\left(-\frac{(x-\mu_x)^2}{2a}\right)
\end{aligned}$$

なので,

$$p(x) = \frac{1}{\sqrt{2\pi a}} \exp\left(-\frac{(x-\mu_x)^2}{2a}\right).$$

これは x の周辺分布が平均 μ_x, 分散 a の正規分布になることを意味している. (実は多変量正規分布の一般論からこれは言えることでもある.) さらに,

$$\begin{aligned}
&a(y-\mu_y)^2 - 2b(x-\mu_x)(y-\mu_y) + c(x-\mu_x)^2 - \frac{ac-b^2}{a}(x-\mu_x)^2 \\
&= a\left(y - \mu_y - \frac{b}{a}(x-\mu_x)\right)^2
\end{aligned}$$

より,

$$p(y|x) = \frac{p(x,y)}{p(x)} = \frac{1}{\sqrt{2\pi(ac-b^2)/a}} \exp\left(-\frac{a\big(y-\mu_y-(b/a)(x-\mu_x)\big)^2}{2(ac-b^2)}\right).$$

これは条件 $X = x$ が定める y の条件付き確率分布が平均 $(b/a)(x-\mu_x)+\mu_y$, 分散が $(ac-b^2)/a$ の正規分布になることを意味している.

注意: 以上の計算を一般の多変量正規分布に一般化すれば **Gauss過程回帰** が得られる.

注意: 最小二乗法による線形回帰との関係. 条件付き確率分布の平均と分散は $\sigma_x^2 = a$, $\sigma_{xy} = b$, $\sigma_y^2 = c$ と書くと,

$$\frac{b}{a}(x-\mu_x) + \mu_y = \frac{\sigma_{xy}}{\sigma_x^2}(x-\mu_x) + \mu_y, \quad \frac{ac-b^2}{a} = \frac{\sigma_x^2\sigma_y^2 - \sigma_{xy}^2}{\sigma_x^2}$$

と表される. これは「標本分布について」のノートで導出したデータ $(x_1, y_1), \ldots, (x_n, y_n)$ に関する最小二乗法による線形回帰の公式

$$\hat{\alpha} + \hat{\beta}x = \frac{s_{xy}}{s_x^2}(x - \bar{x}) + \bar{y}, \quad \hat{\sigma}^2 = \frac{n-1}{n}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}$$

に非常に似ている. ここで

$$
\begin{aligned}
\bar{x} &= \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \\
s_x^2 &= \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2, \quad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2, \\
s_{xy} &= \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}).
\end{aligned}
$$

さらに,

$$
\begin{aligned}
\hat{\mu}_x &= \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \hat{\mu}_y = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \\
\hat{\sigma}_x^2 &= \frac{n-1}{n} s_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2, \quad \hat{\sigma}_y^2 = \frac{n-1}{n} s_y^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2, \\
\hat{\sigma}_{xy} &= \frac{n-1}{n} s_{xy} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}).
\end{aligned}
$$

とおくと,

$$
\hat{\alpha} + \hat{\beta} x = \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x^2}(x - \hat{\mu}_x) + \hat{\mu}_y, \quad \hat{\sigma}^2 = \frac{\hat{\sigma}_x^2 \hat{\sigma}_y^2 - \hat{\sigma}_{xy}^2}{\hat{\sigma}_x^2}
$$

となり, 式の類似性はさらに増す.

この類似は偶然ではない. 最小二乗法による線形回帰は, データ $(x_1, y_1), \ldots, (x_n, y_n)$ に最もフィットする2変量正規分布を最尤法で構成し($\hat{\mu}_x$, $\hat{\mu}_y$, $\hat{\sigma}_x^2$, $\hat{\sigma}_y^2$, $\hat{\sigma}_{xy}$ がその最尤法の解になっている), 条件 $X = x$ で定められた y に関する条件付き確率分布を求めることと数学的に同じことになっている.

1.17 2変量の正規分布とその条件付き確率分布の例の視覚化

以下では前節の内容を2変量正規分布の乱数を大量に生成することによって数値的に確認してみよう.

In [22]:
```
# 二変量正規分布
μ = [10, 20]
Σ = [
    6 2
    2 3
]
mvnormal = MvNormal(μ, Σ)
```

Out[22]:
```
FullNormal(
dim: 2
μ: [10.0, 20.0]
Σ: [6.0 2.0; 2.0 3.0]
)
```

In [23]:
```
# X, Y 単独の分布
xdist = Normal(μ[1], √Σ[1,1])
ydist = Normal(μ[2], √Σ[2,2])
@show xdist ydist;
```

```
xdist = Normal{Float64}(μ=10.0, σ=2.449489742783178)
ydist = Normal{Float64}(μ=20.0, σ=1.7320508075688772)
```

In [24]:
```
# 条件X=xが定めるyの条件付き確率分布の平均値 (≈ 回帰直線 (regression line))
f(x, μ, Σ) = (Σ[1,2]/Σ[1,1])*(x - μ[1]) + μ[2]
```

Out[24]:
```
f (generic function with 1 method)
```

In [25]:

```
# テストデータの生成: 二変量正規分布の分布は楕円状になる.
n = 10^4
XY = rand(mvnormal, n)
X, Y = XY[1,:], XY[2,:]
scatter(X, Y; ma=0.3, ms=2, msw=0, label="", title="data")
```

Out[25]:

In [26]:

```
# データでのX単独の分布
stephist(X; norm=true, label="data X")
plot!(xdist; label="true", ls=:dash)
```

Out[26]:

In [27]:

```
# データでのY単独の分布
stephist(Y; norm=true, label="data Y")
plot!(ydist; label="true", ls=:dash)
```

Out[27]:

In [28]:
```
# μの最尤推定
μ̂ = mean(XY, dims=2)
```

Out[28]:
```
2×1 Matrix{Float64}:
  9.986931890736575
 19.987253275638245
```

In [29]:
```
# Σの最尤推定
Σ̂ = cov(XY'; corrected=false)
```

Out[29]:
```
2×2 Matrix{Float64}:
 6.04319  1.9508
 1.9508   2.95852
```

In [30]:
```
# 条件X=xが定めるyの条件付き確率分布の期待値と回帰直線の比較
xlim = (μ[1] - 4√Σ[1,1], μ[1] + 4√Σ[1,1])
plot(x -> f(x, μ̂, Σ̂), xlim...; label="linear regression", c=:red)
plot!(x -> f(x, μ, Σ), xlim...; label="true", ls=:dash, c=:blue)
plot!(; legend=:topleft)
```

Out[30]:

In [31]:
```
# サンプルとまとめてプロット
scatter(X, Y; ma=0.3, ms=2, msw=0, label="", title="data")
plot!(x -> f(x, μ̂, Σ̂); label="linear regression", c=:red)
plot!(x -> f(x, μ, Σ); label="true", ls=:dash, c=:blue)
plot!(; legend=:topleft)
```

Out[31]:

In [32]:
```
# 条件X=xが定めるyの条件付き確率分布の分散とその推定値の比較
σ² = det(Σ)/Σ[1,1]
σ̂² = det(Σ̂)/Σ̂[1,1]
@show σ² σ̂²;
```

```
σ² = 2.3333333333333335
σ̂² = 2.328779918715761
```

以下では, x, y の同時確率密度函数 $p(x,y)$ (これは平均 μ と分散共分散行列 Σ を持つ二変量正規分布の密度函数)と x で条件付けられた y に関する条件付き確率分布の密度函数 $p(y|x) = p(x,y)/p(x)$ のヒートマップ(値が大きなところほど明るくプロットされる)を比較してみよう.

In [33]:

```
f(x, y) = pdf(mvnormal, SVector(x, y)) # p(x,y)
f(x) = pdf(xdist, x) # p(x)
g(x, y) = f(x, y)/f(x)

x = range(0, 20, 300)
y = range(10, 30, 300)

P1 = heatmap(x, y, f; colorbar=false, title="p(x,y)")
P2 = heatmap(x, y, g; colorbar=false, title="p(y|x)")
plot(P1, P2; size=(600, 250))
```

Out[33]:

$p(x,y)$ は xy 平面上の確率密度函数になっている.

$p(y|x)$ は各 x ごとに y 軸に平行な直線上の確率密度函数になっている.

その様子の違いの典型例が上のグラフからわかる.

$p(y|x)$ のヒートマップを見れば, パラメータ x を持つ y の確率密度函数は x を y に対応させる函数の一般化(x に対応する y の値が確率的に揺らぐ)になっていることもわかる.

2 2×2の分割表の分布

実践的によく使われている2×2の分割表のモデルになる確率分布を紹介しよう.

2×2の分割表とは数学的には0以上の整数を成分とする2×2の行列

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad a_{ij} \in \mathbb{Z}_{\geq 0}$$

のことである. 以下では2×2の分割表の集合に順次制限して狭くして行きながら, 4種類の確率分布を紹介する.

それら4種の確率分布達は条件付け確率分布の階層をなしている.

2.1 4つのPoisson分布の積

$\lambda_{ij} \geq 0$ であるとし,

$$\lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix}$$

とおく. 制限のない分割表 $A = [a_{ij}]$ 達の確率質量函数

$$P(A|\lambda) = \prod_{i,j=1}^{2} \left(e^{-\lambda_{ij}} \frac{\lambda_{ij}^{a_{ij}}}{a_{ij}!} \right) \qquad (a_{ij} \in \mathbb{Z}_{\geq 0})$$

で定まる分割表の確率分布を **4つのPoisson分布の積** と呼ぶことにする.

分割表の4つのPoisson分布の積モデルはある一定期間のあいだ収集したサンプルを2つの条件 X, Y を使って次のように分類することによって得られた2×2の分割表のモデルになっていると考えられる:

- a_{11} はサンプル中の X でかつ Y であるものの個数で, a_{12} は X でかつ Y でないものの個数で, a_{21} は X でなくて Y であるものの個数で, a_{22} は X でも Y でもないものの個数であるとする.

以上の設定のもとで4つのPoisson分布の期待値は

$$\lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix}$$

に一致する.

2.2 四項分布

$\theta_{ij} \geq 0, \sum_{i,j=1}^{2} \theta_{ij} = 1$ であるとし,

$$\theta = \begin{bmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix}$$

とおく. $n \in \mathbb{Z}_{\geq 0}$ を任意に取って固定し, 成分の総和が n になるという条件

$$\sum_{i,j=1}^{2} a_{ij} = n$$

を満たす分割表 $A = [a_{ij}]$ に関する確率分布が次の確率質量函数によって定められる:

$$P(A|n,\theta) = n! \prod_{i,j=1}^{4} \frac{\theta_{ij}^{a_{ij}}}{a_{ij}!} \qquad (a_{11} + a_{12} + a_{21} + a_{22} = n)$$

これは **四項分布** である.

分割表の四項分布モデルは, 最初に決めておいたサイズ n のサンプルを収集して, 2つの条件 X, Y を使って前節を同じようにサンプルを分類することによって得られた分割表のモデル化になっていると考えられる.

この四項分布の期待値は

$$n\theta = \begin{bmatrix} n\theta_{11} & n\theta_{12} \\ n\theta_{21} & n\theta_{22} \end{bmatrix}$$

になる.

2.3 2つの二項分布の積

$0 \leq \eta_i \leq 1$ $(i = 1, 2)$であるとし, $\eta = (\eta_1, \eta_2)$ とおく. $r_1, r_2 \in \mathbb{Z}_{\geq 0}$ を任意に取って固定し, $r = (r_1, r_2)$ とおく. このとき, 横方向の合計(横方向のマージン)が

$$\sum_{j=1}^{n} a_{ij} = r_i \quad (i = 1, 2)$$

と固定されているような分割表の確率分布を次の確率質量函数によって定めることができる:

$$P(A|r,\eta) = \binom{r_1}{a_{11}} \eta_1^{a_{11}} (1-\eta_2)^{a_{12}} \cdot \binom{r_2}{a_{21}} \eta_1^{a_{21}} (1-\eta_2)^{a_{22}} \quad (a_{i1} + a_{i2} = r_i)$$

これを **2つの二項分布の積** と呼ぶ.

分割表の2つの二項分布の積モデルは, 前もって決めておいたサイズ r_1, r_2 の条件 X を満たす群と満たさない群のサンプルを取得して, 条件 Y によって分類することによって得られた分割表のモデル化になっていると考えられる.

この設定のもとで, 2つの二項分布の積の期待値は

$$\begin{gathered} r_1\eta_1 r_1(1-\eta_1) \\ r_2\eta_2 r_2(1-\eta_2) \end{gathered}$$

になる.

2.4 Fisherの非心超幾何分布

$\omega > 0$ であるとする. $r_1, r_2, s_1, s_2, n \in \mathbb{Z}_{\geq 0}$ で

$$n = r_1 + r_2 = s_1 + s_2$$

という条件を満たすものを任意に取って固定し, $r = (r_1, r_2)$, $s = (s_1, s_2)$ とおく. このとき, 横方向の合計と縦方向の合計の両方(全マージン)が

$$\sum_{j=1}^{2} a_{ij} = r_i \quad (i = 1, 2), \qquad \sum_{i=1}^{2} a_{ij} = s_j \quad (j = 1, 2)$$

と固定されているような分割表の確率分布を次の形の確率質量函数によって定めることができる:

$$\begin{aligned}
&P(A|r, s, \omega) \\
&= Z(r, s, \omega)^{-1} \frac{\omega^{a_{11}}}{\prod_{i,j=1}^{2} a_{ij}!} = \frac{\omega^{a_{11}}}{\widetilde{Z}(r, s, \omega)} \binom{n}{r_1}^{-1} \binom{s_1}{a_{11}} \binom{s_2}{a_{12}} \\
&= \frac{\omega^{a_{11}}}{\widetilde{Z}(r, s, \omega)} \binom{r_1}{a_{11}} \frac{s_1(s_1 - 1) \cdots (s_1 - a_{11} + 1)\, s_2(s_2 - 1) \cdots (s_2 - (r_1 - a_{11}) + 1)}{n(n-1) \cdots (n - r_1 + 1)}. \\
&(a_{i1} + a_{i2} = r_i,\ a_{1j} + a_{2j} = s_j)
\end{aligned}$$

これを **Fisherの非心超幾何分布** (Fisher's noncentral hypergeometric distribution)と呼ぶ.

パラメータの函数達 $Z(r, s, \omega)$, $\widetilde{Z}(r, s, \omega)$ は確率の総和が 1 になるという条件で決める. 一般にそれらの定数の具体形はシンプルに書けないが, $\omega = 1$ の場合は例外的に,

$$\widetilde{Z}(r, s, 1) = 1, \quad Z(r, s, 1) = \frac{n!}{r_1! r_2! s_1! s_2!}$$

とシンプルな形になり,

$$P(A|r, s, 1) = \binom{n}{r_1}^{-1} \binom{s_1}{a_{11}} \binom{s_2}{a_{12}} = \frac{r_1! r_2! s_1! s_2!}{n! a_{11}! a_{12}! a_{21}! a_{22}!}$$

となる. この $\omega = 1$ の場合のFisherの非心超幾何分布は単に **超幾何分布** と呼ばれている.

注意: 超幾何分布の期待値は

$$\begin{bmatrix} r_1 s_1/n & r_1 s_2/n \\ r_2 s_1/n & r_2 s_2/n \end{bmatrix}$$

になる. Fisherの非心超幾何分布の期待値についてはこのようなシンプルな公式はない.

注意: 超幾何分布の確率質量函数の上の表示は以下のような解釈を持つ.

袋の中に s_1 個の赤い玉と s_2 個の白い玉が入っているとし, その中から r_1 個の玉を一度にまとめて無作為に取り出す. 取り出した r_1 個の玉の中の赤い玉の個数を a_{11} とし, 白い玉の個数を $a_{12} = r_1 - a_{11}$ と書き, 袋の中に残された r_2 個の玉の中の赤い玉の個数を $a_{21} = s_1 - a11$ と書き, 白い玉の個数を $a_{22} = s_2 - a_{12} = s_2 - r_1 + a_{11}$ と書き, 2×2の分割表 $A = [a_{ij}]$ を作る. a_{11} だけから残りの $a_{12}, a_{21}. a_{22}$ の値が決定される.

このとき, A の分布は前節の超幾何分布に従う. なぜならば, 袋の中の $n = s_1 + s_2$ 個の玉の中から r_1 個を一度に取り出す組み合わせ全体の個数は $\binom{n}{r_1}$ であり(分母), 袋の中の s_1 個の赤い玉の中から a_{11} 個選ぶ組み合わせ全体の個数と s_2 個の白い玉の中から a_{12} 個選ぶ組み合わせ全体の個数の積は $\binom{s_1}{a_{11}}\binom{s_2}{a_{12}}$ になる(分子)からである.になる.

注意: 超幾何分布と非復元抽出の関係. 超幾何分布の確率質量函数は次のようにも表される:

$$P(A|r, s, 1) = \binom{r_1}{a_{11}} \frac{s_1(s_1 - 1) \cdots (s_1 - a_{11} + 1)\, s_2(s_2 - 1) \cdots (s_2 - (r_1 - a_{11}) + 1)}{n(n-1) \cdots (n - r_1 + 1)}.$$

この公式は以下のような解釈を持つ. 袋の中に s_1 個の赤い玉と s_2 個の白い玉が入っているとし, $s_1 + s_2 = n$ であるとする. 袋の中から無作為に1個ずつ玉を取り出す. ただし, 取り出した玉は袋に戻さないとする(非復元抽出). そのように玉を r_1 個取り出したとき, その中に赤い玉が a_{11} 個含まれる確率が超幾何分布における確率の値 $P(A|r, s, 1)$ になる.

注意: 以上の状況のもとで a_{11} が動ける範囲はちょうど次になる:

$$\max(0, r_1 - s_2) \leq a_{11} \leq \min(s_1, r_1).$$

この条件は以下の条件を整理することによって得られる:

$$
\begin{aligned}
&a_{11} \geq 0, \\
&a_{11} = r_1 - a_{12} \leq r_1, \\
&a_{11} = s_1 - a_{21} \leq s_1, \\
&a_{11} = r_1 - s_2 + a_{22} \geq r_1 - s_2.
\end{aligned}
$$

これらの等式部分は以下から得られる:

$$
\begin{aligned}
&a_{12} = r_1 - a_{11}, \\
&a_{21} = s_1 - a_{11}, \\
&a_{22} = s_2 - a_{12} = s_2 - r_1 + a_{11}.
\end{aligned}
$$

注意: 2×2の分割表のFisherの非心超幾何分布およびその特別な場合である超幾何分布においては a_{11} の値を決めれば他の a_{12}, a_{21}, a_{22} の値が決まるので, それらは

$$\max(0, r_1 - s_2) \leq a_{11} \leq \min(s_1, r_1)$$

を満たす整数 a_{11} のみに関する確率分布だとみなされる. a_{11} のみに関するFisherの非心超幾何分布を

$$\mathrm{FisherNoncentralHypergeometric}(s_1, s_2, r_1, \omega)$$

と書き, 超幾何分布を

$$\mathrm{Hypergeometric}(s_1, s_2, r_1)$$

と書くことにする. $\mathrm{Hypergeometric}(s_1, s_2, r_1)$ は s_1 個の赤い玉と s_2 個の白い玉が入っている袋から取り出した r_1 個の玉の中の赤い玉の個数 a_{11} の分布になっている.

袋の中の赤い玉の個数 s_1 が取り出す玉の個数 r_1 より小さい場合には取り出した r_1 個の玉がすべて赤い玉になること ($a_{11} = s_1$) はありえず, 袋の中の白い玉の個数 s_2 が取り出す玉の個数 r_1 より小さい場合には取り出した r_1 個の玉がすべて白い玉になること($a_{11} = 0$)となることはありえない. このことに注意すれば a_{11} の動く範囲についても理解し易いだろう.

2.5 Fisherの非心超幾何分布の正規分布近似の視覚化

Fisherの非心超幾何分布 $\mathrm{FisherNoncentralHypergeometric}(s, f, n)$ も s, f, n が十分大きければ正規分布で近似される.

In [34]:
```
function plot_fnch(s₁, s₂, r₁, ω)
    fnch = FisherNoncentralHypergeometric(s₁, s₂, r₁, ω)
    μ, σ = mean(fnch), std(fnch)
    xlim = (max(-1.0, μ-4σ), min(maximum(fnch)+1.0, μ+4σ))
    x = range(xlim..., 1000)
    plot(x -> mypdf(fnch, x), xlim...; label="")
    plot!(Normal(μ, σ); label="", ls=:dash)
    title!("s₁=$s₁, s₂=$s₂, r₁=$r₁, ω=$ω")
end
```

Out[34]: plot_fnch (generic function with 1 method)

In [35]:
```
plot(plot_fnch(6, 14, 10, 0.5), plot_fnch(6, 14, 10, 2.0); size=(800, 250))
```

Out[35]:

In [36]:
```
plot(plot_fnch(12, 28, 20, 0.5), plot_fnch(12, 28, 20, 2.0); size=(800, 250))
```

Out[36]:

In [37]:
```
plot(plot_fnch(24, 56, 40, 0.5), plot_fnch(24, 56, 40, 2.0); size=(800, 250))
```

Out[37]:

In [38]:
```
plot(plot_fnch(48, 112, 80, 0.5), plot_fnch(48, 112, 80, 2.0); size=(800, 250))
```

Out[38]:

2.6 四項分布は4つのPoisson分布の積の条件付き確率分布

オッズ比 ω を次のように定める:

$$\omega = \frac{\lambda_{11}/\lambda_{12}}{\lambda_{21}/\lambda_{22}} = \frac{\lambda_{11}\lambda_{22}}{\lambda_{12}\lambda_{21}}.$$

4つのPoisson分布の積の確率質量函数を $\sum_{i,j=1} a_{ij} = n$ が固定された場合に制限してみよう.

$$\theta_{ij} = \frac{\lambda_{ij}}{\sum_{i',j'=1}^{n} \lambda_{i'j'}}$$

とおくと,

$$\omega = \frac{\theta_{11}/\theta_{12}}{\theta_{21}/\theta_{22}} = \frac{\theta_{11}\theta_{22}}{\theta_{12}\theta_{21}}$$

でかつ,

$$P(A|\lambda) = \prod_{i,j=1}^{2}\left(e^{-\lambda_{ij}}\frac{\lambda_{ij}^{a_{ij}}}{a_{ij}!}\right) = \underbrace{\frac{1}{n!}e^{-\sum_{i,j=1}^{2}\lambda_{ij}}\left(\sum_{i,j=1}^{2}\lambda_{ij}\right)^{n}}_{\text{independent of } a_{ij}} n!\prod_{i,j=1}^{2}\frac{\theta_{ij}^{a_{ij}}}{a_{ij}!}.$$

ゆえに, 四項分布における確率の総和が1であること

$$\sum_{\sum_{i,j=1}^{2}a_{ij}=n} n!\prod_{i,j=1}^{2}\frac{\theta_{ij}^{a_{ij}}}{a_{ij}!} = 1$$

より,

$$P(n|\lambda) = \sum_{\sum_{i,j=1}^{2}a_{ij}=n} P(A|\lambda) = \frac{1}{n!}e^{-\sum_{i,j=1}^{2}\lambda_{ij}}\left(\sum_{i,j=1}^{2}\lambda_{ij}\right)^{n}.$$

ゆえに, $\sum_{i,j=1}^{2}a_{ij} = n$ という条件で定義される条件付き確率分布の確率質量函数は

$$\frac{P(A|\lambda)}{P(n|\lambda)} = n!\prod_{i,j=1}^{2}\frac{\theta_{ij}^{a_{ij}}}{a_{ij}!}$$

と, 四項分布の確率質量函数になる.

2.7 2つの二項分布の積は四項分布は4つの条件付き確率分布

四項分布の確率質量函数を $\sum_{j=1}^{2}a_{ij} = r_i$, $r_1 + r_2 = n$ の場合に制限してみよう.

$$\eta_i = \frac{\theta_{i1}}{\theta_{i1}+\theta_{i2}}$$

とおくと,

$$\omega = \frac{\eta_1/(1-\eta_1)}{\eta_2/(1-\eta_2)} = \frac{\eta_1(1-\eta_2)}{(1-\eta_1)\eta_2}$$

でかつ,

$$P(A|n,\theta) = n!\prod_{i,j=1}^{4}\frac{\theta_{ij}^{a_{ij}}}{a_{ij}!} = \underbrace{n!\prod_{i=1}^{2}(\theta_{i1}+\theta_{i2})^{r_i}}_{\text{independent of } a_{ij}} \cdot \frac{\eta_1^{a_{11}}(1-\eta_1)^{r_1-a_{11}}}{a_{11}!(r_1-a_{11})!}\frac{\eta_2^{a_{21}}(1-\eta_2)^{r_1-a_{21}}}{a_{21}!(r_2-a_{21})!}$$

$$= \underbrace{\binom{n}{r_1}\prod_{i=1}^{2}(\theta_{i1}+\theta_{i2})^{r_i}}_{\text{independent of } a_{ij}} \cdot \binom{r_1}{a_{11}}\eta_1^{a_{11}}(1-\eta_1)^{r_1-a_{11}} \cdot \binom{r_2}{a_{21}}\eta_2^{a_{21}}(1-\eta_2)^{r_1-a_{21}}$$

これを $a_{11}+a_{12} = r_1$, $a_{21}+a_{22} = r_2$ を満たす分割表について和を取ると, 2つの二項分布の積の部分の和が 1 になる. ゆえに

$$P(r|n,\theta) = \binom{n}{r_1}\prod_{i=1}^{2}(\theta_{i1}+\theta_{i2})^{r_i} = \binom{n}{r_1}(\theta_{11}+\theta_{12})^{r_1}(1-(\theta_{11}+\theta_{12}))^{n-r_1}.$$

したがって, 条件 $a_{11}+a_{12} = r_1$, $a_{21}+a_{22} = r_2$ が定める条件付き確率分布の確率質量函数は次の形になる:

$$\frac{P(A|n,\theta)}{P(r|n,\theta)} = \binom{r_1}{a_{11}}\eta_1^{a_{11}}(1-\eta_1)^{r_1-a_{11}} \cdot \binom{r_2}{a_{21}}\eta_2^{a_{21}}(1-\eta_2)^{r_1-a_{21}}.$$

これは2つの二項分布の積の確率質量函数である.

2.8 Fisherの非心超幾何分布は2つの二項分布の積の条件付き確率分布

2つの二項分布の積の確率質量函数をさらに $\sum_{i=1}^{2}a_{ij} = s_j$, $s_1 + s_2 = n$ の場合に制限してみよう. そのとき,

$$a_{21} = s_1 - a_{11}, \quad a_{22} = s_2 - r_1 + a_{11}, \quad \omega = \frac{\eta_1(1-\eta_2)}{(1-\eta_1)\eta_2}$$

であることを使うと,

$$\begin{aligned}
P(A|r,\theta) &= \binom{r_1}{a_{11}}\eta_1^{a_{11}}(1-\eta_1)^{r_1-a_{11}} \cdot \binom{r_2}{a_{21}}\eta_2^{a_{21}}(1-\eta_2)^{r_2-a_{21}} \\
&= \frac{r_1!}{a_{11}!a_{12}!}\eta_1^{a_{11}}(1-\eta_1)^{r_1-a_{11}} \cdot \frac{r_2!}{a_{21}!a_{22}!}\eta_2^{s_1-a_{11}}(1-\eta_2)^{s_2-r_1+a_{11}} \\
&= \underbrace{r_1!r_2!(1-\eta_1)^{r_1}\eta_2^{s_1}(1-\eta_2)^{s_2-r_1}}_{\text{independent of } a_{ij}} \frac{\omega^{a_{11}}}{a_{11}!a_{12}!a_{21}!a_{22}!}.
\end{aligned}$$

これより, 2つの二項分布の積における条件 $\sum_{i=1}^{2} a_{ij} = s_j$ に関する条件付き確率分布がFisherの非心超幾何分布になることがわかる.

2.9 独立性条件

以上における条件付き確率分布の構成でオッズ比 ω が保たれて行くことに注意せよ. 以上で扱った4種類の確率分布は $\omega = 1$ を満たすとき **独立である**, もしくは, **独立性条件を満たす** という.

2つの二項分布の積においては独立であることと $\eta_1 = \eta_2$ であることは同値である. η_i は **比率** と呼ばれることがあり, $\eta_1 = \eta_2$ は比率が等しいという条件になる.

「比率が等しい」という仮説の検定で $\omega = 1$ の統計モデルが使用される.

たとえば, タバコを吸っているかどうかの条件を X と書き, 肺がんになるかどうかの条件を Y と書くとき, その調査結果から作った分割表

	$Y=1$	$Y=0$	
$X=1$	a人	b人	r_1人
$X=0$	c人	d人	r_2人
	s_1人	s_2人	n人

に関する独立性は条件 X と条件 Y が無関係であることを意味する.

2.10 Pearsonのχ²統計量

以上で紹介した2×2の分割表

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad a_{ij} \in \mathbb{Z}_{\geq 0}$$

の4つの確率分布について, 独立性条件が成立しているならば, 次の式で定義される **Pearsonのχ²統計量**

$$X^2 = \frac{(a+b+c+d)(ad-bc)^2}{(a+b)(c+d)(a+c)(b+d)}$$

は近似的に自由度 1 のχ²分布に従うことが知られている. 以下でそれを視覚化によって確認しておこう.

In [39]:

```julia
function plot_pearson_chisq(randfunc, A, B, C, D; L=10^5)
    # 仮の期待値 A, B, C, D から独立性条件を満たす期待値を作成
    N = A + B + C + D
    Ā = (A+B)*(A+C)/N
    B̄ = (A+B)*(B+D)/N
    C̄ = (C+D)*(A+C)/N
    D̄ = (C+D)*(B+D)/N
    # Pearsonのχ² 統計量を配列 X² に格納
    X² = Vector{Float64}(undef, L)
    Threads.@threads for i in 1:L
        a, b, c, d = randfunc(Ā, B̄, C̄, D̄)
        X²[i] = safediv((a+b+c+d)*(a*d-b*c)^2, (a+b)*(c+d)*(a+c)*(b+d))
    end
    ecdf_chisq = ecdf(X²) # Pearsonのχ² 統計量の経験累積分布函数
    plot(; legend=:bottomright)
    plot!(x -> ecdf_chisq(x), -0.2, 6.5; label="ecdf of Pearson's χ²")
    plot!(x -> cdf(Chisq(1), x); label="cdf of Chisq(1)", ls=:dash)
    title!("$randfunc($Ā, $B̄, $C̄, $D̄)")
end

function rand_4poisson(A, B, C, D)
    rand(Poisson(A)), rand(Poisson(B)), rand(Poisson(C)), rand(Poisson(D))
end

function rand_quadnomial(A, B, C, D)
    N = Int(A + B + C + D)
    rand(Multinomial(N, [A/N, B/N, C/N, D/N]))
end

function rand_2binomial(A, B, C, D)
    R₁, R₂ = Int.((A+B, C+D))
    a, c = rand(Binomial(R₁, A/(A+B))), rand(Binomial(R₂, C/(C+D)))
    b, d = R₁-a, R₂-c
    a, b, c, d
end

function rand_hypergeometric(A, B, C, D)
    A, B, C, D = Int.((A, B, C, D))
    a = rand(Hypergeometric(A+C, B+D, A+B))
    b = A+B-a
    c = A+C-a
    d = D-A+a
    a, b, c, d
end
```

Out[39]: rand_hypergeometric (generic function with 1 method)

以下では期待値が独立性条件を満たす

$$\begin{bmatrix} 4 & 6 \\ 6 & 9 \end{bmatrix}$$

の場合のPearsonのχ²統計量の分布の累積分布函数をプロットし, 自由度 1 のχ²分布と比較している.

In [40]:

```julia
plot_pearson_chisq(rand_4poisson, 4, 6, 6, 9)
```

Out[40]:

In [41]:
```
plot_pearson_chisq(rand_quadnomial, 4, 6, 6, 9)
```

Out[41]:

In [42]:
```
plot_pearson_chisq(rand_2binomial, 4, 6, 6, 9)
```

Out[42]:

In [43]:
```
plot_pearson_chisq(rand_hypergeometric, 4, 6, 6, 9)
```

Out[43]:

最後の超幾何分布の場合にのみ自由度 1 のχ²分布による近似が大幅に悪化することに注意せよ. それ以外の場合にはχ²分布による近似の精度はかなりよいように見える.

3 尤度 (ゆうど)と推定

3.1 尤度に関するまとめ

「尤度」は「ゆうど」と読むが, 正しい読み方を知っても統計学的実践に関する有用な知見は得られない.

尤度の定義は **統計モデル内でデータと同じ数値が生成される確率もしくは確率の密度** である.

尤度は統計モデルの「もっともらしさ」(尤もらしさ)の指標では **ない**. (尤度の英語名は「もっともらしさ」を意味する likelihood なのでこの点には注意を要する.)

尤度は「証拠の強さ」の指標でも **ない**. (尤度は証拠の強さの指標として不適切であるにもかかわらず, 歴史的な事情で evidence (証拠)という別名が付いているので, この点にも注意が必要である.)

尤度は **統計モデルのデータへの数値への適合度**(フィッティングの良さ)の指標の1つ(他にも沢山ある)として基本的かつ有用である.

尤度をそのような道具として使用することは, 統計モデルのデータの数値への適合度(フィッティングの良さ)をモデル内でデータと同じ数値がどれだけ生成され易いかで測ることを意味している.

尤度が高い(＝モデルがデータの数値によくフィットしている)にも関わらず役に立たない統計モデルを作ってしまうことはよくある. そうなってしまうことを **オーバーフィッティング** (overfitting, 過剰適合)という.

3.2 尤度函数と尤度の定義と諸注意

パラメータ $\theta = (\theta_1, \ldots, \theta_d)$ を持つ確率質量函数 $P(x|\theta)$ または確率密度函数 $p(x|\theta)$ が現実から取得するデータ(数値列) $x = (x_1, \ldots, x_n)$ の生成法則のモデル化になっているとき, それらを **統計モデル** と呼ぶ.

データの数値 x が得られたとき, パラメータ θ の函数 $\theta \mapsto P(x|\theta)$ または $\theta \mapsto p(x|\theta)$ をデータ x に関する統計モデルの **尤度函数** (ゆうどかんすう, likelihood function)と呼ぶ.

さらにパラメータの値 θ が与えられたとき, そこでの尤度函数の値 $P(x|\theta)$ または $p(x|\theta)$ を **尤度** (ゆうど, likelihood)と呼ぶ.

すなわち, **統計モデル内においてデータ x と同じ数値が生成される確率 $P(x|\theta)$ または確率の密度 $p(x|\theta)$ を尤度(ゆうど)と呼ぶ**.

尤度は **統計モデルのデータの数値への適合度の指標の1つ**(フィッティングの良さの指標の1つ, 他にも沢山ある)として使われる.

データの数値 x が与えられたとき, 統計モデル $p(x|\theta)$ をデータの数値 x に最も適合するようなパラメータ θ の値は, 尤度を適合度の指標として採用したとき, 尤度函数 $\theta \mapsto p(x|\theta)$ を最大化する $\theta = \hat{\theta}$ として得られる.

この方法を **最尤法** (maximum likelihood method)と呼び, **最尤法の解** $\hat{\theta}$ を **最尤推定値** (maximum likelihood estimate)と呼び, $\theta = \hat{\theta}$ での尤度函数の値 $p(x|\hat{\theta})$ をデータの数値 x に関するモデルの **最大尤度** と呼ぶ.

実際の計算や理論的な取り扱いでは尤度函数そのものではなく, その対数である **対数尤度函数** (log-likelihood function) $\theta \mapsto \log p(x|\theta)$ を使った方が便利なことが多い. 実際にはその -1 倍の **負の対数尤度函数** (negative log-likelihood function) $\theta \mapsto -\log p(x|\theta)$ が使われることが多い. 最尤法の解は負の対数尤度函数を最小化することによって得られる. (χ²分布との関係で -1 倍ではなく, -2 倍することも多い.)

注意: 尤度(ゆうど)は英語では「もっともらしさ」(尤もらしさ)を意味する likelihood という単語で表されるが, 統計学における専門用語としての尤度は「もっともらしさ」と呼ぶには適さない量になっているので注意が必要である. (非常に残念なことに, 多くの教科書で「尤度はもっともらしさを意味する」のような誤誘導的な解説が書いてある.)

注意: 尤度は「もっともらしさ」の指標としては不適切だが, 統計モデルのデータへの適合度(フィットの度合い)の指標の1つとして基本的かつ有用である.

注意: 尤度函数 $\theta \mapsto P(x|\theta)$ または $\theta \mapsto p(x|\theta)$ は一般に θ の確率密度函数にはなっていない.

注意: 「尤度函数は一般には確率密度函数ではない」を「尤度は確率ではない」と要約すると誤りになるので注意が必要である. データ x に関するパラメータ θ の離散分布モデルにおける尤度 $P(x|\theta)$ は確率の値になっている.

注意: 繰り返しになるが, **尤度はモデルのデータへの適合度の指標の1つ**(他にも沢山ある)として基本的かつ有用である.

注意: 尤度をデータへのモデルの適合度の指標として使うことは, モデルのデータへの適合度をモデル内でどれだけデータと同じ数値が生成され易いかでモデルのデータへの適合度を測るということになる.

注意: データの数値を得てから, 後付けで統計モデルを作れば, いくらでも尤度の高い統計モデルを作れる. さらに, データの数値を見てから後付けで統計モデルを作らなくても, 事前に統計モデルに大量のパラメータを入れておけば, 後付けで採用しそうなモデルもその中に含まれているようにでき, 事前に決めておいた方法でパラメータをデータの数値に合わせて調節することによって, 統計モデルの尤度を高めることも可能である. 何らかの理由で尤度が高い(すなわち既知のデータへの適合度が高い)のにこれから得られるデータの予測に役に立たないモデルができてしまうことを **オーバーフィッティング** (overfitting, 過剰適合)と呼ぶ. 過剰適合の問題があるので, 尤度を単純に「もっともらしさ」の指標として安易に採用することは危険である.

3.3 例: Bernoulli試行の尤度函数と最尤法

当たりが出る確率 p が未知のルーレットを n 回まわして得た当たりとはずれの長さ n 列 $x_1, x_2, \ldots, x_n$ がデータとして得られる状況を考える. 当たりは 1 ではずれは 0 で表しておくことにする. 例えば, ルーレットを $n = 5$ 回まわして, 当たり, はずれ, はずれ, 当たり, はずれが順に出たとき, $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 0, 1, 0)$ となるとする.

その状況をBernoulli試行でモデル化する. すなわち, 次の確率質量函数を統計モデルとして採用する:

$$P(x_1, \ldots, x_n|p) = p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)} \quad (x_i = 1, 0).$$

このとき, データの数値 $(x_1, \ldots, x_n)$ が得られたとき, この統計モデルの尤度函数は

$$L(p) = p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)}$$

になる. これの右辺の式の形は上の確率質量函数と同じである. しかし, 尤度函数の側では $x_1, \ldots, x_n$ が具体的なデータの数値になっている点が異なる.

記号の簡単のため, データ $x_1, \ldots, x_n$ の中の当たり(=1)の個数を k と書くことにする:

$$k = x_1 + \cdots + x_n.$$

このとき, 尤度函数は次のように書かれる:

$$L(p) = p^k(1-p)^{n-k}$$

実際の計算では尤度函数そのものではなく, その対数である **対数尤度函数** (log-likelihood function)

$$\ell(p) = \log L(p) = k\log p + (n-k)\log(1-p)$$

の方が便利であることが多い. 実際にはさらにこれの -1 倍(**負の対数尤度函数** (negative log-likelihood function)

$$-\ell(p) = -\log L(p) = -k\log p - (n-k)\log(1-p)$$

が使われることが多い. 最尤法の解は負の対数尤度函数を最小化することによって得られる. (χ²分布との関係で -1 倍ではなく, -2 倍することも多い.)

パラメータの値 p の尤度 $L(p)$ もしくはその対数 $\log L(p)$ はBernoulli試行モデルのデータ $x_1, \ldots, x_n$ への適合度の指標の1つとみなされる. だから, Bernoulli試行モデルのデータ $x_1, \ldots, x_n$ への適合度を最大化することは, 対数尤度 $\log L(p)$ を最大化するパラメータ値 p を採用することに等しいと考えることができる. このようにしてモデルのパラメータを決定することを **最尤法** (maximum likelihood method)や **最尤推定法** (maximum likelihood estimation)と呼ぶ. そして, そのようにして求めたパラメータ値を **最尤推定値** (maximum likelihood esitimate, **最尤法の解**) と呼び, 最尤推定値における尤度函数の値を **最大尤度** (maximum likelihood)と呼ぶ. **この段落の定義はBernoulli試行モデル以外の場合にもそのまま一般化される.**

最尤法の解 $p = \hat{p}$ を求めてみよう. そのために対数尤度函数を微分してみる:

$$\ell'(p) = \frac{k}{p} - \frac{n-k}{1-p} = \frac{k-np}{p(1-p)}$$

なので, $\ell'(p)$ は $p < k/n$ で正になり, $p > k/n$ で負になる. ゆえに $\ell(p)$ は $p < k/n$ で単調増加し, $p > k/n$ で単調減少する. ゆえに, 対数尤度函数 $\ell(p)$ を最大化する $p = \hat{p}$ は

$$\hat{p} = \frac{k}{n}$$

になる. Bernoulli試行モデルの「n 回中 k 回が当たりが出た」というデータに関する尤度は成功確率パラメータの値を $p = \hat{p} = k/n$ としたときに最大になり, Bernoulli試行モデルがデータに最もよくフィットするようになる.

この結果を見て, 「n 回中 k 回が当たりが出た」というデータによれば, そのルーレットで当たりが出る確率は k/n であるに違いないと推測することは **誤り** である. 以上で行った計算は単にモデルがデータに最もよくフィットするパラメータ値を求めただけであり, データを生成したルーレットで当たりが出る確率を求めてはいない.

ルーレットを n 回まわしたときに当たりが出る回数 k は, ルーレットを n 回まわす操作を行うごとに違う値になるだろう. データは確率的に揺らぐ. ルーレットで当たりが出る確率(これは未知)を適切に推測するためには, データの確率的揺らぎについても考慮する必要がある. この点については様々な考え方があるが, 後で我々は区間推定(信頼区間)の考え方を学ぶことになるだろう.

3.4 Bernoulli試行モデルの尤度函数の視覚化

前節で求めた「n 回中当たりが k 回出た」というデータに関するBernoulli試行モデルの尤度函数

$$L(p) = p^k(1-p)^{n-k}$$

のグラフを描いてみよう.

しかし, 浮動小数点数での計算では $0 < p < 1$ を満たす p の高次のべきは容易に 0 になってしまう. そこで最大値が 1 になるように定数をかけた次の函数のグラフをプロットすることにしよう:

$$\frac{L(p)}{L(k/n)} = \left(\frac{p}{k/n}\right)^k \left(\frac{1-p}{1-k/n}\right)^{n-k}.$$

In [44]:
```
function plot_berlik(n, k)
    if k < 0 || k > n
        return plot(; axis=false, tick=false, frame=false, margin=0Plots.mm)
    end
    L(p) = if 0 < p < 1
        exp(-(xlogy(k, safediv(k/n, p))+xlogy(n-k, safediv(1-k/n, 1-p))))
    else
        zero(p)
    end
    plot(L, -0.02, 1.02; label="", title="n=$n, k=$k")
    plot!(; xtick=0:0.1:1, tickfontsize=6)
end
```

Out[44]: plot_berlik (generic function with 1 method)

$n = 10$ を固定した場合

In [45]:
```
n = 10
plot(plot_berlik.(n,(0,1,2))...; size=(800,150), layout=(1,3))
```

Out[45]:

In [46]:
```
plot(plot_berlik.(n,(3,4,5))...; size=(800,150), layout=(1,3))
```

Out[46]:

In [47]:
```
n = 10
plot(plot_berlik.(n,(6,7,8))...; size=(800,150), layout=(1,3))
```

Out[47]:

In [48]:
```
n = 10
plot(plot_berlik.(n,(9,10,11))...; size=(800,150), layout=(1,3))
```

Out[48]:

k/n を固定して n を大きくした場合には, 尤度函数 $L(p)$ の台は k/n に集中して行く.

In [49]:
```
plot(plot_berlik.((10, 100, 1000), (3, 30, 300))...; size=(800,150), layout=(1,3))
```

Out[49]:

このグラフから, データに基づくパラメータの推定において, パラメータの推定値としてどれだけの幅を持たせるかについて, 尤度函数がなにがしかの情報を持っていそうなこともわかる.

3.5 問題: 尤度が高くても全然もっともらしくない例(オーバーフィッティングの例)

データは当たり(1 と書く)とはずれ(0 と書く)がランダムに出るルーレットを n 回まわして得た 1 と 0 の列 $x_1, x_2, \ldots, x_n$ として得られるとする.

統計モデルとして以下の2つを考える.

モデル0: Bernoulli試行モデル: このモデルのパラメータ $0 \leq p \leq 1$ を持つ確率質量函数は次のように書けるのであった:

$$P_0(x_1, \ldots, x_n|p) = p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)} \quad (x_i = 1, 0).$$

モデル1: ルーレットをまわすごとに当たりが出る確率が変わってもよいと想定したモデル: このモデルは n 個のパラメータ $0 \leq p_1, \ldots, p_n \leq 1$ を持つ次の確率質量函数によって与えられる:

$$P_1(x_1, \ldots, x_n|p_1, \ldots, p_n) = \prod_{i=1}^{n} \left(p_i^{x_i}(1-p_i)^{1-x_i}\right) \quad (x_i = 1, 0).$$

データ(1 と 0 の列) $x_1, \ldots, x_n$ について, モデル0の最尤法の解 $p = \hat{p}$ は $\hat{p} = k/n$ ($k = x_1 + \cdots + x_n$) になるのであった. 以下の2つを同時に示せ:

(1) モデル1の最尤法の解 $p_i = \hat{p}_i$ は $\hat{p}_i = x_i$ になる.

(2) どのようなデータ $x_1, x_2, \ldots, x_n$ についてもモデル1の最大尤度は離散分布モデルの場合に可能な最大の値 1 になる.

注意: 確率と違って確率密度の値はいくらでも大きくなりえるので, 尤度が確率密度の値になる連続分布モデルにおいては尤度はいくらでも大きな値を取り得る. 離散分布モデルの場合には尤度は確率の値になるのでその最大値は 1 になる.

注意: この例はオーバーフィッティング(過剰適合)のシンプルな例になっている. パラメータの個数が多いモデルではオーバーフィッティングし易くなる.

注意: この例を知っていれば, 「尤度が高ければもっともらしい」と考えることは単純に誤りであることがわかる. 尤度を常識的な意味でのもっともらしさと繋げるためには特別な条件が必要になる.

解答例: モデル1において, データ $x_1, x_2, \ldots, x_n$ (1 と 0 の列)について, $p_i = x_i$ とおくと,

$$x_i^{x_i}(1-x_i)^{1-x_i} = 1 \quad (x_i = 1, 0)$$

なので($0^0 = 1$ と約束しておく), その尤度は

$$P_1(x_1, \ldots, x_n|x_1, \ldots, x_n) = \prod_{i=1}^{n} \left(x_i^{x_i}(1-x_i)^{1-x_i}\right) = 1$$

となり, 可能な最大の値になる. ゆえに, 最尤推定値は $\hat{p} = x_i$ となる.

解答終

3.6 問題: 正規分布の標本分布モデルの尤度函数と最尤法

データは n 個の実数達 $x_1, \ldots, x_n$ であるとし, 統計モデルとして平均 μ, 分散 σ^2 の正規分布のサイズ n の標本分布を考える. その統計モデルの確率密度函数 $p(x_1, \ldots, x_n|\mu, \sigma^2)$ は,

$$p(x_i|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(x_i-\mu)^2}{2\sigma^2}\right)$$

とおくとき次のように表される:

$$p(x_1,\ldots,x_n|\mu,\sigma^2) = \prod_{i=1}^n p(x_i|\mu,\sigma^2).$$

データ $x_1,\ldots,x_n$ に関するこのモデルの尤度函数を最大化する $\mu=\hat{\mu}$, $\sigma^2=\hat{\sigma}^2$ はそれぞれ次になることを示せ:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^n x_i, \quad \hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (x_i-\hat{\mu})^2.$$

これは, 正規分布の標本分布モデルの最尤法の解が標本平均と(不偏にするための補正無しの)標本分散になることを意味している.

注意: この結果はデータの標本平均と標本分散を求めることは, 正規分布の標本分布モデルがデータに最もよくフィットするような μ と σ^2 を求めることに等しいことがわかる. この点は次の節の要約統計の説明でも触れる.

解答例: 記号の簡単のため以下のように書くことにする:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i, \quad v^2 = \frac{1}{n}\sum_{i=1}^n (x_i-\bar{x})^2.$$

このとき

$$\begin{aligned}
\sum_{i=1}^n (x_i-\mu)^2 &= \sum_{i=1}^n ((x_i-\bar{x})-(\mu-\bar{x}))^2 \\
&= \underbrace{\sum_{i=1}^n (x_i-\bar{x})^2}_{=nv^2} - 2(\mu-\bar{x})\underbrace{\sum_{i=1}^n (x_i-\bar{x})}_{=0} + \sum_{i=1}^n (\mu-\bar{x})^2 \\
&= n(v^2+(\mu-\bar{x})^2)
\end{aligned}$$

なので, 正規分布の標本分布の密度函数は以下のように表される:

$$\begin{aligned}
p(x_1,\ldots,x_n|\mu,\sigma^2) &= \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n (x_i-\mu)^2\right) \\
&= \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{n}{2\sigma^2}(v^2+(\mu-\bar{x})^2)\right).
\end{aligned}$$

ゆえにデータ $x_1,\ldots,x_n$ に関するこれの尤度函数を $\ell(\mu,\sigma)$ と書くと,

$$-\frac{2}{n}\ell(\mu,\sigma) = \frac{(\mu-\bar{x})^2+v^2}{\sigma^2} + \log\sigma^2 + \log(2\pi).$$

これを最小化する $\mu=\hat{\mu}$ と $\sigma^2=\hat{\sigma}^2$ が $\hat{\mu}=\bar{x}$, $\hat{\sigma}^2=v^2$ となることを示せばよい.

σ^2 を固定するとき, 上の式を最小化する $\mu=\hat{\mu}$ が $\hat{\mu}=\bar{x}$ になることはすぐにわかる.

$\mu=\bar{x}$ のとき

$$-\frac{2}{n}\ell(\bar{x},\sigma) = \frac{v^2}{\sigma^2} + \log\sigma^2 + \mathrm{const}. \tag{$\sharp$}$$

右辺を σ^2 を1つの変数だとみなして, σ^2 で微分すると

$$-\frac{v^2}{(\sigma^2)^2} + \frac{1}{\sigma^2} = \frac{\sigma^2-v^2}{(\sigma^2)^2}$$

であることから, これは σ^2 の函数として, $\sigma^2 < v_2$ のとき単調減少し, $\sigma^2 > v^2$ のとき単調増加するので, ($\sharp$)を最小化する $\sigma^2=\hat{\sigma}^2$ が $\hat{\sigma}^2=v^2$ であることがわかる.

解答終

3.7 正規分布の標本分布モデルの尤度函数の視覚化

値が大きな点ほど明るくプロットされるヒートマップによって正規分布の標本分布モデルの尤度函数をプロットしてみよう. 尤度函数を与えるためにはデータ $x_1, \ldots, x_n$ が必要になるが, データは簡単のため標準正規分布に従う乱数で生成することにする.

In [50]:
```
function plot_normallik(X, xlim, ylim; kwargs...)
    n = length(X)
    mle = fit_mle(Normal, X)
    μ̂, σ̂² = mean(mle), var(mle)
    f(μ, σ²) = exp(
        loglikelihood(Normal(μ, √σ²), X) -
        loglikelihood(Normal(μ̂, √σ̂²), X)
    )
    x = range(xlim..., 200)
    y = range(ylim..., 200)
    heatmap(x, y, f; colorbar=false)
    plot!(; xguide="μ", yguide="σ²", tickfontsize=7)
    title!("n = $n")
    plot!(; fmt=:png, size=(250, 250))
end
```

Out[50]: plot_normallik (generic function with 1 method)

In [51]:
```
Random.seed!(4649373) # よろしくみなさん
plot((plot_normallik(randn(10), (-1.5, 1.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[51]:

In [52]:
```
plot((plot_normallik(randn(40), (-1.5, 1.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[52]:

In [53]:
```
plot((plot_normallik(randn(160), (-1.5, 1.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[53]:

In [54]:
```
plot((plot_normallik(randn(640), (-1.5, 1.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[54]:

n を大きくすると, サンプルを生成した分布の期待値と分散の値の近くに尤度函数の台が集中して行く.

以下はサンプルを生成する分布を正規分布とは大きく違う指数分布にした場合

In [55]:
```
Random.seed!(4649373) # よろしくみなさん
plot((plot_normallik(rand(Exponential(), 10), (-0.5, 2.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[55]:

In [56]:
```
plot((plot_normallik(rand(Exponential(), 40), (-0.5, 2.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[56]:

In [57]:
```
plot((plot_normallik(rand(Exponential(), 160), (-0.5, 2.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[57]:

In [58]:
```
plot((plot_normallik(rand(Exponential(), 640), (-0.5, 2.5), (0, 4)) for _ in 1:3)...;
    size=(750,250), layout=(1,3), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[58]:

3.8 問題: 最小二乗法がある種の正規分布モデルの最尤法に一致すること

データは2つの実数の組が n 個 $(x_1, y_1), \ldots, (x_n, y_n)$ の形式であるとし, 記号の簡単のために次のようにおく:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad v_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2,$$
$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \quad v_y^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2, \quad v_{xy} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}).$$

統計モデルとして, パラメータ $x_1, \ldots, x_n, \alpha, \beta$ と $\sigma^2 > 0$ を持つ次の確率密度函数で定まる $y_1, \ldots, y_n$ に関する確率分布を考える:

$$p(y_1, \ldots, y_n | x_1, \ldots, x_n, \alpha, \beta, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - (\alpha + \beta x_i))^2\right).$$

このモデルのデータ $(x_1, y_1), \ldots, (x_n, y_n)$ に関する尤度函数

$$L(\alpha,\beta,\sigma^2)=\frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n(y_i-(\alpha+\beta x_i))^2\right)$$

を最大化する $\alpha=\hat\alpha$, $\beta=\hat\beta$, $\sigma^2=\hat\sigma^2$ は次になることを示せ:

$$\hat\beta=\frac{v_{xy}}{v_x^2},\quad \hat\alpha=\bar y-\hat\beta\bar x,\quad \hat\sigma^2=\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}.$$

注意: この公式は最小二乗法の公式と一致する.

解答例: 尤度函数の形を整理し直そう.

$$\begin{aligned}(y_i-(\alpha+\beta x_i))^2&=(y_i-\bar y+\bar y-(\alpha+\beta\bar x)+(\alpha+\beta\bar x)-(\alpha+\beta x_i))^2\\&=((y_i-\bar y)-\beta(x_i-\bar x)+(\bar y-(\alpha+\beta\bar x)))^2\\&=\beta^2(x_i-\bar x)^2-2\beta(x_i-\bar x)(y_i-\bar y)+(y_i-\bar y)^2\\&+(\bar y-(\alpha+\beta\bar x))^2+2(y_i-\bar y)(\bar y-(\alpha+\beta\bar x))-2\beta(x_i-\bar x)(\bar y-(\alpha+\beta\bar x))\end{aligned}$$

より

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^n(y_i-(\alpha+\beta x_i))^2&=v_x^2\beta^2-2v_{xy}\beta+v_y^2+(\bar y-(\alpha+\beta\bar x))^2\\&=v_x^2\left(\beta-\frac{v_{xy}}{v_x^2}\right)^2+\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}+(\bar y-(\alpha+\beta\bar x))^2.\end{aligned}$$

ゆえに, 尤度函数の対数の $-2/n$ 倍を $f(\alpha,\beta,\sigma^2)$ と書くと,

$$\begin{aligned}&f(\alpha,\beta,\sigma^2)=-\frac{2}{n}\log L(\alpha,\beta,\sigma^2)\\&=\frac{1}{\sigma^2}\left(v_x^2\left(\beta-\frac{v_{xy}}{v_x^2}\right)^2+\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}+(\bar y-(\alpha+\beta\bar x))^2\right)+\log\sigma^2+\log(2\pi).\end{aligned}$$

これを最小化する $\alpha=\hat\alpha$, $\beta=\hat\beta$, $\sigma^2=\hat\sigma^2$ が

$$\hat\beta=\frac{v_{xy}}{v_x^2},\quad \hat\alpha=\bar y-\hat\beta\bar x,\quad \hat\sigma^2=\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}.$$

になることを示せばよい. σ^2 を固定したとき, $f(\alpha,\beta,\sigma^2)$ を最小化する $\alpha=\hat\alpha$, $\beta=\hat\beta$ が

$$\hat\beta=\frac{v_{xy}}{v_x^2},\quad \hat\alpha=\bar y-\hat\beta\bar x$$

になることはすぐにわかる. このとき

$$f(\hat\alpha,\hat\beta,\sigma^2)=\frac{1}{\sigma^2}\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}+\log\sigma^2+\log(2\pi)$$

であり, 前節と同様に σ^2 を1つの変数だとみなして σ^2 で微分することによって, これを最小化する $\sigma^2=\hat\sigma^2$ が

$$\hat\sigma^2=\frac{v_x^2v_y^2-v_{xy}^2}{v_x^2}$$

になることがわかる.

解答終

3.9 最尤法の観点から見た十分統計量

$p(x|\theta)$ はパラメータ θ を持つ x に関する確率密度函数であるとする(確率質量函数の場合も同様である). $p(x|\theta)$ をデータ x に関する統計モデルの密度函数である場合について考える.

このとき, Fiher-Neymanの分解定理より, x の函数 $t(x)$ が **十分統計量** (sufficient statistic)であることと, モデルの密度函数が 0 以上の値を持つ函数たち $a(x)$, $b(t|\theta)$ によって

$$p(x|\theta)=a(x)b(t(x)|\theta)$$

と表されることは同値である. すなわちこの条件を十分統計量の定義として採用してもよい. この条件はデータ x の尤度函数

$$L(\theta) = a(x)b(t(x)|\theta)$$

が θ に関する定数 $a(x)$ 倍の違いを除けば, $t(x)$ と θ だけで尤度函数が表されることと同値である. 尤度 $L(\theta)$ を最大化することと $b(t(x)|\theta)$ を最大化することは同値なので, 最尤法の解は, データ x そのものの値ではなく, データ x から計算される $t(x)$ の値だけで決まることになる.

十分統計量はデータの要約を与えるためにも有用である.

3.9.1 Bernoulli試行モデルの十分統計量

Bernoulli試行モデルのパラメータ p を持つ確率質量函数は

$$P(x_1, \ldots, x_n|p) = p^{x_1+\cdots+x_n}(1-p)^{n-(x_1+\cdots+x_n)} \quad (x_i = 1, 0).$$

と書けるので,

$$t(x_1, \ldots, x_n) = x_1 + \cdots + x_n$$

は十分統計量になっている. $x_1, \ldots, x_n$ そのものの情報を忘れても, $k = x_1 + \cdots + x_n$ の情報さえあれば最尤法を実行できる.

3.9.2 正規分布の標本分布モデルの十分統計量

正規分布の標本分布モデルのパラメータ μ, σ^2 を持つ確率密度函数は

$$\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i, \quad v^2 = \frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})^2$$

とおくと,

$$p(x_1, \ldots, x_n|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{n}{2\sigma^2}(v^2 + (\mu - \bar{x})^2)\right)$$

と書けるので,

$$t(x_1, \ldots, x_n) = (\bar{x}, v^2) = \left(\frac{1}{n}\sum_{i=1}^n x_i, \frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})^2\right)$$

は十分統計量になっている. すなわち, 標本平均と標本分散の組み合わせは正規分布の標本分布モデルの十分統計量になっている. 補正無しの標本分散 v^2 を不偏分散 $s^2 = (n/(n-1))v^2$ で置き換えても十分統計量のままである.

3.9.3 最小二乗法を与える正規分布モデルの尤度函数を記述する統計量

最小二乗法を与える正規分布モデルのパラメータ $x_1, \ldots, x_n, \alpha, \beta, \sigma^2$ を持つ確率密度函数は

$$\begin{aligned}
\bar{x} &= \frac{1}{n}\sum_{i=1}^n x_i, \quad v_x^2 = \frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})^2, \\
\bar{y} &= \frac{1}{n}\sum_{i=1}^n y_i, \quad v_y^2 = \frac{1}{n}\sum_{i=1}^n (y_i - \bar{y})^2, \quad v_{xy} = \frac{1}{n}\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}).
\end{aligned}$$

とおくと,

$$\begin{aligned}
&p(y_1, \ldots, y_n|x_1, \ldots, x_n, \alpha, \beta, \sigma^2) \\
&= \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{n}{2\sigma^2}\left(v_x^2\beta^2 - 2v_{xy}\beta + v_y^2 + (\bar{y} - (\alpha + \beta\bar{x}))^2\right)\right) \\
&= \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{n}{2\sigma^2}\left(v_x^2\left(\beta - \frac{v_{xy}}{v_x^2}\right)^2 + \frac{v_x^2 v_y^2 - v_{xy}^2}{v_x^2} + (\bar{y} - (\alpha + \beta\bar{x}))^2\right)\right)
\end{aligned}$$

と書けるので,

$$t(y_1, \ldots, y_n|x_1, \ldots, x_n) = (\bar{x}, \bar{y}, v_x^2, v_y^2, v_{xy})$$

は少し拡張した意味での十分統計量だと言ってよいだろう.

注意: $(\bar{x}, \bar{y}, v_x^2, v_y^2, v_{xy})$ は2変量正規分布の標本分布モデルの十分統計量になっている.

4 記述統計 (要約統計)

$x_1, \ldots, x_n$ の形式のデータや $(x_1, y_1), \ldots, (x_n, y_n)$ の形式のデータが得られたとき, それらの情報を要約するためによく使われる量がある.

それらについて簡単にまとめておこう.

4.1 n 個の 1 と 0 からなるデータ $x_1, \ldots, x_n$ の要約

n 個の 1 と 0 からなるデータ $x_1, \ldots, x_n$ の要約として, $x_1, \ldots, x_n$ の中の 1 の個数 $k = x_1 + \cdots + x_n$ がよく使われる. 報告の形式は「n 個中 k 個が 1 である」になる. その k はBernoulli試行モデルの十分統計量になっているのであった.

そのようなデータの要約の仕方は $x_1, \ldots, x_n$ が並んでいる順序が重要な場合には適さない.

「n 個中の k 個が 1 である」と報告するときには n と k の両方の数値を報告することが重要である. なぜならばデータのサイズ n は統計分析の精度を見積もる上で非常に重要な情報だからである.

n の情報を省いて「1 の割合は 30% であった」のように k/n の情報だけを伝えることは好ましくない.

4.2 より一般の離散的なデータの要約

n 個の $1, 2, \ldots, r$ からなるデータ $x_1, \ldots, x_n$ の要約として, $x_1, \ldots, x_n$ の中含まれる $a = 1, 2, \ldots, r$ の個数 k_a がまとめた $(k_1, \ldots, k_r)$ がよく使われる. ($(k_1, \ldots, k_r)$ はカテゴリカル分布の標本分布モデルの十分統計量になっている.)

そのようなデータの要約の仕方は $x_1, \ldots, x_n$ が並んでいる順序が重要な場合には適さない.

データの要約(summary)を報告する場合には, 必ずデータのサイズがわかるように報告しなければいけない. なぜならばデータのサイズ n は統計分析の精度を見積もる上で非常に重要な情報だからである.

例えば「$1, 2, 3, 4$ の割合がそれぞれ $10\%, 20\%, 30\%, 40\%$ であった」のように k_a/n の情報だけを伝えることは好ましくない. 全体の割合の情報だけではなく, 全体の個数の情報も必ず報告しなければいけない.

4.3 n 個の実数からなるデータ $x_1, \ldots, x_n$ の要約

n 個の実数からなるデータ $x_1, \ldots, x_n$ の要約としては以下がよく使われている:

- 標本平均 $\bar{x}$ と不偏分散 s^2
- 中央値
- 四分位数

標本平均と不偏分散の組み合わせは正規分布の標本分布モデルの十分統計量になっており, それだけではなく, その最尤法の解にもなっているのであった.

標本平均だけではなく, 中央値の情報も報告されているとき, それらに大きなずれが存在するならば, 分布に非対称性があることもわかる.

四分位数は外れ値がある場合の分布の散らばり方の要約として頑健である.

以上で挙げた, 標本平均, 不偏分散, 中央値, 四分位数はデータの分布が単峰型の場合にはデータの要約の仕方として適切だが, 山が2以上ある分布の場合には適さない.

この型のデータの様子は, ヒストグラムや経験累積分布函数(x に x 以下の x_i 達の割合を対応させる函数)をプロットして, **データ全体の様子を視覚的に必ず確認した方がよい.**

4.4 n 個の実数の対からなるデータ $(x_1, y_1), \ldots, (x_n, y_n)$ の要約

n 個の実数の対からなるデータ $(x_1, y_1), \ldots, (x_n, y_n)$ の要約としてよく使われているのは,

- 標本平均 $\bar{x}, \bar{y}$ と不偏分散と不偏共分散 s_x^2, s_y^2, s_{xy}

である.　この5つの量を合わせたものは二変量正規分布の標本分布モデルの十分統計量になっている.

それらの統計量を使って線形回帰の結果を表示することもできる. 線形回帰もデータの要約の仕方だとみなされる.

このようなデータを取得する場合には x_i と y_i のあいだの関係がどうなっているかに興味がある.

そのときに上の5つの要約統計量(や線形回帰の結果)だけしか見ないと, x_i と y_i のあいだの重要な関係を見逃してしまうことがある.

この型のデータを扱う場合には, **散布図** (scatter plot)を描くなどして、**データ全体の様子を視覚的に必ず確認した方がよい**.

4.5 Anscombeの例(アンスコムの例, 再)

以下の散布図(scatter plot)を見れば, 要約統計量だけを見てはいけないことがわかる.

In [59]:

```
using RDatasets
anscombe = dataset("datasets", "anscombe")

SA1 = scatter(anscombe.X1, anscombe.Y1; title="Anscombe 1", label="", msw=0)
SA2 = scatter(anscombe.X2, anscombe.Y2; title="Anscombe 2", label="", msw=0)
SA3 = scatter(anscombe.X3, anscombe.Y3; title="Anscombe 3", label="", msw=0)
SA4 = scatter(anscombe.X4, anscombe.Y4; title="Anscombe 4", label="", msw=0)
plot(SA1, SA2, SA3, SA4; size=(500, 450), layout=(2,2),
    xlim=(3, 20), ylim=(2, 14), xlabel="x", ylabel="y")
```

Out[59]:

In [60]:

```julia
function plot_ols(x, y, α̂, β̂; xlim=nothing,
        title="data and regression line", kwargs...)
    a, b = extrema(x)
    isnothing(xlim) && (xlim = (a - 0.1(b-a), b + 0.1(b-a)))
    scatter(x, y; label="", msw=0)
    plot!(x -> α̂ + β̂*x, xlim...; label="", lw=2)
    plot!(; xlabel="x", ylabel="y", msw=0)
    plot!(; size=(400, 300))
    plot!(; title, kwargs...)
end

function solve_anscombe(x, y;
        title="", xlim=(3, 20), ylim=(2, 14), size=(250, 225))
    @show x̄ = mean(x)
    @show ȳ = mean(y)
    @show sx² = var(x)
    @show sy² = var(y)
    @show sxy = cov(x, y)
    β̂ = sxy/sx²
    α̂ = ȳ - β̂*x̄
    σ̂ = √((n - 1)/n*(sx²*sy² - sxy^2)/sx²)
    @show α̂ β̂ σ̂

    plot_ols(x, y, α̂, β̂; title, xlim, ylim, size)
end

println("---------- Anscombe 1")
PA1 = solve_anscombe(anscombe.X1, anscombe.Y1, title="Anscombe 1")
println("\n---------- Anscombe 2")
PA2 = solve_anscombe(anscombe.X2, anscombe.Y2, title="Anscombe 1")
println("\n---------- Anscombe 3")
PA3 = solve_anscombe(anscombe.X3, anscombe.Y3, title="Anscombe 1")
println("\n---------- Anscombe 4")
PA4 = solve_anscombe(anscombe.X4, anscombe.Y4, title="Anscombe 1")

plot(PA1, PA2, PA3, PA4; size=(500, 450), layout=(2,2))
```

```
---------- Anscombe 1
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500909090909093
sx² = var(x) = 11.0
sy² = var(y) = 4.127269090909091
sxy = cov(x, y) = 5.501
α̂ = 3.0000909090909103
β̂ = 0.5000909090909091
σ̂ = 1.1129429904536887

---------- Anscombe 2
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500909090909091
sx² = var(x) = 11.0
sy² = var(y) = 4.127629090909091
sxy = cov(x, y) = 5.5
α̂ = 3.000909090909091
β̂ = 0.5
σ̂ = 1.1134927848074192

---------- Anscombe 3
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.500000000000001
sx² = var(x) = 11.0
sy² = var(y) = 4.12262
sxy = cov(x, y) = 5.497
α̂ = 3.0024545454545466
β̂ = 0.49972727272727274
σ̂ = 1.1126802162509963

---------- Anscombe 4
x̄ = mean(x) = 9.0
ȳ = mean(y) = 7.50090909090909
sx² = var(x) = 11.0
sy² = var(y) = 4.12324909090909
sxy = cov(x, y) = 5.499
α̂ = 3.0017272727272726
β̂ = 0.49990909090909086
σ̂ = 1.112125937113239
```

Out[60]:

4.6 問題: DataSaurusの例

以下のデータは

- http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html (http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html)

から取得したものである.

- CSVファイル (https://raw.githubusercontent.com/genkuroki/Statistics/master/2022/data/Datasaurus_data.csv)
- x座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X.txt), y座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y.txt)
- x座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X_with_commas.txt), y座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y_with_commas.txt)

以下のデータの左列はx座標のデータであり, 右列はy座標のデータである.

(1) 標本平均 $\bar{x}$, $\bar{y}$ と不偏分散と不偏共分散 s_x^2, s_y^2, s_{xy} を小数点以下2桁以上求めよ.

(2) 何らかの方法で散布図(scatter plot)を描け.

さらに datasaurus same stats についてインターネットで検索して統計分析で注意するべきことを学べるサイトを見つけて内容を理解するように努力せよ.

ヒント: (1),(2)についてはインターネット上で適切に検索すれば以上の問題を解くことができるサイトを発見できる.

In [61]:

```python
datasaurus = [
    55.3846 97.1795
    51.5385 96.0256
    46.1538 94.4872
    42.8205 91.4103
    40.7692 88.3333
    38.7179 84.8718
    35.6410 79.8718
    33.0769 77.5641
    28.9744 74.4872
    26.1538 71.4103
    23.0769 66.4103
    22.3077 61.7949
    22.3077 57.1795
    23.3333 52.9487
    25.8974 51.0256
    29.4872 51.0256
    32.8205 51.0256
    35.3846 51.4103
    40.2564 51.4103
    44.1026 52.9487
    46.6667 54.1026
    50.0000 55.2564
    53.0769 55.6410
    56.6667 56.0256
    59.2308 57.9487
    61.2821 62.1795
    61.5385 66.4103
    61.7949 69.1026
    57.4359 55.2564
    54.8718 49.8718
    52.5641 46.0256
    48.2051 38.3333
    49.4872 42.1795
    51.0256 44.1026
    45.3846 36.4103
    42.8205 32.5641
    38.7179 31.4103
    35.1282 30.2564
    32.5641 32.1795
    30.0000 36.7949
    33.5897 41.4103
    36.6667 45.6410
    38.2051 49.1026
    29.7436 36.0256
    29.7436 32.1795
    30.0000 29.1026
    32.0513 26.7949
    35.8974 25.2564
    41.0256 25.2564
    44.1026 25.6410
    47.1795 28.7180
    49.4872 31.4103
    51.5385 34.8718
    53.5897 37.5641
    55.1282 40.6410
    56.6667 42.1795
    59.2308 44.4872
    62.3077 46.0256
    64.8718 46.7949
    67.9487 47.9487
    70.5128 53.7180
    71.5385 60.6410
    71.5385 64.4872
    69.4872 69.4872
    46.9231 79.8718
    48.2051 84.1026
    50.0000 85.2564
    53.0769 85.2564
    55.3846 86.0256
    56.6667 86.0256
    56.1538 82.9487
    53.8462 80.6410
    51.2821 78.7180
    50.0000 78.7180
    47.9487 77.5641
    29.7436 59.8718
```

```
    29.7436 62.1795
    31.2821 62.5641
    57.9487 99.4872
    61.7949 99.1026
    64.8718 97.5641
    68.4615 94.1026
    70.7692 91.0256
    72.0513 86.4103
    73.8462 83.3333
    75.1282 79.1026
    76.6667 75.2564
    77.6923 71.4103
    79.7436 66.7949
    81.7949 60.2564
    83.3333 55.2564
    85.1282 51.4103
    86.4103 47.5641
    87.9487 46.0256
    89.4872 42.5641
    93.3333 39.8718
    95.3846 36.7949
    98.2051 33.7180
    56.6667 40.6410
    59.2308 38.3333
    60.7692 33.7180
    63.0769 29.1026
    64.1026 25.2564
    64.3590 24.1026
    74.3590 22.9487
    71.2821 22.9487
    67.9487 22.1795
    65.8974 20.2564
    63.0769 19.1026
    61.2821 19.1026
    58.7179 18.3333
    55.1282 18.3333
    52.3077 18.3333
    49.7436 17.5641
    47.4359 16.0256
    44.8718 13.7180
    48.7179 14.8718
    51.2821 14.8718
    54.1026 14.8718
    56.1538 14.1026
    52.0513 12.5641
    48.7179 11.0256
    47.1795  9.8718
    46.1538  6.0256
    50.5128  9.4872
    53.8462 10.2564
    57.4359 10.2564
    60.0000 10.6410
    64.1026 10.6410
    66.9231 10.6410
    71.2821 10.6410
    74.3590 10.6410
    78.2051 10.6410
    67.9487  8.7180
    68.4615  5.2564
    68.2051  2.9487
    37.6923 25.7692
    39.4872 25.3846
    91.2821 41.5385
    50.0000 95.7692
    47.9487 95.0000
    44.1026 92.6923
];
```

注意: この例についての詳しい解説が

- Justin Matejka, George Fitzmaurice. Same Stats, Different Graphs: Generating Datasets with Varied Appearance and Identical Statistics through Simulated Annealing. Honourable Mention, ACM SIGCHI Conference on Human Factors in Computing Systems, 2017 [link (https://www.autodesk.com/research/publications/same-stats-different-graphs)]

にある. この解説は非常に面白い. データ全体の様子がわかるようなグラフを作成することが統計分析では必須なことがよくわかる解説になっている.

このサイトは "datasaurus same stats" をインターネットで検索すればすぐに見つかる.

4.6.1 Julia言語を使う解答例

In [62]:
```
# (1)
x, y = datasaurus[:,1], datasaurus[:,2]
@show x̄ = mean(x)
@show ȳ = mean(y)
@show sx² = var(x)
@show sy² = var(y)
@show sxy = cov(x,y);
```

```
x̄ = mean(x) = 54.26327323943663
ȳ = mean(y) = 47.832252816901395
sx² = var(x) = 281.06998759176105
sy² = var(y) = 725.5159610042119
sxy = cov(x, y) = -29.11393267347017
```

In [63]:
```
# (2)
scatter(x, y; label="", title="Datasaurus")
```

Out[63]:

こんなことになっていることは要約統計量たちを眺めてもわからない!

データを得たら, 必ず全体の様子を何らかの視覚化によって確認しなければいけない.

これは絶対に必要なことである.

4.6.2 Julia言語の使い方

(1) https://julialang.org/downloads/ (https://julialang.org/downloads/) から最新安定板 (current stable release)のインストーラをダウンロードして実行し, 最新安定板の公式バイナリを自分のパソコンにインストールする.

(2) juliaを起動して, 適当にコードを打ち込んで正常に計算できることを確認する.

```
julia> sinpi(1/6)
0.5
```

(3) 以下を実行してPlots.jlパッケージ (https://github.com/JuliaPlots/Plots.jl)をJuliaの環境に入れる.

```
julia> ]
pkg> add Plots
ここにインストールの様子が表示される.
pkg> ここでバックスペースキーを押す.
julia>
```

(4) 以下を実行して, パッケージを読み込む.

```
julia> using Statistics
julia> using Plots
```

`using Plots` には少しだけ時間を取られる. `julia>` の部分を含めてコピー&ペーストしてもその部分が自動的に削除されて正常に実行される.

(4) `datasaurus = [ ... ]` のセルの内容を `julia>` と表示されている画面にコピー&ペーストして実行する.

(5) 以下を順次実行する.

```
julia> x = datasaurus[:,1];
julia> y = datasaurus[:,2];
julia> mean(x)
julia> mean(y)
julia> var(x)
julia> var(y)
julia> cov(x,y)
```

これで問題の(1)の答えが得られる.

(6) 次を実行すれば問題(2)の答えが得られる.

```
julia> scatter(x, y)
```

これも最初の実行のときに少し時間が取られる. 2回目以降は瞬時に実行される.

注意: 複雑な計算を十分にできるソフトを自分のパソコンに入れておいた方がよい. もちろんそれはJulia言語である必要はない. 統計学を学習するという目的のためにはR言語やPython言語もお勧めである.

4.6.3 インターネット上の各種計算サイトを使う方法

jp.foxcalculators.com (https://jp.foxcalculators.com/)とkeisan.casio.jp (https://keisan.casio.jp)を利用してみよう.

(1) 平均と不偏分散と不偏共分散はjp.foxcalculators.comの共分散計算機 (https://jp.foxcalculators.com/math/3003.html)で計算できる. データの入力は

- x座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X_with_commas.txt), y座標のみコンマ付き (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y_with_commas.txt)

からコピー＆ペーストすればよい. 不偏分散が不偏共分散の特別な場合であることに注意せよ. (x_i と x_i の不偏共分散は x_i の不偏分散になる.)

- $\bar{x} \approx 54.263$
- $s_x^2 \approx 281.07$

- $\bar{y} \approx 47.832$
- $s_y^2 \approx 725.516$

$s_{xy} \approx -29.114$

(2) 散布図はkeisan.casio.jpでの散布図の描画 (https://keisan.casio.jp/exec/system/1412059302)で描ける. データの入力は

- x座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_X.txt), y座標のみ (https://github.com/genkuroki/Statistics/blob/master/2022/data/Datasaurus_Y.txt)

からx座標とy座標を別々にコピー＆ペーストすればよい.

散布図の描画

気温別のコーヒーとアイスの売上から散布図で描画します。

	X	Y
1	55.3846	97.1795
2	51.5385	96.0256
3	46.1538	94.4872
4	42.8205	91.4103
5	40.7692	88.3333
6	38.7179	84.8718
7	35.6410	79.8718
8	33.0769	77.5641
9	28.9744	74.4872
10	26.1538	71.4103

In [ ]:
```
1
```

例：ベータ函数と二項分布の関係とその応用

- 黒木玄

- 2022-05-27〜2022-06-08

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 ベータ分布と二項分布の累積分布函数の関係
  - 1.1 ベータ分布と二項分布の累積分布函数の関係の証明
  - 1.2 基本特殊函数ライブラリを使うと効率的に計算できる
  - 1.3 ベータ分布と二項分布の累積分布函数の関係の別証明
  - 1.4 累積分布函数と分位点函数について
  - 1.5 パラメータに関する区間推定での利用の仕方
  - 1.6 まとめ
- 2 二項分布モデルでのClopper-Pearsonの信頼区間
  - 2.1 Clopper-Pearsonの信頼区間の定義
  - 2.2 問題: n=100, k=30 の95%Clopper-Pearson信頼区間
  - 2.3 問題: n=400, k=120 の95% Clopper-Pearson信頼区間
- 3 関連の問題
  - 3.1 問題: Waldの信頼区間との比較
  - 3.2 Waldの信頼区間の基礎
  - 3.3 問題: Waldの信頼区間が0〜1の範囲をはみ出す場合
  - 3.4 問題: Wilsonの信頼区間との比較
  - 3.5 Wilsonの信頼区間の基礎
  - 3.6 Sterneの信頼区間について
  - 3.7 問題: Poisson分布とガンマ分布の関係
- 4 おまけ: 二項分布モデルのBayes統計との関係
  - 4.1 二項分布モデルのBayes統計
  - 4.2 事前分布が共役事前分布(ベータ分布)の場合
  - 4.3 二項分布モデルでの片側P値のBayes統計での解釈
  - 4.4 Pólyaの壺との関係

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using Distributions
using LinearAlgebra
using Printf
using QuadGK
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, titlefontsize = 10, size = (400, 250))
using SymPy
```

In [2]:

```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```
safemul(x, y) = x == 0 ? x : x*y
safediv(x, y) = x == 0 ? x : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 4) - 3
```

Out[3]: `mykurtosis (generic function with 2 methods)`

1 ベータ分布と二項分布の累積分布函数の関係

このノートではすでに紹介済み(「確率分布達の解釈」のノートの「ベータ分布の一様乱数生成の繰り返しによる解釈」の節を参照)のベータ分布と二項分布の累積分布函数の関係について再度説明することにする. 再度説明する理由は複数ある:

- 二項分布モデルでの信頼区間の計算で使われるから.
- 異なる確率分布のあいだの関係の典型的な例になっているから.
- コンピュータによる確率分布の効率的な計算とも関係しているから.
- その他にも色々なことと関係しているから.

以下で証明したいこと: $k = 1, 2, \ldots, n$ のとき,

$$\begin{aligned}&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})\\ &= (\text{ベータ分布 } \mathrm{Beta}(k, n-k+1) \text{ において } p \text{ 以下になる確率})\end{aligned}$$

これは

$$1 - \mathrm{cdf}(\mathrm{Binomial}(n,p), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p)$$

や次のように書き直される:

$$\sum_{i \geq k} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)}.$$

補事象を取り, 「k 未満になる」という条件を「k 以下になる」に置き換えるために, k を $k+1$ で置き換えると, 以上の結果は以下と同値である: $k = 0, 1, \ldots, n-1$ のとき,

$$\begin{aligned}&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以下になる確率})\\ &= (\text{ベータ分布 } \mathrm{Beta}(k+1, n-k) \text{ において } p \text{ 以上になる確率})\end{aligned}$$

これは

$$\mathrm{cdf}(\mathrm{Binomial}(n,p), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p)$$

や次のように書き直される:

$$\sum_{i \leq k} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_p^1 t^k (1-t)^{n-i-1}\, dt}{B(k+1, n-k)}.$$

1.1 ベータ分布と二項分布の累積分布函数の関係の証明

$k = 1, 2, \ldots, n$ と仮定し, 次が成立することを示そう:

$$\begin{aligned}&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})\\ &= (\text{ベータ分布 } \mathrm{Beta}(k, n-k+1) \text{ において } p \text{ 以下になる確率})\end{aligned}$$

証明: $p = t$ のときの左辺(二項分布 $\mathrm{Binomial}(n,t)$ で成功回数が k 以上になる確率)を $F(t)$ と書くと,

$$F(t) = \sum_{i \geq k} \binom{n}{i} t^i (1-t)^{n-i} = \sum_{i \geq k} \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i}.$$

このとき, $k > 0$ より, $F(0) = 0$ であり,

$$\begin{aligned}F'(t) &= \sum_{i \geq k} \frac{n!}{i!(n-i)!} i t^{i-1} (1-t)^{n-i} - \sum_{i \geq k} \frac{n!}{i!(n-i)!} (n-i) t^i (1-t)^{n-i-1}\\ &= \sum_{i \geq k} \frac{n!}{(i-1)!(n-i)!} t^{i-1} (1-t)^{n-i} - \sum_{i \geq k} \frac{n!}{i!(n-i-1)!} t^i (1-t)^{n-i-1}\\ &= \sum_{i \geq k} \frac{n!}{(i-1)!(n-i)!} t^{i-1} (1-t)^{n-i} - \sum_{i \geq k+1} \frac{n!}{(i-1)!(n-i)!} t^{i-1} (1-t)^{n-i}\\ &= \frac{n!}{(k-1)!(n-k)!} t^{k-1} (1-t)^{n-k}.\end{aligned}$$

3つめの等号で2つめの和の i を $i-1$ に置き換えた. さらに, $\Gamma(a+1) = a!$ と $\Gamma(a)\Gamma(b) = B(a,b)\Gamma(a+b)$ を使うと,

$$\frac{n!}{(k-1)!(n-k)!} = \frac{\Gamma(n+1)}{\Gamma(k)\Gamma(n-k+1)} = \frac{1}{B(k, n-k+1)}$$

なので,

$$F'(t) = \frac{t^{k-1}(1-t)^{n-k}}{B(k, n-k+1)}.$$

これと $F(0) = 0$ を合わせると, $F'(t)$ を $t = 0$ から $t = p$ まで積分することによって,

$$F(p) = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\, dt}{B(k, n-k+1)}.$$

これは, ベータ分布 $\mathrm{Beta}(k, n-k+1)$ において p 以下になる確率である.

証明終

1.2 基本特殊函数ライブラリを使うと効率的に計算できる

以上によって, 次の公式が示された:

$$\sum_{i \geq k} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\, dt}{B(k, n-k+1)}.$$

次は誤解である:

- 左辺の和よりも右辺の積分の方が計算が大変である.

実際には, 右辺の方が効率的に計算できる!

- 左辺の和の計算は $n \geq 10^5$ のような場合にはそれなりに大変である.
- 左辺の計算は一度だけ計算すればよいのであれば大変ではない.
- しかし, 繰り返し計算する場合には時間がかかってしまう場合がある.
- 左辺は正則化不完全ベータ函数 $I_p(k, n-k+1)$ であり, 必要な計算量は小さい.
- 実際, 正則化不完全ベータ函数は基本特殊函数ライブラリに含まれている.

In [4]:
```
n, k, p = 10^6, 123456, 0.123
```
Out[4]:
```
(1000000, 123456, 0.123)
```

この場合には, 左辺の和は単純に計算しようとするとそもそも計算できない.

```
sum(binomial(n, i) * p^i * (1-p)^(n-i) for i in k:n)
```

```
OverflowError: binomial(1000000, 123456) overflows
```

次のように工夫すれば計算できる.

In [5]:
```
F_logsumexp(n, k, p) = exp(logsumexp(
        logabsbinomial(n, i)[1] + i*log(p) + (n-i)*log(1-p) for i in k:n))
F_logsumexp(n, k, p)
```
Out[5]:
```
0.0827943265753984
```

正則化不完全ベータ函数という名の特殊函数による計算は以下の通り.

In [6]:
```
F_beta_inc(n, k, p) = beta_inc(k, n-k+1, p)[1]
F_beta_inc(n, k, p)
```
Out[6]:
```
0.08279432657965562
```

Julia言語のDistributions.jlによる計算結果は上とぴったり同じになる.

In [7]:
```
F_distributionsjl(n, k, p) = ccdf(Binomial(n, p), k-1)
F_distributionsjl(n, k, p)
```
Out[7]:
```
0.08279432657965562
```

計算にかかる時間の計測:

In [8]:
```
@btime F_logsumexp(n, k, p)
@btime F_beta_inc(n, k, p);
```
```
  81.692 ms (1 allocation: 16 bytes)
  424.242 ns (5 allocations: 976 bytes)
```

この場合には, 正則化不完全ベータ函数を使った方が計算が約200倍以上速くなる!

1.3 ベータ分布と二項分布の累積分布函数の関係の別証明

0 から 1 のあいだの値になる一様乱数(コンピュータでの `rand()` と同じだと思ってよい)を n 個独立に生成した結果を $T_1, T_2, \ldots, T_n$ と書き, その中で k 番目に小さな値を $T_{(k)}$ と書く. T_i や $T_{(i)}$ は確率変数とみなされる.

$$\begin{aligned}(T_{(k)} \leq p \text{ となる確率}) &= (T_1, T_2, \ldots, T_n \text{ の中に } p \text{ 以下のものが } k \text{ 個以上ある確率}) \\ &= (\text{二項分布 } \mathrm{Binomial}(n, p) \text{ で成功回数が } k \text{ 以上になる確率})\end{aligned}$$

が成立している. 一方, $T_{(k)}$ が $[t, t+dt]$ に含まれる確率は dt に関する高次の微小量を無視する近似で

$$\frac{n!}{(k-1)!1!(n-k)!} t^{k-1}\, dt\, (1-t)^{n-k} = \frac{t^{k-1}(1-t)^{(n-k+1)-1}\, dt}{B(k, n-k+1)}$$

になる. なぜならば, $T_{(k)}$ が $[t, t+dt]$ に含まれることは, 大雑把には, $T_1, T_2, \ldots, T_n$ の中に t 未満のものが $k-1$ 個存在し, 1 個が $[t, t+dt]$ に含まれ, それ以外の $n-k$ 個が t より大きくなることと同値である. このことから, 左辺が得られる. $n!/((k-1)!1!(n-k)!)$ は $T_1, T_2, \ldots, T_n$ の $k-1$ 個, 1 個, $n-k$ 個へのグループ分けの仕方の個数になっている.

これで, $T_{(k)}$ は確率密度函数

$$p(t) = \frac{t^{k-1}(1-t)^{(n-k+1)-1}}{B(k, n-k+1)}$$

を持つことがわかった. すなわち, $T_{(k)} \sim \mathrm{Beta}(k, n-k+1)$ である. ゆえに,

$$(T_{(k)} \leq p \text{ となる確率}) = (\text{ベータ分布 } \mathrm{Beta}(k, n-k+1) \text{ において } p \text{ 以下になる確率})$$

以上を合わせると,

$$\begin{aligned}&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})\\ &= (\text{ベータ分布 } \mathrm{Beta}(k, n-k+1) \text{ において } p \text{ 以下になる確率})\end{aligned}$$

証明終

注意: 以上の証明法の利点は以下の通り:

- 二項分布の累積分布函数をパラメータ p で微分するという面倒な計算を避けられる.
- ベータ函数の逆数 $1/B(a,b) = \Gamma(a+b)/\Gamma(a)\Gamma(b)$ が $\Gamma(a) = (a-1)!$ であり, 階乗とぴったり一致しないせいで, 二項係数に一致しない理由もわかる. ベータ函数の逆数は二項係数ではなく, n 個の $k-1$ 個, 1 個, $n-k$ 個へのグループ分けの仕方の個数(多項係数の特別な場合)になっている.
- 上と同様の方法ならば, 複数の $T_{(k)}$ 達の同時確率分布が本質的にDirichlet分布になっていることも容易に確認できる.

1.4 累積分布函数と分位点函数について

分布 D に従う確率変数 X について x の函数

$$\mathrm{cdf}(D, x) = F_D(x) = P(X \leq x)$$

を確率変数 X が従う分布 D の **累積分布函数** (cumulative distribution function, cdf)と呼び, その逆函数(もしくは逆函数の適切な類似物)

$$\mathrm{quantile}(D, p) = Q_D(p) = F_X^{-1}(p)$$

を確率変数 X が従う分布 D の **分位点函数** (quantile function)と呼ぶのであった.

これらの函数はコンピュータにおいて確率分布を扱うライブラリの中で定義されていることが多い.

例えば, Julia言語のDistributions.jlパッケージでは

```
ccdf(Binomial(10^6, 0.123), 123456-1)
cdf(Beta(123456, 10^6-123456+1), 0.123)
quantile(Beta(123456, 10^6-123456+1), 0.0828)
```

のように書く. ただし, $\mathrm{ccdf}(D,x) = 1 - \mathrm{cdf}(D,x)$ であり, `ccdf(Binomial(10^6, 0.123), 123456-1)` は二項分布 $\mathrm{Binomial}(10^6, 0.123)$ で 123456 以上になる確率に一致する.

In [9]:
```
ccdf(Binomial(10^6, 0.123), 123456-1) # 下と一致
```
Out[9]:
```
0.08279432657965562
```

In [10]:
```
cdf(Beta(123456, 10^6-123456+1), 0.123) # 上と一致
```
Out[10]:
```
0.08279432657965562
```

In [11]:
```
quantile(Beta(123456, 10^6-123456+1), 0.0828)
```
Out[11]:
```
0.12300001220665846
```

WolframAlphaでは

```
1 - CDF[BinomialDistribution[10^6, 0.123], 123455]
CDF[BetaDistribution[123456, 10^6 -  123456 + 1], 0.123]
Quantile[BetaDistribution[123456, 10^6 -  123456 + 1], 0.0828]
```

のように入力すれば計算してくれる. WolframAlphaでは表記の揺れ(大文字小文字, 括弧の違い)があっても適切に解釈して計算してくれる:

- 1 - cdf(BinomialDistribution(10^6, 0.123), 123455) (https://www.wolframalpha.com/input?i=1+-+cdf%28BinomialDistribution%2810%5E6%2C+0.123%29%2C+123455%29&lang=ja) → 0.0827943 (下と一致)
- cdf(BetaDistribution(123456, 10^6 - 123456 + 1), 0.123) (https://www.wolframalpha.com/input?i=cdf%28BetaDistribution%28123456%2C+10%5E6+-++123456+%2B+1%29%2C+0.123%29) → 0.0827943 (上と一致)
- quantile(BetaDistribution(123456, 10^6 - 123456 + 1), 0.0828) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28123456%2C+10%5E6+-+123456+%2B+1%29%2C+0.0828%29) → 0.123

これらの函数ではすでに特殊函数を使った効率的な実装がされており, 使用可能ならばこれを使うとよい.

1.5 パラメータに関する区間推定での利用の仕方

コンピュータにおける確率分布を扱うライブラリ(パッケージ)を利用すれば, 確率分布の累積分布函数やその逆函数である分位点函数を効率的に計算できる.

しかし, 統計学においては, パラメータ θ を持つ分布 $D(\theta)$ とその分布で扱えるデータの数値 x から定まるパラメータ θ の函数

$$\theta \mapsto (\text{分布 } D(\theta) \text{ における数値 } x \text{ から決まる確率})$$

の逆函数が必要になることがある.

例えば, 「n 回中 k 回成功」というデータが得られたとき, それを二項分布モデル $\mathrm{Binomial}(n,p)$ で扱うときに, 成功確率 p の函数

$$p \mapsto (\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})$$

の逆函数が成功確率 p の区間推定(Clopper-Pearsonの信頼区間)のために必要になる.

データの数値からモデルのパラメータを推定することは統計学では最も普通に行われることなので, このような場合は非常に多い.

一般には上のようにして定義されたパラメータの逆函数の計算は面倒だが, 二項分布モデルの場合にはベータ分布との関係を用いて極めて効率的にそれを計算できる!

なぜならば, すでに示したように

$$(\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率}) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p)$$

が成立しているからである. 左辺でパラメータであった p が右辺では累積分布函数の引数になっている. ゆえに,

$$p \mapsto a = (\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})$$

の逆函数は

$$a \mapsto p = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), a)$$

になる. これは確率分布を扱うライブラリで効率的に計算可能である!

1.6 まとめ

次が成立している:

$$\begin{aligned}
\sum_{i \geq k} \binom{n}{i} p^i (1-p)^{n-i} &= \frac{\int_0^p t^{k-1}(1-t)^{n-i}\,dt}{B(k, n-k+1)}, \\
\sum_{i \leq k} \binom{n}{i} p^i (1-p)^{n-i} &= \frac{\int_p^1 t^{k}(1-t)^{n-i-1}\,dt}{B(k+1, n-k)}.
\end{aligned}$$

すなわち, 以下が成立している:

$$\begin{aligned}
&(\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率}) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p), \\
&(\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以下になる確率}) = \mathrm{ccdf}(\mathrm{Beta}(k+1, n-k), p).
\end{aligned}$$

ここで $a = \mathrm{ccdf}(D, x) = 1 - \mathrm{cdf}(D, x)$ であり, これの逆函数は $x = \mathrm{quantile}(D, 1-a)$ になる.

したがって,

$$p \mapsto a = (\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上になる確率})$$

の逆函数は

$$a \mapsto p = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), a)$$

になり,

$$p \mapsto a = (\text{分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以下になる確率})$$

の逆函数は

$$a \mapsto p = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-a)$$

になる. これらの逆函数はコンピュータにおける確率分布を扱うライブラリ(パッケージ)を使えば効率的にかつ楽に計算できる.

次節で使う図の準備

一般に図の準備は相当に面倒である.

In [12]:
```
n, k = 100, 30
kmin, kmax = 0, 65

p = 0.27
var"p = 0.27 は n = 100, k = 30 にそこそこ適合している" = plot(; size=(500, 300))
plot!(k -> mypdf(Binomial(n, p), k), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(k -> mypdf(Binomial(n, p), k), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+6, 0.03, text("probability of K ≥ $k", :blue, :left, 10))
title!("Binomial(n=$n, p=$p)")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p]; label="np=$(n*p)", c=:blue, ls=:dash)
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

In [13]:
```
n, k = 100, 30
kmin, kmax = 0, 65

p = 0.19
var"p = 0.19 は n = 100, k = 30 にあまりにも適合しない" = plot(; size=(500, 300))
plot!(k -> mypdf(Binomial(n, p), k), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(k -> mypdf(Binomial(n, p), k), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+1, 0.01, text("probability of K ≥ $k", :blue, :left, 10))
title!("Binomial(n=$n, p=$p)")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p]; label="np=$(n*p)", c=:blue, ls=:dash)
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

In [14]:
```
p = 0.42
var"p = 0.42 は n = 100, k = 30 にあまりにも適合しない" = plot(; size=(500, 300))
plot!(k -> mypdf(Binomial(n, p), k), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(k -> mypdf(Binomial(n, p), k), kmin-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
annotate!(k-1, 0.01, text("probability of K ≤ $k", :red, :right, 10))
title!("Binomial(n=$n, p=$p)")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p]; label="np=$(n*p)", c=:red, ls=:dashdot)
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

In [15]:
```
α = 0.05
p_L = quantile(Beta(k, n-k+1), α/2)
var"下側では p = p_L がぎりぎり" = plot(; size=(500, 300))
plot!(k -> mypdf(Binomial(n, p_L), k), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(k -> mypdf(Binomial(n, p_L), k), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(100α/2)%", :blue, :left, 10))
title!("Binomial(n=$n, p=p_L),  p_L=$(round(p_L; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

In [16]:
```
p_U = quantile(Beta(k+1, n-k), 1 - 0.025)
var"上側では p = p_U がぎりぎり" = plot(; size=(500, 300))
plot!(k -> mypdf(Binomial(n, p_U), k), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(k -> mypdf(Binomial(n, p_U), k), kmin-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
annotate!(k-2, 0.01, text("$(100α/2)%", :red, :right, 10))
title!("Binomial(n=$n, p=p_U),  p_U=$(round(p_U; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_U]; label="n p_L = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

In [17]:

```
var"n × 95%信頼区間" = plot(; size=(500, 300))

plot!(k → mypdf(Binomial(n, p_L), k), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(k → mypdf(Binomial(n, p_L), k), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(100α/2)%", :blue, :left, 10))

plot!(k → mypdf(Binomial(n, p_U), k), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(k → mypdf(Binomial(n, p_U), k), kmin-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
annotate!(k-2, 0.01, text("$(100α/2)%", :red, :right, 10))

vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
vline!([n*p_U]; label="n p_U = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!([n*p_L, n*p_U], fill(-0.003, 2); label="", c=:red, lw=5)
title!("n = $n,  [n p_L, n p_U] = [$(round(n*p_L; digits=2)), $(round(n*p_U; digits=2))]")
plot!(; xguide="K", ylim=(-0.007, 0.105));
```

2 二項分布モデルでのClopper-Pearsonの信頼区間

信頼区間については後でもっと詳しく扱う.

以下では二項分布モデルにおけるClopper-Pearsonの信頼区間について必要最小限の事柄を説明する.

2.1 Clopper-Pearsonの信頼区間の定義

「n 回中 k 回成功」の型のデータが得られたとする.

このデータの数値と二項分布モデルを比較して, データの数値にあまりにも適合しない成功確率パラメータ値 p を排除して得られる区間を成功確率パラメータ p の **信頼区間** と呼ぶ(もしくは **比率** p の信頼区間と呼んだりする).

より正確には, **信頼度** (もしくは信頼係数) $1-\alpha$ ($0<\alpha<1$ を **有意水準** と呼ぶ)という閾値を決めて, その閾値を超えてデータに適合しないようなパラメータ値を除いてできる区間を信頼度 $1-\alpha$ の信頼区間と呼ぶ. (データへのモデルの適合度の定義の仕方は別に決める. その決め方は複数あり, 以下ではClopper-Pearsonの信頼区間の場合について説明する.)

$\alpha=5\%$ のときには, 単に 95% 信頼区間と呼ぶことが多い.

二項分布モデルで伝統的によく使われているClopper-Pearsonの信頼区間における「あまりにも適合しない」の定義は以下の通り:

- 二項分布モデル $\mathrm{Binomial}(n,p)$ において成功回数が k 回以上になる確率は $\alpha/2$ 未満である.
- 二項分布モデル $\mathrm{Binomial}(n,p)$ において成功回数が k 回以下になる確率は $\alpha/2$ 未満である.

このどちらかの条件を満たす p 達を除いてできる区間が信頼係数 $1-\alpha$ の **Clopper-Pearsonの信頼区間** と呼ばれる.

In [18]:

```
var"p = 0.27 は n = 100, k = 30 にそこそこ適合している"
```

Out[18]:

In [19]:
```
var"p = 0.19 は n = 100, k = 30 にあまりにも適合しない"
```

Out[19]:

In [20]:
```
var"p = 0.42 は n = 100, k = 30 にあまりにも適合しない"
```

Out[20]:

Clopper-Pearsonの信頼区間は具体的には以下のように計算される.

- 二項分布モデル $\mathrm{Binomial}(n,p)$ において成功回数が k 以上になる確率が $\alpha/2$ に等しくなる p を求め p_L と書く.
- 二項分布モデル $\mathrm{Binomial}(n,p)$ において成功回数が k 以下になる確率が $\alpha/2$ に等しくなる p を求め p_U と書く.
- Clopper-Pearsonの信頼区間は $[p_L, p_U]$ になる.

以下は $\alpha = 5\%$ の場合の $p_L = p_L$ と $p_U = p_U$ の図である.

In [21]:
```
var"下側では p = p_L がぎりぎり"
```

Out[21]:

In [22]:
```
var"上側では p = p_U がぎりぎり"
```

Out[22]:

In [23]:
```
var"n × 95%信頼区間"
```

Out[23]:

2.2 問題: n=100, k=30 の95%Clopper-Pearson信頼区間

「$n = 100$ 回中 $k = 30$ 回成功」というデータから決まる 95% Clopper-Pearson信頼区間を求めよ. (値は小数点以下第4桁まで求めよ.)

解答例: データ $n = 100$, $k = 30$ から決まる 95% Clopper-Pearson信頼区間は

$$[0.2124, 0.3998]$$

である. (計算の仕方については以下を見よ.)

解答終

In [24]:
```
# n回中k回成功というデータが得られたとする
n, k = 100, 30
```

Out[24]: (100, 30)

In [25]:
```
# 有意水準を5%に設定 (α は \alpha TAB で入力できる)
α = 0.05
```

Out[25]: 0.05

In [26]:
```
# 二項分布Binomial(n,p)でk以上になる確率が2.5%になるp
p_L = quantile(Beta(k, n-k+1), α/2)
```

Out[26]: 0.21240642048953662

In [27]:
```
# 検算: 分布 Binomial(n, p_L) で k 以上になる確率
ccdf(Binomial(n, p_L), k-1)
```

Out[27]: 0.025000000000000026

In [28]:
```
# 二項分布Binomial(n,p)でk以下になる確率が2.5%になるp
p_U = quantile(Beta(k+1, n-k), 1 - α/2)
```

Out[28]: 0.39981467617980404

In [29]:
```
# 検算: 分布 Binomial(n, p_U) で k 以下になる確率
cdf(Binomial(n, p_U), k) # k
```

Out[29]: 0.02500000000000001

In [30]:
```
# n回中k回成功というデータから決まるClopper-Pearsonの信頼区間
@show [p_L, p_U];
```

```
[p_L, p_U] = [0.21240642048953662, 0.39981467617980404]
```

In [31]:
```
@show round.([p_L, p_U]; digits=4);
```

```
round.([p_L, p_U]; digits = 4) = [0.2124, 0.3998]
```

以上で求めたClopper-Pearson信頼区間は R (https://cran.r-project.org/) の `binom.test` の結果に一致する.

```
> binom.test(30, 100)

    Exact binomial test

data:  30 and 100
number of successes = 30, number of trials = 100, p-value = 7.85e-05
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.2124064 0.3998147
sample estimates:
probability of success
                   0.3
```

WolframAlphaで求めるには以下のようにすればよい.

- quantile(BetaDistribution(30, 71), 0.025) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%2830%2C+71%29%2C+0.025%29) → 0.212406
- quantile(BetaDistribution(31, 70), 0.975) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%2831%2C+70%29%2C+0.975%29) → 0.399815

2.3 問題: n=400, k=120 の95% Clopper-Pearson信頼区間

前節の例は「$n = 100$ 回中 $k = 30$ 回成功」というデータから決まる 95% Clopper-Pearson信頼区間であった.

n を 4 倍した「$n = 400$ 回中 $k = 120$ 回成功」というデータから決まる 95% Clopper-Pearson信頼区間を求め, n を4倍すると信頼区間の幅は約半分になることを確認せよ. (値は小数点以下第4桁まで求めよ.)

注意: 大雑把には, 信頼区間の幅は $\sqrt{n}$ に反比例する. 例えば, 精度を1桁上げるためには, n を 100 倍する必要がある.

解答例: データ $n = 100$, $k = 30$ から決まる 95% Clopper-Pearson信頼区間は

$$[0.2124, 0.3998]$$

であった. データ $n = 400$, $k = 120$ から決まる 95% Clopper-Pearson信頼区間は

$$[0.2555, 0.3475]$$

になる. 確かに幅が約半分になっているように見える.

解答終

In [32]:
```
# n回中k回成功というデータが得られたとする
n, k = 400, 120
```

Out[32]: (400, 120)

In [33]:
```
# 有意水準を5%に設定 (α は \alpha TAB で入力できる)
α = 0.05
```

Out[33]: 0.05

In [34]:
```
# 二項分布Binomial(n,p)でk以上になる確率が2.5%になるp
p_L = quantile(Beta(k, n-k+1), α/2)
```

Out[34]: 0.25546672809530985

In [35]:
```
# 検算
ccdf(Binomial(n, p_L), k-1)
```

Out[35]: 0.025000000000000112

In [36]:
```
# 二項分布Binomial(n,p)でk以下になる確率が2.5%になるp
p_U = quantile(Beta(k+1, n-k), 1 - α/2)
```

Out[36]: 0.3475218790736496

In [37]:
```
# 検算
cdf(Binomial(n, p_U), k)
```

Out[37]: 0.025000000000000085

In [38]:
```
# n回中k回成功というデータから決まるClopper-Pearsonの信頼区間
@show [p_L, p_U];
```

```
[p_L, p_U] = [0.25546672809530985, 0.3475218790736496]
```

In [39]:
```
@show round.([p_L, p_U]; digits=4);
```

```
round.([p_L, p_U]; digits = 4) = [0.2555, 0.3475]
```

以上で求めたClopper-Pearson信頼区間は R (https://cran.r-project.org/) の `binom.test` の結果に一致する.

```
> binom.test(120, 400)

    Exact binomial test

data:  120 and 400
number of successes = 120, number of trials = 400, p-value = 7.666e-16
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.2554667 0.3475219
sample estimates:
probability of success
                   0.3
```

WolframAlphaで求めるには以下のようにすればよい.

- quantile(BetaDistribution(120, 281), 0.025) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28120%2C+281%29%2C+0.025%29) → 0.255467
- quantile(BetaDistribution(121, 280), 0.975) (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28121%2C+280%29%2C+0.975%29) → 0.347522

In [40]:

```
α = 0.05
z = quantile(Normal(), 1 - α/2)

var"z_{0.025}の定義" = plot()
plot!(Normal(),  -4,  4; label="")
plot!(Normal(), -4, -z; label="", c=1, fillrange=0, fc=:red, fa=0.3)
plot!(Normal(),  z,  4; label="", c=1, fillrange=0, fc=:red, fa=0.3)
plot!([-z, -z, NaN, z, z],
    [0, pdf(Normal(), -z), NaN, 0, pdf(Normal(), z)];
    label="±z_{0.025}", c=:red)
title!("Normal(0,1)"; xtick=-4:4)
annotate!(-2.2, 0.06, text("2.5%", 10, :red, :right))
annotate!(2.2, 0.06, text("2.5%", 10, :red, :left));
```

3 関連の問題

以下では信頼度を $1-\alpha$ と表す($0<\alpha<1$ は有意水準と呼ばれる). さらに,

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2)$$

とおく. これは標準正規分布において $z_{\alpha/2}$ 以上になる確率が $\alpha/2$ になることを意味している. 例えば, $\alpha = 5\%$ のとき,

$$z_{0.025} = 1.95996\cdots \approx 1.96.$$

この 1.96 は非常に有名な数値である. 大雑把な分析で十分な場合には 2 で代替してよい.

In [41]:

```
var"z_{0.025}の定義"
```

Out[41]:

3.1 問題: Waldの信頼区間との比較

Waldの信頼区間の定義: データ「n 回中 k 回成功」が得られたとする. このとき,

$$\hat{p} = \frac{k}{n}, \quad \widehat{\mathrm{SE}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

とおき,

$$\hat{p} - z_{\alpha/2}\widehat{\mathrm{SE}} \le p \le \hat{p} + z_{\alpha/2}\widehat{\mathrm{SE}}$$

を成功確率パラメータ p の信頼度 $1-\alpha$ の **Waldの信頼区間** と呼ぶ. 95% Wald信頼区間は近似的に次のように書ける:

$$\hat{p} - 1.96\widehat{\mathrm{SE}} \le p \le \hat{p} + 1.96\widehat{\mathrm{SE}}.$$

(1) データ「$n = 100$ 回中 $k = 30$ 回成功」の 95% Wald信頼区間を求めよ.

(2) データ「$n = 400$ 回中 $k = 120$ 回成功」の 95% Wald信頼区間を求めよ.

(3) これらの場合にWald信頼区間とClopper-Pearson信頼区間は大きくは違わないことを確認せよ.

数値は小数点以下第4桁まで求めよ.

注意: Waldの信頼区間の計算はClopper-Pearsonの信頼区間と比較すると圧倒的に易しい. k と $n-k$ が十分に大きければ, 実用的には計算が容易なWaldの信頼区間で用が足りることが多い.

解答例: (1),(3) データ「$n = 100$ 回中 $k = 30$ 回成功」の 95% Wald信頼区間は

$$[0.2102, 0.3898].$$

同じデータから決まる95% Clopper-Pearson信頼区間は

$$[0.2124, 0.3998]$$

確かに大きくは違わない.

(2),(3) データ「$n = 400$ 回中 $k = 120$ 回成功」の 95% Wald信頼区間は

$$[0.2551, 0.3449].$$

同じデータから決まる95% Clopper-Pearson信頼区間は

$$[0.2555, 0.3475]$$

この2つは近似的によく一致しているように見える.

解答終

In [42]:
```
α = 0.05
z = quantile(Normal(), 1 - α/2)
```

Out[42]:
```
1.9599639845400576
```

In [43]:
```
# (1)
n, k = 100, 30
p̂ = k/n
SE = √(p̂*(1 - p̂)/n)
p_L, p_U = p̂ - z*SE, p̂ + z*SE
@show [p_L, p_U]
@show round.([p_L, p_U]; digits=4);
```

```
[p_L, p_U] = [0.21018316681457927, 0.3898168331854207]
round.([p_L, p_U]; digits = 4) = [0.2102, 0.3898]
```

In [44]:
```
# (2)
n, k = 400, 120
p̂ = k/n
SE = √(p̂*(1 - p̂)/n)
p_L, p_U = p̂ - z*SE, p̂ + z*SE
@show [p_L, p_U]
@show round.([p_L, p_U]; digits=4);
```

```
[p_L, p_U] = [0.25509158340728966, 0.3449084165927103]
round.([p_L, p_U]; digits = 4) = [0.2551, 0.3449]
```

WolframAlphaで

- quantile(NormalDistribution(0,1), 0.975) (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.975%29)

によって, $z = 1.95996$ を得ることができれば, 後は平方根も計算できる任意の電卓ですべてを計算できる.

3.2 Waldの信頼区間の基礎

二項分布 $\mathrm{Binomial}(n, p)$ の期待値は np で標準偏差は $\sqrt{np(1-p)}$ になる.

二項分布の中心極限定理: 二項分布は正規分布で近似される:

$$\mathrm{Binomial}(n, p) \approx \mathrm{Normal}\left(np, \sqrt{np(1-p)}\right).$$

分散だけが少し違う正規分布による近似: $\hat{p} = k/n$ が p に近いとき,

$$\mathrm{Normal}\left(np, \sqrt{np(1-p)}\right) \approx \mathrm{Normal}\left(np, \sqrt{n\hat{p}(1-\hat{p})}\right).$$

この2つの近似が使える状況では,

$$\mathrm{Binomial}(n, p) \approx \mathrm{Normal}\left(np, \sqrt{n\hat{p}(1-\hat{p})}\right)$$

という近似が使えるので, 確率変数 K が $\mathrm{Binomial}(n, p)$ に従うとき,

$$\frac{K - np}{\sqrt{n\hat{p}(1-\hat{p})}} \sim \mathrm{Normal}(0,1), \text{approximately}$$

という近似が使え, $\hat{p} = k/n$, $\widehat{\mathrm{SE}} = \sqrt{p(1-p)/n}$ のとき

$$\frac{k - np}{\sqrt{n\hat{p}(1-\hat{p})}} = \frac{\hat{p} - p}{\widehat{\mathrm{SE}}}$$

なので,

$$\begin{aligned}
&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上(以下)になる確率}) \\
&\approx (\text{正規分布 } \mathrm{Normal}\left(np, \sqrt{n\hat{p}(1-\hat{p})}\right) \text{ において } k = n\hat{p} \text{ 以上(以下)になる確率}) \\
&\approx (\text{標準正規分布 } \mathrm{Normal}(0,1) \text{ において } (\hat{p} - p)/\widehat{\mathrm{SE}} \text{ 以上(以下)になる確率}).
\end{aligned}$$

この確率が $\alpha/2$ に等しくなるような p を p_L (p_U)と書くと, $z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2)$ ($\alpha = 5\%$ ならば $z_{\alpha/2} \approx 1.96$)のとき,

$$\frac{\hat{p} - p_L}{\widehat{\mathrm{SE}}} \approx z_{\alpha/2}, \quad \frac{\hat{p} - p_U}{\widehat{\mathrm{SE}}} \approx -z_{\alpha/2}$$

という近似が成立するので,

$$p_L \approx \hat{p} - z_{\alpha/2}\widehat{\mathrm{SE}}, \quad p_U \approx \hat{p} + z_{\alpha/2}\widehat{\mathrm{SE}}.$$

p_L, p_U はClopper-Pearsonの信頼区間の両端の値なので, この近似がうまく行く場合には, Clopper-Pearsonの信頼区間とWaldの信頼区間は近似的に一致することになる.

この近似がうまく行っていない場合には, **Waldの信頼区間は0から1の範囲をはみ出してしまうことさえある.**

3.3 問題: Waldの信頼区間が0〜1の範囲をはみ出す場合

データ「$n = 20$ 回中 $k = 2$ 回成功」の 95% Wald信頼区間の下限が 0 未満になっていることを確認せよ.

解答例: データ「$n = 20$ 回中 $k = 2$ 回成功」の 95% Wald信頼区間は

$$[-0.0315, 0.2315]$$

になる. 確かに下限は 0 未満になっている.

解答終

注意: 対称性から, データ「$n = 20$ 回中 $k = 18$ 回成功」の 95% Wald信頼区間の上限が 1 より大きくなることもわかる. 一般に k/n が 0 または 1 に非常に近いとWaldの信頼区間の精度は低くなる.

In [45]:

```
n, k = 20, 2
p̂ = k/n
SE = √(p̂*(1 - p̂)/n)
p_L, p_U = p̂ - z*SE, p̂ + z*SE
@show [p_L, p_U]
@show round.([p_L, p_U]; digits=4);
```

```
[p_L, p_U] = [-0.031478381086487484, 0.2314783810864875]
round.([p_L, p_U]; digits = 4) = [-0.0315, 0.2315]
```

3.4 問題: Wilsonの信頼区間との比較

Wilsonの信頼区間の定義: データ「n 回中 k 回成功」が得られたとする. このとき,

$$\hat{p} = \frac{k}{n}, \quad \widehat{\mathrm{SE}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \quad z = z_{\alpha/2}, \quad a = 1 + \frac{z^2}{n}, \quad b = \hat{p} + \frac{z^2}{2n}, \quad c = \hat{p}^2$$

とおき, p に関する方程式 $ap^2 - 2bp + c = 0$ の2つの解を $p_L < p_I$ と書くき,

$$p_L \le p \le p_U$$

を成功確率パラメータ p の信頼度 $1-\alpha$ の **Wilsonの信頼区間** と呼ぶ.

$p_- = p_L, p_+ = p_U$ と書くと,

$$p_{\pm} = \frac{b \pm \sqrt{b^2 - ac}}{a} = \frac{1}{1 + z^2/n}\left(\hat{p} + \frac{z^2}{2n} \pm z\sqrt{\widehat{\mathrm{SE}}^2 + \frac{z^2}{4n^2}}\right).$$

以下の場合に, 信頼度が 95% の場合の, Clopper-Pearson信頼区間, Wald信頼区間, Wilson信頼区間を求めよ.

(1) データが「$n = 20$ 回中 $k = 2$ 回成功」の場合.

(2) データが「$n = 100$ 回中 $k = 5$ 回成功」の場合.

(3) データが「$n = 400$ 回中 $k = 20$ 回成功」の場合.

解答例: (1) データ「$n = 20$ 回中 $k = 2$ 回成功」ときの 95% 信頼区間はそれぞれ

Clopper-Pearson	$[0.0123, 0.3170]$
Wald	$[-0.0315, 0.2315]$
Wilson	$[0.0279, 0.3010]$

(2) データ「$n = 100$ 回中 $k = 5$ 回成功」ときの 95% 信頼区間はそれぞれ

Clopper-Pearson	$[0.0164, 0.1128]$
Wald	$[0.0073, 0.0927]$
Wilson	$[0.0215, 0.1118]$

(3) データ「$n = 400$ 回中 $k = 20$ 回成功」ときの 95% 信頼区間はそれぞれ

Clopper-Pearson	$[0.0308, 0.0762]$
Wald	$[0.0286, 0.0714]$
Wilson	$[0.0326, 0.0760]$

In [46]:

```julia
function clopper_pearson_ci(n, k; α = 0.05)
    p_L = quantile(Beta(k, n-k+1), α/2)
    p_U = quantile(Beta(k+1, n-k), 1 - α/2)
    [p_L, p_U]
end

function zpse(n, k; α = 0.05)
    z = quantile(Normal(), 1 - α/2)
    p̂ = k/n
    SE = √(p̂ * (1 - p̂) / n)
    z, p̂, SE
end

function wald_ci(n, k; α = 0.05)
    z, p̂, SE = zpse(n, k; α)
    p_L = p̂ - z*SE
    p_U = p̂ + z*SE
    [p_L, p_U]
end

function zpseabc(n, k; α = 0.05)
    z, p̂, SE = zpse(n, k; α)
    a = 1 + z^2/n
    b = p̂ + z^2/(2n)
    c = p̂^2
    z, p̂, SE, a, b, c
end

function wilson_ci_old(n, k; α = 0.05)
    z, p̂, SE, a, b, c = zpseabc(n, k; α)
    D = b^2 - a*c
    # Solutions of a p² - 2b p + c = 0
    p_L = (b - √D)/a
    p_U = (b + √D)/a
    [p_L, p_U]
end

function wilson_ci(n, k; α = 0.05)
    z, p̂, SE, a, b, c = zpseabc(n, k; α)
    D₀ = SE^2 + z^2/(4n^2)
    # Solutions of a p² - 2b p + c = 0
    p_L = (b - z*√D₀)/a
    p_U = (b + z*√D₀)/a
    [p_L, p_U]
end

wilson_ci_old(100, 30) ≈ wilson_ci(100, 30)
```

Out[46]: true

In [47]:

```julia
function print_cis(n, k; α = 0.05)
    ci_cp = round.(clopper_pearson_ci(n, k; α); digits=4)
    ci_wald = round.(wald_ci(n, k; α); digits=4)
    ci_wilson = round.(wilson_ci(n, k; α); digits=4)
    println("Clopper-Pearson: ", ci_cp)
    println("Wald:            ", ci_wald)
    println("Wilson:          ", ci_wilson)
end
```

Out[47]: print_cis (generic function with 1 method)

In [48]:

```julia
# (1)
print_cis(20, 2)
```

```
Clopper-Pearson: [0.0123, 0.317]
Wald:            [-0.0315, 0.2315]
Wilson:          [0.0279, 0.301]
```

In [49]:

```julia
# (2)
print_cis(100, 5)
```

```
Clopper-Pearson: [0.0164, 0.1128]
Wald:            [0.0073, 0.0927]
Wilson:          [0.0215, 0.1118]
```

In [50]:
```python
# (3)
print_cis(400, 20)
```

```
Clopper-Pearson: [0.0308, 0.0762]
Wald:            [0.0286, 0.0714]
Wilson:          [0.0326, 0.076]
```

3.5 Wilsonの信頼区間の基礎

Waldの信頼区間では「分散だけが少し違う正規分布による近似」を使っていた. しかし, その近似を避けて, 二項分布の中心極限定理による近似

$$\mathrm{Binomial}(n,p) \approx \mathrm{Normal}\left(np, \sqrt{np(1-p)}\right)$$

だけで, Clopper-Pearsonの信頼区間の近似を構成できる. それがWilsonの信頼区間である.

二項分布の中心極限定理による近似だけを用いると, 確率変数 K が $\mathrm{Binomial}(n,p)$ に従うとき,

$$\frac{K-np}{\sqrt{np(1-p)}} \sim \mathrm{Normal}(0,1), \mathrm{approximately}.$$

Waldの信頼区間の構成ではこの式の平方根の内側の p が $\hat{p}=k/n$ になっていた. このとき,

$$\frac{k-np}{\sqrt{np(1-p)}} = \frac{\hat{p}-p}{\sqrt{p(1-p)/n}}$$

なので,

$$\begin{aligned}
&(\text{二項分布 } \mathrm{Binomial}(n,p) \text{ で成功回数が } k \text{ 以上(以下)になる確率})\\
&\approx (\text{正規分布 } \mathrm{Normal}\left(np, \sqrt{np(1-p)}\right) \text{ において } k=n\hat{p} \text{ 以上(以下)になる確率})\\
&\approx (\text{標準正規分布 } \mathrm{Normal}(0,1) \text{ において } (\hat{p}-p)\Big/\sqrt{p(1-p)/n} \text{ 以上(以下)になる確率}).
\end{aligned}$$

この確率が $\alpha/2$ に等しくなるような p を p_L (p_U)と書くと, $z=z_{\alpha/2}=\mathrm{quantile}(\mathrm{Normal}(0,1),1-\alpha/2)$ ($\alpha=5\%$ ならば $z=z_{\alpha/2}\approx 1.96$)のとき,

$$\frac{\hat{p}-p_L}{\sqrt{p(1-p)/n}} \approx z \quad \left(\frac{\hat{p}-p_U}{\sqrt{p(1-p)/n}} \approx -z\right)$$

という近似が成立する. これはClopper-Pearsonの信頼区間の両端の値 p_L, p_U が p に関する次の方程式の近似解になっていることを意味している:

$$\frac{(\hat{p}-p)^2}{p(1-p)/n} = z^2.$$

この方程式は次の2次方程式に書き直される:

$$\left(1+\frac{z^2}{n}\right)p^2 - 2\left(\hat{p}+\frac{z^2}{2n}\right)p + \hat{p}^2 = 0.$$

Wilsonの信頼区間の定義はこの2次方程式の2つの解で挟まれた区間になっている. 実際,

$$a = 1+\frac{z^2}{n}, \quad b = \hat{p}+\frac{z^2}{2n}, \quad c=\hat{p}^2, \quad f(p)=ap^2-2bp+c$$

とおくと, $0\le\hat{p}\le 1$ であることより,

$$f(0)=\hat{p}^2\ge 0, \quad f(1)=(1-\hat{p})^2\ge 0, \quad 0\le\frac{b}{a}=\frac{\hat{p}^2+z^2/(2n)}{1+z^2/n}\le 1$$

でかつ

$$\begin{aligned}
b^2-ac &= \hat{p}^2+\frac{z^2}{n}\hat{p}+\frac{z^4}{4n^2}-\hat{p}^2-\frac{z^2}{n}\hat{p}\\
&= \frac{z^2}{n}\hat{p}(1-\hat{p})+\frac{z^4}{4n^2} = z^2\left(\widehat{\mathrm{SE}}^2+\frac{z^2}{4n^2}\right)\ge 0
\end{aligned}$$

なので, $f(p)=ap^2-2bp+c=0$ の2つの解はどちらも 0 以上 1 の範囲に含まれ,

$$p_\pm = \frac{b \pm \sqrt{b^2 - ac}}{a} = \frac{1}{1 + z^2/n}\left(\hat{p} + \frac{z^2}{2n} \pm z\sqrt{\widehat{\mathrm{SE}}^2 + \frac{z^2}{4n^2}}\right)$$

と表される.

3.6 Sterneの信頼区間について

以上ではClopper-Pearsonの信頼区間, Waldの信頼区間, Wilsonの信頼区間の二項分布モデルにおける信頼区間について説明した. これ以外にも多数の信頼区間の定義がある. それら3つ以外の信頼区間として基本的なのは **Sterneの信頼区間** である.

Sterneの信頼区間を与えるP値函数に関する解説が「[標本分布について (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)」のノートの「Sterneの信頼区間とそれを与えるP値函数」の節にある.

3.7 問題: Poisson分布とガンマ分布の関係

$\lambda > 0$ のとき, 公式

$$\sum_{i \ge k} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_0^p t^{k-1}(1-t)^{n-i}\,dt}{B(k, n-k+1)}$$

において, $p = \lambda/n$ とおき, $t = x/n$ とおき, $n \to \infty$ の極限を取ることによって次の公式が得られることを示せ:

$$e^{-\lambda} \sum_{i \ge k} \frac{\lambda^i}{i!} = \frac{\int_0^\lambda e^{-x} x^{k-1}\,dx}{\Gamma(k)}.$$

これは次が成立することを意味している:

$$\begin{aligned}&(\text{Poisson分布 } \mathrm{Poisson}(\lambda) \text{ において } k \text{ 以上になる確率})\\ &= (\text{ガンマ分布 } \mathrm{Gamma}(k,1) \text{ において } \lambda \text{ 以下になる確率}).\end{aligned}$$

注意: この結果は

- 単位時間のあいだに平均して 1 回起こるイベントが時間 λ のあいだに起こる回数が k 以上になる確率

と次が等しいことを意味している:

- 単位時間のあいだに平均して 1 回起こるイベントが k 回起こるまでにかかる時間が λ 以下の確率.

注意: 意味的には, ガンマ分布に収束するのは負の二項分布である. ベータ分布は負の二項分布とも関係がある. それを使えば負の二項分布を使っても同様の結果を得ることができる.

解答例: $p = \lambda/n$ とおき, $n \to \infty$ とすると,

$$\begin{aligned}\binom{n}{i} p^i (1-p)^{n-i} &= \frac{n(n-1)\cdots(n-i+1)}{i!} \frac{\lambda^i}{n^i}\left(1 - \frac{\lambda}{n}\right)^{n-i} \\ &= \underbrace{\frac{n(n-1)\cdots(n-i+1)}{n^i}}_{\to 1} \underbrace{\left(1 - \frac{\lambda}{n}\right)^{n-i}}_{\to \exp(-\lambda)} \frac{\lambda^i}{i!} \to e^{-\lambda}\frac{\lambda^i}{i!}\end{aligned}$$

なので,

$$\sum_{i \ge k} \binom{n}{i} p^i (1-p)^{n-i} \to e^{-\lambda} \sum_{i \ge k} \frac{\lambda^i}{i!}.$$

$t = x/n$ とおき, $n \to \infty$ とすると,

$$\begin{aligned}&n^k B(k, n-k+1) \to \Gamma(k), \\ &n^k \int_0^p t^{k-1}(1-t)^{n-i}\,dt = \int_0^\lambda x^{k-1}\left(1 - \frac{x}{n}\right)^{n-i} dt \to \int_0^\lambda e^{-x} x^{k-1}\,dx\end{aligned}$$

となることより,

$$\frac{\int_0^p t^{k-1}(1-t)^{n-i}\,dt}{B(k, n-k+1)} = \frac{n^k \int_0^p t^{k-1}(1-t)^{n-i}\,dt}{n^k B(k, n-k+1)} \to \frac{\int_0^\lambda e^{-x} x^{k-1}\,dx}{\Gamma(k)}.$$

解答終

4 おまけ: 二項分布モデルのBayes統計との関係

このノート群ではBayes統計に深く触れるつもりはない.

しかし, 興味がある人は多いと思われるので, 二項分布モデルの場合のBayes統計について簡単に説明しておくことにする.

4.1 二項分布モデルのBayes統計

二項分布モデル $\mathrm{Binomial}(n,p)$ とパラメータ p に関する確率密度函数 $\varphi(p)$ $(0 \leq p \leq 1)$の組をモデルとして採用していると仮定する. すなわち, このモデルは離散変数 k と連続変数 p に関する確率分布を記述する次の函数で与えられていると考えることができる:

$$P(k,p|n) = \binom{n}{k} p^k (1-p)^{n-k} \varphi(p) \quad (k = 0,1,2,\dots,\ 0 \leq p \leq 1).$$

この $P(k,p|n)$ の値は 0 以上であり,

$$\sum_{k=0}^{n} P(k,p|n) = \varphi(p), \quad \int_0^1 \left(\sum_{k=0}^{n} P(k,p|n) \right) dp = \int_0^1 \varphi(p)\, dp = 1$$

を満たしている.

「n 回中 k 回成功」というデータが得られたとき, データと同じ数値がモデル内で生成されたという条件が定める条件付き確率分布を考えることが **Bayes法** の基本である(Bayesの定理やBayesルールのような用語を持ち出す必要はない).
この場合には p に関する条件付き確率分布を考える. その確率密度函数を $\varphi(p|n,k)$ と書くと,

$$\begin{aligned}
P(k|n) &= \int_0^1 P(k,p|n)\, dp = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \varphi(p)\, dp, \\
\varphi(p|n,k) &= \frac{P(k,p|n)}{P(k|n)} = \frac{\binom{n}{k} p^k (1-p)^{n-k} \varphi(p)}{\int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \varphi(p)\, dp} = \frac{p^k (1-p)^{n-k} \varphi(p)}{\int_0^1 p^k (1-p)^{n-k} \varphi(p)\, dp}.
\end{aligned}$$

$\varphi(p)$ は **事前分布** (prioir)と呼ばれ, $\varphi(p|n,k)$ はデータ「n 回中 k 回成功」に関する **事後分布** (posterior)と呼ばれる.

4.2 事前分布が共役事前分布(ベータ分布)の場合

この場合には, 事前分布としてベータ分布 $\mathrm{Beta}(a,b)$ を採用すると計算が著しく易しくなる. 以下では事前分布は $\mathrm{Beta}(a,b)$ であると仮定する:

$$\varphi(p) = \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)}.$$

このとき,

$$\begin{aligned}
p^k (1-p)^{n-k} \varphi(p) &= p^{a+k-1} (1-p)^{b+n-k-1}, \\
\int_0^1 p^k (1-p)^{n-k} \varphi(p)\, dp &= B(a+k, b+n-k)
\end{aligned}$$

なのでデータ「n 回中 k 回成功」に関する事後分布は次の形になる:

$$\varphi(p|n,k) = \frac{p^{a+k-1}(1-p)^{b+n-k-1}}{B(a+k, b+n-k)}.$$

つまり, 事後分布は $\mathrm{Beta}(a+k, b+n-k)$ になる.

ベータ分布 $\mathrm{Beta}(\alpha,\beta)$ は α,β が大きなとき, $p = \alpha/(\alpha+\beta)$ の周囲に集中した分布になる.

ゆえに, 事後分布は $\mathrm{Beta}(a+k, b+n-k)$ は $a+k$ と $b+n-k$ が大きいならば $p = (a+k)/(a+b+n)$ の周囲に集中した分布になり, $k, n-k$ が a,b よりも十分に大きいならば $p = k/n$ に集中した分布になり, 最尤法と同じ結果が得られる.

In [51]:

```julia
function plot_beta(a, b; label="", title="Beta($a, $b)", kwargs...)
    if (a == 0 && b > 0) || (a > 0 && b == 0)
        plot(x -> x^(a-1)*(1-x)^(b-1), 0, 1; ylim=(-0.5, 10.0), label, kwargs...)
    else
        beta = Beta(a, b)
        if a == 1 && b == 1
            plot(beta, 0, 1; ylim=(-0.07, 1.4), label, kwargs...)
        elseif a ≥ 1 && b ≥ 1
            plot(beta, 0, 1; label, kwargs...)
        else
            plot(beta, 0, 1; ylim=(-0.5, 10.0), label, kwargs...)
        end
    end
    plot!(; xtick=0:0.1:1)
    title!(title)
end

function plot_betas(a, b, n, k)
    P0 = plot_beta(a, b)
    P1 = plot_beta(a+k, b+n-k)
    P2 = plot_beta(a+100k, b+100(n-k))
    plot(P0, P1, P2; size=(800, 180), layout=(1, 3))
    plot!(; tickfontsize=6)
end
```

Out[51]: plot_betas (generic function with 1 method)

In [52]:

```julia
# 一様事前分布の場合
plot_betas(1, 1, 10, 3)
```

Out[52]:

In [53]:

```julia
# 非対称なimproper事前分布の場合
plot_betas(0, 1, 10, 3)
```

Out[53]:

In [54]:

```julia
# 偏った事前分布の場合
plot_betas(8, 2, 10, 3)
```

Out[54]:

4.3 二項分布モデルでの片側P値のBayes統計での解釈

事前分布無しの試行回数 n の二項分布モデルにおける仮説「成功確率 p は p_0 以下である」のデータ「n 回中 k 回成功」に関する片側検定のP値を二項分布 $\mathrm{Bionmial}(n, p_0)$ において成功回数が k 以上になる確率と定義し, $\mathrm{pvalue}(k|n, p \leq p_0)$ と書くことにする. それは, ベータ分布 $\mathrm{Beta}(k, n-k+1)$ において p_0 以下になる確率に等しいのであった:

$$\mathrm{pvalue}(k|n, p \leq p_0) = \sum_{i \geq k} \binom{n}{i} p_0^i (1-p_0)^{n-i} = \frac{\int_0^{p_0} p^{k-1}(1-p)^{n-k}}{B(k, n-k+1)}.$$

これは形式的には, 成功確率 p に関する事前分布が $\mathrm{Beta}(0,1)$ のときの, 事後分布 $\mathrm{Beta}(k, n-k+1)$ において仮説「成功確率 p は p_0 以下である」が成立する確率に一致する. すなわち, この場合には

- 仮説「成功確率 p は p_0 以下である」のP値

と

- Bayes統計における事後分布で仮説「成功確率 p は p_0 以下である」が成立する確率

がぴったり一致していることになる.

ただし, ベータ分布 $\mathrm{Beta}(\alpha, \beta)$ は α も β も正の場合にのみ定義されているので, 事前分布として $\mathrm{Beta}(0,1)$ を採用することは形式的である. このような場合に事前分布 $\mathrm{Beta}(0,1)$ はimproperであるという.

しかし, 事前分布が $\mathrm{Beta}(a,b)$ のとき, $k, n-k$ が a, b よりも十分に大きければ, 事後分布 $\mathrm{Beta}(a+k, b+n-k)$ はほとんど a, b に依存しなくなる. ゆえに, improper事前分布 $\mathrm{Beta}(0,1)$ を例えば一様事前分布 $\mathrm{Beta}(1,1)$ やJeffreys事前分布 $\mathrm{Beta}(1/2, 1/2)$ に取り換えても, 上の場合のP値とBayes統計での事後分布における確率の一致は近似的に成立することになる.

この意味で, $k, n-k$ が十分大きな場合にはBayes統計の結果と通常のP値を使った統計分析の結果は実践的な違いを生まない.

注意: だから, 二項分布モデル(やBernoulli試行モデル)の場合を例に使って, P値よりもBayes統計を使うべきだと安易に主張している文献を読むときにはその主張を疑いながら読む必要がある.

注意: 上のようなデータサイズが大きい場合の(近似的)一致は, 正則モデルと呼ばれる「単純な」モデルで広く成立している.

注意: 事前分布が大きく偏っている場合には, その分だけデータサイズをさらに大きくしないと, 上のような近似的な一致は得られない. しかし, 偏りの小さい「おとなしめ」の事前分布が採用されている場合には, それほど大きなデータサイズでなくても, 近似的な一致が成立している可能性が高い.

4.4 Pólyaの壺との関係

事前分布 $\varphi(p)$ と二項分布 $\mathrm{Binomial}(n,p)$ の組で構成されたBayes統計のモデル内における成功回数 k の分布の確率質量函数は

$$P(k|n) = \int_0^1 P(k,p|n)\,dp = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \varphi(p)\,dp$$

になるのであった. 事前分布が $\mathrm{Beta}(a,b)$ のとき, これは以下のようになる:

$$P(k|n) = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)}\,dp = \binom{n}{k} \frac{B(a+k, b+n-k)}{B(a,b)}.$$

さらにこれは, $\Gamma(\alpha)\Gamma(\beta) = \Gamma(\alpha+\beta)B(\alpha,\beta)$ や $\Gamma(x+k) = x(x+1)\cdots(x+k-1)\Gamma(x)$ $(k = 0, 1, 2, \ldots)$を使うと,

$$P(k|n) = \binom{n}{k} \frac{a(a+1)\cdots(a+k-1)\cdot b(b+1)\cdots(b+n-k-1)}{(a+b)(a+b+1)\cdots(a+b+n-1)}$$

と書き直される. これは, 最初に壺の中に赤い玉が a 個と白い玉が b 個入っているときに, そこから無作為に玉を取り出して, その色を記録し, 取り出された玉と同じ色の玉を壺に2個返すことを n 回繰り返したときに, 赤い玉が全部で k 回取り出される確率に一致する($n = 5$ の場合に色々な例を考えてみよ, 仕組みはすぐにわかる). ただし, この解釈は a, b がともに整数の場合にのみ成立するが, 上の式は a, b が整数でなくても確率質量函数を与える. この解釈を **Pólyaの壺** (Pólya's urn) による解釈と呼ぶ.

k の分布の

$$P(k|n) = \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \varphi(p)\,dp$$

という表示は, k が次のようにランダムに決まっているという解釈を与える:

(B) 最初に事前分布 $\varphi(p)$ によって一定の成功確率 p をランダムに決定し, その後はその成功確率 p のBernoulli試行を n 回行って, 成功回数を k とする.

この解釈は「最初に "才能" p がランダムに決まっている」と要約することができるだろう.

それに対して, Pólyaの壺の解釈は k が次のようにランダムに決まっているという解釈を与える:

(P) 最初に壺の中には赤い玉が a 個と白い玉 b 個が入っている. その後, 赤い玉が壺から取り出されれば壺の中の赤い玉の個数は1個増え, 白い玉についても同様である. そのような試行を n 回繰り返したときに赤い玉が出た回数 k とする.

赤い玉を「成功」とみなすと, この解釈は「最初の才能 a, b は誰もが一定だが, 成功すればするほど成功し易くなり, 失敗すればするほど失敗し易くなる」と要約することができるだろう. Pólyaの壺の試行は「富める者はさらに富み、貧しい者はさらに貧しくなる」と要約されることがある.

以上の(B)と(P)では話が全然違っている. (B)と(P)が与える確率分布が等しいという数学的結果は, 生まれ付きの才能で k がランダムに決まっているのか, それとも「富める者はさらに富み、貧しい者はさらに貧しくなる」という仕組みが働いて最終的に差がついたのかが, 成功回数 k を観察しただけではわからないことを意味している.

注意: 以上の話題については, 「[大数の法則と中心極限定理 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)」のノートの「大数の法則が成立しない場合: Pólyaの壺」の節での「pがベータ分布に従ってランダムに決まっているベルヌーイ試行」と「富むものがさらに富み, 貧しいものがさらに貧しくなるポリアの壺試行」に関するグラフのプロットも参照せよ. Pólyaの壺については「[確率分布達の解釈 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/03%20Interpretation%20of%20probability%20distributions.ipynb#%](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/03%20Interpretation%20of%20probability%20distributions.ipynb#%)」のノートの「ベータ負の二項分布の解釈 (Pólyaの壺)」の節も参照せよ.

In [ ]:

```
1
```

Julia言語を使った統計学の勉強の仕方

- 黒木玄

- 2022-05-28〜2022-05-31

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思

目次

- 1 Julia言語の使い方
  - 1.1 Juliaのインストールと実行
  - 1.2 グラフの描き方
  - 1.3 標準正規分布乱数のプロット
  - 1.4 確率分布の扱い方
  - 1.5 正規分布の確率密度函数のプロット
- 2 Anscombeの例のプロット
  - 2.1 RDatasets.jlパッケージのインストール
  - 2.2 データのプロットの仕方
- 3 Datasaurusの散布図のプロット
  - 3.1 データの取得
  - 3.2 散布図の作成
- 4 中心極限定理のプロット
  - 4.1 素朴なワークフロー
  - 4.2 Revise.jlを使うワークフロー
  - 4.3 問題: Revise.jlを実際に使ってみよ

In [1]:
```
using StatsPlots
default(fmt = :png)
```

1 Julia言語の使い方

1.1 Juliaのインストールと実行

(1) JuliaLangについて検索 (https://www.google.com/search?q=JuliaLang)する.

(2) Julia言語のウェブサイト (https://julialang.org/)に移動する.

(3) Downloadボタンを押して, Download Julia (https://julialang.org/downloads/) のページに移動する.

(4) 自分のパソコンに合わせて, current stable release のインストーラをダウンロード

(5) それを実行してJulia言語の公式バイナリを自分のパソコンにインストールする.

(6) juliaを実行すると以下のように表示される.

```
               _       _ _(_)_     |  Documentation: https://docs.julialang.org
   (_)     | (_) (_)    |
    _ _   _| |_  __ _   |  Type "?" for help, "]?" for Pkg help.
   | | | | | | |/ _` |  |
   | | |_| | | | (_| |  |  Version 1.7.3 (2022-05-06)
  _/ |\__'_|_|_|\__'_|  |  Official https://julialang.org/ release
 |__/                   |

julia>
```

- `julia>` プロンプトで `?` を入力するとヘルプモードに移る.
- `julia>` プロンプトで `]` を入力するとパッケージ管理モードに移る.
- それらのモードから `julia>` プロンプトに戻るにはバックスペースキーを押す.
- 至るところでタブキーによる補完を使える.
- `julia>` プロンプトでJulia言語のコードを入力して終わりに `;` を付けて実行すると計算結果が表示されない.

問題: 以下を自分がインストールしたjuliaで実行してみよ.

In [2]:
```
1 + 1
```
Out[2]: 2

In [3]:
```
sin(pi/6)
```
Out[3]: 0.49999999999999994

In [4]:
```
sinpi(1/6)
```
Out[4]: 0.5

1.2 グラフの描き方

(7) 統計学対応のグラフ作画パッケージを入れるためには

```
julia> using StatsPlots
```

を実行するだけでよい. インストールするかどうかをjuliaが聞いて来た場合には `y` と答える.

問題: 自分のパソコンで `using StatsPlots` の後に以下を実行してグラフを作画してみよ.

In [5]:
```
using StatsPlots
plot(sin)
```
Out[5]:

1.3 標準正規分布乱数のプロット

In [6]:
```
Z = randn(10^4);
```

In [7]:
```
histogram(Z; norm=true, alpha=0.5, label="")
```

Out[7]:

In [8]:
```
plot!(x -> exp(-x^2/2)/sqrt(2pi), -4, 4; label="", lw=3)
```

Out[8]:

1.4 確率分布の扱い方

(7) 確率分布を扱うためのパッケージを入れるためには

```
julia> using Distributions
```

を実行するだけでよい. インストールするかどうかをjuliaが聞いて来た場合には `y` と答える.

しかし, この機会にパッケージモードでのパッケージの入れ方についてもこの機会に学んでおいた方がよい.

```
julia> ]
```

と `]` を入力すると,

```
pkg>
```

と表示されてパッケージ管理モードに移行できる(戻るには `pkg>` プロンプトの直後にバックスペースを押せばよい).

以下を実行すると, `Distributions.jl` パッケージをjuliaの環境にインストールできる.

```
pkg> add Distributions
```

1.5 正規分布の確率密度函数のプロット

In [9]:
```julia
using Distributions
X = rand(Normal(2, 3), 10^4);
```

In [10]:
```julia
histogram(X; norm=true, alpha=0.5, label="sample of Normal(2, 3)")
```

Out[10]:

In [11]:
```julia
plot!(Normal(2, 3); label="Normal(2, 3)", lw=2)
```

Out[11]:

2 Anscombeの例のプロット

2.1 RDatasets.jlパッケージのインストール

確率分布を扱うためのパッケージを入れるためには

```
julia> using RDatasets
```

を実行するだけでよい. インストールするかどうかをjuliaが聞いて来た場合には y と答える. もしくはパッケージ管理モードで

```
pkg> add RDatasets
```

を実行してもよい. その場合には julia> プロンプトで改めて using RDatasets を実行する必要がある.

In [12]:
```
using RDatasets
anscombe = dataset("datasets", "anscombe")
```

Out[12]: 11 rows × 8 columns

	X1	X2	X3	X4	Y1	Y2	Y3	Y4
	Int64	Int64	Int64	Int64	Float64	Float64	Float64	Float64
1	10	10	10	8	8.04	9.14	7.46	6.58
2	8	8	8	8	6.95	8.14	6.77	5.76
3	13	13	13	8	7.58	8.74	12.74	7.71
4	9	9	9	8	8.81	8.77	7.11	8.84
5	11	11	11	8	8.33	9.26	7.81	8.47
6	14	14	14	8	9.96	8.1	8.84	7.04
7	6	6	6	8	7.24	6.13	6.08	5.25
8	4	4	4	19	4.26	3.1	5.39	12.5
9	12	12	12	8	10.84	9.13	8.15	5.56
10	7	7	7	8	4.82	7.26	6.42	7.91
11	5	5	5	8	5.68	4.74	5.73	6.89

2.2 データのプロットの仕方

以下ではデータ1の場合のプロットの仕方を説明しよう.

In [13]:
```
# x, y にデータを入れる
x, y = anscombe.X1, anscombe.Y1
```

Out[13]:
```
([10, 8, 13, 9, 11, 14, 6, 4, 12, 7, 5], [8.04, 6.95, 7.58, 8.81, 8.33, 9.96, 7.24, 4.26, 10.84, 4.82, 5.68])
```

In [14]:
```
# 散布図を描いてみる
using StatsPlots
scatter(x, y)
```

Out[14]:

In [15]:
```
# xlim, ylimなどを追加
scatter(x, y; label="", xlim=(3, 20), ylim=(2, 14))
```

Out[15]:

In [16]:
```
# データの標本平均や不偏分散・不偏共分散を計算
xbar = mean(x)
```

Out[16]: 9.0

In [17]:
```
ybar = mean(y)
```

Out[17]: 7.500909090909093

In [18]:
```
sx2 = var(x)
```

Out[18]: 11.0

In [19]:
```
sy2 = var(y)
```

Out[19]: 4.127269090909091

In [20]:
```
sxy = cov(x, y)
```

Out[20]: 5.501

In [21]:
```
betahat = sxy/sx2
```

Out[21]: 0.5000909090909091

In [22]:
```
alphahat = ybar - betahat*xbar
```

Out[22]: 3.0000909090909103

In [23]:

```
scatter(x, y; label="", xlim=(3, 20), ylim=(2, 14))
plot!(x -> alphahat + betahat*x, 3, 20; label="", lw=2)
```

Out[23]:

In [24]:

```
scatter(x, y; label="", xlim=(3, 20), ylim=(2, 14), title="Anscombe 1")
plot!(x -> alphahat + betahat*x, 3, 20; label="", lw=2, ls=:dash)
```

Out[24]:

In [25]:
```
# design matrix
X = x .^ (0:1)'
```

Out[25]:
```
11×2 Matrix{Int64}:
 1  10
 1   8
 1  13
 1   9
 1  11
 1  14
 1   6
 1   4
 1  12
 1   7
 1   5
```

In [26]:
```
# 最小二乗法を一発実現（計画行列の一般逆行列をyにかける）
alphahat2, betahat2 = X \ y
```

Out[26]:
```
2-element Vector{Float64}:
 3.000090909090908
 0.5000909090909091
```

In [27]:
```
# 2つの直線はぴったり重なり合う.
scatter(x, y; label="", xlim=(3, 20), ylim=(2, 14), title="Anscombe 1")
plot!(x -> alphahat + betahat*x, 3, 20; label="", lw=2)
plot!(x -> alphahat2 + betahat2*x, 3, 20; label="", lw=2, ls=:dash)
```

Out[27]:

問題: 他のアンスコムのデータについて同様のグラフを作成せよ.

3 Datasaurusの散布図のプロット

以下のデータは「条件付き確率分布, 尤度, 推定, 記述統計 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estin からのコピー＆ペースト.

3.1 データの取得

- http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html (http://www.thefunctionalart.com/2016/08/download-datasaurus-never-trust-summary.html)
  - https://www.dropbox.com/sh/xaxpz3pm5r5awes/AADUbGVagF9i4RmM9JkPtviEa?dl=0 (https://www.dropbox.com/sh/xaxpz3pm5r5awes/AADUbGVagF9i4RmM9JkPtviEa?dl=0)
- https://visualizing.jp/the-datasaurus-dozen/ (https://visualizing.jp/the-datasaurus-dozen/)

In [28]:

```python
datasaurus = [
    55.3846 97.1795
    51.5385 96.0256
    46.1538 94.4872
    42.8205 91.4103
    40.7692 88.3333
    38.7179 84.8718
    35.6410 79.8718
    33.0769 77.5641
    28.9744 74.4872
    26.1538 71.4103
    23.0769 66.4103
    22.3077 61.7949
    22.3077 57.1795
    23.3333 52.9487
    25.8974 51.0256
    29.4872 51.0256
    32.8205 51.0256
    35.3846 51.4103
    40.2564 51.4103
    44.1026 52.9487
    46.6667 54.1026
    50.0000 55.2564
    53.0769 55.6410
    56.6667 56.0256
    59.2308 57.9487
    61.2821 62.1795
    61.5385 66.4103
    61.7949 69.1026
    57.4359 55.2564
    54.8718 49.8718
    52.5641 46.0256
    48.2051 38.3333
    49.4872 42.1795
    51.0256 44.1026
    45.3846 36.4103
    42.8205 32.5641
    38.7179 31.4103
    35.1282 30.2564
    32.5641 32.1795
    30.0000 36.7949
    33.5897 41.4103
    36.6667 45.6410
    38.2051 49.1026
    29.7436 36.0256
    29.7436 32.1795
    30.0000 29.1026
    32.0513 26.7949
    35.8974 25.2564
    41.0256 25.2564
    44.1026 25.6410
    47.1795 28.7180
    49.4872 31.4103
    51.5385 34.8718
    53.5897 37.5641
    55.1282 40.6410
    56.6667 42.1795
    59.2308 44.4872
    62.3077 46.0256
    64.8718 46.7949
    67.9487 47.9487
    70.5128 53.7180
    71.5385 60.6410
    71.5385 64.4872
    69.4872 69.4872
    46.9231 79.8718
    48.2051 84.1026
    50.0000 85.2564
    53.0769 85.2564
    55.3846 86.0256
    56.6667 86.0256
    56.1538 82.9487
    53.8462 80.6410
    51.2821 78.7180
    50.0000 78.7180
    47.9487 77.5641
    29.7436 59.8718
```

```
    29.7436 62.1795
    31.2821 62.5641
    57.9487 99.4872
    61.7949 99.1026
    64.8718 97.5641
    68.4615 94.1026
    70.7692 91.0256
    72.0513 86.4103
    73.8462 83.3333
    75.1282 79.1026
    76.6667 75.2564
    77.6923 71.4103
    79.7436 66.7949
    81.7949 60.2564
    83.3333 55.2564
    85.1282 51.4103
    86.4103 47.5641
    87.9487 46.0256
    89.4872 42.5641
    93.3333 39.8718
    95.3846 36.7949
    98.2051 33.7180
    56.6667 40.6410
    59.2308 38.3333
    60.7692 33.7180
    63.0769 29.1026
    64.1026 25.2564
    64.3590 24.1026
    74.3590 22.9487
    71.2821 22.9487
    67.9487 22.1795
    65.8974 20.2564
    63.0769 19.1026
    61.2821 19.1026
    58.7179 18.3333
    55.1282 18.3333
    52.3077 18.3333
    49.7436 17.5641
    47.4359 16.0256
    44.8718 13.7180
    48.7179 14.8718
    51.2821 14.8718
    54.1026 14.8718
    56.1538 14.1026
    52.0513 12.5641
    48.7179 11.0256
    47.1795  9.8718
    46.1538  6.0256
    50.5128  9.4872
    53.8462 10.2564
    57.4359 10.2564
    60.0000 10.6410
    64.1026 10.6410
    66.9231 10.6410
    71.2821 10.6410
    74.3590 10.6410
    78.2051 10.6410
    67.9487  8.7180
    68.4615  5.2564
    68.2051  2.9487
    37.6923 25.7692
    39.4872 25.3846
    91.2821 41.5385
    50.0000 95.7692
    47.9487 95.0000
    44.1026 92.6923
];
```

3.2 散布図の作成

In [29]:
```
# 行列Aの第j列はA[:,j]
@show datasaurus[:,1];
```

```
datasaurus[:, 1] = [55.3846, 51.5385, 46.1538, 42.8205, 40.7692, 38.7179, 35.641, 33.0769, 28.97
44, 26.1538, 23.0769, 22.3077, 22.3077, 23.3333, 25.8974, 29.4872, 32.8205, 35.3846, 40.2564, 4
4.1026, 46.6667, 50.0, 53.0769, 56.6667, 59.2308, 61.2821, 61.5385, 61.7949, 57.4359, 54.8718, 5
2.5641, 48.2051, 49.4872, 51.0256, 45.3846, 42.8205, 38.7179, 35.1282, 32.5641, 30.0, 33.5897, 3
6.6667, 38.2051, 29.7436, 29.7436, 30.0, 32.0513, 35.8974, 41.0256, 44.1026, 47.1795, 49.4872, 5
1.5385, 53.5897, 55.1282, 56.6667, 59.2308, 62.3077, 64.8718, 67.9487, 70.5128, 71.5385, 71.538
5, 69.4872, 46.9231, 48.2051, 50.0, 53.0769, 55.3846, 56.6667, 56.1538, 53.8462, 51.2821, 50.0,
47.9487, 29.7436, 29.7436, 31.2821, 57.9487, 61.7949, 64.8718, 68.4615, 70.7692, 72.0513, 73.846
2, 75.1282, 76.6667, 77.6923, 79.7436, 81.7949, 83.3333, 85.1282, 86.4103, 87.9487, 89.4872, 93.
3333, 95.3846, 98.2051, 56.6667, 59.2308, 60.7692, 63.0769, 64.1026, 64.359, 74.359, 71.2821, 6
7.9487, 65.8974, 63.0769, 61.2821, 58.7179, 55.1282, 52.3077, 49.7436, 47.4359, 44.8718, 48.717
9, 51.2821, 54.1026, 56.1538, 52.0513, 48.7179, 47.1795, 46.1538, 50.5128, 53.8462, 57.4359, 60.
0, 64.1026, 66.9231, 71.2821, 74.359, 78.2051, 67.9487, 68.4615, 68.2051, 37.6923, 39.4872, 91.2
821, 50.0, 47.9487, 44.1026]
```

In [30]:
```
using StatsPlots
scatter(datasaurus[:,1], datasaurus[:,2]; label="", title="Datasaurus")
```

Out[30]:

問題: [Datasaurusについて検索 (https://www.google.com/search?q=Datasaurus)](https://www.google.com/search?q=Datasaurus)して見つけた解説を読め.

4 中心極限定理のプロット

4.1 素朴なワークフロー

以下のセルの内容を julia の `julia>` プロンプトに順番に入力すれば(コピー＆ペーストすれば)同じ結果が得られる. 各行の最後にセミコロン `;` を追加すれば計算結果の出力を抑制できる.

In [31]:
```
using StatsPlots, Distributions
```

In [32]:
```
# 確率分布を dist と書く.
dist = Gamma(2, 3)
```

Out[32]:
```
Gamma{Float64}(α=2.0, θ=3.0)
```

In [33]:

```
# 確率分布 dist のサイズ n のサンプルを L 個生成
n = 10
L = 10^4
Xs = [rand(dist, n) for _ in 1:L]
```

Out[33]:

```
10000-element Vector{Vector{Float64}}:
 [3.3296118654446527, 0.83295869875402, 1.2707188263984772, 1.8368281021157462, 1.666171484123
7993, 1.3253406397535328, 6.468173216119855, 2.8407578877908235, 3.422879160271081, 4.85117213
5543707]
 [1.5685939681638787, 11.94548387464656, 4.7857004062567565, 4.286473392183487, 4.836729827951
89, 5.918148733393701, 7.002560657607356, 3.0879139993637974, 5.639378080404118, 11.3364446507
98997]
 [3.52195562286654, 13.907112731848198, 6.692030085665792, 2.011867508449219, 10.6820865659086
53, 2.012281066577551, 7.596794024468705, 5.711664633465194, 13.327656588418373, 11.1712871748
44625]
 [6.751830708375646, 4.795131408357556, 7.781621221816383, 0.3534992736641132, 4.3509355507081
05, 4.066024903904726, 5.508541597581041, 4.492675505222506, 14.191625233951939, 5.64011954487
712]
 [8.613773101558804, 4.772092367038441, 6.138274799455682, 10.726744750330692, 6.5200072722492
45, 2.1612888673830484, 3.246440024720253, 1.6688407136567294, 6.200246878210799, 1.9385713175
12742]
 [21.39529295486351, 2.9657425396451664, 6.382197111599519, 2.5883892085061646, 6.332522367761
911, 11.942663464960845, 8.879941924471224, 10.411285502542086, 10.855277548937096, 9.06210483
7429793]
 [1.6055247200718563, 10.68813053908945, 4.5489707665114985, 5.451065895682635, 3.402093560786
9305, 3.300746021962254, 6.184936211255145, 11.909524833356189, 7.050381863896479, 3.543705310
0636584]
 [1.3166446535029752, 9.614268937810735, 2.5068093499518374, 1.8817759297895016, 4.97940700249
1489, 1.6227165840660969, 5.176208118153195, 7.564761049164044, 1.696907214136183, 2.764632941
1562354]
 [9.453611009032466, 8.991634061969577, 1.376103373884916, 1.1936018896367546, 7.1285437907236
275, 3.2533741775781393, 6.554132402834774, 3.9704436201825883, 5.064519865683151, 2.634588914
330506]
 [9.86484235511811, 12.679599631891325, 3.87561269319907, 2.2316090283337604, 4.35354582455952
95, 3.8013271407620874, 4.14441921808316, 16.629675972123376, 1.4973206928458522, 3.8665753552
4746]
 [1.212576529765503, 12.641974924080042, 2.5678652906711275, 4.720672882488012, 3.827614874343
7464, 13.475936823695783, 3.1125506219963683, 4.647517394033705, 1.405092045776946, 14.5108318
71385987]
 [8.881548716557253, 4.568722827260618, 8.870271587952015, 4.07291478081042, 4.95049385611510
4, 8.970854655801492, 7.301767300937174, 12.432630329705777, 4.685230503694575, 2.556068160494
1236]
 [8.266419402747609, 1.3174055072517525, 1.0858458996313212, 2.8193900427767984, 3.53310437277
65383, 5.027921129608131, 10.87105536506613, 1.3499128584862206, 21.89276008289808, 4.22753006
8642161]
 ⋮
 [1.5424282651090855, 8.338133825881924, 1.9134732663982421, 3.9231680328538148, 1.06905578115
6501, 16.72831334368079, 0.305224527356358, 0.2667405993863962, 2.0435661372056035, 2.36210760
76274026]
 [5.5642091601748715, 2.6489345100276687, 4.950118511436671, 2.9194916840589196, 3.46087194457
77845, 14.086308840951492, 6.130466096889542, 3.136945605302111, 2.3449979573146935, 2.1196538
109620793]
 [6.371795177807512, 9.063183627757057, 2.280758982667171, 8.102532760103273, 1.80758070223258
4, 3.3134281067758504, 4.734047238516372, 4.003921568346781, 13.20694190837742, 4.173387290511
294]
 [7.539865360590447, 5.95962239803662, 9.546011297915417, 3.2243138455969595, 0.93647052741390
15, 1.2156439524747786, 5.9862253810968795, 3.4058457971208522, 3.3574803422850854, 3.63980849
7112239]
 [4.4729804417466354, 3.3033057968420954, 6.37915464706959, 3.3674677831739235, 1.443952940025
2176, 6.087508577130226, 6.692137226905423, 1.9985049194193159, 7.197554944818265, 5.818807729
688202]
 [4.571334231694506, 5.275210522837547, 0.8085608370874227, 2.553319538470323, 5.2117285700010
78, 19.110908509080634, 1.8110957546301631, 7.774829202147897, 5.233205258728834, 7.3088998865
517105]
 [4.991754343069065, 6.423731293390635, 1.932171238488043, 9.923030324448117, 7.82491628273350
5, 2.2086439738743837, 12.012828497092167, 3.8468519621213293, 8.293686194545101, 16.487263210
745578]
 [2.291281059582799, 7.381206810256263, 5.439659870176777, 3.348113385129214, 4.30750018949939
6, 2.259256538762605, 5.97745974697421, 3.0864174542378464, 9.16055100823622, 14.4258884529558
56]
 [1.094727294838912, 3.8090576208706297, 2.592213908870006, 7.272571264165248, 30.618460637038
144, 3.0503390898575136, 6.249065053375926, 7.8319999091103245, 3.4382534253822428, 4.82822716
0207275]
 [5.3112845458523505, 5.1470065209434495, 10.989155421725561, 2.3290452299383233, 7.5006482787
85869, 1.6194106476540646, 13.134504957079667, 3.883731301918996, 1.4213797122618348, 16.69990
1520442044]
 [5.490094809082453, 1.669267751830311, 4.781413762023396, 2.5218515544624145, 9.5702379211172
```

```
91, 2.1065751615201744, 14.915499880652991, 6.357949563763956, 12.508485402049889, 6.611825200
4089655]
 [2.930929662885148, 5.14193080803294, 3.7797805171136023, 4.519595407692449, 3.92926319513137
75, 2.2083000680629015, 6.2948433260466, 9.498636460929445, 5.652982646248763, 8.5613706029345
17]
```

In [34]:

```
# L個のサイズnのサンプルの各々の標本平均を計算
Xbars = mean.(Xs)
```

Out[34]:

```
10000-element Vector{Float64}:
 2.784461201631569
 6.0407427590770535
 7.663473600251285
 5.793200494845914
 5.198628009211643
 9.08154174607173
 5.768507972267609
 3.9124131780222293
 4.962055310585649
 6.294452791216374
 6.212263325823722
 6.729050271932854
 6.039134472988474
 ⋮
 3.8492211386656114
 4.736199812169583
 5.705757736309531
 4.481128739964318
 4.6761375006818895
 5.965909231123012
 7.394487732050791
 5.767733451581118
 7.078491536371622
 6.803606813660217
 6.653320100691184
 5.251763269507774
```

In [35]:

```
# Xbarのヒストグラムを表示
histogram(Xbars; norm=true, alpha=0.5, label="", title="$dist, n=$n")
```

Out[35]:

In [36]:

```
# 中心極限定理による正規分布近似を設定
mu = mean(dist)
sigma = std(dist)
normal_approx = Normal(mu, sigma/sqrt(n))
```

Out[36]:

```
Normal{Float64}(μ=6.0, σ=1.3416407864998736)
```

In [37]:

```
# 上のグラフに重ねて正規分布をプロット
plot!(normal_approx; label="normal approx", lw=2)
```

Out[37]:

$n = 10$ が小さすぎてずれが大きい.

In [38]:

```
# nを大きくしてやり直してみる.
n = 100
L = 10^4
Xs = [rand(dist, n) for _ in 1:L]
Xbars = mean.(Xs)
histogram(Xbars; norm=true, alpha=0.5, label="", title="$dist, n=$n")
mu = mean(dist)
sigma = std(dist)
normal_approx = Normal(mu, sigma/sqrt(n))
plot!(normal_approx; label="normal approx", lw=2)
```

Out[38]:

$n = 100$ にしたら, 正規分布とよく一致するようになった.

In [39]:

```
# Lも大きくしてやり直してみる.
n = 100
L = 10^5
Xs = [rand(dist, n) for _ in 1:L]
Xbars = mean.(Xs)
histogram(Xbars; norm=true, alpha=0.5, label="", title="$dist, n=$n")
mu = mean(dist)
sigma = std(dist)
normal_approx = Normal(mu, sigma/sqrt(n))
plot!(normal_approx; label="normal approx", lw=2)
```

Out[39]:

In [40]:

```
# Lも大きくしてやり直してみる.
n = 1000
L = 10^5
Xs = [rand(dist, n) for _ in 1:L]
Xbars = mean.(Xs)
histogram(Xbars; norm=true, alpha=0.5, label="", title="$dist, n=$n")
mu = mean(dist)
sigma = std(dist)
normal_approx = Normal(mu, sigma/sqrt(n))
plot!(normal_approx; label="normal approx", lw=2)
```

Out[40]:

4.2 Revise.jlを使うワークフロー

上のように素朴に毎回コードを入力することは非常に面倒である.

似た仕事は函数化して1行の入力で実行できるようにしておく方がよい.

しかし, 函数の定義を `julia>` プロンプトに直接入力すると, 試行錯誤で函数の定義を何度も変える作業が非常に面倒になる.

もしも, 函数の定義をファイルに書いておき, ファイル内の函数の定義を書き換えると, 自動的に `julia>` プロンプトの側に函数の定義の変更が反映されるようにできれば非常に便利である. それを実現するのが [Revise.jl (https://github.com/timholy/Revise.jl)](https://github.com/timholy/Revise.jl) パッケージである. Revise.jlパッケージは

```
pkg> add Revise
```

でインストールできる.

4.3 問題: Revise.jlを実際に使ってみよ

`using Revise` を実際に使ってみよ.

例えば, 以下のセルの内容で `central_limit_theorem.jl` という名前のファイルを作成し,

```
julia> using Revise
julia> includet("central_limit_theorem.jl")
julia> hello_sine()
```

を実行してみよ. 他の函数の実行例については下の方を参考にせよ.

`using Revise` 後に `includet` (最後の `t` を忘れずに)したファイルの内容を変更すると自動的に `julia>` プロンプト側(juliaのREPL側(read-eval-print loop側))に反映されるので, それ以後は

- `includet` で読み込んだファイルを変更する.
- 例えば, 函数の内容を変更したり, 新しい函数を書いたりする.
- その結果は自動的に `julia>` プロンプト側に反映される.
- ファイルの変更結果をテストするために函数を実行する.

の繰り返して, 色々な仕事をできるようになる.

In [41]:

```julia
using StatsPlots
using Distributions
default(size=(400, 250), titlefontsize=10)

function hello_sine()
    println("Hello, Sine!")
    plot(sin; label="y=sin(x)")
end

function plot_central_limit_theorem(dist, n; L=10^4, bin=:auto)
    distname = mydistname(dist)
    mu = mean(dist)
    sigma = std(dist)
    Xs = [rand(dist, n) for _ in 1:L]
    Xbars = mean.(Xs)
    normal_approx = Normal(mu, sigma/sqrt(n))

    if dist isa DiscreteUnivariateDistribution
        mu = mean(dist)
        sigma = std(dist)
        a = round(n*mu - 4.5sqrt(n)*sigma)
        b = round(n*mu + 4.5sqrt(n)*sigma)
        ran = a-0.5:b+0.5
        bin = ran / n
    end

    histogram(Xbars; bin, norm=true, alpha=0.5, label="Xbars")
    plot!(normal_approx; lw=2, label="normal approx")
    title!("$distname, n=$n")
end

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))
mydistname(dist) = replace(string(dist), r"{[^}]*}"=>"")

function plot_dist(dist; xlim0=nothing)
    distname = mydistname(dist)
    if isnothing(xlim0)
        mu = mean(dist)
        sigma = std(dist)
        a = max(minimum(dist), mu - 4.5sigma)
        b = min(maximum(dist), mu + 4.5sigma)
        if dist isa DiscreteUnivariateDistribution
            a, b = a-1, b+1
        else
            a, b = a-0.025(b-a), b+0.025(b-a)
        end
        xlim0 = (a, b)
    end
    plot(x -> mypdf(dist, x), xlim0...; label="", title="$distname")
end

function plot_dist_clt(dist, n; L=10^4, xlim0=nothing)
    P0 = plot_dist(dist; xlim0)
    P1 = plot_central_limit_theorem(dist, n; L)
    plot(P0, P1; size=(800, 250), layout=(1, 2))
end
```

Out[41]: plot_dist_clt (generic function with 1 method)

In [42]:
```
hello_sine()
```

```
Hello, Sine!
```

Out[42]:

In [43]:
```
plot_dist_clt(Uniform(), 10)
```

Out[43]:

In [44]:
```
plot_dist_clt(Laplace(), 10)
```

Out[44]:

In [45]:
```
plot_dist_clt(Gamma(2, 3), 10)
```

Out[45]:

In [46]:
```
plot_dist_clt(Gamma(2, 3), 100)
```

Out[46]:

In [47]:
```
plot_dist_clt(Gamma(2, 3), 1000)
```

Out[47]:

In [48]:
```
plot_dist_clt(Exponential(), 10)
```

Out[48]:

In [49]:
```
plot_dist_clt(Exponential(), 100)
```

Out[49]:

In [50]:
```
plot_dist_clt(Bernoulli(), 10)
```

Out[50]:

In [51]:
```
plot_dist_clt(Bernoulli(), 100)
```

Out[51]:

In [52]:
```
plot_dist_clt(Bernoulli(0.2), 10)
```

Out[52]:

In [53]:
```
plot_dist_clt(Bernoulli(0.2), 100)
```

Out[53]:

In [54]:
```
plot_dist_clt(Poisson(), 10)
```

Out[54]:

In [55]:
```
plot_dist_clt(Poisson(), 100)
```

Out[55]:

In [ ]:
```
```

検定と信頼区間: 一般論

- 黒木玄

- 2022-05-31～2022-06-05

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 お勧め解説動画とお勧め文献
  - 1.1 お勧め解説動画
  - 1.2 お勧め文献: P値に関するASA声明
- 2 まとめ
  - 2.1 P値
  - 2.2 検定
  - 2.3 信頼区間
  - 2.4 nuisanceパラメータがある場合
- 3 P値の定義
  - 3.1 統計モデルの設定
  - 3.2 P値の定義
  - 3.3 「データの数値以上に極端な値」の意味の定義の仕方
  - 3.4 例(二項分布モデルの場合)
  - 3.5 例(正規分布の標本分布モデルの場合)
  - 3.6 P値は帰無仮説下の統計モデルのデータの数値との整合性の指標
- 4 検定
- 5 第一種の過誤(αエラー)
  - 5.1 第一種の過誤(αエラー)の定義
  - 5.2 第一種の過誤(αエラー)の確率
  - 5.3 二項分布モデルの4種のP値に関する第一種の過誤の確率のグラフ
- 6 信頼区間
  - 6.1 信頼区間のP値もしくは検定を用いた定義
  - 6.2 信頼区間の使い方
- 7 信頼区間と検定の表裏一体性
  - 7.1 検定における棄却領域の合併と信頼区間全体の合併は互いに相手の補集合
  - 7.2 信頼区間と検定の表裏一体性について書かれた教科書の例
  - 7.3 仮説 $\theta = \theta_0$ 下の統計モデル内でパラメータ値 $\theta = \theta_0$ が信頼区間に含まれる確率
  - 7.4 二項分布モデルでのP値函数の視覚化
- 8 統計モデルやP値函数が「よい」かどうかの判断基準達
  - 8.1 計算方法はシンプルな方がよい
  - 8.2 頑健な方がよい
  - 8.3 第一種の過誤の確率は有意水準に近い方がよい
  - 8.4 第一種の過誤の確率は有意水準以下である方がよい
  - 8.5 検出力は高い方がよい
- 9 Neyman-Pearsonの仮説検定
  - 9.1 パラメータを持たない2つの統計モデルに関する仮説検定
  - 9.2 Neyman-Pearsonの補題
  - 9.3 Neyman-Pearsonの補題の証明
  - 9.4 尤度比検定の例: 正規分布モデルの場合
  - 9.5 Neyman-Pearsonの仮説検定に関する極端な解釈の普及の問題
  - 9.6 このノートの立場
- 10 よくある誤解

10.1 P値の定義と使い方の復習
10.2 問題: P値に関するよくある誤解
10.3 信頼区間の定義の復習と使い方の復習
10.4 問題: 信頼区間に関するよく見る誤解
10.5 平均の信頼区間達の視覚化
10.5.1 正規分布の標本達から得られる平均の信頼区間達
10.5.2 ガンマ分布の標本から得られる平均の信頼区間達
10.5.3 対数正規分布の標本達から得られる平均の信頼区間達
10.6 問題解答例: P値に関するよくある誤解
10.7 問題解答例: 信頼区間に関するよく見る誤解

In [1]:

```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using Distributions
using LinearAlgebra
using Printf
using QuadGK
using RCall # requires the R language.
@rlibrary exactci # requires the package exactci of R.
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10,  plot_titlefontsize = 10)
using SymPy
```

In [2]:

```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```
safemul(x, y) = x == 0 ? x : x*y
safediv(x, y) = x == 0 ? x : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

In [4]:

```julia
# 二項分布モデルの4種のP値函数

# Clopper-Pearson

function pvalue_clopper_pearson(n, k, p)
    bin = Binomial(n, p)
    min(1, 2cdf(bin, k), 2ccdf(bin, k-1))
end

# Sterne

_pdf_le(x, (dist, y)) =  pdf(dist, x) ≲ y

function _search_boundary(f, x0, Δx, param; maxiters=10^7)
    x = x0
    if f(x, param)
        for _ in 1:maxiters
            !f(x - Δx, param) && return x
            x -= Δx
        end
    else # if !f(x, param)
        for _ in 1:maxiters
            x += Δx
            f(x, param) && return x
        end
    end
    error("""
    _search_boundary($f, $x0, $Δx, param = $param; maxiters = $maxiters) \
    has exceeded the maximum number $maxiters of iterations.""")
end

function pvalue_sterne(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    0 < Px < 1 || return Px
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = _search_boundary(_pdf_le, 2m - x, 1, (dist, Px))
        cdf(dist, x) + ccdf(dist, y-1)
    else # x > m
        y = _search_boundary(_pdf_le, 2m - x, -1, (dist, Px))
        cdf(dist, y) + ccdf(dist, x-1)
    end
end

pvalue_sterne(n, k, p) = pvalue_sterne(Binomial(n, p), k)

# Wilson

function pvalue_wilson(n, k, p)
    z = safediv(k - n*p, √(n*p*(1-p)))
    2ccdf(Normal(), abs(z))
end

# Wald

function pvalue_wald(n, k, p)
    z = safediv(k - n*p, √(k*(n-k)/n))
    2ccdf(Normal(), abs(z))
end
```

Out[4]: pvalue_wald (generic function with 1 method)

In [5]:

```julia
function illustrate_binomial_pvalues()
    n = 20
    p = 0.25
    k = 9

    p̂ = k/n
    σ̂² = n*p̂*(1-p̂)

    null = Binomial(n, p)
    μ, σ² = mean(null), var(null)
    nullname = distname(null)

    f(x) = mypdf(null, x)
    xlim = (-2, n+1)
    xlimk = (k-0.5, n+0.5)

    pval = @sprintf "%.03f" pvalue_clopper_pearson(n, k, p)
    cval = @sprintf "%.03f" ccdf(null, k-1)
    P1 = plot(title="(1) Clopper-Pearson")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    annotate!(k+5, 4f(k), text("P-value = $pval", 10, :red))
    plot!([k+7, k+7], [2.4f(k), 3.6f(k)]; arrow=true, c=:red, label="")
    annotate!(k+6, 3f(k), text("2×", 10, :red, :right))
    annotate!(k+5, 2f(k), text("one-tailed = $cval", 10, :red))
    plot!([k+7, k+1], [1.5f(k), 1.5f(k+1)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_sterne(n, k, p)
    j = _search_boundary(_pdf_le, 2mode(null)-k, -1, (null, pdf(null, k)))
    xlimj = (-0.5, j+0.5)
    P2 = plot(title="(2) Sterne")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    plot!(f, xlimj...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!([k-0.5, j+0.5], [f(k), f(k)]; label="", c=2, ls=:dash)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+1], [2.5f(k), 0.8f(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+1], [2.5f(k), 0.8f(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wilson(n, k, p)
    normal = Normal(μ, √σ²)
    g(x) = pdf(normal, x)
    P3 = plot(title="(3) Wilson")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(g, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ²)", c=2)
    plot!(g, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(g, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wald(n, k, p)
    normal = Normal(μ, √σ̂²)
    h(x) = pdf(normal, x)
    P4 = plot(title="(4) Wald")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(h, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ̂²)", c=2)
    plot!(h, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(h, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    plot(P1, P2, P3, P4; size=(800, 520), layout=(2, 2))
    plot!(; plot_title="model: $nullname,   data: n=$n, k=$k")
end

#illustrate_binomial_pvalues()
```

Out[5]: illustrate_binomial_pvalues (generic function with 1 method)

In [6]:
```
# 二項分布モデルの4種のP値函数のプロット

function plot_binom_pvaluefunctions(; n = 20, k = 6)
    P1 = plot(p -> pvalue_clopper_pearson(n, k, p), 0, 1;
        label="", title="Clopper-Pearson", c=1)
    P2 = plot(p -> pvalue_sterne(n, k, p), 0, 1;
        label="", title="Sterne", c=2)
    P3 = plot(p -> pvalue_wilson(n, k, p), 0, 1;
        label="", title="Wilson", c=3)
    P4 = plot(p -> pvalue_wald(n, k, p), 0, 1;
        label="", title="Wald", ls=:dash, c=4)
    plot(P1, P2, P3, P4; size=(800, 520), layout=(2, 2))
    plot!(; xtick=0:0.1:1, ytick=0:0.1:1)
    plot!(; xguide="success rate parameter p",
            yguide="P-value", guidefontsize=10)
    plot!(; plot_title="data: n = $n, k = $k",
            plot_titlefontsize=12)
    plot!(; titlefontsize=12)
end

#plot_binomial_pvaluefunctions(; n = 20, k = 6)
```

Out[6]: plot_binom_pvaluefunctions (generic function with 1 method)

In [7]:
```
# 二項分布モデルのP値函数の動画（データ k を動かしてアニメ化）

function gif_binomtest(; n = 20, fn = "images/pvaluefunction20.gif", fps = 5, step=1)
    anim = @animate for k in [0:step:n; n-1:-step:1]
        plot(p -> pvalue_clopper_pearson(n, k, p), 0, 1; label="Clopper-Pearson")
        plot!(p -> pvalue_sterne(n, k, p), 0, 1; label="Sterne")
        plot!(p -> pvalue_wilson(n, k, p), 0, 1; label="Wilson")
        plot!(p -> pvalue_wald(n, k, p), 0, 1; label="Wald", ls=:dash)
        plot!(; xtick=0:0.1:1, ytick=0:0.1:1)
        plot!(; xguide="success rate parameter p", yguide="P-value")
        title!("data: n = $n, k = $k")
        plot!(; size=(600, 300))
        2k > n && plot!(; legend=:topleft)
    end
    gif(anim, fn; fps)
end
```

Out[7]: gif_binomtest (generic function with 1 method)

In [8]:
```
# pvalue(x|p) 達のヒートマップ（明るい所ほど値が大きい）

function heatmap_binom_pvaluefunctions(; n = 20, xtick = 0:n)
    k = 0:n
    p = 0:0.01:1

    P1 = heatmap(k, p, (k, p)->pvalue_clopper_pearson(n, k, p);
        colorbar=false, title="(1) Clopper-Pearson")
    P2 = heatmap(k, p, (k, p)->pvalue_sterne(n, k, p);
        colorbar=false, title="(2) Sterne")
    P3 = heatmap(k, p, (k, p)->pvalue_wilson(n, k, p);
        colorbar=false, title="(3) Wilson")
    P4 = heatmap(k, p, (k, p)->pvalue_wald(n, k, p);
        colorbar=false, title="(4) Wald")
    plot(P1, P2, P3, P4; size=(800, 810), layout=(2, 2),
        xtick, ytick=0:0.1:1, tickfontsize=7,
        xguide="k", yguide="p")
    plot!(; plot_title="data size: n = $n", plot_titlefontsize=12)
end

#heatmap_binom_pvaluefunctions(; n = 100, xtick=0:10:100)
```

Out[8]: heatmap_binom_pvaluefunctions (generic function with 1 method)

In [9]:

```julia
# pvalue(x|p) ≥ α のヒートマップ

function heatmap_binom_rejectionregions(; α = 0.05, n = 20, xtick = 0:n)
    k = 0:n
    p = 0:0.01:1

    c = cgrad([colorant"red", colorant"blue"])
    alpha = 0.5
    P1 = heatmap(k, p, (k, p)->pvalue_clopper_pearson(n, k, p) ≥ α;
        colorbar=false, title="(1) Clopper-Pearson", c, alpha)
    P2 = heatmap(k, p, (k, p)->pvalue_sterne(n, k, p) ≥ α;
        colorbar=false, title="(2) Sterne", c, alpha)
    P3 = heatmap(k, p, (k, p)->pvalue_wilson(n, k, p) ≥ α;
        colorbar=false, title="(3) Wilson", c, alpha)
    P4 = heatmap(k, p, (k, p)->pvalue_wald(n, k, p) ≥ α;
        colorbar=false, title="(4) Wald,", c, alpha)
    plot(P1, P2, P3, P4; size=(800, 800), layout=(2, 2),
        xtick, ytick=0:0.1:1, tickfontsize=7,
        xguide="k", yguide="p")
    plot!(; plot_title="significane level: α = $α,  \
        data size: n = $n", plot_titlefontsize=12)
end

#heatmap_binom_rejectionregions(; α = 0.05, n = 100, xtick=0:10:100)
```

Out[9]: heatmap_binom_rejectionregions (generic function with 1 method)

In [10]:

```julia
# 二項分布モデルの4種のP値函数に関するシミュレーション

function sim_binomtest(n, p; L=10^5)
    bin = Binomial(n, p)
    p_clopper_pearson = Vector{Float64}(undef, L)
    p_sterne = Vector{Float64}(undef, L)
    p_wilson = Vector{Float64}(undef, L)
    p_wald = Vector{Float64}(undef, L)
    @threads for i in 1:L
        k = rand(bin)
        p_clopper_pearson[i] = pvalue_clopper_pearson(n, k, p)
        p_sterne[i] = pvalue_sterne(n, k, p)
        p_wilson[i] = pvalue_wilson(n, k, p)
        p_wald[i] = pvalue_wald(n, k, p)
    end
    F_clopper_pearson = ecdf(p_clopper_pearson)
    F_sterne = ecdf(p_sterne)
    F_wilson = ecdf(p_wilson)
    F_wald = ecdf(p_wald)
    (; F_clopper_pearson, F_sterne, F_wilson, F_wald)
end

function plot_sim_binomtest(n, p; L=10^5, xmax=1)
    (; F_clopper_pearson, F_sterne, F_wilson, F_wald) = sim_binomtest(n, p; L)
    x = range(0, xmax, 1001)
    P1 = plot(x, x->F_clopper_pearson(x);
        label="", title="(1) Clopper-Pearson", c=1)
    plot!(x, x; label="", ls=:dot, c=:black)
    P2 = plot(x, x->F_sterne(x);
        label="", title="(2) Sterne", c=2)
    plot!(x, x; label="", ls=:dot, c=:black)
    P3 = plot(x, x->F_wilson(x);
        label="", title="(3) Wilson", c=3)
    plot!(x, x; label="", ls=:dot, c=:black)
    P4 = plot(x, x->F_wald(x);
        label="", title="(4) Wald", c=4)
    plot!(x, x; label="", ls=:dot, c=:black)

    plot(P1, P2, P3, P4; size=(700, 720), layout=(2,2))
    plot!(; xtick=0:0.1xmax:1, ytick=0:0.1xmax:1, tickfontsize=7)
    plot!(; xlim=(-0.02xmax, 1.02xmax), ylim=(-0.02xmax, 1.02xmax))
    plot!(; xguide="significance level α", yguide="probability of type I error")
    plot!(; plot_title="data: n = $n, p = p₀ = $p", plot_titlefontsize=12)
end
```

Out[10]: plot_sim_binomtest (generic function with 1 method)

In [11]:

```julia
function pvalue_t(X, μ)
    n = length(X)
    X̄ = mean(X)
    S² = var(X)
    T = (X̄ - μ)/√(S²/n)
    2ccdf(TDist(n-1), abs(T))
end

function confint_t(X; α = 0.05)
    n = length(X)
    X̄ = mean(X)
    S² = var(X)
    SE = √(S²/n)
    t = quantile(TDist(n-1), 1 - α/2)
    (X̄ - t*SE, X̄ + t*SE)
end
```

Out[11]: confint_t (generic function with 1 method)

In [12]:

```julia
function illustrate_ttest(;
    x = [10, 4, 20, 10, 18, 7, 5, 8, 4, 16, 8])
    n = length(x)
    x̄ = mean(x)
    s² = var(x)
    μ = 7
    t = (x̄ - μ)/√(s²/n)
    pval = 2ccdf(TDist(n-1), abs(t))
    pval_str = @sprintf "%.03f" pval

    println("null hypothesis: μ = ", μ)
    println("data: x = ", x)
    println("data size: n = ", n)
    println("summary statistics; x̄ = ", x̄, ", s² = ", s²,
        ", √s² = ", round(√s²; digits=1))
    println("t-statistics: t = (x̄ - μ)/√(s²/n) = ", t)
    println("degree of freedom: n - 1 = ", n - 1)
    println("P-value: 2(1 - cdf(TDist(n-1), abs(t))) = ", pval)

    P1 = plot()
    plot!(TDist(n-1), -5, 5; label="TDist($(n-1))", c=1)
    plot!(TDist(n-1), abs(t), 5; label="", c=1, fillrange=0, fc=:blue, fa=0.3)
    plot!(TDist(n-1), -5, -abs(t); label="", c=1, fillrange=0, fc=:blue, fa=0.3)
    plot!([t, t], [0, 3pdf(TDist(n-1), t)]; label="t = $(round(t; digits=2))", c=:blue)
    plot!(Normal(), -5, 5; label="Normal(0,1)", ls=:dash, c=2)
    plot!(; xtick=-10:10)
    annotate!(-5, 0.25, text("P-value = $pval_str", 10, :blue, :left))
    plot!([-2, -2], [0.23, 0.08]; arrow=true, label="", c=:blue)
    plot!([-2, 1.7], [0.23, 0.04]; arrow=true, label="", c=:blue)

    ccdf_t = @sprintf "%.03f" ccdf(TDist(n-1), t)
    ccdf_n = @sprintf "%.03f" ccdf(Normal(), t)
    P2 = plot()
    plot!(TDist(n-1), 1, 5; label="TDist($(n-1))", c=1)
    plot!(TDist(n-1), abs(t), 5; label="", c=1, fillrange=0, fc=:blue, fa=0.3)
    plot!([t, t], [0, 3pdf(TDist(n-1), t)]; label="t = $(round(t; digits=2))", c=:blue)
    plot!(Normal(), 1, 5; label="Normal(0,1)", ls=:dash, c=2)
    plot!(; xtick=-10:10)
    annotate!(5, 0.15, text("1 - cdf(TDist($(n-1)), t) = $ccdf_t", 10, :blue, :right))
    annotate!(5, 0.12, text("1 - cdf(Normal(0,1), t) = $ccdf_n", 10, :red, :right))

    plot_title = """
    null hypothesis: μ = $μ,  \
    data: x = [10, 4, 20, 10, 18, 7, 5, 8, 4, 16, 8],  \
    size: n = $n
    """
    plot(P1, P2; size=(800, 270))
    plot!(; plot_title)
end

#illustrate_ttest()
```

Out[12]: illustrate_ttest (generic function with 1 method)

In [13]:
```
function illustrate_confintmean(; dist = Normal(), n = 30, α = 0.05)
    μ_true = mean(dist)

    plot()
    for i in 1:200
        X = rand(dist, n)
        ci = collect(confint_t(X))
        c = pvalue_t(X, mean(dist)) < α ? 2 : 1
        plot!([i, i], ci; label="", lw=2, c)
    end
    plot!(size=(800, 220))
    hline!([mean(dist)]; label="", c=:black, lw=0.3)
    title!("""
        $(100(1-α))% confidence intervals for mean of \
        size-$n samples of $(distname(dist))
        """)
end

#illustrate_confintmean(dist = LogNormal())
```

Out[13]: illustrate_confintmean (generic function with 1 method)

In [14]:
```
# 正規分布モデルで定めた平均のt検定のシミュレーション

function sim_ttest(; dist=Normal(), n=20, μ=mean(dist), L=10^5)
    pvals = Vector{Float64}(undef, L)
    tmp = [Vector{eltype(dist)}(undef, n) for _ in 1:nthreads()]
    @threads for i in 1:L
        X = rand!(dist, tmp[threadid()])
        pvals[i] = pvalue_t(X, μ)
    end
    ecdf_pvals = ecdf(pvals)
end

function plot_sim_ttest(; dist=Normal(), n=30, μ=mean(dist),
        dist_str=distname(dist), L=10^5)
    ecdf_pvals = sim_ttest(; dist, n, μ, L)
    f(x) = ecdf_pvals(x)
    plot(; legend=:topleft)
    plot!(f, 0, 1; label="")
    plot!(identity, 0, 1; label="", c=:black, lw=0.3, ls=:dash)
    plot!(; xlim=(-0.005, 0.155), ylim=(-0.005, 0.155))
    plot!(; xtick=0:0.01:1, ytick=0:0.01:1, xrotation=45,
        tickfontsize=7)
    plot!(; xguide="α", yguide="probability of P-value < α")
    title!("""
        ecdf of t-test P-values for mean
        sample: $(dist_str), n=$n\
        """)
    plot!(; size=(400, 415))
end

#plot_sim_ttest(; dist=LogNormal())
```

Out[14]: plot_sim_ttest (generic function with 1 method)

In [15]:
```
Random.seed!(4649373)
```

Out[15]: TaskLocalRNG()

1 お勧め解説動画とお勧め文献

1.1 お勧め解説動画

P値と検定と信頼区間については次のリンク先の動画での解説が素晴らしいので, 閲覧を推奨する:

- 京都大学大学院医学研究科 聴講コース
  臨床研究者のための生物統計学「仮説検定とP値の誤解」
  佐藤 俊哉 医学研究科教授
  https://youtu.be/vz9cZnB1d1c (https://youtu.be/vz9cZnB1d1c)

信頼区間の解説は40分あたり以降にある.

多くの入門的な解説が抱えているP値, 検定, 信頼区間の解説の難点は以下の2つに要約される:

- 複雑な現実と統計モデルを混同させるような解説が伝統的に普通になってしまっていること.
- 検定と信頼区間の表裏一体性(双対性)が解説されていないこと.

そのせいで, P値も 95% 信頼区間の 95% も数学的フィクションである統計モデル内での確率であることがクリアに説明されておらず, P値と 95% 信頼区間の 95% についてまっとうな理解が得られ難くなっている.

上で紹介した動画は教科書の説明がまずいことについて明瞭に言及しながら, 伝統的な入門的解説が抱えている問題を解消しようとしている.

注意: 上の解説動画内で説明されている事柄を理解すれば, P値と検定と信頼区間について広まってしまった誤解を避けることができる. そして, その後は個別の場合について詳しく勉強するだけの問題になるだろう.

1.2 お勧め文献: P値に関するASA声明

- 統計的有意性と P 値に関する ASA 声明 (日本語訳) [PDF (https://www.biometrics.gr.jp/news/all/ASA.pdf)]

P値に関する解説ではこれが非常によい. しかし, その第2節の

> 2. P値とは?
>
> おおざっぱにいうと、P 値とは特定の統計モ デルのもとで、データの統計的要約（たとえば、 2 グループ比較での標本平均の差）が観察され た値と等しいか、それよりも極端な値をとる確率 である。

という説明における「データ」という用語の解釈には注意を要する. この意味での「データ」は現実世界での観察で得たデータのことではなく, **統計モデル内で生成された仮想的なデータ**のことである.

現実世界で得たデータは決まった数値になるが, 数学的フィクションである統計モデル内で生成されたデータは確率変数とみなせ, それが現実世界での観察で得たデータの数値以上の極端な値を取る確率を考えることができる.

統計学は, 現実世界でデータを得る活動と数学的世界で統計モデルを考えることを行ったり来たりするので, 現実とモデルを混同しないように注意しなければいけない.

P値は現実において常に意味を持つ確率またはその近似値ではなく, 数学的フィクションである統計モデル内部で測った確率に過ぎない.

2 まとめ

以下のまとめは「理解が進むたびに繰り返しこのまとめに戻る」というような使い方をして欲しい.

2.1 P値

P値 (P-value)は以下を与えることによって定義される:

- 現実世界におけるデータの数値 x の生成のされ方に関するパラメータ θ を持つ統計モデル,
- 「データの数値以上に極端な」の意味の定義,
- さらに必要ならば近似計算法.

データの数値 x とパラメータの値 $\theta = \theta_0$ が与えらえたとき, P値は

- **データの数値 x 以上に極端な値が条件 $\theta = \theta_0$ の下での統計モデル内で生じる確率もしくはその近似値**

と定義される. このとき, 条件 $\theta = \theta_0$ は **帰無仮説** (null hypothesis)と呼ばれることが多い. (「データの数値以上に極端な」の定義は **対立仮説** (alternative hypothesis)の **集まり** を与えることによって与えらえると考えられる.)

このノートでは以上のように定義されたP値を

- データの数値 x に関する仮説 $\theta = \theta_0$ のP値

と呼ぶことにする.

P値は以下の2つの整合性の指標として使われる:

- 現実世界から得たデータの数値 x,
- 統計モデル+パラメータの値 $\theta = \theta_0$.

P値が小さいほど, 統計モデル+パラメータの値 θ は現実世界から得たデータの数値 x と整合していないと考える.

2.2 検定

有意水準 (significance level)と呼ばれる閾値(いきち, しきいち) $0 < \alpha < 1$ が与えられたとき, P値が α 未満ならば,

- 統計モデル+パラメータの値は現実世界から得たデータの数値と整合性がない

とみなす. このとき

- 統計モデル+パラメータの値は現実世界から得たデータによって **棄却** (reject)されたという.

この手続きを **検定** (test, testing, hypothesis testing)と呼ぶ. (P値が有意水準 α 以上になるとき, 「受容 (accept)された」ということがあるが, 混乱の原因になるので, このノートでは用いない. このノートでは「棄却された」「棄却されなかった」の組み合わせを一貫して使うことにする.)

棄却されなかった統計モデルとパラメータの値の組み合わせについては強い結論は何も出せない.

棄却されずにすんだ統計モデルとパラメータの値の組み合わせは単に閾値 α の設定で捨てられずにすんだだけなので, 「棄却されなかった統計モデルとパラメータの値の組み合わせは妥当である」のように考えることは典型的な誤解になる.

有意水準 α として 5% が非常によく使われているが, それは単に慣習的にそうなっているだけのことで, 5% の有意水準を使うことに科学的な合理性はない.

5% の有意水準の下での結果に一喜一憂することは非科学的な考え方である.

検定の手続きは「科学的お墨付きを得るための手段」ではない!

同じことは次の節で説明する信頼区間についても言える.

2.3 信頼区間

信頼区間 (confidence interval)の文脈で $1 - \alpha$ は **信頼度** (信頼係数, 信頼水準, confidence level)と呼ばれる.

統計モデルが実数パラメータ θ を持つとき, データの数値 x から決まる信頼度 $1 - \alpha$ の信頼区間は

- データの数値 x と有意水準 α で棄却されない統計モデルのパラメータ値 θ 全体の集合

として定義される. (この集合が区間にならない場合には信頼領域(confidence region)と呼んだりする. その集合を含む最小の区間を考える場合もある.)

信頼区間を使うことは, 検定の手続きを無数のパラメータ値 θ 達に対して適用することと同じである.

検定で棄却されなかった場合については強い結論は何も出せないので, 信頼区間は

- 妥当性について判断を保留するべきパラメータ値全体の集合

だとみなされる.

2.4 nuisanceパラメータがある場合

注意: この節の内容はこのノートの内容を超えて先走っている.

実際には統計モデルは興味があるパラメータ θ 以外にパラメータ η を含んでいることがある. (例えば平均パラメータ μ のみに興味があるときの正規分布 $\mathrm{Normal}(\mu, \sigma)$ における標準偏差パラメータ σ.)

その場合にはパラメータ θ の値を決めても, 統計モデルの確率分布は唯一つに決まらず, パラメータ η の分だけ不定になる.

P値の定義は「データの数値 x 以上に極端な値が帰無仮説 $\theta = \theta_0$ の下での統計モデル内で生じる確率もしくはその近似値」であった. 帰無仮説 $\theta = \theta_0$ で統計モデルの確率分布が唯一つに決まらない場合にはこのP値の定義を単純に適用することが不可能になる.

このような状況のときに, 余計なパラメータ η を **nuisanceパラメータ** (ニューサンスパラメータ, 局外パラメータ, 撹乱パラメータ, 迷惑パラメータ)と呼ぶ(nuisanceは迷惑や妨害を意味する名詞).

大抵の場合にnuisanceパラメータが存在しているという問題は, P値を使う統計分析の基礎付けが複雑になる主な原因の1つになっている.

nuisanceパラメータへの対処法には例えば以下がある:

- nuisanceパラメータを動かして上限(supremum)を考える.
- 条件付き確率分布に移ってnuisanceパラメータの自由度を消す.
- 最尤法の漸近論を使ってnuisanceパラメータによらずに成立する近似を得る.
- Bayes法を使う.

3 P値の定義

3.1 統計モデルの設定

まず, 現実世界におけるデータの生成のされ方のモデル化として使われる統計モデルを考える.

例 (二項分布モデル):

- 現実世界のデータ: 当たりとはずれが出るルーレットを n 回まわしたときの当たりの回数 k
- 統計モデル: 二項分布モデル $\mathrm{Binomial}(n, p)$

例 (正規分布の標本分布モデル):

- 現実世界のデータ: S市の中学3年生男子全体から n 人を無作為抽出して測った身長の数値達 $x_1, \ldots, x_n$
- 統計モデル: 正規分布のサイズ n の標本分布モデル $\mathrm{Normal}(\mu, \sigma)^n$

このように統計モデルは多くの場合にパラメータ付きの確率分布として与えられる.

3.2 P値の定義

データ x の生成のされ方のモデル化になっているパラメータ θ を持つ統計モデルが与えられているとする.

データの数値 x とパラメータの数値 $\theta = \theta_0$ が与えられたとき,

データの数値 x から定まる仮説 $\theta = \theta_0$ のP値

を次によって定める:

データの数値 x 以上に極端な値が仮説 $\theta = \theta_0$ 下の統計モデル内で生じる確率もしくはその近似値.

「仮説 $\theta = \theta_0$ 下の統計モデル内におけるデータの数値 x 以上に極端な値」の定義は目的ごとに別に与えられる. その概略については後の方の説明を参照せよ.

仮説 $\theta = \theta_0$ は **帰無仮説** (null hypothesis)と呼ばれることが多い.

まとめ: P値＝帰無仮説下のモデル内でデータの数値以上に極端な値が生じる確率(の近似値)

補足: 「x が x' 以上に極端な値である」という条件は **反射性**(reflexivity)と **推移性** (transitivity)と **比較可能性**(comparability)の3つの条件を満たしていることを要請する. すなわち「x が x' 以上に極端な値である」ことを $x \succcurlyeq x'$ と書くと, 以下が成立していると仮定する:

- 反射性(reflexibity): $x \succcurlyeq x$,
- 推移性(transitivity): $x \succcurlyeq x'$ and $x' \succcurlyeq x'' \implies x \succcurlyeq x''$,
- 比較可能性(comparability): $x \succcurlyeq x'$ or $x' \succcurlyeq x$.

注意: 実際にはこの基本に忠実にP値が定義されるとは限らない.

3.3 「データの数値以上に極端な値」の意味の定義の仕方

注意: 以下の説明は抽象的過ぎて分かりにくいので, 後の節の具体例の方を先に参照した方がよいと思われる.

P値の定義を確定させるためには, データの数値 x とパラメータの数値 $\theta = \theta_0$ が与えられたとき, 統計モデルとそのパラメータ値が与える確率分布に従う確率変数 X の値がデータの数値 x_0 以上に極端な値であることの定義を, 統計分析の目的に合わせて適切に設定する必要がある.

「仮説 $\theta = \theta_0$ 下の統計モデルに従う確率変数 X がデータの数値 x 以上に極端な値であること」の定義として, 以下のような条件がよく使われる:

(0) $X \geq x$ (または $X \leq x$).

この(0)の条件は例えばχ²検定達で使われる.

さらにこの(0)の条件は帰無仮説 $\theta \leq \theta_0$ を(θ の値を θ_0 以上のどれかとする)を対立仮説 $\theta > \theta_0$ (θ の値を θ_0 未満のどれかとする)と比較する **片側検定** で使用される.

しかし, 主に使われるのは(一部に見かけ上の例外もある(例: χ²検定達)), 帰無仮説 $\theta = \theta_0$ を対立仮説 $\theta \neq \theta_0$ (θ の値を θ_0 以外のどれかの値とする)と比較する **両側検定** の場合である. 以下はすべて両側検定の場合である. (用語「対立仮説」についてはNeyman-Pearsonの仮説検定の節を参照せよ.)

(1) $X \geq x$ と $X \leq x$ の確率が小さい方の条件. (この場合にP値はその確率の2倍にする.)

(2) モデル内での X の値が生じる確率(もしくはその密度)がデータの数値 x 以下である.

X が実数値の確率変数ではなく, $\mathbb{R}^n$ 値の確率変数の場合には, X の実数値函数 $S(X|\theta_0)$ を用意して,

(3) $S(X|\theta_0) \geq S(x|\theta_0)$ (もしくは $S(X|\theta_0) \leq S(x|\theta_0)$)

という条件で「仮説 $\theta = \theta_0$ の下での統計モデルに従う確率変数 X がデータの数値 x 以上に極端な値であること」を定義することが多い. 函数 $S(x|\theta_0)$ は **検定統計量** と呼ばれ, 目的ごとに適切に選択する必要がある.

(4) 他にも正規分布近似を使う方法も多用される.

3.4 例(二項分布モデルの場合)

注意: 以下の文章による説明はわかりにくいので, 下の方に用意してある図を先に見た方が良いかもしれない.

データの数値「n 回中 k 回成功」と成功確率パラメータの数値 $p = p_0$ が与えられているとし, 仮説 $p = p_0$ 下の二項分布 $\mathrm{Binomial}(n, p_0)$ に従う確率変数 K を用意する:

$$K \sim \mathrm{Binomial}(n, p_0).$$

このとき, 「仮説 $p = p_0$ 下の二項分布モデル内での成功回数 K の値がデータの数値 k 以上に極端であること」を以下のように, 互いに同値でない様々な方法で定義できる:

(0) $K \geq k$ (もしくは $K \leq k$)という条件で「k 以上に極端」の意味を定義する. この定義はぞれぞれ仮説 $p \leq p_0$ (もしくは $p \geq p_0$)の **片側検定** (one-tailed test, one-sided test) で使われる.

(1) **Clopper-Pearsonの信頼区間の場合:** 仮説 $p = p_0$ の下での二項分布モデル内での

$K \geq k$ と $K \leq k$ の確率の小さい方の2倍(と 1 の小さい方)

を **両側検定** のP値として使う.

以下の定義達も両側検定の場合になっている. **検定は, 通常, 両側検定を使用する.**

(2) **Sterneの信頼区間の場合:** K の値がモデル内で生じる確率がデータの数値 k がモデル内で生じる確率以下になるという条件で, すなわち,

二項分布の確率質量函数を $P(k|n, p_0)$ と書くとき, $P(K|n, p_0) \leq P(k|n, p_0)$

という条件で「k 以上に極端」の意味を定義する. すなわち,

仮説 $p = p_0$ 下の統計モデル内でデータの数値 k 以上に確率的に珍しいこと

を「k 以上に極端」の定義とする.

以下のように正規分布近似(中心極限定理)を使って定義することもできる.

(3) **Wilsonの信頼区間の場合:** 二項分布 $\mathrm{Binomial}(n, p_0)$ に関する中心極限定理によれば, $(K - np_0)/\sqrt{np_0(1-p_0)}$ は np と $n(1-p)$ が十分に大きければ近似的に標準正規分布に従う. 「標準正規分布 $\mathrm{Normal}(0, 1)$ に従う確率変数 Z の値がデータの数値以上に極端であること」を

$$|Z| \geq \frac{|k - np_0|}{\sqrt{np_0(1-p_0)}}$$

という条件で定め, こうなる確率を標準正規分布を使って計算してP値とする(近似の一種).

(3)' **Pearsonのχ²検定の場合:** 上の(3)と同値な次の条件を使うこともある:

$$Z^2 \geq \frac{(k - np_0)^2}{np_0(1-p_0)}.$$

$Z \sim \mathrm{Normal}(0, 1)$ のとき, Z^2 は自由度 1 のχ²分布に従うので, こうなる確率を自由度 1 のχ²分布を用いて計算してP値とする(これも近似の一種). さらに上の条件は次とも同値である:

$$Z^2 \geq \frac{(k - np_0)^2}{np_0} + \frac{((n-k) - n(1-p_0))^2}{n(1-p_0)}.$$

これの不等式の右辺を **Pearsonのχ²統計量** と呼ぶ. これはそのように呼ばれる検定統計量達

$$\chi^2 = \sum \frac{(\text{現実における観測値} - \text{帰無仮説下の統計モデル内での期待値})^2}{\text{帰無仮説下の統計モデル内での期待値}}$$

の特別な場合に過ぎない. 上の場合に「現実における観測地」は n 回中の成功回数 k と失敗回数 $n-k$ であり, 「帰無仮説下の統計モデル内での期待値」は成功回数の期待値 np_0 と失敗回数の期待値 $n(1-p_0)$ である. (より進んだ注意: Pearsonのχ²統計量はスコア検定における検定統計量になっている.)

(4) **Waldの信頼区間の場合:** 上の(3)における右辺の分母の p_0 をパラメータの推定量 $\hat{p} = k/n$ で置き換えて得られる次の条件で「データの数値以上に極端であること」を定義することもある:

$$|Z| \geq \frac{|k - np_0|}{\sqrt{n\hat{p}(1-\hat{p})}}.$$

以上のように, 二項分布モデルのP値の定義の仕方も沢山ある. 基本的にどれを使ってもよい. ユーザー側は自分の目的に合わせて合理的だと考えられるものを自由に使えばよい.

In [16]:
```
illustrate_binomial_pvalues()
```

Out[16]:

注意: Waldの信頼区間を与えるP値の計算で使っている正規分布近似では, 正規分布の分散としてモデルの分散 $np(1-p) = 3.75$ ではなく, データ「$n = 20$ 回中 $k = 9$ 回成功」から推定した $k(n-k)/n = 4.95$ を使っているせいで, 正規分布近似の精度が低くなっている. 精度が高いのはWilsonの信頼区間を与えるP値の方である. しかし, 信頼区間の計算はWaldの側がずっと簡単になる.

注意: ここでは詳しく説明できないが, Sterneの信頼区間の方がClopper-Pearsonの信頼区間よりも被覆確率という基準で精度が高いことが知られている. 原論文でもそのことが指摘されている:

- Theodore E. Sterne, Some Remarks on Confidence or Fiducial Limits, Biometrika, Vol. 41, No. 1/2 (Jun., 1954), pp. 275-278 [link (https://www.jstor.org/stable/2333026)]

しかし, 計算法はClopper-Pearsonの信頼区間の方がずっとシンプルである.

注意: P値の定義の仕方の各々には利点と欠点がある.

3.5 例(正規分布の標本分布モデルの場合)

注意: この節の内容は別のノートで詳しく説明する. この段階では以下で説明する複雑な式を覚える必要はない.

データの数値 $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ と興味があるパラメータの数値 $\mu = \mu_0$ が与えられているとする. このとき, 「仮説 $\mu = \mu_0$ の下での正規分布の標本分布モデルに従う確率変数 $X = (X_1, \ldots, X_n)$ の値がデータの数値 $x = (x_1, \ldots, x_n)$ 以上に極端であること」を以下の方法で定義するとよいことが知られている.

まず, T 統計量 $T(X|\mu)$ を次のように定める:

$$T(x|\mu) = \frac{\bar{x} - \mu}{\sqrt{s^2/n}}, \quad \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

そして, 「仮説 $\mu = \mu_0$ の下で確率変数 X の値がデータの数値 x 以上に極端であること」を

$$|T(X|\mu_0)| \geq |T(x|\mu_0)|$$

という条件で定める. これは大雑把に言って,

- モデル内で生成された仮想的なデータ X の標本平均 $\bar{X}$ が $\mu = \mu_0$ からデータの標本平均 $\bar{x}$ 以上に離れている.

という意味の条件になっている. ただし, 分散を推定して適当にスケールして比較している.

これが便利なのは, 仮説 $\mu = \mu_0$ の下での正規分布の標本分布モデル内で, **分散パラメータ σ^2 がどんな値であっても,** 確率変数 $T(X|\mu_0)$ が自由度 $n-1$ の t 分布に従うからである.

注意: 統計モデルが θ 以外にパラメータ η を持っていて, 帰無仮説 $\theta = \theta_0$ を課しても, パラメータ η の分だけ統計モデルの確率分布が唯一つに決まらないとき, η を **nuisanceパラメータ** (ニューサンスパラメータ, 局外パラメータ, 撹乱パラメータ, 迷惑パラメータ)と呼ぶ. 上の例では分散パラメータ σ^2 がnuisanceパラメータになっている. そして, 上の例はnuisanceパラメータの問題をシンプルに解決できる稀有な場合になっている.

In [17]:
```
illustrate_ttest()
```

```
null hypothesis: μ = 7
data: x = [10, 4, 20, 10, 18, 7, 5, 8, 4, 16, 8]
data size: n = 11
summary statistics; x̄ = 10.0, s² = 31.4, √s² = 5.6
t-statistics: t = (x̄ - μ)/√(s²/n) = 1.7756312235479013
degree of freedom: n - 1 = 10
P-value: 2(1 - cdf(TDist(n-1), abs(t))) = 0.106177674255521
```

Out[17]:

注意: 自由度 $n-1 = 10$ の t 分布 $\mathrm{TDist}(10)$ と標準正規分布 $\mathrm{Normal}(0,1)$ は非常に近いように見えるが, その裾野の部分の確率の大きさで違いが生じていることに注意せよ.

問題: サイズ $n = 11$ のデータ

```
x = [10, 4, 20, 10, 18, 7, 5, 8, 4, 16, 8]
```

と仮説 $\mu = 7$ について, 標本平均 $\bar{x}$ と不偏分散 s^2 と $t = (\bar{x} - \mu)/\sqrt{s^2/2}$ を自分で計算してみて, 上の計算例と一致することを確認せよ. さらに, インターネットで検索して, t 検定のP値を求める方法を調べ, 実際に計算してみて, 上の計算例と一致することを確認せよ.

3.6 P値は帰無仮説下の統計モデルのデータの数値との整合性の指標

データの数値 x から計算される帰無仮説 $\theta = \theta_0$ のP値は, データの数値 x と仮説 $\theta = \theta_0$ 下の統計モデルの整合性の指標である.

P値が小さいとき, 仮説 $\theta = \theta_0$ 下の統計モデルとデータの数値 x があまり整合していないと考える.

これがP値の基本的な使い方である.

まとめ: P値はデータの数値と帰無仮説下のモデルの整合性の指標.

4 検定

データ x の生成のされ方のモデル化になっているパラメータ θ を持つ統計モデルが与えられていると仮定する.

さらに **有意水準** と呼ばれる **閾値**(いきち, しきいち) $0 \leq \alpha \leq 1$ が与えられていると仮定する. α は目的に合わせて適当に小さな値としておく. (有意水準として 5% がよく用いられているが, そのことに科学的な合理性はない.)

データの数値 x とパラメータの数値 $\theta = \theta_0$ が与えられているとき, (帰無)仮説 $\theta = \theta_0$ のP値を求め, P値が α 未満になるとき, 仮説 $\theta = \theta_0$ 下の統計モデルは **棄却** (reject)されたという. (実際には「帰無仮説は棄却された」と略した言い方をすることが多い.)

この手続きを **仮説検定** (Hypothesis tesitng)もしくは単に **検定** (test)と呼ぶ.

すなわち, 検定とは, **ある閾値を設けて, データの数値との整合性が閾値未満のモデルのパラメータ値を捨て去る手続き** のことである.

ただし, 検定は閾値を設けて捨て去る行為なので, 捨て去り過ぎてしまう誤りを犯すリスクがある.

有意水準が小さなほどそのリスクは小さくなるが, その分だけ, 科学的に興味深い結果を見逃してしまうリスクが増える.

まとめ: 検定＝閾値を設けてデータの数値との整合性が低過ぎるパラメータ値を捨て去る手続き.

5 第一種の過誤(αエラー)

5.1 第一種の過誤(αエラー)の定義

X はパラメータの数値を $\theta = \theta_0$ に設定した統計モデルに従う確率変数であると仮定する.

データの数値がその確率変数 X の値であるときの仮説 $\theta = \theta_0$ のP値が有意水準 α 未満になる確率を仮説 $\theta = \theta_0$ に関する **第一種の過誤の確率** または **αエラーの確率** と呼ぶ.

すなわち, 仮説 $\theta = \theta_0$ 下の統計モデル内部において, その仮説 $\theta = \theta_0$ が検定の手続きで棄却されてしまうことを **第一種の過誤** (Type I error, αエラー, α-error)と呼び, その確率を **第一種の過誤の確率** と呼ぶ.

第一種の過誤は検定する仮説 $\theta = \theta_0$ が成立しているモデル内部においてその仮説が棄却されてしまうことを意味しており, その確率は小さい方がよい. しかし, 第一種の過誤の確率を小さくすることが原因で生じる害(後で述べる検出力の低下)もあるので, 目的に合わせてバランスを取ることが必要である.

注意: 第一種の過誤は「無実のものを有罪にしてしまう誤り」に例えられることがある.

注意: 第一種の過誤は統計モデルが自分自身を確率的に否定してしまうことであると言うこともできる.

注意: 現実世界から得たデータの数値 x は値が確定した定数だが, 数学的フィクションである統計モデル内ではデータの数値が確率変数 X としてランダムに生成されていると考えることができる. 多くの教科書でこの辺について誤解に誘導するような説明があるので注意して欲しい.

まとめ: 第一種の過誤＝帰無仮説下の統計モデル内で帰無仮説が棄却されること.

5.2 第一種の過誤(αエラー)の確率

定理: 有意水準 α の下での第一種の過誤の確率の値は α で近似される.

証明: x の値が x' 以上に極端であるという条件を $x \succcurlyeq x'$ と書き, X はパラメータ値 $\theta = \theta_0$ の統計モデルに従う確率変数であると仮定する.

このとき, P値の定義(X の値が x 以上に極端な値になる確率もしくはその近似値)より,

$$(\text{データの数値 } x \text{ に関する仮説 } \theta = \theta_0 \text{ のP値}) \approx P(X \succcurlyeq x).$$

$P(X \succcurlyeq x_\alpha) \approx \alpha$ となる x_α を取る. このとき,

$$\begin{aligned}
&(\text{データの数値 } x \text{ について仮説 } \theta = \theta_0 \text{ のP値が } \alpha \text{ 未満になるという条件}) \\
&\approx (P(X \succcurlyeq x) < \alpha \text{ という条件}) \\
&\approx (x \succcurlyeq x_\alpha \text{ という条件}).
\end{aligned}$$

したがって,

$$
\begin{aligned}
&(\text{仮説 } \theta = \theta_0 \text{ に関する第一種の過誤の確率}) \\
&= (\text{データの数値が確率変数 } X \text{ の値であるときに 仮説 } \theta = \theta_0 \text{ のP値が } \alpha \text{ 未満になる確率}) \\
&\approx (X \succcurlyeq x_\alpha \text{ となる確率}) = P(X \succcurlyeq x_\alpha) \approx \alpha.
\end{aligned}
$$

証明終

注意: 上の問題の結果は, 仮説 $\theta = \theta_0$ 下の統計モデル内部において, 同仮説 $\theta = \theta_0$ のP値の分布が一様分布で近似されることを意味している. 次の節の計算例を参照せよ.

注意: 実際にはP値を基本に忠実でない方法で定義することもあり, 上の定理の証明が適用できない場合もあるのだが, 検定の理論では第一種の過誤の確率が有意水準に近くなることを要請することが普通である.

まとめ: 第一種の過誤の確率は有意水準に等しいかそれに近い値になる.

これは実際には「要請」であり, 証明されるべき結果ではないとみなされる.

5.3 二項分布モデルの4種のP値に関する第一種の過誤の確率のグラフ

二項分布モデルの4種のP値について, 0 から 1 のあいだの有意水準 α 達について第一種の過誤の確率を計算してグラフを描いてみよう.

そのグラフは, 仮説 $p = p_0$ 下の二項分布モデル内部における同仮説 $p = p_0$ のP値の分布の累積分布函数のグラフだと言ってよいので, そのグラフが「45度線」に近ければ近いほど, その分布は一様分布に近いということになる.

In [18]:
```
plot_sim_binomtest(20, 0.3)
```

Out[18]:

In [19]:

```
plot_sim_binomtest(100, 0.3)
```

Out[19]:

In [20]:
```
plot_sim_binomtest(1000, 0.3)
```

Out[20]:

確かに二項分布モデルの4種のP値の帰無仮説下の二項分布内での累積分布函数は, 確かに45度線を近似していることが分かる. こ

のことはそのP値の分布が一様分布で近似されることを意味している.

6 信頼区間

6.1 信頼区間のP値もしくは検定を用いた定義

有意水準 α の検定の手続きをパラメータ θ のすべての値に適用したとき, 棄却されなかったパラメータ値全体の集合をパラメータ θ に関する **信頼度** (信頼係数) $1-\alpha$ の **信頼区間** (confidence interval)と呼ぶ. ($\alpha = 5\%$ のとき, 信頼度 $1-\alpha$ の信頼区間を 95% 信頼区間と呼ぶことが多い.)

有意水準 α の検定の手続きでパラメータの値 $\theta = \theta_0$ が棄却されることは, 仮説 $\theta = \theta_0$ 下の統計モデルとデータの値 x との整合性(P値)が有意水準 α 未満になることであった. そのような状況を

- 仮説 $\theta = \theta_0$ 下の統計モデルとデータの値 x との整合性が無さすぎる

と言うことにしよう. このスタイルの下では, 信頼区間は

- 統計モデルの下で, データの数値 x との整合性が無さすぎないパラメータ値全体の集合

であると言える.

まとめ: 信頼区間＝検定で棄却されないモデルのパラメータ値全体の集合.

6.2 信頼区間の使い方

「整合性が無さすぎないこと」は「正しいこと」を意味しないし, 「正しい可能性が高いこと」も意味しない.

信頼区間に含まれるパラメータ値の下での統計モデルの妥当性については判断を保留しなければいけない.

例: 例えば, パラメータ θ がある治療法の治療効果を意味するパラメータであったとしよう. そのとき, 現実世界における調査で得たデータの数値 x に関する θ の信頼区間は

- その区間に含まれる治療効果の数値の各々については, 現実における正しさの判断を保留する.
- その区間のどれかの値が真の治療効果であっても大丈夫なようにしておく.
- 得られたデータの数値へのその統計モデルの使用が妥当でない可能性についても常に注意を払う.

のような使い方をすることが妥当だと思われる.

まとめ: 信頼区間に含まれるパラメータ値の下での統計モデルの妥当性については判断を保留する.

7 信頼区間と検定の表裏一体性

7.1 検定における棄却領域の合併と信頼区間全体の合併は互いに相手の補集合

データ x の生成のされ方のモデル化になっているパラメータ θ を持つ統計モデルが設定されていると仮定し, データの数値 x に関する仮説 $\theta = \theta_0$ のP値 $\mathrm{pvalue}(x|\theta_0)$ が定義されていると仮定し, 有意水準 α が与えられているとする.

データの数値 x が与えらえたとき, パラメータの数値 $\theta = \theta_0$ をP値 $\mathrm{pvalue}(x|\theta_0)$ に対応させる函数

$$\theta_0 \mapsto \mathrm{pvalue}(x|\theta_0)$$

を **P値函数** (P-value function)と呼ぶ. P値函数は無数の仮説 $\theta = \theta_0$ 達の検定結果の情報をすべて持っているので, 信頼区間の情報もそこに含まれる.

このとき, 仮説 $\theta = \theta_0$ を有意水準 α で棄却するようなデータの数値全体の集合を **棄却領域** (rejection region)と呼び, 次のように書くことにする:

$$R_\alpha(\theta_0) = \{\, x \mid \mathrm{pvalue}(x|\theta_0) < \alpha \,\}.$$

信頼度 $1-\alpha$ の信頼区間 $C_\alpha(x)$ は

$$C_\alpha(x) = \{\, \theta_0 \mid \mathrm{pvalue}(x|\theta_0) \geq \alpha \,\}$$

と書ける. これらの関係は以下の図のようになっている.

以下の図を見れば, すべてのデータの数値 x について信頼区間 $C_\alpha(x)$ を与えることと, すべてのパラメータの数値 $\theta = \theta_0$ についての棄却領域 $R_\alpha(\theta_0)$ を与えることが同じであることがわかる. そして, 図中の赤の領域と青の領域の境界上でのP値 $\mathrm{pvalue}(x|\theta_0)$ の値が有意水準 α になる.

まとめ: 信頼区間と検定は表裏一体である.

7.2 信頼区間と検定の表裏一体性について書かれた教科書の例

以下の教科書にはこのノートが採用した信頼区間と検定の表裏一体性に関する考え方が書いてある.

- 竹内啓, 数理統計学—データ解析の方法, 東洋経済新報社, 1963 (のp.103)
- 小針晛宏, 確率・統計入門, 岩波書店, 1973 (のp.197)
- 久保川達也, 現代数理統計学の基礎, 共立出版, 2017 (のp.169)
- 竹村彰通, 新装改訂版 現代数理統計学, 学術図書出版社, 2020 (のp.202)

これらの教科書は有名である.

- https://twitter.com/genkuroki/status/1531827825879445505 (https://twitter.com/genkuroki/status/1531827825879445505)

にその部分の引用がある.

まとめ: 信頼区間と検定が表裏一体であることは有名な教科書群に書いてある.

7.3 仮説 $\theta = \theta_0$ 下の統計モデル内でパラメータ値 $\theta = \theta_0$ が信頼区間に含まれる確率

(x, θ_0) がこの図中の(信頼区間を含む)青の領域の点であることは, 仮説 $\theta = \theta_0$ がデータの数値 x によって棄却されないことと同値であり, さらに, データの数値 x から決まる信頼区間にパラメータ値 $\theta = \theta_0$ が含まれることとも同値である.

X がパラメータ値 $\theta = \theta_0$ の統計モデルに従う確率変数であると仮定する. X は仮説 $\theta = \theta_0$ 下の統計モデル内で生成された仮想的なデータの数値だと考えられる.

P値の定義より $\mathrm{pvalue}(X|\theta_0) < \alpha$ となる確率は α もしくはその近似になるので, $\mathrm{pvalue}(X|\theta_0) \geq \alpha$ となる確率は $1 - \alpha$ もしくはその近似値になる.

そして, $\mathrm{pvalue}(X|\theta_0) \geq \alpha$ という条件は仮説 $\theta = \theta_0$ 下の統計モデル内で生成されたデータの値 X から決まる信頼度 $1 - \alpha$ の信頼区間に θ_0 が含まれることと同値である.

ゆえに, 仮説 $\theta = \theta_0$ 下の統計モデル内で生成されたデータの値 X から決まる信頼度 $1 - \alpha$ の信頼区間にパラメータ値 $\theta = \theta_0$ が含まれる確率は $1 - \alpha$ またはその近似値になる.

例えば, 大雑把に言うと, 95% 信頼区間に統計モデル内でデータを生成したパラメータ値が含まれる確率は 95% またはその近似値になる.

注意: この事実はP値の定義より, 95% 信頼区間の 95% が確率とみなされることを意味している. ただし, その確率は数学的フィクションである **統計モデル内で測った確率** に過ぎない. 何らかの特別な理由があって, 現実のデータの数値 x の生成のされ方を統計モデルが忠実に再現していると考えられるならば, 95% という数値は現実においても意味を持ち得る. しかし, そうでない場合いは, 単なる **モデル内確率** であることに十分な注意を払う必要がある.

注意: 「95% 信頼区間の 95% は確率ではなく, 割合である」というようなことが多くの教科書に書かれているようだが, 以上の説明を読めば分かるように自明に誤りである.

注意: 「95% 信頼区間の 95% は確率ではなく, 割合である」と言いたい人達は, データの数値 x はすでに確定した数値になっており, 確率的に揺らがないので, データの数値 x から計算される信頼区間も確率的に揺らがないので, パラメータ値 $\theta = \theta_0$ が信頼区間に含まれる確率を考えることはできない, というようなことを言いたいのかもしれないが, 上の説明を読めば分かるように的を外している. 95% 信頼区間の 95% は数学的フィクションであるモデル内確率であり, モデル内で生成されたデータの数値は確率変数になっている. 現実とモデルを混同するという典型的に非科学的な考え方に陥っている疑いもある.

まとめ: 仮説 $\theta = \theta_0$ 下の統計モデル内部において, 信頼区間がパラメータ値 $\theta = \theta_0$ を含む確率は信頼度 $1 - \alpha$ に等しい, もしくはそれに近い値になる.

これも証明するべきことではなく, 「要請」だと考えた方がよい.

7.4 二項分布モデルでのP値函数の視覚化

文献: P値函数の使い方については次の文献が詳しい:

- Timothy L. Lash, Tyler J. VanderWeele, Sebastien Haneuse, and Kenneth J. Rothman.
  Modern Epidemiology, 4th edition, 2020. [Google (https://www.google.com/search?q=Modern+Epidemiology+4th)](https://www.google.com/search?q=Modern+Epidemiology+4th)

第4版よりも古い版でもよい.

In [21]:
```
plot_binom_pvaluefunctions(; n = 20, k = 6)
```

Out[21]:

以上は「$n = 20$ 回中 $k = 6$ 回成功」というデータの数値が与えられたときの, P値函数のグラフである.

P値が小さいほどデータの数値と成功確率パラメータ値が p の二項分布モデルのあいだに整合性がないと考える.

上のグラフを見れば, P値函数による判定では, 「$n = 20$ 回中 $k = 6$ 回成功」というデータの数値に最も整合するパラメータ p の値は $p = 6/20 = 0.3$ になっており, そこから離れると整合性が下がる.

どの場合も,「$n = 20$ 回中 $k = 6$ 回成功」というデータの数値にあまりにも整合しないパラメータ p の値全体の集合の補集合は大雑把に 0.1 から 0.6 までの区間になることもわかる. これが信頼区間である. ただし, 有意水準を決めずにグラフの見た目で判断してどんぶり勘定で区間を決めた. 信頼区間の正式な定義はP値が有意水準という名の閾値 α 以上になるパラメータの範囲である.

信頼区間では「そこに含まれるか否か」(α 以上か否か)の情報しか残っていないが, P値函数の様子を直接見れば, パラメータを動かしたときの統計モデルとデータの数値の整合性の度合いについても知ることができる.

注意: 有意水準として $\alpha = 5\%$ がよく使われているが, そのことに科学的合理性はない. だから閾値 $\alpha = 5\%$ で計算した信頼区間を見て一喜一憂するのはバカげている. そのようにバカげたことをするくらいならば, 閾値を決めずにP値函数の様子を直接眺めてどんぶり勘定で判断した方がましだと思われる.

注意: R言語では exactci パッケージを入れると, Clopper-Pearsonの信頼区間を与えるP値函数(`tsmethod = "central"`)とSterneの信頼区間を与えるP値函数(`tsmethod = "minlik"`)を容易にプロットできる.

- https://rdrr.io/cran/exactci/man/binom.exact.html (https://rdrr.io/cran/exactci/man/binom.exact.html)

のExamplesに適当に `, plot=TRUE` を挿入してRunボタンを押してみよ.

In [22]:
```
binom_exact(6, 20; tsmethod = "central", plot = true)
```

Out[22]:
```
RObject{VecSxp}

        Exact two-sided binomial test (central method)

data:  6L and 20L
number of successes = 6, number of trials = 20, p-value = 0.1153
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.1189316 0.5427892
sample estimates:
probability of success
                   0.3
```

In [23]:
```
binom_exact(6, 20; tsmethod = "minlik", plot = true)
```

Out[23]:
```
RObject{VecSxp}

        Exact two-sided binomial test (sum of minimum likelihood method)

data:  6L and 20L
number of successes = 6, number of trials = 20, p-value = 0.1153
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
 0.1396 0.5254
sample estimates:
probability of success
                   0.3
```

In [24]:
```
gif_binomtest()
```

```
┌ Info: Saved animation to
│   fn = D:\OneDrive\work\Statistics\2022\images\pvaluefunction20.gif
└ @ Plots D:\.julia\packages\Plots\1KWPG\src\animation.jl:114
```

Out[24]:

n を大きくするとP値函数の「幅」は狭くなる.

In [25]:
```
plot_binom_pvaluefunctions(; n = 100, k = 30)
```

Out[25]:

In [26]:
```
gif_binomtest(n = 100, fn = "images/pvaluefunction100.gif", fps=20)
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\pvaluefunction100.gif
@ Plots D:\.julia\packages\Plots\1KWPG\src\animation.jl:114
```

Out[26]:

PDFファイルではこの動画を見ることはできない. 作成した動画は

- https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction20.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction20.gif)
- https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction100.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction100.gif)

でも見ることができる.

pvalue(x|p) 達のヒートマップ (明るい所ほど値が大きい)

In [27]:
```
heatmap_binom_pvaluefunctions(; n = 20)
```

Out[27]:

明るい部分ほどP値が大きい. P値が小さな部分はほぼ黒色になっている. そこでは「n 回中 k 回成功」というデータの数値に成功回数パラメータ値が p の二項分布モデルが整合していないと考える.

例えば「$n = 20$ 回中 $k = 6$ 回成功」というデータの数値が得られたとき $k = 6$ でのP値函数の「明るさ」を上のグラフで確認すると, $p = 6/20 = 0.6$ で最も明るくなっており, そこから p が離れると暗くなって行くことがわかる. P値による判定によれば, 「$n = 20$ 回中 $k = 6$ 回成功」というデータの数値に最も整合する二項分布モデルの成功確率パラメータ p の値は $p = 0.3$ であり, そこから p が離れるにつれて整合性は下がって行く.

In [28]:
```
heatmap_binom_pvaluefunctions(; n = 100, xtick = 0:10:100)
```

Out[28]:

このように, n を大きくすると, 明るい部分が「細く」なる. これは, n を大きくすると, k を固定したときに得られるP値函数のグラフが「狭く」なることと同じ.

In [29]:
```
heatmap_binom_rejectionregions(; α = 0.05, n = 20)
```

Out[29]:

薄い赤の領域はP値が $\alpha = 5\%$ 未満になる部分であり, 薄い青の領域はP値が $\alpha = 5\%$ 以上になる部分である. P値の定義の仕方によって結果は異なるが概ね似たような様子になっている.

In [30]:

```
heatmap_binom_rejectionregions(; α = 0.05, n = 100, xtick = 0:10:100)
```

Out[30]:

In [31]:
```
heatmap_binom_rejectionregions(; α = 0.05, n = 1000, xtick = 0:100:1000)
```

Out[31]:

8 統計モデルやP値函数が「よい」かどうかの判断基準達

8.1 計算方法はシンプルな方がよい

このノートで扱っている4種のP値(Clopper-Pearson, Sterne, Wilson, Waldの信頼区間を与えるP値)では, 計算効率を気にする実装を行うと, Sterneの信頼区間を与えるP値の実装がひどく複雑になってしまい, バグも発生し易い.

計算法がシンプルな方が実装のミスも少なく, 計算効率もよいことが多い.

だから, たとえ別の方法が道具としての性能が勝っていたとしても, 計算方法がシンプルな側を採用することは十分に検討に値する.

8.2 頑健な方がよい

統計モデルが現実のデータ生成法則をぴったり記述していると考えることは非現実的な場合が多い.

だから, 統計モデルが現実のデータ生成法則からずれている場合について考えることも非常に重要になる.

統計モデルが現実のデータ生成法則からずれていても統計分析の誤差が小さくなる傾向があるとき, その分析法は **頑健** (robust)であるという.

頑健な方が実践的な適用範囲が広がり, 統計分析の失敗のリスクも減るので好ましい.

8.3 第一種の過誤の確率は有意水準に近い方がよい

第一種の過誤の確率は有意水準に近い方がよい. 応用上特に重要なのは有意水準が $\alpha \leq 5\%$ と小さいところでの近似の精度である.

例えば, 以下の二項分布モデルの4種のP値の例については, Clopper-Pearsonの信頼区間を与えるP値よりもSterneの信頼区間を与えるP値を使った方が誤差は小さくなっており, Waldの信頼区間を与えるP値よりもWilsonの信頼区間を与えるP値を使った方が誤差は小さくなっている.

In [32]:
```
Random.seed!(4649373)
```

Out[32]:
```
TaskLocalRNG()
```

In [33]:
```
plot_sim_binomtest(100, 0.2; xmax=0.1, L=10^6)
```

Out[33]:

In [34]:

```
plot_sim_binomtest(1000, 0.02; xmax=0.1, L=10^6)
```

Out[34]:

In [35]:

```
plot_sim_binomtest(1000, 0.2; xmax=0.1, L=10^6)
```

Out[35]:

8.4 第一種の過誤の確率は有意水準以下である方がよい

「第一種の過誤の確率は有意水準以下である方がよい」という基準のことを **保守性** (conservativity)と呼ぶことがある.

上の例では, Clopper-Pearsonの信頼区間を与えるP値とSterneの信頼区間を与えるP値では, 第一種の過誤の確率が常に有意水準以下になり, **保守的** (conservative)である. しかし, Wilsonの信頼区間を与えるP値とWaldの信頼区間を与えるP値はその意味で保守的ではない(**リベラル** (liberal)ということがある).

8.5 検出力は高い方がよい

大雑把に言うと, 仮説 $\theta = \theta_0$ のP値の **検出力** (power)はその仮説 $\theta = \theta_0$ と別の仮説 $\theta = \theta_1$ との分解能を意味する.

その文脈で別の仮説 $\theta = \theta_1$ を帰無仮説 $\theta = \theta_0$ の **対立仮説** (alternative hypothesis)と呼ぶ.

帰無仮説 $\theta = \theta_0$ のP値の対立仮説 $\theta = \theta_1 (\neq \theta_0)$ に関する **検出力** (power)は

- 対立仮説 $\theta = \theta_1$ 下の統計モデル内部で帰無仮説 $\theta = \theta_0$ のP値が有意水準 α 未満になる確率

と定義される. すなわち,

- X を対立仮説 $\theta = \theta_1$ 下の統計モデルに従う確率変数とし, データの数値を X としたとき仮説 $\theta = \theta_0$ が有意水準 α で棄却される確率

をそのように呼ぶ.

有意水準が等しいなら, 検出力は高い方がよい.

可能ならば任意の θ_1 についてそうであって欲しいがそのようにすることは一般には不可能である.

検出力については次の節も参照せよ.

まとめ: 検出力＝対立仮説下の統計モデル内で帰無仮説が棄却される確率.

復習: 有意水準 ≈ 第一種の過誤の確率＝帰無仮説下の統計モデル内で帰無仮説が棄却される確率.

9 Neyman-Pearsonの仮説検定

9.1 パラメータを持たない2つの統計モデルに関する仮説検定

データ x の生成のされ方に関するパラメータを持たない(もしくはすべてのパラメータの値が固定された)統計モデルが2つ与えられているとする.

簡単のために2つの統計モデルはそれぞれ確率密度函数 $p_0(x)$, $p_1(x)$ によって与えられているとする.

以下では統計モデル $p_0(x)$ を **帰無仮説** (null hypothesis)と呼び, 統計モデル $p_1(x)$ を **対立仮説** (alternative hypothesis)と呼ぶ. (多くの場合に, 帰無仮説は「薬Aには効果がない」のような否定したい仮説になり, 対立仮説は例えば「薬Aには〜という効果がある」のような正しいことを発見したい仮説になる.)

有意水準 $0 \leq \alpha \leq 1$ について, 棄却領域と呼ばれれる x の集合 R_α が与えられており, 帰無仮説 $p_0(x)$ に従う確率変数 X_0 について,

$$P(X_0 \in R_\alpha) \leq \alpha$$

が成立していると仮定する.

このとき, 棄却領域 R_α によって, **帰無仮説に関する有意水準 α の検定法** が与えられたと言い, データの数値 x が R_α に含まれるときに, 帰無仮説 $p_0(x)$ は **棄却** されたと言う.

棄却領域 R_α で与えられた帰無仮説 $p_0(x)$ の検定法における対立仮説 $p_1(x)$ の **検出力** $\mathrm{power}(R_\alpha)$ を, 対立仮説 $p_1(x)$ に従う確率変数 X_1 を使って次のように定める:

$$\mathrm{power}(R_\alpha) = P(X_1 \in R_\alpha).$$

対立仮説の検出力がより高い検定法は **より強力**(more powerful)であるという.

まとめ: 検出力＝power＝対立仮説下の統計モデル内で帰無仮説が棄却される確率.

復習: 第一種の過誤の確率＝帰無仮説下の統計モデル内で帰無仮説が棄却される確率.

9.2 Neyman-Pearsonの補題

尤度比検定 (likelihood ratio test)と呼ばれる検定法を次の棄却領域 L_α によって定める:

$$L_\alpha = \left\{ x \;\middle|\; \frac{p_0(x)}{p_1(x)} < c_\alpha \right\}.$$

ただし, 定数 c_α は $P(X_0 \in L_\alpha) = \alpha$ をみたすものであるとする.

定理(Neyman-Pearsonの補題): 棄却領域 R_α で与えられた任意の検定法について

$$\mathrm{power}(L_\alpha) \geq \mathrm{power}(R_\alpha).$$

すなわち, 尤度比検定は帰無仮説 $p_0(x)$ の検定法の中で対立仮説 $p_1(x)$ の検出力が最大である.

Neyman-Pearsonの補題: 帰無仮説と対立仮説がともにすべてのパラメータが固定された統計モデルであるときには, 尤度比検定が最強力になる.

用語について補足説明: 対立仮説の下で帰無仮説が棄却されない確率を **第二種の過誤の確率** (type II error, βエラー, β-error)と呼ぶ. すなわち,

$$(\text{第二種の過誤の確率}) = 1 - (\text{検出力}).$$

仮説検定法では, 第一種の過誤の確率＝有意水準をある小さな値に定めたとき, 第二種の過誤の確率を小さくしたい.

9.3 Neyman-Pearsonの補題の証明

X_0, X_1 はそれぞれ帰無仮説 $p_0(x)$, 対立仮説 $p_1(x)$ に従う確率変数であるとする. このとき,

$$P(X_0 \in L_\alpha) = \alpha \geq P(X_1 \in R_\alpha).$$

L_α, R_α の補集合を L_α^c, R_α^c と書くことにすると,

$$L_\alpha = (L_\alpha \cap R_\alpha) \cup (L_\alpha \cap R_\alpha^c), \quad R_\alpha = (L_\alpha \cap R_\alpha) \cup (L_\alpha^c \cap R_\alpha). \tag{$*$}$$

$P(X_0 \in L_\alpha) \geq P(X_0 \in R_\alpha)$ より,

$$P(X_0 \in L_\alpha \cap R_\alpha^c) \geq P(X_1 \in L_\alpha^c \cap R_\alpha). \tag{$\sharp$}$$

$x \in L_\alpha$ と $p_1(x) > p_0(x)/c_\alpha$ が同値であることと, すぐ上で示した($\sharp$)と, $x \notin L_\alpha$ と $p_0(x)/c_\alpha \geq p_1(x)$ が同値であることを順番に使うと,

$$\begin{aligned}
P(X_1 \in L_\alpha \cap R_\alpha^c) &= \int_{L_\alpha \cap R_\alpha^c} p_1(x)\,dx \geq \frac{1}{c_\alpha} \int_{L_\alpha \cap R_\alpha^c} p_0(x)\,dx \\
&= \frac{1}{c_\alpha} P(X_0 \in L_\alpha \cap R_\alpha^c) \geq \frac{1}{c_\alpha} P(X_0 \in L_\alpha^c \cap R_\alpha) \\
&= \frac{1}{c_\alpha} \int_{L_\alpha^c \cap R_\alpha} p_0(x)\,dx \geq \int_{L_\alpha^c \cap R_\alpha} p_1(x)\,dx = P(X_1 \in L_\alpha^c \cap R_\alpha).
\end{aligned}$$

上と同様にして($*$)より,

$$P(X_1 \in L_\alpha) - P(X_1 \in R_\alpha) = P(X_1 \in L_\alpha \cap R_\alpha^c) - P(X_1 \in L_\alpha^c \cap R_\alpha) \geq 0.$$

これで示すべきことが示された.

証明終

$\boxed{\text{Neyman-Pearson Lemma}}$

$$P(X_0 \in L_\alpha) = \alpha \geq P(X_0 \in R_\alpha) \overset{\#}{\Rightarrow} P(X_0 \in L_\alpha \cap R_\alpha^c) \geq P(X_0 \in L_\alpha^c \cap R_\alpha)$$

$$\begin{aligned}
&P(X_1 \in L_\alpha \cap R_\alpha^c) \\
&= \int_{L_\alpha \cap R_\alpha^c} p_1(x)\,dx \overset{①}{\geq} \frac{1}{c_\alpha} \int_{L_\alpha \cap R_\alpha^c} p_0(x)\,dx \\
&= \frac{1}{c_\alpha} P(X_0 \in L_\alpha \cap R_\alpha^c) \overset{\#}{\geq} \frac{1}{c_\alpha} P(X_0 \in L_\alpha^c \cap R_\alpha) \\
&= \frac{1}{c_\alpha} \int_{L_\alpha^c \cap R_\alpha} p_0(x)\,dx \underset{②}{\geq} \int_{L_\alpha^c \cap R_\alpha} p_1(x)\,dx \\
&= P(X_1 \in L_\alpha^c \cap R_\alpha)
\end{aligned}$$

$$P(X_1 \in L_\alpha \cap R_\alpha^c) \geq P(X_1 \in L_\alpha^c \cap R_\alpha) \Leftrightarrow \underbrace{P(X_1 \in L_\alpha) \geq P(X_1 \in R_\alpha)}_{\text{示したいこと,}}$$

9.4 尤度比検定の例: 正規分布モデルの場合

$\mu_1 \in \mathbb{R}$ であるとし, $p_0(x)$, $p_1(x)$ が次の場合について考える:

$$p_0(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}, \quad p_1(x) = \frac{e^{-(x-\mu_1)^2/2}}{\sqrt{2\pi}}$$

このとき,

$$\frac{p_0(x)}{p_1(x)} = \exp(\mu_1^2 - 2\mu_1 x).$$

これがある値未満になるという条件は,

- $\mu_1 > 0$ の場合には, x がある値 a より大きいという条件と同値になり,
- $\mu_1 < 0$ の場合には, x がある値 a より小さいという条件と同値になる.

x がそのようになる標準正規分布における確率はそれぞれ

- $\mu_1 > 0$ の場合: $1 - \mathrm{cdf}(\mathrm{Normal}(0,1), a)$,
- $\mu_1 < 0$ の場合: $\mathrm{cdf}(\mathrm{Normal}(0,1), a)$

なので, これらが有意水準 α に等しくなる $a = a_\alpha$ はそれぞれ

- $\mu_1 > 0$ の場合: $a_\alpha = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha)$,
- $\mu_1 < 0$ の場合: $a_\alpha = \mathrm{quantile}(\mathrm{Normal}(0,1), \alpha)$

になる. ゆえに, 帰無仮説 $p_0(x)$ と対立仮説 $p_1(x)$ に関する対数尤度比検定を与える棄却領域はそれぞれ次のようになる:

- $\mu_1 > 0$ の場合: $L_\alpha = \{\, x \mid x > a_\alpha \,\}$,
- $\mu_1 < 0$ の場合: $L_\alpha = \{\, x \mid x < a_\alpha \,\}$.

これは以下のように解釈される.

- 対立仮説の平均 μ_1 が帰無仮説の平均 0 より大きい場合には, データ x の値がある値より大きくなると, 帰無仮説よりも対立仮説と整合的になり, 帰無仮説が棄却される.
- 対立仮説の平均 μ_1 が帰無仮説の平均 0 より小さい場合には, データ x の値がある値より小さくなると, 帰無仮説よりも対立仮説と整合的になり, 帰無仮説が棄却される.

この例より, すべての μ_1 について, 最強力になるような単一の検定法が存在しないことも分かる.

注意: しかし, **不偏** という非常に強い条件を満たす検定法の中ではすべての対立仮説パラメータ値 θ_1 について最強力になるような検定が存在することがある(一様最強力不偏検定(uniformly most powerful unbiased test, UMPU test)). 20世紀の検定の理論はこのような方向で整備された. 詳しくは次の教科書を参照せよ:

- Erich L. Lehmann, Joseph P. Romano, Testing Statistical Hypotheses, Third edition, 2005

しかし, 一様最強力不偏検定(UMPU test)の理論は実践的にはそう役に立つわけではない.

特に, 離散分布モデルを用いてUMPU検定を作るためには特定のデータによって帰無仮説が確率的に棄却される場合が出て来ることを受け入れる必要があり, 通常の科学研究でそれは受け入れらる条件ではない. 離散分布モデルによる実践的に使用されている検定法はどれもUMPU検定ではない.

とは言うものの, たとえ実践的ではないとしても, UMPU検定の話も数学的には結構面白い. 興味がある人は, UMPU検定について自分で調べ, 二項分布モデルなどの場合についてUMPU検定をコンピュータで実装してみるとよいだろう.

9.5 Neyman-Pearsonの仮説検定に関する極端な解釈の普及の問題

Neyman-Pearsonの仮説検定では, 帰無仮説についての第一種の過誤の確率を有意水準 α の設定によって上からおさえ, 対立仮説の検出力を可能な限り大きくすることによって, 帰無仮説と対立仮説をデータの数値によって比較する方法である.

Neyman-Pearsonの仮説検定では, 帰無仮説が棄却された場合には「帰無仮説よりも対立仮説の方が **相対的に** 妥当である」と **暫定的に** 判断してもよいだろう. その判断は単に帰無仮説と対立仮説の相対的な比較の下での暫定的な判断に過ぎず, 対立仮説の側が正しいと最終的に判断することとは異なる.

しかし, 「帰無仮説が棄却されなかった場合には帰無仮説を受容し, 帰無仮説が棄却された場合には対立仮説の側が正しいと判断する」ことは, Neyman-Pearsonの仮説検定の本質的要請であるとみなす極端な解釈が広く普及してしまっている.

その解釈が極端過ぎることは, 少なくとも,

- E. S. Pearson, Statistical Concepts in the Relation to Reality, Journal of the Royal Statistical Society. Series B (Methodological), Vol. 17, No. 2 (1955), pp. 204-207. [link (https://www.jstor.org/stable/2983954)], [pdf (https://errorstatistics.files.wordpress.com/2021/02/pearson_1955-stat-concepts-reality.pdf)]

のp.206からの以下の段落の引用を読めば明らかだと思われる:

> Professor Fisher's next objection is to the use of such terms as the "acceptance" or "rejection" of a statistical hypothesis, and "errors of the first and second kinds". It may be readily agreed that in the first Neyman and Pearson paper of 1928, more space might have been given to discussing how the scientific worker's attitude of mind could be related to the formal structure of the mathematical probability theory that was introduced. Nevertheless it should be clear from the first paragraph of this paper that we were not speaking of the final acceptance or rejection of a scientific hypothesis on the basis of statistical analysis. We speak of accepting or rejecting a hypothesis with a "greater or less degree of confidence". Further, we were very far from suggesting that statistical methods should force an irreversible acceptance procedure upon the experimenter. Indeed, from the start we shared Professor Fisher's view that in scientific enquiry, a statistical test is "a means of learning", for we remark: "the tests themselves give no final verdict, but as tools help the worker who is using them to form his final decision". No doubt we could more aptly have said "his final or provisional decision"; even scientists, if they are employed in research departments by industry or government, may sometimes have to give a final decision.

翻訳:

> Fisher教授の次の反対は, 統計的仮説の「受容」または「棄却」, 「第一種および第二種の過誤」といった用語の使用に対するものである. NeymanとPearsonによる1928年の最初の論文では, 科学的な研究者の心構えが導入された数学的確率論の形式的構造にどのように関係付けられるか, について論じるためにより多くのスペースが設けることができたということについてはただちに同意できる. しかし, この論文の最初の段落から明らかなように, 我々は科学的仮説の統計的分析に基く最終的受容や最終的棄却について語ってはいない. 我々は, 「信頼性の度合いの大小」によって仮説を受容するか棄却するかについて語っている. さらに, 我々は, 統計的手法が実験者に不可逆的な受容手順を強いることの示唆をまったくしていない. 実際我々は, 科学的探求において統計的検定は「学習の手段」であるというFisher教授の考えを最初から共有していた. その証拠に, 我々は「検定そのものは最終的判断を与えないが, 最終的決定のための道具として研究者にとって役に立つ.」と注意している. 間違いなく我々はより適切に「研究者の最終的または暫定的な判断」と言えた. たとえ産業界や政府の研究部門に雇用されていて最終的決定を下さなければいけない科学者達であってもそうである.

このように, Neyman-PearsonのPearson氏は「Neuman-Pearsonの仮説検定では, 検定の手続きによって仮説の受容と棄却に関する最終的判断を強いる」という解釈を明瞭に否定している.

検定の手続きは, 科学的探求における「学習の手段」であり, 最終的判断ではなく暫定的判断を与え, 道具としては最終的決定にも役に立つと述べている. さらに, 帰無仮説または対立仮説の絶対的な受容と棄却ではなく, 信頼度の度合いの大小による相対的な判断に過ぎないことも説明している.

おそらく, 「受容」(acceptance)と「棄却」(rejection)という1か0かの極端な判断を強いるニュアンスを与える用語を専門用語として採用してしまったことが悪かったのだろう. この「受容」「棄却」は現代の教科書でも使われている場合がある. Pearson氏にならって極端な意味に解釈しないように注意することが必要である.

なお, このノート群では「受容する」という言い方は避けて「棄却しない」という言い方を用いることにしている. (通常の意味で)「棄却しない」ことはただちに(通常の意味で)「受容する」ことを意味せず, 「判断を保留する」ことが可能なことも意味している.

さらに次の解説も参照せよ:

- Deborah Mayo, Erich Lehmann's 100 Birthday: Neyman Pearson vs Fisher on P-values, Posted on November 19, 2017. [link (https://errorstatistics.com/2017/11/19/erich-lehmanns-100-birthday-neyman-pearson-vs-fisher-on-p-values/)]

そこで引用されている次の論文では, Fisher流の検定とNeyman-Pearson流の検定の良い点を組み合わせて使うことを提案している:

- E. L. Lehmann, The Fisher, Neyman-Pearson Theories of Testing Hypotheses: One Theory or Two?, ournal of the American Statistical Association, Vol. 88, No. 424 (Dec., 1993), pp. 1242-1249. [link (https://www.jstor.org/stable/2291263)], [pdf (https://errorstatistics.files.wordpress.com/2013/11/lehmann_1-theory-or-2.pdf)]

非常に残念なことに, 極端な解釈によって, Fisher流の検定とNeyman-Pearson流の検定の違いを強調し, 対立を煽っているかのような解説を見ることがある. そのような解説では, Fisher流の検定とNeyman-Pearson流の検定の考え方を混ぜることは許さないという方針になりがちである. (これは上で引用したLehmann氏の論文とは正反対の態度である.)

そのようなことをしても科学的に建設的ではなく, 現代における最良の視点から, 科学的常識に沿った穏健で有用な解釈を普及させる方がよいと思われる.

補足: 上で引用した論文の著者のLehmann氏(Erich Leo Lehmann (1917-2009) (https://en.wikipedia.org/wiki/Erich_Leo_Lehmann) はNeyman氏の弟子の有名な統計学者である.

補足: このノートでは「検定の手続きが1か0かの最終的判断を強いる」という極端な解釈を「統計学は科学的お墨付きを得るための手段ではない」という言い方でも明瞭に否定している.

補足: このノートで採用したP値を「仮説 $\theta = \theta_0$ 下の統計モデルとデータの数値 x の整合性の指標」とみなす考え方はFisherの有意性検定の考え方に近いように思われる. そして, それと同時に, Neyman-Pearson的に対立仮説の検出力も重要であるという立場も採用している(その縛り抜きには無意味な検定法を排除できない).

補足: P値に関する「第一種の過誤の確率が有意水準 α に近い方がよい」という基準は「検出力が高い方がよい」という基準と密接に関係している. 実際, 保守的過ぎる検定法は検出力が低くなってしまいがちである.

補足: この一連のノート群では, 数学的な道具の使い方はユーザーが自分の目的に合わせて自由に決めればよいという立場も採用している. ただし, その使い方は普通の常識的な意味で科学的に合理的なければいけない.

9.6 このノートの立場

検定の手続きは, 例えば, 「薬Aに効果はない」というような仮説をモデルのパラメータに関する仮説 $\theta = 0$ で代替し, 「薬Aに効果はない」という仮説は疑わしい(もしくは誤りである)という結論を出したい状況で使われていること多い.

そのとき「薬Aに効果はない」のような否定したい特別な仮説が帰無仮説と呼ばれることになり, 仮説 $\theta = 0$ の 0 という値は特別な意味を持つものとして固定されていることが多い.

しかし, **このノートでは, 検定にかけられる仮説 $\theta = \theta_0$ を定めるパラメータ値 θ_0 は1つに固定されておらず, 任意の値を取り得ることを強調し, 可能なすべての θ_0 に関する検定の手続きによって信頼区間を定義している.**

すなわち, このノートにおける信頼度 $1 - \alpha$ 信頼区間の定義は, 仮説 $\theta = \theta_0$ がデータの数値 x による有意水準 α の検定の手続きで棄却されないような数値 θ_0 全体の集合だとしている.

このスタイルの利点は, **信頼区間は, 「薬Aに効果はない」という仮説だけではなく, 任意の値 θ_0 に関する「薬Aの効果は θ_0 である」という仮説の検定結果の情報もすべて含んでいる** ことが最初から明らかになることである.

信頼区間を見れば以下のような情報が得られる:

- a より小さな値を信頼区間は含まないので, 薬Aの効果は概ね a 以上だと考えてよさそうだ.
- 信頼区間は 0 を含んでいるので, データの数値 x からは薬Aの効果の有無はわからなかった. (注意:「効果はない」と判断するのは誤り.)
- a より大きな値を信頼区間は含まないので, 薬Aの効果は概ね a 以下だと考えてよさそうだ.

このようなことが可能になったのは, 検定にかける仮説 $\theta = \theta_0$ の θ_0 を自由に動かせる設定を採用したからである.

信頼区間は有意水準 α という閾値を設定することによって定義される.

しかし, $\alpha = 5\%$ のような慣習化された有意水準をP値が下回るか否かに一喜一憂するというような愚行に警戒する必要が生じてしまった.

この人為的な閾値の設定に関わる問題について, このノートではP値函数について紹介することによって対処したつもりである.

データの数値 x が与えられたとき, パラメータ値 θ_0 に対して, 仮説 $\theta = \theta_0$ のP値を対応させる函数をP値函数と呼ぶのであった.

信頼度 $1-\alpha$ の信頼区間の定義は, P値函数の値が有意水準以上になるパラメータ値 θ_0 全体の集合なので, P値函数のグラフを描けば視覚的に信頼区間がどうなるかも確認できる.

すなわち, P値函数のグラフはすべての信頼度 $1-\alpha$ に関する信頼区間の情報を持っている.

ゆえに, P値函数のグラフを描けば, 有意水準 α を固定せずに, 統計モデルのパラメータ値とデータの数値の整合性(P値が小さいほど整合性が低い)を確認できる.

P値函数のグラフを利用すれば, 有意水準を設定しなくても, そこから科学的に有益な情報を引き出すことが可能である.

この考え方はRothman氏達の有名な疫学の教科書 Modern Epidemiology でも紹介されている.

P値函数のグラフは, ある特定の信頼度(例えば 95%) に関する信頼区間よりも, 圧倒的に豊富な情報を持っている. まず, 信頼度を自由に変えられる. さらに, P値函数が最大になるパラメータ値は, 統計モデルがデータの数値に最もフィットするようなパラメータ値だと解釈される. 実際, それは多くの場合に最尤推定値またはそれに近い値になっている.

P値函数がどういうものだったか忘れた人のために以下に二項分布モデルのP値函数のグラフを以下に掲載しておこう. P値函数のグラフは多くの場合に以下の図のように「とんがり帽子」のような形になる. そしてその「とんがり帽子」の幅が概ね信頼区間の幅に対応している.

In [36]:
```
plot_binom_pvaluefunctions(; n = 50, k = 15)
```

Out[36]:

P値函数のグラフの例とR言語による描き方に関する解説が以下の場所にある:

- Denis Infanger, P-value functions: Tutorial using the pvaluefunctions package, 2021-11-30. [link (https://cran.r-project.org/web/packages/pvaluefunctions/vignettes/pvaluefun.html)]
- Plot a P-value Function from one or two Confidence Intervals. [link (https://epijim.shinyapps.io/episheet_shiny/)]

以下のリンク先も面白い.

- https://twitter.com/ken_rothman/status/1258552231286063106 (https://twitter.com/ken_rothman/status/1258552231286063106)
- https://twitter.com/MinatoNakazawa/status/656235447220224000 (https://twitter.com/MinatoNakazawa/status/656235447220224000)
- https://twitter.com/MinatoNakazawa/status/1202738229180416000 (https://twitter.com/MinatoNakazawa/status/1202738229180416000)
- https://twitter.com/MinatoNakazawa/status/1202893936181645312 (https://twitter.com/MinatoNakazawa/status/1202893936181645312)

今後の目標: データ x とパラメータ値 $\theta = \theta_0$ に対してP値を対応させる函数 $\mathrm{pvalue}(x|\theta = \theta_0)$ の例を構成して利用すること.

10 よくある誤解

10.1 P値の定義と使い方の復習

現実世界から得るデータ x の生成のされ方に関するパラメータ θ を持つ統計モデルが与えられているとき, データの数値 x に関する仮説 $\theta = \theta_0$ のP値は

- 仮説 $\theta = \theta_0$ 下の統計モデル内部でデータの数値 x 以上に極端な値が生成される確率もしくはその近似値

と定義されるのであった. (「数値 x 以上に極端な値」の意味は別に定義しなければいけない.)

そして, そのようにして計算されるP値は, **仮説 $\theta = \theta_0$ 下の統計モデルとデータの数値の整合性の指標** として使われ, 適当な閾値(有意水準)を設けて, P値が小さすぎる場合には「整合性がない」と判定するのであった(これが検定の手続き).

10.2 問題: P値に関するよくある誤解

データの数値 x に関する仮説 $\theta = \theta_0$ のP値を以下では単にP値と呼ぶことにする.

以下の主張の中で誤解であるものをすべて挙げよ.

(1) P値は現実世界において仮説 $\theta = \theta_0$ が正しい確率である.

(2) 検定の手続きでは, P値が有意水準より小さいときには, 仮説 $\theta = \theta_0$ は科学的に疑わしいと考える.

(3) 検定の手続きでは, P値が有意水準以上のときには, 仮説 $\theta = \theta_0$ は科学的に肯定されたと考える.

(4) P値が小さい結果ほど重要な結果である.

正しい主張を誤解だとみなさないように注意して欲しい.

この問題の解答例はこのノートの最後の方に掲載する.

10.3 信頼区間の定義の復習と使い方の復習

データの数値 x が与えられたとき, 有意水準 α の検定の手続きによって棄却されないパラメータ値 $\theta = \theta_0$ 全体の集合を, パラメータ θ に関する信頼度 $1 - \alpha$ の信頼区間と呼ぶのであった. $\alpha = 5\%$ のときには 95% 信頼区間と呼ばれる.

10.4 問題: 信頼区間に関するよく見る誤解

以下の主張が誤りもしくは杜撰である理由を説明せよ.

(1) 95% 信頼区間の 95% は確率ではなく, 割合である.

(2) 平均値の 95% 信頼区間について考える. このとき, 現実の母集団からの無作為抽出を繰り返して, 平均値の 95% 信頼区間を計算し直すとき, 現実の母集団の平均値を含む区間達の割合は 95% になる.

この問題の解答例はこのノートの最後に掲載する.

10.5 平均の信頼区間達の視覚化

後で別のノートで解説する予定の正規分布の標本分布モデルと t 分布を用いて計算される平均の信頼区間を正規分布の標本達および正規分布以外の分布の標本達についてプロットしてみよう.

In [37]:
```
Random.seed!(4649373)
```

Out[37]:
```
TaskLocalRNG()
```

10.5.1 正規分布の標本達から得られる平均の信頼区間達

標本(データ)を生成する分布が正規分布の場合は, 信頼区間を計算するために用いる統計モデルで標本の生成のされ方がぴったり記述可能な場合になっている.

これは実践的な統計分析においては非現実的な想定である.

この場合には理論通りの結果がそのまま成立している.

In [38]:
```
dist = Normal(2, 3)
@show μ, σ = mean(dist), std(dist)
plot(dist, μ-4σ, μ+4σ; label="", title="$(distname(dist))",
    xtick=-100:2:100)
```

```
(μ, σ) = (mean(dist), std(dist)) = (2.0, 3.0)
```

Out[38]:

以下のセルでは正規分布 $\mathrm{Normal}(2,3)$ のサイズ $n=30$ の標本を200個ランダムに生成し, その各々について平均の 95% 信頼区間を計算してプロットしている.

橙色でプロットされている 95% 信頼区間は標本を生成した分布の平均値を含まない信頼区間である.

標本を生成した分布の平均値を含まない 95% 信頼区間の割合は 5% であって欲しい.

In [39]:
```
illustrate_confintmean(; dist, n = 30)
```

Out[39]:

In [40]:
```
sim_ttest(; dist, n = 30, L = 10^6)(0.05)
```

Out[40]:
```
0.049977
```

以下のセルでは正規分布 $\mathrm{Normal}(2,3)$ のサイズ $n=30$ の標本を100万個ランダムに生成し, 信頼度 $1-\alpha$ 信頼区間に標本を生成した分布の平均値が含まれない確率(=割合)を計算している.

その割合は α であって欲しい.

この場合には実際にぴったり α になっている.

In [41]:
```
plot_sim_ttest(; dist, n = 30, L = 10^6)
```

Out[41]:

10.5.2 ガンマ分布の標本から得られる平均の信頼区間達

標本(データ)を生成する分布が左右非対称なガンマ分布 $\mathrm{Gamma}(3,4)$ の場合には, 平均の信頼区間を構成するために使った正規分布とは異なる種類の分布で標本を生成しているので, 誤差が生じることになる.

その誤差がどの程度であるかに注目して以下のグラフ達を眺めて欲しい.

In [42]:
```
dist = Gamma(3, 4)
@show μ, σ = mean(dist), std(dist)
plot(dist, -1, 50; label="", title="$(distname(dist))")
```

```
(μ, σ) = (mean(dist), std(dist)) = (12.0, 6.928203230275509)
```

Out[42]:

以下のセルではガンマ分布 $\mathrm{Gamma}(3,4)$ のサイズ $n=30$ の標本を200個ランダムに生成し, その各々について平均の 95% 信頼区間を計算してプロットしている.

橙色でプロットされている 95% 信頼区間は標本を生成した分布の平均値を含まない信頼区間である.

標本を生成した分布の平均値を含まない 95% 信頼区間の割合は 5% であって欲しいが, この場合にはその割合は 5.8% 程度で 5% より少し大きい.

In [43]:
```
illustrate_confintmean(; dist, n = 30)
```

Out[43]:

In [44]:
```
sim_ttest(; dist, n = 30, L = 10^6)(0.05)
```

Out[44]: 0.058224

以下のセルではガンマ分布 $\mathrm{Gamma}(3,4)$ のサイズ $n=30$ の標本を100万個ランダムに生成し, 信頼度 $1-\alpha$ 信頼区間に標本を生成した分布の平均値が含まれない確率(=割合)を計算している.

その割合は α であって欲しい.

この場合にその割合は α より少し大きな値になってしまっている.

In [45]:
```
plot_sim_ttest(; dist, n = 30, L = 10^6)
```

Out[45]:

10.5.3 対数正規分布の標本達から得られる平均の信頼区間達

標本(データ)を生成する分布が左右非対称でかつ非常に大きな値が生成され易い対数正規分布 $\mathrm{LogNormal}(0,1)$ の場合には, 非常に大きな誤差が生じることになる.

In [46]:
```
dist = LogNormal()
@show μ, σ = mean(dist), std(dist)
plot(dist, -0.2, 10.2; label="", title="$(distname(dist))")
```

```
(μ, σ) = (mean(dist), std(dist)) = (1.6487212707001282, 2.1611974158950877)
```

Out[46]:

以下のセルでは対数正規分布 $\mathrm{LogNormal}(0,1)$ のサイズ $n=30$ の標本を200個ランダムに生成し, その各々について平均の 95% 信頼区間を計算してプロットしている.

橙色でプロットされている 95% 信頼区間は標本を生成した分布の平均値を含まない信頼区間である.

標本を生成した分布の平均値を含まない 95% 信頼区間の割合は 5% であって欲しいが, この場合にはその割合は 12% 弱程度で 5% より非常に大きい.

正規分布の標本分布モデルと t 分布を使った平均に関する検定や信頼区間は小さな確率で極端な値が生じる分布(外れ値を持つ分布)の標本については誤差が大きくなり易いので注意を要する.

In [47]:
```
illustrate_confintmean(; dist, n = 30)
```

Out[47]:

In [48]:
```
sim_ttest(; dist, n = 30, L = 10^6)(0.05)
```

Out[48]: 0.11736

以下のセルでは対数正規分布 $\mathrm{LogNormal}(0,1)$ のサイズ $n=30$ の標本を100万個ランダムに生成し, 信頼度 $1-\alpha$ 信頼区間に標本を生成した分布の平均値が含まれない確率(＝割合)を計算している.

その割合は α であって欲しい.

この場合にその割合は α よりもずっと大きな値になってしまっている.

In [49]:
```
plot_sim_ttest(; dist, n = 30, L = 10^6)
```

Out[49]:

サンプルサイズ n を大きくすると誤差は小さくなる.

In [50]:
```
plot_sim_ttest(; dist, n = 100, L = 10^6)
```

Out[50]:

In [51]:
```
plot_sim_ttest(; dist, n = 1000, L = 10^6)
```

Out[51]:

10.6 問題解答例: P値に関するよくある誤解

(1) P値は現実世界において仮説 $\theta = \theta_0$ が正しい確率である.

(2) 検定の手続きでは, P値が有意水準より小さいときには, 仮説 $\theta = \theta_0$ は科学的に疑わしいと考える.

(3) 検定の手続きでは, P値が有意水準以上のときには, 仮説 $\theta = \theta_0$ は科学的に肯定されたと考える.

(4) P値が小さい結果ほど重要な結果である.

はすべて誤りである. 各々についてコメントしておこう.

(1) P値の定義と全然違う. P値の定義は「仮説 $\theta = \theta_0$ 下の統計モデル内でデータの数値以上に極端な値が生じる確率またはその近似値」であった.

P値は「データの数値と仮説 $\theta = \theta_0$ 下の統計モデルの整合性の指標」として使われる. 統計モデル全体とデータの整合性を見ているのであり, 仮説 $\theta = \theta_0$ 単体との整合性を見ているのではない.

(2) 検定の手続きで, P値が有意水準より小さいときには, 仮説 $\theta = \theta_0$ だけが疑わしいと考えるのではなく, 統計モデルの前提の全体のどれかも疑わしいと考える必要がある.

(3) 検定の手続きで, P値が有意水準以上になった場合には, 仮説 $\theta = \theta_0$ 下での統計モデルの現実における妥当性に関する判断を保留する(否定も肯定もしない).

(4) 例えば, ある薬の効果について, 巨大なサイズのデータによって非常に小さなP値が得られたとする.

しかし, 存在することが確からしいその効果は実生活においてほとんど無視できるほど小さなものだったとする.

そのような場合にはP値が小さくても重要な結果が得られたとは言えない.

薬の効果について統計分析をする場合には, P値だけではなく, 効果の大きさの指標にも注意を払う必要がある.

解答終

10.7 問題解答例: 信頼区間に関するよく見る誤解

(1) 95% 信頼区間の 95% は確率ではなく, 割合である.

(2) 平均値の 95% 信頼区間について考える. このとき, 現実の母集団からの無作為抽出を繰り返して, 平均値の 95% 信頼区間を計算し直すとき, 現実の母集団の平均の真の値を含む区間達の割合は 95% になる.

はどちらも誤りもしくは杜撰な考え方である.

(1) 仮説 $\theta = \theta_0$ 下の統計モデル内部で生成された仮想的なデータの数値(確率変数になる)に関するパラメータ θ に関する 95% 信頼区間に θ_0 が含まれる確率は 95% またはそれに近い値になる.

このように 95% 信頼区間の 95% は統計モデル内部での確率になっていると解釈される. ゆえに「確率ではない」と言ってしまうと自明に誤りになる.

ただし, 確率的に揺らぐのはデータを生成するために使われたパラメータ値 $\theta = \theta_0$ ではなく, そのパラメータ値を使って生成されたデータの値とそれを使って計算される信頼区間である. 固定されている値 θ_0 が確率的に様々に変化する 95% 信頼区間に含まれる確率が 95% またはそれに近い値になる.

(2) 現実の母集団の未知の平均値を μ_{real} と書くことにする. もしも平均値の信頼区間を定義するために使った統計モデルで平均値パラメータを $\mu = \mu_{\mathrm{real}}$ とおくことによって得られる確率分布が現実の母集団の分布をよく近似していれば, 問題文の(2)の主張は確かは正しい.

しかし, 数学的フィクションである統計モデルが現実の母集団の分布をよく近似しているという仮定が成立しているは限らない. 成立していない場合には(2)の主張も成立しているとは限らない.

この問題中の(1),(2)の誤解の背景には現実とモデルの混同が隠れていると考えられる.

解答終

In [ ]:

```
1
```

検定と信頼区間: 比率の検定と信頼区間

- 黒木玄

- 2022-05-31～2022-06-14

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 比率の検定と信頼区間
  - 1.1 P値と信頼区間の定義の復習
  - 1.2 二項分布モデルのP値達の定義を1枚の図で要約
  - 1.3 4種のP値函数のプロット
  - 1.4 4種の信頼区間の同時プロット
- 2 Clopper-Pearson
  - 2.1 Clopper-Pearsonの信頼区間を与えるP値関数
  - 2.2 Clopper-Pearsonの信頼区間
  - 2.3 Clopper-Pearsonの信頼区間の図による把握
- 3 Sterne
  - 3.1 Sterneの信頼区間を与えるP値関数
  - 3.2 Sterneの信頼区間
  - 3.3 Sterneの信頼区間の図による把握
  - 3.4 おまけ: R言語のbinom.testについて
  - 3.5 おまけ関連問題: R言語のコードを実行してみよ.
- 4 Wilson
  - 4.1 Wilsonの信頼区間を与えるP値函数
  - 4.2 Wilsonの信頼区間
  - 4.3 Wilsonの信頼区間の図による把握
  - 4.4 おまけ: R言語のprop.testについて
  - 4.5 Pearsonのχ^2検定との関係
  - 4.6 おまけ: 多項分布モデルにおけるPearsonのχ^2検定
  - 4.7 おまけ(より進んだ話題): 一般のχ^2検定で使うχ^2分布の自由度
- 5 Wald
  - 5.1 Waldの信頼区間を与えるP値関数
  - 5.2 Waldの信頼区間
  - 5.3 Waldの信頼区間の図による把握
  - 5.4 必修問題: 大阪都構想に関する住民投票の結果から得られる信頼区間達
    - 5.4.1 Julia言語による計算の仕方
    - 5.4.2 WolframAlphaによる計算の仕方

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Printf
using QuadGK
using RCall
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=fal
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : x*y
safediv(x, y) = x == 0 ? x : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) = standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

Clopper-Pearsonの節の準備

In [4]:
```julia
# cdf = 累積分布函数
# ccdf(dist, x) = 1 - cdf(dist, x)

function pvalue_clopper_pearson(dist::DiscreteUnivariateDistribution, x)
    min(1, 2cdf(dist, x), 2ccdf(dist, x-1))
end
pvalue_clopper_pearson(n, k, p) = pvalue_clopper_pearson(Binomial(n, p), k)

function confint_clopper_pearson(n, k; α = 0.05)
    p_L = k > 0 ? quantile(Beta(k, n-k+1), α/2) : zero(α)
    p_U = k < n ? quantile(Beta(k+1, n-k), 1-α/2) : one(α)
    [p_L, p_U]
end
```

Out[4]:
```
confint_clopper_pearson (generic function with 1 method)
```

In [5]:
```julia
n, k, α = 100, 30, 0.05
kmin, kmax = 0, 65
p_L, p_U = confint_clopper_pearson(n, k; α)
@show [p_L, p_U];
```

```
[p_L, p_U] = [0.21240642048953662, 0.39981467617980404]
```

In [6]:
```julia
dist_L, dist_U = Binomial(n, p_L), Binomial(n, p_U)
```

Out[6]:
```
(Binomial{Float64}(n=100, p=0.21240642048953662), Binomial{Float64}(n=100, p=0.3998146761798040
4))
```

In [7]:
```julia
var"Clopper-Pearson: 下側では p = p_L がぎりぎり" = plot(; size=(500, 300))
plot!(k -> mypdf(dist_L, k), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(k -> mypdf(dist_L, k), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(100α/2)%", :blue, :left, 10))
title!("Binomial(n=$n, p=p_L),  p_L=$(round(p_L; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
plot!(; ylim=(-0.007, 0.105));
```

In [8]:
```julia
var"Clopper-Pearson: 上側では p = p_U がぎりぎり" = plot(; size=(500, 300))
plot!(k -> mypdf(dist_U, k), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(k -> mypdf(dist_U, k), kmin-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
annotate!(k-2, 0.01, text("$(100α/2)%", :red, :right, 10))
title!("Binomial(n=$n, p=p_U),  p_U=$(round(p_U; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_U]; label="n p_L = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!(; ylim=(-0.007, 0.105));
```

In [9]:
```julia
var"Clopper-Pearson: n × 95%信頼区間" = plot(; size=(500, 300))

plot!(k -> mypdf(dist_L, k), kmin-0.5, 47+0.5;
    label="", c=1)
plot!(k -> mypdf(dist_L, k), k-0.5, 47+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(100α/2)%", :blue, :left, 10))

plot!(k -> mypdf(dist_U, k), 15-0.5, kmax+0.5;
    label="", c=2)
plot!(k -> mypdf(dist_U, k), 15-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
annotate!(k-2, 0.01, text("$(100α/2)%", :red, :right, 10))

vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
vline!([n*p_U]; label="n p_U = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!([n*p_L, n*p_U], fill(-0.003, 2); label="", c=:red, lw=5)
title!("n = $n,  [n p_L, n p_U] = [$(round(n*p_L; digits=2)), $(round(n*p_U; digits=2))]")
plot!(; ylim=(-0.007, 0.105));
```

Sterneの節の準備

In [10]:

```julia
_pdf_le(x, (dist, y)) =  pdf(dist, x) ≲ y

function _search_boundary(f, x0, Δx, param)
    x = x0
    if f(x, param)
        while f(x - Δx, param) x -= Δx end
    else
        x += Δx
        while !f(x, param) x += Δx end
    end
    x
end

function pvalue_sterne(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    Px == 0 && return Px
    Px == 1 && return Px
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = _search_boundary(_pdf_le, 2m - x, 1, (dist, Px))
        cdf(dist, x) + ccdf(dist, y-1)
    else # x > m
        y = _search_boundary(_pdf_le, 2m - x, -1, (dist, Px))
        cdf(dist, y) + ccdf(dist, x-1)
    end
end

function pvalue_sterne(n, k, p)
    pvalue_sterne(Binomial(n, p), k)
end

# 大きな n についてもうまく行くように
# Sterneの信頼区間の実装は難しい.
function confint_sterne(n, k; α = 0.05)
    a, b = confint_clopper_pearson(n, k; α = α/10)
    ps = find_zeros(a-√eps(), b+√eps()) do p
        logit(0 < p ≤ 1 ? pvalue_sterne(n, k, p) : zero(p)) - logit(α)
    end
    # 次の行は稀に区間にならない場合への対策
    [first(ps), last(ps)]
end
```

Out[10]: confint_sterne (generic function with 1 method)

In [11]:

```julia
n, k, α = 100, 30, 0.05
kmin, kmax = 0, 65
p_L, p_U = confint_sterne(n, k; α)
@show [p_L, p_U];
```

[p_L, p_U] = [0.21419730401454778, 0.3996107372232876]

In [12]:

```julia
dist_L = Binomial(n, p_L)
k_L = minimum(i for i in 0:n if !(pdf(dist_L, i) ≲ pdf(dist_L, k))) - 1
cdf_L = cdf(dist_L, k_L)
ccdf_L = ccdf(dist_L, k-1)
cdf_L + ccdf_L
```

Out[12]: 0.04999999999999993

In [13]:

```julia
dist_U = Binomial(n, p_U)
k_U = maximum(i for i in 0:n if !(pdf(dist_U, i) ≲ pdf(dist_U, k))) + 1
cdf_U = cdf(dist_U, k)
ccdf_U = ccdf(dist_U, k_U-1)
cdf_U + ccdf_U
```

Out[13]: 0.05184087343317326

In [14]:

```
var"Sterne: 下側では p = p_L がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_L, i), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(i -> mypdf(dist_L, i), k-0.5, kmax+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
plot!(i -> mypdf(dist_L, i), kmin-0.5, k_L+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
plot!([k_L+0.5, k-0.5], fill(pdf(dist_L, k), 2); label="", c=:blue, ls=:dot)
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))
title!("Binomial(n=$n, p=p_L),  p_L=$(round(p_L; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
plot!(; ylim=(-0.007, 0.105));
```

In [15]:

```
var"Sterne: 上側では p = p_U がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_U, i), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(i -> mypdf(dist_U, i), kmin-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
plot!(i -> mypdf(dist_U, i), k_U-0.5, kmax+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
plot!([k+0.5, k_U-0.5], fill(pdf(dist_U, k), 2); label="", c=:red, ls=:dot)
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))
title!("Binomial(n=$n, p=p_U),  p_U=$(round(p_U; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_U]; label="n p_L = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!(; ylim=(-0.007, 0.105));
```

In [16]:

```
var"Sterne: n × 95%信頼区間" = plot(; size=(500, 300))

plot!(k -> mypdf(dist_L, k), kmin-0.5, 47+0.5;
    label="", c=1)
plot!(k -> mypdf(dist_L, k), k-0.5, 47+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
plot!(i -> mypdf(dist_L, i), kmin-0.5, k_L+0.5;
    fillrange=0, c=1, fc=:blue, fa=0.5, label="")
#plot!([k_L+0.5, k-0.5], fill(pdf(dist_L, k), 2); label="", c=:blue, ls=:dot)
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))

plot!(i -> mypdf(dist_U, i), 15-0.5, kmax+0.5;
    label="", c=2)
plot!(i -> mypdf(dist_U, i), 15-0.5, k+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
plot!(i -> mypdf(dist_U, i), k_U-0.5, kmax+0.5;
    fillrange=0, c=2, fc=:red, fa=0.5, label="")
#plot!([k+0.5, k_U-0.5], fill(pdf(dist_U, k), 2); label="", c=:red, ls=:dot)
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))

vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
vline!([n*p_U]; label="n p_U = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!([n*p_L, n*p_U], fill(-0.003, 2); label="", c=:red, lw=5)
title!("n = $n,  [n p_L, n p_U] = [$(round(n*p_L; digits=2)), $(round(n*p_U; digits=2))]")
plot!(; ylim=(-0.007, 0.105));
```

Wilsonの節の準備

In [17]:
```
function pvalue_wilson(n, k, p)
    p̂ = k/n
    SE = √(p*(1-p)/n)
    2ccdf(Normal(), abs(p̂ - p)/SE)
end

function confint_wilson(n, k; α = 0.05)
    p̂ = k/n
    z = quantile(Normal(), 1-α/2)
    a, b, c = 1+z^2/n, p̂+z^2/(2n), p̂^2
    # ap² - 2bp + c = 0 を解く.
    sqrtD = √(b^2 - a*c)
    p_L = (b - sqrtD)/a
    p_U = (b + sqrtD)/a
    [p_L, p_U]
end
```

Out[17]: confint_wilson (generic function with 1 method)

In [18]:
```
α = 0.05
z = quantile(Normal(), 1 - α/2)

var"z_{0.025}の定義" = plot()
plot!(Normal(), -5,  5; label="")
plot!(Normal(), -5, -z; label="", c=1, fillrange=0, fc=:red, fa=0.3)
plot!(Normal(),  z,  5; label="", c=1, fillrange=0, fc=:red, fa=0.3)
plot!([-z, -z, NaN, z, z],
    [0, pdf(Normal(), -z), NaN, 0, pdf(Normal(), z)];
    label="\$\\pm z_{0.025}\$", c=:red)
title!("Normal(0,1)"; xtick=-10:10)
annotate!(-2.2, 0.06, text("2.5%", 10, :red, :right))
annotate!( 2.2, 0.06, text("2.5%", 10, :red, :left))
plot!(; legendfontsize=14);
```

In [19]:
```
n, k, α = 100, 30, 0.05
kmin, kmax = 0, 65
p_L, p_U = confint_wilson(n, k; α)
@show [p_L, p_U];
```

```
[p_L, p_U] = [0.2189488529493274, 0.395848546333467]
```

In [20]:
```
dist_L, dist_U = Binomial(n, p_L), Binomial(n, p_U)
normal_L = Normal(n*p_L, √(n*p_L*(1-p_L)))
normal_U = Normal(n*p_U, √(n*p_U*(1-p_U)))
k_L, k_U = 2n*p_L - k, 2n*p_U - k
cdf_L, ccdf_L = cdf(normal_L, k_L), ccdf(normal_L, k)
cdf_U, ccdf_U = cdf(normal_U, k), ccdf(normal_U, k_U);
```

In [21]:
```
var"Wilson: 下側では p = p_L がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_L, i), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(normal_L; label="normal approx", c=:blue)
plot!(x -> pdf(normal_L, x), k, kmax+0.5;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_L, x), kmin-0.5, k_L;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))
title!("Binomial(n=$n, p=p_L),  p_L=$(round(p_L; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
plot!(; ylim=(-0.007, 0.105));
```

In [22]:
```
var"Wilson: 上側では p = p_U がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_U, i), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(normal_U; label="normal approx", c=:red)
plot!(x -> pdf(normal_U, x), k_U, kmax+0.5;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_U, x), kmin-0.5, k;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="")
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))
title!("Binomial(n=$n, p=p_U),  p_U=$(round(p_U; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_U]; label="n p_L = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!(; ylim=(-0.007, 0.105));
```

In [23]:
```
var"Wilson: n × 95%信頼区間" = plot(; size=(500, 300))

plot!(i -> mypdf(dist_L, i), kmin-0.5, 47+0.5;
    label="", c=1)
plot!(normal_L; label="normal approx", c=:blue)
plot!(x -> pdf(normal_L, x), k, 47+0.5;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_L, x), kmin-0.5, k_L;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))

plot!(i -> mypdf(dist_U, i), 15-0.5, kmax+0.5;
    label="", c=2)
plot!(normal_U; label="normal approx", c=:red)
plot!(x -> pdf(normal_U, x), k_U, kmax+0.5;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_U, x), 15-0.5, k;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="")
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))

vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
vline!([n*p_U]; label="n p_U = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!([n*p_L, n*p_U], fill(-0.003, 2); label="", c=:red, lw=5)
title!("n = $n,  [n p_L, n p_U] = [$(round(n*p_L; digits=2)), $(round(n*p_U; digits=2))]")
plot!(; ylim=(-0.007, 0.105));
```

Waldの節の準備

In [24]:
```
function pvalue_wald(n, k, p)
    p̂ = k/n
    SEhat = √(p̂*(1-p̂)/n)
    2ccdf(Normal(), abs(p̂ - p)/SEhat)
end

function confint_wald(n, k; α = 0.05)
    p̂ = k/n
    SEhat = √(p̂*(1-p̂)/n)
    z = quantile(Normal(), 1-α/2)
    p_L = p̂ - z*SEhat
    p_U = p̂ + z*SEhat
    [p_L, p_U]
end
```

Out[24]: confint_wald (generic function with 1 method)

In [25]:
```
n, k, α = 100, 30, 0.05
kmin, kmax = 0, 65
p_L, p_U = confint_wald(n, k; α)
@show [p_L, p_U];
```

```
[p_L, p_U] = [0.21018316681457927, 0.3898168331854207]
```

In [26]:

```
dist_L, dist_U = Binomial(n, p_L), Binomial(n, p_U)
p̂ = k/n
normal_L = Normal(n*p_L, √(n*p̂*(1-p̂)))
normal_U = Normal(n*p_U, √(n*p̂*(1-p̂)))
k_L, k_U = 2n*p_L - k, 2n*p_U - k
cdf_L, ccdf_L = cdf(normal_L, k_L), ccdf(normal_L, k)
cdf_U, ccdf_U = cdf(normal_U, k), ccdf(normal_U, k_U);
```

In [27]:

```
var"Wald: 下側では p = p_L がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_L, i), kmin-0.5, kmax+0.5;
    label="", c=1)
plot!(normal_L; label="normal approx", c=:blue)
plot!(x -> pdf(normal_L, x), k, kmax+0.5;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_L, x), kmin-0.5, k_L;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))
title!("Binomial(n=$n, p=p_L),  p_L=$(round(p_L; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
plot!(; ylim=(-0.007, 0.105));
```

In [28]:

```
var"Wald: 上側では p = p_U がぎりぎり" = plot(; size=(500, 300))
plot!(i -> mypdf(dist_U, i), kmin-0.5, kmax+0.5;
    label="", c=2)
plot!(normal_U; label="normal approx", c=:red)
plot!(x -> pdf(normal_U, x), k_U, kmax+0.5;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_U, x), kmin-0.5, k;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="")
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))
title!("Binomial(n=$n, p=p_U),  p_U=$(round(p_U; digits=4))")
vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_U]; label="n p_L = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!(; ylim=(-0.007, 0.105));
```

In [29]:

```
var"Wald: n × 95%信頼区間" = plot(; size=(500, 300))

plot!(i -> mypdf(dist_L, i), kmin-0.5, 47+0.5;
    label="", c=1)
plot!(normal_L; label="normal approx", c=:blue)
plot!(x -> pdf(normal_L, x), k, 47+0.5;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_L, x), kmin-0.5, k_L;
    fillrange=0, c=:blue, fc=:blue, fa=0.5, label="")
annotate!(k+2, 0.01, text("$(round(100ccdf_L; digits=1))%", :blue, :left, 10))
annotate!(k_L-2, 0.01, text("$(round(100cdf_L; digits=1))%", :blue, :right, 10))

plot!(i -> mypdf(dist_U, i), 15-0.5, kmax+0.5;
    label="", c=2)
plot!(normal_U; label="normal approx", c=:red)
plot!(x -> pdf(normal_U, x), k_U, kmax+0.5;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="", ls=:dot)
plot!(x -> pdf(normal_U, x), 15-0.5, k;
    fillrange=0, c=:red, fc=:red, fa=0.5, label="")
annotate!(k_U+2, 0.01, text("$(round(100ccdf_U; digits=1))%", :red, :left, 10))
annotate!(k-2, 0.01, text("$(round(100cdf_U; digits=1))%", :red, :right, 10))

vline!([k]; label="k=$k", c=:black, ls=:dot)
vline!([n*p_L]; label="n p_L = $(round(n*p_L; digits=2))", c=:blue, ls=:dash)
vline!([n*p_U]; label="n p_U = $(round(n*p_U; digits=2))", c=:red, ls=:dashdot)
plot!([n*p_L, n*p_U], fill(-0.003, 2); label="", c=:red, lw=5)
title!("n = $n,  [n p_L, n p_U] = [$(round(n*p_L; digits=2)), $(round(n*p_U; digits=2))]")
plot!(; ylim=(-0.007, 0.105));
```

P値達の説明用

In [30]:

```julia
function illustrate_binomial_pvalues()
    n = 20
    p = 0.25
    k = 9

    p̂ = k/n
    σ̂² = n*p̂*(1-p̂)

    null = Binomial(n, p)
    μ, σ² = mean(null), var(null)
    nullname = distname(null)

    f(x) = mypdf(null, x)
    xlim = (-2, n+1)
    xlimk = (k-0.5, n+0.5)

    pval = @sprintf "%.03f" pvalue_clopper_pearson(n, k, p)
    cval = @sprintf "%.03f" ccdf(null, k-1)
    P1 = plot(title="(1) Clopper-Pearson")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    annotate!(k+5, 4f(k), text("P-value = $pval", 10, :red))
    plot!([k+7, k+7], [2.4f(k), 3.6f(k)]; arrow=true, c=:red, label="")
    annotate!(k+6, 3f(k), text("2×", 10, :red, :right))
    annotate!(k+5, 2f(k), text("one-tailed = $cval", 10, :red))
    plot!([k+7, k+1], [1.5f(k), 1.5f(k+1)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_sterne(n, k, p)
    j = _search_boundary(_pdf_le, 2mode(null)-k, -1, (null, pdf(null, k)))
    xlimj = (-0.5, j+0.5)
    P2 = plot(title="(2) Sterne")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    plot!(f, xlimj...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!([k-0.5, j+0.5], [f(k), f(k)]; label="", c=2, ls=:dash)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+1], [2.5f(k), 0.8f(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+1], [2.5f(k), 0.8f(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wilson(n, k, p)
    normal = Normal(μ, √σ²)
    g(x) = pdf(normal, x)
    P3 = plot(title="(3) Wilson")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(g, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ²)", c=2)
    plot!(g, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(g, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wald(n, k, p)
    normal = Normal(μ, √σ̂²)
    h(x) = pdf(normal, x)
    P4 = plot(title="(4) Wald")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(h, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ̂²)", c=2)
    plot!(h, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(h, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    plot(P1, P2, P3, P4; size=(800, 520), layout=(2, 2))
    plot!(; plot_title="model: $nullname,   data: n=$n, k=$k")
end

#illustrate_binomial_pvalues()
```

Out[30]: illustrate_binomial_pvalues (generic function with 1 method)

In [31]:
```
# 二項分布モデルの4種のP値函数のプロット

function plot_binom_pvaluefunctions(; n = 20, k = 6, kwargs...)
    P1 = plot(p -> pvalue_clopper_pearson(n, k, p), 0, 1;
        label="", title="Clopper-Pearson", c=1)
    P2 = plot(p -> pvalue_sterne(n, k, p), 0, 1;
        label="", title="Sterne", c=2)
    P3 = plot(p -> pvalue_wilson(n, k, p), 0, 1;
        label="", title="Wilson", c=3)
    P4 = plot(p -> pvalue_wald(n, k, p), 0, 1;
        label="", title="Wald", ls=:dash, c=4)
    plot(P1, P2, P3, P4; size=(800, 520), layout=(2, 2))
    plot!(; xtick=0:0.1:1, ytick=0:0.1:1)
    plot!(; xguide="success rate parameter p",
            yguide="P-value", guidefontsize=10)
    plot!(; plot_title="data: n = $n, k = $k",
            plot_titlefontsize=12)
    plot!(; titlefontsize=12, kwargs...)
end

#plot_binomial_pvaluefunctions(; n = 20, k = 6)
```

Out[31]: plot_binom_pvaluefunctions (generic function with 1 method)

In [32]:
```
# 二項分布モデルのP値函数の動画（データ k を動かしてアニメ化）

function gif_binomtest(; n = 20, fn = "images/pvaluefunction20.gif", fps = 5, step=1)
    anim = @animate for k in [0:step:n; n-1:-step:1]
        plot(p -> pvalue_clopper_pearson(n, k, p), 0, 1; label="Clopper-Pearson")
        plot!(p -> pvalue_sterne(n, k, p), 0, 1; label="Sterne")
        plot!(p -> pvalue_wilson(n, k, p), 0, 1; label="Wilson")
        plot!(p -> pvalue_wald(n, k, p), 0, 1; label="Wald", ls=:dash)
        plot!(; xtick=0:0.1:1, ytick=0:0.1:1)
        plot!(; xguide="success rate parameter p", yguide="P-value")
        title!("data: n = $n, k = $k")
        plot!(; size=(600, 300))
        2k > n && plot!(; legend=:topleft)
    end
    gif(anim, fn; fps)
end
```

Out[32]: gif_binomtest (generic function with 1 method)

In [33]:
```
# P値函数と信頼区間を同時プロット

function plot_binom_pvalue_and_confint(
        pvalfunc, cifunc, n, k;
        α = 0.05, name = "", size=(600, 360), kwargs...)
    p_L, p_U = ci = cifunc(n, k; α)
    plot(p -> pvalfunc(n, k, p), 0, 1;
        label="P-value function", c=1)
    hline!([α]; label="significance level", c=:red, ls=:dot, lw=1.5)
    plot!(ci, fill(α, 2); label="confidence interval", c=:red, lw=4)
    vline!([p_L]; label="p_L", c=2, ls=:dash)
    vline!([p_U]; label="p_U", c=3, ls=:dashdot)
    plot!(; xtick=0:0.1:1, ytick=0:0.05:1, tickfontsize=7)
    plot!(; xguide="success rate parameter p", yguide="P-value")
    plot!(; legendfontsize=9)
    title!("$(name)data = (n,k) = ($n,$k), significance level = α = $α")
    plot!(; size, kwargs...)
end

#plot_binom_pvalue_and_confint(
#    pvalue_clopper_pearson, confint_clopper_pearson, 20, 6;
#    name = "Clopper-Pearson: ")
```

4種の信頼区間の比較

In [34]:
```
function plot_binom_confints(n, ks=0:n÷10:n; α=0.05, size=(400, 320), kwargs...)
    xs = vcat(([k, k, NaN] for k in ks)...)
    clopper_pearson = vcat(([confint_clopper_pearson(n, k; α); NaN] for k in ks)...)
    sterne = vcat(([confint_sterne(n, k; α); NaN] for k in ks)...)
    wilson = vcat(([confint_wilson(n, k; α); NaN] for k in ks)...)
    wald = vcat(([confint_wald(n, k; α); NaN] for k in ks)...)
    P = plot(; legend=:bottomright)
    plot!(xs .- 0.03n, lw=5, clopper_pearson; label="Clopper-Pearson")
    plot!(xs .- 0.01n, lw=5, sterne; label="Sterne")
    plot!(xs .+ 0.01n, lw=5, wilson; label="Wilson")
    plot!(xs .+ 0.03n, lw=5, wald; label="Wald")
    title!("n = $n,  α = $α", titlefontsize=12)
    plot!(; xguide="data k", yguide="success rate parameter p")
    plot!(; xtick=ks, ytick=0:0.1:1)
    plot!(; size, kwargs...)
end

#plot_binom_confints(100, 0:10:100)
```

Out[34]: plot_binom_confints (generic function with 2 methods)

1 比率の検定と信頼区間

以下のようなことを行いたい.

(1) 当たりが未知の確率で出るルーレットを n 回まわしたときに, k 回当たりが出たというデータから, ルーレットで当たりが出る確率を推定する.

(2) 壺の中に赤い玉と白い玉が大量に入っているとき, そこから n 個を無作為抽出したときに k 個が赤い玉であったというデータから, 壺の赤い玉の割合を推定する.

このような推定を以下では **比率の推定** と呼ぶことにする.

ルーレットでの当たりの確率や壺の中の赤い玉の割合を総称して **比率** (proportion) と呼んでいる.

目標は比率の信頼区間の構成である.

そのためには, 検定と信頼区間の表裏一体性より, P値を適切に定義すればよい.

以下では, ルーレットで当たりが出たり, 壺から赤い玉が出ることを **成功** と呼ぶことにする.

注意: 現実世界の統計分析では「成功」は「死亡」を意味する場合もあって, 良いことばかりとは限らない.

1.1 P値と信頼区間の定義の復習

比率の二項分布モデルでの取り扱いを例にP値と信頼区間の定義を復習しよう.

データ: n 回中 k 回成功

統計モデル: 統計モデルとして二項分布を使用する:

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, 2, \ldots, n)$$

二項分布の累積分布函数を次のように書く:

$$\mathrm{cdf}(\mathrm{Binomial}(n,p), k) = \sum_{i \le k} P(j|n,p).$$

他の分布の累積分布函数も同様に書くことにする.

検定される仮説: $p = p_0$ (p_0 は具体的な数値)

P値の大雑把な定義: 仮説 $p = p_0$ 下の統計モデル内でデータの数値「n 回中 k 回成功」以上に極端な値が生じる確率またはその近似値.

「データの数値以上に極端な」の正確な定義や近似の仕方については別に与えることになる.

P値の使い方: データの数値「n 回中 k 回成功」に関する仮説 $p = p_0$ のP値は, 仮説 $p = p_0$ 下の **統計モデルとデータの数値の整合性** の指標として使われる.

以下ではP値を次のように書く:

$$\mathrm{pvalue}(k|n, p = p_0) = (\text{データ「}n\text{ 回中 }k\text{ 回成功」に関する仮説 }p = p_0\text{ のP値}).$$

有意水準: $0 \le \alpha \le 1$ (5% や 1% など小さな値にする).

検定: P値が有意水準未満ならば仮説 $p = p_0$ は **棄却** されたという.

ただし, これは閾値 α に基いて, 単に仮説 $p = p_0$ 下の統計モデルとデータの数値の整合性が低過ぎると判断することを超えた意味を持たないことには注意しなければいけない.

データの取得法に問題があった可能性や仮説 $p = p_0$ も含めた統計モデルの前提のどれかが妥当でない可能性を疑うことになる.

仮説 $p = p_0$ がデータの数値によって棄却されない場合にはその妥当性に関する判断を保留する.

信頼度(信頼係数): $1 - \alpha$.

信頼区間: データの数値を用いた検定で棄却されない成功確率パラメータ $p = p_0$ の全体の集合.

以下では信頼区間(confidence interval)を次のように書く:

$$\mathrm{confint}(k|n, \alpha) = \{\, p_0 \in [0, 1] \mid \mathrm{pvalue}(k|n, p = p_0) \ge \alpha \,\}.$$

信頼区間の使い方: 信頼区間に含まれるパラメータ値 $p = p_0$ についてはその妥当性についての判断は保留する. 信頼区間に含まれないパラメータ値 $p = p_0$ については, 仮説 $p = p_0$ が検定によって棄却されたと考える. (注意: 実際には統計モデルのすべての前提が疑われることになる.)

1.2 二項分布モデルのP値達の定義を1枚の図で要約

In [35]:
```
illustrate_binomial_pvalues()
```

Out[35]:

1.3 4種のP値函数のプロット

In [36]:
```
plot_binom_pvaluefunctions()
```

Out[36]:

In [37]:
```
gif_binomtest()
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\pvaluefunction20.gif
@ Plots D:\.julia\packages\Plots\mnkn4\src\animation.jl:114
```

Out[37]:

In [38]:
```
plot_binom_pvaluefunctions(n = 100, k = 30)
```

Out[38]:

In [39]:
```
gif_binomtest(; n = 100, fn = "images/pvaluefunction100.gif", fps = 20)
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\pvaluefunction100.gif
@ Plots D:\.julia\packages\Plots\mnkn4\src\animation.jl:114
```

Out[39]:

PDFファイルではこの動画を見ることはできない. 作成した動画は

- https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction20.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction20.gif)
- https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction100.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/pvaluefunction100.gif)

でも見ることができる.

1.4 4種の信頼区間の同時プロット

In [40]:
```
plot(
    plot_binom_confints(10), plot_binom_confints(40),
    plot_binom_confints(160), plot_binom_confints(640);
    size=(800, 700), layout=(2,2),
    plot_title="binomial confidence intervals"
)
```

Out[40]:

- 信頼区間は n を4倍すると概ね半分の長さになる.
- 多くの場合にClopper-Pearsonの信頼区間の長さが最も長くなっている.
- 多くの場合にWilsonの信頼区間が最も短くなっている.
- Waldの信頼区間は k または $n-k$ が小さいときに他の3つからのずれが大きくなる.
- n を大きくするとどれもほぼ同じになる.

2 Clopper-Pearson

2.1 Clopper-Pearsonの信頼区間を与えるP値関数

データ「n 回中 k 回成功」について, 「データの数値以上に極端な」の意味を「k 以上の」または「k 以下の」の確率の小さい方とした場合. ただし, P値はその確率の2倍として定義する(1 を超えないように 1 より大きくなったら 1 に切り詰める):

$$\mathrm{pvalue}_{\mathrm{CP}}(k|n,p=p_0) = \min\begin{pmatrix} 1 \\ 2\,\mathrm{cdf}(\mathrm{Binomial}(n,p_0),k) \\ 2(1-\mathrm{cdf}(\mathrm{Binomial}(n,p_0),k-1)) \end{pmatrix}$$

$\mathrm{cdf}(\mathrm{Binomial}(n,p_0),k)$ は二項分布の累積分布函数なので, 二項分布 $\mathrm{Binomial}(n,p_0)$ 内で k 以下になる確率であり, $1-\mathrm{cdf}(\mathrm{Binomial}(n,p_0),k-1)$ は k 以上になる確率である. それらの2倍の小さい方をP値として採用すると, Clopper-Pearson の信頼区間を与えるP値函数が得られる. ただし, それだとP値の値が 1 を超えてしまう場合が出て来て「見ばえ」がよくないので,

1 を超えていたら 1 に切り詰めている.

復習: $\mathrm{cdf}(D, x)$ のcdfは累積分布函数(cumulative distribution function)の略であり, 確率分布 D に従ってランダムに生成された値が x 以下になる確率を意味している. x の動く範囲が整数に限られている場合には $1 - \mathrm{cdf}(D, x-1)$ は確率分布 D に従ってランダムに生成された値が x 以上になる確率に一致する. 具体的な式で書くと,

$$\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k) = \sum_{i=0}^{k} \binom{n}{i} p_0^i (1-p_0)^{n-i},$$
$$1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k-1) = \sum_{i=k}^{n} \binom{n}{i} p_0^i (1-p_0)^{n-i}.$$

しかし, この公式を使ってコンピュータ上で実装することは非効率的でかつ信頼区間の公式を得るためにも不便であり, 二項分布とベータ分布の関係を使うという驚くべき方法が有効になる. (二項分布とベータ分布の関係がそのように役に立つことを知らない理系の大学卒業生は結構多いので, ここでこれを学んで理解しておいて知識を普及させるとよいように思われる.)

2.2 Clopper-Pearsonの信頼区間

信頼区間の下限と上限の値 p_L, p_U (L, U はそれぞれ lower, upper の頭文字である)を次の条件で定義する:

$$1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_L), k-1) = \alpha/2,$$
$$\mathrm{cdf}(\mathrm{Binomial}(n, p_U), k) = \alpha/2.$$

この式を直観的に理解したい人はすぐ後で説明する図による説明を参照せよ.

前節でのP値の定義は次のClopper-Pearsonの信頼区間を与える:

$$\mathrm{confint}_{\mathrm{CP}}(k|n, \alpha) = [p_L, p_U].$$

実際の計算の方法: 二項分布とベータ分布の関係

$$1 - \mathrm{cdf}(\mathrm{Binomial}(n, p), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p),$$
$$\mathrm{cdf}(\mathrm{Binomial}(n, p), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p)$$

と分位点函数 quantile が累積分布函数 cdf の逆函数であることより,

$$p_L = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2),$$
$$p_U = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2).$$

によって, p_L, p_U を計算可能である.

復習: ベータ分布の分位点函数(quantile function, 累積分布函数の逆函数)は有名な基本特殊函数である正則化された不完全ベータ函数の逆函数になっており, コンピュータでの基本特殊函数ライブラリの中に含まれているので, それを便利に使えばよい.

2.3 Clopper-Pearsonの信頼区間の図による把握

以下の図をよく見れば, $\alpha = 5\%$ の場合の $p_L = p_L$ と $p_U = p_U$ を直観的に把握できるだろう. (L, U はそれぞれ lower, upper の略である.)

まず, データの数値が与えられたとき, モデルのパラメータ値をP値に対応させる **P値函数** (P-value function)の立場では, **信頼区間** (confidence interval)の解釈は易しい.

信頼区間はP値函数が有意水準の値を取る2つのパラメータ値に挟まれた区間になる.

しかし, P値函数全体のグラフを描けば, 信頼区間を計算しなくても, データとの整合性が低過ぎないパラメータ値の範囲は明らかだろう.

In [41]:
```
plot_binom_pvalue_and_confint(
    pvalue_clopper_pearson, confint_clopper_pearson, 100, 30;
    name = "Clopper-Pearson: ", xlim=(0.15, 0.5), xtick=0:0.05:1)
```

Out[41]:

Clopper-Pearsonの信頼区間を与えるP値函数のグラフのてっぺんが平らに切れているのは単に片側確率の2倍をP値の定義としているのではなく, 1 を超えたら 1 に切り詰めるという定義を採用しているからである. これはP値の値の範囲を 0 以上 1 以下の範囲に収めるための便宜に過ぎない.

P値函数を経由せずに信頼区間の計算の仕方を直接的に理解するには以下のように考えればよい.

$1-\alpha=95\%$ 信頼区間の下限を求めるためには, 二項分布モデルの成功確率パラメータ p を小さくして行って, モデル内でデータの数値 k 以上の値が生成される確率の2倍(＝Clopper-Pearsonの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になる p の値を見つければよい. そのとき, モデル内でデータの数値 k 以上の値が生成される確率は $\alpha/2=2.5\%$ になっている. そのような p を $p_L=p_L$ とする.

In [42]:
```
var"Clopper-Pearson: 下側では p = p_L がぎりぎり"
```

Out[42]:

$1-\alpha=95\%$ 信頼区間の上限を求めるためには, 二項分布モデルの成功確率パラメータ p を大きくして行って, モデル内でデータの数値 k 以下の値が生成される確率の2倍(＝Clopper-Pearsonの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になる p の値を見つければよい. そのとき, モデル内でデータの数値 k 以下の値が生成される確率は $\alpha/2=2.5\%$ になっている. そのような p を

$p_U = p_U$ とする.

In [43]:
```
var"Clopper-Pearson: 上側では p = p_U がぎりぎり"
```

Out[43]:

以上をまとめると以下のような図を描ける.

In [44]:
```
var"Clopper-Pearson: n × 95%信頼区間"
```

Out[44]:

この場合のClopper-Pearsonの信頼区間は $[0.2124, 0.3998]$ になっている.

以上の図は[「例：ベータ函数と二項分布の関係とその応用」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)(https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)の「Clopper-Pearsonの信頼区間の定義」の節で使ったのと同じものである.

3 Sterne

3.1 Sterneの信頼区間を与えるP値関数

「データの数値以上に極端な」の意味を「生成される確率がデータの数値以下の」とした場合:

$$\mathrm{pvalue}_{\mathrm{Stern}}(k|n, p = p_0) = \sum_{j \text{ satisfying } P(j|n,p_0) \le P(k|n,p_0)} P(j|n, p_0).$$

これは, モデル内での生成確率 $P(j|n, p_0)$ がデータの数値と同じ値の生成確率 $P(k|n, p_0)$ 以下になる j について確率 $P(j|n, p_0)$ を足し上げたものである.

注意: Clopper-Pearsonの信頼区間を与えるP値もSterneの信頼区間を与えるP値も第一種の過誤の確率が常に有意水準以下になるという意味で保守的(conservative)なP値の定義になっている. そして, Sterneの信頼区間を与えるP値の方が小さな値になることが多いということも知られている. この基準ではSterneの信頼区間を与えるP値に方がClopper-Pearsonの信頼区間を与えるP値よりも優れていると言える. しかし, Sterneの信頼区間を与えるP値の効率的実装はClopper-Pearsonの信頼区間を与えるP値の実装よりもずっと複雑になってしまう.

3.2 Sterneの信頼区間

上で定義したP値の定義はSterneの信頼区間(Sterne (1954) (https://www.jstor.org/stable/2333026))を与える:

$$\mathrm{confint}_{\mathrm{Stern}}(k|n,\alpha) = \{\, p_0 \in [0,1] \mid \mathrm{pvalue}_{\mathrm{Stern}}(k|n, p=p_0) \geq \alpha \,\}.$$

Sterneの信頼区間は被覆確率(coverage probability)や長さの期待値(expected length)の基準から, Clopper-Pearsonの信頼区間よりも優れていることが知られている.

しかし, 効率的な計算の実装はずっと複雑になり, Sterneの信頼区間の計算のためにはコンピュータの取り扱いに習熟する必要がある. このノートでは読者にそこまで要求するつもりはない. しかし, プログラミングに自信がある人達はSterneの信頼区間の実装を自力で試してみると大変良い勉強になると思われる.

注意: このタイプの信頼区間は稀に区間にならない場合がある. そのような稀な場合にも区間にすることにこだわるならば, P値が α 以上になる $p = p_0$ 全体の集合を含む最小の区間を改めて信頼区間の定義とするというような処置が必要になる. その場合には以下のようになる:

$$\mathrm{confint}_{\mathrm{Stern}}(k|n,\alpha) = [p_{\mathrm{inf}}, p_{\mathrm{sup}}].$$

ここで,

$$\begin{aligned} p_{\mathrm{inf}} &= \inf\{\, p_0 \in [0,1] \mid \mathrm{pvalue}_{\mathrm{Stern}}(k|n, p=p_0) \geq \alpha \,\}, \\ p_{\mathrm{sup}} &= \sup\{\, p_0 \in [0,1] \mid \mathrm{pvalue}_{\mathrm{Stern}}(k|n, p=p_0) \geq \alpha \,\}. \end{aligned}$$

注意: Sterneの信頼区間とClopper-Pearsonの信頼区間などの比較については例えば以下の報告がある:

- 原茂恵美子, 武藤彬正, 宮島育哉, 榊原伊織, SASによる二項比率における正確な信頼区間の比較, 2014. [pdf (https://www.sas.com/content/dam/SAS/ja_jp/doc/event/sas-user-groups/usergroups14-d-05.pdf)]

3.3 Sterneの信頼区間の図による把握

以下の図をよく見れば, $\alpha = 5\%$ の場合の $p_L = p_L$ と $p_U = p_U$ を直観的に把握できるだろう.

まず, データの数値が与えられたとき, モデルのパラメータ値をP値に対応させる **P値函数** (P-value function)の立場では, **信頼区間** (confidence interval)の解釈は易しい.

信頼区間はP値函数が有意水準の値を取る2つのパラメータ値に挟まれた区間になる.

しかし, P値函数全体のグラフを描けば, 信頼区間を計算しなくても, データとの整合性が低過ぎないパラメータ値の範囲は明らかだろう.

In [45]:
```
plot_binom_pvalue_and_confint(
    pvalue_sterne, confint_sterne, 100, 30;
    name = "Sterne: ", xlim=(0.15, 0.5), xtick=0:0.05:1)
```

Out[45]:

P値函数を経由せずに信頼区間の計算の仕方を直接的に理解するには以下のように考えればよい.

$1-\alpha = 95\%$ 信頼区間の下限を求めるためには, 二項分布モデルの成功確率パラメータ p を小さくして行って, モデル内でデータの数値 k 以上の値が生成される確率とその反対側の確率の和(＝Sterneの信頼区間を与えるP値)が $\alpha = 5\%$ 以上になるような p の値の下限を見つければよい. そのような p を $p_L = p_L$ とする.

In [46]:
```
var"Sterne: 下側では p = p_L がぎりぎり"
```

Out[46]:

$1-\alpha=95\%$ 信頼区間の上限を求めるためには, 二項分布モデルの成功確率パラメータ p を大きくして行って, モデル内でデータの数値 k 以下の値が生成される確率とその反対側の確率の和(＝Sterneの信頼区間を与えるP値)が $\alpha=5\%$ 以上になるような p の値の上限を見つければよい. そのような p を $p_U=p_U$ とする.

In [47]:
```
var"Sterne: 上側では p = p_U がぎりぎり"
```

Out[47]:

以上をまとめると以下のような図を描ける.

In [48]:
```
var"Sterne: n × 95%信頼区間"
```

Out[48]:

この場合のSterneの信頼区間は $[0.2142, 0.3996]$ になっている.

これは, 同じ場合のClopper-Pearsonの信頼区間の $[0.2124, 0.3998]$ よりもほんの少し狭いが, ほぼ同じであると言ってよいだろう.

3.4 おまけ: R言語のbinom.testについて

R言語の `binom.test` がデフォルトで表示するP値はSterneの信頼区間を与えるP値であるが, それと同時に表示される信頼区間はClopper-Pearsonの信頼区間の方になってしまっている. そのせいで, P値が有意水準未満なのに, 帰無仮説を与える成功確率パラメータ値が信頼区間に含まれてしまう場合がある.

In [49]:
```
R"binom.test(10, 20, p = 0.72, conf.level = 0.95)"
```

Out[49]:
```
RObject{VecSxp}

        Exact binomial test

data:  10 and 20
number of successes = 10, number of trials = 20, p-value = 0.04277
alternative hypothesis: true probability of success is not equal to 0.72
95 percent confidence interval:
 0.2719578 0.7280422
sample estimates:
probability of success
                   0.5
```

上のRの `binom.test` の結果と以下を比較してみよ.

In [50]:
```
α = 0.05
n, k = 20, 10
null = Binomial(n, 0.72)
```

Out[50]:
```
Binomial{Float64}(n=20, p=0.72)
```

In [51]:
```
# Sterneの信頼区間を与えるP値を定義通りに計算(計算効率はよくない)
p_value_sterne =
    sum(pdf(null, i) for i in 0:n if pdf(null, i) ≲ pdf(null, k))
@show p_value_sterne;
```

```
p_value_sterne = 0.04276902774675169
```

In [52]:
```
# p を pvalue_sterne - α に対応させる函数が
# 0 になる 0 から 1 のあいだの p 達を求めれば,
# Sterneの信頼区間を計算できる.
conf_int_sterne = find_zeros(0, 1) do p
    null = Binomial(n, p)
    pvalue_sterne =
        sum(pdf(null, i) for i in 0:n if pdf(null, i) ≲ pdf(null, k))
    pvalue_sterne - α
end
@show conf_int_sterne;
```

```
conf_int_sterne = [0.2927095227968867, 0.7072904772031132]
```

`f(a, b) do ... end` 構文の解説 (https://docs.julialang.org/en/v1/manual/functions/#Do-Block-Syntax-for-Function-Arguments)

In [53]:
```
# Clopper-Pearsonの信頼区間を与えるP値を定義通りに計算
p_value_clopper_pearson = min(1, 2cdf(null, k), 2(1-cdf(null, k-1)))
@show p_value_clopper_pearson;
```

```
p_value_clopper_pearson = 0.06093072477502701
```

In [54]:
```
# Clopper-Pearsonの信頼区間をベータ分布を使って計算
conf_int_clopper_pearson =
    [quantile(Beta(k, n-k+1), α/2), quantile(Beta(k+1, n-k), 1-α/2)]
@show conf_int_clopper_pearson;
```

```
conf_int_clopper_pearson = [0.27195784956079183, 0.7280421504392082]
```

有意水準が $\alpha = 5\%$ で, 「20 回中 10 回成功」というデータが得られたときに, 仮説「成功確率は $p = 0.72$ である」のP値と成功確率 p の信頼区間は以下の表のようになる:

計算法	仮説 $p = 0.72$ のP値	信頼区間
Sterne	$4.277\% < \alpha = 5\%$	$0.72 \notin [0.2927095, 0.7072905]$
Clopper-Pearson	$6.093\% > \alpha = 5\%$	$0.72 \in [0.2719578, 0.7280422]$

このようにP値と信頼区間の表裏一体性に忠実であれば, P値が有意水準未満であることと, 信頼区間に帰無仮説を与えるパラメータ値が含まれないことが同値になる. しかし, R言語の `binom.test` は信頼区間の表裏一体正を無視しているので, 整合性がない結果が稀に表示されることがあるので注意が必要である.

他にも `binom.test(31, 100, p = 0.41)` で表示されるP値と 95% 信頼区間も不整合になる.

In [55]:
```
R"binom.test(31, 100, p = 0.41)"
```

Out[55]:
```
RObject{VecSxp}

        Exact binomial test

data:  31 and 100
number of successes = 31, number of trials = 100, p-value = 0.04238
alternative hypothesis: true probability of success is not equal to 0.41
95 percent confidence interval:
 0.2212888 0.4103146
sample estimates:
probability of success
                  0.31
```

帰無仮説 $p = 0.41$ のP値が 5% 未満なのに, 95% 信頼区間が $p = 0.41$ を含んでいる.

In [56]:
```
@show pvalue_sterne(100, 31, 0.41)
@show confint_sterne(100, 31)
println()
@show pvalue_clopper_pearson(100, 31, 0.41)
@show confint_clopper_pearson(100, 31);
```

```
pvalue_sterne(100, 31, 0.41) = 0.04238064409855778
confint_sterne(100, 31) = [0.22393060068670914, 0.409659837525102]

pvalue_clopper_pearson(100, 31, 0.41) = 0.05074263936104193
confint_clopper_pearson(100, 31) = [0.22128879213067107, 0.41031455416019924]
```

このような例を見れば, 必然性皆無の閾値である有意水準の設定の下で, P値が有意水準未満になったか否かで一喜一憂したり, 帰無仮説のパラメータ値を信頼区間が含んでいるか否かで一喜一憂することが, 科学的にどれだけ無意味であるかもわかるだろう.

P値や信頼区間の実装の仕方で閾値の設定による結果は変わることがあるし, 統計ソフトの函数が表示する1つの結果内でさえ, P値と信頼区間のあいだに整合性が無くなっている場合さえある.

3.5 おまけ関連問題: R言語のコードを実行してみよ.

何らかの方法で以下のR言語のコードを実行してみよ.

```
binom.test(10, 20, p = 0.72, conf.level = 0.95)

binom.test(31, 100, p = 0.41)
```

解答例1(普通): [WindowsでのR/RStudioのインストール (https://oku.edu.mie-u.ac.jp/~okumura/stat/R-win.html)](https://oku.edu.mie-u.ac.jp/~okumura/stat/R-win.html)を参考にして, 自分のパソコンにR言語をインストールして実行してみた.

解答例2(易しい): [paiza.IO の R Online (https://paiza.io/ja/languages/r)](https://paiza.io/ja/languages/r) で「コードを作成してみる(無料)」でR言語のコードを実行してみた.

解答例3(少し面倒): Julia言語とR言語を自分のパソコンで使えるようにして, さらにJulia言語に [RCall.jl (https://github.com/JuliaInterop/RCall.jl)](https://github.com/JuliaInterop/RCall.jl) をインストールして, このノートと同じ方法でJulia言語からR言語のコードを実行してみた.

解答例4(これができれば素晴らしい): 自分のパソコンでJulia言語, Python言語, R言語を使える環境をすでに整備してあるので, R言語のコードはすぐに実行できた. これをやれるだけの能力がある人達は休日にがんばってこれに挑戦してみるとよいと思う.

4 Wilson

4.1 Wilsonの信頼区間を与えるP値函数

データ「n 回中 k 回成功」について,「データの数値以上に極端な」の意味を二項分布の正規分布近似のもとで「k 以上または k 以下の」とした場合:

$$\mathrm{pvalue}_{\mathrm{Wilson}}(k|n, p = p_0) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{|k - np_0|}{\sqrt{np_0(1-p_0)}}\right)\right).$$

右辺は標準正規分布において絶対値が $|k - np_0|/\sqrt{np_0(1-p_0)}$ 以上になる確率である.

4.2 Wilsonの信頼区間

以下で使う記号を準備しよう. $\hat{p}, \widehat{\mathrm{SE}}, \mathrm{SE}$ を以下のように定義する:

$$\hat{p} = \frac{k}{n}, \quad \widehat{\mathrm{SE}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \quad \mathrm{SE} = \sqrt{\frac{p(1-p)}{n}}.$$

ハットが付いている記号はデータの数値から計算した推定値であり, SE は標準誤差(standard error)の略である. SE は k が二項分布 $\mathrm{Binomial}(n, p)$ に従ってランダムに生成されているときの $\hat{p}$ が従う分布の標準偏差になる.

次が成立していることにも注意せよ(左辺の分子分布を n で割る):

$$\frac{k - np}{\sqrt{np_0(1-p)}} = \frac{\hat{p} - p}{\mathrm{SE}}.$$

k が二項分布 $\mathrm{Binomial}(n, p)$ 従ってランダムに生成されているとき, これは近似的に標準正規分布に従うのであった.

この場合の信頼区間は以下のように記述される.

まず, $z_{\alpha/2}$ を次のように定める:

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2).$$

これは標準正規分布において $z_{\alpha/2}$ 以上になる確率が $\alpha/2$ になるという意味である.

In [57]:
```
@show quantile(Normal(0, 1), 1 - 0.05/2)
@show quantile(Normal(0, 1), 1 - 0.01/2)
@show quantile(Normal(0, 1), 1 - 0.001/2);
```

```
quantile(Normal(0, 1), 1 - 0.05 / 2) = 1.9599639845400576
quantile(Normal(0, 1), 1 - 0.01 / 2) = 2.5758293035489053
quantile(Normal(0, 1), 1 - 0.001 / 2) = 3.290526731491931
```

In [58]:
```
var"z_{0.025}の定義"
```

Out[58]:

以下, 記号の簡単のため次のようにおく:

$$z = z_{\alpha/2}.$$

このとき, 標準正規分布において絶対値が $(\hat{p} - p)/\mathrm{SE}$ の絶対値以上の値が生成される確率が α に等しいことと

$$\frac{(\hat{p} - p)^2}{\mathrm{SE}^2} = z_{\alpha/2}^2 = z^2$$

となることは同値である. $\mathrm{SE}^2 = p(1-p)/n$ なので, これは p に関する次の二次方程式と同値である:

$$\left(1+\frac{z^2}{n}\right)p^2 - 2\left(\hat{p}+\frac{z^2}{2n}\right)p + \hat{p}^2 = 0.$$

そこで, a, b, c を

$$a = 1+\frac{z^2}{n}, \quad b = \hat{p}+\frac{z^2}{2n}, \quad c = \hat{p}^2$$

とおく. このとき, 上の二次方程式の2つの解を $p_- \le p_+$ と書く:

$$p_\pm = \frac{b \pm \sqrt{b^2-ac}}{a} = \frac{1}{1+z^2/n}\left(\hat{p}+\frac{z^2}{2n} \pm z\sqrt{\widehat{\mathrm{SE}}^2+\frac{z^2}{4n^2}}\right).$$

このとき,

$$\mathrm{confint}_{\mathrm{Wilson}}(k|n,\alpha) := \{\, p \in [0,1] \mid \mathrm{pvalue}_{\mathrm{Wilson}}(k|n,p) \ge \alpha \,\} = [p_-, p_+].$$

これを **Wilsonの信頼区間** と呼ぶ.

注意: Wilsonの信頼区間を与えるP値はシンプルに定義されたが, 対応する信頼区間の計算には二次方程式を解く必要がある. 次の節で説明するWaldの信頼区間はこの部分を簡略化したものとみなされる.

注意: 上で省略した計算の細部は自分で埋めよ. Wilsonの信頼区間の導出については [「例：ベータ函数と二項分布の関係とその応用」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)の「Wilsonの信頼区間の基礎」の節も参照せよ.

4.3 Wilsonの信頼区間の図による把握

以下の図をよく見れば, $\alpha = 5\%$ の場合の $p_L = p_L$ と $p_U = p_U$ を直観的に把握できるだろう.

まず, データの数値が与えられたとき, モデルのパラメータ値をP値に対応させる **P値函数** (P-value function)の立場では, **信頼区間** (confidence interval)の解釈は易しい.

信頼区間はP値函数が有意水準の値を取る2つのパラメータ値に挟まれた区間になる.

しかし, P値函数全体のグラフを描けば, 信頼区間を計算しなくても, データとの整合性が低過ぎないパラメータ値の範囲は明らかだろう.

In [59]:
```
plot_binom_pvalue_and_confint(
    pvalue_wilson, confint_wilson, 100, 30;
    name = "Wilson: ", xlim=(0.15, 0.5), xtick=0:0.05:1)
```

Out[59]:

P値函数を経由せずに信頼区間の計算の仕方を直接的に理解するには以下のように考えればよい.

$1-\alpha=95\%$ 信頼区間の下限を求めるためには, 二項分布モデルの成功確率パラメータ p を小さくして行って, その二項分布を近似する正規分布内でデータの数値 k 以上の値が生成される確率とその反対側の確率の和(＝Wilsonの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になるような p を見つければよい. そのような p を $p_L=p_L$ とする.

In [60]:
```
var"Wilson: 下側では p = p_L がぎりぎり"
```

Out[60]:

$1-\alpha=95\%$ 信頼区間の上限を求めるためには, 二項分布モデルの成功確率パラメータ p を大きくして行って, その二項分布を近似する正規分布内でデータの数値 k 以下の値が生成される確率とその反対側の確率の和(＝Wilsonの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になるような p を見つければよい. そのような p を $p_U=p_U$ とする.

In [61]:
```
var"Wilson: 上側では p = p_U がぎりぎり"
```

Out[61]:

以上をまとめると以下のような図を描ける.

In [62]:
```
var"Wilson: n × 95%信頼区間"
```

Out[62]:

この場合のWilsonの信頼区間は $[0.2189, 0.3958]$ である.

これは, 同じ場合のClopper-Pearsonの信頼区間の $[0.2124, 0.3998]$ とSterneの信頼区間は $[0.2142, 0.3996]$ の両方よりも少し狭いが, ほぼ同じであると言ってよいだろう.

4.4 おまけ: R言語のprop.testについて

R言語では `prop.test` 函数でWilsonの信頼区間を与えるP値とWilsonの信頼区間を計算できる. ただし, `correct = FALSE` として, 連続性補正(continuity correction)をオフにしなければ, 上で説明した(補正無しの)Wilsonの信頼区間およびそれを与えるP値とはぴったり一致しない値が表示される.

注意: 連続性補正については「大数の法則と中心極限定理」のノート(https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)の「連続性補正とmid-P補正」の節を参照せよ.

注意: 筆者の個人的な意見では, Wilsonの信頼区間の計算で連続性補正は使用しない方が良いように思われる.

その1つ目の理由は連続性補正を使って, Wilsonの信頼区間をClopper-Pearsonの信頼区間に近付けるくらいなら, 最初からClopper-Pearsonの信頼区間を使った方がより正確で好ましいと考えられるからである. コンピュータを自由に使えなかった時代にはClopper-Pearsonの信頼区間の計算に必要な正則化不完全ベータ函数の逆函数を気軽に利用できなかったかもしれないが, 現代においては基本特殊函数ライブラリの実装が普及しており, そのようなことはない.

2つ目の理由は補正無しのWilsonの信頼区間を与えるP値は第一種の過誤の確率が有意水準を超える場合があるという欠点はあるが, 平均的には, Clopper-Pearsonの信頼区間を与えるP値よりも第一種の過誤の確率が有意水準に近くなる.

Clopper-Pearsonの信頼区間を与えるP値は第一種の過誤の確率が確実に有意水準以下になるという保守的な利点を持つが, 無駄に大きめの値になり易いという欠点がある.

トレードオフを考慮して, Clopper-Pearsonの信頼区間やそれを与えるP値の代わりに, 補正無しのWilsonの信頼区間とそれを与えるP値を使うことは十分に合理的な選択肢になる可能性がある.

注意: Sterneの信頼区間を与えるP値も第一種の過誤の確率が確実に有意水準以下になるという保守的な利点を持っており, Clopper-Pearsonの信頼区間を与えるP値と比較したとき相対的に無駄に大きめの値になり難いという利点を持っている. しかし, 離散分布におけるSterneの信頼区間を与えるP値函数は連続にも単峰型にもならないという欠点や, 効率的な実装が複雑になりやすいという欠点を持つ.

あるP値の定義が他のP値の定義よりも一方的に優れているということはなく, どのP値の定義も利点と欠点を合わせ持っている. どれを使用するかは, ユーザー側のトレードオフの考え方に基づく判断に任されることになる.

In [63]:
```
# これは連続性補正有りの計算結果になっており,
# 上で説明した補正無しのWilsonの信頼区間とそれを与えるP値とは異なる.
R"prop.test(10, 20, p = 0.72, conf.level = 0.95)"
```

Out[63]:
```
RObject{VecSxp}

        1-sample proportions test with continuity correction

data:  10 out of 20, null probability 0.72
X-squared = 3.7723, df = 1, p-value = 0.05211
alternative hypothesis: true p is not equal to 0.72
95 percent confidence interval:
 0.2785367 0.7214633
sample estimates:
  p
0.5
```

In [64]:
```
# 連続性補正無しの計算結果
R"prop.test(10, 20, p = 0.72, conf.level = 0.95, correct = FALSE)"
```

Out[64]:
```
RObject{VecSxp}

        1-sample proportions test without continuity correction

data:  10 out of 20, null probability 0.72
X-squared = 4.8016, df = 1, p-value = 0.02843
alternative hypothesis: true p is not equal to 0.72
95 percent confidence interval:
 0.299298 0.700702
sample estimates:
  p
0.5
```

以上のR言語による計算結果と以下のセルでの計算結果を比較してみよ.

In [65]:
```
n, k, p, α = 20, 10, 0.72, 0.05
@show n k p α
println()
@show z = quantile(Normal(), 1-α/2)
@show p̂ = k/n
@show SE = √(p*(1-p)/n)
@show p̂_L, p̂_U = (k - 0.5)/n, (k + 0.5)/n

p_value_wilson_corrected = 2ccdf(Normal(), (abs(p̂ - p)-0.5/n)/SE)
a, b_L, c_L, b_U, c_U = 1+z^2/n, p̂_L+z^2/(2n), p̂_L^2, p̂_U+z^2/(2n), p̂_U^2
conf_int_wilson_corrected = [(b_L - √(b_L^2-a*c_L))/a, (b_U + √(b_U^2-a*c_U))/a]

p_value_wilson = 2ccdf(Normal(), abs(p̂ - p)/SE)
a, b, c = 1+z^2/n, p̂+z^2/(2n), p̂^2
conf_int_wilson = [(b - √(b^2-a*c))/a, (b + √(b^2-a*c))/a]

println()
@show p_value_wilson_corrected
@show conf_int_wilson_corrected
println()
@show p_value_wilson
@show conf_int_wilson;
```

```
n = 20
k = 10
p = 0.72
α = 0.05

z = quantile(Normal(), 1 - α / 2) = 1.9599639845400576
p̂ = k / n = 0.5
SE = √((p * (1 - p)) / n) = 0.10039920318408907
(p̂_L, p̂_U) = ((k - 0.5) / n, (k + 0.5) / n) = (0.475, 0.525)

p_value_wilson_corrected = 0.052107267466663096
conf_int_wilson_corrected = [0.278536702420732, 0.7214632975792679]

p_value_wilson = 0.028433528913982754
conf_int_wilson = [0.299298008198212, 0.7007019918017879]
```

上のセルでは具体的に計算の仕方が分かるようにあえて素朴にコードを書いて実行している. その素朴なコードを解読すれば連続性補正の入れ方もわかるだろう.

4.5 Pearsonのχ²検定との関係

標準正規分布に従う確率変数の2乗は自由度 1 のχ²分布に従うことを使えば, 上のP値は次のように表される:

$$\mathrm{pvalue}_{\mathrm{Wilson}}(k|n, p = p_0) = 1 - \mathrm{cdf}\left(\mathrm{Chisq}(1), \frac{(k - np_0)^2}{np_0(1 - p_0)}\right).$$

このP値を, スコア統計量(score statistics)に基くP値と呼んだり, Pearsonのχ²統計量

$$\frac{(k - np_0)^2}{np_0(1 - p_0)} = \frac{(k - np_0)^2}{np_0} + \frac{((n - k) - n(1 - p_0))^2}{n(1 - p_0)}$$

に基くP値と呼んだりする. 一般にPearsonのχ²統計量は

$$(\text{Pearsonの}\chi^2\text{統計量}) = \sum \frac{(\text{観測値} - \text{帰無仮説下の統計モデル内での期待値})^2}{\text{帰無仮説下の統計モデル内での期待値}}$$

のスタイルで定義される. 上の場合には, 観測値が成功回数 k と失敗回数 $n - k$ であり, 対応する帰無仮説下の統計モデル内での期待値はそれぞれ np_0, $n(1 - p_0)$ である.

注意: Pearsonのχ²統計量は多項分布モデルでの検定や分割表の独立性検定でも大活躍することになる. 興味がわいた人はおまけの解説も参照せよ.

4.6 おまけ: 多項分布モデルにおけるPearsonのχ²検定

Pearsonのχ²統計量は多項分布モデルでも利用できる.

例えば, 検定したい仮説「A, B, Cが生じる確率はそれぞれ $p_{A,0}, p_{B,0}, p_{C,0}$ である」($p_{A,0} + p_{B,0} + p_{C,0} = 1$)のとき, 「$n$ 回中 A, B, C がそれぞれ a, b, c 回生じた」というデータに対して, Pearsonのχ²統計量を

$$(\text{Pearsonの}\chi^2\text{統計量}) = \frac{(a - np_{A,0})^2}{np_{A,0}} + \frac{(b - np_{B,0})^2}{np_{B,0}} + \frac{(c - np_{C,0})^2}{np_{C,0}}$$

で定義できる. 二項分布の中心極限定理の拡張になっている多項分布の中心極限定理によって, このPearsonのχ²統計量は上の仮説下の多項分布内で自由度 2 のχ²分布に近似的に従うことを示せる.

このことを使えば上と同様の方法で多項分布モデルの場合にもP値を定義できる. 例えば, 出目に偏りがあることが疑われているサイコロを n 回ふったら $i = 1, 2, \ldots, 6$ の目が a_i 回出たとき, 仮説「すべての目が等確率で出る」に関するPearsonのχ²統計量は

$$(\text{Pearsonの}\chi^2\text{統計量}) = \sum_{i=1}^{6} \frac{(a_i - n/6)^2}{n/6}$$

になる. 「すべての目が等確率で出る」という仮説下の多項分布内でこのPearsonのχ²統計量は, n が大きいければ, 自由度 5 のχ²分布に近似的に従う. このときP値は, 自由度 5 のχ²分布に従ってランダムに生成される値がデータから計算されたPearsonのχ²統計量の値以上になる確率として定義される.

In [66]:

```
function chisq_multinomial(a, p, n=sum(a))
    sum((a[i] - n*p[i])^2/(n*p[i]) for i in eachindex(a, p))
end

function plot_chisq_multinomial(n, p)
    null = Multinomial(n, float(p)) # model of null hypothesis
    df = ncategories(null) - 1 # degree of freedom

    tmp = zeros(ncategories(null))
    chisqs = [chisq_multinomial(rand!(null, tmp), p) for _ in 1:10^6]

    histogram(chisqs; norm=true, alpha=0.3, bin=0:0.5:40,
        label="Pearson's χ²-statistic")
    plot!(Chisq(df); label="Chisq(df = $df)", lw=1.5)
    title!("n = $n, p = $(replace(string(p), r"Rational{.*}"⇒"", "//"⇒"/"))")
    plot!(; xlim=(-0.5, 20.5), xtick=0:2:30)
end
```

Out[66]:

```
plot_chisq_multinomial (generic function with 1 method)
```

In [67]:

```
plot_chisq_multinomial(300, fill(1//6, 6))
```

Out[67]:

例えば, 有意水準を $\alpha = 0.1\%$ としたとき, $n = 300$ 回中 $1, 2, \ldots, 6$ が出た回数がそれぞれ $75, 35, 43, 40, 47, 60$ ならば, 仮説「すべての目が同じ確率で出る」について, Pearsonのχ²統計量の値は 22.16 になり, 自由度 5 のχ²分布でそれ以上の値になる確率として定義されたP値は 0.05% 未満になるので, 「すべての目が同じ確率で出る」という仮説は有意水準 $\alpha = 0.1\%$ で棄却されることになる.

In [68]:

```
a = [75, 35, 43, 40, 47, 60]
p = fill(1/6, 6)
@show c = chisq_multinomial(a, p)
pvalue_multinomial = ccdf(Chisq(5), c)
@show pvalue_multinomial;
```

```
c = chisq_multinomial(a, p) = 22.16
pvalue_multinomial = 0.0004881674827554182
```

以下のように, サイコロのすべての目が出る確率が $1/6$ でなくても, サイコロの目が出る確率の具体的な値に関する仮説に関する

Pearsonのχ²統計量は自由度 5 のχ²分布に近似的に従う.

In [69]:
```
plot_chisq_multinomial(300, [1//3, 1//6, 1//8, 1//12, 3//16, 5//48])
```

Out[69]:

4.7 おまけ(より進んだ話題): 一般のχ²検定で使うχ²分布の自由度

一般のχ²検定で使用するχ²分布の自由度 df (degree of freedom)はWilks' theorem (https://en.wikipedia.org/wiki/Wilks%27_theorem) より次の公式で決定される:

$$\begin{aligned}\mathrm{df} &= (\text{統計モデルの独立なパラメータの個数}) - (\text{帰無仮説下の統計モデルの独立なパラメータ個数}) \\ &= (\text{帰無仮説によって減るモデルの独立なパラメータ個数}).\end{aligned}$$

例えば, 上のサイコロの例では, 六項分布モデルのパラメータ数は $6-1=5$ 個である. なぜならば $i=1,2,\ldots,6$ の各々の目が出る確率を p_i と書くとき, それらには $p_1+p_2+\cdots+p_6=1$ という条件が課されるので, 独立なパラメータの個数は 5 個になる. そして, 帰無仮説「すべての目が等確率で出る」を課すとすべての p_i の値が $1/6$ に固定されて, 独立に動けるパラメータの個数は 0 になる. 帰無仮説によって独立なパラメータの個数が 5 個減ったので, χ²検定で使うχ²分布の自由度は 5 になる.

例えば, 帰無仮説が

$$p_1+p_6=1/3, \quad p_2+p_5=1/3, \quad p_3+p_4=1/3$$

と書ける. これら3本のうち, 条件 $p_1+p_2+\cdots+p_6=1$ の下で独立な方程式は2本なので, この仮説を課すことによって独立なパラメータの個数は 2 個減る. ゆえに, この場合には, χ²検定で使用するχ²分布の自由度は 2 になる. ただし, この場合には

$$(\text{Pearsonの}\chi^2\text{統計量}) = \sum \frac{(\text{観測値} - \text{帰無仮説下の統計モデル内での期待値})^2}{\text{帰無仮説下の統計モデル内での期待値}}$$

における「帰無仮説下の統計モデル内での期待値」は統計モデルのパラメータを帰無仮説の制限の下での最尤法で決定することによって求めることになる.

In [70]:
```
function chisq_multinomial_example2(a)
    n = sum(a)
    r = length(a)
    @assert r == 6
    mle(i) = if i == 1 || i == 6
        a[i]/(3(a[1] + a[6]))
    elseif i == 2 || i == 5
        a[i]/(3(a[2] + a[5]))
    else # i == 3 || i == 4
        a[i]/(3(a[3] + a[4]))
    end
    sum((a[i] - n*mle(i))^2/(n*mle(i)) for i in 1:r)
end

function plot_chisq_multinomial_example2(n, p)
    null = Multinomial(n, float(p)) # model of null hypothesis
    df = 2 # degree of freedom
    tmp = zeros(ncategories(null))
    chisqs = [chisq_multinomial_example2(rand!(null, tmp)) for _ in 1:10^6]

    histogram(chisqs; norm=true, alpha=0.3, bin=0:0.25:20,
        label="Pearson's χ²-statistic")
    plot!(Chisq(df); label="Chisq(df = $df)", lw=1.5)
    title!("n = $n,  p = $(replace(string(p), r"Rational{.*}"⇒"", "//"⇒"/"))")
    plot!(; xlim=(-0.5, 10.5), xtick=0:20)
end
```

Out[70]: plot_chisq_multinomial_example2 (generic function with 1 method)

In [71]:
```
plot_chisq_multinomial_example2(1000, [1//4, 1//6, 1//8, 5//24, 1//6, 1//12])
```

Out[71]:

上のセルの計算例では $(p_1, p_2, p_3, p_4, p_5, p_6) = (1/4, 1/6, 1/8, 5/24, 1/6, 1/12)$ の場合を扱っているが, その p_i 達は仮説 $p_1 + p_6 = p_2 + p_5 = p_3 + p_4 = 1/3$ を満たしていることに注意せよ. コード中の `mle(i)` のmleは最尤推定(maximum likelihood estimate)を意味している. 仮説 $p_1 + p_6 = p_2 + p_5 = p_3 + p_4 = 1/3$ でパラメータの動く範囲を制限した場合の最尤法の解を手計算で求めて(Lagrangeの未定乗数法を使えば易しい), それを直接的にコーディングしている.

例えば, 有意水準を $\alpha = 0.1\%$ としたとき, $n = 1000$ 回中 $1, 2, \ldots, 6$ が出た回数がそれぞれ $235, 195, 175, 175, 195, 225$ ならば, 仮説 $p_1 + p_6 = p_2 + p_5 = p_3 + p_4 = 1/3$ について, Pearsonのχ²統計量の値は 15.5 になり, 自由度 2 のχ²分布でそれ以上の値になる確率として定義されたP値は 0.05% 未満になるので, 仮説 $p_1 + p_6 = p_2 + p_5 = p_3 + p_4 = 1/3$ は有意水準 $\alpha = 0.1\%$ で棄却されることになる.

In [72]:
```
a = [235, 195, 175, 175, 195, 225]
@show c = chisq_multinomial_example2(a)
pvalue_multinomial_example2 = ccdf(Chisq(2), c)
@show pvalue_multinomial_example2;
```

```
c = chisq_multinomial_example2(a) = 15.5
pvalue_multinomial_example2 = 0.00043074254057568753
```

仮説 $p_1 + p_6 = p_2 + p_5 = p_3 + p_4 = 1/3$ を満たす p_i 達の他の組み合わせでも, その仮説に合わせて作ったPearsonのχ²統計量はその仮説下の多項分布内で近似的に自由度 2 のχ²分布に従う.

In [73]:
```
plot_chisq_multinomial_example2(1000,
    [1//6-1//8, 1//6-1//10, 1//6-1//15, 1//6+1//15, 1//6+1//10, 1//6+1//8])
```

Out[73]:

注意: これに類似した大量の結果が [Wilks' theorem (https://en.wikipedia.org/wiki/Wilks%27_theorem)](https://en.wikipedia.org/wiki/Wilks%27_theorem) から得られることを知った人はこう考えるかもしれない. 統計学を便利に使うためには「Wilks' theorem の証明も理解しなければいけないのか?」と.

統計学ユーザーになるためには証明を理解する必要はない.

しかし, ブラックボックスとして認めて使うことはお勧めできない.

なぜならば, 以上で示したように, コンピュータを使えば Wilks' theorem の帰結が実際に成立していることを用意に確認できるからである.

しかも, 実際に自分でやってみればわかることだが, $n \to \infty$ の理想的な状況で成立している近似が, 有限の n でどれだけ有効な近似になっているかは, コンピュータで数値的に確認しないとよくわからないことが多い. 「百聞は一見に如かず」の世界がそこには広がっている!

応用上実際に使われるのは有限の n の場合なので, 証明を理解するだけの数学的実力の持ち主であっても, コンピュータによる数値的確認を十分にやっていないと, 実践的な応用の場面では失敗する場合が出て来てしまうだろう.

5 Wald

5.1 Waldの信頼区間を与えるP値関数

$\hat{p}, \widehat{\mathrm{SE}}$ を以下のように定義する:

$$\hat{p} = \frac{k}{n}, \quad \widehat{\mathrm{SE}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}.$$

データ「n 回中 k 回成功」について, 「データの数値以上に極端な」の意味を「k 以上または k 以下の」とした場合. ただし, 二項分布の中心極限定理(正規分布近似)と分散を推定値に置き換える粗い近似を使う:

$$\mathrm{pvalue}_{\mathrm{Wald}}(k|n, p = p_0) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{|k - np_0|}{\sqrt{n\hat{p}(1-\hat{p})}}\right)\right).$$

$\mathrm{pvalue}_{\mathrm{Wilson}}(k|n, p = p_0)$ との違うは右辺の cdf の引数の分母の p_0 がその推定値 $\hat{p}$ に置き換えられていることである. これは粗い近似になっているが, その分だけ信頼区間の計算を易しくしてくれる.

5.2 Waldの信頼区間

$z_{\alpha/2} = z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2)$ とおく.

In [74]:
```
var"z_{0.025}の定義"
```

Out[74]:

この場合の信頼区間 $\{\, p \in [0,1] \mid \mathrm{pvalue}_{\mathrm{Wald}}(k|n,p) \geq \alpha \,\}$ は次のように表される:

$$\mathrm{confint}_{\mathrm{Wald}}(k|n, p = p_0) := \left[\hat{p} - z_{\alpha/2}\widehat{\mathrm{SE}},\ \hat{p} + z_{\alpha/2}\widehat{\mathrm{SE}}\right].$$

これを **Waldの信頼区間** と呼ぶ.

前節の公式と比較すれば, n を大きくすると, これはWilsonの信頼区間をよく近似するようになることがわかる.

Waldの信頼区間の導出については, [「例：ベータ函数と二項分布の関係とその応用」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)の「Waldの信頼区間の基礎」の節も参照せよ.

5.3 Waldの信頼区間の図による把握

以下の図をよく見れば, $\alpha = 5\%$ の場合の $p_L = p_L$ と $p_U = p_U$ を直観的に把握できるだろう.

まず, データの数値が与えられたとき, モデルのパラメータ値をP値に対応させる **P値函数** (P-value function)の立場では, **信頼区間** (confidence interval)の解釈は易しい.

信頼区間はP値函数が有意水準の値を取る2つのパラメータ値に挟まれた区間になる.

しかし, P値函数全体のグラフを描けば, 信頼区間を計算しなくても, データとの整合性が低過ぎないパラメータ値の範囲は明らかだろう.

In [75]:
```
plot_binom_pvalue_and_confint(
    pvalue_wald, confint_wald, 100, 30;
    name = "Wald: ", xlim=(0.15, 0.5), xtick=0:0.05:1)
```

Out[75]:

P値函数を経由せずに信頼区間の計算の仕方を直接的に理解するには以下のように考えればよい.

$1-\alpha=95\%$ 信頼区間の下限を求めるためには, 二項分布モデルの成功確率パラメータ p を小さくして行って, その二項分布を近似する正規分布内でデータの数値 k 以上の値が生成される確率とその反対側の確率の和(＝Waldの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になるような p を見つければよい. そのような p を $p_L=p_L$ とする.

ただし, Waldの信頼区間の場合にはWilsonの信頼区間の場合と違って, 二項分布を正規分布で近似するときには, 正規分布の分散をデータの数値から推定したものに固定する. そのことが原因でWaldの信頼区間の場合には二項分布の正規分布近似の誤差は大きめになる. (しかし, k と $n-k$ が十分に大きな場合にはその粗い近似で実用的には十分である.)

In [76]:
```
var"Wald: 下側では p = p_L がぎりぎり"
```

Out[76]:

$1-\alpha=95\%$ 信頼区間の上限を求めるためには, 二項分布モデルの成功確率パラメータ p を大きくして行って, その二項分布を近似する正規分布内でデータの数値 k 以下の値が生成される確率とその反対側の確率の和(＝Waldの信頼区間を与えるP値)がちょうど $\alpha=5\%$ になるような p を見つければよい. そのような p を $p_U=p_U$ とする.

ただし, 上でも説明したように, Waldの信頼区間の場合にはWilsonの信頼区間の場合と違って, 二項分布を正規分布で近似するときには, 正規分布の分散をデータの数値から推定したものに固定する. そのことが原因で二項分布の正規分布近似の誤差は大きくなる.

In [77]:
```
var"Wald: 上側では p = p_U がぎりぎり"
```

Out[77]:

以上をまとめると以下のような図を描ける.

In [78]:
```
var"Wald: n × 95%信頼区間"
```

Out[78]:

二項分布のギザギザした2つの山の高さと幅は揃っていないが, Waldの信頼区間の場合には, 正規分布近似おける分散をデータの数値からの推定値で固定してしまうので, 正規分布の2つの山の高さと幅は同じになってしまう.

この場合のWaldの信頼区間は $[0.2102, 0.3998]$ である.

これは, 同じ場合のClopper-Pearsonの信頼区間の $[0.2124, 0.3998]$ とSterneの信頼区間は $[0.2142, 0.3996]$ とWilsonの信頼区間は $[0.2189, 0.3958]$ のどれよりも少し広くなっているが, ほぼ同じであると言ってよいだろう.

ただし, Waldの信頼区間は $\hat{p}$ または $1-\hat{p}$ が小さい場合には誤差が大きくなる場合があるので注意が必要である.

5.4 必修問題: 大阪都構想に関する住民投票の結果から得られる信頼区間達

2015年と2020年の大阪都構想に関する住民投票の結果は

- 2015年: 賛成: 694,844 (49.6%)　反対: 705,585 (50.4%)
- 2020年: 賛成: 675,829 (49.4%)　反対: 692,996 (50.6%)

であった. これらに対応するデータ「n 人中 k 人賛成」は

- 2015年: $(n, k) = (694844 + 705585, 694844)$

- 2020年: $(n, k) = (675829 + 692996, 675829)$

であるとみなし, 統計モデルとして二項分布を考え, 信頼度を $1 - \alpha = 0.999 = 99.9\%$ (有意水準は $\alpha = 0.001 = 0.1\%$) と設定する. これらについて以下を求めよ.

(1) 99.9% Waldの信頼区間

(2) 99.9% Wilsonの信頼区間

(3) 99.9% Clopper-Pearsonの信頼区間

(4) 99.9% Sterneの信頼区間

数値は小数点以下第4桁まで求めよ.

ただし, 最後の(4)だけは必修問題から外す. 自力で求めることができた人は相当に立派だと思われる.

結果の解釈の仕方: 信頼区間から外れている値の全体は, 有意水準 $\alpha = 0.1\%$ の下で, 「賛成確率は $p = p_0$ である」という仮説下での二項分布モデルとデータの数値が余りにも整合性がないと判定されたパラメータ値 p_0 の全体の集合になる. 信頼区間に含まれる p_0 の値はモデルとデータの整合性の観点から棄却されていない値になる.

解答例

(1) 99.9% Waldの信頼区間

- 2015年: $[0.4948, 0.4976]$
- 2020年: $[0.4923, 0.4951]$

(2) 99.9% Wilsonの信頼区間

- 2015年: $[0.4948, 0.4976]$
- 2020年: $[0.4923, 0.4951]$

(3) 99.9% Clopper-Pearsonの信頼区間

- 2015年: $[0.4948, 0.4976]$
- 2020年: $[0.4923, 0.4951]$

(4) 99.9% Sterneの信頼区間

- 2015年: $[0.4948, 0.4976]$
- 2020年: $[0.4923, 0.4951]$

解答終

注意: 小数点以下第4桁までの比較では4種の信頼区間はすべて同じになる! このようになる理由は二項分布の中心極限定理である. これら4種の信頼区間の中でWaldの信頼区間の計算が最も初等的で易しい. ここまで一致するのであれば, Waldの信頼区間の計算で済ませた方が得だと考えられる.

結果の解釈: 例えば, 99.9% 信頼区間に 0.5 以上の値が含まれていないので, 「賛成確率 p は 0.5 以上である」であるという仮説下の二項分布モデルは大阪都構想に関する住民投票の結果のデータとは, 有意水準 0.1% の閾値で余りにも整合していないと判断される.

警告: 99.9% 信頼区間の 99.9% は整合性の指標であるP値が「低すぎる」と判断するときの閾値(有意水準)を 0.1% に設定したことを意味し, それ以上の意味を持たないことに注意せよ.

5.4.1 Julia言語による計算の仕方

具体的な実装についてはこのノートの最初の方にある函数の定義を見て欲しい.

部分的にR言語の計算結果とも比較してみる.

In [79]:
```
α = 0.001
(n2015, k2015) = (694844 + 705585, 694844)
(n2020, k2020) = (675829 + 692996, 675829)
@rput n2015 k2015 n2020 k2020
r4(x) = round(x; digits=4)
```

Out[79]:
```
r4 (generic function with 1 method)
```

In [80]:

```
# (1) 99.9% Waldの信頼区間
@show confint_wald(n2015, k2015; α)
@show confint_wald(n2020, k2020; α)
println()
@show confint_wald(n2015, k2015; α) .▷ r4
@show confint_wald(n2020, k2020; α) .▷ r4;
```

```
confint_wald(n2015, k2015; α) = [0.4947748562481919, 0.49755535113825894]
confint_wald(n2020, k2020; α) = [0.49232315744006405, 0.4951354293040413]

confint_wald(n2015, k2015; α) .▷ r4 = [0.4948, 0.4976]
confint_wald(n2020, k2020; α) .▷ r4 = [0.4923, 0.4951]
```

In [81]:

```
# (2) 99.9% Wilsonの信頼区間
@show confint_wilson(n2015, k2015; α)
@show confint_wilson(n2020, k2020; α)
println()
@show confint_wilson(n2015, k2015; α) .▷ r4
@show confint_wilson(n2020, k2020; α) .▷ r4;
```

```
confint_wilson(n2015, k2015; α) = [0.4947748912719356, 0.49755537541387646]
confint_wilson(n2020, k2020; α) = [0.4923232126021383, 0.495135473345231]

confint_wilson(n2015, k2015; α) .▷ r4 = [0.4948, 0.4976]
confint_wilson(n2020, k2020; α) .▷ r4 = [0.4923, 0.4951]
```

In [82]:

```
@show R"prop.test(k2015, n2015, conf.level=0.999, correct=F)$conf.int"
@show R"prop.test(k2020, n2020, conf.level=0.999, correct=F)$conf.int";
```

```
R"prop.test(k2015, n2015, conf.level=0.999, correct=F)$conf.int" = RObject{RealSxp}
[1] 0.4947749 0.4975554
attr(,"conf.level")
[1] 0.999

R"prop.test(k2020, n2020, conf.level=0.999, correct=F)$conf.int" = RObject{RealSxp}
[1] 0.4923232 0.4951355
attr(,"conf.level")
[1] 0.999
```

In [83]:

```
# (3) 99.9% Clopper-Pearsonの信頼区間
@show confint_clopper_pearson(n2015, k2015; α)
@show confint_clopper_pearson(n2020, k2020; α)
println()
@show confint_clopper_pearson(n2015, k2015; α) .▷ r4
@show confint_clopper_pearson(n2020, k2020; α) .▷ r4;
```

```
confint_clopper_pearson(n2015, k2015; α) = [0.49477452283892703, 0.49755572592143626]
confint_clopper_pearson(n2020, k2020; α) = [0.4923228298103796, 0.49513582614939644]

confint_clopper_pearson(n2015, k2015; α) .▷ r4 = [0.4948, 0.4976]
confint_clopper_pearson(n2020, k2020; α) .▷ r4 = [0.4923, 0.4951]
```

In [84]:

```
@show R"binom.test(k2015, n2015, conf.level=0.999)$conf.int"
@show R"binom.test(k2020, n2020, conf.level=0.999)$conf.int";
```

```
R"binom.test(k2015, n2015, conf.level=0.999)$conf.int" = RObject{RealSxp}
[1] 0.4947745 0.4975557
attr(,"conf.level")
[1] 0.999

R"binom.test(k2020, n2020, conf.level=0.999)$conf.int" = RObject{RealSxp}
[1] 0.4923228 0.4951358
attr(,"conf.level")
[1] 0.999
```

In [85]:

```
# (4) 99.9% Sterneの信頼区間
@show confint_sterne(n2015, k2015; α)
@show confint_sterne(n2020, k2020; α)
println()
@show confint_sterne(n2015, k2015; α) .▷ r4
@show confint_sterne(n2020, k2020; α) .▷ r4;
```

```
confint_sterne(n2015, k2015; α) = [0.494774805693787, 0.4975553874009835]
confint_sterne(n2020, k2020; α) = [0.4923229630950927, 0.4951355997338051]

confint_sterne(n2015, k2015; α) .▷ r4 = [0.4948, 0.4976]
confint_sterne(n2020, k2020; α) .▷ r4 = [0.4923, 0.4951]
```

5.4.2 WolframAlphaによる計算の仕方

(4)を除いて計算してみよう.

上のJulia言語を用いた計算法は筆者が書いたコードに依存しているので, それを真似たとしても理解に繋がらない危険性がある. WolframAlphaを使って素朴に計算した場合には各ステップが明瞭に見えているので, 何をやっているかが見易い.

$z_{\alpha/2} = z_{0.0005}$ の計算:

`quantile(NormalDistribution(0,1), 0.9995)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.9995%29) → 3.29053

(1) 99.9% Waldの信頼区間

この場合には $z_{\alpha/2}$ が既知ならば, 残りの計算は四則演算と平方根だけなので電卓でも計算できる.

2015年:

`{k/n - z*sqrt(k/n(1-k/n)/n), k/n + z*sqrt(k/n(1-k/n)/n)} where z = 3.29053, n = 694844 + 705585, k = 694844` → 実行 (https://www.wolframalpha.com/input?i=%7Bk%2Fn+-+z*sqrt%28k%28n-k%29%2Fn%5E3%29%2C+k%2Fn+%2B+z*sqrt%28k%28n-k%29%2Fn%5E3%29%7D+where+z+%3D+3.29053%2C+n+%3D+694844+%2B+705585%2C+k+%3D+694844) → {0.494775, 0.497555}

2020年:

`{k/n - z*sqrt(k/n(1-k/n)/n), k/n + z*sqrt(k/n(1-k/n)/n)} where z = 3.29053, n = 675829 + 692996, k = 675829` → 実行 (https://www.wolframalpha.com/input?i=%7Bk%2Fn+-+z*sqrt%28k%2Fn%281-k%2Fn%29%2Fn%29%2C+k%2Fn+%2B+z*sqrt%28k%2Fn%281-k%2Fn%29%2Fn%29%7D+where+z+%3D+3.29053%2C+n+%3D+675829+%2B+692996%2C+k+%3D+675829) → {0.492323, 0.495135}

(2) 99.9% Wilsonの信頼区間

この場合には二次方程式を解く必要がある. この計算も $z_{\alpha/2}$ さえ既知ならば, 残りの計算は四則演算と平方根だけなので電卓でも計算できる.

2015年:

`solve (1+z^2/n) p^2 - 2(k/n+z^2/(2n)) p + (k/n)^2 = 0 where z = 3.29053, n = 694844 + 705585, k = 694844` → 実行 (https://www.wolframalpha.com/input?i=solve+%281%2Bz%5E2%2Fn%29+p%5E2+-+2%28k%2Fn%2Bz%5E2%2F%282n%29%29+p+%2B+%28k%2Fn%29%5E2+%3D+0+where+z+%3D+3.29053%2C+n+%3D+694
→ p == 0.494775 || p == 0.497555

2020年: `solve (1+z^2/n) p^2 - 2(k/n+z^2/(2n)) p + (k/n)^2 = 0 where z = 3.29053, n = 675829 + 692996, k = 675829` → 実行 (https://www.wolframalpha.com/input?i=solve+%281%2Bz%5E2%2Fn%29+p%5E2+-+2%28k%2Fn%2Bz%5E2%2F%282n%29%29+p+%2B+%28k%2Fn%29%5E2+%3D+0+where+z+%3D+3.29053%2C+n+%3D+675
→ p == 0.492323 || p == 0.495135

(3) 99.9% Clopper-Pearsonの信頼区間

この場合にはベータ分布の分位点函数(quantile function)が必要になる.

2015年:

`quantile(BetaDistribution(694844, 705585+1), 0.0005)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28694844%2C+705585%2B1%29%2C+0.0005%29) → 0.494775

`quantile(BetaDistribution(694844+1, 705585), 0.9995)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28694844%2B1%2C+705585%29%2C+0.9995%29) → 0.497556

2020年:

`quantile(BetaDistribution(675829, 692996+1), 0.0005)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28675829%2C+692996%2B1%29%2C+0.0005%29) → 0.492323

`quantile(BetaDistribution(675829+1, 692996), 0.9995)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%28675829%2B1%2C+692996%29%2C+0.9995%29&lang=ja) → 0.495136

In [ ]:

```
1
```

検定と信頼区間: 平均の検定と信頼区間

- 黒木玄

- 2022-05-31～2022-06-18

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 平均に関するP値と信頼区間
  - 1.1 平均に関するP値と信頼区間を使って行いたいこと
  - 1.2 平均の検定で使用されるP値の定義(1) 中心極限定理経由で標準正規分布を使う場合
  - 1.3 標本平均と不偏分散に関する中心極限定理と大数の法則の可視化
  - 1.4 問題: 他の様々な分布について T(μ) の分布を確認
  - 1.5 P値の定義(1)の標準正規分布を使う場合に対応する平均の信頼区間
  - 1.6 平均の検定で使用されるP値の定義(2) さらに t 分布を使う場合
  - 1.7 標本分布におけるT統計量の分布の視覚化
  - 1.8 分布の対数の標本分布のT統計量の視覚化
  - 1.9 平均に関するP値の使い方
  - 1.10 P値の定義(2)のt分布を使う場合に対応する平均の信頼区間
- 2 平均に関するP値と信頼区間の計算例
  - 2.1 平均に関するP値と信頼区間の計算の仕方
  - 2.2 平均に関するP値と信頼区間の計算例
    - 2.2.1 WolframAlphaでの平均に関するP値と信頼区間の計算例
    - 2.2.2 Julia言語での平均に関するP値と信頼区間の計算例
    - 2.2.3 Julia言語での平均に関するP値と信頼区間のグラフ作成
    - 2.2.4 R言語での平均に関するP値と信頼区間の計算例
  - 2.3 必修練習問題: 平均に関するP値と信頼区間の計算
    - 2.3.1 WaolframAlphaによる必修練習問題の解き方
    - 2.3.2 Julia言語による必修練習問題の解き方
    - 2.3.3 R言語による必修練習問題の解き方
    - 2.3.4 必修練習問題解答例

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Printf
using QuadGK
using RCall
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : x*y
safediv(x, y) = x == 0 ? x : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

In [4]:

```julia
pdflog(dist, x) = exp(logpdf(dist, exp(x)) + x)

@memoize function logmean(dist; a=-1e2, b=1e2)
    quadgk(x -> x * pdflog(dist, x), a, b)[1]
end

@memoize function logvar(dist; a=-1e2, b=1e2)
    μ = logmean(dist; a, b)
    quadgk(x -> (x - μ)^2 * pdflog(dist, x), a, b)[1]
end
logstd(dist; a=-1e2, b=1e2) = √logvar(dist; a, b)

function plot_T(dist, n; L=10^6,
        plot_chi=false, plot_t=false, logsample=false, kwargs...)
    if logsample
        μ, σ = logmean(dist), logstd(dist)
    else
        μ, σ = mean(dist), std(dist)
    end
    println("skewness = ", myskewness(dist), ",  kurtosis = ", mykurtosis(dist))

    se = σ/√n
    Z = Vector{Float64}(undef, L)
    S² = Vector{Float64}(undef, L)
    T = Vector{Float64}(undef, L)
    tmp = [Vector{eltype(dist)}(undef, n) for _ in 1:nthreads()]
    @threads for i in 1:L
        X = rand!(dist, tmp[threadid()])
        if logsample
            @. X = log(X)
        end
        X̄ = mean(X)
        Z[i] = (X̄ - μ)/se
        S²[i] = var(X)
        T[i] = √n * (X̄ - μ)/√S²[i]
    end

    xlim = (-4.0, 5.9)
    bin = range(-6, 6, 150)
    P = stephist(Z; norm=true, xlim, bin, label="Z(μ)")
    plot!(Normal(), xlim...; label="Normal(0,1)", ls=:dash)
    plot!(; xtick=-10:10)

    W = .√(S²/σ^2)
    xlim = (0, 3)
    bin = range(quantile.(Ref(W), (0.0, 0.999))..., 150)
    Q = stephist(W; norm=true, xlim, bin, label="√(S²/σ²)")
    if plot_chi
        plot!(Chi(n-1)/√(n-1), xlim...; label="Chi(n-1)/√(n-1)", ls=:dash)
    else
        vline!([1]; label="", ls=:dash)
    end
    plot!(; xtick=range(xlim...; step=0.5))

    xlim = (-4.0, 5.9)
    bin = range(-6, 6, 150)
    R = stephist(T; norm=true, xlim, bin, label="T(μ)")
    if plot_t
        plot!(TDist(n-1), xlim...; label="TDist(n-1)", ls=:dash)
    else
        plot!(Normal(), xlim...; label="Normal(0,1)", ls=:dash)
    end
    plot!(; xtick=-10:10)

    plot(P, Q, R; size=(800, 200), layout=(1, 3))
    plot_title = if logsample
        "Log of $(distname(dist)), n=$n"
    else
        "$(distname(dist)), n=$n"
    end
    plot!(; plot_title, plot_titlefontsize=10)
    plot!(; kwargs...)
end

#plot_T(Normal(2, 3), 10; plot_chi=true, plot_t=true)
#plot_T(Gamma(2, 3), 10; plot_chi=true, plot_t=true)
#plot_T(Gamma(2, 3), 10; plot_chi=true, plot_t=true, logsample=true)
```

Out[4]: plot_T (generic function with 1 method)

In [5]:

```
function pvalue_tdist(x̄, s², n, μ)
    t = (x̄ - μ)/√(s²/n)
    2ccdf(TDist(n-1), abs(t))
end

function pvalue_tdist(x, μ)
    x̄, s², n = mean(x), var(x), length(x)
    pvalue_tdist(x̄, s², n, μ)
end

function confint_tdist(x̄, s², n; α = 0.05)
    c = quantile(TDist(n-1), 1-α/2)
    [x̄ - c*√(s²/n), x̄ + c*√(s²/n)]
end

function confint_tdist(x; α = 0.05)
    x̄, s², n = mean(x), var(x), length(x)
    confint_tdist(x̄, s², n; α)
end

function show_confint_of_mean(x; μ = mean(x), α = 0.05)
    println("(P-value of hypothesis \"mean = $μ\") = ", pvalue_tdist(x, μ))
    println("($(100(1-α))% confidence interval of mean) = ", confint_tdist(x; α))
end

function plot_confint_of_mean(x; α = 0.05,
        xlim = nothing,
        plot_pvaluefunc = false,
        xtick=-100:100, kwargs...)
    n = length(x)
    if isnothing(xlim)
        a, b = extrema(x)
        a, b = a - 0.05(b-a), b + 0.05(b-a)
    else
        a, b = xlim
    end
    confidence_interval = confint_tdist(x; α)
    if plot_pvaluefunc
        scatter(x, fill(-0.05, n); label="", msc=:auto, alpha=0.7)
        plot!(μ -> pvalue_tdist(x, μ), a, b; label="", c=2)
        plot!(confidence_interval, fill(α, 2); label="", lw=3, c=:red)
        plot!(; ylim=(-0.1, 1.02), ytick=0:0.1:1, yguide="P-value")
        title!("P-value function and $(100(1-α))% conf. int. of mean for data of size n=$n")
        plot!(; size=(600, 200), leftmargin=4Plots.mm)
    else
        scatter(x, fill(-0.05, n); label="", msc=:auto, alpha=0.7)
        plot!(confidence_interval, fill(0.06, 2); label="", lw=6, c=:red)
        plot!(; ylim=(-0.1, 0.12), yaxis=false, ytick=false)
        title!("$(100(1-α))% confidence interval of mean for data of size n=$n")
        plot!(; size=(600, 60))
    end
    plot!(; xlim=(a, b), xtick)
    plot!(; kwargs...)
end
```

Out[5]: plot_confint_of_mean (generic function with 1 method)

In [6]:
```julia
#Random.seed!(7749); x = rand(Normal(0, 1), 20) # Julia v1.7.3
x = [
-0.14468070220160034
-0.18969247015207036
 0.30151049054895623
-0.18480852071580925
 0.5445553231881434
 0.429751332626428
 2.044565039620567
-0.3214242933678388
 0.10607974695801257
-0.2046040127070169
-1.187351237918782
 0.3615628094981996
 0.42388796016145336
 0.6223382190805758
 1.4389043328536288
 1.300418561526944
 0.9756607771970455
 0.29276870729869514
 1.685408534649871
-0.23026435459236955
];
```

In [7]:
```julia
show_confint_of_mean(x; μ = 1, α = 0.05)
plot_confint_of_mean(x; α = 0.05, plot_pvaluefunc=true, size=(600, 300))
```

```
(P-value of hypothesis "mean = 1") = 0.00285040897858587
(95.0% confidence interval of mean) = [0.038356469299347495, 0.7681021550559557]
```

Out[7]:

上のグラフの説明

薄い青のドット達は, 標準正規分布に従う乱数で生成したサイズ $n = 20$ の標本の数値(データの数値とも呼ぶ).

青くて太い線は, 標本の数値から計算される平均の 95% 信頼区間.

「とんがり帽子」型の曲線は, そのデータの数値から決まる平均に関するP値函数のグラフ.

横軸が $\mu = \mu_0$ の数値で, そのP値函数は t 分布を使って定義されている.

生成されたデータの数値は大幅に右側に偏っていた.

データを生成した標準正規分布の平均は 0 なのに, は右側に大きくずれてしまっている.

以下の目標

- 上のグラフの意味を理解できるようになる.
- 平均に関するP値と信頼区間を計算できるようになる.

1 平均に関するP値と信頼区間

1.1 平均に関するP値と信頼区間を使って行いたいこと

以下のようなことを行いたい.

(1) S市の中学3年生男子達から n 人を無作為抽出して身長を測って得た数値のデータ $x_1, \ldots, x_n$ から, S市の中学3年生男子達全員分の身長の平均値を推定したい.

(2) とある店で出されるフライドポテトの長さを n 本分測って得た数値のデータ $x_1, \ldots, x_n$ から, その店で出されるフライドポテトの長さの平均値を推定したい.

このような推定を以下では **平均の推定** と呼ぶことにする.

目標は平均の信頼区間の構成である.

そのためには, 検定と信頼区間の表裏一体性より, P値を適切に定義すればよい.

1.2 平均の検定で使用されるP値の定義(1) 中心極限定理経由で標準正規分布を使う場合

データ: n 個の実数値 $x_1, \ldots, x_n$.

データの標本平均と不偏分散をそれぞれ $\bar{x}$, s^2 と書く:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})\,.$$

統計モデル: 平均 μ と分散 σ^2 を持つ未知の確率分布 D のサイズ n の標本分布 D^n を統計モデルとして採用する.

簡単のため D は連続分布であると仮定し, その確率密度函数を $p(x)$ と書く. このとき標本分布 D^n の確率密度函数は次のように表される:

$$p(x_1, \ldots, x_n) = p(x_1)\cdots p(x_n).$$

この設定は, データ $x_1, \ldots, x_n$ がある母集団からの無作為抽出で得られた数値の列である場合に適切になる.

未知だと想定されている分布 D は母集団の分布モデル化になっていると考えている.

例えば, S市の中学3年生男子達から n 人を無作為抽出して身長を測って得た数値のデータ $x_1, \ldots, x_n$ を用いた統計分析を行う場合には, モデルの確率分布 D はS市の中学3年生男子達全員分の身長の数値の分布を近似していることを期待している.

仮に, S市の中学3年生男子達全員分の身長の数値の分布がすでに分かっているならば, 平均身長を推定するために無作為抽出を行う必要はない. 実際には未知であるからこそ無作為抽出で得たデータを使って統計分析をする必要が出て来る. だから, モデルの分布 D は「未知の分布」であることにしておきたい.

確率分布 D が未知であるという設定は, 分布 D をある特定の分布(例えば正規分布)であると考えるのではなく, (適当な緩い条件を満たす)任意の分布であると考えることを意味している.

分布の平均 μ, 分布の分散 σ^2 と無限個のパラメータ $\theta_3, \theta_4, \ldots$ を使って任意の分布を $D(\mu, \sigma, \theta_3, \theta_4, \ldots)$ とパラメトライズすると, この場合の統計モデルはこの無限個のパラメータを持つ確率分布 $D(\mu, \sigma^2, \theta_3, \theta_4, \ldots)$ であるということになる.

ただし, 以下の問題設定で興味があるのは平均パラメータ μ だけであり, 残りの $\sigma^2, \theta_3, \theta_4, \ldots$ は平均 μ に関する統計分析を邪魔するニューサンスパラメータ(nuisance parameters, 局外パラメータ, 攪乱パラメータ, 迷惑パラメータ)とみなされる.

ニューサンスパラメータの問題は二項分布モデルを使った比率の検定や信頼区間にはなかったので, これは新しい問題である.

以下では中心極限定理を使ってニューサンスパラメータの問題を回避する.

検定したい仮説: $\mu = \mu_0$ (μ_0 は具体的な数値).

$X_1, \ldots, X_n$ は統計モデル D^n に従う確率変数達であるとする. すなわち, $X_1, \ldots, X_n$ は独立な確率変数達であり, 各々がモデルの確率分布 D に従っていると仮定する.

標本平均に関する中心極限定理: 中心極限定理によって, n が十分に大きいならば, モデル内でランダムに生成された仮想的標本 $X_1, \ldots, X_n$ の標本平均

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

は平均 μ, 分散 σ^2/n の正規分布に近似的に従う:

$$\bar{X} \sim \mathrm{Normal}\left(\mu, \sqrt{\sigma^2/n}\right), \text{ approximately.}$$

これを使えると以下では仮定する. (D ごとに n が十分に大きいと仮定する.)

不偏分散に関する大数の法則: 大数の法則より, n を十分に大きくすると, モデルの確率分布 D の分散 σ^2 の値は, モデル内でランダムに生成された仮想的標本 $X_1, \ldots, X_n$ の不偏分散(不偏推定量になるように補正された標本分散)

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)$$

で近似される:

$$S^2 \approx \sigma^2.$$

検定で使われる T 統計量:　以上の状況の下で, T 統計量を

$$T(\mu) := \frac{\bar{X}-\mu}{\sqrt{S^2/n}} \approx \frac{\bar{X}-\mu}{\sqrt{\sigma^2/n}} \sim \mathrm{Normal}(0,1),\ \text{approximately}.$$

と定める. このとき $T(\mu)$ は近似的に標準正規分布に従う:

$$T(\mu) \sim \mathrm{Normal}(0,1),\ \text{approximately}.$$

データの t 値を次のように定める:

$$t(\mu) := \frac{\bar{x}-\mu}{\sqrt{s^2/n}}.$$

中心極限定理によって $(\bar{X}-\mu)/\sqrt{\sigma^2/n}$ という無限個のパラメータ $\theta_3, \theta_4, \ldots$ が含まれない統計量を利用できるようになり, さらに大数の法則より, S^2 で σ^2 を近似することによって, σ^2 も含まらない統計量 $T(\mu)$ を利用できるようになった. これが中心極限定理と大数の法則を使ったニューサンスパラメータを回避する方法である.

P値の定義:　仮説 $\mu = \mu_0$ 下の統計モデル内で $|T(\mu_0)| \geq |t(\mu_0)|$ となる確率を, 標準正規分布に従ってランダムに生成される値の絶対値が $|t(\mu_0)|$ 以上になる確率として近似的に求めて, その値をデータ $x_1, \ldots, x_n$ に関する仮説 $\mu = \mu_0$ のP値として採用する. そのP値を次のように書く:

$$\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, s^2 | n, \mu = \mu_0) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |t(\mu_0)|)).$$

1.3 標本平均と不偏分散に関する中心極限定理と大数の法則の可視化

前節のP値の定義では, 中心極限定理によって, 平均 μ, 分散 σ^2 を持つ確率分布 D のサイズ n の標本 $X_1, \ldots, X_n$ (独立同分布な確率変数達になる)の標本平均 $\bar{X}$ (確率変数になる)が, n が大きなとき, 平均 μ, 分散 σ^2/n を持つ正規分布に **近似的に** 従うという結果(中心極限定理)を使った.

そしてさらに, 大数の法則によって, n を大きくすると, $X_1, \ldots, X_n$ の不偏分散が σ を近似するようになることも使った.

前節のP値に定義が意味を持つためには, $T(\mu) = (\bar{X}-\mu)/\sqrt{S^2/n}$ が近似的に標準正規分布に従うことが必要である.

以下でそれを様々な確率分布 D について確認してみよう.

以下では, 平均 μ と分散 σ^2 を持つ様々な確率分布 D の標本分布 D^n における以下の3つの確率変数の分布のグラフを描く:

$$Z(\mu) = \frac{\bar{X}-\mu}{\sqrt{\sigma^2/n}}, \quad \sqrt{\frac{S^2}{\sigma^2}}, \quad T(\mu) = \frac{\bar{X}-\mu}{\sqrt{S^2/n}} = \frac{Z(\mu)}{\sqrt{S^2/\sigma^2}}.$$

n を大きくして行くと, 大数の法則より $\sqrt{S^2/\sigma^2}$ の分布は 1 に集中して行き, 中心極限定理より $Z(\mu)$ の分布は標準正規分布に近付いて行く.

特に, 確率分布 D の種類ごとに, $T(\mu)$ の分布の標準正規分布による近似の誤差の大きさが違うことに注目せよ.

正規分布の場合

In [8]:
```
plot(Normal(), -4.5, 4.5; label="Normal(0,1)")
```

Out[8]:

In [9]:
```
plot_T(Normal(), 40)
```

```
skewness = 0.0,  kurtosis = 0.0
```

Out[9]:

一様分布の場合

In [10]:
```
plot(Uniform(), -0.3, 1.3; label="Uniform(0,1)", ylim=(-0.1, 1.3))
```

Out[10]:

In [11]:
```
plot_T(Uniform(), 40)
```

```
skewness = 0.0,  kurtosis = -1.2
```

Out[11]:

Laplace分布の場合

In [12]:
```
plot(Laplace(), -4.5, 4.5; label="Laplace(0,1)")
```

Out[12]:

In [13]:
```
plot_T(Laplace(), 40)
```

```
skewness = 0.0,  kurtosis = 3.0
```

Out[13]:

ガンマ分布の場合

ガンマ分布は左右非対称な分布の典型例である.

In [14]:
```
plot(Gamma(2,3); label="Gamma(2,3)")
```

Out[14]:

In [15]:
```
plot_T(Gamma(2,3), 40)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[15]:

In [16]:
```
plot_T(Gamma(2,3), 160)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[16]:

In [17]:
```
plot_T(Gamma(2,3), 640)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[17]:

対数正規分布の場合

対数正規分布は左右非対称でかつ **外れ値が出易い** 分布の典型例である.

このような場合は n を非常に大きくしないと近似の精度が高まらない.

In [18]:
```
plot(LogNormal(), -0.1, 10; label="LogNormal(0,1)")
```

Out[18]:

In [19]:
```
plot_T(LogNormal(), 40)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[19]:

In [20]:
```
plot_T(LogNormal(), 160)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[20]:

In [21]:
```
plot_T(LogNormal(), 640)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[21]:

In [22]:
```
plot_T(LogNormal(), 2560)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[22]:

この例から, 左右非対称で外れ値を含む母集団分布からの無作為抽出で得たデータを扱う場合には, 中心極限定理を使った近似を用いることは要注意であることがわかる.

注意: $X \sim D$ と仮定する. 一般に, 分布 D の歪度(わいど, skewness)と尖度(せんど, kurtosis, 過剰尖度, excessive kurtosis)をそれぞれ

$$\bar{\kappa}_3 = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right], \quad \bar{\kappa}_4 = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] - 3$$

と書く. (D が正規分布の場合には $\bar{\kappa}_3 = \bar{\kappa}_4 = 0$ となる.)

このとき, $\bar{\kappa}_4 \geq \bar{\kappa}_3^2 - 2$ および以下が成立することを示せる:

$$\begin{bmatrix} E[\bar{X}] \\ E[S^2] \end{bmatrix} = \begin{bmatrix} \mu \\ \sigma^2 \end{bmatrix},$$
$$\begin{bmatrix} \mathrm{var}(\bar{X}) & \mathrm{cov}(\bar{X}, S^2) \\ \mathrm{cov}(\bar{X}, S^2) & \mathrm{var}(S^2) \end{bmatrix} = \frac{1}{n}\begin{bmatrix} \sigma^2 & \sigma^3\bar{\kappa}_3 \\ \sigma^3\bar{\kappa}_3 & \sigma^4(\bar{\kappa}_4 + 2/(1-1/n)) \end{bmatrix}.$$

そして, 多変量分布の中心極限定理(これはこのノート群では証明していないが, 1変量の場合と同じ方法で証明される)によって, n が大きなとき, $\bar{X}, S^2$ の同時分布は, それらを平均と分散共分散行列に持つ2変量正規分布で近似される. $n \to \infty$ で分散共分散行列が 0 に収束することから, $(\bar{X}, S^2)$ の分布が (μ, σ^2) に集中して行くこともわかる(大数の法則).

この一般的な結果の視覚化については[「標本分布について」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)を参照せよ. そこで詳しく解説されている結果と視覚化は「平均の検定や信頼区間」のような統計学における基本を理解するためにかなり重要なことなのだが, 入門的な教科書ではほとんど説明されていない. 理解を深めるために貴重な解説になっていると思われるので是非とも繰り返し参照して欲しい.

1.4 問題: 他の様々な分布について T(μ) の分布を確認

上の例以外の分布についても $T(\mu)$ の分布がどうなっているかを確認せよ.

ヒント: [「標本分布について」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)を参照せよ.

次節で使う $z_{\alpha/2}$ の値の例: $\alpha = 5\%, 1\%, 0.1\%$ のそれぞれについて,

In [23]:
```
@. quantile(Normal(), 1 - (0.05, 0.01, 0.001) / 2)
```
Out[23]:
```
(1.9599639845400576, 2.5758293035489053, 3.290526731491931)
```

In [24]:
```
@. round(quantile(Normal(), 1 - (0.05, 0.01, 0.001) / 2); digits=4)
```
Out[24]:
```
(1.96, 2.5758, 3.2905)
```

1.5 P値の定義(1)の標準正規分布を使う場合に対応する平均の信頼区間

有意水準を $0 \le \alpha \le 1$ と書き, 標準正規分布において $z_{\alpha/2}$ 以上になる確率は $\alpha/2$ になると仮定する:

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2).$$

例えば,

$$z_{5\%/2} \approx 1.9600, \quad z_{1\%/2} \approx 2.5758, \quad z_{0.1\%/2} \approx 3.2905.$$

P値函数 $\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |t(\mu_0)|))$ に対応する信頼度 $1-\alpha$ の信頼区間は次のようになる:

$$\mathrm{confint}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0) = \left[\bar{x} - z_{\alpha/2}\sqrt{s^2/n},\ \bar{x} + z_{\alpha/2}\sqrt{s^2/n}\right].$$

証明: 検定と信頼区間の表裏一体性より, P値函数 $\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0)$ に対応する信頼度 $1-\alpha$ の信頼区間は次のように定義されるのであった:

$$\mathrm{confint}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0) = \{\, \mu_0 \in \mathbb{R} \mid \mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0) \ge \alpha \,\}.$$

そして,

$$\begin{aligned}
&\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu = \mu_0) \ge \alpha \\
&\iff 1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |t(\mu_0)|)) \ge \alpha/2 \\
&\iff |t(\mu_0)| = \frac{|\bar{x} - \mu_0|}{\sqrt{s^2/n}} \le z_{\alpha/2} \\
&\iff \bar{x} - z_{\alpha/2}\sqrt{s^2/n} \le \mu_0 \le \bar{x} + z_{\alpha/2}\sqrt{s^2/n}.
\end{aligned}$$

これより, P値の定義(1)に対応する平均の信頼区間が上のようになることがわかった.

証明終

1.6 平均の検定で使用されるP値の定義(2) さらに t 分布を使う場合

分布 D が左右対称の分布ならば $n = 10$ のようなかなり小さな n で中心極限定理による近似の誤差は非常に小さくなる場合がある. しかし, そういう場合であっても, 大数の法則を使った不偏分散 S^2 による σ^2 の近似の精度は低いままの場合がある.

そういう場合の補正を t 分布を使って行う処方箋を採用しよう.

統計モデルとして正規分布の標本分布を仮定: 以下では, 統計モデルとして, 平均 μ, 分散 σ^2 を持つ正規分布のサイズ n の標本分布を採用する. その確率密度函数は次のように表される:

$$p(x_1, \ldots, x_n|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \mu)^2\right).$$

$X_1, \ldots, X_n$ はこの統計モデルに従う確率変数達であるとし, それらの標本平均と不偏分散をそれぞれ $\bar{X}$, S^2 と表す.

$\bar{X}$とS^2の同時分布: 標本平均と不偏分散

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad s^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

を使うと,

$$\sum_{i=1}^{n}(x_i-\mu)^2=\sum_{i=1}^{n}((x_i-\bar{x})(\bar{x}-\mu))^2=(n-1)s^2+n(\bar{x}-\mu))^2.$$

なので,

$$p(x_1,\ldots,x_n|\mu,\sigma^2)=\frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2}\frac{(n-1)s^2}{\sigma^2}-\frac{(\bar{x}-\mu)^2}{2\sigma^2/n}\right).$$

このことから, $\bar{X},S^2$ は独立な確率変数であり, $\bar{X}$ は平均 μ, 分散 σ^2/n の正規分布に従い, $(n-1)S^2/\sigma^2$ は自由度 $n-1$ のχ²分布に従うことを示せる:

$$\bar{X}\sim\mathrm{Normal}\left(\mu,\sqrt{\sigma^2/n}\right),\quad \frac{(n-1)S^2}{\sigma^2}\sim\mathrm{Chisq}(n-1).$$

詳しくは「標本分布について」のノートの「正規分布の標本分布の場合」の節を参照せよ. 自由度 $n-1$ のχ²分布の密度函数を作るために必要な, $u=s^2$ とおいたときの因子 $u^{(n-1)/2-1} は

$$dy_1\cdots dy_{n-1}\propto\left(\sqrt{u}\right)^{n-2}d\sqrt{u}\,d\theta\propto u^{(n-1)/2-1}\,du\,d\theta$$

のような計算で出て来る. ここで $d\theta$ は $n-2$ 次元単位球面の微小面積要素である.

T統計量が従う分布: ゆえに, T 統計量

$$T(\mu)=\frac{\bar{X}-\mu}{\sqrt{S^2/n}}$$

は自由度 $n-1$ の t 分布に従う:

$$T(\mu)\sim\mathrm{TDist}(n-1).$$

詳しくは, 「正規分布の標本分布から自然にt分布に従う確率変数が得られること」を参照せよ.

注意: 自由度が大きな t 分布は標準正規分布とほぼ同じになるので, この結果はP値の定義(1)で使った結果と整合的であり, この結果が意味を持つのは n が大きくない場合にのみ意味を持つ. 実践的には n が十分に大きな場合には $\mathrm{TDist}(n-1)$ を標準正規分布で置き換えてよい.

P値の定義: 仮説 $\mu=\mu_0$ の下で $|T(\mu_0)|\geq|t(\mu_0)|$ となる確率を, 自由度 $n-1$ の t 分布に従ってランダムに生成される値の絶対値が $|t(\mu_0)|$ 以上になる確率として正確に求めて, その値をデータ $x_1,\ldots,x_n$ に関する仮説 $\mu=\mu_0$ のP値として採用する. そのP値を次のように書く:

$$\mathrm{pvalue}_{\mathrm{TDist}}(\bar{x},s^2|n,\mu=\mu_0)=2(1-\mathrm{cdf}(\mathrm{TDist}(n-1),|t(\mu_0)|)).$$

このP値は n が大きな場合には定義(1)のP値 $\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x},s^2|n,\mu=\mu_0)$ と近似的に一致する. ゆえに, P値の定義(1)を使うことを止めて, こちらの定義(2)の方だけを使うことにしても害がないと考えられる.

我々はこれ以後こちらのP値の定義のみを使用する.

注意: ただし, こちらの定義(2)のP値は先の定義(1)のP値よりも少し大きくなり, 定義(2)の場合の信頼区間は定義(1)の場合の信頼区間よりも少し広くなる.

In [25]:
```
df = 10:10:100
α = [0.05, 0.01, 0.001]
z = @. round(quantile(Normal(), 1-α'/2); digits=4)
t = @. round(quantile(TDist(df), 1-α'/2); digits=4)
table_of_t = DataFrame(
    df = Any[df; Inf],
    var"α = 5%" = [t[:,1]; z[1]],
    var"α = 1%" = [t[:,2]; z[2]],
    var"α = 0.1%" = [t[:,3]; z[3]],
)
```

Out[25]: 11 rows × 4 columns

	df	α = 5%	α = 1%	α = 0.1%
	Any	Float64	Float64	Float64
1	10	2.2281	3.1693	4.5869
2	20	2.086	2.8453	3.8495
3	30	2.0423	2.75	3.646
4	40	2.0211	2.7045	3.551
5	50	2.0086	2.6778	3.496
6	60	2.0003	2.6603	3.4602
7	70	1.9944	2.6479	3.435
8	80	1.9901	2.6387	3.4163
9	90	1.9867	2.6316	3.4019
10	100	1.984	2.6259	3.3905
11	Inf	1.96	2.5758	3.2905

In [26]:
```
quantile(TDist(30), 1-0.05/2) / quantile(Normal(), 1-0.05/2)
```

Out[26]: 1.0419948899114568

1.7 標本分布におけるT統計量の分布の視覚化

前節では, t 分布を使用することによって, 分布 D が左右対称でおとなしめもしくは正規分布に非常に近い分布でかつ n が小さな場合に精度が高まることを期待してP値を再定義した.

その期待が実際に正しそうなことを確認するために, 以下では再度 $T(\mu)$ の分布の視覚化を行う.

そのとき, 非対称な分布の場合に n を大きくしたとき, t 分布の使用に害がないことも確認しておこう.

さらに, $\sqrt{S^2/\sigma^2}$ の分布と $\sqrt{\mathrm{Chisq}(n-1)/(n-1)}$ の比較も同時に行うことにしよう. ($X \sim \mathrm{Chisq}(\nu)$ のとき $\sqrt{X}$ が従う分布を $\mathrm{Chi}(\nu)$ と書き **χ分布** と呼ぶ.)

以下では, 平均 μ と分散 σ^2 を持つ様々な確率分布 D の標本分布 D^n における以下の3つの確率変数の分布のグラフを描く:

$$Z(\mu) = \frac{\bar{X} - \mu}{\sqrt{\sigma^2/n}}, \quad \sqrt{\frac{S^2}{\sigma^2}}, \quad T(\mu) = \frac{\bar{X} - \mu}{\sqrt{S^2/n}} = \frac{Z(\mu)}{\sqrt{S^2/\sigma^2}}.$$

正規分布の場合

これは理想的(過ぎる)場合である.

In [27]:
```
plot(Normal(), -4.5, 4.5; label="Normal(0,1)")
```

Out[27]:

In [28]:
```
plot_T(Normal(), 10, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = 0.0
```

Out[28]:

一様分布の場合

In [29]:
```
plot(Uniform(), -0.3, 1.3; label="Uniform(0,1)", ylim=(-0.1, 1.3))
```

Out[29]:

In [30]:
```
plot_T(Uniform(), 10, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = -1.2
```

Out[30]:

In [31]:
```
plot_T(Uniform(), 40, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = -1.2
```

Out[31]:

In [32]:
```
plot_T(Uniform(), 160, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = -1.2
```

Out[32]:

上の例を見れば, $T(\mu)$ の分布が t 分布でよく近似されている状況であっても, $(n-1)S^2/\sigma^2$ の分布がχ²分布で近似されていると考えてはいけないことがわかる. この点は以下でも同様である.

だから, **正規分布の標本分布モデルにおいて成り立っている $(n-1)S^2/\sigma^2 \sim \mathrm{Chisq}(n-1)$ を用いて, 分散 σ^2 の区間推定を行うことは大きなリスクが伴う.**

平均の区間推定と違って, **分散の区間推定は非常に難しい.**

Laplace分布の場合

In [33]:
```
plot(Laplace(), -4.5, 4.5; label="Laplace(0,1)")
```

Out[33]:

In [34]:
```
plot_T(Laplace(), 10, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = 3.0
```

Out[34]:

In [35]:
```
plot_T(Laplace(), 20, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = 3.0
```

Out[35]:

In [36]:
```
plot_T(Laplace(), 40, plot_chi=true, plot_t=true)
```

```
skewness = 0.0,  kurtosis = 3.0
```

Out[36]:

ガンマ分布の場合

ガンマ分布は左右非対称な分布の典型例である.

In [37]:
```
plot(Gamma(2,3); label="Gamma(2,3)")
```

Out[37]:

In [38]:
```
plot_T(Gamma(2,3), 10, plot_chi=true, plot_t=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[38]:

In [39]:
```
plot_T(Gamma(2,3), 40, plot_chi=true, plot_t=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[39]:

In [40]:
```
plot_T(Gamma(2,3), 160, plot_chi=true, plot_t=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[40]:

In [41]:
```
plot_T(Gamma(2,3), 640, plot_chi=true, plot_t=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[41]:

対数正規分布の場合

対数正規分布は左右非対称でかつ **外れ値が出易い** 分布の典型例である.

このような場合は n を非常に大きくしないと近似の精度が高まらない.

In [42]:
```
plot(LogNormal(), -0.1, 10; label="LogNormal(0,1)")
```

Out[42]:

In [43]:
```
plot_T(LogNormal(), 40, plot_chi=true, plot_t=true)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[43]:

In [44]:
```
plot_T(LogNormal(), 160, plot_chi=true, plot_t=true)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[44]:

In [45]:
```
plot_T(LogNormal(), 640, plot_chi=true, plot_t=true)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[45]:

In [46]:
```
plot_T(LogNormal(), 2560, plot_chi=true, plot_t=true)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[46]:

この例から, 左右非対称で外れ値を含む母集団分布からの無作為抽出で得たデータを扱う場合には, 中心極限定理を使った近似を用いることは要注意であることがわかる.

1.8 分布の対数の標本分布のT統計量の視覚化

データの数値 x_i がすべて正の値の場合には, $y_i = \log x_i$ と対数を取った後に, 「分布の対数の平均が λ_0 である」という仮説を扱った方が誤差が減る場合がある.

より一般に対数に限らず, データを適切な座標に変換してから統計分析にかけた方が誤差が小さくなる場合がある.

対数正規分布の場合

対数正規分布に従う確率変数の対数は定義より正規分布に従う. この場合には正値のデータの対数を取ってから計算することの効果は自明に絶大になる. (この場合は自明過ぎてあまり意味がない.)

In [47]:
```
plot_T(LogNormal(), 10, plot_chi=true, plot_t=true, logsample=true)
```

```
skewness = 6.184877138632554,  kurtosis = 110.9363921763115
```

Out[47]:

ガンマ分布の場合

ガンマ分布とその対数の密度函数達

In [48]:
```
μ, σ = mean(Gamma(2, 3)), std(Gamma(2, 3))
P = plot(x -> pdf(Gamma(2, 3), x), μ-4.5σ, μ+4.5σ; label="", title="Gamma(2,3)")
μ, σ = logmean(Gamma(2, 3)), logstd(Gamma(2, 3))
Q = plot(x -> pdflog(Gamma(2, 3), x), μ-4.5σ, μ+4.5σ; label="", title="Log of Gamma(2,3)")
plot(P, Q; size=(800, 250))
```

Out[48]:

対数を取る前

In [49]:
```
plot_T(Gamma(2,3), 10, plot_chi=true, plot_t=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[49]:

対数を取った後

In [50]:
```
plot_T(Gamma(2,3), 10, plot_chi=true, plot_t=true, logsample=true)
```

```
skewness = 1.414213562373095,  kurtosis = 3.0
```

Out[50]:

よく見ないと分からないかもしれないが, 対数を取った後の方が近似の精度が上がっていることが分かる.

以上では対数を取ることによって誤差を減らすことができる場合があることを示したが, より一般に, データを適切な座標に変換してから統計分析にかけた方が誤差が小さくなる場合がある.

データの数値の生成のされ方に関してなにがしかの知識がある場合には, その知識をうまく統計分析の枠組みに取り入れた方が誤差が小さな分析が可能になる可能性が増える.

1.9 平均に関するP値の使い方

母集団からの無作為抽出で得たサイズ n のデータ $x_1, \ldots, x_n$ の標本平均と不偏分散をそれぞれ $\bar{x}$, s^2 と書くとき, そのデータの数値に関する仮説「平均は $\mu = \mu_0$ である」のP値は

　データの数値と仮説「平均は $\mu = \mu_0$ である」の整合性の指標

として使われる.

ただし, P値を計算するときに用いた統計モデルや近似がうまく行っているというような仮定が不適切なことが原因で整合性の指標をおかしな値になってしまうリスクにも十分に配慮しなければいけない. さらに, データの取得法に問題があったことを疑わなければいけない場合は珍しくない(むしろ普通である). 統計分析の妥当性を確保するためには, 様々な可能性を疑って, 問題を一つひとつ丁寧に潰して行く必要がある.

有意水準と呼ばれる閾値 $0 \leq \alpha \leq 1$ を最初に決めておいて(目的に合わせて小さな正の値に取る), P値が有意水準 α 未満になったときに,

　データの数値と仮説「平均は $\mu = \mu_0$ である」のあいだに整合性が無さすぎる

と判定し,

　データの数値によって仮説「平均は $\mu = \mu_0$ である」が **棄却** された

と言うことがある. このような手続きを **検定**(testing, test)と呼ぶのであった.

1.10 P値の定義(2)のt分布を使う場合に対応する平均の信頼区間

有意水準を $0 \leq \alpha \leq 1$ と書き, 自由度 ν の t 分布において $t_{\nu,\alpha/2}$ 以上になる確率は $\alpha/2$ になると仮定する:

$$t_{\nu,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(\nu), 1 - \alpha/2).$$

例えば,

$$t_{10,5\%/2} \approx 2.2281, \quad t_{20,5\%/2} \approx 2.0860, \quad t_{30,5\%/2} \approx 2.0423.$$

自由度を大きくする極限では, $t_{\infty,5\%/2} = z_{5\%/2} \approx 1.9600$ となる. $t_{30,5\%/2} \approx 2.0423$ はその値よりも 4.2% 程度大きい. (信頼区間もその割合で広くなる.)

P値 $\mathrm{pvalue}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0) = 2(1-\mathrm{cdf}(\mathrm{TDist}(n-2), |t(\mu_0)|))$ に対応する信頼度 $1-\alpha$ の信頼区間は次のようになる:

$$\mathrm{confint}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0) = \left[\bar{x} - t_{n-1,\alpha/2}\sqrt{s^2/n},\ \bar{x} + t_{n-1,\alpha/2}\sqrt{s^2/n}\right].$$

これと $\mathrm{confint}_{\mathrm{Normal}}(\bar{x}, s^2|n, \mu=\mu_0)$ の違いは $z_{\alpha/2}$ を $t_{n-1,\alpha/2}$ に置き換える分の違いしかない.

証明もP値の定義(1)の場合と完全に同様である. 標準正規分布を t 分布で置き換えるだけでよい.

証明: 検定と信頼区間の表裏一体性より, P値函数 $\mathrm{pvalue}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0)$ に対応する信頼度 $1-\alpha$ の信頼区間は次のように定義されるのであった:

$$\mathrm{confint}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0) = \{\, \mu_0 \in \mathbb{R} \mid \mathrm{pvalue}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0) \geq \alpha \,\}.$$

そして,

$$\begin{aligned}
&\mathrm{pvalue}_{\mathrm{TDist}}(\bar{x}, s^2|n, \mu=\mu_0) \geq \alpha \\
&\iff 1-\mathrm{cdf}(\mathrm{TDist}(n-1), |t(\mu_0)|)) \geq \alpha/2 \\
&\iff |t(\mu_0)| = \frac{|\bar{x}-\mu_0|}{\sqrt{s^2/n}} \leq t_{n-1,\alpha/2} \\
&\iff \bar{x} - t_{n-1,\alpha/2}\sqrt{s^2/n} \leq \mu_0 \leq \bar{x} + t_{n-1,\alpha/2}\sqrt{s^2/n}.
\end{aligned}$$

これより, P値の定義(2)に対応する平均の信頼区間が上のようになることがわかった.

証明終

2 平均に関するP値と信頼区間の計算例

2.1 平均に関するP値と信頼区間の計算の仕方

(0) 前提

有意水準: $0 < \alpha < 1$.

検定したい仮説: 平均は $\mu = \mu_0$ である.

データの数値: $x_1, \ldots, x_n$.

母集団からの無作為抽出(または同分布の独立試行)で得られたサイズ n のデータの数値 $x_1, \ldots, x_n$ が得られていると仮定する.

計算したい信頼区間: 平均 μ に関する信頼度 $1-\alpha$ の信頼区間.

(1) データの数値の標本平均 $\bar{x}$ と不偏分散 s^2 を計算する:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2.$$

(2) データの数値の数値に関する仮説「平均は $\mu = \mu_0$ である」の t 値 $t = t(\mu_0)$ を計算する:

$$t = t(\mu_0) = \frac{\bar{x}-\mu_0}{\sqrt{s^2/n}}.$$

(3) データの数値に関する仮説「平均は $\mu = \mu_0$ である」のP値を自由度 $n-1$ の t 分布に従ってランダムに生成される値の絶対値が データの数値の t 値の絶対値以上になる確率として求める:

$$(\text{仮説「平均は } \mu = \mu_0 \text{ である」のP値}) = 2(1-\mathrm{cdf}(\mathrm{TDist}(n-1), |t(\mu_0)|)).$$

(4) 定数 $t_{n-1,\alpha/2}$ を, 自由度 $n-1$ の t 分布に従ってランダムに生成される値が $t_{n-1,\alpha/2}$ 以上になる確率は $\alpha/2$ であるという条件によって定める:

$$t_{n-1,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(n-1), 1-\alpha/2).$$

(5) 次の公式によって, 平均 μ に関する信頼度 $1-\alpha$ の信頼区間を求める:

$$(\text{平均 } \mu \text{ に関する信頼度 } 1-\alpha \text{ の信頼区間}) = \left[\bar{x} - t_{n-1,\alpha/2}\sqrt{s^2/n},\ \bar{x} + t_{n-1,\alpha/2}\sqrt{s^2/n}\right].$$

注意・警告: 中心極限定理による近似がうまく行っていない場合には, 以上のようにして求めたP値や信頼区間の誤差は大きくなる. 特にデータを取得した母集団の分布が左右非対称で外れ値を含んでいる場合には, サンプルサイズ n を相当に大きくしないと誤差が小さくならない. そのような場合には何らかの工夫や別の方法を検討した方がよいだろう. (上で説明した方法の使用にこだわる必要はない.)

2.2 平均に関するP値と信頼区間の計算例

有意水準: α = 5% = 0.05

サイズ $n = 11$ のデータの数値: 13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6

標本平均: mean(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6) = 6.6

不偏分散: var(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6) = 15.35

検定したい仮説: 平均は $\mu = 9.0$ である.

t値: (6.6 - 9.0)/sqrt(15.35/11) ≈ -2.03167106

P値: 2(1 - cdf(TDistribution(10), 2.03167106)) ≈ 0.0696099

検定: 上にデータに関する仮説「平均は $\mu = 9.0$ である」のP値は約 7% で最初に設定した有意水準 α より大きいので, 上にデータよって仮説「平均は $\mu = 9.0$ である」は棄却され **ない**.

t分布の分位点: $c = t_{n-1,\alpha/2}$ = quantile(TDistribution(10), 0.975) ≈ 2.22814

信頼区間: 6.6 - 2.22814 sqrt(15.35/11), 6.6 - 2.22814 sqrt(15.35/11) ≈ 3.96791, 9.23209

以上の計算法はWolframAlpha (https://www.wolframalpha.com/)でそのまま使える.

2.2.1 WolframAlphaでの平均に関するP値と信頼区間の計算例

mean(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6) → 実行 (https://www.wolframalpha.com/input?i=mean%2813.7%2C+12.9%2C+4.4%2C+5.2%2C+3.1%2C+2.9%2C+7.2%2C+10.3%2C+4.7%2C+4.6%2C+3.6%29)

var(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6) → 実行 (https://www.wolframalpha.com/input?i=var%2813.7%2C+12.9%2C+4.4%2C+5.2%2C+3.1%2C+2.9%2C+7.2%2C+10.3%2C+4.7%2C+4.6%2C+3.6%29)

(6.6 - 9.0)/sqrt(15.35/11) → 実行 (https://www.wolframalpha.com/input?i=%286.6+-+9.0%29%2Fsqrt%2815.35%2F11%29)

2(1 - cdf(TDistribution(10), 2.03167106)) → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2810%29%2C+2.03167106%29%29)

quantile(TDistribution(10), 0.975) → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%2810%29%2C+0.975%29)

6.6 - 2.22814 sqrt(15.35/11), 6.6 - 2.22814 sqrt(15.35/11) → 実行 (https://www.wolframalpha.com/input?i=6.6+-+2.22814+sqrt%2815.35%2F11%29%2C+6.6+-+2.22814+sqrt%2815.35%2F11%29)

2.2.2 Julia言語での平均に関するP値と信頼区間の計算例

ほとんど定義通りにJulia言語のコードを入力すれば計算できる.

In [51]:
```julia
x = [13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6]
n = length(x)
x̄ = mean(x)
s² = var(x)

μ = 9.0
t_value = (x̄ - μ)/√(s²/n)
P_value = 2ccdf(TDist(n-1), abs(t_value))

α = 0.05
c = quantile(TDist(n-1), 1 - α/2)
confidence_interval = [x̄ - c*√(s²/n), x̄ + c*√(s²/n)]

@show x n x̄ s²
println()
@show t_value P_value
println()
@show α c confidence_interval;
```

```
x = [13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6]
n = 11
x̄ = 6.6
s² = 15.35

t_value = -2.031671060092391
P_value = 0.06960994092988786

α = 0.05
c = 2.228138851986274
confidence_interval = [3.967913856820957, 9.232086143179043]
```

In [52]:
```julia
plot_confint_of_mean(x; α)
```

Out[52]:

この図の区間を見ると, 一瞬

　データを示す青丸達のかなりの部分が区間をはみ出している!

と思ってしまいがちだが, この区間はデータの分布の範囲を表す区間ではなく, モデルの分布の平均パラメータ μ に関する区間であることを思い出せばこれで問題ないことがわかる.

以下の図のようにサンプルサイズ n を大きくすれば, 平均の信頼区間の幅は狭くなる.

サンプルサイズ n を a 倍すると信頼区間の幅はおおよそ $\sqrt{a}$ 分の 1 になる.

精度を1桁上げるためには, サンプルサイズ n を 100 倍に増やす必要がある!

2.2.3 Julia言語での平均に関するP値と信頼区間のグラフ作成

In [53]:
```julia
plot(Gamma(5, 2); label="pdf of Gamma(5, 2)")
vline!([mean(Gamma(5, 2))]; label="mean(Gamma(5, 2))", c=1, ls=:dash)
```

Out[53]:

In [54]:
```
Random.seed!(4649373)
dist = Gamma(5,2)
x = rand(dist, 640)
a, b = extrema(x)
xlim = (a - 0.05(b - a), b + 0.05(b-a))
```

Out[54]: (0.4536438309971049, 29.70804423866017)

In [55]:
```
plot_confint_of_mean(x[1:10]; xlim)
```

Out[55]:

In [56]:
```
plot_confint_of_mean(x[1:40]; xlim)
```

Out[56]:

In [57]:
```
plot_confint_of_mean(x[1:160]; xlim)
```

Out[57]:

In [58]:
```
plot_confint_of_mean(x; xlim)
```

Out[58]:

P値函数も同時にプロットすると以下のようになる.

In [59]:
```
plot_confint_of_mean(x[1:10]; xlim, plot_pvaluefunc=true)
```

Out[59]:

In [60]:
```
plot_confint_of_mean(x[1:40]; xlim, plot_pvaluefunc=true)
```

Out[60]:

In [61]:
```
plot_confint_of_mean(x[1:160]; xlim, plot_pvaluefunc=true)
```

Out[61]:

In [62]:
```
plot_confint_of_mean(x; xlim, plot_pvaluefunc=true)
```

Out[62]:

データはガンマ分布 $\mathrm{Gamma}(5,2)$ の標本(独立同分布確率変数列の実現値)としてランダムに生成されている. 標本のサイズ n が大きくなるにつれて, P値函数と信頼区間の幅は縮まって行き, データを生成した分布の平均値の 10 に集中するようになる.

In [63]:
```
L = length(x)
anim = @animate for n in [fill(10,20); 10:2:L; fill(L, 20)]
    plot_confint_of_mean(x[1:n]; xlim, xtick=0:2:100, plot_pvaluefunc=true)
end
gif(anim, "images/confint_of_mean.gif")
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\confint_of_mean.gif
@ Plots D:\.julia\packages\Plots\3E7jF\src\animation.jl:114
```

Out[63]:

このノートのPDF版で上の動画をみることはできない. しかし, 以下の場所でみることができる:

- https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_mean.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_mean.gif)

2.2.4 R言語での平均に関するP値と信頼区間の計算例

ほとんど定義通りにJulia言語のコードを入力すれば計算できる.

```
x = c(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6)
n = length(x)
xbar = mean(x)
s2 = var(x)
mu = 9
t.value = (xbar - mu)/sqrt(s2/n)
p.value = 2*(1 - pt(abs(t.value), n-1))
alpha = 0.05
c = qt(1 - alpha/2, n-1)
conf.int = c(xbar - c*sqrt(s2/n), xbar + c*sqrt(s2/n))

cat("data: x = ", x, "\n")
cat("data size n = ", n, "\n")
cat("sample mean of data = ", xbar, "\n")
cat("unbiased variance of data = ", s2, "\n")
cat("\n")
cat("t.value = ", t.value, "\n")
cat("p.value = ", p.value, "\n")
cat("\n")
cat("alpha = ", alpha, "\n")
cat("t_{n-1, alpha/2} = ", c, "\n")
cat("conf.int = ", conf.int, "\n")
```

以下のようにすれば一発で計算することもできる:

```
t.test(c(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6), mu=9)
```

In [64]:

```
R"""
x = c(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6)
n = length(x)
xbar = mean(x)
s2 = var(x)
mu = 9
t.value = (xbar - mu)/sqrt(s2/n)
p.value = 2*(1 - pt(abs(t.value), n-1))
alpha = 0.05
c = qt(1 - alpha/2, n-1)
conf.int = c(xbar - c*sqrt(s2/n), xbar + c*sqrt(s2/n))

cat("data: x = ", x, "\n")
cat("data size n = ", n, "\n")
cat("sample mean of data = ", xbar, "\n")
cat("unbiased variance of data = ", s2, "\n")
cat("\n")
cat("t.value = ", t.value, "\n")
cat("p.value = ", p.value, "\n")
cat("\n")
cat("alpha = ", alpha, "\n")
cat("t_{n-1, alpha/2} = ", c, "\n")
cat("conf.int = ", conf.int, "\n")
""";
```

```
data: x =  13.7 12.9 4.4 5.2 3.1 2.9 7.2 10.3 4.7 4.6 3.6
data size n =  11
sample mean of data =  6.6
unbiased variance of data =  15.35

t.value =  -2.031671
p.value =  0.06960994

alpha =  0.05
t_{n-1, alpha/2} =  2.228139
conf.int =  3.967914 9.232086
```

In [65]:
```
R"""
t.test(c(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6), mu=9)
"""
```

Out[65]:
```
RObject{VecSxp}

        One Sample t-test

data:  c(13.7, 12.9, 4.4, 5.2, 3.1, 2.9, 7.2, 10.3, 4.7, 4.6, 3.6)
t = -2.0317, df = 10, p-value = 0.06961
alternative hypothesis: true mean is not equal to 9
95 percent confidence interval:
 3.967914 9.232086
sample estimates:
mean of x
      6.6
```

2.3 必修練習問題: 平均に関するP値と信頼区間の計算

サイズ 20 のデータとして次が与えられているとする:

> 38.1, 28.9, 30.4, 27.6, 38.0, 35.3, 30.0, 29.3, 32.9, 34.8,
> 39.4, 38.4, 28.4, 27.0, 35.9, 30.7, 28.2, 33.2, 33.0, 36.0

以上で説明した t 分布を使う方法によって以下を求めよ.

(1) 「平均は $\mu = 30.0$ である」という仮説のP値.

(2) 平均の 99% 信頼区間.

まずは, この問題よりも上の方にある説明だけを見て(これより下の方を見ないで), 計算してみよ.

自力で計算した結果が正解に一致していれば爽快な気分を味わえるだろう.

勉強の仕方: ある程度統計学の勉強が進んだら, 適当に統計分析の報告を見つけて, 同じ結果を数値的に再現できるかどうかを試してみるとよい.

2.3.1 WaolframAlphaによる必修練習問題の解き方

`mean(38.1, 28.9, ... 36.0)` → 実行 (https://www.wolframalpha.com/input?i=mean%2838.1%2C+28.9%2C+30.4%2C+27.6%2C+38.0%2C+35.3%2C+30.0%2C+29.3%2C+32.9%2C+34.8%2C+39.4%2C+38.4
→ 32.775

`var(38.1, 28.9, ... 36.0)` → 実行 (https://www.wolframalpha.com/input?i=var%2838.1%2C+28.9%2C+30.4%2C+27.6%2C+38.0%2C+35.3%2C+30.0%2C+29.3%2C+32.9%2C+34.8%2C+39.4%2C+38.4%
→ 16.1588

√(s²/n) の計算: `sqrt(16.1588/20)` → 実行 (https://www.wolframalpha.com/input?i=sqrt%2816.1588%2F20%29) → 0.898855

t値の計算: `(32.775 - 30)/0.898855` → 実行 (https://www.wolframalpha.com/input?i=%2832.775+-+30%29%2F0.898855) → 3.087261

P値の計算: `2(1 - cdf(TDistribution(19), 3.087261))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2819%29%2C+3.087261%29%29) → ??????

$\alpha = 0.01, n-1 = 19$ のときの $t_{\alpha/2,n-1}$ の計算: `quantile(TDistribution(19), 0.995)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%2819%29%2C+0.995%29) → 2.86093

99%信頼区間の計算: `32.775-2.86093*0.898855, 32.775+2.86093*.898855` → 実行 (https://www.wolframalpha.com/input?i=32.775-2.86093*0.898855%2C+32.775%2B2.86093*.898855) → ??????

2.3.2 Julia言語による必修練習問題の解き方

ほぼ定義通りの式をJulia言語のコードとして書けば計算できる.

以下のように函数にまとめてしまうと便利である.

```julia
function pvalue_tdist(x̄, s², n, μ)
    t = (x̄ - μ)/√(s²/n)
    2ccdf(TDist(n-1), abs(t))
end

function pvalue_tdist(x, μ)
    x̄, s², n = mean(x), var(x), length(x)
    pvalue_tdist(x̄, s², n, μ)
end

function confint_tdist(x̄, s², n; α = 0.05)
    c = quantile(TDist(n-1), 1-α/2)
    [x̄ - c*√(s²/n), x̄ + c*√(s²/n)]
end

function confint_tdist(x; α = 0.05)
    x̄, s², n = mean(x), var(x), length(x)
    confint_tdist(x̄, s², n; α)
end

function show_confint_of_mean(x; μ = mean(x), α = 0.05)
    println("(P-value of hypothesis \"mean = $μ\") = ", pvalue_tdist(x, μ))
    println("($(100(1-α))% confidence interval of mean) = ", confint_tdist(x; α))
end
```

`plot_confint_of_mean(x; α, plot_pvaluefunc)` の定義についてはこのノートの最初の方を見よ.

In [66]:

```julia
x = [38.1, 28.9, 30.4, 27.6, 38.0, 35.3, 30.0, 29.3, 32.9, 34.8,
     39.4, 38.4, 28.4, 27.0, 35.9, 30.7, 28.2, 33.2, 33.0, 36.0]
show_confint_of_mean(x; μ=30.0, α=0.01)
plot_confint_of_mean(x; α=0.01, plot_pvaluefunc=true, size=(600, 300))
```

```
(P-value of hypothesis "mean = 30.0") = 0.006064850644067268
(99.0% confidence interval of mean) = [30.203433862887117, 35.34656613711288]
```

Out[66]:

2.3.3 R言語による必修練習問題の解き方

R言語のインストール直後にすでに使えるようになっている `t.test` 函数一発で欲しい情報が得られる.

```r
x = c(38.1, 28.9, 30.4, 27.6, 38.0, 35.3, 30.0, 29.3, 32.9, 34.8,
    39.4, 38.4, 28.4, 27.0, 35.9, 30.7, 28.2, 33.2, 33.0, 36.0)
t.test(x, mu=30, conf.level=0.99)
```

実際に統計分析を行う場合にはこのような道具を使った方がよいだろう.

しかし, 統計学の勉強をする場合には, 計算の過程がわかるような道具の使い方をした方がよい.

あえてエレガントではなく, エレファントな方法を使ってみる ことは結構重要でである.

In [67]:

```
R"""
x = c(38.1, 28.9, 30.4, 27.6, 38.0, 35.3, 30.0, 29.3, 32.9, 34.8,
    39.4, 38.4, 28.4, 27.0, 35.9, 30.7, 28.2, 33.2, 33.0, 36.0)
t.test(x, mu=30, conf.level=0.99)
"""
```

Out[67]:

```
RObject{VecSxp}

        One Sample t-test

data:  x
t = 3.0873, df = 19, p-value = 0.006065
alternative hypothesis: true mean is not equal to 30
99 percent confidence interval:
 30.20343 35.34657
sample estimates:
mean of x
   32.775
```

しかし, これはとても便利である.

2.3.4 必修練習問題解答例

(1) (P値) ≈ 0.006065

(2) (99%信頼区間) ≈ [30.20, 35.35]

In [ ]:

```
```

検定と信頼区間: 比率の比較

- 黒木玄

- 2022-06-14〜2022-06-18

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 比率の比較に関するP値と信頼区間
  - 1.1 比率の比較に関するP値と信頼区間を使って行いたいこと
  - 1.2 2×2の分割表型データとその2つの二項分布モデル
  - 1.3 比率の違いを表す2つの指標: オッズ比パラメータ OR とリスク比パラメータ RR
- 2 Wald版のP値と信頼区間
  - 2.1 Wald版のオッズ比に関するP値と信頼区間の定義
  - 2.2 Wald版のオッズ比に関するP値と信頼区間の計算例
    - 2.2.1 WolframAlphaによるWald版のオッズ比に関するP値と信頼区間の計算の仕方
    - 2.2.2 Julia言語によるWald版のオッズ比に関するP値と信頼区間の計算の仕方(1)
    - 2.2.3 Julia言語によるWald版のオッズ比に関するP値と信頼区間の計算の仕方(2)
    - 2.2.4 Julia言語によるWald版のオッズ比に関するP値函数の視覚化
    - 2.2.5 R言語でのWald版のオッズ比に関するP値と信頼区間の計算の仕方
  - 2.3 確率変数の対数オッズ比の分散の近似値の導出
  - 2.4 問題: 対数リスク比の場合の正規分布近似
  - 2.5 問題: Wald版のリスク比に関するP値と信頼区間の定義
  - 2.6 問題: Wald版のリスク比に関するP値と信頼区間の計算例
    - 2.6.1 Julia言語によるWald版のリスク比に関するP値と信頼区間の計算の仕方
    - 2.6.2 Julia言語によるWald版のリスク比に関するP値函数の視覚化
    - 2.6.3 R言語によるWald版のリスク比に関する信頼区間の計算の仕方
    - 2.6.4 Wald版のリスク比に関するP値と信頼区間の計算例の問題の解答例
  - 2.7 問題: 比率の差の場合の正規分布近似
  - 2.8 問題: Wald版の比率の差に関するP値と信頼区間の定義
  - 2.9 問題: Wald版の比率の差に関するP値と信頼区間の計算例
    - 2.9.1 Julia言語によるWald版の比率の差に関するP値と信頼区間の計算の仕方
    - 2.9.2 Wald版の比率の差に関するP値と信頼区間の計算の解答例
  - 2.10 警告: 以上で紹介したWald版の比率の差に関するP値と信頼区間の精度は低い
- 3 オッズ比とリスク比に関するWald版のP値と信頼区間の必修計算問題
  - 3.1 オッズ比に関するWald版のP値と信頼区間の計算の仕方
  - 3.2 リスク比に関するWald版のP値と信頼区間の計算の仕方
  - 3.3 必修問題: Wald版のオッズ比とリスク比のP値と信頼区間の計算例
  - 3.4 WolframAlphaによる必修問題の解答例
  - 3.5 Julia言語による必修問題の解答例
  - 3.6 必修問題解答例
- 4 Pearsonのχ²検定版のP値と信頼区間
  - 4.1 Pearsonのχ²検定版のオッズ比に関するP値と信頼区間の定義
  - 4.2 Pearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算例
    - 4.2.1 WolframAlphaによるPearsonのχ²検定のP値の計算の仕方
    - 4.2.2 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算の仕方(1)
    - 4.2.3 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算の仕方(2)
    - 4.2.4 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値の視覚化
    - 4.2.5 R言語によるPearsonのχ²検定のP値の計算の仕方
  - 4.3 「オッズ比パラメータはOR=ωである」という仮説下での最尤法との関係

4.3.1 2つの二項分布モデル全体での最尤法
4.3.2 条件OR=ωでパラメータの動ける範囲が制限された2つの二項分布モデルでの最尤法
4.3.3 Wilks's theoremの適用
4.4 Wilks' theoremの数値的確認例
4.4.1 2つの二項分布モデルのパラメータを2つとも固定した場合 (自由度2)
4.4.2 2つの二項分布モデルの独立性に関する対数尤度比検定の場合 (OR=1とした場合, 自由度1)
4.4.3 正規分布モデルのパラメータを2つとも固定した場合 (自由度2)
4.4.4 正規分布モデルでσのみを固定した場合
4.4.5 正規分布モデルでμのみを固定した場合 (自由度1)
4.4.6 ガンマ分布モデルのパラメータを2つとも固定した場合 (自由度2)
4.4.7 ガンマ分布モデルで形状パラメータαのみを固定した場合 (自由度1)
4.5 問題: 「リスク比パラメータはRR=ρである」という仮説下での最尤法
4.6 問題: Pearsonのχ²検定版のリスク比に関するP値と信頼区間の定義
4.7 問題: Pearsonのχ²検定版のリスク比に関するP値と信頼区間の計算例
4.7.1 Julia言語によるPearsonのχ²検定版のリスク比に関するP値と信頼区間の計算例
4.7.2 Julia言語によるPearsonのχ²検定版のリスク比に関するP値函数の視覚化
4.7.3 Pearsonのχ²検定版のリスク比に関するP値と信頼区間の計算の解答例
4.8 ニューサンスパラメータの問題に関する注意
5 Fisher検定版のオッズ比に関するP値と信頼区間
5.1 条件付き確率分布を利用するアイデア
5.2 Sterne型のFisher検定のP値とそれに付随する信頼区間の定義
5.3 Clopper-Pearson型のFisher検定のP値とそれに付随する信頼区間
5.4 問題: Fisher検定版のオッズ比に関するP値と信頼区間の計算例
5.4.1 Julia言語によるFisher検定版のオッズ比に関するP値と信頼区間の計算例
5.4.2 Julia言語によるFisher²検定のオッズ比に関するP値函数の視覚化
5.4.3 R言語によるFisher検定版のP値と信頼区間の計算の仕方
5.4.4 Fisher検定版のオッズ比に関するP値と信頼区間の計算の解答例
5.5 Fisher検定における第一種の過誤の確率
6 おまけ: Bayes信用区間
6.1 オッズ比パラメータとリスク比パラメータのBayes信用区間
6.2 Bayes信用区間の計算例
6.3 Bayes信用区間に付随するP値函数
6.4 Bayes版P値函数の視覚化
6.5 Bayes版P値函数の視覚化の別の例
7 関連講義動画

In [1]:

```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Printf
using QuadGK
using RCall
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]: mykurtosis (generic function with 2 methods)

In [4]:

```julia
oddsratiohat(a, b, c, d) = safediv(a*d, b*c)
stderr_logoddsratiohat(a, b, c, d) = √(1/a + 1/b + 1/c + 1/d)

function pvalue_or_wald(a, b, c, d; ω=1)
    logORhat = log(oddsratiohat(a, b, c, d))
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logORhat - log(ω)), SEhat_logORhat))
end

function confint_or_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    ORhat = oddsratiohat(a, b, c, d)
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logORhat), ORhat), safemul(exp(z*SEhat_logORhat), ORhat)]
end
```

Out[4]: confint_or_wald (generic function with 1 method)

In [5]:

```
@show confint_or_wald(0, 1, 1, 1) confint_or_wald(1e-4, 1, 1, 1)
@show confint_or_wald(1, 0, 1, 1) confint_or_wald(1, 1e-4, 1, 1)
@show confint_or_wald(1, 1, 0, 1) confint_or_wald(1, 1, 1e-4, 1)
@show confint_or_wald(1, 1, 1, 0) confint_or_wald(1, 1, 1, 1e-4)
@show confint_or_wald(0, 0, 1, 1) confint_or_wald(1e-4, 1e-4, 1, 1)
@show confint_or_wald(1, 1, 0, 0) confint_or_wald(1, 1, 1e-4, 1e-4)
@show confint_or_wald(0, 1, 0, 1) confint_or_wald(1e-4, 1, 1e-4, 1)
@show confint_or_wald(1, 0, 1, 0) confint_or_wald(1, 1e-4, 1, 1e-4);
```

```
confint_or_wald(0, 1, 1, 1) = [0.0, Inf]
confint_or_wald(0.0001, 1, 1, 1) = [7.363403724036157e-90, 1.3580675968312418e81]
confint_or_wald(1, 0, 1, 1) = [0.0, Inf]
confint_or_wald(1, 0.0001, 1, 1) = [7.363403724036156e-82, 1.3580675968312418e89]
confint_or_wald(1, 1, 0, 1) = [0.0, Inf]
confint_or_wald(1, 1, 0.0001, 1) = [7.363403724036156e-82, 1.3580675968312418e89]
confint_or_wald(1, 1, 1, 0) = [0.0, Inf]
confint_or_wald(1, 1, 1, 0.0001) = [7.363403724036157e-90, 1.3580675968312418e81]
confint_or_wald(0, 0, 1, 1) = [0.0, Inf]
confint_or_wald(0.0001, 0.0001, 1, 1) = [4.1295675438516624e-121, 2.4215610699693177e120]
confint_or_wald(1, 1, 0, 0) = [0.0, Inf]
confint_or_wald(1, 1, 0.0001, 0.0001) = [4.1295675438516624e-121, 2.4215610699693177e120]
confint_or_wald(0, 1, 0, 1) = [0.0, Inf]
confint_or_wald(0.0001, 1, 0.0001, 1) = [4.1295675438516624e-121, 2.4215610699693177e120]
confint_or_wald(1, 0, 1, 0) = [0.0, Inf]
confint_or_wald(1, 0.0001, 1, 0.0001) = [4.1295675438516624e-121, 2.4215610699693177e120]
```

In [6]:

```
riskratiohat(a, b, c, d) = safediv(a*(c+d), (a+b)*c)
stderr_logriskratiohat(a, b, c, d) = √(1/a - 1/(a+b) + 1/c - 1/(c+d))

function pvalue_rr_wald(a, b, c, d; ρ=1)
    (a+b==0 || c+d==0) && return 1.0
    logRRhat = log(riskratiohat(a, b, c, d))
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logRRhat - log(ρ)), SEhat_logRRhat))
end

function confint_rr_wald(a, b, c, d; α=0.05)
    (a+b==0 || c+d==0) && return [0, Inf]
    z = quantile(Normal(), 1-α/2)
    RRhat = riskratiohat(a, b, c, d)
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logRRhat), RRhat), safemul(exp(z*SEhat_logRRhat), RRhat)]
end
```

Out[6]:

```
confint_rr_wald (generic function with 1 method)
```

In [7]:

```
@show confint_rr_wald(0, 1, 1, 1) confint_rr_wald(1e-4, 1, 1, 1)
@show confint_rr_wald(1, 0, 1, 1) confint_rr_wald(1, 1e-4, 1, 1)
@show confint_rr_wald(1, 1, 0, 1) confint_rr_wald(1, 1, 1e-4, 1)
@show confint_rr_wald(1, 1, 1, 0) confint_rr_wald(1, 1, 1, 1e-4)
@show confint_rr_wald(0, 0, 1, 1) confint_rr_wald(1e-4, 1e-4, 1, 1)
@show confint_rr_wald(1, 1, 0, 0) confint_rr_wald(1, 1, 1e-4, 1e-4)
@show confint_rr_wald(0, 1, 0, 1) confint_rr_wald(1e-4, 1, 1e-4, 1)
@show confint_rr_wald(1, 0, 1, 0) confint_rr_wald(1, 1e-4, 1, 1e-4);
```

```
confint_rr_wald(0, 1, 1, 1) = [0.0, Inf]
confint_rr_wald(0.0001, 1, 1, 1) = [1.5239118697601845e-89, 2.62429881894243e81]
confint_rr_wald(1, 0, 1, 1) = [0.5001953065198115, 7.996876315834783]
confint_rr_wald(1, 0.0001, 1, 1) = [0.5000759918604454, 7.997184798065742]
confint_rr_wald(1, 1, 0, 1) = [0.0, Inf]
confint_rr_wald(1, 1, 0.0001, 1) = [3.810541668433138e-82, 6.562059262373016e88]
confint_rr_wald(1, 1, 1, 0) = [0.12504882662995287, 1.9992190789586957]
confint_rr_wald(1, 1, 1, 0.0001) = [0.12504400301489435, 1.999696078749301]
confint_rr_wald(0, 0, 1, 1) = [0.0, Inf]
confint_rr_wald(0.0001, 0.0001, 1, 1) = [6.426171133615772e-61, 1.5561365846124554e60]
confint_rr_wald(1, 1, 0, 0) = [0.0, Inf]
confint_rr_wald(1, 1, 0.0001, 0.0001) = [6.426171133615772e-61, 1.5561365846124554e60]
confint_rr_wald(0, 1, 0, 1) = [0.0, Inf]
confint_rr_wald(0.0001, 1, 0.0001, 1) = [4.245626444711269e-121, 2.3553650162644178e120]
confint_rr_wald(1, 0, 1, 0) = [1.0, 1.0]
confint_rr_wald(1, 0.0001, 1, 0.0001) = [0.9726638925089683, 1.0281043716144522]
```

In [8]:

```
# 警告: 以下の実装の精度は低い. 改良の仕方が
#
# Kenneth J. Rothman, Sander Greenland, and Timothy L. Lash
# Modern Epistemology, Third Edition, 2008, 888 pages
#
# の
#
# Chapter 14. Instroduction to Categorical Statistics
# Section. Two Study Groups: Large-Sample Methods, pp.299-300
#
# に書いてある. そこでは, 次の文献が引用されている:
#
# Guangyong Zou and Allan Donner
# A simple alternative confidence interval for the difference between two proportions
# Controlled Clinical Trials, Volume 25, Issue 1, February 2004, Pages 3-12
# https://doi.org/10.1016/j.cct.2003.08.010
#
# Zou-Donnerの信頼区間に対応するP値函数の実装については
#
# https://github.com/genkuroki/public/blob/main/0033/probability%20of%20alpha%20error%20of%202
#
# を参照せよ.

riskdiffhat(a, b, c, d) = safediv(a, a+b) - safediv(c, c+d)

function stderr_riskdiffhat(a, b, c, d)
    m, n = a+b, c+d
    p̂, q̂ = safediv(a, m), safediv(c, n)
    √(safediv(p̂*(1-p̂), m) + safediv(q̂*(1-q̂), n))
end

function pvalue_rd_wald(a, b, c, d; Δ=0)
    RDhat = riskdiffhat(a, b, c, d)
    SEhat_riskdiffhat = stderr_riskdiffhat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(RDhat - Δ), SEhat_riskdiffhat))
end

function confint_rd_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    RDhat = riskdiffhat(a, b, c, d)
    SEhat_riskdiffhat = stderr_riskdiffhat(a, b, c, d)
    [RDhat - z*SEhat_riskdiffhat, RDhat + z*SEhat_riskdiffhat]
end
```

Out[8]:

```
confint_rd_wald (generic function with 1 method)
```

In [9]:

```
@show confint_rd_wald(0, 1, 1, 1) confint_rd_wald(1e-4, 1, 1, 1)
@show confint_rd_wald(1, 0, 1, 1) confint_rd_wald(1, 1e-4, 1, 1)
@show confint_rd_wald(1, 1, 0, 1) confint_rd_wald(1, 1, 1e-4, 1)
@show confint_rd_wald(1, 1, 1, 0) confint_rd_wald(1, 1, 1, 1e-4)
@show confint_rd_wald(0, 0, 1, 1) confint_rd_wald(1e-4, 1e-8, 1, 1)
@show confint_rd_wald(1, 1, 0, 0) confint_rd_wald(1, 1, 1e-4, 1e-8)
@show confint_rd_wald(0, 1, 0, 1) confint_rd_wald(1e-4, 1, 1e-8, 1)
@show confint_rd_wald(1, 0, 1, 0) confint_rd_wald(1, 1e-4, 1, 1e-8);
```

```
confint_rd_wald(0, 1, 1, 1) = [-1.1929519121748402, 0.1929519121748402]
confint_rd_wald(0.0001, 1, 1, 1) = [-1.1931289644203495, 0.19332894442234927]
confint_rd_wald(1, 0, 1, 1) = [-0.1929519121748402, 1.1929519121748402]
confint_rd_wald(1, 0.0001, 1, 1) = [-0.19332894442234927, 1.1931289644203495]
confint_rd_wald(1, 1, 0, 1) = [-0.1929519121748402, 1.1929519121748402]
confint_rd_wald(1, 1, 0.0001, 1) = [-0.19332894442234927, 1.1931289644203495]
confint_rd_wald(1, 1, 1, 0) = [-1.1929519121748402, 0.1929519121748402]
confint_rd_wald(1, 1, 1, 0.0001) = [-1.1931289644203495, 0.19332894442234927]
confint_rd_wald(0, 0, 1, 1) = [-1.1929519121748402, 0.1929519121748402]
confint_rd_wald(0.0001, 1.0e-8, 1, 1) = [-1.5786785827129024, 2.5784786027109026]
confint_rd_wald(1, 1, 0, 0) = [-0.1929519121748402, 1.1929519121748402]
confint_rd_wald(1, 1, 0.0001, 1.0e-8) = [-2.5784786027109026, 1.5786785827129024]
confint_rd_wald(0, 1, 0, 1) = [0.0, 0.0]
confint_rd_wald(0.0001, 1, 1.0e-8, 1) = [-0.019497700370328632, 0.019697660372328635]
confint_rd_wald(1, 0, 1, 0) = [0.0, 0.0]
confint_rd_wald(1, 0.0001, 1, 1.0e-8) = [-0.019697660372323317, 0.019497700370323314]
```

In [10]:

```julia
function delta(a, b, c, d; ω=1)
    A, B, C = 1-ω, a+d+ω*(b+c), a*d-ω*b*c
    isinf(ω) ? typeof(ω)(-min(b, c)) : safediv(2C, B + √(B^2 - 4A*C))
end

# correction = 0.5 は連続性補正を与える.
function _chisqstat_or(a, b, c, d, δ; correction=0.0)
    ã, b̃, c̃, d̃ = a-δ, b+δ, c+δ, d-δ
    safemul(max(0, abs(δ)-correction)^2, 1/ã + 1/b̃ + 1/c̃ + 1/d̃)
end

function chisqstat_or(a, b, c, d; ω=1, correction=0.0)
    δ = delta(a, b, c, d; ω)
    _chisqstat_or(a, b, c, d, δ; correction)
end

function pvalue_or_pearson_chisq(a, b, c, d; ω=1, correction=0.0)
    χ² = chisqstat_or(a, b, c, d; ω, correction)
    ccdf(Chisq(1), χ²)
end

function confint_or_pearson_chisq(a, b, c, d; α=0.05, correction=0.0)
    (a+b==0 || c+d==0 || a+c==0 || b+d==0) && return [0, Inf]
    f(logω) = logit(pvalue_or_pearson_chisq(a, b, c, d; ω=exp(logω), correction)) - logit(α)
    ps = if a == 0 || d == 0
        [0, exp(find_zero(f, 0.0))]
    elseif b == 0 || c == 0
        [exp(find_zero(f, 0.0)), Inf]
    else
        ORhat = oddsratiohat(a, b, c, d)
        ω_L, ω_U = ORhat/2, 2ORhat
        [exp(find_zero(f, log(ω_L))), exp(find_zero(f, log(ω_U)))]
    end
end
```

Out[10]:

```
confint_or_pearson_chisq (generic function with 1 method)
```

In [11]:

```julia
@show confint_or_pearson_chisq(0, 1, 1, 1) confint_or_pearson_chisq(1e-4, 1, 1, 1)
@show confint_or_pearson_chisq(1, 0, 1, 1) confint_or_pearson_chisq(1, 1e-4, 1, 1)
@show confint_or_pearson_chisq(1, 1, 0, 1) confint_or_pearson_chisq(1, 1, 1e-4, 1)
@show confint_or_pearson_chisq(1, 1, 1, 0) confint_or_pearson_chisq(1, 1, 1, 1e-4)
@show confint_or_pearson_chisq(0, 0, 1, 1) confint_or_pearson_chisq(1e-4, 1e-4, 1, 1)
@show confint_or_pearson_chisq(1, 1, 0, 0) confint_or_pearson_chisq(1, 1, 1e-4, 1e-4)
@show confint_or_pearson_chisq(0, 1, 0, 1) confint_or_pearson_chisq(1e-4, 1, 1e-4, 1)
@show confint_or_pearson_chisq(1, 0, 1, 0) confint_or_pearson_chisq(1, 1e-4, 1, 1e-4);
```

```
confint_or_pearson_chisq(0, 1, 1, 1) = [0.0, 11.137973937616904]
confint_or_pearson_chisq(0.0001, 1, 1, 1) = [2.602261452525905e-9, 11.140095739591606]
confint_or_pearson_chisq(1, 0, 1, 1) = [0.08978293589129746, Inf]
confint_or_pearson_chisq(1, 0.0001, 1, 1) = [0.08976583535507918, 3.842811409403394e8]
confint_or_pearson_chisq(1, 1, 0, 1) = [0.08978293589129746, Inf]
confint_or_pearson_chisq(1, 1, 0.0001, 1) = [0.08976583535507918, 3.842811409403394e8]
confint_or_pearson_chisq(1, 1, 1, 0) = [0.0, 11.137973937616904]
confint_or_pearson_chisq(1, 1, 1, 0.0001) = [2.602261452525905e-9, 11.140095739591601]
confint_or_pearson_chisq(0, 0, 1, 1) = [0.0, Inf]
confint_or_pearson_chisq(0.0001, 0.0001, 1, 1) = [1.3012947051505169e-5, 76846.54337277808]
confint_or_pearson_chisq(1, 1, 0, 0) = [0.0, Inf]
confint_or_pearson_chisq(1, 1, 0.0001, 0.0001) = [1.3012947051505169e-5, 76846.54337277808]
confint_or_pearson_chisq(0, 1, 0, 1) = [0.0, Inf]
confint_or_pearson_chisq(0.0001, 1, 0.0001, 1) = [1.3012947051505169e-5, 76846.54337277808]
confint_or_pearson_chisq(1, 0, 1, 0) = [0.0, Inf]
confint_or_pearson_chisq(1, 0.0001, 1, 0.0001) = [1.3012947051505169e-5, 76846.54337277808]
```

次のセルの計算結果を Cornfeild (1956), p.139, (4.2) と比較せよ.

- Jerome Cornfield, A Statistical Problem Arising from Retrospective Studies, Berkeley Symposium on Mathematical Statistics and Probability, 1956: 135-148 (1956) [link (https://projecteuclid.org/ebooks/berkeley-symposium-on-mathematical-statistics-and-probability/Proceedings%20of%20the%20Third%20Berkeley%20Symposium%20on%20Mathematical%20Statistics%20and%20

$$z_2' = 0.6229$$

(4.2)

$$z_1' = 0.0296\,.$$

In [12]:

```julia
# Cornfeild (1956), p.139, (4.2) と比較せよ.
a, b, c, d = 3, 11, 60, 32
@show confint_or_wald(a, b, c, d)
@show confint_or_pearson_chisq(a, b, c, d)
@show confint_or_pearson_chisq(a, b, c, d; correction=0.5);
```

```
confint_or_wald(a, b, c, d) = [0.037829803607039435, 0.5592686923030045]
confint_or_pearson_chisq(a, b, c, d) = [0.04084291410483445, 0.5253472140108638]
confint_or_pearson_chisq(a, b, c, d; correction = 0.5) = [0.02962955731952911, 0.622811954245519
2]
```

上のセルで計算した連続補正版の信頼区間の値が Cornfeild (1956), p.139, (4.2) の値に一致している.

In [13]:

```julia
function Delta(a, b, c, d; ρ=1)
    m, n = a+b, c+d
    A, B, C = ρ-1, n-a+ρ*(m-c), a*n-ρ*m*c
    isinf(ρ) ? typeof(ω)(-c) : safediv(2C, B + √(B^2 - 4A*C))
end

function _chisqstat_rr(a, b, c, d, Δ)
    m, n = a+b, c+d
    safemul(Δ^2, safediv(b, m*(a-Δ)) + safediv(d, n*(c+Δ)))
end

function chisqstat_rr(a, b, c, d; ρ=1)
    Δ = Delta(a, b, c, d; ρ)
    _chisqstat_rr(a, b, c, d, Δ)
end

function pvalue_rr_pearson_chisq(a, b, c, d; ρ=1)
    χ² = chisqstat_rr(a, b, c, d; ρ)
    ccdf(Chisq(1), χ²)
end

function confint_rr_pearson_chisq(a, b, c, d; α=0.05)
    (a+b==0 || c+d==0 || a+c==0 || b+d==0) && return [0, Inf]
    f(logρ) = logit(pvalue_rr_pearson_chisq(a, b, c, d; ρ=exp(logρ))) - logit(α)
    RRhat = riskratiohat(a, b, c, d)
    if a == 0
        [0.0, exp(find_zero(f, 0.0))]
    elseif c == 0
        [exp(find_zero(f, 0.0)), Inf]
    elseif b == 0
        [0.0, exp(find_zero(f, log(2RRhat)))]
    elseif d == 0
        [exp(find_zero(f, log(RRhat/2))), Inf]
    else
        ρ_L, ρ_U = RRhat/2, 2RRhat
        [exp(find_zero(f, log(ρ_L))), exp(find_zero(f, log(ρ_U)))]
    end
end
```

Out[13]:

```
confint_rr_pearson_chisq (generic function with 1 method)
```

In [14]:

```
@show confint_rr_pearson_chisq(0, 1, 1, 1) confint_rr_pearson_chisq(1e-4, 1, 1, 1)
@show confint_rr_pearson_chisq(1, 0, 1, 1) confint_rr_pearson_chisq(1, 1e-4, 1, 1)
@show confint_rr_pearson_chisq(1, 1, 0, 1) confint_rr_pearson_chisq(1, 1, 1e-4, 1)
@show confint_rr_pearson_chisq(1, 1, 1, 0) confint_rr_pearson_chisq(1, 1, 1, 1e-4)
@show confint_rr_pearson_chisq(0, 0, 1, 1) confint_rr_pearson_chisq(1e-4, 1e-4, 1, 1)
@show confint_rr_pearson_chisq(1, 1, 0, 0) confint_rr_pearson_chisq(1, 1, 1e-4, 1e-4)
@show confint_rr_pearson_chisq(0, 1, 0, 1) confint_rr_pearson_chisq(1e-4, 1, 1e-4, 1)
@show confint_rr_pearson_chisq(1, 0, 1, 0) confint_rr_pearson_chisq(1, 1e-4, 1, 1e-4);
```

```
confint_rr_pearson_chisq(0, 1, 1, 1) = [0.0, 4.10417860525958]
confint_rr_pearson_chisq(0.0001, 1, 1, 1) = [5.205303607358693e-9, 4.104616767401084]
confint_rr_pearson_chisq(1, 0, 1, 1) = [0.0, 10.578517350254106]
confint_rr_pearson_chisq(1, 0.0001, 1, 1) = [0.32665454735880284, 10.577873776111911]
confint_rr_pearson_chisq(1, 1, 0, 1) = [0.2436541135706135, Inf]
confint_rr_pearson_chisq(1, 1, 0.0001, 1) = [0.24362810383225364, 1.9211175282574818e8]
confint_rr_pearson_chisq(1, 1, 1, 0) = [0.09453120573423061, Inf]
confint_rr_pearson_chisq(1, 1, 1, 0.0001) = [0.0945369571584704, 3.0613380652042177]
confint_rr_pearson_chisq(0, 0, 1, 1) = [0.0, Inf]
confint_rr_pearson_chisq(0.0001, 0.0001, 1, 1) = [2.602945936963888e-5, 9.681923544184194]
confint_rr_pearson_chisq(1, 1, 0, 0) = [0.0, Inf]
confint_rr_pearson_chisq(1, 1, 0.0001, 0.0001) = [0.10328526097487344, 38418.00883361873]
confint_rr_pearson_chisq(0, 1, 0, 1) = [0.0, Inf]
confint_rr_pearson_chisq(0.0001, 1, 0.0001, 1) = [1.3016851156610122e-5, 76823.49501927945]
confint_rr_pearson_chisq(1, 0, 1, 0) = [0.0, Inf]
confint_rr_pearson_chisq(1, 0.0001, 1, 0.0001) = [0.20653685531726315, 4.84175087521499]
```

In [15]:

```julia
_pdf_le(x, (dist, y)) =  pdf(dist, x) ≲ y

function _search_boundary(f, x0, Δx, param)
    x = x0
    if f(x, param)
        while f(x - Δx, param) x -= Δx end
    else
        x += Δx
        while !f(x, param) x += Δx end
    end
    x
end

function pvalue_sterne(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    Px == 0 && return Px
    Px == 1 && return Px
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = _search_boundary(_pdf_le, 2m - x, 1, (dist, Px))
        cdf(dist, x) + ccdf(dist, y-1)
    else # x > m
        y = _search_boundary(_pdf_le, 2m - x, -1, (dist, Px))
        cdf(dist, y) + ccdf(dist, x-1)
    end
end

function pvalue_or_sterne(a, b, c, d; ω=1)
    fnch = if ω == 1
        Hypergeometric(a+b, c+d, a+c)
    else
        FisherNoncentralHypergeometric(a+b, c+d, a+c, ω)
    end
    pvalue_sterne(fnch, a)
end

function confint_or_sterne(a, b, c, d; α = 0.05)
    (a+b==0 || c+d==0 || a+c==0 || b+d==0) && return [0, Inf]
    f(logω) = logit(pvalue_or_sterne(a, b, c, d; ω=exp(logω))) - logit(α)
    if a == 0 || d == 0
        [0.0, exp(find_zero(f, 0.0))]
    elseif b == 0 || c == 0
        [exp(find_zero(f, 0.0)), Inf]
    else
        ω_L, ω_U = confint_or_wald(a, b, c, d; α = α/10)
        ps = exp.(find_zeros(f, log(ω_L), log(ω_U)))
        # 次の行は稀に区間にならない場合への対策
        [first(ps), last(ps)]
    end
end
```

Out[15]:

```
confint_or_sterne (generic function with 1 method)
```

In [16]:

```
@show confint_or_sterne(0, 10, 10, 10) confint_or_pearson_chisq(1e-4, 10, 10, 10)
@show confint_or_sterne(10, 0, 10, 10) confint_or_pearson_chisq(10, 1e-4, 10, 10)
@show confint_or_sterne(10, 10, 0, 10) confint_or_pearson_chisq(10, 10, 1e-4, 10)
@show confint_or_sterne(10, 10, 10, 0) confint_or_pearson_chisq(10, 10, 10, 1e-4)
@show confint_or_sterne(0, 0, 10, 10) confint_or_pearson_chisq(1e-4, 1e-4, 10, 10)
@show confint_or_sterne(10, 10, 0, 0) confint_or_pearson_chisq(10, 10, 1e-4, 1e-4)
@show confint_or_sterne(0, 10, 0, 10) confint_or_pearson_chisq(1e-4, 10, 1e-4, 10)
@show confint_or_sterne(10, 0, 10, 0) confint_or_pearson_chisq(10, 1e-4, 10, 1e-4);
```

```
confint_or_sterne(0, 10, 10, 10) = [0.0, 0.5431888559914949]
confint_or_pearson_chisq(0.0001, 10, 10, 10) = [2.6029641085026625e-10, 0.44168708894449726]
confint_or_sterne(10, 0, 10, 10) = [1.8409803311864292, Inf]
confint_or_pearson_chisq(10, 0.0001, 10, 10) = [2.264046255890583, 3.8417740633827114e9]
confint_or_sterne(10, 10, 0, 10) = [1.8409803311864295, Inf]
confint_or_pearson_chisq(10, 10, 0.0001, 10) = [2.264046255890583, 3.841774063392443e9]
confint_or_sterne(10, 10, 10, 0) = [0.0, 0.5431888559914949]
confint_or_pearson_chisq(10, 10, 10, 0.0001) = [2.6029641085026625e-10, 0.44168708894449726]
confint_or_sterne(0, 0, 10, 10) = [0.0, Inf]
confint_or_pearson_chisq(0.0001, 0.0001, 10, 10) = [1.301528947084912e-5, 76832.712959483]
confint_or_sterne(10, 10, 0, 0) = [0.0, Inf]
confint_or_pearson_chisq(10, 10, 0.0001, 0.0001) = [1.301528947084912e-5, 76832.7129594542]
confint_or_sterne(0, 10, 0, 10) = [0.0, Inf]
confint_or_pearson_chisq(0.0001, 10, 0.0001, 10) = [1.301528947084912e-5, 76832.7129594542]
confint_or_sterne(10, 0, 10, 0) = [0.0, Inf]
confint_or_pearson_chisq(10, 0.0001, 10, 0.0001) = [1.301528947084912e-5, 76832.712959483]
```

In [17]:

```
function pvalue_or_clopper_pearson(a, b, c, d; ω=1)
    fnch = if ω == 1
        Hypergeometric(a+b, c+d, a+c)
    else
        FisherNoncentralHypergeometric(a+b, c+d, a+c, ω)
    end
    min(1, 2cdf(fnch, a), 2ccdf(fnch, a-1))
end

function confint_or_clopper_pearson(a, b, c, d; α = 0.05)
    (a+b==0 || c+d==0 || a+c==0 || b+d==0) && return [0, Inf]
    f(ω) = logit(pvalue_or_clopper_pearson(a, b, c, d; ω)) - logit(α)
    if a == 0 || d == 0
        [0.0, find_zero(f, 1.0)]
    elseif b == 0 || c == 0
        [find_zero(f, 1.0), Inf]
    else
        ω_L, ω_U = confint_or_wald(a, b, c, d; α = α/10)
        find_zeros(f, ω_L, ω_U)
    end
end
```

Out[17]:

```
confint_or_clopper_pearson (generic function with 1 method)
```

In [18]:

```
@show confint_or_clopper_pearson(0, 10, 10, 10) confint_or_sterne(0, 10, 10, 10)
@show confint_or_clopper_pearson(10, 0, 10, 10) confint_or_sterne(10, 0, 10, 10)
@show confint_or_clopper_pearson(10, 10, 0, 10) confint_or_sterne(10, 10, 0, 10)
@show confint_or_clopper_pearson(10, 10, 10, 0) confint_or_sterne(10, 10, 10, 0)
@show confint_or_clopper_pearson(0, 0, 10, 10) confint_or_sterne(0, 0, 10, 10)
@show confint_or_clopper_pearson(10, 10, 0, 0) confint_or_sterne(10, 10, 0, 0)
@show confint_or_clopper_pearson(0, 10, 0, 10) confint_or_sterne(0, 10, 0, 10)
@show confint_or_clopper_pearson(10, 0, 10, 0) confint_or_sterne(10, 0, 10, 0);
```

```
confint_or_clopper_pearson(0, 10, 10, 10) = [0.0, 0.6246724957742202]
confint_or_sterne(0, 10, 10, 10) = [0.0, 0.5431888559914949]
confint_or_clopper_pearson(10, 0, 10, 10) = [1.6008388503812658, Inf]
confint_or_sterne(10, 0, 10, 10) = [1.8409803311864292, Inf]
confint_or_clopper_pearson(10, 10, 0, 10) = [1.600838850381266, Inf]
confint_or_sterne(10, 10, 0, 10) = [1.8409803311864295, Inf]
confint_or_clopper_pearson(10, 10, 10, 0) = [0.0, 0.6246724957742202]
confint_or_sterne(10, 10, 10, 0) = [0.0, 0.5431888559914949]
confint_or_clopper_pearson(0, 0, 10, 10) = [0.0, Inf]
confint_or_sterne(0, 0, 10, 10) = [0.0, Inf]
confint_or_clopper_pearson(10, 10, 0, 0) = [0.0, Inf]
confint_or_sterne(10, 10, 0, 0) = [0.0, Inf]
confint_or_clopper_pearson(0, 10, 0, 10) = [0.0, Inf]
confint_or_sterne(0, 10, 0, 10) = [0.0, Inf]
confint_or_clopper_pearson(10, 0, 10, 0) = [0.0, Inf]
confint_or_sterne(10, 0, 10, 0) = [0.0, Inf]
```

In [19]:

```julia
function sim_probabilities_of_type_I_error(m, n, p, q=p;
        ω=p*(1-q)/((1-p)*q), L=10^5)
    bina, binc = Binomial(m, p), Binomial(n, q)
    p_sterne = similar(zeros(), L)
    p_clopper_pearson = similar(zeros(), L)
    p_pearson_chisq = similar(zeros(), L)
    p_wald = similar(zeros(), L)
    @threads for i in 1:L
        a, c = rand(bina), rand(binc)
        b, d = m-a, n-c
        p_sterne[i] = pvalue_or_sterne(a, b, c, d; ω)
        p_clopper_pearson[i] = pvalue_or_clopper_pearson(a, b, c, d; ω)
        p_pearson_chisq[i] = pvalue_or_pearson_chisq(a, b, c, d; ω)
        p_wald[i] = pvalue_or_wald(a, b, c, d; ω)
    end
    (; p_sterne, p_clopper_pearson, p_pearson_chisq, p_wald)
end

function plot_probabilities_of_type_I_error(m, n, p, q=p;
        ω=p*(1-q)/((1-p)*q), L=10^6, kwargs...)
    (; p_sterne, p_clopper_pearson, p_pearson_chisq, p_wald) =
        sim_probabilities_of_type_I_error(m, n, p, q; ω, L)
    F_sterne = ecdf(p_sterne)
    F_clopper_pearson = ecdf(p_clopper_pearson)
    F_pearson_chisq = ecdf(p_pearson_chisq)
    F_wald = ecdf(p_wald)

    x = 0:0.001:1
    tick = 0:0.1:1
    P = plot(; legend=:topleft)
    plot!(x, x -> F_sterne(x); label="Fisher (Sterne)")
    plot!(x, x -> F_clopper_pearson(x); label="Fisher (CP)", ls=:dash)
    plot!(x, x -> F_pearson_chisq(x); label="Pearson χ²", ls=:dashdot)
    plot!(x, x -> F_wald(x); label="Wald", ls=:dot, lw=2)
    plot!(x, identity; label="", c=:black, ls=:dot)
    plot!(; xtick=tick, ytick=tick, tickfontsize=6)
    plot!(; xguide="α", yguide="probability of type I error",
        guidefontsize=9)

    x = 0:0.0001:0.1
    tick = 0:0.01:1
    Q = plot(; legend=:topleft)
    plot!(x, x -> F_sterne(x); label="Fisher (Sterne)")
    plot!(x, x -> F_clopper_pearson(x); label="Fisher (CP)", ls=:dash)
    plot!(x, x -> F_pearson_chisq(x); label="Pearson χ²", ls=:dashdot)
    plot!(x, x -> F_wald(x); label="Wald", ls=:dot, lw=2)
    plot!(x, identity; label="", c=:black, ls=:dot)
    plot!(; xtick=tick, ytick=tick, tickfontsize=6)
    plot!(; xguide="α", yguide="probability of type I error",
        guidefontsize=9)

    plot(P, Q; size=(700, 350), layout=(1, 2))
    plot!(; plot_title="Binomial(m=$m, p=$p)×Binomial(n=$n, q=$q)",
        plot_titlefontsize=10)
    plot!(; kwargs...)
end

plot_probabilities_of_type_I_error(20, 30, 0.4)
```

Out[19]:

1 比率の比較に関するP値と信頼区間

1.1 比率の比較に関するP値と信頼区間を使って行いたいこと

我々は以下のようなことを知りたいとする.

(1) ウェブページのデザインAとBではどちらの側がどれだけ商品が売れ易いかを知りたい.

(2) 薬Xを処方した側ではしなかった側よりも症状が悪化しなかった人の割合がどれだけ増えるかを知りたい.

例えば, (1)の場合に次のようなデータが得られたとする:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

このデータの数値は, デザインAの側にアクセスした1014人中の49人が商品を購入し, デザインBの側にアクセスした880人中の26人が商品を購入したことを表している.

このデータの数値からの印象では, デザインAの方がデザインBよりも商品の購入確率が高いように見える. 実際, デザインAでの購入者割合は $49/1014 \approx 4.8\%$ 程度で, デザインBでの購入者割合の $26/880 \approx 3.0\%$ より大きい.

しかし, データの確率的揺らぎのせいで単なる偶然で, デザインAの側がよく売れるように見えるデータの数値が得られただけなのかもしれない. (実際には他にも様々な原因で偏ったデータが得られる場合がある.)

もしもそうならば, 上のデータに基くウェブページのデザイン採用に関する意思決定は無駄に終わることになってしまう. だから, そのようなリスクの程度を見積もる必要がある. そのための道具がP値や信頼区間である.

このようなデータに適当な統計モデルを適用して, デザインAとデザインBでの購入確率の違いを信頼区間で表したい.

そのためには以下のようにすればよい:

1. 統計モデルの設定.
2. 購入確率の違いを表す指標を導入.
3. その指標の値が〇〇であるという仮説のP値を定義する.
4. そのP値を使って信頼区間を定義する.
5. 以上によって得られたP値函数や信頼区間を利用する.

1.2 2×2の分割表型データとその2つの二項分布モデル

このノートでは次の形の 2×2 の分割表型のデータを扱う:

	$Y=1$	$Y=0$	
$X=1$	a	b	$m=a+b$
$X=0$	c	d	$n=c+d$
	$r=a+c$	$s=b+d$	$N=a+b+c+d$

ここで a, b, c, d は 0 以上の整数である.

このノートでは2×2の分割表型データに関する以下のような設定の統計モデルを扱う.

(1) 横方向の合計 $m = a + b$, $n = c + d$ (ゆえに全体の合計 $N = m + n$) は固定されていると仮定する.

(2) a は成功確率パラメータ p の二項分布に従っており, c は成功確率パラメータ q の二項分布に従っているとし, a, c は独立であると仮定する. p, q を比率と呼ぶこともある.

このとき, a, b, c, d の同時確率質量函数は次のように表される:

$$P(a, b, c, d|m, n, p, q) = \binom{m}{a} p^a (1-p)^b \binom{n}{c} q^c (1-q)^d$$
$$(a, b, c, d \in \mathbb{Z}_{\geq 0},\ a + b = m,\ c + d = n).$$

これを **2つの二項分布モデル** と呼ぶことにする.

(3) このモデルは上の分割表において, 以下が成立しているという設定になっている:

- $X = 1$ のときには $Y = 1$ となる確率は p で, $Y = 0$ となる確率は $1 - p$ である.
- $X = 0$ のときには $Y = 1$ となる確率は q で, $Y = 0$ となる確率は $1 - q$ である.

そして, $X = 1$ となる人達を m 人選び, $X = 0$ となる人を n 人選んで Y がどうなるかを調べた.

具体的に X, Y として以下のような場合を想定している:

- $X = 1, 0$ はウェブページのデザインがA,Bのどちらであるかを意味し, $Y = 1, 0$ は商品を購入したかしないかを意味する.
- $X = 1, 0$ は薬Xを処方したか否かを意味し, $Y = 1, 0$ は症状が悪化したか否かを意味している. ($Y = 1$ は何らかの基準で症状が悪化したことを意味する.)

注意: このモデルの採用が妥当であるかどうかはデータの数値とは別の情報を使って判断しなければいけない. 例えば, 比較したい母集団とは異なる偏った母集団からの無作為抽出になってしまっている疑いが強い場合には, モデルをデータの偏りを考慮したものに変更するか, データの取得の仕方を変更するか, どちらかのもしくは両方の処置が必要になる.

注意: 分割表の縦方向と横方向の役割を逆転させた場合も数学的には同様である. ただし, データの分析結果の解釈では立場を交換したことによって違いが出るので注意する必要がある.

注意: 以上で説明した2つの二項分布モデル以外に, 多項分布(四項分布)モデルや4つのPoisson分布モデルも考えられるが, 以下では省略する. 実は統計モデルをそのように変更しても以下で説明するP値や信頼区間の計算の仕方はどれも同じになることが知られている. 2×2の分割表に関する統計モデルについては,

- 「条件付き確率分布, 尤度, 推定, 記述統計」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20

の「2×2の分割表の分布」の節を参照せよ.

1.3 比率の違いを表す2つの指標: オッズ比パラメータ OR とリスク比パラメータ RR

比率 $0 < p < 1$ に関する

$$u = \frac{p}{1-p}$$

を **オッズ**(odds)と呼び, これの対数 $\log u = \log(p/(1-p))$ を **対数オッズ** (log odds)と呼ぶ. 逆に p はオッズ u で

$$p = \frac{u}{1+u}$$

と表されるので, 比率 $p = u/(1+u)$ の代わりにオッズ $u = p/(1-p)$ を扱っても同等のことをできる. (この事実は計算を見通り良くするときに役に立つ.)

2つの比率 $0 < p < 1$, $0 < q < 1$ に関する

$$\mathrm{OR} = \frac{p/(1-p)}{q/(1-q)} = \frac{p(1-q)}{(1-p)q}$$

を **オッズ比**(odds ratio, OR)と呼び, これの対数 $\log \mathrm{OR}$ を **対数オッズ比** (log odds ratio)と呼ぶ:

$$\log \mathrm{OR} = \log \frac{p(1-q)}{(1-p)q} = \log \frac{p}{1-p} - \log \frac{q}{1-q}.$$

さらにこのノートでは, 2つの比率 $0 < p < 1$, $0 < q < 1$ に関する

$$\mathrm{RR} = \frac{p}{q}$$

を **リスク比**(risk ratio, RR)と呼び, これの対数 $\log \mathrm{RR}$ を **対数リスク比** (log risk ratio)と呼ぶ:

$$\log \mathrm{RR} = \log \frac{p}{q} = \log p - \log q.$$

もしも, p, q の両方が 0 に近ければ, $1-p \approx 1$, $1-q \approx 1$ という近似を使えるので, リスク比はオッズ比で近似される:

$$1-p \approx 1,\ 1-q \approx 1 \implies \mathrm{RR} \approx \mathrm{OR}.$$

比率 p と q が等しいという条件はオッズ比やリスク比を使って以下のように言い換えられる:

$$p = q \iff \mathrm{OR} = 1 \iff \mathrm{RR} = 1.$$

だから, p と q が等しいという仮説は $\mathrm{OR} = 1$ や $\mathrm{RR} = 1$ のように表現できる. (もちろん, $\log \mathrm{OR} = 0$ や $\log \mathrm{RR} = 0$ とも書ける.)

だから, オッズ比 OR とリスク比 RR は2つの比率 p, q の違いを表すための指標として使える. (もちろん, それらの対数版である $\log \mathrm{OR}$ や $\log \mathrm{RR}$ が 0 を p, q の違いを表すための指標として採用してもよい.)

注意: 他にも **比率の差** $p-q$ も比率の違いを表す重要な指標だが, このノートでは問題と解答例の形式で軽く触れるだけにする予定である.

注意: オッズは賭け事ではよく使われる用語である. 例えばある勝負でAが勝つことに賭けるときに

$$\text{Aが勝つ確率} : \text{Aが負ける確率} = 1 : 4$$

だと思っているなら, あなたは「オッズは 1 対 4 だ」と思っていることになる.

注意: 例えば確率 p, q が「症状悪化の確率」や「死亡する確率」ならば「リスク」という呼び方は非常に適切だろう. しかし上ではそうでない場合も p/q を「リスク比」と呼ぶことにした. 具体的な応用先で, この呼び方に違和感を感じる場合には適宜別の呼び方をするようにして欲しい.

注意: このノートでは比率の違いを表す指標として主にオッズ比を利用する. その理由は. 応用上の理由ではなく, 単に数学的に扱いやすいからである. そして, 以下の注意で説明するように, オッズ比を考えることは, その対数を考えることによって, ロジスティック回帰の話とも関係付けることができる.

注意: p の対数オッズを $x = \log(p/(1-p))$ と書くことにする. このとき, 以下の公式によって,

$$0 < p < 1, \quad 0 < u < \infty, \quad -\infty < x < \infty$$

の3つが一対一に対応する:

$$\begin{aligned}
p &= \frac{u}{1+u} = \frac{1}{1+e^{-x}} = \mathrm{logistic}(x), \\
u &= \frac{p}{1-p} = \exp(x), \\
x &= \log(u) = \log \frac{p}{1-p} = \mathrm{logit}(p).
\end{aligned}$$

x を p に対応させる函数は **ロジスティック函数** と呼ばれ, その逆函数は **ロジット函数** と呼ばれる.

注意: 対数オッズ比を $\beta = \log \mathrm{OR}$ と書き, さらに q の対数オッズを $\alpha = \log(q/(1-q))$ と書くと, p, q の対数オッズはそれぞれ次のように表される:

$$\mathrm{logit}(p) = \log \frac{p}{1-p} = \alpha + \beta, \quad \mathrm{logit}(q) = \log \frac{q}{1-p} = \alpha.$$

これらは次と同値である:

$$p = \mathrm{logistic}(\alpha + \beta), \quad q = \mathrm{logistic}(\alpha).$$

モデルの2つの比率パラメータ p, q をこのように表して, データの数値から α, β の値を推定することを **ロジスティック回帰** (logisti regression)と呼ぶ. (実際にはもっと一般の場合もロジスティック回帰と呼ぶ.)

注意: 対数リスク比を $\beta = \log \mathrm{RR} = \log p - \log q$ と書き, q の対数を $\alpha = \log q$ と書くと, p, q の対数はそれぞれ次のように表される:

$$\log p = \alpha + \beta, \quad \log q = \alpha.$$

これは次と同値である:

$$p = \exp(\alpha + \beta), \quad q = \exp(\alpha).$$

これと上の違いは logistic と exp の違いになっている.

2 Wald版のP値と信頼区間

2.1 Wald版のオッズ比に関するP値と信頼区間の定義

検定したい仮説: 検定したい仮説は

$$\text{オッズ比パラメータは } \mathrm{OR} = \omega \text{ である.}$$

であるとする. ここで $\omega > 0$ は具体的な正の実数である.

正規分布近似: 仮に a, b, c, d が2つの二項分布モデルに従う確率変数ならば, それらの **オッズ比**

$$\widehat{\mathrm{OR}} = \frac{a/b}{c/d} = \frac{ad}{bc}$$

の対数 $\log \widehat{\mathrm{OR}}$ (確率変数としての対数オッズ比)は, 平均がモデルのパラメータの対数オッズ比

$$\log \mathrm{OR} = \log \frac{p(1-q)}{(1-p)q}$$

で, 分散が

$$\widehat{\mathrm{SE}}^2_{\log \widehat{\mathrm{OR}}} = \frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}$$

であるような正規分布に近似的に従うことを示せる. ゆえに,

$$\frac{\log \widehat{\mathrm{OR}} - \log \mathrm{OR}}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}} \sim \mathrm{Normal}(0,1), \text{ approximately.}$$

([「大数の法則と中心極限定理」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)で解説したデルタ法を使えば示せる. この点については後の節で詳しく説明する.)

記号法に関する注意: 以上においてハットがついている $\widehat{\mathrm{OR}}$ や $\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ は確率変数であることに注意せよ. ハットがついていない OR はモデルのパラメータで表された値である.

P値の構成法: 上の近似を使えば, 「オッズ比パラメータは $\mathrm{OR} = \omega$ である」という仮説のP値を次のように定めることができる:

$$\mathrm{pvalue}_{\mathrm{Wald}}(a,b,c,d|\,\mathrm{OR} = \omega) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{\left|\log \widehat{\mathrm{OR}} - \log \omega\right|}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}}\right)\right)$$

ただし, $\log \widehat{\mathrm{OR}}$ と $\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ は **データの数値 a, b, c, d から上で示した式で計算された値である** とする.
$\mathrm{pvalue}_{\mathrm{Wald}}(a,b,c,d|\,\mathrm{OR} = \omega)$ の定義式は

- 標準正規分布に従ってランダムに生成される値の絶対値がデータから計算された値 $\left(\log \widehat{\mathrm{OR}} - \log \mathrm{OR}\right) \Big/ \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の絶対値以上になる確率

を意味している.

対応する信頼区間: このP値の定義に対応する対数オッズ比パラメータ $\log \mathrm{OR}$ に関する信頼度 $1-\alpha$ の信頼区間は次のようになる:

$$\mathrm{confint}^{\log \mathrm{OR}}_{\mathrm{Wald}}(a,b,c,d|\alpha) = \left[\log \widehat{\mathrm{OR}} - z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}},\ \log \widehat{\mathrm{OR}} + z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right]$$

対応するオッズ比パラメータ OR の信頼区間は次のようになる:

$$\mathrm{confint}^{\mathrm{OR}}_{\mathrm{Wald}}(a,b,c,d|\alpha) = \left[\exp\left(-z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right)\widehat{\mathrm{OR}},\ \exp\left(z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right)\widehat{\mathrm{OR}}\right]$$

ここで, $z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2)$ である.

2.2 Wald版のオッズ比に関するP値と信頼区間の計算例

データが次の場合の仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のWald版のP値とオッズ比パラメータ OR の 95% 信頼区間を計算してみよう:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

結果は次のようになる:

- (仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のWald版のP値) ≈ 0.03847
- (オッズ比パラメータ OR のWald版の 95% 信頼区間) ≈ [1.0275, 2.7072]

2.2.1 WolframAlphaによるWald版のオッズ比に関するP値と信頼区間の計算の仕方

$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の計算:

`sqrt(1/a+1/b+1/c+1/d) where a=49.0, b=965, c=26, d=854` → 実行 (https://www.wolframalpha.com/input?i=sqrt%281%2Fa%2B1%2Fb%2B1%2Fc%2B1%2Fd%29+where+a%3D49.0%2C+b%3D965%2C+c%3D26%2C+d%3D854) → 0.247137

$(\log \widehat{\mathrm{OR}} - \log 1)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の計算:

`log(a*d/(b*c))/0.247137 where a=49.0, b=965, c=26, d=854` → 実行 (https://www.wolframalpha.com/input?i=log%28a*d%2F%28b*c%29%29%2F0.247137+where+a%3D49.0%2C+b%3D965%2C+c%3D26%2C+d%3D854) → 2.06981

P値の計算:

`2(1 - cdf(NormalDistrubution(0,1), 2.06981))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistrubution%280%2C1%29%2C+2.06981%29%29) → 0.0384701 (P値)

$\alpha = 0.05$ の場合の $z_{\alpha/2}$ の計算:

`quantile(NormalDistribution(0,1), 0.975))` → 実行 (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.975%29) → 1.95996

95%信頼区間の計算:

`{exp(-0.247137z)*a*d/(b*c), exp(0.247137z)*a*d/(b*c)} where a=49.0, b=965, c=26, d=854, z=1.95996` → 実行 (https://www.wolframalpha.com/input?i=%7Bexp%28-0.247137z%29*a*d%2F%28b*c%29%2C+exp%280.247137z%29*a*d%2F%28b*c%29%7D+where+a%3D49.0%2C+b%3D965%2C → {1.02752, 2.70717}

2.2.2 Julia言語によるWald版のオッズ比に関するP値と信頼区間の計算の仕方(1)

素朴に定義通りにコードを入力すれば計算できる.

In [20]:

```julia
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
ω = 1.0
@show ω
@show ORhat = a*d/(b*c)
@show SEhat = √(1/a + 1/b + 1/c + 1/d)
@show pvalue = 2ccdf(Normal(), abs(log(ORhat) - log(ω))/SEhat)
α = 0.05
@show α
@show z = quantile(Normal(), 1-α/2)
@show confint = [exp(-z*SEhat)*ORhat, exp(z*SEhat)*ORhat];
```

```
(a, b, c, d) = (49, 965, 26, 854)
ω = 1.0
ORhat = (a * d) / (b * c) = 1.6678357911518533
SEhat = √(1 / a + 1 / b + 1 / c + 1 / d) = 0.247137474584997
pvalue = 2 * ccdf(Normal(), abs(log(ORhat) - log(ω)) / SEhat) = 0.03847043006362489
α = 0.05
z = quantile(Normal(), 1 - α / 2) = 1.9599639845400576
confint = [exp(-z * SEhat) * ORhat, exp(z * SEhat) * ORhat] = [1.0275181208761617, 2.70717972727
83877]
```

2.2.3 Julia言語によるWald版のオッズ比に関するP値と信頼区間の計算の仕方(2)

このノートブックの最初の方で定義した函数を使って計算.

In [21]:
```
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
@show pvalue_or_wald(a, b, c, d; ω=1)
@show confint_or_wald(a, b, c, d; α=0.05);
```

```
(a, b, c, d) = (49, 965, 26, 854)
pvalue_or_wald(a, b, c, d; ω = 1) = 0.03847043006362489
confint_or_wald(a, b, c, d; α = 0.05) = [1.0275181208761617, 2.7071797272783877]
```

2.2.4 Julia言語によるWald版のオッズ比に関するP値函数の視覚化

P値函数 $\omega \mapsto \mathrm{pvalue}_{\mathrm{Wald}}(a,b,c,d|\ \mathrm{OR}=\omega)$ のグラフを描いてみよう.

In [22]:
```
a, b, c, d = 49, 965, 26, 854
plot(ω → pvalue_or_wald(a,b,c,d; ω), 0.5, 3.5; label="Wald P-value func.")
vline!([oddsratiohat(a,b,c,d)]; label="ORhat = (ad)/(bc)", ls=:dash)
plot!(; xguide="OR = ω", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
```

Out[22]:

このようにWald版P値函数は, $\omega = \widehat{\mathrm{OR}} = (ad)/(bc)$ で最大値の 1 になり, 「とんがり帽子」型のグラフになる.

2.2.5 R言語でのWald版のオッズ比に関するP値と信頼区間の計算の仕方

定義通りにコードを入力すれば計算できる.

```
a = 49
b = 965
c = 26
d = 854
omega = 1
ORhat = a*d/(b*c)
SEhat = sqrt(1/a + 1/b + 1/c + 1/d)
p.value = 2*(1 - pnorm(abs(log(ORhat) - log(omega))/SEhat))
alpha = 0.05
z = qnorm(1 - alpha/2)
conf.int = c(exp(-z*SEhat)*ORhat, exp(z*SEhat)*ORhat)

cat("data: a b c d = ", a, b, c, d, "\n")
cat("null hypothesis: OR = ", omega, "\n")
cat("ORhat = ", ORhat, "\n")
cat("SEhat = ", SEhat, "\n")
cat("p.value = ", p.value, "\n")
cat("conf.level = ", 1 - alpha, "\n")
cat("conf.int = ", conf.int, "\n")
```

In [23]:

```
R"""
a = 49
b = 965
c = 26
d = 854
omega = 1
ORhat = a*d/(b*c)
SEhat = sqrt(1/a + 1/b + 1/c + 1/d)
p.value = 2*(1 - pnorm(abs(log(ORhat) - log(omega))/SEhat))
alpha = 0.05
z = qnorm(1 - alpha/2)
conf.int = c(exp(-z*SEhat)*ORhat, exp(z*SEhat)*ORhat)

cat("data: a b c d = ", a, b, c, d, "\n")
cat("null hypothesis: OR = ", omega, "\n")
cat("ORhat = ", ORhat, "\n")
cat("SEhat = ", SEhat, "\n")
cat("p.value = ", p.value, "\n")
cat("conf.level = ", 1 - alpha, "\n")
cat("conf.int = ", conf.int, "\n")
""";
```

```
data: a b c d =  49 965 26 854
null hypothesis: OR =  1
ORhat =  1.667836
SEhat =  0.2471375
p.value =  0.03847043
conf.level =  0.95
conf.int =  1.027518 2.70718
```

95% 信頼区間はepiR パッケージを使えば

```
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = epiR::epi.2by2(A, digits=4, conf.level=0.95)
```

で計算できる. 色々表示されるが,

```
Odds ratio                                        1.6678 (1.0275, 2.7072)
```

の行の括弧の内側にWald版の信頼区間が表示されている.

In [24]:

```
R"""
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = epiR::epi.2by2(A, digits=4, conf.level=0.95)
"""
```

Out[24]:

```
RObject{VecSxp}
             Outcome +    Outcome -      Total        Inc risk *        Odds
Exposed +           49          965       1014              4.83      0.0508
Exposed -           26          854        880              2.95      0.0304
Total               75         1819       1894              3.96      0.0412

Point estimates and 95% CIs:
---------------------------------------------------------------------
Inc risk ratio                                  1.6356 (1.0254, 2.6088)
Odds ratio                                      1.6678 (1.0275, 2.7072)
Attrib risk in the exposed *                    1.8778 (0.1475, 3.6081)
Attrib fraction in the exposed (%)             38.8590 (2.4779, 61.6679)
Attrib risk in the population *                 1.0053 (-0.4170, 2.4276)
Attrib fraction in the population (%)          25.3879 (-1.2434, 45.0140)
---------------------------------------------------------------------
Uncorrected chi2 test that OR = 1: chi2(1) = 4.368 Pr>chi2 = 0.037
Fisher exact test that OR = 1: Pr>chi2 = 0.044
 Wald confidence limits
 CI: confidence interval
 * Outcomes per 100 population units
```

2.3 確率変数の対数オッズ比の分散の近似値の導出

この節では a, b, c, d は2つの二項分布モデルに従う確率変数であるとし,

$$\hat{p} = \frac{a}{a+b} = \frac{a}{m}, \quad \hat{q} = \frac{c}{c+d} = \frac{c}{n}$$

とおく. これの対数オッズ比

$$\log \widehat{\mathrm{OR}} = \log \frac{a/b}{c/d} = \log \frac{\hat{p}/(1-\hat{p})}{\hat{q}(1-\hat{q})} = \log \frac{\hat{p}}{1-\hat{p}} + \log \frac{\hat{q}}{1-\hat{q}}$$

の分散の近似値を求めたい. $a \sim \mathrm{Binomial}(m, p)$, $c \sim \mathrm{Binomial}(n, q)$ なので, 二項分布の中心極限定理より,

$$\hat{p} \sim \mathrm{Normal}\left(p, \sqrt{\frac{p(1-p)}{m}}\right), \quad \hat{q} \sim \mathrm{Normal}\left(q, \sqrt{\frac{q(1-q)}{n}}\right), \quad \text{approximately.}$$

a, c は独立なので $\hat{p}, \hat{q}$ も独立になる.

$f(t) = \log(t/(1-t)) = \log t - \log(1-t)$ とおくと, $f'(t) = 1/t - 1/(1-t) = 1/(t(1-t))$ なので,

$$\log \frac{\hat{p}}{1-\hat{p}} = \log \frac{p}{1-p} + \frac{\hat{p}-p}{p(1-p)} + O((\hat{p}-p)^2).$$

これより, デルタ法を使って, すなわち,

$$\log \frac{\hat{p}}{1-\hat{p}} \approx \log \frac{p}{1-p} + \frac{\hat{p}-p}{p(1-p)}$$

と近似することによって, 次の近似が得られる:

$$\log \frac{\hat{p}}{1-\hat{p}} \sim \mathrm{Normal}\left(\log \frac{p}{1-p}, \frac{1}{\sqrt{mp(1-p)}}\right), \quad \text{approximately.}$$

右辺の平均 $\log(p/(1-p))$ は上の近似式の右辺の期待値として得られ($E[\hat{p}] = p$ を使う), 右辺の標準偏差 $1/\sqrt{mp(1-p)}$ は $\hat{p}$ の標準偏差 $\sqrt{p(1-p)/m}$ を $p(1-p)$ で割ることによって得られる.

同様にして,

$$\log \frac{\hat{q}}{1-\hat{q}} \sim \mathrm{Normal}\left(\log \frac{q}{1-q}, \frac{1}{\sqrt{nq(1-q)}}\right), \quad \text{approximately.}$$

デルタ法については

- 「大数の法則と中心極限定理」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)

に解説がある. しかし, 上のように単に一次近似するだけの話に過ぎないので, 自分でよく考えればそちらの解説を読む必要はない可能性が高い.

一般に X, Y が独立な確率変数で $X \sim \mathrm{Normal}(\mu_X, \sigma_X)$, $Y \sim \mathrm{Normal}(\mu_Y, \sigma_Y)$ のとき,
$X + Y \sim \mathrm{Normal}\left(\mu_X + \mu_Y, \sqrt{\sigma_X^2 + \sigma_Y^2}\right)$ となる. そして,

$$\begin{aligned}\left(\frac{1}{\sqrt{mp(1-p)}}\right)^2 + \left(\frac{1}{\sqrt{nq(1-q)}}\right)^2 &= \frac{1}{mp(1-p)} + \frac{1}{nq(1-q)} \\ &= \frac{1}{mp} + \frac{1}{m(1-p)} + \frac{1}{nq} + \frac{1}{n(1-q)}\end{aligned}$$

なので,

$$\mathrm{SE}_{\log \widehat{\mathrm{OR}}} = \sqrt{\frac{1}{mp} + \frac{1}{m(1-p)} + \frac{1}{nq} + \frac{1}{n(1-q)}}$$

とおいて, $\log \widehat{\mathrm{OR}}$ の定義と

$$\log \mathrm{OR} = \log \frac{p/(1-p)}{q/(1-q)} = \log \frac{p}{1-p} - \log \frac{q}{1-q}$$

を使うと,

$$\log \widehat{\mathrm{OR}} \sim \mathrm{Normal}\left(\log \mathrm{OR}, \mathrm{SE}_{\log \widehat{\mathrm{OR}}}\right), \quad \text{approximately}$$

となることがわかる.

もしも $mp, m(1-p), nq, n(1-q)$ のそれぞれを a, b, c, d で近似できるならば, SE の中の $mp, m(1-p), nq, n(1-q)$ のそれぞれを a, b, c, d で置き換えて,

$$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}} = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$$

を作って,

$$\log \widehat{\mathrm{OR}} \sim \mathrm{Normal}\Big(\log \mathrm{OR}, \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\Big), \quad \text{approximately}$$

という近似が得られる. この近似のもとで

$$\frac{\log \widehat{\mathrm{OR}} - \log \mathrm{OR}}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}} \sim \mathrm{Normal}(0, 1), \quad \text{approximately}.$$

注意: 以上のような議論ではハット付きの $\widehat{\mathrm{OR}}$ とハットが付いていない OR の区別が曖昧になると理解できなくなる. ハットが付いていない $\mathrm{OR} = (p/(1-p))/(q/(1-q))$ はモデルのパラメータ p, q で書かれた量であり, ハットが付いている $\widehat{\mathrm{OR}} = (a/b)/(c/d)$ はモデルに従う確率変数 a, b, c, d で書かれた量になっていることに注意すれば区別がし易くなるだろう. 実際にその方法で $\mathrm{SE}_{\log \widehat{\mathrm{OR}}}$ と $\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の区別を容易にできることを確認してみよ.

注意: ハット付きの記号はデータの数値で表された同様の量にもそのまま流用されるので, 文脈によって, 確率変数になっているかと, データの数値から計算された数値になっているかを区別しなければいけない. この問題はモデル内の確率変数とデータの数値から計算さらた数値を記号的にも区別できるようにすれば解決するのだが, 実際にやってみればわかるように, 記号が煩雑過ぎるという点でつらくなる場合が多数出て来る. しかし, 初学者にとって「文脈を利用して正しく解釈すること」は難しい場合があるので, 実際に難しいと感じる人は, 文脈に頼らずに記号を見ただけで区別できるように書かれたノートを **自分で** 作成してみるとよいだろう. (おそらくそれを本当に実行した人は結果的に文脈を利用して正しく解釈することが楽にできるようになるだろう.)

2.4 問題: 対数リスク比の場合の正規分布近似

a, b, c, d は2つの二項分布モデルに従う確率変数であるとし, $\hat{p} = a/m$, $\hat{q} = c/n$ とおき, それらに関する確率変数としての対数リスク比を次のように定める:

$$\log \widehat{\mathrm{RR}} = \log \frac{\hat{p}}{\hat{q}} = \log \hat{p} - \log \hat{q}.$$

このとき,

$$\log \mathrm{RR} = \log \frac{p}{q}, \quad \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}} = \sqrt{\frac{1}{a} - \frac{1}{a+b} + \frac{1}{c} - \frac{1}{c+d}}$$

とおくと, 二項分布の中心極限定理を使って, 次の近似が得られることを示せ:

$$\log \widehat{\mathrm{RR}} \sim \mathrm{Normal}\Big(\log \mathrm{RR}, \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\Big), \quad \text{approximately}.$$

解答例: デルタ法を利用する. $f(t) = \log t$ とおくと, $f'(t) = 1/t$ なので,

$$\log \hat{p} = \log p + \frac{\hat{p} - p}{p} + O((\hat{p} - p)^2).$$

これと二項分布の中心極限定理より近似的に $\hat{p} \sim \mathrm{Normal}\Big(p, \sqrt{p(1-p)/m}\Big)$ となるので次の近似を得る:

$$\log \hat{p} \sim \mathrm{Normal}\left(\log p, \sqrt{\frac{1-p}{mp}}\right), \quad \text{approximately}.$$

同様にして,

$$\log \hat{q} \sim \mathrm{Normal}\left(\log q, \sqrt{\frac{1-q}{nq}}\right), \quad \text{approximately}.$$

ゆえに, $\mathrm{SE}_{\log \widehat{\mathrm{RR}}}$ を

$$
\begin{aligned}
\mathrm{SE}_{\log \widehat{\mathrm{RR}}} &= \sqrt{\frac{1-p}{mp} + \frac{1-q}{nq}} \\
&= \sqrt{\frac{1}{mp} - \frac{1}{m} + \frac{1}{nq} - \frac{1}{n}}
\end{aligned}
$$

と定めると,

$$
\log \widehat{\mathrm{RR}} \sim \mathrm{Normal}\Big(\log \mathrm{RR}, \mathrm{SE}_{\log \widehat{\mathrm{RR}}}\Big), \quad \text{approximately.}
$$

ゆえにもしも, mp, nq のそれぞれを a, c で近似できるならば, $a+b=m$, $c+d=n$ も使って

$$
\log \widehat{\mathrm{RR}} \sim \mathrm{Normal}\Big(\log \mathrm{RR}, \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\Big), \quad \text{approximately.}
$$

が得られる.

解答終

2.5 問題: Wald版のリスク比に関するP値と信頼区間の定義

上の問題の結果を用いてWald版のリスク比に関するP値と信頼区間を適切に定義せよ.

注意: 数学的にややこしい事柄を理解したい場合には「定義だけは最初に覚えなければいけない」のように考えると失敗することが多い. 適切な定義を自分で作れるような考え方をした方がよい. この問題を適切に解けた人は今度は自力でP値と信頼区間を適切に定義して自由自在に使えるようになるだろう.

解答例: 現実世界における調査によって2×2の分割表のデータの数値 a, b, c, d が得られているとする. そのデータの数値の対数リスク比を次のように定める:

$$
\log \widehat{\mathrm{RR}} = \frac{a/(a+b)}{c/(c+d)}.
$$

仮説「リスク比パラメータは $\mathrm{RR} = \rho$ である」の下での2つの二項分布モデル内でランダムに生成されたデータの対数リスク比が現実世界で得たデータの対数リスク比以上に $\mathrm{RR} = \rho$ から離れる確率の近似値として, 「リスク比パラメータは $\mathrm{RR} = \rho$ である」という仮説のP値を定める:

$$
\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d | \, \mathrm{RR} = \rho) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{\left|\log \widehat{\mathrm{RR}} - \log \rho\right|}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}}\right)\right).
$$

ここで1つ前の問題の結果と記号

$$
\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}} = \sqrt{\frac{1}{a} - \frac{1}{a+b} + \frac{1}{c} - \frac{1}{c+d}}
$$

を使った. このP値に定義に対応する対数リスク比パラメータ $\log \mathrm{RR}$ の信頼度 $1-\alpha$ の信頼区間は

$$
\mathrm{confint}_{\mathrm{Wald}}^{\log \mathrm{RR}}(a, b, c, d | \alpha) = \left[\log \widehat{\mathrm{RR}} - z_{\alpha/2} \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}, \ \log \widehat{\mathrm{RR}} + z_{\alpha/2} \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\right]
$$

になり, リスク比パラメータ RR の信頼度 $1-\alpha$ の信頼区間は

$$
\mathrm{confint}_{\mathrm{Wald}}^{\mathrm{RR}}(a, b, c, d | \alpha) = \left[\exp\left(-z_{\alpha/2} \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\right) \widehat{\mathrm{RR}}, \ \exp\left(z_{\alpha/2} \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\right) \widehat{\mathrm{RR}}\right]
$$

になる. ここで $z_{\alpha/2} = \mathrm{cdf}(\mathrm{Normal}(0,1), 1-\alpha/2)$.

2.6 問題: Wald版のリスク比に関するP値と信頼区間の計算例

データが次の場合のWald版の仮説「リスク比パラメータは $\mathrm{RR} = 1$ である」のP値とリスク比パラメータ RR の 95% 信頼区間を計算してみよ:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

2.6.1 Julia言語によるWald版のリスク比に関するP値と信頼区間の計算の仕方

In [25]:
```
a, b, c, d = 49, 965, 26, 854
@show riskratiohat(a, b, c, d)
@show pvalue_rr_wald(a, b, c, d; ρ=1)
@show confint_rr_wald(a, b, c, d; α = 0.05);
```

```
riskratiohat(a, b, c, d) = 1.635563647397967
pvalue_rr_wald(a, b, c, d; ρ = 1) = 0.038895117565198624
confint_rr_wald(a, b, c, d; α = 0.05) = [1.025408984412048, 2.6087819449169145]
```

2.6.2 Julia言語によるWald版のリスク比に関するP値函数の視覚化

In [26]:
```
a, b, c, d = 49, 965, 26, 854
plot(ρ → pvalue_rr_wald(a,b,c,d; ρ), 0.5, 3.5; label="Wald P-value func.")
vline!([riskratiohat(a,b,c,d)]; label="RRhat = (a/m)/(c/n)", ls=:dash)
plot!(; xguide="RR = ρ", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, m, c, n = $a, $(a+b), $c, $(c+d)")
```

Out[26]:

2.6.3 R言語によるWald版のリスク比に関する信頼区間の計算の仕方

epiRパッケージをRにインストールしておけば,

```
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = epiR::epi.2by2(A, digits=4, conf.level=0.95)
```

でWald版のリスク比に関する信頼区間を計算できる. 色々表示されているが,

```
Inc risk ratio                                  1.6356 (1.0254, 2.6088)
```

の行の括弧の中に信頼区間が表示されている.

In [27]:
```
R"""
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = epiR::epi.2by2(A, digits=4, conf.level=0.95)
"""
```

Out[27]:
```
RObject{VecSxp}
              Outcome +    Outcome -      Total        Inc risk *        Odds
Exposed +            49          965       1014              4.83      0.0508
Exposed -            26          854        880              2.95      0.0304
Total                75         1819       1894              3.96      0.0412

Point estimates and 95% CIs:
---------------------------------------------------------------------
Inc risk ratio                                   1.6356 (1.0254, 2.6088)
Odds ratio                                       1.6678 (1.0275, 2.7072)
Attrib risk in the exposed *                     1.8778 (0.1475, 3.6081)
Attrib fraction in the exposed (%)              38.8590 (2.4779, 61.6679)
Attrib risk in the population *                  1.0053 (-0.4170, 2.4276)
Attrib fraction in the population (%)           25.3879 (-1.2434, 45.0140)
---------------------------------------------------------------------
Uncorrected chi2 test that OR = 1: chi2(1) = 4.368 Pr>chi2 = 0.037
Fisher exact test that OR = 1: Pr>chi2 = 0.044
 Wald confidence limits
 CI: confidence interval
 * Outcomes per 100 population units
```

2.6.4 Wald版のリスク比に関するP値と信頼区間の計算例の問題の解答例

- (仮説「リスク比パラメータは $\mathrm{RR} = 1$ である」のWald版のP値) ≈ 0.03890
- (リスク比パラメータ RR のWald版の 95% 信頼区間) ≈ [1.0254, 2.6088]

注意: この場合にはリスク比がオッズ比で近似される場合になっている.

In [28]:
```
a, b, c, d = 49, 965, 26, 854
@show riskratiohat(a, b, c, d)
@show oddsratiohat(a, b, c, d)
@show pvalue_rr_wald(a, b, c, d; ρ=1)
@show pvalue_or_wald(a, b, c, d; ω=1)
@show confint_rr_wald(a, b, c, d; α = 0.05)
@show confint_or_wald(a, b, c, d; α = 0.05);
```

```
riskratiohat(a, b, c, d) = 1.635563647397967
oddsratiohat(a, b, c, d) = 1.6678357911518533
pvalue_rr_wald(a, b, c, d; ρ = 1) = 0.038895117565198624
pvalue_or_wald(a, b, c, d; ω = 1) = 0.03847043006362489
confint_rr_wald(a, b, c, d; α = 0.05) = [1.025408984412048, 2.6087819449169145]
confint_or_wald(a, b, c, d; α = 0.05) = [1.0275181208761617, 2.7071797272783877]
```

2.7 問題: 比率の差の場合の正規分布近似

a, b, c, d は2つの二項分布モデルに従う確率変数であるとし, $\hat{p} = a/m$, $\hat{q} = c/n$ とおき, $\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}}$ を

$$\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{m} + \frac{\hat{q}(1-\hat{q})}{n}}$$

と定める. このとき, 二項分布の中心極限定理から, 次の近似が得られることを示せ:

$$\hat{p} - \hat{q} \sim \mathrm{Normal}\left(p - q, \widehat{\mathrm{SE}}_{\hat{p}-\hat{q}}\right) \quad \text{approximately.}$$

解答例: 二項分布の中心極限定理によって,

$$\hat{p} \sim \mathrm{Normal}\left(p, \sqrt{\frac{p(1-p)}{m}}\right), \quad \hat{q} \sim \mathrm{Normal}\left(q, \sqrt{\frac{q(1-q)}{n}}\right), \quad \text{approximately.}$$

$\hat{p}, \hat{q}$ が独立であることと, 正規分布に従う独立な確率変数達の和が分散がそれらの分散の和であるような正規分布に従うことより,

$$\mathrm{SE}_{\hat{p}-\hat{q}} = \sqrt{\frac{p(1-p)}{m} + \frac{q(1-q)}{n}}$$

とおくと,

$$\hat{p} - \hat{q} \sim \mathrm{Normal}\big(q, \mathrm{SE}_{\hat{p}-\hat{q}}\big), \quad \text{approximately.}$$

ゆえに, p, q が $\hat{p}, \hat{q}$ で近似できているならば,

$$\hat{p} - \hat{q} \sim \mathrm{Normal}\Big(q, \widehat{\mathrm{SE}}_{\hat{p}-\hat{q}}\Big), \quad \text{approximately.}$$

解答終

2.8 問題: Wald版の比率の差に関するP値と信頼区間の定義

上の問題の結果を利用して, 仮説「$p - q = \Delta$」のP値とパラメータ $p - q$ の信頼区間を適切に定義せよ.

解答例: 仮説「$p - q = \Delta$」のP値を

$$\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}} = \sqrt{\frac{\hat{p}(1-\hat{p})}{m} + \frac{\hat{q}(1-\hat{q})}{n}}$$

を用いて,

$$\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d | p - q = \Delta) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{|\hat{p} - \hat{q} - \Delta|}{\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}}}\right)\right)$$

と定める. 対応する信頼度 $1 - \alpha$ の信頼区間は次のようになる.

$$\mathrm{confint}^{p-q}_{\mathrm{Wald}}(a, b, c, d | \alpha) = \left[\hat{p} - \hat{q} - z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}},\ \hat{p} - \hat{q} + z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{p}-\hat{q}}\right].$$

ただし $z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2)$ (標準正規分布において $z_{\alpha/2}$ 以上になる確率は $\alpha/2$)である.

解答終

2.9 問題: Wald版の比率の差に関するP値と信頼区間の計算例

データが次の場合の仮説「$p - q = 0$」のP値と $p - q$ のWald版の 95% 信頼区間を計算してみよ:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

2.9.1 Julia言語によるWald版の比率の差に関するP値と信頼区間の計算の仕方

以下のようにほぼ定義通りにコードを書けば計算できる.

```
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

riskdiffhat(a, b, c, d) = safediv(a, a+b) - safediv(c, c+d)

function stderr_riskdiffhat(a, b, c, d)
    m, n = a+b, c+d
    p̂, q̂ = safediv(a, m), safediv(c, n)
    √(safediv(p̂*(1-p̂), m) + safediv(q̂*(1-q̂), n))
end

function pvalue_rd_wald(a, b, c, d; Δ=0)
    RDhat = riskdiffhat(a, b, c, d)
    SEhat_riskdiffhat = stderr_riskdiffhat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(RDhat - Δ), SEhat_riskdiffhat))
end

function confint_rd_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    RDhat = riskdiffhat(a, b, c, d)
    SEhat_riskdiffhat = stderr_riskdiffhat(a, b, c, d)
    [RDhat - z*SEhat_riskdiffhat, RDhat + z*SEhat_riskdiffhat]
end
```

In [29]:

```
a, b, c, d = 49, 965, 26, 854
@show pvalue_rd_wald(a, b, c, d; Δ=0)
@show confint_rd_wald(a, b, c, d; α=0.05);
```

```
pvalue_rd_wald(a, b, c, d; Δ = 0) = 0.033414467036392496
confint_rd_wald(a, b, c, d; α = 0.05) = [0.0014752114457742202, 0.03608082226410564]
```

2.9.2 Wald版の比率の差に関するP値と信頼区間の計算の解答例

仮説「$p-q=0$」のP値: 0.03341 $p-q$ の 95% 信頼区間: [0.001475, 0.03608]

2.10 警告: 以上で紹介したWald版の比率の差に関するP値と信頼区間の精度は低い

以上で紹介したWald版の比率の差に関するP値と信頼区間の精度は低いので注意せよ. 改良の仕方が

- Kenneth J. Rothman, Sander Greenland, and Timothy L. Lash. Modern Epistemology, Third Edition, 2008, 888 pages

の Chapter 14. Instroduction to Categorical Statistics の Two Study Groups: Large-Sample Methods, pp.299-300 に書いてある. そこでは, 次の文献が引用されている:

- Guangyong Zou and Allan Donner, A simple alternative confidence interval for the difference between two proportions, Controlled Clinical Trials, Volume 25, Issue 1, February 2004, Pages 3-12. https://doi.org/10.1016/j.cct.2003.08.010 (https://doi.org/10.1016/j.cct.2003.08.010)

この論文の方法による比率の差の信頼区間と対応するP値函数の実装例が

- probability of alpha error of Zou-Donner.ipynb (https://github.com/genkuroki/public/blob/main/0033/probability%20of%20alpha%20error%20of%20Zou-Donner.ipynb)

にある. そこでは比率の差に関するWald型P値函数とZou-Donner型P値函数の第一種の過誤の確率を比較している. 確かにZou-Donnerの方法の方が優れているようだ.

3 オッズ比とリスク比に関するWald版のP値と信頼区間の必修計算問題

3.1 オッズ比に関するWald版のP値と信頼区間の計算の仕方

次の形のデータを扱う.

	$Y=1$	$Y=0$
$X=1$	a	b
$X=0$	c	d

「オッズ比パラメータは $\mathrm{OR} = \omega$ である」という仮説のWald版P値の計算の仕方は以下の通り.

(1) データの数値のオッズ比 $\widehat{\mathrm{OR}}$ を次の公式で求める:

$$\widehat{\mathrm{OR}} = \frac{a/b}{c/d} = \frac{ad}{bc}.$$

(2) $\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ を次の公式で求める:

$$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}} = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}.$$

(3) $(\log \widehat{\mathrm{OR}} - \log \omega)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ を計算する.

(4) P値は次の公式で求まる:

$$\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d|\, \mathrm{OR} = \omega) = 2\left(1 - \mathrm{cdf}\Big(\mathrm{Normal}(0,1), \mathrm{abs}(\log \widehat{\mathrm{OR}} - \log \omega)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}})\Big)\right).$$

オッズ比パラメータ OR に関するWald版の信頼度 $1-\alpha$ の信頼区間の計算の仕方は以下の通り.

(5) $z_{\alpha/2}$ を次の公式で求める:

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2).$$

(6) 信頼区間は次の公式で求まる:

$$\mathrm{confint}^{\mathrm{OR}}_{\mathrm{Wald}}(a, b, c, d|\alpha) = \left[\exp\Big(-z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\Big)\widehat{\mathrm{OR}},\ \exp\Big(z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\Big)\widehat{\mathrm{OR}}\right].$$

3.2 リスク比に関するWald版のP値と信頼区間の計算の仕方

この構成法の導出については上の方にある問題を見よ.

次の形のデータを扱う.

	$Y=1$	$Y=0$	
$X=1$	a	b	$a+b=m$
$X=0$	c	d	$c+d=n$

「リスク比パラメータは $\mathrm{RR} = \rho$ である」という仮説のWald版P値の計算の仕方は以下の通り.

(1) データの数値のsリスク比 $\widehat{\mathrm{RR}}$ を次の公式で求める:

$$\widehat{\mathrm{RR}} = \frac{a/(a+b)}{c/(c+d)} = \frac{a(c+d)}{(a+b)c}.$$

(2) $\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}$ を次の公式で求める:

$$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}} = \sqrt{\frac{1}{a} - \frac{1}{a+b} + \frac{1}{c} - \frac{1}{c+d}}.$$

(3) $(\log \widehat{\mathrm{RR}} - \log \rho)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}$ を計算する.

(4) P値は次の公式で求まる:

$$\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d|\, \mathrm{RR} = \rho) = 2\left(1 - \mathrm{cdf}\Big(\mathrm{Normal}(0,1), \mathrm{abs}(\log \widehat{\mathrm{RR}} - \log \rho)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}})\Big)\right).$$

リスク比パラメータ RR に関するWald版の信頼度 $1-\alpha$ の信頼区間の計算の仕方は以下の通り.

(5) $z_{\alpha/2}$ を次の公式で求める:

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2).$$

(6) 信頼区間は次の公式で求まる:

$$\mathrm{confint}^{\mathrm{RR}}_{\mathrm{Wald}}(a, b, c, d|\alpha) = \left[\exp\Big(-z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\Big)\widehat{\mathrm{RR}},\ \exp\Big(z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}\Big)\widehat{\mathrm{RR}}\right].$$

3.3 必修問題: Wald版のオッズ比とリスク比のP値と信頼区間の計算例

	症状悪化	悪化せず	合計
薬Iを処方	100	579	679
プラセボ	111	568	679
合計	211	1147	1358

この表は次の論文

- Gilmar Reis, M.D., Ph.D. et al., Effect of Early Treatment with Ivermectin among Patients with Covid-19, N Engl J Med 2022; 386:1721-1731. [link (https://www.nejm.org/doi/full/10.1056/NEJMoa2115869)]

の Table 2 から Intention-to-treat population (https://www.google.com/search?q=Intention-to-treat)の部分だけを抜き出したものである. 「症状悪化」の正確な意味などに関する詳しい説明は論文を参照せよ.

上のデータについて, Wald版の場合について以下を求めよ:

(1) 仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のP値

(2) 仮説「オッズ比パラメータは $\mathrm{OR} = 0.5$ である」のP値

(3) オッズ比パラメータ OR の 95% 信頼区間

(4) 仮説「リスク比パラメータは $\mathrm{RR} = 1$ である」のP値

(5) 仮説「リスク比パラメータは $\mathrm{RR} = 0.5$ である」のP値

(6) リスク比パラメータ OR の 95% 信頼区間

3.4 WolframAlphaによる必修問題の解答例

(1)

$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の計算:

`sqrt(1/a+1/b+1/c+1/d) where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=sqrt%281%2Fa%2B1%2Fb%2B1%2Fc%2B1%2Fd%29+where+a%3D100.0%2C+b%3D579%2C+c%3D111%2C+d%3D568) → 0.149989

$(\log \widehat{\mathrm{OR}} - \log 1)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の計算:

`log(a*d/(b*c))/0.149989 where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=log%28a*d%2F%28b*c%29%29%2F0.149989+where+a%3D100.0%2C+b%3D579%2C+c%3D111%2C+d%3D568) → -0.823668

P値の計算:

`2(1 - cdf(NormalDistrubution(0,1), 0.823668))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistrubution%280%2C1%29%2C+0.823668%29%29) → 0.410128 ((1)のP値)

(2)

$(\log \widehat{\mathrm{OR}} - \log 0.5)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ の計算:

`(log(a*d/(b*c)) - log(0.5))/0.149989 where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=%28log%28a*d%2F%28b*c%29%29+-+log%280.5%29%29%2F0.149989+where+a%3D100.0%2C+b%3D579%2C+c%3D111%2C+d%3D568) → 3.79765

P値の計算:

`2(1 - cdf(NormalDistrubution(0,1), 3.79765))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistrubution%280%2C1%29%2C+3.79765%29%29) → 0.000146074 ((2)のP値)

(3)

$\alpha = 0.05$ のときの $z_{\alpha/2}$ の計算:

`quantile(NormalDistribution(0,1), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.975%29) → 1.95996

オッズ比パラメータの信頼区間の計算:

`{exp(-0.149989z)*a*d/(b*c), exp(0.149989z)*a*d/(b*c)} where a=100.0, b=579, c=111, d=568, z=1.95996` → 実行 (https://www.wolframalpha.com/input?i=%7Bexp%28-0.149989z%29*a*d%2F%28b*c%29%2C+exp%280.149989z%29*a*d%2F%28b*c%29%7D+where+a%3D100.0%2C+b%3D579%2C
→ {0.658683, 1.18582} ((3)の信頼区間)

(4)

$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}$ の計算:

`sqrt(1/a-1/(a+b)+1/c-1/(c+d)) where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=sqrt%281%2Fa-1%2F%28a%2Bb%29%2B1%2Fc-1%2F%28c%2Bd%29%29+where+a%3D100.0%2C+b%3D579%2C+c%3D111%2C+d%3D568) → 0.126742

$(\log \widehat{\mathrm{RR}} - \log 1)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}$ の計算:

`log(a*(c+d)/((a+b)*c))/0.126742 where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=log%28a*%28c%2Bd%29%2F%28%28a%2Bb%29*c%29%29%2F0.126742+where+a%3D100.0%2C+b%3D579%2C+c%3D111%
→ -0.823405

P値の計算:

`2(1 - cdf(NormalDistrubution(0,1), 0.823405))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistrubution%280%2C1%29%2C+0.823405%29%29) → 0.410278 ((4)のP値)

(5)

$(\log \widehat{\mathrm{RR}} - \log 0.5)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{RR}}}$ の計算:

`(log(a*(c+d)/((a+b)*c)) - log(0.5))/0.126742 where a=100.0, b=579, c=111, d=568` → 実行 (https://www.wolframalpha.com/input?i=%28log%28a*%28c%2Bd%29%2F%28%28a%2Bb%29*c%29%29+-+log%280.5%29%29%2F0.126742+where+a%3D100.0%2C+b%3D579%2C+c%3D111%2C+d%3D568) → 4.64556

P値の計算:

`2(1 - cdf(NormalDistrubution(0,1), 4.64556))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistrubution%280%2C1%29%2C+4.64556%29%29) → 3.39155×10^-6 = 3.39155e-6 ((5)のP値)

このP値は極めて小さい. 2つの二項分布モデルに関するリスク比が 0.5 であるという仮説はデータとの整合性がほとんどないと考えられる. (この場合にリスク比が 0.5 であることは症状が悪化することが半分に減るという意味になるが, モデル内におけるその想定はP値が小さ過ぎるという理由で否定される.)

(6)

$\alpha = 0.05$ のときの $z_{\alpha/2}$ の計算:

`quantile(NormalDistribution(0,1), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28NormalDistribution%280%2C1%29%2C+0.975%29) → 1.95996

リスク比パラメータの信頼区間の計算:

`{exp(-0.126742z)*a*(c+d)/((a+b)*c), exp(0.126742z)*a*(c+d)/((a+b)*c)} where a=100.0, b=579, c=111, d=568, z=1.95996` → 実行 (https://www.wolframalpha.com/input?i=%7Bexp%28-0.126742z%29*a*%28c%2Bd%29%2F%28%28a%2Bb%29*c%29%2C+exp%280.126742z%29*a*%28c%2Bd%29%2F%28%28a%
→ {0.702739, 1.15494} ((6)の信頼区間)

3.5 Julia言語による必修問題の解答例

Julia言語のシンプルな使い方については

- Julia言語を使った統計学の勉強の仕方 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-2%20How%20to%20use%20Julia%20language%20for%20learning%20statistics.ipynb)

を参照せよ.

左から 0 をかけると常に 0 になり, 左から Inf をかけると常に Inf になる掛け算と分子が 0 ならば何で割っても 0 になり, 分母が無限大なら分子が何であっても 0 になる割り算の定義:

```
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y
```

これを使うと浮動小数点数に関する計算が少し安全になる場合がある.

`0/0` は非数 `NaN` になる. 計算の途中で `NaN` が生じることを防ぎたい. `1/0` は無限大 `Inf` になる. `Inf` は悪さをせずに済むことがある. ただし, `Inf - Inf` や `Inf/Inf` も `NaN` になるので注意が必要である.

In [30]:
```
@show 0/0 1/0 Inf-Inf Inf/Inf;
```

```
0 / 0 = NaN
1 / 0 = Inf
Inf - Inf = NaN
Inf / Inf = NaN
```

オッズ比に関するWald版のP値と信頼区間を計算する函数

ほぼ定義通りにコードが記述されている.

```
oddsratiohat(a, b, c, d) = safediv(a*d, b*c)
stderr_logoddsratiohat(a, b, c, d) = √(1/a + 1/b + 1/c + 1/d)

function pvalue_or_wald(a, b, c, d; ω=1)
    logORhat = log(oddsratiohat(a, b, c, d))
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logORhat - log(ω)), SEhat_logORhat))
end

function confint_or_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    ORhat = oddsratiohat(a, b, c, d)
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logORhat), ORhat), safemul(exp(z*SEhat_logORhat), ORhat)]
end
```

リスク比に関するWald版のP値と信頼区間を計算する函数

ほぼ定義通りにコードが記述されている.

```
riskratiohat(a, b, c, d) = safediv(a*(c+d), (a+b)*c)
stderr_logriskratiohat(a, b, c, d) = √(1/a - 1/(a+b) + 1/c - 1/(c+d))

function pvalue_rr_wald(a, b, c, d; ρ=1)
    logRRhat = log(riskratiohat(a, b, c, d))
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logRRhat - log(ρ)), SEhat_logRRhat))
end

function confint_rr_wald(a, b, c, d; α=0.05)
    (a+b==0 || c+d==0 ) && return [0, Inf]
    z = quantile(Normal(), 1-α/2)
    RRhat = riskratiohat(a, b, c, d)
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logRRhat), RRhat), safemul(exp(z*SEhat_logRRhat), RRhat)]
end
```

In [31]:
```
a, b, c, d = 100, 579, 111, 568
```

Out[31]:
```
(100, 579, 111, 568)
```

In [32]:
```
# (1)
@show pvalue_or_wald(a, b, c, d; ω=1.0);
```

```
pvalue_or_wald(a, b, c, d; ω = 1.0) = 0.4101283497257791
```

In [33]:
```
# (2)
@show pvalue_or_wald(a, b, c, d; ω=0.5);
```

```
pvalue_or_wald(a, b, c, d; ω = 0.5) = 0.0001460722899582654
```

In [34]:
```
# (3)
@show confint_or_wald(a, b, c, d; α=0.05);
```

```
confint_or_wald(a, b, c, d; α = 0.05) = [0.6586822480003675, 1.1858168777394855]
```

In [35]:
```
# (4)
@show pvalue_rr_wald(a, b, c, d; ρ=1.0);
```

```
pvalue_rr_wald(a, b, c, d; ρ = 1.0) = 0.4102771459204757
```

In [36]:
```
# (4)
@show pvalue_rr_wald(a, b, c, d; ρ=0.5);
```

```
pvalue_rr_wald(a, b, c, d; ρ = 0.5) = 3.391523737996353e-6
```

In [37]:
```
# (6)
@show confint_rr_wald(a, b, c, d; α=0.05);
```

```
confint_rr_wald(a, b, c, d; α = 0.05) = [0.7027391487578177, 1.1549412533494148]
```

3.6 必修問題解答例

(1) 0.4101

(2) 0.0001461

(3) [0.6587, 1.1858]

(4) 0.4103

(5) 3.392e-6

(6) [0.7027, 1.1549]

4 Pearsonのχ²検定版のP値と信頼区間

4.1 Pearsonのχ²検定版のオッズ比に関するP値と信頼区間の定義

天下り的になってしまい非常に申し訳ないのだが, オッズ比に関するP値のPearsonのχ²検定版を以下のように定義する. 特にP値の構成法の(1)の段階がひどく天下り的である(後でWilks' theoremへの帰着の仕方を説明する). しかし, それ以外の部分については計算の筋道を詳しく書いておくので, 自分の手で計算して確認して欲しい.

そして, さらに, それだけだとどういうことなのか分かりにくいので, 後の方で示す具体的な数値の計算例と以下で説明する各ステップの対応を繰り返し確認して欲しい.

検定したい仮説: 検定したい仮説は

$$\text{オッズ比パラメータは } \mathrm{OR} = \omega \text{ である.}$$

であるとする. ここで $\omega > 0$ は具体的な正の実数である.

仮説が $\mathrm{OR} = 1$ である場合に, 以下で定義するP値はちょうど **独立性に関するPearsonのχ²検定** で使うP値になっている.

P値の構成法: オッズ比パラメータの値に関する仮説のP値のPearsonのχ²検定版の構成法は以下の通り.

(1) 分割表のデータの数値 a, b, c, d に対して, 次を満たす $\delta = \delta(a, b, c, d|\omega)$ を求める:

$$\frac{(a-\delta)(d-\delta)}{(b+\delta)(c+\delta)} = \omega, \quad -\min(b, c) \le \delta \le \min(a, d).$$

これの左側の等式は

$$A = 1 - \omega, \quad B = a + d + \omega(b + c), \quad C = ad - \omega bc$$

とおくと, 次の二次方程式に書き直される:

$$A\delta^2 - B\delta + C = 0.$$

これの解で $-\min(b,c) \leq \delta \leq \min(a,d)$ を満たす側が欲しい $\delta = \delta(a,b,c,d|\omega)$ である. それは次のように表される:

$$\delta = \delta(a,b,c,d|\omega) = \frac{2C}{B + \sqrt{B^2 - 4AC}}.$$

この表示で浮動小数点数の計算をした方が安全でかつ誤差も小さくなり易い.

特に $\omega = 1$ の場合(独立性検定の場合)には, δ を決める方程式は $-(a+b+c+d)\delta + ad - bc = 0$ になるので,

$$\omega = 1 \implies \delta = \delta(a,b,c,d|1) = \frac{ad-bc}{a+b+c+d} = \frac{ad-bc}{N}.$$

(2) $\tilde{a}, \tilde{b}, \tilde{c}, \tilde{d}$ を次のように定める:

$$\tilde{a} = a - \delta, \quad \tilde{b} = b + \delta, \quad \tilde{c} = c + \delta, \quad \tilde{d} = d - \delta.$$

これが実は仮説 $\mathrm{OR} = 1$ 下での期待値の最尤推定値になっていることを後の方の節で説明する.

特に $\omega = 1$ の場合(独立性検定の場合)には次のようになっている:

$$\omega = 1 \implies \begin{cases} \tilde{a} = \dfrac{(a+b)(a+c)}{N} = \dfrac{ms}{N}, & \tilde{b} = \dfrac{(a+b)(b+d)}{N} = \dfrac{mr}{N}, \\ \tilde{c} = \dfrac{(c+d)(a+c)}{N} = \dfrac{ns}{N}, & \tilde{d} = \dfrac{(c+d)(b+d)}{N} = \dfrac{nr}{N}. \end{cases}$$

(3) データの数値に対応するPearsonのχ²統計量 $\chi^2 = \chi^2(a,b,c,d|\delta)$ の値を次のように定める:

$$\begin{aligned} \chi^2 = \chi^2(a,b,c,d|\delta) &= \delta^2 \left(\frac{1}{a-\delta} + \frac{1}{b+\delta} + \frac{1}{c+\delta} + \frac{1}{d-\delta} \right) \\ &= \frac{(a-\tilde{a})^2}{\tilde{a}} + \frac{(b-\tilde{b})^2}{\tilde{b}} + \frac{(c-\tilde{c})^2}{\tilde{c}} + \frac{(d-\tilde{d})^2}{\tilde{d}}. \end{aligned}$$

ここで, $\delta = \delta(a,b,c,d|\omega)$ は a,b,c,d,ω のみを使って計算できる数値なので, この $\chi^2 = \chi^2(a,b,c,d|\delta)$ も a,b,c,d,ω のみを使って計算できる数値になっている.

上の χ^2 に関する後者の表示はPearsonのχ²統計量の一般的な表示

$$\chi^2 = \sum \frac{((\text{観測値}) - (\text{帰無仮説下のモデル内での期待値}))^2}{(\text{帰無仮説下のモデル内での期待値})}$$

の特別な場合になっていることに注意せよ. ただし, 「帰無仮説下のモデル内での期待値」は最尤推定値として求めたものになっている.

特に $\omega = 1$ の場合(独立性検定の場合)には次のようになっている:

$$\chi^2 = \frac{N(ad-bc)^2}{(a+b)(c+d)(a+c)(b+d)} = \frac{N(ad-bc)^2}{mnrs}$$

これは独立性に関するPearsonのχ²統計量の公式として有名である.

(4) 仮説 $\mathrm{OR} = \omega$ の下での統計モデル内で同様に定義されたPearsonのχ²統計量の値がデータから計算した χ^2 の値以上になる確率の近似値としてP値を定義したい. (一般にP値は, 統計モデル内でデータの数値以上に極端な場合が生じる確率またはその近似値として定義されるのであった.)

仮に a,b,c,d が現実から得たデータの数値ではなく, 仮説 $\mathrm{OR} = \omega$ の下での統計モデル内でランダムに生成された仮想的なデータであるとき, (3)のように定義された χ^2 は a,b,c,d が十分に大きなとき(実はそう大きくなくてもよい), 近似的に自由度 1 のχ²分布に従うことを示せる.

このことを使って, 仮説 $\mathrm{OR} = \omega$ のP値を次のように定義する:

$$\mathrm{pvalue}_{\mathrm{Pearson}}(a,b,c,d|\,\mathrm{OR} = \omega) = 1 - \mathrm{cdf}(\mathrm{Chisq}(1), \chi^2).$$

仮説 $\mathrm{OR} = \omega$ に関するデータの数値 a,b,c,d に関するPearsonのχ²統計量の値 $\chi^2 = \chi^2(a,b,c,d|\delta(a,b,c,d|\omega)$ は a,b,c,d,ω だけで計算される数値なので, これでP値が定義されている.

対応する信頼区間: このP値の定義に対応するオッズ比パラメータ OR の信頼度 $1-\alpha$ の信頼区間は次のように定義される:

$$\mathrm{confint}^{\mathrm{OR}}_{\mathrm{Pearson}}(a,b,c,d|\alpha) = \{\, \omega > 0 \mid \mathrm{pvalue}_{\mathrm{Pearson}}(a,b,c,d|\,\mathrm{OR} = \omega) \geq \alpha \,\}.$$

この信頼区間を計算するために使えるシンプルな公式はないように思われる.

信頼区間の計算法A: ω に関する次の方程式の解を2つ求めれば, その2つがオッズ比パラメータ OR の信頼区間の両端の値になる:

$$\mathrm{pvalue}_{\mathrm{Pearson}}(a,b,c,d|\,\mathrm{OR}=\omega)=\alpha.$$

信頼区間の計算法B: まず, $\chi^2_{\nu,\alpha}$ を

$$\chi^2_{\nu,\alpha}=\mathrm{quantile}(\mathrm{Chisq}(\nu),1-\alpha)$$

と定める. そして, $-\min(b,c)<\delta<\min(a,d)$ と次を満たす δ を2つ求める:

$$\chi^2(a,b,c,d|\delta)=\delta^2\left(\frac{1}{a-\delta}+\frac{1}{b+\delta}+\frac{1}{c+\delta}+\frac{1}{d-\delta}\right)=\chi^2_{1,\alpha}$$

そして, それらの δ に対して, ω を

$$\omega=\frac{(a-\delta)(d-\delta)}{(b+\delta)(c+\delta)}$$

によって対応させれば, 信頼区間の両端の値が得られる.

文献: この節の構成は次の論文に書いてある方法の連続補正無し版になっている:

- Jerome Cornfield, A Statistical Problem Arising from Retrospective Studies, Berkeley Symposium on Mathematical Statistics and Probability, 1956: 135-148 (1956) [link (https://projecteuclid.org/ebooks/berkeley-symposium-on-mathematical-statistics-and-probability/Proceedings%20of%20the%20Third%20Berkeley%20Symposium%20on%20Mathematical%20Statistics%20and%20

4.2 Pearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算例

データが次の場合の仮説「オッズ比パラメータは $\mathrm{OR}=1$ である」のPearsonのχ²検定版のP値とオッズ比 OR の 95% 信頼区間を計算してみよう:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

結果は次のようになる:

- (仮説「オッズ比パラメータは $\mathrm{OR}=1$ である」のPearsonのχ²検定版のP値) ≈ 0.03661
- (オッズ比パラメータ OR のPearsonのχ²検定版の 95% 信頼区間) ≈ [1.0318, 2.6957]

信頼区間ついてはJulia言語版の計算例のみを示す. WolframAlphaでこの信頼区間を求めることはかなり面倒である.

4.2.1 WolframAlphaによるPearsonのχ²検定のP値の計算の仕方

$\omega=1$ の場合には, Pearsonのχ²統計量は,

$$\chi^2=\frac{(a+b+c+d)(ad-bc)^2}{(a+b)(c+d)(a+c)(b+d)}$$

となるのであった. これを用いてP値を計算してみよう.

`(a+b+c+d)(ad-bc)^2/((a+b)(c+d)(a+c)(b+d)) where a=49.0, b=965, c=26, d=854` → 実行 (https://www.wolframalpha.com/input?i=%28a%2Bb%2Bc%2Bd%29%28ad-bc%29%5E2%2F%28%28a%2Bb%29%28c%2Bd%29%28a%2Bc%29%28b%2Bd%29%29+where+a%3D49.0%2C+b%3D965%2C- → 4.36824 (独立性に関するPearsonのχ²統計量)

`1 - cdf(ChisqDistribution(1), 4.36824)` → 実行 (https://www.wolframalpha.com/input?i=1+-+cdf%28ChisqDistribution%281%29%2C+4.36824%29) → 0.0366148 (P値)

4.2.2 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算の仕方(1)

$\omega=1$ の場合のオッズ比に関するPearsonのχ²検定版のP値は, 独立性に関するPearsonのχ²検定のP値そのものになる. それを上でやったのと同じ方法で計算する.

信頼区間の側は函数の零点を見つけてくれる函数を使って求めてみよう.

In [38]:

```
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
χ² = (a+b+c+d)*(a*d-b*c)^2/((a+b)*(c+d)*(a+c)*(b+d))
@show χ²
@show pvalue = ccdf(Chisq(1), χ²)
α = 0.05
@show α
confint = find_zeros(0.5, 3.0) do ω
    A, B, C = 1-ω, a+d+ω*(b+c), a*d-ω*b*c
    δ = 2C/(B + √(B^2 - 4A*C))
    χ² = δ^2 * (1/(a-δ) + 1/(b+δ) + 1/(c+δ) + 1/(d-δ))
    ccdf(Chisq(1), χ²) - α
end
@show confint;
```

```
(a, b, c, d) = (49, 965, 26, 854)
χ² = 4.368235947209686
pvalue = ccdf(Chisq(1), χ²) = 0.03661487827607113
α = 0.05
confint = [1.031778110800563, 2.6957054358850328]
```

`f(x, y) do z ... end` 構文については

- https://docs.julialang.org/en/v1/manual/functions/#Do-Block-Syntax-for-Function-Arguments (https://docs.julialang.org/en/v1/manual/functions/#Do-Block-Syntax-for-Function-Arguments)

を参照せよ．`find_zeros` 函数の使い方については，

- https://juliamath.github.io/Roots.jl/stable/#Basic-usage (https://juliamath.github.io/Roots.jl/stable/#Basic-usage)

を参照せよ．

4.2.3 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値と信頼区間の計算の仕方(2)

このノートブックの最初の方で定義した函数を使って計算．

In [39]:

```
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
@show chisqstat_or(a, b, c, d; ω=1)
@show pvalue_or_pearson_chisq(a, b, c, d; ω=1)
@show confint_or_pearson_chisq(a, b, c, d; α=0.05);
```

```
(a, b, c, d) = (49, 965, 26, 854)
chisqstat_or(a, b, c, d; ω = 1) = 4.368235947209686
pvalue_or_pearson_chisq(a, b, c, d; ω = 1) = 0.03661487827607113
confint_or_pearson_chisq(a, b, c, d; α = 0.05) = [1.031778110800563, 2.6957054358850328]
```

4.2.4 Julia言語によるPearsonのχ²検定版のオッズ比に関するP値の視覚化

まず，次の函数のグラフを描いてみよう:

$$\delta \mapsto \chi^2(a,b,c,d|\delta) = \delta^2 \left(\frac{1}{a-\delta} + \frac{1}{b+\delta} + \frac{1}{c+\delta} + \frac{1}{d-\delta} \right).$$

In [40]:
```
a, b, c, d = 49, 965, 26, 854
plot(δ -> _chisqstat_or(a,b,c,d,δ), -12, 15; label="")
plot!(; xguide="δ", yguide="χ²(a,b,c,d|δ)")
title!("a, b, c, d = $a, $b, $c, $d")
```

Out[40]:

このように $\delta \mapsto \chi^2(a,b,c,d|\delta)$ は $\delta = 0$ で最小値 0 になるU型のグラフになる. ゆえに, δ に関する方程式 $\chi^2(a,b,c,d|\delta) = \chi^2_{1,\alpha}$ の $-\min(b,c) < \delta < \min(a,d)$ を満たす解は負値と正値の2つになる.

次に, P値函数 $\omega \mapsto \mathrm{pvalue}(a,b,c,d|\,\mathrm{OR} = \omega)$ のグラフを描いてみよう.

In [41]:
```
a, b, c, d = 49, 965, 26, 854
plot(ω -> pvalue_or_pearson_chisq(a,b,c,d;ω), 0.7, 3.5; label="Pearson χ² P-value func.")
vline!([oddsratiohat(a,b,c,d)]; label="ORhat = (ad)/(bc)", ls=:dash)
plot!(; xguide="OR = ω", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
```

Out[41]:

このようにP値函数 $\omega \mapsto \mathrm{pvalue}(a,b,c,d|\,\mathrm{OR} = \omega)$ は $\omega = \widehat{\mathrm{OR}} = (ad)/(bc)$ で最大値 1 になり, 「とんがり帽子」の形のグラフになる.

Wald版のP値函数のグラフと重ねてプロットすると次のようになる.

In [42]:

```
a, b, c, d = 49, 965, 26, 854
plot(ω → pvalue_or_pearson_chisq(a,b,c,d; ω), 0.7, 3.5;
    label="Pearson χ² P-value func.")
plot!(ω → pvalue_or_wald(a,b,c,d; ω), 0.7, 3.5;
    label="Wald P-value func.", ls=:dash)
plot!(; xguide="OR = ω", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
```

Out[42]:

この場合には a, b, c, d が十分に大きいので, ほぼぴったり重なっている!

このことから, a, b, c, d が十分に大きな場合には, 計算法が複雑なPearsonのχ²検定版のP値と信頼区間を使う必要がないことがわかる. そのような場合にはWald型のP値と信頼区間で実用的には十分である.

P値と 95% 信頼区間も以下のようにほぼ同じになっている:

- P値の比較:
  - (仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のPearsonのχ²検定版のP値) ≈ 3.7%
  - (仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のWald版のP値) ≈ 3.8%
- 信頼区間の比較:
  - (オッズ比パラメータ OR のPearsonのχ²検定版の 95% 信頼区間) ≈ [1.03, 2.70]
  - (オッズ比パラメータ OR のWald版の 95% 信頼区間) ≈ [1.03, 2.71]

数学的注意: 数学の世界では, 定義が全く異なる2つの量がある条件のもとで近似的によく一致するというようなことが起こる. 異なる動機や思想のもとで定義された2つの量が現実世界での応用では「同じ」とみなしてよいだけ近似的によく一致することがある. そのような場合に元の動機や思想にこだわって解釈しようとすることは誤りである.

注意: a, b, c, d が小さい場合には違いが大きくなる場合がある.

In [43]:

```
a, b, c, d = 4, 1, 1, 5
@show pvalue_or_pearson_chisq(a,b,c,d)
@show pvalue_or_wald(a,b,c,d)
println()
@show confint_or_pearson_chisq(a,b,c,d)
@show confint_or_wald(a,b,c,d);
```

```
pvalue_or_pearson_chisq(a, b, c, d) = 0.03568232934330229
pvalue_or_wald(a, b, c, d) = 0.05563253253702121

confint_or_pearson_chisq(a, b, c, d) = [1.1871236326245318, 337.7203835779611]
confint_or_wald(a, b, c, d) = [0.930440993688474, 429.9036722514896]
```

このように a, b, c, d が小さい場合に信頼区間の幅はものすごく大きくなる.

4.2.5 R言語によるPearsonのχ²検定のP値の計算の仕方

独立性に関するPearsonのχ²検定のP値は次のようにして計算できる.

```
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = chisq.test(A, correct=F)
```

In [44]:
```
R"""
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = chisq.test(A, correct=F)
"""
```

Out[44]:
```
RObject{VecSxp}

        Pearson's Chi-squared test

data:  A
X-squared = 4.3682, df = 1, p-value = 0.03661
```

4.3 「オッズ比パラメータはOR=ωである」という仮説下での最尤法との関係

データの数値 a, b, c, d が与えらていると仮定し, $a+b=m$, $c+d=n$ であるとする.

4.3.1 2つの二項分布モデル全体での最尤法

2つの二項分布モデルの対数尤度函数は

$$\begin{aligned}(p,q) \mapsto L &= \log P(a,b,c,d|m,n,p,q) \\ &= \log\left(\binom{m}{a}p^a(1-p)^b\binom{n}{c}q^c(1-q)^d\right) \\ &= a\log p + b\log(1-p) + c\log q + d\log(1-q) + (\text{constant w.r.t. } p,q).\end{aligned}$$

なので,

$$\frac{\partial L}{\partial p} = \frac{a}{p} - \frac{b}{1-p} = \frac{a-(a+b)p}{p(1-p)}.$$

これより, $\partial L/\partial p = 0$ と $p = \hat{p} := a/(a+b) = a/m$ は同値になる. 同様にして, $\partial L/\partial q = 0$ と $q = \hat{q} := c/(c+d) = c/n$ は同値になる. このことより, 対数尤度 L を最大化する p, q は $\hat{p} = a/m$, $\hat{q} = c/n$ になることがわかる.

$p = \hat{p} = a/m$, $q = \hat{q} = c/n$ のときの2つの二項分布モデルでの期待値は

$$m\hat{p} = a, \quad m(1-\hat{p}) = b, \quad n\hat{q} = c, \quad n(1-\hat{q}) = d$$

になる. これはパラメータの動く範囲に制限を付けない2つの二項分布モデルの場合には, パラメータに最尤推定値を代入して得られる確率分布(**予測分布** という)の期待値が, データの値 a, b, c, d そのものになることを意味している.

注意: この性質は, 4つのPoisson分布モデルでも四項分布モデルでも成立している.

4.3.2 条件OR=ωでパラメータの動ける範囲が制限された2つの二項分布モデルでの最尤法

$\omega > 0$ を任意に取って固定する.

条件 $\mathrm{OR} = (p/(1-p))/(q/(1-q)) = \omega$ という条件の下で対数尤度を最大化する問題をLagrangeの未定乗数法を使って解こう. 函数 $(p,q,\delta) \mapsto M$ を次のように定める:

$$\begin{aligned}M &= L - \delta(\log\mathrm{OR} - \log\omega) \\ &= a\log p + b\log(1-p) + c\log q + d\log(1-q) \\ &\quad - \delta(\log p - \log(1-p) - \log q + \log(1-q) - \log\omega) + \text{const}.\end{aligned}$$

このとき,

$$\begin{aligned}\frac{\partial M}{\partial p} &= \frac{a}{p} - \frac{b}{1-p} - \frac{\delta}{p} - \frac{\delta}{1-p} = \frac{a-\delta}{p} - \frac{b+\delta}{1-p}, \\ \frac{\partial M}{\partial q} &= \frac{c}{q} - \frac{d}{1-q} + \frac{\delta}{q} + \frac{\delta}{1-q} = \frac{c+\delta}{q} - \frac{d-\delta}{1-q}, \\ -\frac{\partial M}{\partial \delta} &= \log\mathrm{OR} - \log\omega = \log\frac{p/(1-p)}{q/(1-q)} - \log\omega.\end{aligned}$$

これらがすべて 0 になることと次が成立することは同値である:

$$\frac{p}{1-p} = \frac{a-\delta}{b+\delta}, \quad \frac{q}{1-q} = \frac{c+\delta}{d-\delta}, \quad \mathrm{OR} = \frac{(a-\delta)(d-\delta)}{(b+\delta)(c+\delta)} = \omega.$$

ゆえに

$$\frac{(a-\delta)(d-\delta)}{(b+\delta)(c+\delta)} = \omega, \quad -\min(b,c) < \delta < \min(a,d)$$

を満たすように δ を取り, $\tilde{p}, \tilde{q}$ を

$$\frac{\tilde{p}}{1-\tilde{p}} = \frac{a-\delta}{b+\delta}, \quad \frac{\tilde{q}}{1-\tilde{q}} = \frac{c+\delta}{d-\delta}$$

を満たすように

$$\tilde{p} = \frac{\tilde{a}}{\tilde{a}+\tilde{b}} = \frac{\tilde{a}}{m}, \quad \tilde{q} = \frac{\tilde{c}}{\tilde{c}+\tilde{d}} = \frac{\tilde{c}}{n}$$

と定めれば, $p = \tilde{p}$, $q = \tilde{q}$ が条件 $\mathrm{OR} = \omega$ の下での最尤推定値になる.

そのとき, モデルの期待値は

$$m\tilde{p} = a - \delta, \quad m(1-\tilde{p}) = b + \delta, \quad n\tilde{q} = c + \delta, \quad n(1-\tilde{q}) = d - \delta$$

になる. **これでオッズ比に関するPearsonのχ²検定版のP値の構成法の(1)のステップが本質的に条件 $\mathrm{OR} = \omega$ の下での最尤法そのものであったことがわかった.**

4.3.3 Wilks's theoremの適用

[Wilks' theorem (https://en.wikipedia.org/wiki/Wilks%27_theorem)](https://en.wikipedia.org/wiki/Wilks%27_theorem) について説明しよう.

適当な緩い条件のもとで, パラメータ θ, η を持つ統計モデル $M(\theta,\eta) = D(\theta,\eta)^n$ (ただし $\theta = (\theta_1, \ldots, \theta_\nu)$ は ν 個の独立なパラメータ達であるとする)について以下が成立している.

$X = (X_1, \ldots, X_n)$ はパラメータの値を $\theta = \theta_0$, $\eta = \eta_0$ に設定して得られるモデルの分布 $M(\theta_0, \eta_0)$ 従う確率変数であると仮定する.

パラメータの動ける範囲に制限を付けないモデル $M(\theta,\eta)$ におけるデータ X に関する最尤推定量を $\hat{\theta}, \hat{\eta}$ と表す. ($\hat{\theta}, \hat{\eta}$ はモデル内確率変数になる.)

パラメータの動ける範囲に $\theta = \theta_0$ という制限によってパラメータ η しか動けない統計モデル $M(\theta_0,\eta)$ におけるデータ X に関する最尤推定量を $\tilde{\eta}$ と表す. ($\tilde{\eta}$ もモデル内確率変数になる.)

このとき, n が大きければ, 対数尤度比は自由度 ν のχ²分布に近似的に従う:

$$2\log\frac{(X \text{ に関する } M(\hat{\theta},\hat{\eta}) \text{ の尤度})}{(X \text{ に関する } M(\theta_0,\tilde{\eta}) \text{ の尤度})} \sim \mathrm{Chisq}(\nu), \quad \text{approximately.} \tag{W}$$

これが **Wilksの定理** である.

要するにパラメータの動ける範囲の次元を, $\theta = (\theta_1, \ldots, \theta_\nu)$ を $\theta_0 = (\theta_{0,1}, \ldots, \theta_{0,\nu})$ に固定することによって, ν 次元下げた場合の最尤法を考えると, 自然に自由度 ν のχ²分布に近似的に従う統計量(確率変数)が得られるという仕組みになっている.

注意: 以上では統計モデル $M(\theta,\eta)$ は標本分布の形をしていると仮定したが, その条件は緩められる. 例えばBernoulli分布の標本分布を二項分布で置き換えても, 尤度函数はパラメータに依存しない定数倍を除いて変わらないので, 上と同様の結果が成立する. **注意終**

Wilksの定理(W)の左辺の対数尤度比は $n \to \infty$ で漸近的に同値な別の確率変数に置き換えられる.

例えば, 統計モデル $M(\theta,\eta)$ が2つの二項分布モデルで $\nu = 1$, $\theta = \mathrm{OR}$, $\theta_0 = \omega$, $\eta = q$ のとき,

$$\mathrm{OR} = \frac{p/(1-p)}{q/(1-q)}$$

という条件によって p を定めると, 統計モデルの確率質量函数は次のように書ける:

$$P(x|\theta,\eta) = P(a,b,c,d|\,\mathrm{OR}, q) = \binom{a+b}{a} p^a (1-p)^b \binom{c+d}{c} q^c (1-q)^d.$$

この統計モデルに従う確率変数を $X = (a,b,c,d)$ と書き, $m = a+b$, $n = c+d$, $\hat{p} = a/m$, $\hat{q} = c/n$ とおくと,

$$\hat{\theta} = \widehat{\mathrm{OR}} = \frac{ad}{bc} = \frac{\hat{p}/(1-\hat{p})}{\hat{q}(1-\hat{q})}, \quad \hat{\eta} = \hat{q}$$

となるのであった. さらに,

$$\frac{(a-\delta)(d-\delta)}{(b+\delta)(c+\delta)} = \omega, \quad -\min(b,c) < \delta < \min(a,d)$$

を満たす δ を取り,

$$\tilde{a} = a-\delta, \quad \tilde{b} = b+\delta, \quad \tilde{c} = c+\delta, \quad \tilde{d} = d-\delta, \quad \tilde{p} = \frac{\tilde{a}}{m}, \quad \tilde{q} = \frac{\tilde{c}}{n},$$

とおくと, $\tilde{\eta} = \tilde{q}$ となることから, 以下が成立することがわかる:

$$(X \text{ に関する } M(\hat{\theta},\hat{\eta}) \text{ の尤度}) = \binom{a+b}{a}\left(\frac{a}{m}\right)^a\left(\frac{b}{m}\right)^b\binom{c+d}{c}\left(\frac{c}{n}\right)^c\left(\frac{d}{n}\right)^d,$$
$$(X \text{ に関する } M(\theta_0,\tilde{\eta}) \text{ の尤度}) = \binom{a+b}{a}\left(\frac{\tilde{a}}{m}\right)^a\left(\frac{\tilde{b}}{m}\right)^b\binom{c+d}{c}\left(\frac{\tilde{c}}{n}\right)^c\left(\frac{\tilde{d}}{n}\right)^d$$

となるので,

$$\frac{(X \text{ に関する } M(\hat{\theta},\hat{\eta}) \text{ の尤度})}{(X \text{ に関する } M(\theta_0,\tilde{\eta}) \text{ の尤度})} = \left(\frac{a}{\tilde{a}}\right)^a\left(\frac{b}{\tilde{b}}\right)^b\left(\frac{c}{\tilde{c}}\right)^c\left(\frac{d}{\tilde{d}}\right)^d.$$

ゆえに近似的に自由度 1 のχ²分布に従う対数尤度比は次のようにシンプルな式で表される:

$$\begin{aligned}&2\log\frac{(X \text{ に関する } M(\hat{\theta},\hat{\eta}) \text{ の尤度})}{(X \text{ に関する } M(\theta_0,\tilde{\eta}) \text{ の尤度})}\\ &= 2a\log\frac{a}{\tilde{a}} + 2b\log\frac{b}{\tilde{b}} + 2c\log\frac{c}{\tilde{c}} + 2d\log\frac{d}{\tilde{d}}.\end{aligned}$$

これに, $f(x) = 2x\log(x/\tilde{x})$ のとき, $f'(x) = 2\log(x/\tilde{x}) + 2$, $f''(x) = 2/x$ より,

$$2x\log\frac{x}{\tilde{x}} = 2(x-\tilde{x}) + \frac{(x-\tilde{x})^2}{\tilde{x}} + O((x-\tilde{x})^2)$$

となることを適用すると, $a+b+c+d = \tilde{a}+\tilde{b}+\tilde{c}+\tilde{d}$ より,

$$\begin{aligned}&2a\log\frac{a}{\tilde{a}} + 2b\log\frac{b}{\tilde{b}} + 2c\log\frac{c}{\tilde{c}} + 2d\log\frac{d}{\tilde{d}}\\ &\approx \frac{(a-\tilde{a})^2}{\tilde{a}} + \frac{(b-\tilde{b})^2}{\tilde{b}} + \frac{(c-\tilde{c})^2}{\tilde{c}} + \frac{(d-\tilde{d})^2}{\tilde{d}} = (\text{Pearsonの}\chi^2\text{統計量})\end{aligned}$$

とPearsonのχ²統計量が得られ,

$$(\text{Pearsonの}\chi^2\text{統計量}) \sim \mathrm{Chisq}(1), \quad \text{approximately}$$

が得られる. これがPearsonのχ²検定の基礎になる.

注意: 以上の議論は多項分布モデルや2×2よりもサイズが大きな分割表にも比較的容易に一般化される.

注意: 以上の議論をさらに一般化するとPearsonのχ²統計量の一般化である **スコア統計量** を使ったχ²検定の構成法が得られる.

4.4 Wilks' theoremの数値的確認例

すでに,

- [「検定と信頼区間: 比率の検定と信頼区間」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)

の「おまけ(より進んだ話題): 一般のχ²検定で使うχ²分布の自由度」の節で, 多項分布の場合で自由度が 1 以外のχ²分布を利用する場合の例を示している.

以下では例を少しだけ追加しておこう.

In [45]:
```
function plot_wilks_theorem(logmle1, logmle0, randsample0, tmpsample0, df;
        L=10^5, bin=:auto, kwargs...)
    loglikrat = similar(zeros(), L)
    tmp = [tmpsample0() for _ in 1:nthreads()]
    @threads for i in 1:L
        X = randsample0(tmp[threadid()])
        loglikrat[i] = 2(logmle1(X) - logmle0(X))
    end
    stephist(loglikrat; norm=true, bin, label="log lik. ratio")
    plot!(Chisq(df); label="Chisq(df=$df)", ls=:dash, c=:red)
    plot!(; kwargs...)
end
```

Out[45]: plot_wilks_theorem (generic function with 1 method)

4.4.1 2つの二項分布モデルのパラメータを2つとも固定した場合 (自由度2)

In [46]:
```
plot_wilks_theorem(
    ((a,b,c,d),) -> xlogx(a) + xlogx(b) + xlogx(c) + xlogx(d),
    ((a,b,c,d),) -> begin
        m, n, p = 160, 320, 0.2
        ã = m*p
        b̃ = m*(1-p)
        c̃ = n*p
        d̃ = n*(1-p)
        xlogy(a, ã) + xlogy(b, b̃) + xlogy(c, c̃) + xlogy(d, d̃)
    end,
    tmp -> begin
        m, n, p = 160, 320, 0.2
        a = rand(Binomial(m, p))
        c = rand(Binomial(n, p))
        a, m-a, c, n-c
    end,
    () -> nothing,
    2;
    title="M(OR, q) vs M(1, 0.2) (m=160, n=320, p=0.2)",
    xlim=(-0.3, 10),
    bin=0:0.25:10,
    L=10^6,
)
```

Out[46]:

4.4.2 2つの二項分布モデルの独立性に関する対数尤度比検定の場合 (OR=1とした場合, 自由度1)

In [47]:

```
plot_wilks_theorem(
    ((a,b,c,d),) -> xlogx(a) + xlogx(b) + xlogx(c) + xlogx(d),
    ((a,b,c,d),) -> begin
        N = a + b + c + d
        ã = (a+b)*(a+c)/N
        b̃ = (a+b)*(b+d)/N
        c̃ = (c+d)*(a+c)/N
        d̃ = (c+d)*(b+d)/N
        xlogy(a, ã) + xlogy(b, b̃) + xlogy(c, c̃) + xlogy(d, d̃)
    end,
    tmp -> begin
        m, n, p = 160, 320, 0.2
        a = rand(Binomial(m, p))
        c = rand(Binomial(n, p))
        a, m-a, c, n-c
    end,
    () -> nothing,
    1;
    title="M(OR, q) vs M(1, q) (m=160, n=320, p=0.2)",
    xlim=(-0.3, 5), ylim=(-0.1, 2.5),
    bin=0:0.1:10,
    L=10^6,
)
```

Out[47]:

4.4.3 正規分布モデルのパラメータを2つとも固定した場合 (自由度2)

In [48]:

```
plot_wilks_theorem(
    X -> (normal = fit_mle(Normal, X); loglikelihood(normal, X)),
    X -> loglikelihood(Normal(), X),
    tmp -> rand!(Normal(), tmp),
    () -> zeros(20),
    2;
    title="Normal(μ,σ)ⁿ vs. Normal(0,1)ⁿ (n=20)",
    xlim=(-0.3, 10),
    L=10^6
)
```

Out[48]:

4.4.4 正規分布モデルでσのみを固定した場合

In [49]:
```
plot_wilks_theorem(
    X → (normal = fit_mle(Normal, X); loglikelihood(normal, X)),
    X → loglikelihood(Normal(mean(X), 1), X),
    tmp → rand!(Normal(), tmp),
    () → zeros(20),
    1;
    title="Normal(μ,σ)ⁿ vs. Normal(μ,1)ⁿ (n=20)",
    xlim=(-0.2, 5), ylim=(-0.1, 2.5),
    L=10^6
)
```

Out[49]:

4.4.5 正規分布モデルでμのみを固定した場合 (自由度1)

In [50]:
```
plot_wilks_theorem(
    X → (normal = fit_mle(Normal, X); loglikelihood(normal, X)),
    X → loglikelihood(Normal(0, std(X; mean=0)), X),
    tmp → rand!(Normal(), tmp),
    () → zeros(20),
    1;
    title="Normal(μ,σ)ⁿ vs. Normal(0,σ)ⁿ (n=20)",
    xlim=(-0.2, 5), ylim=(-0.1, 2.5),
    L=10^6
)
```

Out[50]:

4.4.6 ガンマ分布モデルのパラメータを2つとも固定した場合 (自由度2)

In [51]:
```
plot_wilks_theorem(
    X → (gamma = fit_mle(Gamma, X); loglikelihood(gamma, X)),
    X → loglikelihood(Gamma(2, 3), X),
    tmp → rand!(Gamma(2, 3), tmp),
    () → zeros(20),
    2;
    title="Gamma(α, θ)ⁿ vs. Gamma(2, 3)ⁿ (n=20)",
    xlim=(-0.2, 10),
    L=10^6
)
```

Out[51]:

4.4.7 ガンマ分布モデルで形状パラメータαのみを固定した場合 (自由度1)

In [52]:
```
plot_wilks_theorem(
    X → (gamma = fit_mle(Gamma, X); loglikelihood(gamma, X)),
    X → (μ = mean(X); loglikelihood(Gamma(2, μ/2), X)),
    tmp → rand!(Gamma(2, 3), tmp),
    () → zeros(20),
    1;
    title="Gamma(α, μ/α)ⁿ vs. Gamma(2, μ/2)ⁿ (n=20)",
    xlim=(-0.2, 5), ylim=(-0.1, 2.5),
    L=10^6
)
```

Out[52]:

4.5 問題: 「リスク比パラメータはRR=ρである」という仮説下での最尤法

以下示せ.

$\mathrm{RR} = \rho$ という条件の下での, データの数値 a, b, c, d に関する2つの二項分布モデルの最尤推定値 $\tilde{p}, \tilde{q}$ が

$$\tilde{p} = \frac{a-\Delta}{a-\Delta+b}, \quad \tilde{q} = \frac{c+\Delta}{c+\Delta+d}$$

になることを示せ. ただし Δ は次の条件で定める:

$$\frac{(a-\Delta)(c+\Delta+d)}{(a-\Delta+b)(c+\Delta)} = \rho, \quad -c < \Delta < a.$$

注意: パラメータの値をその最尤推定値にしたとき, 2つの二項分布モデルの期待値は次のようになる:

$$\begin{aligned} m\tilde{p} &= \frac{m}{m-\Delta}(a-\Delta), \quad m(1-\tilde{p}) = \frac{m}{m-\Delta}b, \\ n\tilde{q} &= \frac{n}{n+\Delta}(c+\Delta), \quad m(1-\tilde{q}) = \frac{n}{n+\Delta}d. \end{aligned}$$

ただし, $m = a+b$, $n = c+d$ である.

解答例: Lagrangeの未定乗数法をオッズ比を扱った場合と同様に用いる.

函数 $(p, q, \Delta) \mapsto M$ を次のように定める:

$$\begin{aligned} M &= \log P(a,b,c,d|m,n,p,q) - \Delta(\log \mathrm{RR} - \log\rho) \\ &= a\log p + b\log(1-p) + c\log q + d\log(1-q) \\ &- \Delta(\log p - \log q - \log\rho) + \mathrm{const}. \end{aligned}$$

このとき,

$$\begin{aligned} \frac{\partial M}{\partial p} &= \frac{a}{p} - \frac{b}{1-p} - \frac{\Delta}{p} = \frac{a-\Delta}{p} - \frac{b}{1-p}, \\ \frac{\partial M}{\partial q} &= \frac{c}{q} - \frac{d}{1-q} + \frac{\Delta}{q} = \frac{c+\Delta}{q} - \frac{d}{1-q}, \\ -\frac{\partial M}{\partial \Delta} &= \log \mathrm{RR} - \log\rho = \log\frac{p}{q} - \log\rho. \end{aligned}$$

これらがすべて 0 になることと次が成立することは同値である:

$$\frac{p}{1-p} = \frac{a-\Delta}{b}, \quad \frac{q}{1-q} = \frac{c+\Delta}{d}, \quad \mathrm{RR} = \frac{p}{q} = \rho.$$

さらに

$$\frac{p}{1-p} = \frac{a-\Delta}{b}, \quad \frac{q}{1-q} = \frac{c+\Delta}{d}$$

と

$$p = \tilde{p} := \frac{a-\Delta}{a-\Delta+b}, \quad q = \tilde{q} := \frac{c+\Delta}{c+\Delta+d}$$

は同値であり, このとき, $\mathrm{RR} = p/q = \rho$ は

$$\frac{(a-\Delta)(c+\Delta+d)}{(a-\Delta+b)(c+\Delta)} = \rho$$

と同値である. これと $-c < \Delta < a$ を満たす Δ を取って, すぐ上にある式で $\tilde{p}$, $\tilde{q}$ を定義すると, $p = \tilde{p}$, $q = \tilde{q}$ が $\mathrm{RR} = \rho$ という条件の下での最尤推定量になる.

解答終

4.6 問題: Pearsonのχ²検定版のリスク比に関するP値と信頼区間の定義

上の問題の結果を用いてリスク比に関するPearsonのχ²検定版のP値と信頼区間を適切に定義せよ.

注意: 適切な定義を自分で考えることは物事を楽に理解するために必要なことである.

解答例: $m = a+b$, $n = c+d$ とおき, Δ を

$$\frac{(a-\Delta)(n+\Delta)}{(m-\Delta)(c+\Delta)} = \rho, \quad -c < \Delta < a.$$

という条件で定める. このような Δ は

$$A = \rho - 1, \quad B = n - a + \rho(m - c), \quad C = an - \rho mc$$

と置き,

$$\Delta = \frac{2C}{B + \sqrt{B^2 - 4AC}}$$

とおけば求まる. さらに, $\tilde{a}, \tilde{b}, \tilde{c}, \tilde{d}$ を次のように定める:

$$\begin{aligned} \tilde{a} &= \frac{m}{m - \Delta}(a - \Delta), \quad \tilde{b} = \frac{m}{m - \Delta} b, \\ \tilde{c} &= \frac{n}{n + \Delta}(c + \Delta), \quad \tilde{d} = \frac{n}{n + \Delta} d. \end{aligned}$$

このとき, $(\tilde{a}/m)/(\tilde{c}/n) = \rho$ となっていることに注意せよ.

Pearsonのχ²統計量 χ^2 を次のように定める:

$$\chi^2 = \frac{(a - \tilde{a})^2}{\tilde{a}} + \frac{(b - \tilde{b})^2}{\tilde{b}} + \frac{(c - \tilde{c})^2}{\tilde{c}} + \frac{(d - \tilde{d})^2}{\tilde{d}}.$$

上の定義のもとで,

$$\begin{aligned} a - \tilde{a} &= \frac{b\Delta}{m - \Delta}, \quad b - \tilde{b} = \frac{-b\Delta}{m - \Delta}, \\ c - \tilde{c} &= \frac{-d\Delta}{n + \Delta}, \quad d - \tilde{d} = \frac{d\Delta}{n + \Delta} \end{aligned}$$

でかつ

$$\frac{1}{\tilde{a}} + \frac{1}{\tilde{b}} = \frac{(m - \Delta)^2}{m(a - \Delta)b}, \quad \frac{1}{\tilde{c}} + \frac{1}{\tilde{d}} = \frac{(n + \Delta)^2}{n(c + \Delta)d}$$

なので,

$$\frac{(a - \tilde{a})^2}{\tilde{a}} + \frac{(b - \tilde{b})^2}{\tilde{b}} = \frac{b\Delta^2}{m(a - \Delta)}, \quad \frac{(c - \tilde{a})^2}{\tilde{c}} + \frac{(d - \tilde{b})^2}{\tilde{d}} = \frac{d\Delta^2}{n(c + \Delta)}$$

となることより,

$$\chi^2 = \Delta^2 \left(\frac{b}{m(a - \Delta)} + \frac{d}{n(c + \Delta)} \right).$$

仮説「リスク比パラメータは $\mathrm{RR} = \rho$ である」のP値を次のように定めることができる:

$$\mathrm{pvalue}_{\mathrm{Pearson}}(a, b, c, d | \, \mathrm{RR} = \rho) = 1 - \mathrm{cdf}(\mathrm{Chisq}(1), \chi^2).$$

これに対応するリスク比パラメータ RR の信頼度 $1 - \alpha$ の信頼区間は

$$\mathrm{confint}^{\mathrm{RR}}_{\mathrm{Pearson}}(a, b, c, d | \alpha) = \{\rho > 0 \mid \mathrm{pvalue}_{\mathrm{Pearson}}(a, b, c, d | \, \mathrm{RR} = \rho) \geq \alpha \}.$$

になる.

解答終

4.7 問題: Pearsonのχ²検定版のリスク比に関するP値と信頼区間の計算例

データが次の場合の仮説「リスク比パラメータは $\mathrm{RR} = 1$ である」のPearsonのχ²検定版のP値とリスク比パラメータ RR の 95% 信頼区間を計算してみよ:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

4.7.1 Julia言語によるPearsonのχ²検定版のリスク比に関するP値と信頼区間の計算例

In [53]:
```
a, b, c, d = 49, 965, 26, 854
@show oddsratiohat(a, b, c, d)
@show riskratiohat(a, b, c, d)
@show pvalue_rr_wald(a, b, c, d)
@show pvalue_or_wald(a, b, c, d)
@show pvalue_rr_pearson_chisq(a, b, c, d)
@show pvalue_or_pearson_chisq(a, b, c, d)
@show confint_rr_wald(a, b, c, d)
@show confint_or_wald(a, b, c, d)
@show confint_rr_pearson_chisq(a, b, c, d)
@show confint_or_pearson_chisq(a, b, c, d);
```

```
oddsratiohat(a, b, c, d) = 1.6678357911518533
riskratiohat(a, b, c, d) = 1.635563647397967
pvalue_rr_wald(a, b, c, d) = 0.038895117565198624
pvalue_or_wald(a, b, c, d) = 0.03847043006362489
pvalue_rr_pearson_chisq(a, b, c, d) = 0.03661487827607109
pvalue_or_pearson_chisq(a, b, c, d) = 0.03661487827607113
confint_rr_wald(a, b, c, d) = [1.025408984412048, 2.6087819449169145]
confint_or_wald(a, b, c, d) = [1.0275181208761617, 2.7071797272783877]
confint_rr_pearson_chisq(a, b, c, d) = [1.03047149323473, 2.599845901327348]
confint_or_pearson_chisq(a, b, c, d) = [1.031778110800563, 2.6957054358850328]
```

4.7.2 Julia言語によるPearsonのχ²検定版のリスク比に関するP値函数の視覚化

In [54]:
```
a, b, c, d = 49, 965, 26, 854
plot(ρ -> pvalue_rr_pearson_chisq(a,b,c,d; ρ), 0.5, 3.5; label="Pearson χ² P-value func.")
plot!(ρ -> pvalue_rr_wald(a,b,c,d; ρ), 0.5, 3.5; label="Wald P-value func.", ls=:dash)
plot!(; xguide="RR = ρ", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
```

Out[54]:

この場合には a, b, c, d が十分に大きいので, ほぼぴったり重なっている!

4.7.3 Pearsonのχ²検定版のリスク比に関するP値と信頼区間の計算の解答例

- (仮説「リスク比パラメータは $\mathrm{RR} = 1$ である」のPearsonのχ²検定版P値) ≈ 0.03661
- (リスク比パラメータ RR のPearsonのχ²検定版 95% 信頼区間) ≈ [1.0305, 2.5998]

4.8 ニューサンスパラメータの問題に関する注意

我々が想定している統計モデルには p と q という2つのパラメータが含まれている. そこに仮説 $\mathrm{OR} = \omega$ によって制限を課しても, 独立なパラメータの個数は1つしか減らず, たとえば q の側を自由に動けるパラメータとして採用し, 仮説 $\mathrm{OR} = \omega$ によって p の値は q の値から決まると考えることができる.

だから, P値を定義するために必要な「仮説 $\mathrm{OR} = \omega$ の下での統計モデル内でデータの数値以上に極端な値が生じる確率」は, パラメータ q の値を決めるごとに別々に決まる値になってしまう. このような状況のとき, q はニューサンスパラメータ(nuisance parameter, 局外パラメータ, 攪乱パラメータ, 迷惑パラメータ)であるという.

しかし, 「データの数値以上に極端な値」の意味を, Wald的な正規分布近似を使ったり, Pearsonのχ²統計量を使ったりして定義すれば, 「仮説 $\mathrm{OR} = \omega$ の下での統計モデル内でデータの数値以上に極端な値が生じる確率」が, a, b, c, d が十分に大きなとき, 近似的にパラメータ q の値によらなくなることを示せる.

以上で紹介したP値の定義にはこのような非常に巧妙な方法が使われている.

ニューサンスパラメータの問題への対処の仕方にはそれら以外にも, 以下の節で説明する条件付き確率分布を利用する方法がある. その方法による検定は **Fisher検定** (Fisher's test, Fisher exact test)と呼ばれている.

5 Fisher検定版のオッズ比に関するP値と信頼区間

5.1 条件付き確率分布を利用するアイデア

2つの二項分布モデルの確率質量函数は

$$P(a,b,c,d|m,n,p,q) = \binom{m}{a}p^a(1-p)^b\binom{n}{c}q^c(1-q)^d$$
$$(a,b,c,d \in \mathbb{Z}_{\geq 0},\ a+b=m,\ c+d=n).$$

と書けるのであった. この分布を条件 $a+c=r$ (そのとき $b+d=m+n-r$)という条件で制限して得られる条件付き確率分布を求めてみよう. パラメータのオッズ比 OR を

$$\mathrm{OR} = \frac{p(1-q)}{(1-p)q}$$

と定めると, 条件 $a+b=m$, $c+d=n$, $a+c=r$ の下で,

$$\begin{aligned} &a=a, \quad b=m-a, \quad c=r-a, \quad d=n-r+a, \\ &a \geq 0, \quad a \leq m, \qquad a \leq r, \qquad a \geq r-n \end{aligned}$$

が成立しているので, 上の確率質量函数は

$$P(a,b,c,d|m,n,p,q) = \binom{m}{a}\binom{n}{r-a}(1-p)^m q^r (1-q)^{n-r}(\mathrm{OR})^a \propto \binom{m}{a}\binom{n}{r-a}(\mathrm{OR})^a$$

(a に依存しない因子を比例定数扱いした.) ゆえに

$$Z(m,n,r,\mathrm{OR}) = \sum_a \binom{m}{a}\binom{n}{r-a}(\mathrm{OR})^a$$

とおいて, 確率質量函数 $P(a|m,n,r,\mathrm{OR})$ を

$$P(a|m,n,r,\mathrm{OR}) = \frac{1}{Z(m,n,r,\mathrm{OR})}\binom{m}{a}\binom{n}{r-a}(\mathrm{OR})^a \quad (\max(0,r-n) \leq a \leq \min(m,r))$$

と定義すると, この確率質量函数で定義される確率分布が条件 $a+c=r$ で2つの二項分布モデルを制限して得られる条件付き確率分布になる. この確率分布を **Fisherの非心超幾何分布** (Fisher's noncentral hypergeometric distribution)と呼び,

$$\mathrm{FisherNoncentralHypergeometric}(m,n,r,\mathrm{OR})$$

と表すことにする.

このように, 条件 $a+c=r$ によって条件付き確率分布に移れば, モデルのパラメータはオッズ比パラメータ OR だけになる. ニューサンスパラメータ(nuisance parameter, 局外パラメータ, 攪乱パラメータ, 迷惑パラメータ)が消えてなくなるのである!

だから, 2×2の分割表のデータの数値 a,b,c,d が得られたときに, $a+b=m$, $c+d=n$, $a+c=r$, $b+d=m+n-r$ という縦と横方向のすべての合計(すべてのマージン)の値を固定することによって得られる条件付き確率分布(Fisherの非心超幾何分布)に移って, 「データの数値以上に極端な値が生じる確率」を正確に求めることによってP値が定義できることがわかる. P値の定義の方法は二項分布モデルの場合にならえばよい.

この方法には利点と欠点がある.

- **利点:** Wald版やPearsonのχ²検定版と違って, 確率の値を近似計算する必要がなくなった. それによって, 第一種の過誤が起こる確率を確実に有意水準以下にできる.
- **欠点:** 有限離散分布では分布によって生じる値の個数が減ると, 確率を正確に計算する方法で定義されたP値について, 第一種の過誤の確率が有意水準よりも真に小さくなってしまう影響が大きくなる. しかもそれを条件付き確率分布に移ることによって行うと, 分布内で生じ得る値の種類が大幅に減って, 第一種の過誤の確率が有意水準よりも大幅に小さくなってしまうことになる.

第一種の過誤の確率を確実に有意水準以下にしたい場合にはFisher検定は優れた方法になるが, 検出力はその分だけ下がってしまうという欠点がある. 利点と欠点を理解して, 自分の目的に合わせて適切な道具を使えばよいだろう.

注意: Fisherの非心超幾何分布を $\mathrm{OR}=1$ に特殊化すると, 超幾何分布が得られる:

$$P(a|m,n,r,1) = \binom{m+n}{r}^{-1}\binom{m}{a}\binom{n}{r-a} \quad (\max(0,r-n) \le a \le \min(m,r)).$$

この確率は, 赤い玉が m 個, 白い玉が n 個入っている壺から, r 個の玉を無作為に取り出すときに(非復元抽出, r 個をまとめて一度に取り出す様子を想像すればよい), 取り出した r 個の玉の中に含まれる赤い玉の個数が a 個になる確率だと解釈される.

5.2 Sterne型のFisher検定のP値とそれに付随する信頼区間の定義

前節の記号の下で, データの数値 a, b, c, d に対して, $m = a + b$, $n = c + d$, $r = a + c$ とおき, 仮説「オッズ比パラメータは $\mathrm{OR} = \omega$ である」のSterne型Fisher検定のP値を次のように定める:

$$\mathrm{pvalue}_{\mathrm{Sterne}}(a,b,c,d|\,\mathrm{OR}=\omega) = \sum_{P(i|m,n,r,\omega)\le P(a|m,n,r,\omega)} P(i|m,n,r,\omega).$$

これに対応する信頼度 $1-\alpha$ の信頼区間を

$$\mathrm{confint}^{\mathrm{OR}}_{\mathrm{Sterne}}(a,b,c,d|\alpha) = \{\,\omega > 0 \mid \mathrm{pvalue}_{\mathrm{Sterne}}(a,b,c,d|\,\mathrm{OR}=\omega) \ge \alpha\,\}$$

によって定める. もしくは, 実用的には, これの上限と下限に挟まれた区間として, $\mathrm{confint}_{\mathrm{Sterne}}(a,b,c,d|\,\mathrm{OR}=\omega)$ を定義し直したものを使用する. (このノートでの実装ではそうなっている.)

5.3 Clopper-Pearson型のFisher検定のP値とそれに付随する信頼区間

さらに続けて, Clopper-Pearson型Fisher検定のP値を次のように定める:

$$\begin{aligned}&\mathrm{pvalue}_{\mathrm{CP}}(a,b,c,d|\,\mathrm{OR}=\omega)\\ &= \min\begin{pmatrix}1\\ 2\,\mathrm{cdf}(\mathrm{FisherNoncentralHypergeometric}(a+b,c+d,a+c,\omega),a)\\ 2(1-\mathrm{cdf}(\mathrm{FisherNoncentralHypergeometric}(a+b,c+d,a+c,\omega),a-1))\end{pmatrix}\end{aligned}$$

これに対応する信頼度 $1-\alpha$ の信頼区間を

$$\mathrm{confint}^{\mathrm{OR}}_{\mathrm{CP}}(a,b,c,d|\alpha) = \{\,\omega > 0 \mid \mathrm{pvalue}_{\mathrm{CP}}(a,b,c,d|\,\mathrm{OR}=\omega) \ge \alpha\,\}$$

によって定める.

5.4 問題: Fisher検定版のオッズ比に関するP値と信頼区間の計算例

以下のデータについて, Sterne型およびClopper-Pearsonの型の2種類のFisher検定のP値とそれに付随する信頼区間を計算せよ:

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

ただし, 検定する仮説は「オッズ比パラメータ $\mathrm{OR} = 1$ である」であるとし(独立性に関するFisher検定の場合), 求める信頼区間は信頼度 95% のオッズ比パラメータの信頼区間であるとする.

5.4.1 Julia言語によるFisher検定版のオッズ比に関するP値と信頼区間の計算例

In [55]:
```
a, b, c, d = 49, 965, 26, 854
@show oddsratiohat(a, b, c, d)
@show pvalue_or_wald(a, b, c, d)
@show pvalue_or_pearson_chisq(a, b, c, d)
@show pvalue_or_sterne(a, b, c, d)
@show pvalue_or_clopper_pearson(a, b, c, d)
@show confint_or_wald(a, b, c, d)
@show confint_or_pearson_chisq(a, b, c, d)
@show confint_or_sterne(a, b, c, d)
@show confint_or_clopper_pearson(a, b, c, d);
```

```
oddsratiohat(a, b, c, d) = 1.6678357911518533
pvalue_or_wald(a, b, c, d) = 0.03847043006362489
pvalue_or_pearson_chisq(a, b, c, d) = 0.03661487827607113
pvalue_or_sterne(a, b, c, d) = 0.044079014645063516
pvalue_or_clopper_pearson(a, b, c, d) = 0.047080761570760604
confint_or_wald(a, b, c, d) = [1.0275181208761617, 2.7071797272783877]
confint_or_pearson_chisq(a, b, c, d) = [1.031778110800563, 2.6957054358850328]
confint_or_sterne(a, b, c, d) = [1.0202716568785046, 2.7656505208089848]
confint_or_clopper_pearson(a, b, c, d) = [1.0061031380773773, 2.8212565352058787]
```

5.4.2 Julia言語によるFisher²検定のオッズ比に関するP値函数の視覚化

In [56]:
```
a, b, c, d = 49, 965, 26, 854
plot(ω → pvalue_or_sterne(a,b,c,d; ω), 0.5, 3.5; label="Fisher (Sterne)")
plot!(ω → pvalue_or_clopper_pearson(a,b,c,d; ω), 0.5, 3.5; label="Fisher (CP)", ls=:dash)
plot!(ω → pvalue_or_pearson_chisq(a,b,c,d; ω), 0.5, 3.5; label="Pearson χ²", ls=:dashdot)
plot!(ω → pvalue_or_wald(a,b,c,d; ω), 0.5, 3.5; label="Wald", ls=:dot)
plot!(; xguide="OR = ω", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
plot!(size=(600, 300))
```

Out[56]:

Wald型とPearson型のP値函数はほぼぴったり一致している.

2種のFisher検定にP値函数はそれらと一致していないが, 互いに相手を近似しあう関係になっている.

5.4.3 R言語によるFisher検定版のP値と信頼区間の計算の仕方

Sterne型の独立性検定の場合のP値とClopper-Pearson型のオッズ比の信頼区間をR言語の `fisher.test` 函数で求めることができる.

```
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = fisher.test(A)
```

In [57]:
```
R"""
A = matrix(c(49, 965, 26, 854), 2, 2, byrow=T)
result = fisher.test(A)
"""
```

Out[57]:
```
RObject{VecSxp}

        Fisher's Exact Test for Count Data

data:  A
p-value = 0.04408
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 1.006104 2.821251
sample estimates:
odds ratio
  1.667393
```

R言語の `fisher.test` では表示されるP値と信頼区間のあいだに整合性が無くなる場合が出て来てしまう.

In [58]:
```
R"""
result = fisher.test(matrix(c(16, 4, 4, 6), 2, 2, byrow=T))
"""
```

Out[58]:
```
RObject{VecSxp}

        Fisher's Exact Test for Count Data

data:  matrix(c(16, 4, 4, 6), 2, 2, byrow = T)
p-value = 0.04486
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
  0.8602909 44.0017191
sample estimates:
odds ratio
  5.585888
```

仮説「オッズ比パラメータは 1 である」のP値が 5% を切っているのに, 95% 信頼区間に 1 が含まれてしまっている.

この問題は exact2x2 パッケージを利用すれば解決する.

```
exact2x2::exact2x2(matrix(c(16, 4, 4, 6), 2, 2, byrow=T), plot=T)

exact2x2::exact2x2(matrix(c(16, 4, 4, 6), 2, 2, byrow=T), tsmethod="central", plot=T)
```

`exact2x2` 函数では `tsmethod="minlik"` がSterne型を意味し, `tsmethod="central"` がClopper-Pearson型を意味している.

In [59]:

```
R"""
exact2x2::exact2x2(matrix(c(16, 4, 4, 6), 2, 2, byrow=T), plot=T)
"""
```

Out[59]:

```
RObject{VecSxp}

        Two-sided Fisher's Exact Test (usual method using minimum likelihood)

data:  matrix(c(16, 4, 4, 6), 2, 2, byrow = T)
p-value = 0.04486
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
  1.0549 32.6994
sample estimates:
odds ratio
  5.585888
```

In [60]:

```
R"""
exact2x2::exact2x2(matrix(c(16, 4, 4, 6), 2, 2, byrow=T), tsmethod="central", plot=T)
"""
```

Out[60]:

```
RObject{VecSxp}

        Central Fisher's Exact Test

data:  matrix(c(16, 4, 4, 6), 2, 2, byrow = T)
p-value = 0.07742
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
  0.8602909 44.0017191
sample estimates:
odds ratio
  5.585888
```

In [61]:

```
a, b, c, d = 16, 4, 4, 6
plot(ω -> pvalue_or_sterne(a,b,c,d; ω), 0.5, 80; label="Fisher (Sterne)")
plot!(ω -> pvalue_or_clopper_pearson(a,b,c,d; ω), 0.5, 80; label="Fisher (CP)", ls=:dash)
plot!(ω -> pvalue_or_pearson_chisq(a,b,c,d; ω), 0.5, 80; label="Pearson χ²", ls=:dashdot)
plot!(ω -> pvalue_or_wald(a,b,c,d; ω), 0.5, 80; label="Wald", ls=:dot)
plot!(; xguide="OR = ω", yguide="P-value")
plot!(; ytick=0:0.1:1)
title!("a, b, c, d = $a, $b, $c, $d")
plot!(; size=(500, 250))
plot!(; xscale=:log, xtick=([1,2,5,10,20,50], string.([1,2,5,10,20,50])))
```

Out[61]:

5.4.4 Fisher検定版のオッズ比に関するP値と信頼区間の計算の解答例

P値

- (仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のSterne型のFisher検定版P値) ≈ 4.408%
- (仮説「オッズ比パラメータは $\mathrm{OR} = 1$ である」のClopper-Pearson型のFisher検定版P値) ≈ 4.708%

95%信頼区間

- (オッズ比パラメータ OR の 95% Sterne型のFisher検定版の信頼区間) ≈ [1.0202, 2.7657]
- (オッズ比パラメータ OR の 95% Clopper-Pearson型のFisher検定版の信頼区間) ≈ [1.006, 2.8213]

5.5 Fisher検定における第一種の過誤の確率

「オッズ比パラメータは $\mathrm{OR} = 1$ である」という仮説のP値を条件付き確率を正確に計算することによって計算することによって行われる検定を **Fisher検定** (Fisherの正確検定, Fisher's exact test) と呼ぶ.

上で説明したように, Fisher検定には少なくともSterne版とClopper-Pearson版の2種類があるが(実はさらに別の変種も考えることができるが), どれを使っても, 第一種の過誤の確率を有意水準 α 以下に抑えることができるという利点を持っている.

しかし, その利点は欠点の裏返しでもあり, Fisher検定での第一種の過誤の確率を有意水準 α よりかなり小さくなってしまいがちである. 第一種の過誤の確率は有意水準 α に近い方がよい.

以下ではFisher検定における第一種の過誤の確率のグラフとPearsonのχ²検定およびWald検定での第一種の過誤の確率を重ねてプロットしてみよう. 良し悪しの基準は次の2つである:

- 第一種の過誤の確率は有意水準 α に近い方がよい. (グラフでは45度線に近い方がよい.)
- 第一種の過誤の確率は有意水準 α 以下であって欲しい. (グラフでは45度線の上にはみ出していない方がよい.)

In [62]:
```
plot_probabilities_of_type_I_error(10, 20, 0.2)
```

Out[62]:

In [63]:
```
plot_probabilities_of_type_I_error(40, 80, 0.2)
```

Out[63]:

In [64]:
```
plot_probabilities_of_type_I_error(160, 320, 0.2)
```

Out[64]:

In [65]:
```
plot_probabilities_of_type_I_error(640, 1280, 0.2)
```

Out[65]:

このように, Fisher検定はデータサイズを大きくしても, 第一種の過誤の確率が有意水準 α になかなか近付かない.

In [66]:
```
plot_probabilities_of_type_I_error(100, 200, 0.02)
```

Out[66]:

In [67]:
```
plot_probabilities_of_type_I_error(400, 800, 0.02)
```

Out[67]:

In [68]:
```
plot_probabilities_of_type_I_error(1600, 3200, 0.02)
```

Out[68]:

注意: 以上と同じことをR言語で行う場合には以下のことに注意する必要がある:

- R言語の `chisq.test` はデフォルトではYatesの連続性補正を行うので，`correct=F` として補正を無効にする必要がある.
- R言語の `fisher.test` が表示するP値はSterne版(もしくは "minlik" 版)である.
- Clopper-Pearson型のP値が必要ならば `exact2x2::exact2x2` を `tsmthod="central"` で使用すること.

6 おまけ: Bayes信用区間

6.1 オッズ比パラメータとリスク比パラメータのBayes信用区間

以下の手続きでオッズ比パラメータ OR やリスク比パラメータ RR の区間推定を行うことができる.

(1) パラメータ (p, q) に関する事前分布 $\mathrm{Beta}(1,1) \times \mathrm{Beta}(1,1)$ を用意する. (モデルのパラメータ p, q を確率変数とみなす.)

(2) データの数値 a, b, c, d から事後分布 $\mathrm{Beta}(a+1, b+1) \times \mathrm{Beta}(c+1, d+1)$ を作る. (事後分布はデータの数値を加味することによって得られたモデル内条件付き確率分布において確率変数 p, q が従う分布である.)

(3) 事後分布の従う乱数を $M = 10^6$ 個生成する. その結果を (p_i, q_i) $(i = 1, 2, \ldots, M)$ と表す.

(4) $i = 1, 2, \ldots, M$ についてその乱数のオッズ比 $\omega_i = (p_i/(1-p_i))/(q_i(1-q_i))$ とリスク比 $\rho_i = p_i/q_i$ を計算する.

(5) $\omega_1, \omega_2, \ldots, \omega_M$ の $\alpha/2$ 分位点 OR_L と $1-\alpha/2$ 分位点 OR_U を計算する:

$$\mathrm{OR}_L = \mathrm{quantile}([\omega_1, \omega_2, \ldots, \omega_M], \alpha/2), \quad \mathrm{OR}_U = \mathrm{quantile}([\omega_1, \omega_2, \ldots, \omega_M], 1-\alpha/2).$$

$\rho_1, \rho_2, \ldots, \rho_M$ の $\alpha/2$ 分位点 RR_L と $1-\alpha/2$ 分位点 RR_U を計算する:

$$\mathrm{RR}_L = \mathrm{quantile}([\rho_1, \rho_2, \ldots, \rho_M], \alpha/2), \quad \mathrm{RR}_U = \mathrm{quantile}([\rho_1, \rho_2, \ldots, \rho_M], 1-\alpha/2).$$

(6) 区間 $[\mathrm{OR}_L, \mathrm{OR}_U]$ をオッズ比パラメータ OR の信用度 $1-\alpha$ の **信用区間** (確信区間, credible interval)と呼び, 区間 $[\mathrm{RR}_L, \mathrm{RR}_U]$ をリスク比パラメータ RR の信用度 $1-\alpha$ の信用区間と呼ぶ:

$$\begin{aligned}\mathrm{credint}^{\mathrm{OR}}_{\mathrm{Bayes}}(a, b, c, d|\alpha) &= [\mathrm{OR}_L, \mathrm{OR}_U], \\ \mathrm{credint}^{\mathrm{RR}}_{\mathrm{Bayes}}(a, b, c, d|\alpha) &= [\mathrm{RR}_L, \mathrm{RR}_U].\end{aligned}$$

これらの区間は乱数の出目によって確率的に少し揺らぐが $M = 10^6$ が大きなおかげで微小にしか揺らがない.

注意: 以上はMonte Carlo法によるP値と信頼区間の計算の仕方の説明である. 数値積分で実装することも可能である. リスク比の場合の数値積分による実装については次のノートを参照せよ:

- [イベルメクチン論文の図の再現 (https://github.com/genkuroki/public/blob/main/0029/Supplementary%20Appendix%20Figure%20S6.ipynb)](https://github.com/genkuroki/public/blob/main/0029/Supplementary%20Appendix%20Figure%20S6.ipynb)

In [69]:
```
function credint_or_bayes(a, b, c, d; α=0.05, M=10^6)
    p = rand(Beta(a+1, b+2), M)
    q = rand(Beta(c+1, d+2), M)
    ω = @. (p*(1-q))/((1-p)*q)
    [quantile(ω, α/2), quantile(ω, 1-α)]
end

function credint_rr_bayes(a, b, c, d; α=0.05, M=10^6)
    p = rand(Beta(a+1, b+2), M)
    q = rand(Beta(c+1, d+2), M)
    ρ = @. p/q
    [quantile(ρ, α/2), quantile(ρ, 1-α)]
end
```

Out[69]:
```
credint_rr_bayes (generic function with 1 method)
```

6.2 Bayes信用区間の計算例

以下のデータについて今まで定義した信頼区間と信用区間を計算してみよう(信頼度は 95% とする).

	商品を購入した	購入しなかった	合計
デザインA	49	965	1014
デザインB	26	854	880
合計	75	1819	1894

In [70]:

```
a, b, c, d = 49, 965, 26, 854
@show a, b, c, d
println()
@show confint_or_wald(a, b, c, d)
@show confint_or_pearson_chisq(a, b, c, d)
@show confint_or_sterne(a, b, c, d)
@show confint_or_clopper_pearson(a, b, c, d)
@show credint_or_bayes(a, b, c, d)
println()
@show confint_rr_wald(a, b, c, d)
@show confint_rr_pearson_chisq(a, b, c, d)
@show credint_rr_bayes(a, b, c, d);
```

```
(a, b, c, d) = (49, 965, 26, 854)

confint_or_wald(a, b, c, d) = [1.0275181208761617, 2.7071797272783877]
confint_or_pearson_chisq(a, b, c, d) = [1.031778110800563, 2.6957054358850328]
confint_or_sterne(a, b, c, d) = [1.0202716568785046, 2.7656505208089848]
confint_or_clopper_pearson(a, b, c, d) = [1.0061031380773773, 2.8212565352058787]
credint_or_bayes(a, b, c, d) = [1.0305854335887983, 2.4860429564105297]

confint_rr_wald(a, b, c, d) = [1.025408984412048, 2.6087819449169145]
confint_rr_pearson_chisq(a, b, c, d) = [1.03047149323473, 2.599845901327348]
credint_rr_bayes(a, b, c, d) = [1.0289972191837617, 2.4014843593162727]
```

このようにどれも同じような区間になる.

6.3 Bayes信用区間に付随するP値函数

任意の区間推定から逆にP値函数を作れるのであった.

$\omega_1, \omega_2, \ldots, \omega_M$ の中の ω 以下の値の割合を $\mathrm{ecdf}([\omega_1, \omega_2, \ldots, \omega_M], \omega)$ と書く:

$$\mathrm{ecdf}([\omega_1, \omega_2, \ldots, \omega_M], \omega) = (\omega_i \leq \omega \text{ となる } i \text{ の個数})/M.$$

これを $\omega_1, \omega_2, \ldots, \omega_M$ の **経験累積分布函数** (empirical cumulative distribution function)と呼ぶ.

(7) 仮説「オッズ比パラメータは $\mathrm{OR} = \omega$ である」の **Bayes版P値** (Bayesian P-value)を

$$\mathrm{pvalue}_{\mathrm{Bayes}}(a, b, c, d | \mathrm{OR} = \omega) = \min \begin{pmatrix} 1 \\ 2\,\mathrm{ecdf}([\omega_1, \omega_2, \ldots, \omega_M], \omega) \\ 2(1 - \mathrm{ecdf}([\omega_1, \omega_2, \ldots, \omega_M], \omega) \end{pmatrix}$$

と定める. 同様に, 仮説「リスク比パラメータは $\mathrm{RR} = \rho$ である」のBayes版P値を

$$\mathrm{pvalue}_{\mathrm{Bayes}}(a, b, c, d | \mathrm{RR} = \rho) = \min \begin{pmatrix} 1 \\ 2\,\mathrm{ecdf}([\rho_1, \rho_2, \ldots, \rho_M], \rho) \\ 2(1 - \mathrm{ecdf}([\rho_1, \rho_2, \ldots, \rho_M], \rho) \end{pmatrix}$$

と定める. これらのP値達は乱数の出目によって確率的に少し揺らぐが $M = 10^6$ が大きなおかげで微小にしか揺らがない.

注意: これは乱数生成による Monte Carlo 法によるP値の計算法である. 数値積分でP値を計算するようにもできる. 以下のノートではリスク比のP値の場合に実際にそれを行っている:

- [イベルメクチン論文の図の再現 (https://github.com/genkuroki/public/blob/main/0029/Supplementary%20Appendix%20Figure%20S6.ipynb)](https://github.com/genkuroki/public/blob/main/0029/Supplementary%20Appendix%20Figure%20S6.ipynb)

In [71]:

```julia
"""Bayes版P値函数達を作る函数"""
function make_pvalue_or_rr_bayes(a, b, c, d; M=10^6, conjprior=(1, 1))
    α, β = conjprior
    p = rand(Beta(α+a, β+b), M)
    q = rand(Beta(α+c, β+d), M)
    ω = @. (p*(1-q))/((1-p)*q)
    ρ = @. p/q
    ecdf_OR = ecdf(ω)
    ecdf_RR = ecdf(ρ)
    pvalue_or_bayes(ω) = min(1, 2ecdf_OR(ω), 2(1-ecdf_OR(ω)))
    pvalue_rr_bayes(ρ) = min(1, 2ecdf_RR(ρ), 2(1-ecdf_RR(ρ)))
    pvalue_or_bayes, pvalue_rr_bayes
end

function plot_pvalue_functions(a, b, c, d;
        xlim_or=Tuple(confint_or_pearson_chisq(a, b, c, d; α=1e-3)),
        xlim_rr=Tuple(confint_rr_pearson_chisq(a, b, c, d; α=1e-3)),
        M=10^6, conjprior=(1, 1), kwargs...)
    pvalue_or_bayes, pvalue_rr_bayes = make_pvalue_or_rr_bayes(a, b, c, d; M, conjprior)

    P = plot()
    plot!(ω -> pvalue_or_bayes(ω), xlim_or...; label="Bayesian")
    plot!(ω -> pvalue_or_wald(a,b,c,d; ω), xlim_or...; label="Wald", ls=:dashdot)
    plot!(ω -> pvalue_or_pearson_chisq(a,b,c,d; ω), xlim_or...; label="Pearson χ²", ls=:dash)
    #plot!(ω -> pvalue_or_sterne(a,b,c,d; ω), xlim_or...; label="Fisher (Sterne)")
    #plot!(ω -> pvalue_or_clopper_pearson(a,b,c,d; ω), xlim_or...; label="Fisher (CP)")
    plot!(; xguide="OR = ω", yguide="P-value")
    plot!(; ytick=0:0.1:1)
    title!("a, b, c, d = $a, $b, $c, $d")

    Q = plot()
    plot!(ρ -> pvalue_rr_bayes(ρ), xlim_rr...; label="Bayesian")
    plot!(ρ -> pvalue_rr_wald(a,b,c,d; ρ), xlim_rr...; label="Wald", ls=:dashdot)
    plot!(ρ -> pvalue_rr_pearson_chisq(a,b,c,d; ρ), xlim_rr...; label="Pearson χ²", ls=:dash)
    plot!(; xguide="RR = ρ", yguide="P-value")
    plot!(; ytick=0:0.1:1)
    title!("a, b, c, d = $a, $b, $c, $d")

    plot(P, Q; size=(800, 250),
        leftmargin=4Plots.mm, bottommargin=4Plots.mm)
    plot!(; kwargs...)
end
```

Out[71]: plot_pvalue_functions (generic function with 1 method)

6.4 Bayes版P値函数の視覚化

In [72]:

```julia
a, b, c, d = 49, 965, 26, 854
plot_pvalue_functions(a, b, c, d)
```

Out[72]:

このように, Bayes版P値函数は通常のWald版およびPearsonのχ²検定版のP値函数によく一致している.

以下のように事前分布は $\mathrm{Beta}(0.5, 0.5)$ にした方がよく一致する.

In [73]:
```
a, b, c, d = 10, 3, 5, 9
plot_pvalue_functions(a, b, c, d; conjprior=(1, 1))
```

Out[73]:

In [74]:
```
a, b, c, d = 10, 3, 5, 9
plot_pvalue_functions(a, b, c, d; conjprior=(0.5, 0.5))
```

Out[74]:

6.5 Bayes版P値函数の視覚化の別の例

	症状悪化	悪化せず	合計
薬Iを処方	100	579	679
プラセボ	111	568	679
合計	211	1147	1358

この表は次の論文

- Gilmar Reis, M.D., Ph.D. et al., Effect of Early Treatment with Ivermectin among Patients with Covid-19, N Engl J Med 2022; 386:1721-1731. [link (https://www.nejm.org/doi/full/10.1056/NEJMoa2115869)]

の Table 2 から Intention-to-treat population (https://www.google.com/search?q=Intention-to-treat)の部分だけを抜き出したものである. 「症状悪化」の正確な意味などに関する詳しい説明は論文を参照せよ. (このデータは必修問題において既出である.)

この論文ではリスク比の区間推定方法としてちょうど上で説明したリスク比のBayes信用区間を採用している.

上の表のデータについてP値函数のグラフを描いてみよう.

In [75]:
```
a, b, c, d = 100, 579, 111, 568
plot_pvalue_functions(a, b, c, d)
```

Out[75]:

論文ではBayes統計の方法を使って区間推定しているが, P値函数達に関するこのグラフを見れば分かるように, この場合には, Bayes統計版ではないWald版やPearsonのχ²検定版の信頼区間を使っても同じ結果が得られる.

この一致は偶然ではない. 十分に a, b, c, d が大きいがゆえに正規分布近似が有効になっているので一致している.

巷に溢れる「ベイズ統計入門」では「ベイズ統計学は通常の統計学とは異なる主義・思想・哲学に基く本質的に異なる統計学である」のような感じの解説が目立つ. そのような解説に騙されないように注意することが必要である.

どんなに主義・思想・哲学が違っていても, 実践的には無意味な小さな違いしかない量(実践的には同じ量とみなされる)を計算しているのであれば, 主義・思想・哲学の違いは実践的には無視できるということを意味している.

7 関連講義動画

このノートの内容に関連したより先のより進んだ話題については

- 聴講コース 臨床研究者のための生物統計学 (https://ocw.kyoto-u.ac.jp/course/328/)

にある以下の3つの講義動画は視聴せよ:

- 佐藤俊哉, ランダム化ができないとき, 2018/10/25, 長さ 1:02:31) [link (https://youtu.be/tUkyjZXU1vc)]
- 佐藤俊哉, 交絡とその調整, 2018/12/20, 長さ 1:00:52) [link (https://youtu.be/ybdkQFEdCPM)]
- 佐藤俊哉, 回帰モデルと傾向スコア, 2019/02/21, 長さ 1:04:44) [link (https://youtu.be/cOHN444kBlo)]

このノートで解説した2×2の分割表の統計学の続きとして非常に楽しめるように思われる.

In [ ]:
```
```

検定と信頼区間: 平均の比較

- 黒木玄

- 2022-06-16〜2022-06-27

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 平均の差に関するP値と信頼区間
  - 1.1 平均の差に関するP値と信頼区間を使って行いたいこと
  - 1.2 平均の差の検定で使用されるP値の定義
- 2 データサイズを大きくしたときの信頼区間の収束の視覚化
  - 2.1 信頼区間のみを静的にプロット
  - 2.2 信頼区間とP値函数を同時に静的にプロット
  - 2.3 動画
- 3 Welchの t 検定のP値と信頼区間の計算例
  - 3.1 必修問題: Welchの t 検定のP値と信頼区間の計算
    - 3.1.1 WolframAlphaによるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例
    - 3.1.2 Julia言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例
    - 3.1.3 Julia言語による必修問題のデータの視覚化
    - 3.1.4 Julia言語による必修問題のP値函数の視覚化
    - 3.1.5 R言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例
    - 3.1.6 Welchの t 検定のP値と信頼区間の計算の必修問題の解答例
  - 3.2 必修問題: 12歳の男子と女子の平均身長の差
    - 3.2.1 WolframAlphaによる12歳の男子と女子の平均身長の差に関する必修問題の解答例
    - 3.2.2 Julia言語による12歳の男子と女子の平均身長の差に関する必修問題の解答例
    - 3.2.3 Julia言語による12歳の男子と女子の平均身長の差のP値函数の視覚化
    - 3.2.4 R言語による12歳の男子と女子の平均身長の差に関する必修問題の解答例
    - 3.2.5 12歳の男子と女子の平均身長の差に関する必修問題の解答例
    - 3.2.6 仮に2012年のデータのサンプルサイズだけが実際のサイズの2倍だったとしたら
- 4 Welchの t 検定で使う自由度の式の導出
  - 4.1 2つの正規分布の標本分布の設定
  - 4.2 χ^2分布で近似することによる自由度の式の導出
  - 4.3 t 分布による近似
- 5 Welchの t 検定での第一種の過誤の確率の視覚化
  - 5.1 異なる分散を持つ正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.2 異なる分散を持つ一様分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.3 正規分布とガンマ分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.4 一様分布とガンマ分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.5 同一のガンマ分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.6 異なるガンマ分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.7 正規分布と対数正規分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.8 同一の対数正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
  - 5.9 異なる対数正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化
- 6 補足: Studentの t 検定について
  - 6.1 Studentの t 検定のP値と信頼区間の定義
  - 6.2 Studentの t 検定とWelchの t 検定に付随する信頼区間の比較
  - 6.3 Studentの t 検定とWelchの t 検定で使用する t 分布の自由度の比較
  - 6.4 Studentの t 検定とWelchの t 検定が異なる結果を与える例
  - 6.5 Studentの t 検定での第一種の過誤の確率の視覚化

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Printf
using QuadGK
using RCall
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]: mykurtosis (generic function with 2 methods)

In [4]:

```julia
function tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    (x̄ - ȳ - Δμ) / √(sx²/m + sy²/n)
end

function tvalue_welch(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
end

function degree_of_freedom_welch(m, sx², n, sy²)
    (sx²/m + sy²/n)^2 / ((sx²/m)^2/(m-1) + (sy²/n)^2/(n-1))
end

function degree_of_freedom_welch(x, y)
    m, sx² = length(x), var(x)
    n, sy² = length(y), var(y)
    degree_of_freedom_welch(m, sx², n, sy²)
end

function pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    t = tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
    ν = degree_of_freedom_welch(m, sx², n, sy²)
    2ccdf(TDist(ν), abs(t))
end

function pvalue_welch(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
end

function confint_welch(m, x̄, sx², n, ȳ, sy²; α=0.05)
    ν = degree_of_freedom_welch(m, sx², n, sy²)
    c = quantile(TDist(ν), 1-α/2)
    SEhat = √(sx²/m + sy²/n)
    [x̄-ȳ-c*SEhat, x̄-ȳ+c*SEhat]
end

function confint_welch(x, y; α=0.05)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    confint_welch(m, x̄, sx², n, ȳ, sy²; α)
end
```

Out[4]:

```
confint_welch (generic function with 2 methods)
```

In [5]:

```julia
s²_student(m, sx², n, sy²) = ((m-1)*sx² + (n-1)*sy²)/(m+n-2)

function tvalue_student(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    s² = s²_student(m, sx², n, sy²)
    (x̄ - ȳ - Δμ) / √(s²*(1/m + 1/n))
end

function tvalue_student(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    tvalue_student(m, x̄, sx², n, ȳ, sy²; Δμ)
end

function pvalue_student(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    t = tvalue_student(m, x̄, sx², n, ȳ, sy²; Δμ)
    2ccdf(TDist(m+n-2), abs(t))
end

function pvalue_student(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    pvalue_student(m, x̄, sx², n, ȳ, sy²; Δμ)
end

function confint_student(m, x̄, sx², n, ȳ, sy²; α=0.05)
    c = quantile(TDist(m+n-2), 1-α/2)
    s² = s²_student(m, sx², n, sy²)
    SEhat = √(s²*(1/m + 1/n))
    [x̄-ȳ-c*SEhat, x̄-ȳ+c*SEhat]
end

function confint_student(x, y; α=0.05)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    confint_student(m, x̄, sx², n, ȳ, sy²; α)
end
```

Out[5]:

```
confint_student (generic function with 2 methods)
```

In [6]:

```julia
"""引数 x, y として平均0を持つ分布のサンプルを与えること"""
function plot_confint_of_diffmeans(x, y; α = 0.05,
        xlim = nothing,
        plot_pvaluefunc = false,
        plot_student = false,
        xtick=-100:100, kwargs...)
    m, n = length(x), length(y)
    if isnothing(xlim)
        a, b = extrema([x; y])
        a, b = a - 0.05(b-a), b + 0.05(b-a)
    else
        a, b = xlim
    end
    if plot_pvaluefunc
        scatter(x, fill(-0.05, m); label="", msc=:auto, alpha=0.7, c=1)
        scatter!(y, fill(-0.10, n); label="", msc=:auto, alpha=0.7, c=2)
        if plot_student
            plot!(Δμ -> pvalue_welch(x, y; Δμ), a, b;
                label="Welch P-value", c=:blue)
            plot!(confint_welch(x, y; α), fill(α, 2);
                label="Welch conf. int.", lw=3, c=:red)
            plot!(Δμ -> pvalue_student(x, y; Δμ);
                label="Student P-value", ls=:dash, c=:magenta)
            plot!(confint_student(x, y; α), fill(0.03, 2);
                label="Student conf. int.", lw=3, c=6)
        else
            plot!(Δμ -> pvalue_welch(x, y; Δμ), a, b; label="", c=:blue)
            plot!(confint_welch(x, y; α), fill(α, 2); label="", lw=3, c=:red)
        end
        plot!(; ylim=(-0.15, 1.03), ytick=0:0.1:1, yguide="P-value")
        title!("P-value func. and $(100(1-α))% conf. int." *
            " of Δμ for data of size m=$m, n=$n")
        plot!(; size=(600, 200), leftmargin=4Plots.mm)
    else
        scatter(x, fill(-0.05, m); label="", msc=:auto, alpha=0.7, c=1)
        scatter!(y, fill(-0.13, n); label="", msc=:auto, alpha=0.7, c=2)
        if plot_student
            plot!(confint_welch(x, y; α), fill(0.06, 2);
                label="Welch conf. int.", lw=6, c=:red)
            plot!(confint_student(x, y; α), fill(0.06, 2);
                label="Student conf. int.", lw=6, c=:red)
        else
            plot!(confint_welch(x, y; α), fill(0.06, 2);
                label="", lw=6, c=:red)
        end
        plot!(; ylim=(-0.2, 0.2), yaxis=false, ytick=false)
        title!("$(100(1-α))% confidence interval"
            * " of Δμ for data of size m=$m, n=$n")
        plot!(; size=(600, 80))
    end
    plot!(; xlim=(a, b), xtick)
    plot!(; kwargs...)
end

x, y = rand(Gamma(3,2)-6, 20), rand(Gamma(2,3)-6, 30)
plot_confint_of_diffmeans(x, y) ▷ display
plot_confint_of_diffmeans(x, y; plot_pvaluefunc=true) ▷ display
plot_confint_of_diffmeans(x, y;
    plot_pvaluefunc=true, plot_student=true,
    size=(600, 300))
```

P-value func. and 95.0% conf. int. of Δμ for data of size m=20, n=30
P-value
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−6 −5 −4 −3 −2 −1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Out[6]:

P-value func. and 95.0% conf. int. of Δμ for data of size m=20, n=30
P-value
Welch P-value
Welch conf. int.
Student P-value
Student conf. int.
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−6 −5 −4 −3 −2 −1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

In [7]:

```julia
rd(x) = @sprintf "%.3f" x

function plot_welch(;
        distx=Normal(0,1), m=10, distxlim=nothing,
        disty=Normal(0,2), n=20, distylim=nothing,
        L=10^6, M=10^4, kwargs...)
    μ_x, σ_x = mean(distx), std(distx)
    μ_y, σ_y = mean(disty), std(disty)
    Δμ = μ_x - μ_y

    println("m = ", m,
        ",  μ_x = ", rd(μ_x),
        ",  σ_x = ", rd(σ_x),
        ",  skewness_x = ", rd(myskewness(distx)),
        ",  kurtosis_x = ", rd(mykurtosis(distx)))
    println("n = ", n,
        ",  μ_y = ", rd(μ_y),
        ",  σ_y = ", rd(σ_y),
        ",  skewness_y = ", rd(myskewness(disty)),
        ",  kurtosis_y = ", rd(mykurtosis(disty)))
    @show Δμ

    pval = similar(zeros(), L)
    T = similar(zeros(), L)
    df = similar(zeros(), L)
    tmpx = [similar(zeros(), m) for _ in 1:nthreads()]
    tmpy = [similar(zeros(), n) for _ in 1:nthreads()]
    @threads for i in 1:L
        x = rand!(distx, tmpx[threadid()])
        y = rand!(disty, tmpy[threadid()])
        pval[i] = pvalue_welch(x, y; Δμ)
        T[i] = tvalue_welch(x, y; Δμ)
        df[i] = degree_of_freedom_welch(x, y)
    end
    ecdf_pval = ecdf(pval)
    f(x) = ecdf_pval(x)
    @show ν̄ = mean(df)

    P1 = if isnothing(distxlim)
        plot(distx; label="", title="$(distname(distx)), m=$m")
    else
        plot(distx, distxlim...; label="", title="$(distname(distx)), m=$m")
    end
    plot!(; xguide="x", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P2 = if isnothing(distylim)
        plot(disty; label="", title="$(distname(disty)), n=$n")
    else
        plot(disty, distylim...; label="", title="$(distname(disty)), n=$n")
    end
    plot!(; xguide="y", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P3 = scatter(T[1:M], df[1:M]; label="", ms=1.5, msw=0, alpha=0.3)
    plot!(; xlim=(-5, 5), xtick=-5:5)
    plot!(; xguide="T = T-value", yguide="ν = df", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P4 = stephist(T; norm=true, label="T")
    plot!(TDist(ν̄); label="TDist(ν̄)", ls=:dash)
    plot!(; xlim=(-5, 5), xtick=-5:5)
    plot!(; xguide="t", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P5 = plot(f, 0, 1; label="")
    plot!(identity, 0, 1; label="", ls=:dot)
    plot!(; xtick=0:0.1:1, ytick=0:0.1:1)
    plot!(; xguide="significance level α", yguide="probability of type I error",
        guidefontsize=9)
    plot!(; tickfontsize=6)

    P6 = plot(f, 0, 0.1; label="")
```

```
    plot!(identity, 0, 0.1; label="", ls=:dot)
    plot!(; xtick=0:0.01:1, ytick=0:0.01:1)
    plot!(; xguide="significance level α", yguide="probability of type I error",
        guidefontsize=9)
    plot!(; tickfontsize=6)

    layout = @layout [a{0.25h} b{0.25h}; c{0.25*h} d{0.25*h}; e f]
    plot(P1, P2, P3, P4, P5, P6; size=(700, 700), layout)
end

plot_welch(distx=Gamma(2,3), m=40, disty=Gamma(3,2), n=80)
```

```
m = 40,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 80,  μ_y = 6.000,  σ_y = 3.464,  skewness_y = 1.155,  kurtosis_y = 2.000
Δμ = 0.0
ν̄ = mean(df) = 68.32711835884052
```

Out[7]:

In [8]:

```julia
rd(x) = @sprintf "%.3f" x

function plot_student(;
        distx=Normal(0,1), m=10, distxlim=nothing,
        disty=Normal(0,2), n=20, distylim=nothing,
        L=10^6, M=10^4, kwargs...)
    μ_x, σ_x = mean(distx), std(distx)
    μ_y, σ_y = mean(disty), std(disty)
    Δμ = μ_x - μ_y

    println("m = ", m,
        ",  μ_x = ", rd(μ_x),
        ",  σ_x = ", rd(σ_x),
        ",  skewness_x = ", rd(myskewness(distx)),
        ",  kurtosis_x = ", rd(mykurtosis(distx)))
    println("n = ", n,
        ",  μ_y = ", rd(μ_y),
        ",  σ_y = ", rd(σ_y),
        ",  skewness_y = ", rd(myskewness(disty)),
        ",  kurtosis_y = ", rd(mykurtosis(disty)))
    @show Δμ

    pval_student = similar(zeros(), L)
    pval_welch = similar(zeros(), L)
    T_student = similar(zeros(), L)
    T_welch = similar(zeros(), L)
    tmpx = [similar(zeros(), m) for _ in 1:nthreads()]
    tmpy = [similar(zeros(), n) for _ in 1:nthreads()]
    @threads for i in 1:L
        x = rand!(distx, tmpx[threadid()])
        y = rand!(disty, tmpy[threadid()])
        pval_student[i] = pvalue_student(x, y; Δμ)
        pval_welch[i] = pvalue_welch(x, y; Δμ)
        T_student[i] = tvalue_student(x, y; Δμ)
        T_welch[i] = tvalue_welch(x, y; Δμ)
    end
    ecdf_pval_student = ecdf(pval_student)
    ecdf_pval_welch = ecdf(pval_welch)
    f_student(x) = ecdf_pval_student(x)
    f_welch(x) = ecdf_pval_welch(x)

    P1 = if isnothing(distxlim)
        plot(distx; label="", title="$(distname(distx)), m=$m")
    else
        plot(distx, distxlim...; label="", title="$(distname(distx)), m=$m")
    end
    plot!(; xguide="x", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P2 = if isnothing(distylim)
        plot(disty; label="", title="$(distname(disty)), n=$n")
    else
        plot(disty, distylim...; label="", title="$(distname(disty)), n=$n")
    end
    plot!(; xguide="y", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P4 = stephist(T_student; norm=true, label="Student's T")
    stephist!(T_welch; norm=true, label="Welch's T", ls=:dashdot)
    plot!(TDist(m+n-2); label="TDist(m+n-2)", ls=:dash)
    plot!(; xlim=(-5, 5), xtick=-5:5)
    plot!(; xguide="t", yguide="density", guidefontsize=9)
    plot!(; leftmargin=4Plots.mm)#, bottommargin=4Plots.mm)
    plot!(; tickfontsize=6)

    P6 = plot(; legend=:bottomright)
    plot!(f_student, 0, 0.1; label="Student")
    plot!(f_welch, 0, 0.1; label="Welch", ls=:dashdot)
    plot!(identity, 0, 0.1; label="", ls=:dot)
    plot!(; xtick=0:0.01:1, ytick=0:0.01:1)
    plot!(; xguide="significance level α", yguide="probability of type I error",
        guidefontsize=9)
    plot!(; tickfontsize=6, xrotation=30)

    P12 = plot(P1, P2; layout=@layout [a b])
```

```
    P46 = plot(P4, P6; layout=@layout [a{0.6w} b])
    plot(P12, P46; size=(700, 480), layout=@layout [a{0.4h}; b])
end

plot_student(distx=Gamma(2,3), m=40, disty=Gamma(3,2), n=80)
```

```
m = 40,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 80,  μ_y = 6.000,  σ_y = 3.464,  skewness_y = 1.155,  kurtosis_y = 2.000
Δμ = 0.0
```

Out[8]:

1 平均の差に関するP値と信頼区間

このノートでは2群の平均の比較に関する **Welchの t 検定** のP値とそれに付随する信頼区間について解説する.

多くの入門的教科書では, 2群の平均の比較について, Welchの t 検定ではなく, Studentの t 検定について解説している場合が多い.

しかし, 2群の平均の比較に関するStudentの t 検定は現実の母集団について仮定するには非現実的な「等分散の仮定」を使うので頑健ではない.

昔の教科書には正規性検定と等分散検定を通過した場合にStudentの t 検定を使うように指示しているものがあるが, その方法には様々な害がある. どのような害があるかについては

- https://okumuralab.org/~okumura/stat/ttest.html (https://okumuralab.org/~okumura/stat/ttest.html)

の「分散が等しいかどうか不明の場合」の節を参照せよ.

そこで, このノートでは思い切ってStudentの t 検定の解説を止めて, Welchの t 検定のみを解説することにした.

Welchの t 検定は一定の頑健さを持っており, 母集団分布の正規性を仮定しなくても使用可能である. (ただし, 誤差が大きくなる場合に関する注意は必要になる.)

1.1 平均の差に関するP値と信頼区間を使って行いたいこと

以下のようなことを行いたい.

(1) 日本にいる12歳の男子と女子をそれぞれ m 人と n 人無作為抽出して, 身長を測って得た数値のデータをそれぞれ $x_1, \dots, x_m$, $y_1, \dots, y_n$ と書くことにする. そのようなデータを用いて日本にいる12歳の男子と女子の平均身長の差がどれだけあるか(もしくはないか)を推定したい.

(2) 薬Xの効き目を調べるために, 同一の母集団から $m+n$ 人を無作為に選んで, ランダムに選んだそのうち m 人は薬Xを与え, 残りの n 人にはプラセボ(偽薬)を与えた. そして, $m+n$ 人について治療効果を表す指標の数値を測定し, 薬Xを与えた m 人分の数値は $x_1, \dots, x_m$ でプラセボを与えた n 人分の数値は $y_1, \dots, y_n$ であったとする. そのデータから, 薬Xを与えた場合の治療効果を表す指標の平均値と与えなかった場合の治療効果を表す指標の平均値の差がどうなっているかについて推定したい.

目標は2つの群の平均の差の信頼区間の構成である.

そのためには, 検定と信頼区間の表裏一体性より, P値を適切に定義すればよい.

1.2 平均の差の検定で使用されるP値の定義

データ: m 個の実数値 $x_1, \dots, x_m$ と n 個の実数値 $y_1, \dots, y_n$.

x_i 達と y_i 達の標本平均と不偏分散を以下のように書くことにする:

$$\begin{aligned} \bar{x} &= \frac{1}{m}\sum_{i=1}^{m} x_i, \quad s_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2, \\ \bar{y} &= \frac{1}{n}\sum_{i=1}^{n} y_i, \quad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{m}(y_i - \bar{y})^2. \end{aligned}$$

統計モデル: 平均 μ_x と分散 σ_x^2 を持つ確率分布 D_x のサイズ m の標本分布 D_x^m と平均 μ_y と分散 σ_y^2 を持つ確率分布 D_y のサイズ n の標本分布 D_y^n の直積分布 $D_x^m \times D_y^n$ を統計モデルとして採用する.

以下では簡単のため D_x も D_y も連続分布であると仮定し, それぞれの確率密度函数を $p_x(x), p_y(y)$ と書くことにする. このとき, 統計モデル $D_x^m \times D_y^n$ の確率密度函数は

$$p(x_1, \dots, x_m, y_1, \dots, y_n) = p_x(x_1) \cdots p_x(x_m) \cdot p_y(y_1) \cdots p_y(y_n)$$

になる. この確率分布に従う確率変数達(独立になる)を $X_1, \dots, X_m, Y_1, \dots, Y_n$ と書き, X_i 達と Y_i 達の標本平均と不偏分散を以下のように書くことにする:

$$\begin{aligned} \bar{X} &= \frac{1}{m}\sum_{i=1}^{m} X_i, \quad S_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(X_i - \bar{X})^2, \\ \bar{Y} &= \frac{1}{n}\sum_{i=1}^{n} Y_i, \quad S_y^2 = \frac{1}{n-1}\sum_{i=1}^{m}(Y_i - \bar{Y})^2. \end{aligned}$$

この統計モデルが適用可能な場合: 上の統計モデルでは, データの数値 x_i, y_i の生成のされ方のモデル化である X_i, Y_i 達が独立な確率変数になっており, X_i 達は同じ分布に従い, Y_i 達も同じ分布に従うが, X_i 達と Y_i 達が従う分布は違っていてよいという設定になっている. データの生成のされ方がこの条件を満たしていない場合には, 上の統計モデルの適用は不適切になる.

例えば, $m = n$ でデータ中の x_i の値と y_i の値が関係している場合には上のモデルの使用は不適切になる. 例えば, i 番目の人の薬を与える前の数値が x_i で薬を与えた後の数値が y_i の場合には y_i の値は x_i の値に依存して決まるので, このような場合に上のモデルの適用は不適切になる. そのような場合には, 例えばそれらの差に平均の値に関する検定や信頼区間を使用すればよい.

検定したい仮説: $\mu_x - \mu_y = \Delta\mu$ ($\Delta\mu$ は具体的な数値).

中心極限定理: モデル内確率変数としての2つの標本平均達の分布について, 中心極限定理による正規分布近似が使えると仮定する.

$\bar{X}, \bar{Y}$ の平均(期待値)と分散は以下のようになる:

$$E[\bar{X}] = \mu_x, \quad E[\bar{Y}] = \mu_y, \quad \mathrm{var}(\bar{X}) = \frac{\sigma_x^2}{m}, \quad \mathrm{var}(\bar{Y}) = \frac{\sigma_y^2}{n}.$$

さらに, $\bar{X}$ と $\bar{Y}$ が確率変数として独立であることより,

$$E[\bar{X} - \bar{Y}] = \mu_x - \mu_y, \quad \mathrm{var}(\bar{X} - \bar{Y}) = \frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}.$$

さらに, 中心極限定理より, 次の近似が使える:

$$\bar{X} - \bar{Y} \sim \mathrm{Normal}\left(\mu_x - \mu_y,\ \sqrt{\frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}}\right), \quad \text{approximately.}$$

すなわち,

$$\frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{\sqrt{\frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}}} \sim \mathrm{Normal}(0, 1), \quad \text{approximately.}$$

大数の法則: モデル内確率変数としての不偏分散 S_x^2, S_y^2 でモデルの分散 σ_x^2, σ_y^2 がよく近似されていると仮定する.

このとき, 次の近似が使える.

$$T := \frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{\sqrt{\frac{S_x^2}{m} + \frac{S_y^2}{n}}} \sim \mathrm{Normal}(0, 1), \quad \text{approximately.}$$

P値の定義: これを用いて, 具体的に与えられた数値 $\Delta\mu$ に関する仮説「$\mu_x - \mu_y = \Delta\mu$」のP値を以下のように定義する. まず, データの数値の t 値 $t = t(\Delta\mu)$ を次のように定義する:

$$t = t(\Delta\mu) = \frac{(\bar{x} - \bar{y}) - \Delta\mu}{\sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}}}.$$

仮説「$\mu_x - \mu_y = \Delta\mu$」のP値を, その仮説下のモデル内の確率変数としての t 値 T の値の絶対値がデータから計算した t 値 $t = t(\Delta\mu)$ の絶対値以上になる確率の近似値として定義する:

$$\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \mu_x - \mu_y = \Delta\mu) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0, 1), |t(\Delta\mu)|)).$$

しかし, 実際に使用されるのは次に定義する t 分布を使って定義されたP値の方である.

t分布を使って補正されたP値の定義: 上のP値の t 分布版を定義しよう.

天下り的になってしまうが, 自由度 ν を次のように定義する(これの導出は別の節で行う):

$$\nu = \frac{\left(s_x^2/m + s_y^2/n\right)^2}{\frac{(s_x^2/m)^2}{m-1} + \frac{(s_y^2/n)^2}{n-1}}.$$

仮に $s_x^2 = s_y^2, m = n$ だとすると, $\nu = 2n - 2$ となる.

一般にこの ν は整数にならないがそのまま用いる.

この ν を用いて t 分布を使って計算されるP値を次のように定める:

$$\mathrm{pvalue}_{\mathrm{Welch}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \mu_x - \mu_y = \Delta\mu) = 2(1 - \mathrm{cdf}(\mathrm{TDist}(\nu), |t(\Delta\mu)|)).$$

以下ではこれを使うことにする. このP値は **Welchの t 検定** のP値である.

ν の定義を覚える必要はない. m, n が大きいならば, ν も大きくなり, 自由度 ν の t 分布 $\mathrm{TDist}(\nu)$ は標準正規分布 $\mathrm{Normal}(0, 1)$ でよく近似されるようになるので, ν の値がどうであるかを実質的に気にする必要がなくなることにも注意せよ. この t 分布による補正が有効なのは特別な場合に限るが, 有効でない場合にも害はないのでこちらの方を使うことにする.

信頼区間: Welchの t 検定のP値から定まる信頼区間は以下のように書ける.

まず, 自由度 ν の t 分布において $t_{\nu,\alpha/2}$ 以上になる確率は $\alpha/2$ になると仮定する:

$$t_{\nu,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(\nu), 1 - \alpha/2).$$

このとき, 平均の差 $\mu_x - \mu_y$ の信頼度 $1 - \alpha$ の信頼区間が次のように定義される:

$$\begin{aligned}&\mathrm{confint}_{\mathrm{Welch}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \alpha) \\ &= \left[\bar{x} - \bar{y} - t_{\nu,\alpha/2}\sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}},\ \bar{x} - \bar{y} + t_{\nu,\alpha/2}\sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}}\right].\end{aligned}$$

2 データサイズを大きくしたときの信頼区間の収束の視覚化

In [9]:
```
# テストデータの生成
# 後で使う函数との相性で平均が0の分布で標本を生成する必要がある.

Random.seed!(845145105963) # はよこい しごと ごくろうさん
distx, disty = Gamma(2,1), Gamma(5,1)
distx, disty = distx-mean(distx), disty-mean(disty)
x = rand(distx, 320)
y = rand(disty, 640)
a, b = extrema([x; y])
xlim = (a - 0.05(b - a), b + 0.05(b-a))
```

Out[9]: (-4.849858804201334, 10.08008066780289)

2.1 信頼区間のみを静的にプロット

In [10]:
```
plot_confint_of_diffmeans(x[1:5], y[1:10]; xlim)
```

Out[10]:

In [11]:
```
plot_confint_of_diffmeans(x[1:20], y[1:40]; xlim)
```

Out[11]:

In [12]:
```
plot_confint_of_diffmeans(x[1:80], y[1:160]; xlim)
```

Out[12]:

In [13]:
```
plot_confint_of_diffmeans(x[1:320], y[1:640]; xlim)
```

Out[13]:

2.2 信頼区間とP値函数を同時に静的にプロット

In [14]:
```
plot_confint_of_diffmeans(x[1:5], y[1:10]; xlim, plot_pvaluefunc=true)
```

Out[14]:

In [15]:
```
plot_confint_of_diffmeans(x[1:20], y[1:40]; xlim, plot_pvaluefunc=true)
```

Out[15]:

In [16]:
```
plot_confint_of_diffmeans(x[1:80], y[1:160]; xlim, plot_pvaluefunc=true)
```

Out[16]:

In [17]:
```
plot_confint_of_diffmeans(x[1:320], y[1:640]; xlim, plot_pvaluefunc=true)
```

Out[17]:

2.3 動画

In [18]:
```
L = length(x)
anim = @animate for m in [fill(5, 20); 10:L; fill(L, 20)]
    plot_confint_of_diffmeans(x[1:m], y[1:2m]; xlim, plot_pvaluefunc=true)
end
gif(anim, "images/confint_of_diffmeans.gif")
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\confint_of_diffmeans.gif
@ Plots D:\.julia\packages\Plots\MzlNY\src\animation.jl:130
```

Out[18]:

このノートのPDF版で上の動画をみることはできない. しかし, 以下の場所でみることができる:

- https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_diffmeans.gif (https://github.com/genkuroki/Statistics/blob/master/2022/images/confint_of_diffmeans.gif)

3 Welchの t 検定のP値と信頼区間の計算例

3.1 必修問題: Welchの t 検定のP値と信頼区間の計算

サイズが $m = 20, n = 30$ のデータ

```
x = [19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9,
     11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6]

y = [44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7,
     74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7,
     124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4]
```

について以下を求めよ.

(1) 仮説 $\mu_x - \mu_y = 0$ のP値.

(2) 仮説 $\mu_x - \mu_y = -30$ のP値.

(3) Welchの t 検定に付随する 95% 信頼区間.

3.1.1 WolframAlphaによるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例

`mean {19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9, 11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6}` → 実行 (https://www.wolframalpha.com/input?i=mean+%7B19.2%2C+22.7%2C+7.8%2C+138.5%2C+70.5%2C+44.3%2C+84.0%2C+35.6%2C+72.4%2C+23.9%2C+11.7%2C+26
→ 59.445 $= \bar{x}$

`var {19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9, 11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6}` → 実行 (https://www.wolframalpha.com/input?i=var+%7B19.2%2C+22.7%2C+7.8%2C+138.5%2C+70.5%2C+44.3%2C+84.0%2C+35.6%2C+72.4%2C+23.9%2C+11.7%2C+26.6
→ 1448.48 $= s_x^2$

`mean {44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7, 74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7, 124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4}` → 実行 (https://www.wolframalpha.com/input?i=mean+%7B44.3%2C+66.9%2C+62.9%2C+78.4%2C+71.2%2C+32.5%2C+111.4%2C+38.2%2C+68.2%2C+50.7%2C+74.5%2C+4
→ 67.5433 $= \bar{y}$

`var {44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7, 74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7, 124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4}` → 実行 (https://www.wolframalpha.com/input?i=var+%7B44.3%2C+66.9%2C+62.9%2C+78.4%2C+71.2%2C+32.5%2C+111.4%2C+38.2%2C+68.2%2C+50.7%2C+74.5%2C+46.2
→ 479.481 $= s_y^2$

`(v/m+w/n)^2/((v/m)^2/(m-1)+(w/n)^2/(n-1)) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28v%2Fm%2Bw%2Fn%29%5E2%2F%28%28v%2Fm%29%5E2%2F%28m-1%29%2B%28w%2Fn%29%5E2%2F%28n-1%29%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C+y%3D67.5433%2C+w%3D479.481%7
→ 27.4358 $= \nu$ (自由度)

(1)

`(x-y-0)/sqrt(v/m+w/n) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y-0%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C+y%3D
→ -0.861294 $= t$ (t値)

`2(1 - cdf(TDistribution(27.4358), 0.861294))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2827.4358%29%2C+0.861294%29%29) → 0.396541 ((1)のP値)

(2)

`(x-y+30)/sqrt(v/m+w/n) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y%2B30%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C
→ 2.32935 $= t$ (t値)

`2(1 - cdf(TDistribution(27.4358), 2.32935))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2827.4358%29%2C+2.32935%29%29) → 0.0274389 ((2)のP値)

(3)

`quantile(TDistribution(27.4358), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%2827.4358%29%2C+0.975%29) → 2.05031 $= t_{\nu,\alpha/2}$

`{x-y-2.05031*sqrt(v/m+w/n), x-y+2.05031*sqrt(v/m+w/n)} where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%7Bx-y-2.05031*sqrt%28v%2Fm%2Bw%2Fn%29%2C+x-y%2B2.05031*sqrt%28v%2Fm%2Bw%2Fn%29%7D+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2
→ {-27.3763, 11.1797} ((3)の信頼区間)

3.1.2 Julia言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例

函数の定義についてはこのノートの最初の方を見よ.

In [19]:
```
x = [19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9,
    11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6]
y = [44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7,
    74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7,
    124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4]
@show length(x)
@show mean(x)
@show std(x)
@show length(y)
@show mean(y)
@show std(y)
@show degree_of_freedom_welch(x, y);
```

```
length(x) = 20
mean(x) = 59.44499999999999
std(x) = 38.05893594496774
length(y) = 30
mean(y) = 67.54333333333334
std(y) = 21.897058270907998
degree_of_freedom_welch(x, y) = 27.43582631782945
```

In [20]:
```
# (1)
@show tvalue_welch(x, y; Δμ = 0)
@show pvalue_welch(x, y; Δμ = 0);
```

```
tvalue_welch(x, y; Δμ = 0) = -0.8612965858138025
pvalue_welch(x, y; Δμ = 0) = 0.39653998689489345
```

In [21]:
```
# (2)
@show tvalue_welch(x, y; Δμ = -30)
@show pvalue_welch(x, y; Δμ = -30);
```

```
tvalue_welch(x, y; Δμ = -30) = 2.3293472801356576
pvalue_welch(x, y; Δμ = -30) = 0.027439073239531347
```

In [22]:
```
# (3)
@show confint_welch(x, y; α=0.05);
```

```
confint_welch(x, y; α = 0.05) = [-27.376321534615272, 11.179654867948585]
```

3.1.3 Julia言語による必修問題のデータの視覚化

In [23]:
```
ecdf_x, ecdf_y = ecdf(x), ecdf(y)
plot(; legend=:bottomright, ytick=0:0.1:1)
plot!(x -> ecdf_x(x), -10, 170; label="ecdf of x")
plot!(x -> ecdf_y(x), -10, 170; label="ecdf of y")
```

Out[23]:

x に対してデータ中の x 以下の要素の個数の割合を対応させる函数をデータの経験累積分布函数(empirical cumulative distribution function, ecdf)と呼ぶ. 上のグラフはデータ x, y の経験累積分布函数のプロットである.

経験分布函数のプロットには, ヒストグラムと違って, ビン(横軸の分割)の仕方に神経を使わずにプロットしても問題がないという利点がある. 欠点は直観的に意味を把握できず, 少し考えなければグラフを適切に解釈できないことである.

上のecdf達のグラフからデータxはデータyよりも広く分布していることがわかる.

3.1.4 Julia言語による必修問題のP値函数の視覚化

In [24]:
```
plot(Δμ -> pvalue_welch(x, y; Δμ), -40, 30; label="P-value function")
vline!([mean(x)-mean(y)]; label="x̄ - ȳ", ls=:dash)
plot!(; xtick=-100:10:100, ytick=0:0.1:1)
```

Out[24]:

3.1.5 R言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例

In [25]:
```
# (1), (3)
@rput x y
R"""t.test(x, y)"""
```

Out[25]:
```
RObject{VecSxp}

        Welch Two Sample t-test

data:  x and y
t = -0.8613, df = 27.436, p-value = 0.3965
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -27.37632  11.17965
sample estimates:
mean of x mean of y
 59.44500  67.54333
```

In [26]:
```
# (2), (3)
R"""t.test(x, y, mu=-30)"""
```

Out[26]:
```
RObject{VecSxp}

        Welch Two Sample t-test

data:  x and y
t = 2.3293, df = 27.436, p-value = 0.02744
alternative hypothesis: true difference in means is not equal to -30
95 percent confidence interval:
 -27.37632  11.17965
sample estimates:
mean of x mean of y
 59.44500  67.54333
```

3.1.6 Welchの t 検定のP値と信頼区間の計算の必修問題の解答例

(1) 0.3965

(2) 0.02744

(3) [-27.38, 11.18]

3.2 必修問題: 12歳の男子と女子の平均身長の差

国民健康・栄養調査14 身長・体重の平均値及び標準偏差 (https://www.e-stat.go.jp/dbview?sid=0003224177)における2012年と2018年の12歳の男女のデータによれば, 男子 m 人の身長のデータの標本平均 $\bar{x}$ と標本分散の平方根 s_x と女子 n 人の身長のデータの標本平均 $\bar{y}$ と不偏分散の平方根 s_y は以下の値になった.

	m	$\bar{x}$	s_x	n	$\bar{y}$	s_y
2012	111	149.9 cm	7.1 cm	131	151.1 cm	6.3 cm
2018	31	153.1 cm	7.9 cm	19	150.1 cm	5.7 cm

このデータについて以下を計算せよ:

(1) 2012年のデータについて, 仮説「平均の差は $\mu_x - \mu_y = 0$ である」に関するWelchの t 検定のP値.

(2) 2012年のデータについて, 平均の差 $\mu_x - \mu_y$ に関するWelchの t 検定に付随する 95% 信頼区間.

(3) 2018年のデータについて, 仮説「平均の差は $\mu_x - \mu_y = 0$ である」に関するWelchの t 検定のP値.

(4) 2018年のデータについて, 平均の差 $\mu_x - \mu_y$ に関するWelchの t 検定に付随する 95% 信頼区間.

(5) 信頼区間の幅がどのようにどのような理由で変わったか?

3.2.1 WolframAlphaによる12歳の男子と女子の平均身長の差に関する必修問題の解答例

以下 $\bar{x}, s_x^2, \bar{y}, s_y^2$ をそれぞれ `x`, `v`, `y`, `w` と書く.

(1)

`(x-y)/sqrt(v/m+w/n) where {m=111 x=149.9, v=7.1^2, n=131, y=151.1, w=6.3^2}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D111+x%3D149.9%2C+v%3D7.1%5E2%2C+n%3D131%2C+y%3D15
→ -1.37911 $= t$ (t値)

`(v/m+w/n)^2/((v/m)^2/(m-1)+(w/n)^2/(n-1)) where {m=111 x=149.9, v=7.1^2, n=131, y=151.1, w=6.3^2}` → 実行 (https://www.wolframalpha.com/input?i=%28v%2Fm%2Bw%2Fn%29%5E2%2F%28%28v%2Fm%29%5E2%2F%28m-1%29%2B%28w%2Fn%29%5E2%2F%28n-1%29%29+where+%7Bm%3D111+x%3D149.9%2C+v%3D7.1%5E2%2C+n%3D131%2C+y%3D151.1%2C+w%3D6.3%5E2%7D)
→ 222.089 $= \nu$ (自由度)

`2(1 - cdf(TDistribution(222.089), 1.37911))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%28222.089%29%2C+1.37911%29%29) → 0.169248 (P値)

(2)

`quantile(TDistribution(222.089), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%28222.089%29%2C+0.975%29) → 1.9707 $= t_{\nu,\alpha/2}$

`{x-y-1.9707*sqrt(v/m+w/n), x-y+1.9707*sqrt(v/m+w/n)} where {m=111 x=149.9, v=7.1^2, n=131, y=151.1, w=6.3^2}` → 実行 (https://www.wolframalpha.com/input?i=%7Bx-y-1.9707*sqrt%28v%2Fm%2Bw%2Fn%29%2C+x-y%2B1.9707*sqrt%28v%2Fm%2Bw%2Fn%29%7D+where+%7Bm%3D111+x%3D149.9%2C+v%3D7.1%5E2%2C+n%3D131%2C+y
→ {-2.91476, 0.51476} (95%信頼区間)

(3)

`(x-y)/sqrt(v/m+w/n) where {m=31, x=153.1, v=7.9^2, n=19, y=150.1, w=5.7^2}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D31%2C+x%3D153.1%2C+v%3D7.9%5E2%2C+n%3D19%2C+y%3D
→ 1.55475 $= t$ (t値)

`(v/m+w/n)^2/((v/m)^2/(m-1)+(w/n)^2/(n-1)) where {m=31, x=153.1, v=7.9^2, n=19, y=150.1, w=5.7^2}` → 実行 (https://www.wolframalpha.com/input?i=%28v%2Fm%2Bw%2Fn%29%5E2%2F%28%28v%2Fm%29%5E2%2F%28m-1%29%2B%28w%2Fn%29%5E2%2F%28n-1%29%29+where+%7Bm%3D31%2C+x%3D153.1%2C+v%3D7.9%5E2%2C+n%3D19%2C+y%3D150.1%2C+w%3D5.7%5E2%7D
→ 46.5881 $= \nu$ (自由度)

`2(1 - cdf(TDistribution(46.5881), 1.55475))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2846.5881%29%2C+1.55475%29%29) → 0.126773 (P値)

(4)

`quantile(TDistribution(46.5881), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%2846.5881%29%2C+0.975%29) → 2.01221 $= t_{\nu,\alpha/2}$

`{x-y-2.01221*sqrt(v/m+w/n), x-y+2.01221*sqrt(v/m+w/n)} where {m=31, x=153.1, v=7.9^2, n=19, y=150.1, w=5.7^2}` → 実行 (https://www.wolframalpha.com/input?i=%7Bx-y-2.01221*sqrt%28v%2Fm%2Bw%2Fn%29%2C+x-y%2B2.01221*sqrt%28v%2Fm%2Bw%2Fn%29%7D+where+%7Bm%3D31%2C+x%3D153.1%2C+v%3D7.9%5E2%2C+n%3D19%2
→ {-0.882692, 6.88269} (95%信頼区間)

3.2.2 Julia言語による12歳の男子と女子の平均身長の差に関する必修問題の解答例

函数の定義についてはこのノートの最初の方を見よ.

In [27]:

```
# (1), (2)
m, x̄, sx² = 111, 149.9, 7.1^2
n, ȳ, sy² = 131, 151.1, 6.3^2
@show tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show degree_of_freedom_welch(m, sx², n, sy²)
@show pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show confint_welch(m, x̄, sx², n, ȳ, sy²; α=0.05);
```

```
tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = -1.3791088014266908
degree_of_freedom_welch(m, sx², n, sy²) = 222.08949764026912
pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = 0.16924845911510278
confint_welch(m, x̄, sx², n, ȳ, sy²; α = 0.05) = [-2.914762206412952, 0.5147622064129749]
```

In [28]:

```
# (3), (4)
m, x̄, sx² = 31, 153.1, 7.9^2
n, ȳ, sy² = 19, 150.1, 5.7^2
@show tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show degree_of_freedom_welch(m, sx², n, sy²)
@show pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show confint_welch(m, x̄, sx², n, ȳ, sy²; α=0.05);
```

```
tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = 1.5547535841186384
degree_of_freedom_welch(m, sx², n, sy²) = 46.58809963805082
pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = 0.12677251777391707
confint_welch(m, x̄, sx², n, ȳ, sy²; α = 0.05) = [-0.8826927055711793, 6.882692705571179]
```

3.2.3 Julia言語による12歳の男子と女子の平均身長の差のP値函数の視覚化

In [29]:

```
plot()
m, x̄, sx² = 111, 149.9, 7.1^2
n, ȳ, sy² = 131, 151.1, 6.3^2
plot(Δμ -> pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ), -5, 9; label="2012", c=1)
vline!([x̄-ȳ]; label="", ls=:dash, c=1)
m, x̄, sx² = 31, 153.1, 7.9^2
n, ȳ, sy² = 19, 150.1, 5.7^2
plot!(Δμ -> pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ), -5, 9; label="2018", c=2)
vline!([x̄-ȳ]; label="", ls=:dash, c=2)
plot!(; xtick=-100:1:100, ytick=0:0.1:1)
vline!([0]; label="", ls=:dot, c=:black)
title!("P-value functions")
```

Out[29]:

3.2.4 R言語による12歳の男子と女子の平均身長の差に関する必修問題の解答例

以下のように与えられた $m, \bar{x}, s_x^2$ に対して, そのような標本平均と不偏分散を持つ擬似的なデータを構成すれば `t.test` 函数で容易に計算できる.

In [30]:

```
function fakedata(m, x̄, sx²)
    if isodd(m)
        [(x̄ + (-1)^k*√sx² for k in 1:m-1)...; x̄]
    else
        [(x̄ + (-1)^k*√sx² for k in 1:m-2)...; x̄-√(sx²/2); x̄+√(sx²/2)]
    end
end

@show x = fakedata(7, 10, 3^2)
@show length(x), mean(x), std(x)
println()
@show x = fakedata(6, 10, 3^2)
@show length(x), mean(x), std(x);
```

```
x = fakedata(7, 10, 3 ^ 2) = [7.0, 13.0, 7.0, 13.0, 7.0, 13.0, 10.0]
(length(x), mean(x), std(x)) = (7, 10.0, 3.0)

x = fakedata(6, 10, 3 ^ 2) = [7.0, 13.0, 7.0, 13.0, 7.878679656440358, 12.121320343559642]
(length(x), mean(x), std(x)) = (6, 10.0, 3.0)
```

In [31]:

```
# (1), (2)
m, x̄, sx² = 111, 149.9, 7.1^2
n, ȳ, sy² = 131, 151.1, 6.3^2
x = fakedata(m, x̄, sx²)
y = fakedata(n, ȳ, sy²)
@show length(x), mean(x), std(x)
@show length(y), mean(y), std(y)
@rput x y
R"""
t.test(x, y, conf.level=0.95)
"""
```

```
(length(x), mean(x), std(x)) = (111, 149.89999999999998, 7.099999999999997)
(length(y), mean(y), std(y)) = (131, 151.1, 6.300000000000011)
```

Out[31]:

```
RObject{VecSxp}

        Welch Two Sample t-test

data:  x and y
t = -1.3791, df = 222.09, p-value = 0.1692
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.9147622  0.5147622
sample estimates:
mean of x mean of y
    149.9     151.1
```

In [32]:

```
# (3), (4)
m, x̄, sx² = 31, 153.1, 7.9^2
n, ȳ, sy² = 19, 150.1, 5.7^2
x = fakedata(m, x̄, sx²)
y = fakedata(n, ȳ, sy²)
@show length(x), mean(x), std(x)
@show length(y), mean(y), std(y)
@rput x y
R"""
t.test(x, y, conf.level=0.95)
"""
```

```
(length(x), mean(x), std(x)) = (31, 153.1, 7.900000000000006)
(length(y), mean(y), std(y)) = (19, 150.1, 5.699999999999989)
```

Out[32]:

```
RObject{VecSxp}

        Welch Two Sample t-test

data:  x and y
t = 1.5548, df = 46.588, p-value = 0.1268
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.8826927  6.8826927
sample estimates:
mean of x mean of y
    153.1     150.1
```

3.2.5 12歳の男子と女子の平均身長の差に関する必修問題の解答例

(1) 0.1692

(2) [-2.9148, 0.5148]

(3) 0.1268

(4) [-0.8827, 6.8827]

(5) 後者の信頼区間の幅の方がかなり広くなっている. その理由はデータのサイズの違いになると考えられる. 前者の2012年のデータは男子111人と女子131人のデータだったが, 後者の2018年のデータは男子31人と女子19人のデータだった.

注意: このことから, データの要約値を報告する場合には必ずデータのサイズの情報も報告する必要があることがわかる.

注意: 以上の計算結果は通常「そのデータだけからは, 2012年と2018年における12歳の男子と女子の平均身長に差があるかどうかは分からない」と解釈される.

3.2.6 仮に2012年のデータのサンプルサイズだけが実際のサイズの2倍だったとしたら

仮に2012年のデータのサンプルサイズだけが実際のサイズの2倍ならば(すなわち m, n が実際のサイズ $m = 31, n = 19$ の2倍だったならば), 「男女の平均の差は 0 である」という仮説のP値は約 13% から約 3% に下がり, 信頼区間の幅も狭くなる.

サンプルサイズはデータに関する最も重要な情報になる.

In [33]:

```
# 2012年のデータのサンプルサイズだけが実際のデータの2倍だった場合
m, x̄, sx² = 31*2, 153.1, 7.9^2
n, ȳ, sy² = 19*2, 150.1, 5.7^2
@show tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show degree_of_freedom_welch(m, sx², n, sy²)
@show pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
@show confint_welch(m, x̄, sx², n, ȳ, sy²; α=0.05);
```

```
tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = 2.1987536048087573
degree_of_freedom_welch(m, sx², n, sy²) = 95.29156620713648
pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ = 0) = 0.03031571373213725
confint_welch(m, x̄, sx², n, ȳ, sy²; α = 0.05) = [0.291411840600591, 5.708588159399409]
```

In [34]:

```
plot()
m, x̄, sx² = 111, 149.9, 7.1^2
n, ȳ, sy² = 131, 151.1, 6.3^2
plot(Δμ -> pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ), -5, 10;
    label="2012", c=1)
vline!([x̄-ȳ]; label="", ls=:dash, c=1)
m, x̄, sx² = 31, 153.1, 7.9^2
n, ȳ, sy² = 19, 150.1, 5.7^2
plot!(Δμ -> pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ), -5, 10;
    label="2018", c=2)
vline!([x̄-ȳ]; label="", ls=:dash, c=2)
m, x̄, sx² = 31*2, 153.1, 7.9^2
n, ȳ, sy² = 19*2, 150.1, 5.7^2
plot!(Δμ -> pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ), -5, 10;
    label="2018 with\ndoubled m,n", c=3)
vline!([x̄-ȳ]; label="", ls=:dash, c=3)
plot!(; xtick=-100:1:100, ytick=0:0.1:1)
vline!([0]; label="", ls=:dot, c=:black)
title!("P-value functions")
plot!(;size=(500, 250))#, legend=:outertopright)
```

Out[34]:

4 Welchの t 検定で使う自由度の式の導出

Welchのt検定の原論文:

- B. L. Welch, The Significance of the Difference Between Two Means when the Population Variances are Unequal, Biometrika, Vol. 29, No. 3/4 (Feb., 1938), pp. 350-362. [Google Scholar (https://scholar.google.co.jp/scholar?cluster=10421437430510190115)]

Welchのt検定で使用される自由度は非常に大雑把な近似を使って導出される.

データサイズが十分に大きな場合には, 自由度が大きな t 分布は標準正規分布でよく近似されるので, Welchの自由度の t 分布を使った補正は実践的には意味がなくなる. 例えば m, n が 100 程度以上のデータを扱う場合には, 以下の複雑な近似の過程をフォローしても益がない.

だから, この節は後の方に回されることになった.

しかし, Welchによる t 分布を使った補正は **Brunner-Munzel 検定** のような他の検定法でも使われているので, 興味がある人はフォローしておくとよいだろう.

- Edgar Brunner and Ullrich Munzel, The Nonparametric Behrens-Fisher Problem: Asymptotic Theory and a Small-Sample Approximation, Biometrical Journal, Volume42, Issue1, January 2000, Pages 17-25. [Google Scholar (https://scholar.google.com/scholar?cluster=5072504718959959846)]

以下で紹介する議論をフォローできた人はこの Brunner-Munzel (2000)にも挑戦してみると良いかもしれない.

4.1 2つの正規分布の標本分布の設定

$X_1, \ldots, X_m$ は各々が平均 μ_x と分散 σ_x^2 を持つ **正規分布** に従う独立同分布な確率変数達であると仮定する.

$Y_1, \ldots, Y_n$ は各々が平均 μ_y と分散 σ_y^2 を持つ **正規分布** に従う独立同分布な確率変数達であると仮定する.

X_i, Y_j の全体も独立であると仮定する.

$X_1, \ldots, X_m$ の不偏分散を S_x^2 と書き, $Y_1, \ldots, Y_n$ の不偏分散を S_y^2 と書くことにする.

このとき, 正規分布の標本分布に関する一般論より, 次が成立している:

$$\frac{(m-1)S_x^2}{\sigma_x^2} \sim \mathrm{Chisq}(m-1), \quad \frac{(n-1)S_y^2}{\sigma_y^2} \sim \mathrm{Chisq}(n-1).$$

S_x^2 と S_y^2 も独立になることに注意せよ. 正規分布の標本分布に関する一般論については

- 「標本分布について」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)

における「正規分布の標本分布の場合」の節を参照せよ. ゆえに

$$S_\Delta^2 := \frac{S_x^2}{m} + \frac{S_y^2}{n} \sim \frac{\sigma_x^2}{m}\frac{\mathrm{Chisq}(m-1)}{m-1} + \frac{\sigma_y^2}{n}\frac{\mathrm{Chisq}(n-1)}{n-1}.$$

$\mathrm{Chisq}(\nu)$ の平均と分散はそれぞれ ν, 2ν なので, 上のように S_Δ^2 とおくと,

$$\begin{aligned} E[S_\Delta^2] &= \frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}, \\ \mathrm{var}(S_\Delta^2) &= \left(\frac{\sigma_x^2}{m}\right)^2 \frac{2}{m-1} + \left(\frac{\sigma_y^2}{n}\right)^2 \frac{2}{n-1} \\ &= 2\left(\frac{(\sigma_x^2/m)^2}{m-1} + \frac{(\sigma_y^2/n)^2}{n-1}\right). \end{aligned}$$

4.2 χ²分布で近似することによる自由度の式の導出

S_Δ^2 の分布を近似するχ²分布の定数倍 $a\,\mathrm{Chisq}(\nu)$ を求めたい. そのために, χ²分布の定数倍 $a\,\mathrm{Chisq}(\nu)$ が S_Δ^2 と同じ平均と分散を持つような正定数 a と自由度 ν を求めよう.

その条件は a と ν に関する次の連立方程式になる:

$$\begin{aligned} a\nu &= (a\,\mathrm{Chisq}(\nu)\text{ の平均}) = E[S_\Delta^2] = \frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}, \\ 2a^2\nu &= (a\,\mathrm{Chisq}(\nu)\text{ の分散}) = \mathrm{var}(S_\Delta^2) = 2\left(\frac{(\sigma_x^2/m)^2}{m-1} + \frac{(\sigma_y^2/n)^2}{n-1}\right). \end{aligned}$$

この方程式を満たす a は後者の式の両辺を 2 で割って, さらに前者の式の両辺で割れば容易に得られる. この方程式を満たす ν はそうやって求めた a を前者に代入すれば得られる. その結果は次の通り:

$$a = \frac{\dfrac{(\sigma_x^2/m)^2}{m-1} + \dfrac{(\sigma_y^2/n)^2}{n-1}}{\dfrac{\sigma_x^2}{m} + \dfrac{\sigma_y^2}{n}}, \quad \nu = \frac{\left(\dfrac{\sigma_x^2}{m} + \dfrac{\sigma_y^2}{n}\right)^2}{\dfrac{(\sigma_x^2/m)^2}{m-1} + \dfrac{(\sigma_y^2/n)^2}{n-1}}.$$

S_Δ^2 と同じ平均と分散を持つχ²分布の定数倍における自由度はこの ν になる.

この ν の値の推定量として次を採用することにする:

$$\hat{\nu} = \frac{\left(\dfrac{S_x^2}{m} + \dfrac{S_y^2}{n}\right)^2}{\dfrac{(S_x^2/m)^2}{m-1} + \dfrac{(S_y^2/n)^2}{n-1}}.$$

これは大数の法則によって $S_x^2 \approx \sigma_x^2$, $S_y^2 \approx \sigma_y^2$ という近似が成立しているならば, 上の ν を近似することになる.

これがWelchの t 検定で使われる t 分布の自由度の式にちょうどなっている.

ポイントと注意: Welchの t 検定で使われる t 分布の自由度の式は, $S_\Delta^2 = S_x^2/m + S_y^2/n$ の分布と同じ平均と分散を持つχ²分布の定数倍 $a\,\mathrm{Chisq}(\nu)$ を求めることによって得られる. S_Δ^2 の分布を $a\,\mathrm{Chisq}(\nu)$ で近似することはかなり大胆な近似であると考えられる.

4.3 t 分布による近似

X_i 達と Y_j 達は独立な正規分布に従う確率変数達であるという仮定から, 次の確率変数 Z は標準正規分布に従うことがわかる:

$$Z = \frac{\bar{X} - \bar{Y} - (\mu_x - \mu_y)}{\sqrt{\dfrac{\sigma_x^2}{m} + \dfrac{\sigma_y^2}{n}}}.$$

確率変数 W を次のように定める:

$$W = \frac{S_\Delta^2}{a} = \frac{1}{a}\left(\frac{S_x^2}{m} + \frac{S_y^2}{n}\right).$$

X_i 達と Y_j 達は独立な正規分布に従う確率変数達であるという仮定から, $\bar{X}, S_x^2, \bar{Y}, S_y^2$ は独立な確率変数達になる. この点については,

- [「標本分布について」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)

における「正規分布の標本分布の場合」の節に書いてある正規分布の標本分布に関する一般論を参照せよ.

ゆえに, Z, W は独立になる. そして, 前節の結果より,

$$\frac{W}{\nu} = \frac{S_\Delta^2}{a\nu}, \quad a\nu = \frac{\sigma_x^2}{m} + \frac{\sigma_y^2}{n}, \quad S_\Delta^2 = \frac{S_x^2}{m} + \frac{S_y^2}{n}$$

なので,

$$\begin{aligned}
\frac{Z}{\sqrt{W/\nu}} &= \frac{\bar{X} - \bar{Y} - (\mu_x - \mu_y)}{\sqrt{\dfrac{\sigma_x^2}{m} + \dfrac{\sigma_y^2}{n}}} \times \sqrt{\frac{a\nu}{S_\Delta^2}} \\
&= \frac{\bar{X} - \bar{Y} - (\mu_x - \mu_y)}{\sqrt{S_\Delta^2}} \\
&= \frac{\bar{X} - \bar{Y} - (\mu_x - \mu_y)}{\sqrt{\dfrac{S_x^2}{m} + \dfrac{S_y^2}{n}}} = T(\mu_x - \mu_y).
\end{aligned}$$

前節の結果より, W の分布を近似するχ²分布は $\mathrm{Chisq}(\nu)$ になるので, $T(\mu_x - \mu_y)$ の分布を近似する t 分布の自由度も ν だと考えられる.

その ν を近似する統計量として,

$$\hat{\nu} = \frac{\left(\dfrac{S_x^2}{m} + \dfrac{S_y^2}{n}\right)^2}{\dfrac{(S_x^2/m)^2}{m-1} + \dfrac{(S_y^2/n)^2}{n-1}}.$$

を採用すれば, Welchの t 検定の方法が得られる.

5 Welchの t 検定での第一種の過誤の確率の視覚化

平均 $\mu_{x,0}$ の分布 D_x でサイズ m のサンプルを, 平均 $\mu_{y,0}$ の分布 D_y でサイズ n のサンプルを大量に生成して, そのサンプル達に関する「平均の差は $\mu_x - \mu_y = \mu_{x,0} - \mu_{y,0}$ えある」という仮説のWelchの t 検定におけるP値が有意水準 α 以下(未満でも同じ)になる確率(第一種の過誤の確率と呼ばれる)を求めてプロットしてみよう.

理想的には第一種の過誤の確率は有意水準に等しくなって欲しいが, Welchの t 検定は中心極限定理や対数の法則を使った近似を使っているので, データのサイズが十分に大きくない場合には誤差が生じてしまう.

その様子をグラフで確認することが以下の目標である.

3×2 のプロットの内訳は以下の通り:

サイズ m のデータ X_i を生成する分布	サイズ n のデータ Y_i を生成する分布
T 統計量と自由度の分布	T 統計量の分布
第一種の過誤の確率 $0 < \alpha < 1$	第一種の過誤の確率 $0 < \alpha < 0.1$

5.1 異なる分散を持つ正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

In [35]:
```
plot_welch(distx=Normal(0,1), m=10, disty=Normal(0,2), n=10)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
ν̄ = mean(df) = 13.540037126888508
```

Out[35]:

In [36]:
```
plot_welch(distx=Normal(0,1), m=10, disty=Normal(0,2), n=40)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 40,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
ν̄ = mean(df) = 31.1515464432953
```

Out[36]:

In [37]:
```
plot_welch(distx=Normal(0,1), m=40, disty=Normal(0,2), n=10)
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
ν̄ = mean(df) = 10.501568177803936
```

Out[37]:

5.2 異なる分散を持つ一様分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

2つのサンプル x_i と y_j のサイズ m, n のアンバランスは誤差増大の原因になり得る.

In [38]:
```
plot_welch(distx=Uniform(-1,1), m=10, disty=Uniform(-2,2), n=10,
    distxlim=(-2.5, 2.5), distylim=(-2.5, 2.5))
```

```
m = 10,  μ_x = 0.000,  σ_x = 0.577,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 10,  μ_y = 0.000,  σ_y = 1.155,  skewness_y = 0.000,  kurtosis_y = -1.200
Δμ = 0.0
ν̄ = mean(df) = 13.44142252050102
```

Out[38]:

In [39]:

```
plot_welch(distx=Uniform(-1,1), m=10, disty=Uniform(-2,2), n=40,
    distxlim=(-2.5, 2.5), distylim=(-2.5, 2.5))
```

```
m = 10,  μ_x = 0.000,  σ_x = 0.577,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 40,  μ_y = 0.000,  σ_y = 1.155,  skewness_y = 0.000,  kurtosis_y = -1.200
Δμ = 0.0
ν̄ = mean(df) = 30.333388674084826
```

Out[39]:

In [40]:

```
plot_welch(distx=Uniform(-1,1), m=40, disty=Uniform(-2,2), n=10,
    distxlim=(-2.5, 2.5), distylim=(-2.5, 2.5))
```

```
m = 40,  μ_x = 0.000,  σ_x = 0.577,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 10,  μ_y = 0.000,  σ_y = 1.155,  skewness_y = 0.000,  kurtosis_y = -1.200
Δμ = 0.0
ν̄ = mean(df) = 10.319694128395549
```

Out[40]:

5.3 正規分布とガンマ分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

正規分布の標本と左右非対称な分布の平均を比較したい場合には, 後者の左右非対称な分布の側のサンプルサイズを大きくした方が誤差が小さくなり易そうなことを以下の計算は示唆している.

In [41]:
```
plot_welch(distx=Normal(0,1), m=10, disty=Gamma(2,3), n=10)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 10.507446436415803
```

Out[41]:

In [42]:
```
plot_welch(distx=Normal(0,1), m=10, disty=Gamma(2,3), n=40)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 40,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 45.92679392427271
```

Out[42]:

In [43]:
```
plot_welch(distx=Normal(0,1), m=10, disty=Gamma(2,3), n=160)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 160,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 47.349959632075056
```

Out[43]:

In [44]:
```
plot_welch(distx=Normal(0,1), m=40, disty=Gamma(2,3), n=10)
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 9.401955759038113
```

Out[44]:

In [45]:
```
plot_welch(distx=Normal(0,1), m=160, disty=Gamma(2,3), n=10)
```

```
m = 160,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 9.099512063767147
```

Out[45]:

5.4 一様分布とガンマ分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

In [46]:

```
plot_welch(distx=Uniform(-2,2), m=10, disty=Gamma(2,3), n=10,
    distxlim=(-2.5, 2.5))
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.155,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 10.982748150653544
```

Out[46]:

In [47]:

```
plot_welch(distx=Uniform(-2,2), m=10, disty=Gamma(2,3), n=40,
    distxlim=(-2.5, 2.5))
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.155,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 40,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 45.401533049798104
```

Out[47]:

In [48]:

```
plot_welch(distx=Uniform(-2,2), m=40, disty=Gamma(2,3), n=10,
    distxlim=(-2.5, 2.5))
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.155,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 9.537927066646965
```

Out[48]:

In [49]:

```
plot_welch(distx=Uniform(-2,2), m=40, disty=Gamma(2,3), n=40,
    distxlim=(-2.5, 2.5))
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.155,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 40,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 45.437985835525495
```

Out[49]:

In [50]:
```
plot_welch(distx=Uniform(-2,2), m=40, disty=Gamma(2,3), n=160,
    distxlim=(-2.5, 2.5))
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.155,  skewness_x = 0.000,  kurtosis_x = -1.200
n = 160,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = -6.0
ν̄ = mean(df) = 194.60051887802356
```

Out[50]:

5.5 同一のガンマ分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

X_i 達と Y_j 達が同一の左右非対称な分布のサンプルであるとき, 2つのサンプルサイズを同じにした方が誤差が小さくなりそうなことを以下の計算結果は示唆している.

In [51]:
```
plot_welch(distx=Gamma(2,3), m=10, disty=Gamma(2,3), n=10)
```

```
m = 10,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 10,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 15.799269001324523
```

Out[51]:

In [52]:
```
plot_welch(distx=Gamma(2,3), m=10, disty=Gamma(2,3), n=40)
```

```
m = 10,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 40,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 16.687696645629813
```

Out[52]:

In [53]:
```
plot_welch(distx=Gamma(2,3), m=10, disty=Gamma(2,3), n=160)
```

```
m = 10,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 160,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 10.921556361918844
```

Out[53]:

In [54]:
```
plot_welch(distx=Gamma(2,3), m=40, disty=Gamma(2,3), n=160)
```

```
m = 40,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 160,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 62.91931721573467
```

Out[54]:

In [55]:
```
plot_welch(distx=Gamma(2,3), m=160, disty=Gamma(2,3), n=160)
```

```
m = 160,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 160,  μ_y = 6.000,  σ_y = 4.243,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 313.4552897349271
```

Out[55]:

5.6 異なるガンマ分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

In [56]:
```
plot_welch(distx=Gamma(6,2), m=10, disty=Gamma(2,6), n=10)
```

```
m = 10,  μ_x = 12.000,  σ_x = 4.899,  skewness_x = 0.816,  kurtosis_x = 1.000
n = 10,  μ_y = 12.000,  σ_y = 8.485,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 14.517640411168712
```

Out[56]:

In [57]:
```
plot_welch(distx=Gamma(6,2), m=10, disty=Gamma(2,6), n=40)
```

```
m = 10,  μ_x = 12.000,  σ_x = 4.899,  skewness_x = 0.816,  kurtosis_x = 1.000
n = 40,  μ_y = 12.000,  σ_y = 8.485,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 27.18576226375518
```

Out[57]:

In [58]:
```
plot_welch(distx=Gamma(6,2), m=40, disty=Gamma(2,6), n=10)
```

```
m = 40,  μ_x = 12.000,  σ_x = 4.899,  skewness_x = 0.816,  kurtosis_x = 1.000
n = 10,  μ_y = 12.000,  σ_y = 8.485,  skewness_y = 1.414,  kurtosis_y = 3.000
Δμ = 0.0
ν̄ = mean(df) = 11.532330677616473
```

Out[58]:

5.7 正規分布と対数正規分布の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

In [59]:
```
plot_welch(distx=Normal(0,1), m=10, disty=LogNormal(), n=10)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 10,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -1.6487212707001282
ν̄ = mean(df) = 14.485474368552183
```

Out[59]:

In [60]:
```
plot_welch(distx=Normal(0,1), m=10, disty=LogNormal(), n=40)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 40,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -1.6487212707001282
ν̄ = mean(df) = 28.555399482642507
```

Out[60]:

In [61]:
```
plot_welch(distx=Normal(0,1), m=10, disty=LogNormal(), n=160)
```

```
m = 10,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 160,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -1.6487212707001282
ν̄ = mean(df) = 17.402741365856087
```

Out[61]:

5.8 同一の対数正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

In [62]:
```
plot_welch(distx=LogNormal(), m=10, disty=LogNormal(), n=10)
```

```
m = 10,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 10,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = 0.0
ν̄ = mean(df) = 14.114896151322853
```

Out[62]:

In [63]:
```
plot_welch(distx=LogNormal(), m=10, disty=LogNormal(), n=40)
```

```
m = 10,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 40,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = 0.0
ν̄ = mean(df) = 22.14784395106809
```

Out[63]:

In [64]:
```
plot_welch(distx=LogNormal(), m=40, disty=LogNormal(), n=40)
```

```
m = 40,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 40,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = 0.0
ν̄ = mean(df) = 67.11277328855677
```

Out[64]:

In [65]:
```
plot_welch(distx=LogNormal(), m=40, disty=LogNormal(), n=160)
```

```
m = 40,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 160,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = 0.0
ν̄ = mean(df) = 76.55500798158911
```

Out[65]:

In [66]:
```
plot_welch(distx=LogNormal(), m=160, disty=LogNormal(), n=160)
```

```
m = 160,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 160,  μ_y = 1.649,  σ_y = 2.161,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = 0.0
ν̄ = mean(df) = 293.3332485524279
```

Out[66]:

5.9 異なる対数正規分布達の標本でのWelchの t 検定での第一種の過誤の確率の視覚化

以下はWelchの t 検定およびそれに付随する信頼区間の使用が危険になる場合があることがわかる計算例になっている.

In [67]:
```
plot_welch(distx=LogNormal(), m=10, disty=LogNormal(1), n=10)
```

```
m = 10,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 10,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 12.305907287778117
```

Out[67]:

In [68]:
```
plot_welch(distx=LogNormal(), m=10, disty=LogNormal(1), n=40)
```

```
m = 10,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 40,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 38.24579164803262
```

Out[68]:

In [69]:
```
plot_welch(distx=LogNormal(), m=40, disty=LogNormal(1), n=40)
```

```
m = 40,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 40,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 52.60583836725731
```

Out[69]:

In [70]:
```
plot_welch(distx=LogNormal(), m=40, disty=LogNormal(1), n=160)
```

```
m = 40,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 160,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 165.3019258372765
```

Out[70]:

In [71]:
```
plot_welch(distx=LogNormal(), m=160, disty=LogNormal(1), n=160)
```

```
m = 160,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 160,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 208.6607725137111
```

Out[71]:

In [72]:
```
plot_welch(distx=LogNormal(), m=160, disty=LogNormal(1), n=640)
```

```
m = 160,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 640,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 684.2386220164234
```

Out[72]:

In [73]:
```
plot_welch(distx=LogNormal(), m=640, disty=LogNormal(1), n=640)
```

```
m = 640,  μ_x = 1.649,  σ_x = 2.161,  skewness_x = 6.185,  kurtosis_x = 110.936
n = 640,  μ_y = 4.482,  σ_y = 5.875,  skewness_y = 6.185,  kurtosis_y = 110.936
Δμ = -2.8329677996379363
ν̄ = mean(df) = 821.3712069009249
```

Out[73]:

6 補足: Studentの t 検定について

このノートでは2群の平均の比較に関するStudentの t 検定は扱わずに, Welchの t 検定のP値と信頼区間で押し通すことにした. しかし, 読者の便のためにStudentの t 検定のP値と信頼区間についても補足しておくことにする.

6.1 Studentの t 検定のP値と信頼区間の定義

扱うデータの形式はWelchの t 検定の場合と同じである.

データ: m 個の実数値 $x_1, \ldots, x_m$ と n 個の実数値 $y_1, \ldots, y_n$.

x_i 達と y_i 達の標本平均と不偏分散を以下のように書くことにする:

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i, \quad s_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2,$$
$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \quad s_y^2 = \frac{1}{n-1}\sum_{i=1}^{m}(y_i - \bar{y})^2.$$

Studentの t 検定ではWelchの t 検定の場合と違って **等分散の条件を仮定** をする.

統計モデル: 平均 μ_x と分散 σ_x^2 を持つ確率分布 D_x のサイズ m の標本分布 D_x^m と平均 μ_y と分散 σ_y^2 を持つ確率分布 D_y のサイズ n の標本分布 D_y^n の直積分布 $D_x^m \times D_y^n$ を統計モデルとして採用する. ただし, **等分散の条件 $\sigma_x^2 = \sigma_y^2 = \sigma^2$ を仮定する.**

このモデルの確率分布に従う確率変数達(独立になる)を $X_1, \ldots, X_m, Y_1, \ldots, Y_n$ と書き, X_i 達と Y_i 達の標本平均と不偏分散を以下のように書くことにする:

$$\bar{X} = \frac{1}{m}\sum_{i=1}^{m} X_i, \quad S_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(X_i - \bar{X})^2,$$
$$\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i, \quad S_y^2 = \frac{1}{n-1}\sum_{i=1}^{m}(Y_i - \bar{Y})^2.$$

この統計モデルが適用可能な場合: 分散が等しい(もしくはほぼ等しい)2つの母集団からの無作為抽出で x_i, y_j のそれぞれが得られている場合には上のモデルの適用は適切になる.

検定したい仮説: $\mu_x - \mu_y = \Delta\mu$ ($\Delta\mu$ は具体的な数値).

中心極限定理: モデル内確率変数としての2つの標本平均達の分布について, 中心極限定理による正規分布近似が使えると仮定する. (等分散の条件 $\sigma_x^2 = \sigma_y^2 = \sigma^2$ が仮定されていることに注意せよ.)

$\bar{X}, \bar{Y}$ の平均(期待値)と分散は以下のようになる:

$$E[\bar{X}] = \mu_x, \quad E[\bar{Y}] = \mu_y, \quad \mathrm{var}(\bar{X}) = \frac{\sigma^2}{m}, \quad \mathrm{var}(\bar{Y}) = \frac{\sigma^2}{n}.$$

さらに, $\bar{X}$ と $\bar{Y}$ が確率変数として独立であることより,

$$E[\bar{X} - \bar{Y}] = \mu_x - \mu_y, \quad \mathrm{var}(\bar{X} - \bar{Y}) = \sigma^2\left(\frac{1}{m} + \frac{1}{n}\right).$$

さらに, 中心極限定理より, 次の近似が使える:

$$\bar{X} - \bar{Y} \sim \mathrm{Normal}\left(\mu_x - \mu_y, \sqrt{\sigma^2\left(\frac{1}{m} + \frac{1}{n}\right)}\right), \quad \text{approximately.}$$

すなわち,

$$\frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{\sqrt{\sigma^2\left(\frac{1}{m} + \frac{1}{n}\right)}} \sim \mathrm{Normal}(0, 1), \quad \text{approximately.}$$

大数の法則: 確率変数 S^2 を次のように定義する:

$$S^2 = \frac{(m-1)S_x^2 + (n-1)S_y^2}{m+n-2}.$$

このとき,

$$E[S^2] = \frac{(m-1)E[S_x^2] + (n-1)E[S_y^2]}{m+n-2} = \frac{(m-1)\sigma^2 + (n-1)\sigma^2}{m+n-2} = \sigma^2$$

より, S^2 は σ^2 の不偏推定量である. そして, さらに大数の法則によって, m, n が十分に大きいならば, S^2 は σ^2 をよく近似するようになる.

このとき, 次の近似が使える:

$$T := \frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{\sqrt{S^2\left(\frac{1}{m} + \frac{1}{n}\right)}} \sim \mathrm{Normal}(0, 1), \quad \text{approximately.}$$

この T の定義はWelchの t 検定で使った T の定義とは分母の形が本質的に異なる.

P値の定義: これを用いて, 具体的に与えられた数値 $\Delta\mu$ に関する仮説「$\mu_x - \mu_y = \Delta\mu$」のP値を以下のように定義する. まず,

$$s^2 = \frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2}$$

とおき, データの数値の t 値 $t = t(\Delta\mu)$ を次のように定義する:

$$t = t(\Delta\mu) = \frac{(\bar{x} - \bar{y}) - \Delta\mu}{\sqrt{s^2\left(\frac{1}{m} + \frac{1}{n}\right)}}.$$

この $t = t(\Delta\mu)$ の定義はWelchの t 検定で使った $t = t(\Delta\mu)$ の定義とは分母の形が本質的に異なる.

仮説「$\mu_x - \mu_y = \Delta\mu$」のP値を, その仮説下のモデル内の確率変数としての t 値 T の値の絶対値がデータから計算した t 値 $t = t(\Delta\mu)$ の絶対値以上になる確率の近似値として定義する:

$$\mathrm{pvalue}_{\mathrm{Normal}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \mu_x - \mu_y = \Delta\mu) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |t(\Delta\mu)|)).$$

しかし, 実際に使用されるのは以下のようにして t 分布を使って補正されたP値の方である.

モデルの分布として正規分布を採用: モデルの分布 D_x, D_y は正規分布であると仮定する.

正規分布の標本分布で成立していること: そのとき, 上のように定義された $\bar{X}, \bar{Y}, S_x^2, S_y^2$ は独立になり, 以下を近似無しで満たしている:

$$\bar{X} \sim \mathrm{Normal}\left(\mu_x, \sqrt{\frac{\sigma^2}{m}}\right), \qquad \bar{Y} \sim \mathrm{Normal}\left(\mu_y, \sqrt{\frac{\sigma^2}{n}}\right),$$
$$\frac{(m-1)S_x^2}{\sigma^2} \sim \mathrm{Chisq}(m-1), \quad \frac{(n-1)S_y^2}{\sigma^2} \sim \mathrm{Chisq}(n-1).$$

前者の2つより

$$Z := \frac{(\bar{X} - \bar{Y}) - (\mu_x - \mu_y)}{\sqrt{\sigma^2\left(\frac{1}{m} + \frac{1}{n}\right)}} \sim \mathrm{Normal}(0,1)$$

が導かれ, 一般に A, B がそれぞれ $\mathrm{Chisq}(a)$, $\mathrm{Chisq}(b)$ に従う独立な確率変数達であるとき $A + B$ が $\mathrm{Chisq}(a+b)$ に従うことを使うと,

$$W := \frac{(m+n-2)S^2}{\sigma^2} = \frac{(m-1)S_x^2 + (n-1)S_y^2}{\sigma^2} \sim \mathrm{Chisq}(m+n-2)$$

が得られる. このことから, 上で定義した T について

$$T = \frac{Z}{\sqrt{W/(m+n-2)}} \sim \mathrm{TDist}(m+n-2)$$

が成立していることが導かれる.

t分布を使って補正されたP値の定義: Studentの t 検定のP値を定義しよう:

$$\mathrm{pvalue}_{\mathrm{Student}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \mu_x - \mu_y = \Delta\mu) = 2(1 - \mathrm{cdf}(\mathrm{TDist}(m+n-2), |t(\Delta\mu)|)).$$

右辺の形がWelchの t 検定と似ているが, $t(\Delta\mu)$ の定義式の分母がWelchの t 検定の場合と本質的に異なり, t 分布の自由度が $m+n-2$ と定数になる点が違っている.

この補正は, $m+n-2$ が大きなとき, $\mathrm{TDist}(m+n-2)$ はほぼ標準正規分布になるので, 些細な補正に留まる.

信頼区間: Welchの t 検定のP値から定まる信頼区間は以下のように書ける.

まず, 自由度 ν の t 分布において $t_{\nu,\alpha/2}$ 以上になる確率は $\alpha/2$ になると仮定する:

$$t_{\nu,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(\nu), 1 - \alpha/2).$$

このとき, 平均の差 $\mu_x - \mu_y$ の信頼度 $1 - \alpha$ の信頼区間が次のように定義される:

$$
\begin{aligned}
&\mathrm{confint}_{\mathrm{Student}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \alpha) \\
&= \left[\bar{x} - \bar{y} - t_{m+n-2,\alpha/2} \sqrt{s^2 \left(\frac{1}{m} + \frac{1}{n} \right)},\ \bar{x} - \bar{y} + t_{m+n-2,\alpha/2} \sqrt{s^2 \left(\frac{1}{m} + \frac{1}{n} \right)} \right].
\end{aligned}
$$

ここで,

$$
s^2 = \frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2}.
$$

6.2 Studentの t 検定とWelchの t 検定に付随する信頼区間の比較

2つの信頼区間の定義を並べて書いてみよう:

$$
\begin{aligned}
&\mathrm{confint}_{\mathrm{Welch}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \alpha) \\
&= \left[\bar{x} - \bar{y} - t_{\nu,\alpha/2} \sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}},\ \bar{x} - \bar{y} + t_{\nu,\alpha/2} \sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}} \right], \\
&\mathrm{confint}_{\mathrm{Student}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \alpha) \\
&= \left[\bar{x} - \bar{y} - t_{m+n-2,\alpha/2} \sqrt{s^2 \left(\frac{1}{m} + \frac{1}{n} \right)},\ \bar{x} - \bar{y} + t_{m+n-2,\alpha/2} \sqrt{s^2 \left(\frac{1}{m} + \frac{1}{n} \right)} \right].
\end{aligned}
$$

ここで,

$$
s^2 = \frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2}.
$$

ν の定義式は省略した. しかし, m, n が大きいとき, ν と $m+n-2$ も大きくなり, $t_{\nu,\alpha/2}$ と $t_{m+n-2,\alpha/2}$ の値はほぼ等しくなる. その場合に, これら2つの信頼区間の違いは

$$
\begin{aligned}
\widehat{\mathrm{se}}^2_{\mathrm{Welch}} &= \frac{s_x^2}{m} + \frac{s_y^2}{n}, \\
\widehat{\mathrm{se}}^2_{\mathrm{Student}} &= s^2 \left(\frac{1}{m} + \frac{1}{n} \right) = \frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2} \left(\frac{1}{m} + \frac{1}{n} \right)
\end{aligned}
$$

の大きさの違いによって生じることになる. これらの差は次のようになる:

$$
\widehat{\mathrm{se}}^2_{\mathrm{Student}} - \widehat{\mathrm{se}}^2_{\mathrm{Welch}} = \frac{m+n-1}{mn(m+n-2)}(m-n)(s_x^2 - s_y^2).
$$

手計算による証明も難しくないが, 数式処理による証明については以下を見よ.

In [74]:

```
@vars s²_x s²_y m n
expr = ((m-1)*s²_x+(n-1)*s²_y)/(m+n-2)*(1/m+1/n) - (s²_x/m + s²_y/n)
Eq(expr, expr.expand().factor())
```

Out[74]:

$$
\frac{\left(\frac{1}{n} + \frac{1}{m}\right)\left({s^2}_x (m-1) + {s^2}_y (n-1)\right)}{m+n-2} - \frac{{s^2}_y}{n} - \frac{{s^2}_x}{m} = \frac{(m-n)\left({s^2}_x - {s^2}_y\right)(m+n-1)}{mn\,(m+n-2)}
$$

したがって, 以下が成立している.

(1) $s_x^2 \approx s_y^2$ または $m \approx n$ のとき, $\widehat{\mathrm{se}}^2_{\mathrm{Student}} \approx \widehat{\mathrm{se}}^2_{\mathrm{Welch}}$ となり, 2つの信頼区間はほぼ等しくなる.

すなわち, 等サンプルサイズまたは等分散の場合にはStundentの t 検定とWelchの t 検定のP値達およびそれらに付随する信頼区間達はほぼ同じになる.

(2) $\widehat{\mathrm{se}}^2_{\mathrm{Student}}$ は $\widehat{\mathrm{se}}^2_{\mathrm{Welch}}$ よりも $(m-n)(s_x^2 - s_y^2)$ に比例して大きくなる. 例えば, $m < n$ のとき, $s_x^2 > s_y^2$ ならば $\widehat{\mathrm{se}}^2_{\mathrm{Student}} > \widehat{\mathrm{se}}^2_{\mathrm{Welch}}$ となり, $s_x^2 < s_y^2$ ならば $\widehat{\mathrm{se}}^2_{\mathrm{Student}} < \widehat{\mathrm{se}}^2_{\mathrm{Welch}}$ となる.

等サンプルサイズでも等分散でない場合にはStundentの t 検定とWelchの t 検定のP値達およびそれらに付随する信頼区間達のあいだには違いが生じる.

等分散でない場合の標準誤差の推定量として正しいのは $\widehat{\mathrm{se}}_{\mathrm{Welch}}$ の方なので, 等サンプルサイズでも等分散でない場合には, Studentの t 検定のP値とそれに付随する信頼区間の誤差は大きくなる.

6.3 Studentの t 検定とWelchの t 検定で使用する t 分布の自由度の比較

Welchの t 検定で使用する t 分布の自由度は

$$\nu = \frac{\left(s_x^2/m + s_y^2/n\right)^2}{\dfrac{(s_x^2/m)^2}{m-1} + \dfrac{(s_y^2/n)^2}{n-1}}.$$

であった. このとき,

$$(m+n-2) - \nu = \frac{\left(s_x^2/(m(m-1)) - s_y^2/(n(n-1))\right)^2}{\dfrac{(s_x^2/(m(m-1)))^2}{n-1} + \dfrac{(s_y^2/(n(n-1)))^2}{m-1}} \geq 0.$$

これの手計算での確認は面倒なので数式処理を使って確認してみよう.

In [75]:

```
@vars s²_x s²_y m n ν
ν = (s²_x/m + s²_y/n)^2/((s²_x/m)^2/(m-1) + (s²_y/n)^2/(n-1))
lhs = (m+n-2) - ν
rhsnum = (s²_x/(m*(m-1)) - s²_y/(n*(n-1)))^2
rhsden = (s²_x/(m*(m-1)))^2/(n-1) + (s²_y/(n*(n-1)))^2/(m-1)
@show (lhs - rhsnum/rhsden).simplify()
Eq(lhs, rhsnum/rhsden)
```

```
(lhs - rhsnum / rhsden).simplify() = 0
```

Out[75]:

$$m + n - \frac{\left(\frac{s^2{}_y}{n} + \frac{s^2{}_x}{m}\right)^2}{\frac{s^2{}_y^2}{n^2(n-1)} + \frac{s^2{}_x^2}{m^2(m-1)}} - 2 = \frac{\left(-\frac{s^2{}_y}{n(n-1)} + \frac{s^2{}_x}{m(m-1)}\right)^2}{\frac{s^2{}_y^2}{n^2(m-1)(n-1)^2} + \frac{s^2{}_x^2}{m^2(m-1)^2(n-1)}}$$

このことから, Welchの t 検定で使用する自由度 ν はStudentの t 検定で使用する自由度 $m+n-2$ 以下になることがわかる.

6.4 Studentの t 検定とWelchの t 検定が異なる結果を与える例

以下のデータではStudentの t 検定側のP値が小さくなり, 信頼区間も狭くなる.

In [76]:

```
x = [-3.5, 0.0, 0.5, 0.8, 1.3, 0.3, 1.8, 0.6, -0.2, 0.2,
    0.8, 0.7, 0.4, -0.2, 1.2, -0.3, 0.9, 2.1, 0.2, -1.0,
    -1.0, 1.2, -0.7, 0.5, 0.4, -0.3, -1.8, -1.0, 0.4, 0.8,
    0.0, 2.2, -0.3, -0.4, -0.9, -1.0, 0.5, -1.7, 2.1, -0.1]
y = [-3.9, 0.0, -1.8, 3.3, -3.6, 2.2, 1.4, -1.4, -0.7, -0.7,
    -1.7, 1.2, -0.7, 1.0, -3.7, -1.8, -0.7, 0.3, -1.0, -1.3];
```

In [77]:

```
@show pvalue_student(x, y)
@show pvalue_welch(x, y);
```

```
pvalue_student(x, y) = 0.04133234806978632
pvalue_welch(x, y) = 0.08831793908916963
```

In [78]:

```
@show confint_student(x, y)
@show confint_welch(x, y);
```

```
confint_student(x, y) = [0.03324861267191648, 1.6017513873280838]
confint_welch(x, y) = [-0.13147593952109426, 1.7664759395210945]
```

In [79]:
```
plot_confint_of_diffmeans(x, y; plot_pvaluefunc=true, plot_student=true,
    size=(600, 300), legend=:topleft)
```

Out[79]:

6.5 Studentの t 検定での第一種の過誤の確率の視覚化

以下のグラフを見ると, 等分散でなくても等サンプルサイズならばStudentの t 検定は使えそうに見える.

In [80]:
```
plot_student(distx=Normal(), m=40, disty=Normal(0,2), n=40)
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 40,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
```

Out[80]:

しかし, 以下のように等分散でなくてかつ等サンプルサイズでない場合にはStudentの t 検定の方法では大きな誤差が生じてしまう.

In [81]:
```
plot_student(distx=Normal(), m=40, disty=Normal(0,2), n=80)
```

```
m = 40,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 80,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
```

Out[81]:

In [82]:
```
plot_student(distx=Normal(), m=80, disty=Normal(0,2), n=40)
```

```
m = 80,  μ_x = 0.000,  σ_x = 1.000,  skewness_x = 0.000,  kurtosis_x = 0.000
n = 40,  μ_y = 0.000,  σ_y = 2.000,  skewness_y = 0.000,  kurtosis_y = 0.000
Δμ = 0.0
```

Out[82]:

以上の2つの場合にはStudentの t 検定の脆弱さとWelchの t 検定の頑健さが印象的である.

以下のように等サンプルサイズにすれば左右非対称で等分散でない場合にもStudentの t 検定は十分に使用可能なように見える.

In [83]:
```
plot_student(distx=Gamma(2, 3), m=40, disty=Gamma(3,2), n=40)
```

```
m = 40,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 40,  μ_y = 6.000,  σ_y = 3.464,  skewness_y = 1.155,  kurtosis_y = 2.000
Δμ = 0.0
```

Out[83]:

In [84]:
```
plot_student(distx=Gamma(2, 3), m=40, disty=Gamma(3,2), n=50)
```

```
m = 40,  μ_x = 6.000,  σ_x = 4.243,  skewness_x = 1.414,  kurtosis_x = 3.000
n = 50,  μ_y = 6.000,  σ_y = 3.464,  skewness_y = 1.155,  kurtosis_y = 2.000
Δμ = 0.0
```

Out[84]:

このように等サンプルサイズになるように($m = n$ になるように)注意を払っていれば, Studentの t 検定も十分に実用的であると考えられる.

In [ ]:

```
1
```

誤用を避けるための注意

- 黒木玄

- 2022-07-01〜2022-07-11

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 データ全体の様子を適切なグラフ作画によって確認せよ！
  - 1.1 Anscombeの例
  - 1.2 問題: Anscombeの例
  - 1.3 データサウルス
  - 1.4 問題: データサウルス
  - 1.5 箱ひげ図は十分に大きなサイズの単峰型のデータにのみ使用すること
  - 1.6 不適切な箱ひげ図の例
  - 1.7 箱ひげ図などの正しい使い方
  - 1.8 ゴリラ
  - 1.9 問題: ゴリラ
- 2 印象操作のためにグラフを利用しない
  - 2.1 3次元円グラフは絶対に避ける
- 3 無作為抽出の失敗に注意を払う
  - 3.1 1936年のアメリカ大統領選挙の結果の予測の失敗
  - 3.2 1948年のアメリカ大統領選挙の結果の予測の失敗
  - 3.3 選挙の予測に関する参考文献
- 4 2つのグループを比較するときの注意
  - 4.1 データは偏っていることが多い
  - 4.2 2つのグループを比較するときの注意に関する参考資料
- 5 P値や信頼区間の誤用
  - 5.1 P値の定義
  - 5.2 P値の正しい使い方
  - 5.3 復習: 累積分布函数 cdf と分位点函数 quantile の定義
  - 5.4 信頼区間の定義
  - 5.5 信頼区間の正しい解釈の仕方
  - 5.6 P値の誤用の例
    - 5.6.1 P値は仮説が正しい確率ではない
    - 5.6.2 単独のP値のみを使って重要な決定をしてはいけない
    - 5.6.3 すべてを正直に報告しなければいけない
  - 5.7 P値や信頼区間の誤用に関する参考資料

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Printf
using QuadGK
using RCall
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

In [4]:
```
function plot_ols(x, y;
        ms=4, lw=1.5, alpha=0.8, xguide="", yguide="",
        xlim=nothing, ylim=nothing, kwargs...)
    X = x .^ (0:1)'
    α̂, β̂ = X \ y

    if isnothing(xlim)
        a, b = extrema(x)
        xlim = (a - 0.1(b-a), b + 0.1(b-a))
    end
    if isnothing(ylim)
        c, d = extrema(y)
        ylim = (a - 0.1(d-c), b + 0.1(d-c))
    end

    scatter(x, y; label="", msw=0, ms, alpha)
    plot!(x -> α̂ + β̂*x, xlim...; label="", lw)
    plot!(; xguide, yguide, xlim, ylim)
    plot!(; kwargs...)
end
```

Out[4]: plot_ols (generic function with 1 method)

In [5]:
```
using RDatasets
anscombe = dataset("datasets", "anscombe")
xlim, ylim = (3, 20), (2, 14)
xguide, yguide = "x", "y"
PP = []
for i in 1:4
    x, y = anscombe[!, "X$i"], anscombe[!, "Y$i"]
    P = plot_ols(x, y; xguide, yguide, xlim, ylim, title="Ansbombe $i")
    push!(PP, P)
end
Anscombe_quartet = plot(PP...; size=(500, 450));
```

In [6]:
```
# Rの側にdatasauRusパッケージをインストールしておくこと
datasaurus = rcopy(R"datasauRus::datasaurus_dozen")
datanames = unique(datasaurus.dataset)
xlim = ylim = (-5, 105)
xguide = yguide = ""
PP = []
for name in datanames
    x = datasaurus.x[datasaurus.dataset .== name]
    y = datasaurus.y[datasaurus.dataset .== name]
    P = plot_ols(x, y; xlim, ylim, title=name, ms=3)
    push!(PP, P)
end
layout = @layout[_ a _; b c d; e f g; h i j; k l m]
Datasaurus_dozen = plot(PP...; layout, size=(800, 1000));
```

In [7]:

```julia
function pvalue_clopper_pearson(dist::DiscreteUnivariateDistribution, x)
    min(1, 2cdf(dist, x), 2ccdf(dist, x-1))
end
pvalue_clopper_pearson(n, k, p) = pvalue_clopper_pearson(Binomial(n, p), k)

_pdf_le(x, (dist, y)) =  pdf(dist, x) ≲ y

function _search_boundary(f, x0, Δx, param)
    x = x0
    if f(x, param)
        while f(x - Δx, param) x -= Δx end
    else
        x += Δx
        while !f(x, param) x += Δx end
    end
    x
end

function pvalue_sterne(dist::DiscreteUnivariateDistribution, x)
    Px = pdf(dist, x)
    Px == 0 && return Px
    Px == 1 && return Px
    m = mode(dist)
    Px ≈ pdf(dist, m) && return one(Px)
    if x < m
        y = _search_boundary(_pdf_le, 2m - x, 1, (dist, Px))
        cdf(dist, x) + ccdf(dist, y-1)
    else # x > m
        y = _search_boundary(_pdf_le, 2m - x, -1, (dist, Px))
        cdf(dist, y) + ccdf(dist, x-1)
    end
end

function pvalue_sterne(n, k, p)
    pvalue_sterne(Binomial(n, p), k)
end

function pvalue_wilson(n, k, p)
    p̂ = k/n
    SE = √(p*(1-p)/n)
    2ccdf(Normal(), abs(p̂ - p)/SE)
end

function pvalue_wald(n, k, p)
    p̂ = k/n
    SEhat = √(p̂*(1-p̂)/n)
    2ccdf(Normal(), abs(p̂ - p)/SEhat)
end

function illustrate_binomial_pvalues()
    n = 20
    p = 0.25
    k = 9

    p̂ = k/n
    σ̂² = n*p̂*(1-p̂)

    null = Binomial(n, p)
    μ, σ² = mean(null), var(null)
    nullname = distname(null)

    f(x) = mypdf(null, x)
    xlim = (-2, n+1)
    xlimk = (k-0.5, n+0.5)

    pval = @sprintf "%.03f" pvalue_clopper_pearson(n, k, p)
    cval = @sprintf "%.03f" ccdf(null, k-1)
    P1 = plot(title="(1) Clopper-Pearson")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    annotate!(k+5, 4f(k), text("P-value = $pval", 10, :red))
    plot!([k+7, k+7], [2.4f(k), 3.6f(k)]; arrow=true, c=:red, label="")
    annotate!(k+6, 3f(k), text("2×", 10, :red, :right))
    annotate!(k+5, 2f(k), text("one-tailed = $cval", 10, :red))
    plot!([k+7, k+1], [1.5f(k), 1.5f(k+1)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)
```

```julia

    pval = @sprintf "%.03f" pvalue_sterne(n, k, p)
    j = _search_boundary(_pdf_le, 2mode(null)-k, -1, (null, pdf(null, k)))
    xlimj = (-0.5, j+0.5)
    P2 = plot(title="(2) Sterne")
    plot!(f, xlim...; label="", c=1)
    plot!(f, xlimk...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    plot!(f, xlimj...; label="", fillrange=0, c=1, fc=:red, fa=0.5)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!([k-0.5, j+0.5], [f(k), f(k)]; label="", c=2, ls=:dash)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+1], [2.5f(k), 0.8f(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+1], [2.5f(k), 0.8f(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wilson(n, k, p)
    normal = Normal(μ, √σ²)
    g(x) = pdf(normal, x)
    P3 = plot(title="(3) Wilson")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(g, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ²)", c=2)
    plot!(g, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(g, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    pval = @sprintf "%.03f" pvalue_wald(n, k, p)
    normal = Normal(μ, √σ̂²)
    h(x) = pdf(normal, x)
    P4 = plot(title="(4) Wald")
    plot!(f, xlim...; label="", c=1)
    vline!([μ]; label="μ=$μ", c=1, lw=0.5, ls=:dash)
    plot!(h, xlim...; ls=:dash, label="Normal(μ=$μ, σ=√$σ̂²)", c=2)
    plot!(h, k, n; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    plot!(h, -2, 2μ-k; label="", fillrange=0, c=2, fc=:red, fa=0.5)
    annotate!(k+5, 3f(k), text("P-value = $pval", 10, :red))
    plot!([k+5, k+0.5], [2.5f(k), 0.8g(k)]; arrow=true, c=:red, label="")
    plot!([k+5, j+0.2], [2.5f(k), 0.7g(j)]; arrow=true, c=:red, label="")
    plot!(; xtick=0:n, tickfontsize=7)

    plot(P1, P2, P3, P4; size=(800, 520), layout=(2, 2))
    plot!(; plot_title="model: $nullname,   data: n=$n, k=$k")
end

#illustrate_binomial_pvalues()
```

Out[7]: illustrate_binomial_pvalues (generic function with 1 method)

In [8]:

```julia
oddsratiohat(a, b, c, d) = safediv(a*d, b*c)
stderr_logoddsratiohat(a, b, c, d) = √(1/a + 1/b + 1/c + 1/d)

function pvalue_or_wald(a, b, c, d; ω=1)
    logORhat = log(oddsratiohat(a, b, c, d))
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logORhat - log(ω)), SEhat_logORhat))
end

function confint_or_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    ORhat = oddsratiohat(a, b, c, d)
    SEhat_logORhat = stderr_logoddsratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logORhat), ORhat), safemul(exp(z*SEhat_logORhat), ORhat)]
end
```

Out[8]: confint_or_wald (generic function with 1 method)

In [9]:

```
riskratiohat(a, b, c, d) = safediv(a*(c+d), (a+b)*c)
stderr_logriskratiohat(a, b, c, d) = √(1/a - 1/(a+b) + 1/c - 1/(c+d))

function pvalue_rr_wald(a, b, c, d; ρ=1)
    (a+b==0 || c+d==0) && return 1.0
    logRRhat = log(riskratiohat(a, b, c, d))
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logRRhat - log(ρ)), SEhat_logRRhat))
end

function confint_rr_wald(a, b, c, d; α=0.05)
    (a+b==0 || c+d==0) && return [0, Inf]
    z = quantile(Normal(), 1-α/2)
    RRhat = riskratiohat(a, b, c, d)
    SEhat_logRRhat = stderr_logriskratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logRRhat), RRhat), safemul(exp(z*SEhat_logRRhat), RRhat)]
end
```

Out[9]: confint_rr_wald (generic function with 1 method)

In [10]:

```
function logtick(; xlim=(0.03, 30))
    xmin, xmax = xlim
    a = floor(Int, log10(xmin))
    b = ceil(Int, log10(xmax))
    nums =     [1, 2, 3, 4, 5, 6, 7, 8, 9]
    mask = Bool[1, 1, 0, 0, 1, 0, 0, 0, 0]

    logtick = foldl(vcat, ([10.0^k*x for x in nums if xmin ≤ 10.0^k*x ≤ xmax] for k in a:b))
    logticklabel_a = foldl(vcat,
        ([mask[i] ? string(round(10.0^k*x; digits=-k)) : ""
                for (i, x) in enumerate(nums) if xmin ≤ 10.0^k*x ≤ xmax]
            for k in a:-1))
    logticklabel_b = foldl(vcat,
        ([mask[i] ? string(10^k*x) : ""
                for (i, x) in enumerate(nums) if xmin ≤ 10.0^k*x ≤ xmax]
            for k in 0:b))
    logticklabel = vcat(logticklabel_a, logticklabel_b)
    (logtick, logticklabel)
end

#logtick()
```

Out[10]: logtick (generic function with 1 method)

In [11]:

```
risk_ratio_example1 = plot()

xlim = (0.1, 1.5)

k=250
a, b, c, d = 10000-k, 10000+k, 10000, 10000
RRhat = riskratiohat(a, b, c, d)
pval = pvalue_rr_wald(a, b, c, d)
ci = confint_rr_wald(a, b, c, d)
label = """
---------- A ----------
Data:  $a  $b
         $c $d
RRhat: $(round(RRhat; digits=3))
95% CI: $(round.(ci; digits=3))
RR=1 P-val.: $(round(100pval; digits=2))%
"""
plot!(ρ -> pvalue_rr_wald(a, b, c, d; ρ), xlim...; label)

a, b, c, d = 4, 16, 10, 10
RRhat = riskratiohat(a, b, c, d)
pval = pvalue_rr_wald(a, b, c, d)
ci = confint_rr_wald(a, b, c, d)
label = """
---------- B ----------
Data:  $a  $b
         $c $d
RRhat: $(round(RRhat; digits=3))
95% CI: $(round.(ci; digits=3))
RR=1 P-val.: $(round(100pval; digits=2))%
"""
plot!(ρ -> pvalue_rr_wald(a, b, c, d; ρ), xlim...; label, ls=:dash)

plot!(xscale=:log10, xtick=logtick(; xlim), ytick=0:0.1:1)
#plot!(ytick=0:0.1:1)
vline!([1]; label="", c=:red, lw=0.8)
plot!(xguide="RR (risk ratio)", yguide="P-value")
plot!(size=(800, 300), legend=:outertopright)
plot!(leftmargin=4Plots.mm, bottommargin=4Plots.mm)

risk_ratio_example1;
```

In [12]:

```
risk_ratio_example2 = plot()

xlim = (0.8, 3.5)

k=250
a, b, c, d = 10000+k, 10000-k, 10000, 10000
RRhat = riskratiohat(a, b, c, d)
pval = pvalue_rr_wald(a, b, c, d)
ci = confint_rr_wald(a, b, c, d)
label = """
---------- A ----------
Data: $a  $b
         $c $d
RRhat: $(round(RRhat; digits=3))
95% CI: $(round.(ci; digits=3))
RR=1 P-val.: $(round(100pval; digits=2))%
"""
plot!(ρ -> pvalue_rr_wald(a, b, c, d; ρ), xlim...; label)

a, b, c, d = 16, 4, 10, 10
RRhat = riskratiohat(a, b, c, d)
pval = pvalue_rr_wald(a, b, c, d)
ci = confint_rr_wald(a, b, c, d)
label = """
---------- B ----------
Data: $a  $b
         $c $d
RRhat: $(round(RRhat; digits=3))
95% CI: $(round.(ci; digits=3))
RR=1 P-val.: $(round(100pval; digits=2))%
"""
plot!(ρ -> pvalue_rr_wald(a, b, c, d; ρ), xlim...; label, ls=:dash)

plot!(xscale=:log10, xtick=logtick(; xlim), ytick=0:0.1:1)
#plot!(ytick=0:0.1:1)
vline!([1]; label="", c=:red, lw=0.8)
plot!(xguide="RR (risk ratio)", yguide="P-value")
plot!(size=(800, 300), legend=:outertopright)
plot!(leftmargin=4Plots.mm, bottommargin=4Plots.mm)

risk_ratio_example2;
```

1 データ全体の様子を適切なグラフ作画によって確認せよ！

1.1 Anscombeの例

Anscombeの例 (https://en.wikipedia.org/wiki/Anscombe%27s_quartet)については

- 「標本分布について」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)

の必修問題を参照せよ.

In [13]:
```
Anscombe_quartet
```

Out[13]:

4つのデータの標本平均 $\bar{x}, \bar{y}$ と不偏分散 s_x^2, s_y^2 と不偏共分散 s_{xy} はほぼ一致し, その結果, 回帰直線 $y = \hat{\alpha} + \hat{\beta}x$ もほぼ一致する.

このようなデータの場合にコンピュータで回帰直線だけを計算してしまうと, 不適切な回帰分析をしてしまう可能性がある.

最小二乗法によって回帰直線を求めることが適切なのはAnscombe 1の形のデータだけである.

現実世界で得られるリアルなデータでも, Anscombe 2, 3, 4 のような様子をしている場合がある.

Anscombe 2は直線で近似するには適さない場合である.

何らかの事故によって一度だけ測定ミスが起こるとAnscombe 3のようなデータができあがる.

特別に大きな x の値を持つものが1つだけ含まれているデータはAnscombe 4に似た形になる.

これらのデータは平均, 分散, 共分散のような要約統計量だけを見ても区別をできない.

1.2 問題: Anscombeの例

[「標本分布について」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb) にあるAnscombeの例に関する必修問題をまだ解いていない人は解いてみよ.

1.3 データサウルス

データサウルスの例については

- 「条件付き確率分布, 尤度, 推定, 記述統計」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20

の最後の方の問題で紹介した. データサウルスについては

- https://www.autodesk.com/research/publications/same-stats-different-graphs (https://www.autodesk.com/research/publications/same-stats-different-graphs)

に英語による詳しい説明があり,

- https://visualizing.jp/the-datasaurus-dozen/ (https://visualizing.jp/the-datasaurus-dozen/)

に日本語による簡単な解説がある.

In [14]:
```
Datasaurus_dozen
```

Out[14]:

以上の13個のデータの標本平均 $\bar{x}, \bar{y}$ と不偏分散 s_x^2, s_y^2 と不偏共分散 s_{xy} はほぼ一致し, その結果, 回帰直線 $y = \hat{\alpha} + \hat{\beta}x$ もほぼ一致する.

以上のデータの中に最小二乗法によって回帰直線を求めることが適切な場合はない.

左下の "slant_up" のデータでは, 斜めに並んでいる5つのグループごとに傾きは正になっているが, 回帰直線の傾きは負になっている.

このような場合が生じることを俗に **Simpson's paradox** と言うことがある.

私が読んだ範囲内での **Simpsonのパラドックス** の「正しい理解の仕方」に関する最良の解説は次のブログ記事である:

- KRSK, データ分析の不思議、シンプソンのパラドックスを統計的因果推論から考える (https://www.krsk-phs.com/entry/simpsonparadox), 2018-08-31

1.4 問題: データサウルス

[「条件付き確率分布, 尤度, 推定, 記述統計」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estin](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20estin)
の最後の方にあるDataSaurusの例に関する問題をまだ解いていない人は解いてみよ.

1.5 箱ひげ図は十分に大きなサイズの単峰型のデータにのみ使用すること

以下の図は既出のデータサウルスの解説ページ

- https://www.autodesk.com/research/publications/same-stats-different-graphs (https://www.autodesk.com/research/publications/same-stats-different-graphs)

の最後の方にある動画のスクリーンショットとして取得したものである.

上段がヒストグラムで中断がなまデータの様子で下段が箱ひげ図(box plot)である.

左側がなまデータ(raw data)の様子であり, 真ん中と右側はそれぞれそれに対応する箱ひげ図(box plot)とバイオリンプロット(violin plot)である.

以上の図から, なまデータの様子を見ておくことが必須であることがわかる.

どの場合にも, なまデータの様子が全然違うのに, 箱ひげ図の様子はどれもほぼ同じになってしまっている.

箱ひげ図は十分な大きさのサイズを持つ単峰型のデータの場合にのみ適切なグラフの描き方である.

さらに, 箱ひげ図に限らず, グラフのもとになったデータのサイズを必ず示しておくべきである.

「単峰型」とは「山が1つの」という意味である.

右側のバイオリン・プロットは箱ひげ図よりもずっとましだが, なまデータの様子を誤解なく伝えるだけの力はないように見える.

ヒストグラムはなまデータの様子を伝えるためには箱ひげ図やバイオリンプロットよりも適切な方法に成り得るが, 横軸を区切る幅(ビン, bin)の設定に恣意性があり, ビンの恣意性によって誤解を招くグラフを作ってしまう危険性がある.

データの様子をグラフで確認する場合にも, グラフの描き方によって誤解が生じる場合が出て来てしまうことを知っておく必要がある.

1.6 不適切な箱ひげ図の例

上で紹介した例はサイズは十分に大きいが単峰型でないデータで箱ひげ図を描いた場合(非適切な場合)の例になっている. サイズが小さ過ぎるデータの箱ひげ図を描いた場合(不適切な場合)の例が

- 藤井良宜, 中学校数学におけるこれからの統計教育, 特集統計教育はどう変わるかII, Rimse, 理数教育研究所, 2018年10月20日発行, pp.2-6. https://rimse.or.jp/report/pdf/Rimse23.pdf (https://rimse.or.jp/report/pdf/Rimse23.pdf)

のpp.2-6にある. 以下の図はこの文献からの引用である.

図１　あるクラスの１か月に読んだ本の冊数

これがなまデータである. データサイズが小さくて, しかも 5 に山がもう1つある可能性もあり, 自信を持って単峰型のデータとは言い難い. しかし, 上の文献ではこのデータに箱ひげ図を適用することを教えるべきだとしている.

図２　図１のデータの箱ひげ図

この箱ひげ図による要約によってなまデータの様子を誤解なく伝えられるはずがない.

- この箱ひげ図には重要な情報である標本サイズが記されていない.
- この箱ひげ図を見ても, 7,8,9冊の子が0人であることは分からない.

さらに次のような図も見つかる.

図３　地域別の合計特殊出生率

東北地方, 関東地方, … の各々には数県しか含まれていない. このように, 各々のグループがほんの数個の数値からなるデータをこのように箱ひげ図で「要約」することは不適切な行為である.

この不適切なグラフの描き方を見れば, データサイズの情報が非常に重要なことも分かる.

1.7 箱ひげ図などの正しい使い方

以下の表は

- https://twitter.com/T_Weissgerber/status/1087646465374281728 (https://twitter.com/T_Weissgerber/status/1087646465374281728)

からの引用である.

Figure Types	Example	Type of Variable	What the Plot Shows	Sample Size	Data Distribution	Best Practices
Dot plot		Continuous	Individual data points & mean or median line Other summary statistics (i.e. error bars) can be added for larger samples	Very small OR small; can also be useful with medium samples	Sample size is too small to determine data distribution OR Any data distribution	• Make all data points visible - use symmetric jittering • Many groups: Increase white space between groups, emphasize summary statistics & de-emphasize points • Only add error bars if the sample size is large enough to avoid creating a false sense of certainty • Avoid "histograms with dots"
Dot plot with box plot or violin plot		Continuous	Combination of dot plot & box plot or violin plot (see descriptions above and below)	Medium	Any	• Make all data points visible (symmetric jittering) • Smaller n: Emphasize data points and de-emphasize box plot, delete box plot and show only median line for groups with very small n • Larger n: Emphasize box plot and de-emphasize points
Box plot		Continuous	Horizontal lines on box: 75th, 50th (median) and 25th percentile Whiskers: varies; often most extreme data points that are not outliers Dots above or below whiskers: outliers	Large	Do not use for bimodal data	• List sample size below group name on x-axis • Specify what whiskers represent in legend
Violin plot		Continuous	Gives an estimated outline of the data distribution. The precision of the outline increases with increasing sample size.	Large	Any	• List sample size below group name on x-axis • The violin plot should not include biologically impossible values
Bar graph		Counts or proportions	Bar height shows the value of the count or proportion	Any	Any	• **Do not use for continuous data** Source: @T_Weissgerber

例えば, box plot (箱ひげ図)については,

- サンプルサイズが大きい場合に使用
- 単峰型でないデータには使用しない

ということになっており, ベストプラクティス(最良実践)の項目には

- x軸上のグループ名の下にサンプルのサイズも示しておくこと
- ひげが何を表しているかを凡例で示しておくこと

と書いてある.

1.8 ゴリラ

以下の例はこの一連のノート群の中では初めて紹介するものである. その例は論文

- Itai Yanai & Martin Lercher, A hypothesis is a liability (https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02133-w), Genome Biology volume 21, Article number: 231 (2020)

で使われた擬似データセットのお話である. 以下で使う図はこの論文からの引用になっている.

論文の著者たちは, 学生達に以下のように伝えた:

> データセットには1786人分のボディマス指数(BMI)とある特定の日にそれぞれが歩いた歩数(steps)が記録されており, 男性と女性について2つのファイル分けてデータが格納されている.

学生に配布したデータセットの様子は以下の通り:

(https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02133-w/figures/1)

そして, 学生を次の2つのグループに分けた.

1つ目の「仮説あり」の(hypthothesis-focusedの)グループの学生には, 以下の3つの仮説を検討するように求めた.

(i) 男性と女性の平均歩数に統計的に有意な差があること
(ii) 女性では歩数とBMIの間には負の相関があること
(iii) 男性では歩数とBMIの間には正の相関があること

さらに, データセットから結論できることが他にあるかどうかも質問した.

2つ目の「仮説なし」の(hypothesis-freeの)グループの学生には, 単に「データセットから何を結論できますか?」と質問した.

その結果は以下の通り.

	Gorilla not discovered	Gorilla discovered
Hypothesis-focused	14	5
Hypothesis-free	5	9

(https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02133-w/figures/1)

この表の日本語訳:

	ゴリラを発見できなかった	ゴリラを発見した
仮説ありグループ	14	5
仮説なしグループ	5	9

「ゴリラを発見できなかった」と「ゴリラを発見した」とは一体どういう意味なのだろうか?

その理由はデータの散布図を描くとわかる!

(https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-02133-w/figures/1)

学生に課された統計学のレポート課題のデータセットは実は偽物のデータセットであり, 散布図を描くとゴリラの絵が現れるように作られていたのである!

レポートを提出した学生は2つのグループを合わせて33人で, そのうちの過半数の19人がゴリラを発見できなかった.

さらに, (i),(ii),(iii)の3つの仮説を検討するように指示を出された「仮説あり」のグループでは19人中14人もゴリラを発見できなかった! 3つの仮説の検討の指示によって誤誘導されてしまった格好になってしまっている.

このようなレポート課題を出すことは統計学の講義として教育的であり, 学生の側も良い勉強になったと思われる.

1.9 問題: ゴリラ

「検定と信頼区間: 比率の比較」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20interval%%20Two%20proportions.ipynb) の必修計算問題の節の解答例を参照して, 2×2の分割表

	ゴリラを発見できず	ゴリラを発見した
仮説ありグループ	14	5
仮説なしグループ	5	9

のデータから「仮説あり」と「仮説なし」の2つのグループでゴリラを発見できない確率(比率)の違いに関ついて以下を求めよ:

- 「オッズ比パラメータは $\mathrm{OR} = 1$ である」という仮説のWald版P値 `P_Wald` .
- オッズ比パラメータ OR に関するWald版 95% 信頼区間 `CI_Wald` .

上の問題の続き. 可能ならば以下も求めよ.

- 「オッズ比パラメータは $\mathrm{OR} = 1$ である」という仮説のPearsonのχ²検定版P値 `P_Chisq` .
- オッズ比パラメータ OR に関するPearsonのχ²検定版の 95% 信頼区間 `CI_Chisq` .
- 「オッズ比パラメータは $\mathrm{OR} = 1$ である」という仮説のSterne版Fisher検定のP値 `P_Sterne` .
- オッズ比パラメータ OR に関するSterne版Fisher検定の 95% 信頼区間 `CI_Sterne` .
- 「オッズ比パラメータは $\mathrm{OR} = 1$ である」という仮説のClopper-Pearson版Fisher検定のP値 `P_CP` .
- オッズ比パラメータ OR に関するClopper-Pearson版Fisher検定の 95% 信頼区間 `CI_CP` .

可能ならばそれぞれの場合のP値函数のグラフをプロットしてみよ.

無理して3から8の小問を解いたり, P値函数のグラフを描いたりする必要はないが, もしもそれらの答えを数値的に求めたり, グラフを描く方法をマスターできたりしたら, 後で役に立つ可能性がある.

これらの問題の解答例はここには書かない.

2 印象操作のためにグラフを利用しない

統計学云々以前の問題として, 印象操作のためにグラフを悪用することはやめた方がよい.

多くの場合に誤った印象を与えることになることで有名なグラフの描き方に3次元円グラフがある. (一般に円グラフはよくないグラフの描き方だとされている.)

2.1 3次元円グラフは絶対に避ける

3次元グラフがダメな理由および円グラフ一般が良いグラフの描き方ではないことについては

- 3D円グラフ撲滅委員会より、なぜ3D円グラフを使ってはいけないか (https://www.cresco.co.jp/blog/entry/950/)

が見易く読み易い. 以下の図はこのページからの引用である. ただし, 矢線は筆者が追加した.

問題: 隣り合っている20代と30代ではどちらの割合が大きいか?

見た目的には圧倒的に30代の割合の方が大きく見えるが, 実際には20代が20%で30代は18%であり, 30代の割合の方が少し小さい. 手前の40代の方が奥側の10代や70代よりも割合が大きく見えるが, 実際にはそうではない.

このように3次元円グラフは割合について間違った印象を与えるために利用できる!

3 無作為抽出の失敗に注意を払う

「独立同分布」という条件を満たす確率分布を統計モデルとして採用した場合には, 調べたい母集団からの適切な無作為抽出でデータを得ることが適切な統計分析のために必要になる.

しかし, 実際にはそれはそう簡単ではない.

調べたい母集団よりも狭い集団からの偏ったサンプルの抽出になってしまっている危険がいつでも生じる.

そのような例として有名な事例を以下で紹介しよう.

3.1 1936年のアメリカ大統領選挙の結果の予測の失敗

1936年のアメリカ大統領選挙は民主党のルーズベルトと共和党のランドンの対決になった.

以下の得票率の予測の表は, 鈴木督久『世論調査の真実』(日経BP 日本経済新聞出版本部, 2021年)の図表5-1の孫引きであり, そこには

> (出所) 小山栄三(1946)『輿論調査概要』時事通信社(名称を一部省略)

と書いてある.

	ルーズベルト	ランドン
実際の得票率 (%)	60.2	39.8
リテラリー・ダイジェスト	42.6	57.4
ギャラップ	54	46
フォーチュン	74	26
クロスレー	52	48
バルチモア・サン	64	36
ファーム・ジャーナル	43	57
グラスルーツ	39.5	60.5

ここで, リテラリー・ダイジェスト, ギャラップ, ... は予測を出した主体の名前(会社や雑誌の名前など)である.

例えば, リテラリー・ダイジェスト誌は, ルーズベルト候補は 42.6%, ランドン候補は 57.4% の票を得て, ランドン候補が勝利するだろうと予想した. しかし, 実際にはルーズベルト候補が 60.2% の票を得て勝利することになった.

リテラリー・ダイジェスト誌は, 1936年8月22日号で調査開始を宣言し, 9回にわたる連載で最終的に237万6523票分のアンケート結果を集計して, 上の予測を立てた.

母集団中の6割がルーズベルト支持で残りの4割がランドン支持のとき, 200万人を理想的に無作為抽出したならば, その中のルーズベルト支持者の割合が 43% 以下(86万人以下)になってしまう確率はほぼ 0 である:

$$\mathrm{cdf}(\mathrm{Binomial}(2000000, 0.6), 860000) \approx 2 \times 10^{-50925}.$$

左辺は, ルーズベルト氏の支持者が6割いるときに, 理想的に200万人無作為抽出した中にルーズベルト支持者が86万人以下しか含まれていない確率を意味する.

このことから, リテラリー・ダイジェスト誌によるアンケート調査には大きな偏りがあったと考えられる.

リテラリー・ダイジェスト誌が調査対象としていた人達は富裕層が多く, 偏っていたことが問題だった.

この事例から, たとえサイズ200万の **ビッグデータ** であっても, その内訳が偏っているならば, そのデータを使った予測は大外れしてしまうことがわかる.

In [15]:
```
setprecision(50) do
    @time sum(pdf(Binomial(2000000, big"0.6"), k) for k in 0:860000)
end
```

```
37.820768 seconds (106.64 M allocations: 3.172 GiB, 4.70% gc time, 0.01% compilation time)
```

Out[15]:
```
2.0771983651745741e-50925
```

3.2 1948年のアメリカ大統領選挙の結果の予測の失敗

以下の表は

- [なるほど統計学園 アメリカ大統領選挙の番狂わせ（後編）標本調査における偏り2 (https://www.stat.go.jp/naruhodo/15_episode/episode/senkyo2.html)](https://www.stat.go.jp/naruhodo/15_episode/episode/senkyo2.html)

にあった表のフォーマットを変えたものである. そこには

> 候補者の当選予想と得票率のグラフ

> （注）上記2人の他にも候補者がいたため、合計は100%にならない。

と書いてある.

	民主党 トルーマン	共和党 デューイ
実際の得票率 (%)	○49.5	45.1
ギャラップの予想	44.5	○49.5
ローバーの予想	37.1	○52.2
クロスレーの予想	44.8	○49.9

歴史的には1936年の大統領選挙でルーズベルトが当選することを正しく予測できたギャラップは1948年の大統領選挙では予測に失敗することになった.

ギャラップによる調査方法は無作為抽出法ではなく, 割り当て法であった. 詳しくは上のリンク先を参照して欲しい.

このように, アンケート調査の結果は偏っていることがあり, 信頼できる予測を統計学的な方法で行うためにはデータの取得時に特別な注意が必要になる.

3.3 選挙の予測に関する参考文献

以上については以下の文献を参考にした.

- なるほど統計学園
  - アメリカ大統領選挙の番狂わせ（前編）標本調査における偏り1 (https://www.stat.go.jp/naruhodo/15_episode/episode/senkyo1.html)
  - アメリカ大統領選挙の番狂わせ（後編）標本調査における偏り2 (https://www.stat.go.jp/naruhodo/15_episode/episode/senkyo2.html)
- 鈴木督久, 『世論調査の真実』, 日経BP 日本経済新聞出版本部, 2021年

歴史的には1936年の米大統領選挙の予測についてはギャラップ氏達の予測が有名である. その物語に興味がある人はこれらの文献を参照して欲しい.

4 2つのグループを比較するときの注意

4.1 データは偏っていることが多い

我々は別のノートで「2つのグループの比率の違いは○○である」の型の仮説や「2つのグループの平均の違いは○○である」の型の仮説に関するP値や信頼区間の計算の仕方について学んだ.

- 検定と信頼区間: 比率の比較 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)
- 検定と信頼区間: 平均の比較 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb)

例えば, 風邪をひいた人に薬Xを与えることに効果があるかどうかを知るために, 薬Xを与えたグループと薬を与えずに経過観察したグループに分けて取得した「5日以内に回復した人数」のデータを利用する場合には, 薬Xを与えたかどうか以外の回復日数に影響を与える条件が2つのグループで揃っていることが必要である.

風邪症状が重い人達には薬Xを与える傾向が強く, 症状が軽い人達は経過観察する傾向が強い場合には, 薬Xに本当は効き目があったとしても, 重い症状の人達に特に薬Xを与えてしまったせいで, 経過観察グループとの比較で, 回復にかかる日数が平均的に増えてしまうかもしれない.

統計学の現実での応用時にはこのような問題が発生することの方が普通なので注意が必要である.

例えば, ○○ウイルス対策としてワクチンを接種したグループと接種しなかったグループを比較するために, ○○ウイルスによって重症になった人数を調べたとしよう.

このときもしもワクチンを重症になりやすい人達に優先的に接種しているならば, ワクチンに効き目があったとしても, ワクチン接種群の方が非接種群よりも重傷者の割合が多くなってしまうこともあり得る.

このような問題は, ○○ウイルスによって重症になりやすいか否かを完全に無視してワクチン接種するようにすれば解消されるのだが, 人の命を守るために行われるべきワクチン接種ではそういうことは倫理的にするべきではない.

他にも様々な要因でデータが偏ってしまうことがありえる.

特別な場合を除けば, データは必ず偏っており, 問題はその程度がどのくらい重大であるかということになる.

4.2 2つのグループを比較するときの注意に関する参考資料

次の動画が参考になるだろう.

- 佐藤俊哉, ランダム化ができないとき (https://youtu.be/tUkyjZXU1vc), 2018年10月25日

同じ著者による次の本も参考になる.

- 佐藤俊哉, 宇宙怪人しまりす　医療統計を学ぶ　検定の巻, 岩波科学ライブラリー 194, 2012年

データの偏りについて語るときの統計学的なキーワードの一つは **交絡** (confounding)である. 「交絡」の正確な定義を知りたい人は次の講義動画の18:00あたりをみて欲しい.

- 佐藤俊哉, 交絡とその調整 (https://youtu.be/ybdkQFEdCPM), 2018年12月20日

5 P値や信頼区間の誤用

5.1 P値の定義

データの数値 x とデータの数値の生成のされ方に関するパラメータ θ を持つ統計モデル $M(\theta)$ と具体的なパラメータ値 θ_0 が与えられたとき, 仮説 $\theta = \theta_0$ のP値はその仮説下での統計モデル $M(\theta = \theta_0)$ 内でデータの数値 x 以上に極端な値が生成される確率またはその近似値として定義される.

「〜以上に極端な」や「近似値」の意味を適切に与えるごとにP値の正確な定義が得られる.

例えば, データの数値が「n 回中 k 回成功」の形をしていて, 統計モデル $M(p)$ が試行回数 n, 成功確率パラメータ p の二項分布モデルのとき, $p = p_0$ という仮説のClopper-Pearson型のP値は, モデル $M(p = p_0)$ 内でデータの数値 k 以下の値が生成される確率と k 以上の値が生成される確率の小さい方の2倍として定義される. 実際には, P値の値が 1 を超えないように, その値と 1 の小さな方として, Clopper-Pearson型のP値が定義される. 具体的に式で書くと以下の通り:

$$\mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0) = \min \begin{pmatrix} 1 \\ 2\,\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k) \\ 2(1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k - 1)) \end{pmatrix}.$$

「〜以上に極端な」の大雑把な意味は「モデル内確率分布の平均値 np_0 からの距離がデータの数値 k 以上に大きい」という意味に設定しているが, モデルの分布が左右非対称になっていることを考慮してこのような定義になっている. この他にも

- 「検定と信頼区間: 比率の検定と信頼区間」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)

では, Sterne型のP値と

- 「大数の法則と中心極限定理」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)

で説明した二項分布の中心極限定理(正規分布近似)を使って定義されたP値を2種類定義した(Wilson型, Wald型).

二項分布モデルの場合のそれら4種のP値の定義で使われた考え方は他の統計モデルでのP値を定義するときにも使われる. その意味で非常に基本的である.

In [16]:
```
illustrate_binomial_pvalues()
```

Out[16]:

5.2 P値の正しい使い方

仮説 $\theta = \theta_0$ のP値は, その仮説下での統計モデル $M(\theta = \theta_0)$ とデータの数値 x の整合性の指標として使われる.

P値が非常に小さい場合には整合性がほとんどないと考える.

P値が 0 より十分に大きな場合には, モデル $M(p)$ の下での仮説 $p = p_0$ のデータの数値 x との整合性がある程度あると考える. しかし, 整合性がある程度あることと, そのパラメータ値が真実を表しているかもしれないことは別の話になるので注意すること.

P値だけから得られる結論は弱い という点を強調したいがために, 「P値は整合性の指標である」と述べていることにも注意せよ!

例えば, 統計モデルとデータの数値のどちらかに問題があるせいで整合性があるように見えてしまったり, 整合性がないように見えてしまったりすることがあり得る.

P値を使った統計的推論は, 使用した統計モデルが妥当でかつ, 用いたデータの数値も信頼できる場合にのみ, 真っ当な結論を出し得る.

実際には, データが偏っている可能性は常に心配する必要があるし, 使用した統計モデルがそのようなデータの偏りを考慮していない不適切なモデルになっている可能性も常に心配する必要がある. モデルが妥当でなくなるパターンは多彩である.

注意: 使用している統計モデルが妥当であることの意味は, 目的ごとにユーザー側が決める必要があり, さらに妥当であることの証拠の提出もユーザー側の責任になる. 統計学入門の解説に書いてある統計分析の手続きに機械的に従えば適切な統計的推論が可能になるという考え方は完全に誤りである. 使用した統計モデルの妥当性やデータの数値の信憑性のような重要な事柄については統計学のユーザー側が責任を持つ必要がある.

注意: P値 (P-value)はデータの数値とモデル+パラメータ値の **整合性の指標の1つに過ぎない**. 別の考え方で整合性を測ることもあり得る. 例えば, **尤度** (likelihood)やその拡張版としてのベイズ統計における **事後分布** (posterior)の値もデータの数値とモデル+パラメータ値の整合性の指標とみなされる. どの道具を使うにしても, 安易に強い結論を出すことを控えて, 検挙になることが重要である.

5.3 復習: 累積分布函数 cdf と分位点函数 quantile の定義

X が確率分布 D に従う確率変数であるとき, $X \leq x$ となる確率 $P(X \leq x)$ を x の函数とみなしたものを **累積分布函数** (cumulative distribution function, cdf)と呼び,

$$\mathrm{cdf}(D, x) = P(X \leq x)$$

と書くことにしていたのであった. 累積分布函数 $p = F(x) = \mathrm{cdf}(D, x)$ の逆函数 $x = Q(p) = F^{-1}(p)$ (逆函数が存在しない場合には適当に代替物を定義する)を **分位点函数** (quantile function)と呼び,

$$x = \mathrm{quantile}(D, p)$$

のように書くことにしていたのであった. よく使われる基本的な確率分布について, 累積分布函数 cdf と分位点函数 quantile はコンピュータ上での各種の環境で利用可能な場合が多いので, 便利に利用するとよい.

例えば,

$$\begin{aligned}
\mathrm{cdf}(\mathrm{Binomial}(n, p), k) &= \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}, \\
\mathrm{cdf}(\mathrm{Beta}(a, b), k) &= \frac{\int_0^p t^{a-1}(1-t)^{b-1}\, dt}{B(a, b)}, \\
\mathrm{cdf}(\mathrm{Normal}(0, 1), z) &= \int_{-\infty}^{z} \frac{e^{-x^2/2}}{\sqrt{2\pi}}\, dx.
\end{aligned}$$

In [17]:
```
@show cdf(Binomial(100, 0.5), 40)
@show 1 - cdf(Beta(40+1, 100-40), 0.5)
@show cdf(Normal(0, 1), (40 - 100*0.5)/√(100*0.5*0.5));
```

```
cdf(Binomial(100, 0.5), 40) = 0.02844396682049039
1 - cdf(Beta(40 + 1, 100 - 40), 0.5) = 0.02844396682049044
cdf(Normal(0, 1), (40 - 100 * 0.5) / √(100 * 0.5 * 0.5)) = 0.022750131948179205
```

In [18]:
```
@show quantile(Beta(40, 100-40+1), 0.025)
@show quantile(Beta(40+1, 100-40), 0.975)
@show quantile(Normal(0, 1), 0.975);
```

```
quantile(Beta(40, (100 - 40) + 1), 0.025) = 0.3032947687028774
quantile(Beta(40 + 1, 100 - 40), 0.975) = 0.5027908495776652
quantile(Normal(0, 1), 0.975) = 1.9599639845400576
```

5.4 信頼区間の定義

データの数値 x とデータの数値の生成のされ方に関するパラメータ θ を持つ統計モデル $M(\theta)$ についてP値 $\mathrm{pvalue}(x|\theta = \theta_0)$ が定義されているとき, 有意水準 $0 < \alpha < 1$ について, パラメータ θ に関する信頼度 $1 - \alpha$ の信頼区間は, $\mathrm{pvalue}(x|\theta = \theta_0) \geq \alpha$ となるパラメータ値 θ_0 全体の集合として定義される:

$$\mathrm{confint}(x|\alpha) = \{\, \theta_0 \mid \mathrm{pvalue}(x|\theta = \theta_0) \geq \alpha \,\}.$$

例えば, 二項分布モデルのClopper-Pearson型のP値の定義から得られる信頼区間はClopper-Pearsonの信頼区間と呼ばれ, 具体的には,

$$\begin{aligned}
1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_L), k-1) &= \alpha/2, \\
\mathrm{cdf}(\mathrm{Binomial}(n, p_U), k) &= \alpha/2.
\end{aligned}$$

によって p_L, p_U を定義することによって得られる区間

$$[p_L, p_U] = \{\, p_0 \mid p_L \le p_0 \le p_U \,\}$$

になる. どうやって p_L, p_U を求めるかが問題になるが, この場合には, 二項分布とベータ分布の累積分布函数のあいだの素晴らしい関係

$$\begin{aligned} 1 - \mathrm{cdf}(\mathrm{Binomial}(n,p), k-1) &= \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p), \\ \mathrm{cdf}(\mathrm{Binomial}(n,p), k) &= 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p) \end{aligned}$$

を使えば, ベータ分布の分位点函数(quantile function, 累積分布函数 cdf の逆函数)を使って,

$$\begin{aligned} p_L &= \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2), \\ p_U &= \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2). \end{aligned}$$

によって, p_L, p_U を効率的に求められるのであった.

Sterne型のP値に対応する信頼区間の計算はこのようにきれいにはまとまらない. 正規分布近似を使って定義されたP値から得られる信頼区間の計算は標準正規分布の分位点函数の $1-\alpha/2$ での値

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2)$$

の計算に帰着することになる. 二項分布モデルの場合の信頼区間に関する非常に詳しい説明が

- [「検定と信頼区間: 比率の検定と信頼区間」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/09%20Hypothesis%20testing%20and%20confidence%20inter%20Proportion.ipynb)

にある.

二項分布モデルの場合の信頼区間の構成法のパターンは他の統計モデルにおける信頼区間の計算でも同様に使われる. 標準正規分布が t 分布で置き換えられる場合もある.

5.5 信頼区間の正しい解釈の仕方

P値は **統計モデル内部** の確率分布におけるデータの数値以上に極端な値が生じる確率またはその近似値であった.

P値は現実における何らかの意味での確率ではなく, 数学的フィクションであるモデル内部における確率またはその近似値であるにすぎない.

ゆえに, 信頼度 $1-\alpha$ の信頼区間すなわち $100(1-\alpha)\%$ の信頼区間における $1-\alpha = 100(1-\alpha)\%$ は **モデル内部** における確率またはその近似値になる.

すなわち, X をモデル $M(\theta=\theta_0)$ に従う確率変数とするとき, $\mathrm{pvalue}(X|\theta=\theta_0) \ge \alpha$ となる確率は $1-\alpha$ またはその近似値になる.

これは, 信頼度 $1-\alpha$ の信頼区間 $\mathrm{confint}(X|\alpha) = \{\, \theta_0 \mid \mathrm{pvalue}(X|\theta=\theta_0) \ge \alpha \,\}$ に θ_0 が含まれる確率が $1-\alpha$ またはその近似値になることを意味している.

そのとき, 確率変数 X の値は確率的にランダムに決まると考えるので, その函数である信頼区間 $\mathrm{confint}(X|\alpha)$ も確率的にランダムに決まる. すなわち区間の両端の点がランダムに決まる. そのような状況でその区間に定数 θ_0 が含まれる確率が $1-\alpha$ またはその近似値になる. しかもその確率は数学的フィクションであるモデル内部における確率に過ぎない.

現実の未知の母集団から標本を何度も取り直すことによって 95% 信頼区間の 95% の意味を説明しようとしている解説をよくみかけるが, ミスリーディング(misleading, 誤誘導的)なので注意が必要である.

この節における以上の部分もミスリーディングであるという考え方もできる.

この節での以上の解説では 95% 信頼区間の 95% をどのような確率(またはその近似値)とみなすことができるかを説明した.

確かにそういう知識を持っていた方が良いのだが, **信頼区間については「P値はデータの数値とモデル+パラメータ値の整合性の指標である」というP値の解釈の基本に戻った方が分かり易い.**

P値から真っ当な結論を出すためには, データの数値が信頼できてかつ, 使用した統計モデルも何らかの意味で妥当でなければいけないのであった. 以下ではこの条件を仮定する.

このとき, パラメータ θ に関する $100(1-\alpha)\%$ 信頼区間は, 閾値 α によってデータと整合性がないと判断されずにすむパラメータ値 θ_0 全体の集合になる.

P値から導かれる結論は弱いので, **信頼区間に含まれるパラメータ値についてはその現実における適切さについて判断保留することが正しい.**

例えば, パラメータ θ がウェブページのデザインAのデザインBに対する優位性の指標であるときに, パラメータ θ に関する信頼区間については以下のように考える.

(1) 信頼区間がゼロを含むならばデザインAとデザインBは同等の働きをしている可能性を排除できないと考える.

(2) 信頼区間がたとえ正の値を含んでいても, 単にデザインAの方がデザインBよりも優れている可能性を排除できないと考えるだけにとどめ, デザインAの方がデザインBよりも優れているとは安易に結論を出さない.

(3) 信頼区間の左端の値が十分にゼロよりも大きいならば, デザインAの方が優れていると判断できる. ただし, 上で「以下ではこの条件を仮定する」と述べていたことが重要であることに注意せよ! データが信用できなかったり, モデルが妥当でなかったりする場合にはこのような判断はひどい失敗を招く可能性がある.

5.6 P値の誤用の例

仮想的に, 薬Aを与えた m 人と与えなかった n 人について〇〇日以内に回復した人の人数がそれぞれ a 人と c 人というデータが得られたとする.

その状況を2つの二項分布モデルを用いて分析するとする. 薬Aを与えたときに〇〇日以内に回復する確率はパラメータ p でモデル化されており, 与えなかった場合の回復確率はパラメータ q でモデル化されているとする.

そのモデルを使って仮説 $p = q$ のP値が計算されたとする. (P値の構成法は複数あるがどれか1つを選んでいると仮定する.)

5.6.1 P値は仮説が正しい確率ではない

そのP値は「薬Aを与えた場合と与えない場合で〇〇日以内の回復確率は等しい」という仮説が成立する確率ではない.

仮説 $p = q$ のP値の大雑把な定義は「データの数値 (m, a, n, c) 以上に $p = q$ に見える状況から離れた値が $p = q$ という仮説下での2つの二項分布モデル内で生じる確率」である.

P値は仮説が正しい確率では全然ないし, P値はモデル内確率に過ぎない.

注意: 「ベイズ統計では仮説が正しい確率を扱える」と言っている人達がいるが, ベイズ統計における「仮説が正しい確率」はモデル内確率分布である事後分布で測った確率であるに過ぎない. ベイズ統計においても使用した統計モデルが不適切ならば事後分布で測った確率も信用できないものになる. P値を使った場合に生じる困難がベイズ統計を使えば解消されるかのような主張は誤りである.

5.6.2 単独のP値のみを使って重要な決定をしてはいけない

事前に有意水準と呼ばれる整合性に関する閾値 α を決めてあったと仮定する(よく使われているのは $\alpha = 5\%$).

さらに, データ中の回復者の割合を比較すると薬Aを与えた側の a/m の方が与えなかった側の c/n より大きい状況を仮定する.

(1) 帰無仮説 $p = q$ のP値が有意水準 α 未満になったとする.

そのときP値が α 未満になっただけで, 「薬Aには効き目がある」と判断してはいけない.

P値以外の他の条件も確認する必要がある.

例えば, データから予想される薬Aの効き目は実生活においては無視できるほど些細な大きさなのに, P値が α 未満になる場合があり得る.

その他にも注意するべきことが沢山ある.

- データの数値が偏っている可能性も心配する必要がある.
- 使用した統計モデルが不適切な可能性にも配慮する必要がある.

統計学を使った分析は「すっきりしない」結果に終わることが多い.

分かり易さを求めすぎないように注意が必要である.

特に統計学を「科学的な御墨付きを得るための道具」扱いしてはいけない.

(2) 帰無仮説 $p = q$ のP値が有意水準 α 以上になったとする.

そのときに「統計的に有意ではない」という理由で, 「ゆえに薬Aには効き目がない」と判断してはいけない.

なぜならば, P値が有意水準 α 以上であることは, 仮説 $p = q$ の下での統計モデルとデータの数値の整合性が閾値 α によって無さすぎると判定されないことを意味するに過ぎないからである.

「ゆえに薬Aには効き目がない」と結論を出すことは誤りで, 「薬Aに効き目があるかどうかを閾値 α によって判定できなかった」と **判断を保留** しなければいけない.

ある研究においてP値が有意水準 α 以上になったとしても, その後の研究によって「薬Aには十分な大きさの効果がある」という結論が出る可能性があり得る.

In [19]:
```
risk_ratio_example1
```

Out[19]:

上のグラフは仮想的なデータに関するP値函数のプロットであり, 薬Aと薬Bの効き目を推定しようとしている状況を想定している.

「薬Aを処方してもリスクは変わらない」という仮説のP値は1.24%で, 有意水準5%で棄却される. しかし, リスクはほんの少ししか減らないように見える.

「薬Bを処方してもリスクは変わらない」という仮説のP値は6.69%で, 有意水準5%で棄却されない. しかし, リスクは大きく減る可能性がある. 薬Bについては, 効果の大きさを期待しつつ判断を保留する必要がある.

このようなときに, 「リスクは変わらない」という仮説のP値が5%を切ったかどうかだけで, 薬Aには価値があり, 薬Bには価値がないと判断することは誤りである.

In [20]:
```
risk_ratio_example2
```

Out[20]:

上のグラフも仮想的なデータに関するP値函数のプロットであり, 物質Aと物質Bにさらされたときのリスクの大きさを推定している状況を想定している.

「物質Aにさらされてもリスクは変わらない」という仮説のP値は1.24%で, 有意水準5%で棄却される. しかし, 物質Aにさらされることによってリスクは少ししか増えないと考えられる.

「物質Bにさらされてもリスクは変わらない」という仮説のP値は6.01%で, 有意水準5%で棄却されない. しかし, 物質Bにさらされることによってリスクが大きく増える可能性があるかもしれない. 物質Bについては, さらされることのリスクの大きさに注意しながら, 判断を保留し, 必要ならば研究を継続する必要がある.

このようなときに, 「さらされてもリスクは変わらない」という仮説のP値だけを見て, 物質Aにさらされることによって増えるリスクは気にする必要はあるが, 物質Bについては気にする必要はないと考えることは明らかに間違っている.

5.6.3 すべてを正直に報告しなければいけない

実際には薬Aだけではなく, 薬B,C,D,…と多数の薬を試してみた場合には, その中でP値が $\alpha = 5\%$ 未満になった薬Aのみについて結果を報告するというようなことをしてはいけない.

薬A,B,C,...のすべてに効き目がゼロ(害もゼロ)であるとき, データから計算したP値が $\alpha = 5\%$ 未満になるものはその中に 5% 程度ある. 有意水準 $\alpha = 5\%$ という閾値はかなり緩く, 色々試してみれば容易にP値を $\alpha = 5\%$ 未満にできる.

都合の良い結果だけを選択的に報告してしまうと, その研究結果がどれだけどのように信頼できるかが分からなくなってしまう.

自分が所属する研究室のボスが部下に対して, 「統計的に有意になった結果のみを報告すればよい」と指示している場合には, 研究不正を意識的に指示している疑いがあるので, 適切な窓口に相談した方が良いと思われる.

研究室のボスによる不適切な指示に黙って従い続けると, 後に研究不正の協力者とみなされ, 大変なことになるかもしれない.

5.7 P値や信頼区間の誤用に関する参考資料

次の文献は必読である:

- 統計的有意性と P 値に関する ASA 声明 (https://www.biometrics.gr.jp/news/all/ASA.pdf), (翻訳: 佐藤俊哉, 2017年4月23日, 英語の原著は2016年)

ただし, この声明における

> 2. P 値とは?
> 　おおざっぱにいうと、P値とは特定の統計モデルのもとで、データの統計的要約（たとえば、2グループ比較での標本平均の差）が観察された値と等しいか、それよりも極端な値をとる確率である。

という説明の中の「データ」は「統計モデル内部で生成された仮想的なデータ」であることに注意する必要がある. 次のように言い直した方が誤解が少ないように思われる.

- おおざっぱにいうと, P値とは特定の統計モデル内で生成された仮想的データの統計的要約(たとえば, 2グループでの比較での標本平均の差)が現実で観察されたデータの値以上に極端な値になる確率もしくはその近似値である.

P値に関するASA声明の翻訳者によるP値に関する解説動画が次の場所にある.

- 佐藤俊哉, 仮説検定とP値の誤解 (https://youtu.be/vz9cZnB1d1c), 2017年10月19日

分かり難いとよく言われている「信頼区間」については, 検定(もしくはP値函数)と信頼区間の表裏一体性に基いて信頼区間について理解しようとすることが正しい理解の方針で思われる. 検定と信頼区間の表裏一体性については, すぐ上の動画の48:37以降で解説されている.

可能ならば以下の論文も参照せよ:

- Valentin Amrhein, Sander Greenland. Discuss practical importance of results based on interval estimates and p-value functions, not only on point estimates and null p-values. First Published June 3, 2022. [doi (https://doi.org/10.1177%2F02683962221105904)]

この一連のノートの内容はこの論文が出る前に計画されていたのであるが, P値の使い方については内容的に非常に近いものになっている.

In [ ]:
```
1
```

回帰 (regression)

- 黒木玄

- 2022-07-13〜2022-07-18

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 回帰 (regression)
  - 1.1 回帰の超一般論
  - 1.2 データの数値から値が決まるパラメータ x がない場合
  - 1.3 回帰の例
- 2 線形回帰
  - 2.1 線形回帰のデータの形
  - 2.2 線形回帰モデルの構成要素
  - 2.3 デザイン行列 (計画行列, design matrix)
  - 2.4 線形回帰の統計モデルの正規分布による記述
  - 2.5 正規分布で書かれた統計モデルの最尤法から最小二乗法による線形回帰が得られること
  - 2.6 直交射影の公式
  - 2.7 βとσ²の不偏推定量
  - 2.8 例: 平均の推定の場合
  - 2.9 例: 単回帰の場合
  - 2.10 Julia言語による回帰直線の計算の最も簡単な例
  - 2.11 多変量正規分布の定義
  - 2.12 問題: 多変量正規分布とχ²分布の関係
  - 2.13 真の回帰函数と推定された回帰函数
  - 2.14 (真の)回帰直線の値の信頼区間 (標準正規分布版)
  - 2.15 (真の)回帰直線の値の信頼区間 (t分布版)
  - 2.16 予測区間
  - 2.17 「回帰函数の値の信頼区間」と「予測区間」の違い
- 3 線形回帰の計算例
  - 3.1 信頼区間と予測区間のプロット
    - 3.1.1 信頼区間と予測区間のテストプロット
    - 3.1.2 回帰直線の信頼区間と予測区間
    - 3.1.3 多項式回帰の信頼区間と予測区間 (オーバーフィッティングの例)
  - 3.2 信頼区間と予測区間に対応するP値函数のプロット
    - 3.2.1 信頼区間と予測区間に対応するP値函数のテストプロット
    - 3.2.2 回帰直線の信頼区間と予測区間に対応するP値函数
    - 3.2.3 回帰直線の信頼区間に対応するP値函数の動画
    - 3.2.4 多項式回帰の信頼区間と予測区間に対応するP値函数
- 4 ロジスティック回帰
  - 4.1 ロジスティック函数とロジット函数
  - 4.2 ロジスティック回帰のデータの形
  - 4.3 ロジスティック回帰でのリンク函数
  - 4.4 ロジスティック回帰の統計モデル
  - 4.5 最尤法
  - 4.6 スコア統計量とFisher情報量行列
  - 4.7 問題: 一般の場合のスコア統計量とFisher情報量行列
  - 4.8 βの最尤推定量の分布の正規分布近似
  - 4.9 ロジスティック回帰における $\beta_0+\beta_1 x$ に関するWald型のP値函数と信頼区間

4.10 ロジスティック回帰における β_1 に関するWald型のP値函数と信頼区間
4.11 ロジスティック回帰における β_1 に関するWald型のP値函数と信頼区間の動画
5 x_i 達の値も1または0の場合のロジスティック回帰
5.1 x_i 達の値も1または0の場合にロジスティック回帰モデルは2つの二項分布モデルに等しい
5.2 x_i 達の値も1または0の場合のスコア統計量とFisher情報行行列
5.3 x_i 達の値も1または0の場合のWald型のP値函数と信頼区間
5.4 x_i 達の値も1または0の場合のWilson型のP値函数と信頼区間
5.4.1 A=0で定まる条件付き確率分布の正規分布近似
5.4.2 与えられた対数オッズ比パラメータの値 β_1 に対する β_0 の推定量に関する公式
5.4.3 対数オッズ比パラメータ β_1 に関するWilson型のP値函数と信頼区間の構成
5.5 x_i 達の値も1または0の場合にロジスティック回帰の一般化の役に立ち方

In [1]:

```
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Optim
using Printf
using QuadGK
using RCall
@rimport stats as R
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, guidefontsize=9, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:
```
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]:
```
mykurtosis (generic function with 2 methods)
```

1 回帰 (regression)

このノートでは **回帰** (regression)について簡単に説明する.

統計学入門の教科書でこのノートにあるような解説が十分に書いてあるものを見つけることができなかったので, このノートの解説は貴重なものになる可能性がある.

1.1 回帰の超一般論

回帰 (regression)は, データの数値が

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$

の形式で,

$$\text{各々の } y_i \text{ が数値 } x_i \text{ に依存して確率的に決まっている}$$

と考えられる状況を, パラメータ $x = (x_1, \ldots, x_n)$, $\theta = (\theta_1, \ldots, \theta_d)$ を持つ $y = (y_1, \ldots, y_n)$ に関する確率密度函数(または確率質量函数)

$$p(y|x, \theta) \quad (\text{または } P(y|x, \theta))$$

でモデル化したときのパラメータ θ の推定の問題として, 一般的に定式化可能である.

1.2 データの数値から値が決まるパラメータ x がない場合

例: 比率の信頼区間の計算で使われる統計モデルは, 二項分布モデル

$$P(k|n, p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

であった. この場合には y にあたる変数が k で, x にあたる変数はない.

例: 平均の信頼区間の t 分布を使った計算法の基礎になる統計モデルは, 正規分布の標本分布

$$p(y|\mu, \sigma) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mu)^2\right)$$
$$(\mu, \sigma \in \mathbb{R},\ \sigma > 0,\ y = (y_1, \ldots, y_n) \in \mathbb{R}^n)$$

であるとみなせる(実際には中心極限定理による近似がうまく行っているという弱い仮定のもとで, その平均の信頼区間は実用的に使用可能). この場合には上の y にあたる変数がこの場合の y で, x にあたる変数はない.

x にあたる変数がモデルのパラメータとして登場することが, 回帰(regression)の特徴である.

1.3 回帰の例

例: 線形回帰 (linear regression)の最も簡単な単回帰の場合の統計モデルは次になる:

$$p(y|x,\beta_0,\beta_1,\sigma) = \frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i - (\beta_0+\beta_1 x_i))^2\right).$$

パラメータ $\beta_0,\beta_1,\sigma \in \mathbb{R}$, $\sigma > 0$ の推定は線形回帰と呼ばれる. この場合には, 最尤法によるパラメータの推定は最小二乗法そのものになる. **例終**

例: ロジスティック回帰 (logistic regression)の最も簡単な場合の統計モデルは次になる:

$$P(y|x,\beta_0,\beta_1) = \prod_{i=1}^n \left(f(\beta_0+\beta_1 x_i)^{y_i}(1-f(\beta_0+\beta_1 x_i)^{y_i})^{1-y_i}\right) \quad (y_i \in \{1,0\}).$$

ここで $f(t) = \mathrm{logistic}(t) = 1/(1+e^{-t})$ である. この統計モデルは各 x_i ごとに, y_i が 1 になる確率 p_i が x_i から

$$p_i = \mathrm{logistic}(\beta_0+\beta_1 x_i) = \frac{1}{1+\exp(-(\beta_0+\beta_1 x_i))}$$

で決まることを意味している. これは

$$\mathrm{logit}(p_i) = \log\frac{p_i}{1-p_i} = \beta_0+\beta_1 x_i.$$

と同値である. **例終**

以下の節ではこれらの回帰について詳しく説明する.

2 線形回帰

以下の説明よりももっと複雑な場合も線形回帰として扱えるが, 基本になる簡単な場合のみを扱う.

2.1 線形回帰のデータの形

データは以下の形で得られると仮定する:

$$(x_1,y_1),\ldots,(x_n,y_n) \in \mathbb{R}^2.$$

これを2つのベクトルで表す:

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}.$$

2.2 線形回帰モデルの構成要素

$r < n$ であると仮定する.

r 個の函数 $f_1,\ldots,f_r : \mathbb{R} \to \mathbb{R}$ が与えられているとし, その一次結合を決める **回帰係数** パラメータ $\beta_1,\ldots,\beta_r \in \mathbb{R}$ が与えられているとし,

$$\beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_r \end{bmatrix}, \quad f(x_i) = \begin{bmatrix} f_1(x_i) \\ \vdots \\ f_r(x_n) \end{bmatrix}$$

とおく. このとき,

$$f(x_i)^T\beta = [\, f_1(x_i) \quad \cdots \quad f_r(x_n)\,] \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_r \end{bmatrix} = \sum_{j=1}^r \beta_j f_j(x_i)$$

ここで, $f(x_i)^T$ は縦ベクトル $f(x_i)$ を転置して得られる横ベクトルを意味する.

さらに, 分散パラメータ $\sigma^2 > 0$ が与えられているとする.

例: $r = 2$ で $f_j(x_i)$, β_j のインデックス j を $1,2$ ではなく, $0,1$ を動かすことにする. このとき,

$$f_0(x_i) = 1, \quad f_1(x_i) = x_i$$

の場合は次のようになる.

$$f(x_i)^T\beta = \beta_0 + \beta_1 x_i.$$

以下の節を最初に読む場合にはこの例の場合に限定して考えた方がよいように思われる.

2.3 デザイン行列 (計画行列, design matrix)

デザイン行列 X を次のように定める:

$$\begin{aligned} X &= [\, f(x_1) \quad f(x_2) \quad \cdots \quad f(x_n) \,]^T \\ &= \begin{bmatrix} f(x_1)^T \\ f(x_2)^T \\ \vdots \\ f(x_n)^T \end{bmatrix} = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_r(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_r(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_n) & f_2(x_n) & \cdots & f_r(x_n) \end{bmatrix}. \end{aligned}$$

すなわち, 以上の文脈において, デザイン行列 X は r 次元の縦ベクトル $f(x_i)$ 達を横に n 個並べてできる行列の転置として定義され, $n \times r$ 行列になる. $r < n$ と仮定していたので, X は縦方向に長い行列になる.

このとき,

$$X\beta = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_r(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_r(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_n) & f_2(x_n) & \cdots & f_r(x_n) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_r \end{bmatrix} = \left[\sum_{j=1}^r \beta_j f_j(x_i) \right]_{i=1}^n.$$

以下では簡単のため, デザイン行列 X のランクは可能な最大値である r であると仮定する. (ほとんどの場合にそうなる.)

このとき $\mathbb{R}^n$ の部分空間 $X\mathbb{R}^r = \{\, X\beta \mid \beta \in \mathbb{R}^r \,\}$ の次元は r になり, 行列 X の定める線形写像 $X : \mathbb{R}^r \to \mathbb{R}^n$, $\beta \mapsto X\beta$ は単射になる.

さらに, そのとき $r \times r$ 行列 $X^T X$ が可逆になることも示せる. ここで X^T は X の転置を表す.

例: $r = 2$ で $f_j(x_i)$, β_j のインデックス j は $1, 2$ ではなく, $0, 1$ を動くことにし,

$$f_0(x_i) = 1, \quad f_1(x_i) = x_i$$

と仮定する. このとき, デザイン行列 X は次のようになる:

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}.$$

以下の節を最初に読む場合にはこの例の場合に限定して考えてもよい. より一般的な場合についてはこの特別な場合について理解した後に考えてもよい.

2.4 線形回帰の統計モデルの正規分布による記述

統計モデルとして, $y \in \mathbb{R}^n$ に関する次の確率密度函数を採用する:

$$p(y|X, \beta, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \|y - X\beta\|^2 \right).$$

ここで, $\| \ \|^2$ は通常のEuclidノルムの2乗(成分の2乗の和)を意味する:

$$\|y - X\beta\|^2 = \sum_{i=1}^n \left(y_i - \sum_{j=1}^r \beta_j f_j(x_i) \right)^2.$$

$Y = [Y_i]_{i=1}^n$ をこの統計モデルに従うベクトル値確率変数であると仮定する. Y は次のように表される:

$$Y = X\beta + \varepsilon$$

ここで, ベクトル値確率変数 $\varepsilon = [\varepsilon_i]_{i=1}^n$ の成分達 $\varepsilon_1, \ldots, \varepsilon_n$ はそれぞれが平均 0, 分散 σ^2 の正規分布に従う独立な確率変数達になる. ゆえに,

$$E[Y] = X\beta, \quad E[\varepsilon\varepsilon^T] = \sigma^2 I.$$

ここで I は n 次の単位行列を表す.

このことは一般に次が成立していることから確かめられる:

$$\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \begin{bmatrix} b_1 & b_2 & \cdots & b_n \end{bmatrix} = \begin{bmatrix} a_1b_1 & a_1b_2 & \cdots & a_1b_n \\ a_2b_1 & a_2b_2 & \cdots & a_2b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_nb_1 & a_nb_2 & \cdots & a_nb_n \end{bmatrix}.$$

(沢山の数をまとめて1つの記号で表すベクトルや行列の記号法は複雑な式を簡潔に表現するために非常に強力であるが, 簡潔になり過ぎてしまい, 初学者にとって何をやっているのか, 分かり難くなってしまうことも多い. そういう状況に自分が陥った場合には, すべての成分を書き下して, 具体的な式を書いてみると納得できることが多い. 簡潔で抽象的な式を見たら, 面倒臭く具体的な式を書くと理解し易くなることがあり, 逆に面倒臭く具体的に書かれた式を, ベクトルや行列をうまく使って簡潔に書き直すと理解し易くなることもある. 解説文の通りの式の表記にこだわる必然性はまったくない.)

このモデル内では, 仮想的なデータの数値 Y_i が $f_j(x_i)$ 達の一次結合 $\sum_{j=1}^r \beta_j f_j(x_i)$ で近似され, それらの差(残差と呼ばれる)

$$Y_i - \sum_{j=1}^r \beta_j f_j(x_i)$$

が平均 0, 分散 σ^2 の正規分布に従って独立にランダムに決まっている:

$$Y_i - \sum_{j=1}^r \beta_j f_j(x_i) \sim_{\text{i.i.d.}} \mathrm{Normal}(0, \sigma) \quad (i = 1, 2, \ldots, n).$$

注意: 以上の設定において Y はベクトル値確率変数だが, デザイン行列 X は確率変数では **ない**.

2.5 正規分布で書かれた統計モデルの最尤法から最小二乗法による線形回帰が得られること

上の統計モデルのデータの数値 x, y に関する尤度函数(ゆうどかんすう, 尤度はデータの数値と統計モデルの適合度(フィットの良さ)の指標の1つ)

$$(\beta, \sigma^2) \mapsto p(y|X, \beta, \sigma^2)$$

を最小化するパラメータ β, σ^2 の値から, 最小二乗法が得られることを説明しよう. (デザイン行列 X はベクトル x から決まる.)

尤度函数(ゆうどかんすう, モデルのパラメータとデータの数値の適合度の指標の1つ)

$$(\beta, \sigma^2) \mapsto p(y|X, \beta, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\|y - X\beta\|^2\right)$$

の対数の -2 倍は次の形になる:

$$L(\beta, \sigma^2) = -2\log p(y|X, \beta, \sigma^2) = \frac{1}{\sigma^2}\|y - X\beta\|^2 + n\log\sigma^2 + n\log(2\pi).$$

これを最小化するパラメータ β, σ^2 の値を求めよう.

データの数値として与えられた y とパラメータ β に依存する $X\beta$ の距離の2乗 $\|y - X\beta\|^2$ を最小にするベクトル $X\beta$ はベクトル $y \in \mathbb{R}^n$ の部分空間 $X\mathbb{R}^r$ への直交射影 $\hat{y}$ になる.

そのとき, $X : \mathbb{R}^r \to \mathbb{R}^n$ は単射なので $\hat{y} = X\hat{\beta}$, $\hat{\beta} \in \mathbb{R}^r$ と一意に表される.

以下の図を見よ.

回帰係数 β をそのような $\hat{\beta}$ として求める方法は, 距離の2乗 $\|y - X\beta\|^2$ を最小化するので, **最小二乗法** と呼ばれる.

以上の議論から, **最小二乗法は直交射影そのものに過ぎない** ことがわかる.

$y - \hat{y} = y - X\hat{\beta}$ を **残差** (residual error)と呼ぶ.

$\hat{\sigma}^2$ を y とその直交射影 $\hat{y}$ の距離の2乗の n 分の 1 と定める:

$$\hat{\sigma}^2 = \frac{1}{n}\|y - \hat{y}\|^2 = \frac{1}{n}\|y - X\hat{\beta}\|^2.$$

このとき,

$$L(\hat{\beta}, \sigma^2) = \frac{n\hat{\sigma}^2}{\sigma^2} + n\log\sigma^2 + n\log(2\pi).$$

を最小化する σ^2 は $\sigma^2 = \hat{\sigma}^2$ になることを確認できる.

(一般に $\sigma^2 > 0$ の函数 $\sigma^2 \mapsto a/\sigma^2 + \log\sigma^2$ $(a > 0)$ は $\sigma^2 = a$ で最小になる. 実際, $f(s) = a/s + logs$ について $f'(s) = (s - a)/s^2$ なので, $s < a$ のとき $f'(s) < 0$ で $f(s)$ は単調減少し, $s > a$ のとき $f'(s) > 0$ で $f(s)$ は単調増加する. ゆえに $f(s)$ は $s = a$ で最小になる.)

以上のようにして, 直交射影で求めた $\hat{\beta}$ と $\hat{\sigma}^2$ は最尤法の解になっている.

($\hat{\beta}$ と $\hat{\sigma}^2$ の具体的な形は後で求める.)

In [4]:
```
plot(s -> 3/s + log(s), 2, 4; label="",
    title="f(s) = 3/s + log(s)", xguide="s")
```

Out[4]:

2.6 直交射影の公式

ベクトル y の部分空間 $X\mathbb{R}^r$ への直交射影 $\hat{y} = X\hat{\beta}$ の具体的な形を求めよう.

ベクトル $y - \hat{y} = y - X\hat{\beta}$ は部分空間 $X\mathbb{R}^r$ と直交するので, 任意の $\gamma \in \mathbb{R}^r$ について

$$0 = (X\gamma, y - X\hat{\beta}) = (X\gamma)^T(y - X\hat{\beta}) = \gamma^T X^T(y - X\hat{\beta}).$$

ここで, $(\ ,\ )$ は通常の内積(成分の積の和)を表す. ベクトル $\gamma \in \mathbb{R}^r$ は任意なので,

$$X^T(y - X\hat{\beta}) = 0, \quad \text{すなわち} \quad X^T X\hat{\beta} = X^T y.$$

X のランクが可能な最大値 r になると仮定していたので, $X^T X$ は可逆になるのであった. ゆえに

$$\hat{\beta} = (X^T X)^{-1} X^T y, \quad \hat{y} = X(X^T X)^{-1} X^T y.$$

これと,

$$\hat{\sigma}^2 = \frac{1}{n}\|y - X\hat{\beta}\|^2$$

を合わせると, 尤度函数 $(\beta, \sigma^2) \mapsto p(y|X, \beta, \sigma^2)$ を最大化するパラメータ値を計算できる(コンピュータ上に実装できる).

これが正規分布で記述された統計モデルの最尤法の解である.

注意: 以上の議論によって, 最小二乗法は本質的に直交射影を作る操作であることも分かる. このように内積に関する線形代数を理解していれば, その中に最小二乗法の理論も含まれていると考えることができる. 線形代数は普遍的に役に立つ道具である.

2.7 βとσ²の不偏推定量

以下では, $\hat{\beta}$ と $\hat{s}^2 = (n/(n-r))\hat{\sigma}^2 = (1/(n-r))\|y - X\hat{\beta}\|^2$ がそれぞれ β と σ^2 の不偏推定量になっていることを示す.

この節では, 前々節で記述した統計モデルに従う確率変数 Y とは別に, ベクトル値確率変数 $y = [y_i]_{i=1}^n$ で次の条件を満たすものを使用する:

$$y = X\beta + e, \quad E[e] = 0, \quad E[ee^T] = \sigma^2 I.$$

このようなベクトル値確率変数 y を使うことは, 正規分布で記述された統計モデルの設定を大幅に弱めることを意味している. この節ではそのような状況を扱う.

上の条件中の $E[ee^T] = \sigma^2 I$ は e の分散共分散行列であることに注意せよ.

(一般に n 次元ベクトル値確率変数 $V = [V_i]_{i=1}^n$ について, $\mu = E[V]$ のとき, $n \times n$ の対称行列 $E[(V-\mu)(V-\mu)^T]$ を V の分散共分散行列 (variance-covariance matrix) と呼ぶ. その (i,i) 成分は V_i の分散になっており, $i \neq j$ に関する (i,j) 成分は V_i と V_j の共分散になっている.)

$e = [e_i]_{i=1}^n$ と書くとき, $E[ee^T] = \sigma^2 I$ は

$$E[e_i e_j] = \sigma^2 \delta_{ij}$$

が成立することを意味している. ここで δ_{ij} はKroneckerのデルタである. ($i = j$ のとき $\delta_{ij} = 1$ で $i \neq j$ のとき $\delta_{ij} = 0$.)

上の仮定の下で, y の平均と分散共分散行列はそれぞれ次のようになる:

$$E[y] = X\beta, \quad E[(y - X\beta)(y - X\beta)^T] = E[ee^T] = \sigma^2 I.$$

この設定における最小二乗法は次のように書ける:

$$\hat{\beta} = (X^T X)^{-1} X^T y, \quad \hat{\sigma}^2 = \frac{1}{n}\|y - X\hat{\beta}\|^2.$$

このとき,

$$E[\hat{\beta}] = (X^T X)^{-1} X^T X\beta = \beta$$

なので, $\hat{\beta}$ は β の不偏推定量になっている.

以下では σ^2 の不偏推定量を構成しよう.

$$\hat{\beta} - \beta = (X^T X)^{-1} X^T (X\beta + e) - \beta = (X^T X)^{-1} X^T e$$

より,

$$(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T = (X^T X)^{-1} X^T e e^T X (X^T X)^{-1}$$

なので, ベクトル値確率変数 $\hat{\beta} = (X^T X)^{-1} X^T y$ の分散共分散行列は次のようになる:

$$E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T] = (X^T X)^{-1} X^T E[ee^T] X (X^T X)^{-1} = \sigma^2 (X^T X)^{-1}.$$

2つ目の等号で $E[ee^T] = \sigma^2 I$ を使った.

$$X\hat{\beta} - X\beta = X(\hat{\beta} - \beta), \quad (X\hat{\beta} - X\beta)(X\hat{\beta} - X\beta)^T = X(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T X^T$$

より, $X\hat{\beta}$ の分散共分散行列は次のように計算される:

$$E[(X\hat{\beta} - X\beta)(X\hat{\beta} - X\beta)^T] = XE[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T]X^T = \sigma^2 X (X^T X)^{-1} X^T.$$

2つ目の等号で, 上で示した $E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T] = \sigma^2 (X^T X)^{-1}$ を使った.

$X(X^T X)^{-1} X^T$ は $\mathbb{R}^n$ からその部分空間 $X\mathbb{R}^r$ への直交射影を与える行列であった. (特に対称行列でかつ二乗しても不変であること $\left(X(X^T X)^{-1} X^T\right)^2 = X(X^T X)^{-1} X^T$ にも注意せよ.)

$$\begin{aligned} y - X\hat{\beta} &= y - X(X^T X)^{-1} X^T y = (I - X(X^T X)^{-1} X^T) y \\ E[y - X\hat{\beta}] &= E[y] - X(X^T X)^{-1} X^T E[y] = X\beta - X(X^T X)^{-1} X^T X\beta = 0 \end{aligned}$$

より, 残差 $y - X\hat{\beta}$ の分散共分散行列は次のように計算される:

$$
\begin{aligned}
E[(y-X\hat{\beta})(y-X\hat{\beta})^T] &= (I-X(X^TX)^{-1}X^T)E[yy^T](I-X(X^TX)^{-1}X^T)^T \\
&= \sigma^2(I-X(X^TX)^{-1}X^T)(I-X(X^TX)^{-1}X^T)^T \\
&= \sigma^2(I-X(X^TX)^{-1}X^T).
\end{aligned}
$$

ここで, $E[yy^T]=\sigma^2 I$ および, $X(X^TX)^{-1}X^T$ が $\mathbb{R}^n$ から $X\mathbb{R}^r$ への直交射影を与える行列であったことから, $I-X(X^TX)^{-1}X^T$ が $\mathbb{R}^n$ から $X\mathbb{R}^r$ の直交補空間への射影を与える行列になり, 特に対称行列でかつ二乗しても不変になることを使った.

(一般に $P^T=P$, $P^2=P$ のとき, $(I-P)^T=I-P$ かつ $(I-P)^2=I-P$ となる.)

トレースの中で行列の順序を巡回的に回してもトレースの値は不変なので,

$$
\mathrm{tr}(X(X^TX)^{-1}X^T)=\mathrm{tr}((X^TX)^{-1}X^TX)=\mathrm{tr}(I_r)=r.
$$

ここで I_r は r 次の単位行列を表す. この結果と

$$
\|y-X\hat{\beta}\|^2=(y-X\hat{\beta})^T(y-X\hat{\beta})=\mathrm{tr}\Big((y-X\hat{\beta})(y-X\hat{\beta})^T\Big)
$$

を使うと,

$$
\begin{aligned}
E[\|y-X\hat{\beta}\|^2] &= \mathrm{tr}\Big(E[(y-X\hat{\beta})(y-X\hat{\beta})^T]\Big) \\
&= \sigma^2\,\mathrm{tr}(I-X(X^TX)^{-1}X^T)=(n-r)\sigma^2.
\end{aligned}
$$

2つめの等号で上で示した $E[(y-X\hat{\beta})(y-X\hat{\beta})^T]=\sigma^2(I-X(X^TX)^{-1}X^T)$ を用い, 3つ目の等号で上で示した $\mathrm{tr}(X(X^TX)^{-1}X^T)=r$ を使った.

ゆえに,

$$
\hat{s}^2=\frac{1}{n-r}\|y-X\hat{\beta}\|^2=\frac{n}{n-r}\hat{\sigma}^2
$$

とおくと,

$$
E[\hat{s}^2]=\sigma^2.
$$

すなわち, $\hat{s}^2=\|y-X\hat{\beta}\|^2/(n-r)$ は σ^2 の不偏推定量である.

σ^2 の不偏推定量を作るためには残差の二乗和 $\|y-X\hat{\beta}\|^2$ を n ではなく, $n-r$ で割らなければいけない.

注意: 特にこの節を見れば分かるように, 線形回帰の理解のために最も重要な基礎知識は線形代数である. 特に直交射影に関する線形代数が重要になる.

2.8 例: 平均の推定の場合

線形回帰は平均の推定を含む.

$r=1$ とし, $f_j(x_i)$ と β_j のインデックス j として 1 ではなく 0 を使うことにし,

$$
f_0(x_i)=1,\quad \beta_0=\mu,\quad \hat{\beta}_0=\hat{\mu}
$$

の場合について考える. このとき, デザイン行列 X は n 次元縦ベクトルになり,

$$
X=\begin{bmatrix}1\\1\\\vdots\\1\end{bmatrix},\quad X\beta=\begin{bmatrix}\mu\\\mu\\\vdots\\\mu\end{bmatrix}.
$$

ゆえに, $X^TX=n$, $X^Ty=\sum_{i=1}^n y_i$ となるので,

$$
\hat{\mu}=(X^TX)^{-1}X^Ty=\frac{1}{n}\sum_{i=1}^n y_i=:\bar{y}.
$$

さらに, $y-X\hat{\beta}=[y_i-\hat{\mu}]_{i=1}^n=[y_i-\bar{y}]_{i=1}^n$ より,

$$
\hat{\sigma}^2=\frac{1}{n}\|y-X\hat{\beta}\|=\frac{1}{n}\sum_{i=1}^n(y_i-\bar{y})^2.
$$

これは. σ^2 を. 不偏推定量になるように $n-1$ で割らずに. n で割って作った場合の標本分散になっている.

前節で示したように σ^2 の不偏推定量 $\hat{s}^2$ は $n-r=n-1$ で割ることによって得られる:

$$\hat{s}^2 = \frac{1}{n-1}\|y - X\hat{\beta}\| = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2.$$

この結果は標本の不偏分散の話の線形回帰に関する結果を使った再現になっている.

2.9 例: 単回帰の場合

$r=2$ とし, $f_j(x_i)$ と β_j のインデックス j として $1,2$ ではなく $0,1$ を使うことにし,

$$f_0(x_i) = 1, \quad f_1(x_i) = x_i$$

の場合を考える. さらに, 以下のようにおく:

$$\begin{aligned}
&\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i, \\
&\overline{x^2} = \frac{1}{n}\sum_{i=1}^{n} x_i^2, \quad \overline{y^2} = \frac{1}{n}\sum_{i=1}^{n} y_i^2, \quad \overline{xy} = \frac{1}{n}\sum_{i=1}^{n} x_i y_i.
\end{aligned}$$

このとき, x_i 達と y_i 達の不偏補正をしていない標本分散と標本共分散はそれぞれ次のように書ける:

$$\begin{aligned}
\hat{\sigma}_x^2 &:= \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \overline{x^2} - \bar{x}^2, \\
\hat{\sigma}_y^2 &:= \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2 = \overline{y^2} - \bar{y}^2, \\
\hat{\sigma}_{xy} &:= \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = \overline{xy} - \bar{x}\bar{y}.
\end{aligned} \tag{$*$}$$

このとき, デザイン行列 X については, 以下が成立していることをちょっとした計算で確認できる(自分で確認してみよ):

$$\begin{aligned}
&X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}, \quad X^T X = n\begin{bmatrix} 1 & \bar{x} \\ \bar{x} & \overline{x^2} \end{bmatrix}, \\
&X^T y = n\begin{bmatrix} \bar{y} \\ \overline{xy} \end{bmatrix}, \quad (X^T X)^{-1} = \frac{1}{n(\overline{x^2} - \bar{x}^2)}\begin{bmatrix} \overline{x^2} & -\bar{x} \\ -\bar{x} & 1 \end{bmatrix}.
\end{aligned}$$

ゆえに,

$$\overline{x^2}\bar{y} - \bar{x}\overline{xy} = \overline{x^2}\bar{y} - \bar{x}^2\bar{y} + \bar{x}^2\bar{y} - \bar{x}\overline{xy} = \hat{\sigma}_x^2\bar{y} - \hat{\sigma}_{xy}\bar{x}$$

が成立していることに注意すれば,

$$\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} = \hat{\beta} = (X^T X)^{-1} X^T y = \begin{bmatrix} \bar{y} - (\hat{\sigma}_{xy}/\hat{\sigma}_x^2)\bar{x} \\ \hat{\sigma}_{xy}/\hat{\sigma}_x^2 \end{bmatrix}.$$

すなわち, 次の回帰直線の公式が得られた:

$$\hat{\beta}_0 + \hat{\beta}_1 x_* = \bar{y} + \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x^2}(x_* - \bar{x}).$$

$\hat{\sigma}^2$ の公式も求めよう.

$$y - X\hat{\beta} = (I - X(X^T X)^{-1} X^T)y$$

より,

$$\begin{aligned}
\|y - X\hat{\beta}\|^2 &= y^T(I - X(X^T X)^{-1} X^T)^T(I - X(X^T X)^{-1} X^T)y \\
&= y^T(I - X(X^T X)^{-1} X^T)y \\
&= y^T y - (X^T y)^T (X^T X)^{-1} X^T y.
\end{aligned}$$

そして,

$$
\begin{aligned}
&y^T y = n\overline{y^2}, \\
&(X^T y)^T (X^T X)^{-1} X^T y = \frac{n}{\hat{\sigma}_x^2}\left(\overline{x^2} - 2\bar{x}\bar{y}\overline{xy} + \overline{xy}^2\right)
\end{aligned}
$$

および上の(*)を使って整理すると,

$$
\hat{\sigma}^2 = \frac{1}{n}\|y - X\hat{\beta}\|^2 = \frac{\hat{\sigma}_x^2\hat{\sigma}_y^2 - \hat{\sigma}_{xy}^2}{\hat{\sigma}_x^2}.
$$

前々節の結果より, σ^2 の不偏推定量 $\hat{s}^2$ は n ではなく, $n - r = n - 2$ で割ることによって次のようにして得られる:

$$
\hat{s}^2 = \frac{1}{n-2}\|y - X\hat{\beta}\|^2 = \frac{n}{n-2}\frac{\hat{\sigma}_x^2\hat{\sigma}_y^2 - \hat{\sigma}_{xy}^2}{\hat{\sigma}_x^2}.
$$

x_i 達と y_i 達の不偏分散と不偏共分散を

$$
s_x^2 := \frac{n}{n-1}\hat{\sigma}_x^2, \quad s_y^2 := \frac{n}{n-1}\hat{\sigma}_y^2, \quad s_{xy} := \frac{n}{n-1}\hat{\sigma}_{xy}
$$

と書くと,

$$
\begin{aligned}
&\beta_0 = \bar{y} - \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x^2}\bar{x} = \bar{y} - \frac{s_{xy}}{s_x^2}\bar{x}, \quad \beta_1 = \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x^2} = \frac{s_{xy}}{s_x^2}, \\
&\hat{\sigma}^2 = \frac{n-1}{n}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}, \quad \hat{s}^2 = \frac{n-1}{n-2}\frac{s_x^2 s_y^2 - s_{xy}^2}{s_x^2}.
\end{aligned}
$$

以上の公式は

- [「標本分布について」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb) (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/04%20Distribution%20of%20samples.ipynb)

の「最小二乗法による線形回帰」の節で得た公式の再現になっている.

2.10 Julia言語による回帰直線の計算の最も簡単な例

In [5]:

```julia
# パッケージの読み込みなど
using Distributions
using StatsPlots
using Random
Random.seed!(4649373)

# テストデータのランダム生成
n = 100
x = rand(Normal(0, 1), n);
e = rand(Normal(0, 1), n);
β = [-1, 2]
σ = 2
y = @. β[1] + β[2]*x + σ*e;
scatter(x, y; label="data", legend=:topleft, msc=:auto)
plot!(xstar -> β[1] + β[2]*xstar; label="true model", c=:black, ls=:dash)
```

Out[5]:

In [6]:
```
# デザイン行列
X = x .^ (0:1)';

# デザイン行列の上から5行分を表示
X[1:5, :]
```

Out[6]:
```
5×2 Matrix{Float64}:
 1.0   0.843695
 1.0   0.286651
 1.0  -0.117361
 1.0   1.43215
 1.0  -1.39703
```

In [7]:
```
# 線形回帰
# 以下は β̂ = X'X\X'y もしくは β̂ = inv(X'*X)*X'*y と同値
β̂ = X \ y
```

Out[7]:
```
2-element Vector{Float64}:
 -1.0404163851842536
  2.235682111246987
```

In [8]:
```
inv(X'*X)*X'*y
```

Out[8]:
```
2-element Vector{Float64}:
 -1.0404163851842536
  2.235682111246987
```

In [9]:
```
# 結果をプロット
scatter(x, y; label="data", legend=:topleft, msc=:auto)
plot!(xstar -> β[1] + β[2]*xstar; label="true model", c=:black, ls=:dash)
plot!(xstar -> β̂[1] + β̂[2]*xstar; label="MLE", c=2, lw=1.5)
```

Out[9]:

MLEは最尤推定(さいゆうほう, maximum likelihood estimate)の略である.

[Julia言語 (https://julialang.org/)](download (https://julialang.org/downloads/)) の current stable release をダウンロードして, インストールして, 実行して,

```
julia>
```

で] を押した後に,

```
pkg> add Distributions, StatsPlots
```

を実行し,

```
pkg>
```

でバックスペースキーを押して,

```
julia>
```

の状態に上のコードを貼り付ければ上と同じことをできるはずである.

Julia言語では最小二乗法による回帰は `β̂ = X \ y` だけで可能である.

`X'` は `X` の転置(の複素共役)を意味する.

上のコードをコピー＆ペーストすることによって以下のスクリーンショットのように計算して，グラフを作ることができる．

```
julia> σ = 2
2

julia> y = @. β[1] + β[2]*x + σ*e;

julia> scatter(x, y; label="data", legend=:topleft, msc=:auto)


julia> plot!(xstar -> β[1] + β[2]*xstar; label="true model", c=:black, ls=:dash)


julia> X = x .^ (0:1)';

julia> X[1:5, :]
5×2 Matrix{Float64}:
 1.0   0.843695
 1.0   0.286651
 1.0  -0.117361
 1.0   1.43215
 1.0  -1.39703

julia> β = X \ y
2-element Vector{Float64}:
 -1.0404163851842536
  2.235682111246987

julia> scatter(x, y; label="data", legend=:topleft, msc=:auto)

julia> plot!(xstar -> β[1] + β[2]*xstar; label="true model", c=:black, ls=:dash)


julia> plot!(xstar -> β[1] + β[2]*xstar; label="estimate", c=2, lw=1.5)

julia>
```

$\hat{\beta}$ = X \ y の部分の $\hat{\beta}$ が潰れて β と表示されている点は気にしないことにした.

できあがったグラフは以下のように表示されている.

ここまで無事にたどり付けなかった場合には, エラーメッセージをインターネットで検索すれば解決法が見つかることが多い.

2.11 多変量正規分布の定義

$\mu \in \mathbb{R}^n$ と固有値がすべて正の n 次の実対称行列 Σ に対して, 多変量正規分布

$$\mathrm{MvNormal}(\mu, \Sigma)$$

の確率密度函数を次のように定める:

$$p(y|\mu, \Sigma) = \frac{1}{\det(2\pi\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(y-\mu)^T \Sigma^{-1} (y-\mu)\right) \quad (y \in \mathbb{R}^n).$$

例: $\mu = (m, m, \ldots, m)$, $\Sigma = \sigma^2 I$ (I は n 次の単位行列で $\sigma^2 > 0$)のとき,

$$p(y|\mu, \Sigma) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - m)^2\right).$$

これは正規分布の標本分布 $\mathrm{Normal}(m, \sigma^2)^n$ の密度函数に等しい.

2.12 問題: 多変量正規分布とχ²分布の関係

前節の n 変量正規分布 $\mathrm{MvNormal}(\mu, \Sigma)$ を考える:

$$p(y|\mu, \Sigma) = \frac{1}{\det(2\pi\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(y-\mu)^T \Sigma^{-1} (y-\mu)\right) \quad (y \in \mathbb{R}^n).$$

この分布に従う n 次元ベクトル値確率変数を Y と書き, 確率変数 χ^2 を

$$\chi^2 = (Y-\mu)^T \Sigma^{-1} (Y-\mu)$$

と定めると, χ^2 は自由度 n のχ²分布に従うことを示せ.

解答例: 固有値達 σ_i^2 がすべて正の実対称行列 Σ はある直交行列 U によって次のように対角化できる:

$$\Sigma = UD^2U^{-1} = UD^2U^T, \quad D = \mathrm{diag}(\sigma_1, \ldots, \sigma_n), \quad \sigma_i > 0.$$

このとき,

$$\det(2\pi\Sigma)^{1/2} = \prod_{i=1}^n (2\pi\sigma_i^2)^{1/2} = (2\pi)^{n/2} \sigma_1 \cdots \sigma_n.$$

さらに, $x = [x_i]_{i=1}^n = D^{-1}U^T(y-\mu)$ とおくと, $\Sigma^{-1} = UD^{-2}U^T$ より

$$(y-\mu)^T \Sigma^{-1} (y-\mu) = x^T x$$

となり, 直交行列による変換が体積を保つことより,

$$|dy_1 \cdots dy_n| = \sigma_1 \cdots \sigma_n \, |dx_1 \cdots dx_n|$$

となるので,

$$\begin{aligned}
p(y|\mu, \Sigma) \, |dy_1 \cdots dy_n| &= \frac{1}{(2\pi)^{n/2} \sigma_1 \cdots \sigma_n} \exp\left(-\frac{1}{2}(y-\mu)^T \Sigma^{-1} (y-\mu)\right) |dy_1 \cdots dy_n| \\
&= \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{1}{2} x^T x\right) |dx_1 \cdots dx_n| \\
&= \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{1}{2} \sum_{i=1}^n x_i^2\right) |dx_1 \cdots dx_n|.
\end{aligned}$$

ゆえに, 変数 x に対応するベクトル値確率変数を $X = [X_i]_{i=1}^n$ と書くと, $X_1, \ldots, X_n$ はそれぞれが標準正規分布に従う独立な確率変数達になり,

$$\chi^2 = (Y-\mu)^T \Sigma^{-1} (Y-\mu) = X^T X = \sum_{i=1}^n X_i^2$$

となるので, χ^2 は自由度 n のχ²分布に従う.

解答終

2.13 真の回帰函数と推定された回帰函数

「βとσ²の不偏推定量」の節の仮定の下で, $x_* \in \mathbb{R}$ の函数

$$f(x_*)^T\beta = \sum_{j=1}^{r} \beta_j f_j(x_*)$$

を **真の回帰函数** と呼び,

$$f(x_*)^T\hat{\beta} = \sum_{j=1}^{r} \hat{\beta}_j f_j(x_*)$$

を **推定された回帰函数** と呼ぶことにする.

注意: 真の回帰函数の「真の」の意味は「現実における真の」という意味ではない. 仮想的なデータを生成するために用いたモデルのパラメータの設定を意味するに過ぎない.

2.14 (真の)回帰直線の値の信頼区間 (標準正規分布版)

この節では, 「βとσ²の不偏推定量」の節で仮定した条件に加えて, ベクトル値 $\hat{\beta} = (X^TX)^{-1}X^Ty$ が近似的に多変量正規分布に従い, $\hat{s}^2 \approx \sigma^2$ という近似が成立していると仮定する.

このとき, 「βとσ²の不偏推定量」の節で示した結果

$$E[\hat{\beta}] = \beta, \quad E[(\hat{\beta}-\beta)(\hat{\beta}-\beta)^T] = \sigma^2(X^TX)^{-1}$$

より, 次の近似が成立している:

$$\hat{\beta} \sim \mathrm{MvNormal}(\beta,\ \hat{s}^2(X^TX)^{-1}),\ \text{approximately}.$$

ゆえに, $x_* \in \mathbb{R}$ について,

$$f(x_*)^T\hat{\beta} \sim \mathrm{Normal}\Big(f(x_*)^T\beta,\ \widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}\Big),\ \text{approximately}.$$

ここで,

$$\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}} = \hat{s}\sqrt{f(x_*)^T(X^TX)^{-1}f(x_*)}$$

これより, 真の回帰函数の $x_* \in \mathbb{R}$ の値 $f(x_*)^T\beta$ に関する仮説 $f(x_*)^T\beta = f(x_*)^T\beta_0$ のP値と $f(x_*)^T\beta$ の値の信頼区間を定義できる.

(真の)回帰函数の x_* における値に関する仮説 $f(x_*)^T\beta = f(x_*)^T\beta_0$ のP値の定義: $t(x_*, \beta_0)$ を

$$t(x_*, \beta_0) = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta_0}{\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}} = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta_0}{\hat{s}\sqrt{f(x_*)^T(X^TX)^{-1}f(x_*)}}$$

と定め, P値を次のように定める:

$$\mathrm{pvalue}_{\mathrm{Normal}}(y|X, f(x_*)^T\beta = f(x_*)\beta_0) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |t(x_*, \beta_0)|)).$$

(真の)回帰函数の値 $f(x_*)^T\beta$ に関する信頼度 $1-\alpha$ の信頼区間の定義: $z_{\alpha/2}$ を

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1-\alpha/2).$$

と定め, 信頼区間を次のように定める:

$$\mathrm{confint}_{\mathrm{Normal}}^{f(x_*)^T\beta}(y|X) = \left[f(x_*)^T\hat{\beta} - z_{\alpha/2}\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}},\ f(x_*)^T\hat{\beta} + z_{\alpha/2}\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}\right].$$

注意: 「真の」は「現実における真の」を意味しない. 単に仮想的なデータを生成するモデルのパラメータの設定を意味するに過ぎない.

2.15 (真の)回帰直線の値の信頼区間 (t分布版)

前節の標準正規分布を使って定義されたP値と信頼区間の t 分布を使った補正を構成しよう.

そのために, 前節までに仮定していた条件よりもさらに強い次の条件を仮定する:

$$e = [e_i]_{i=1}^n \sim \mathrm{Normal}(0,\sigma)^n = \mathrm{MvNormal}(0, \sigma^2 I).$$

これは正規分布で記述されていた統計モデル内の設定に戻ったことを意味している.

このとき,

$$\hat{\beta} \sim \mathrm{MvNormal}(\beta,\ \sigma^2(X^TX)^{-1}).$$

ゆえに, $x_* \in \mathbb{R}$ について,

$$f(x_*)^T\hat{\beta} \sim \mathrm{Normal}\Big(f(x_*)^T\beta,\ \mathrm{SE}_{f(x_*)^T\hat{\beta}}\Big).$$

ここで,

$$\mathrm{SE}_{f(x_*)^T\beta} = \sigma\sqrt{f(x_*)^T(X^TX)^{-1}f(x_*)}.$$

さらに,

$$y - X\hat{\beta} = (X\beta + e) - X(X^TX)^{-1}X^T(X\beta + e) = (I - X(X^TX)^{-1}X^T)e$$

が部分空間 $X\mathbb{R}^r$ の直交補空間($n-r$ 次元になる)へのベクトル値確率変数 $e \sim \mathrm{MvNormal}(0, \sigma^2 I)$ の直交射影であることから,

$$\frac{(n-r)\hat{s}^2}{\sigma^2} = \|y - X\hat{\beta}\|^2 \sim \mathrm{Chisq}(n-r)$$

となることを示せる. (「多変量正規分布とχ²分布の関係」を使えば容易に示される.)

$\hat{y} = X\hat{\beta}$ と $y - \hat{y} = y - X\hat{\beta}$ が独立であることより(その独立性はそれらが直交することから導かれる), $f(x_*)^T\hat{\beta}$ と $(n-r)\hat{s}^2/\sigma^2 = \|y - X\hat{\beta}\|^2$ が独立になることも導かれるので, 一般に独立な確率変数達 $Z \sim \mathrm{Normal}(0,1)$ と $Y \sim \mathrm{Chisq}(\nu)$ について,

$$\frac{Z}{\sqrt{Y/\nu}} \sim \mathrm{TDist}(\nu)$$

が成立することより,

$$t(x_*, \beta) = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta}{\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}} = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta}{\hat{s}\sqrt{f(x_*)^T(X^TX)^{-1}f(x_*)}} \sim \mathrm{TDist}(n-r).$$

これを使うと t 分布を使って補正したP値と信頼区間を以下のように定義できる:

(真の)回帰函数の値に関する仮説 $f(x_*)^T\beta = f(x_*)^T\beta_0$ のP値の定義: $t(x_*, \beta_0)$ を

$$t(x_*, \beta_0) = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta_0}{\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}} = \frac{f(x_*)^T\hat{\beta} - f(x_*)^T\beta_0}{\hat{s}\sqrt{f(x_*)^T(X^TX)^{-1}f(x_*)}}$$

と定め, P値を次のように定める:

$$\mathrm{pvalue}_{\mathrm{TDist}}(y|X, f(x_*)^T\beta = f(x_*)\beta_0) = 2(1 - \mathrm{cdf}(\mathrm{TDist}(n-r), |t(x_*, \beta_0)|)).$$

(真の)回帰函数の値 $f(x_*)^T\beta$ の信頼度 $1-\alpha$ の信頼区間の定義: $t_{\nu,\alpha/2}$ を

$$t_{\nu,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(\nu), 1 - \alpha/2).$$

と定め, 信頼区間を次のように定める:

$$\mathrm{confint}_{\mathrm{TDist}}^{f(x_*)^T\beta}(y|X) = \left[f(x_*)^T\hat{\beta} - t_{n-r,\alpha/2}\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}},\ f(x_*)^T\hat{\beta} + t_{n-r,\alpha/2}\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}\right].$$

注意: 「真の」は「現実における真の」を意味しない. 単に仮想的なデータを生成するモデルのパラメータの設定を意味するに過ぎない.

2.16 予測区間

前節の仮定 $e \sim \mathrm{Normal}(0,\sigma)^n$ に加えて、さらに

$$\begin{bmatrix} e \\ e_* \end{bmatrix} \sim \mathrm{Normal}(0,\sigma)^{n+1}$$

と仮定し, 任意に $x_* \in \mathbb{R}$ を取り,

$$y_* = f(x_*)^T\beta + e_*$$

とおく. このとき,

$$y_* - f(x_*)^T\hat{\beta} = e_* + f(x_*)^T\beta - f(x_*)^T\hat{\beta} \sim \mathrm{Normal}\Big(0, \mathrm{SE}_{y_*-f(x_*)^T\hat{\beta}}\Big).$$

ここで,

$$\begin{aligned}\Big(\mathrm{SE}_{y_*-f(x_*)^T\hat{\beta}}\Big)^2 &= (e_* \text{ の分散}) + (f(x_*)^T\beta - f(x_*)^T\hat{\beta} \text{ の分散}) \\ &= \sigma^2 + \sigma^2 f(x_*)^T(X^TX)^{-1}f(x_*).\end{aligned}$$

すなわち,

$$\mathrm{SE}_{y_*-f(x_*)^T\hat{\beta}} = \sigma\sqrt{1 + f(x_*)^T(X^TX)^{-1}f(x_*)}.$$

前節の $\mathrm{SE}_{f(x_*)^T\hat{\beta}}$ との違いは平方根の中に e_* の分散から出て来た項の 1 が含まれていることである.

これより,

$$\widehat{\mathrm{SE}}_{y_*-f(x_*)^T\hat{\beta}} = \hat{s}\sqrt{1 + f(x_*)^T(X^TX)^{-1}f(x_*)}.$$

とおくと, 前節と同様にして,

$$\frac{y_* - f(x_*)^T\hat{\beta}}{\widehat{\mathrm{SE}}_{y_*-f(x_*)^T\hat{\beta}}} \sim \mathrm{TDist}(n-r).$$

これを使って, y_* の **予測区間** (prediction interval)を次のように定義できる:

$$\mathrm{predint}^{y_*}_{\mathrm{TDist}}(y|X, x_*) = \left[f(x_*)^T\hat{\beta} - t_{n-r,\alpha/2}\widehat{\mathrm{SE}}_{y_*-f(x_*)^T\hat{\beta}},\ f(x_*)^T\hat{\beta} + t_{n-r,\alpha/2}\widehat{\mathrm{SE}}_{y_*-f(x_*)^T\hat{\beta}}\right].$$

前節の $\widehat{\mathrm{SE}}_{f(x_*)^T\hat{\beta}}$ と $\widehat{\mathrm{SE}}_{y_*-f(x_*)^T\hat{\beta}}$ の違いは, 後者の定義式の平方根中に 1 が含まれていることである. だから, 予測区間は信頼区間よりも必ず広くなる. そうなる理由は y_* の定義を見れば明らかで, y_* の定義にはノイズの項 e_* が含まれている. その分だけ区間の幅が広くなる.

2.17 「回帰函数の値の信頼区間」と「予測区間」の違い

回帰函数の値の信頼区間は, 残差を含まない回帰函数の値(最も簡単な場合には回帰直線上の値)の信頼区間であり, 残差の分の確率的揺らぎを持つデータの数値の予測には使えない.

予測区間は, 残差による確率的揺らぎの大きさを含んでおり, データの数値の予測に使える.

実用的にはこの節より前の部分にある面倒な計算を理解していなくても, この違いだけを理解していれば足りることが多いように思われる.

しかし, 言葉での説明では分かり難いの次の節の計算例の視覚化を参照せよ.

3 線形回帰の計算例

以下において CI, PI はそれぞれ

- 真の回帰函数の値の信頼区間 (confidence interval)
- データの値(＝真の回帰函数＋ノイズ)の予測区間 (prediction interval)

の略である.

3.1 信頼区間と予測区間のプロット

In [10]:

```julia
function plot_linreg_confint_predint(x, y, F...; α = 0.05, kwargs...)
    n = length(x)
    r = length(F)
    X = [fⱼ(xᵢ) for xᵢ in x, fⱼ in F] # design matrix
    β̂ = X \ y # equivalent to β̂ = X*(X'X)\X'y
    ŷ = X * β̂ # orthogonal projection of y onto X𝕣
    ŝ = norm(y - ŷ)/√(n - r)  # ŝ² is the unbiased estimator of σ²
    @show n r α
    m = @show quantile(TDist(n-r), 1-α/2)
    @show β̂ ŝ

    f̂(xstar) = sum(β̂ⱼ * fⱼ(xstar) for (β̂ⱼ, fⱼ) in zip(β̂, F))
    f(xstar) = [fⱼ(xstar) for fⱼ in F]
    invXX = inv(X'X)
    g(xstar) = ŝ * √(    f(xstar)' * invXX * f(xstar)) # for CI
    h(xstar) = ŝ * √(1 + f(xstar)' * invXX * f(xstar)) # for PI

    a, b = extrema(x)
    a, b = a - 0.05(b-a), b + 0.05(b-a)
    scatter(x, y; label="data", msc=:auto, c=1)
    plot!(xstar -> f̂(xstar), a, b; label="", c=:red)
    plot!(xstar -> f̂(xstar) - m*g(xstar), a, b;
        label="$(100(1-α))% CI", c=2, ls=:dash)
    plot!(xstar -> f̂(xstar) + m*g(xstar), a, b;
        label="", c=2, ls=:dash)
    plot!(xstar -> f̂(xstar) - m*h(xstar), a, b;
        label="$(100(1-α))% PI", c=3, ls=:dashdot)
    plot!(xstar -> f̂(xstar) + m*h(xstar), a, b;
        label="", c=3, ls=:dashdot)
    plot!(size=(520, 250), legend=:outertopright)
    plot!(; kwargs...)
end
```

Out[10]: plot_linreg_confint_predint (generic function with 1 method)

3.1.1 信頼区間と予測区間のテストプロット

In [11]:

```julia
Random.seed!(464937345105963)
x = 0:20
e = rand(Normal(0, 5), length(x))
y = @. 1 + x + e
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:1)...)
```

```
n = 21
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.093024054408309
β̂ = [-2.4436068097922625, 1.2636653169073329]
ŝ = 6.1883934314865146
```

Out[11]:

In [12]:

```
@rput x y
reg = R"""reg = lm(y ~ x)"""
R"""
xstars = data.frame(x = seq(-1, 21, 0.02))
conf.interval = predict(reg, newdata = xstars, interval = 'confidence', level = 0.95)
pred.interval = predict(reg, newdata = xstars, interval = 'prediction', level = 0.95)

plot(x, y, ylim = c(-17, 37))
lines(xstars$x, conf.interval[, 1], col = 'red')
lines(xstars$x, conf.interval[, 2], col = 'orange', lty=2)
lines(xstars$x, conf.interval[, 3], col = 'orange', lty=2)
lines(xstars$x, pred.interval[, 2], col = 'darkgreen', lty=4)
lines(xstars$x, pred.interval[, 3], col = 'darkgreen', lty=4)
"""
reg
```

Out[12]:

```
RObject{VecSxp}

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
     -2.444        1.264
```

以上によって, このノートのコードの結果とRによる計算結果のプロットは一致していることがわかった.

3.1.2 回帰直線の信頼区間と予測区間

In [13]:

```
Random.seed!(4649373)
n = 40
β₀, β₁ = β = [-1, 2]
@show β
f_true(x) = β₀ + β₁*x
x = rand(Normal(1, 1), n)
y = f_true.(x) + rand(Normal(0, 2), n)

scatter(x, y; label="data", legend=:topleft, msc=:auto)
plot!(f_true, extrema(x)...; label="y = f_true(x)", c=:blue)
plot!(size=(520, 250), legend=:outertopright)
plot!(ylim=(-8, 9))
```

```
β = [-1, 2]
```

Out[13]:

In [14]:

```
# 0次函数による回帰函数の推定
# 実質的に xᵢ 達を無視して, yᵢ 達だけを使った平均の推定になる.

@show β
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:0)...)
plot!(ylim=(-8, 9))
```

```
β = [-1, 2]
n = 40
r = 1
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0226909200367604
β̂ = [0.7224983356350365]
ŝ = 2.745840198360994
```

Out[14]:

In [15]:

```
# 回帰直線の推定

@show β
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:1)...)
plot!(ylim=(-8, 9))
```

```
β = [-1, 2]
n = 40
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0243941639119694
β̂ = [-0.8651568001715194, 1.7549089886753282]
ŝ = 2.222022308287274
```

Out[15]:

3.1.3 多項式回帰の信頼区間と予測区間 (オーバーフィッティングの例)

In [16]:

```
Random.seed!(4649373)

n = 30
f_true(x) = sin(x)
x = range(-π, π, n)
y = f_true.(x) + rand(Normal(0, 0.3), n)
scatter(x, y; label="data", legend=:topleft, msc=:auto)
plot!(f_true, extrema(x)...; label="y = f_true(x)", c=:blue)
plot!(size=(520, 250), legend=:outertopright)
plot!(ylim=(-2.5, 2.5))
```

Out[16]:

In [17]:

```
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:0)...)
plot!(ylim=(-2.5, 2.5))
```

```
n = 30
r = 1
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0452296421327034
β̂ = [-0.012921569974331694]
ŝ = 0.7781229227974805
```

Out[17]:

In [18]:

```
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:1)...)
plot!(ylim=(-2.5, 2.5))
```

```
n = 30
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0484071417952445
β̂ = [-0.012921569974331675, 0.25146544548096655]
ŝ = 0.6235650870768796
```

Out[18]:

In [19]:
```
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:3)...)
plot!(ylim=(-2.5, 2.5))
```

```
n = 30
r = 4
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0555294386428726
β̂ = [-0.004868037850388715, 0.9055404522516881, -0.0022900460140340695, -0.10348024677024828]
ŝ = 0.3018043730527076
```

Out[19]:

In [20]:
```
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:5)...)
plot!(ylim=(-2.5, 2.5))
```

```
n = 30
r = 6
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0638985616280254
β̂ = [-0.026428141042331734, 1.0071582696511716, 0.0182525867063003, -0.14879998365691066, -0.002
2800938849420445, 0.003892016571707278]
ŝ = 0.3078837919499401
```

Out[20]:

In [21]:

```
plot_linreg_confint_predint(x, y, (x -> x^k for k in 0:19)...)
plot!(ylim=(-2.5, 2.5))
```

```
n = 30
r = 20
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.228138851986274
β̂ = [0.14247028140681808, 1.2914469265448518, -2.862708649041329, -3.1780625037772205, 7.5180553
80487802, 7.790792095545583, -8.165629013870868, -8.17395305360277, 4.666047726772347, 4.3453609
37853182, -1.5269201557827485, -1.3012484041405847, 0.2944652989355421, 0.22929508226073686, -0.
03296579977722806, -0.023587239919145065, 0.001976582640073381, 0.0013103573390256058, -4.899517
106916385e-5, -3.036313618793969e-5]
ŝ = 0.24567191751997788
```

Out[21]:

サンプルサイズ $n = 30$ のときに, $r - 1 = 19$ 次の多項式函数で推定された回帰函数はぐちゃぐちゃになってしまっており, 左右の端の方で信頼区間と予測区間の幅が爆発している.

これは **過剰適合** (オーバーフィッティング, overfitting)の典型例になっている.

パラメータが多いモデルで推定を行うと過剰適合が起こり易い.

この問題に対処するための方法に, パラメータの動き方に制限を付ける正則化やモデルの適切な複雑さを知るための赤池情報量規準 (AIC)やベイズ統計でのWAICなどがある. 以下はそれらの検索のためのリンクである.

- 正則化 回帰 (https://www.google.com/search?q=%E6%AD%A3%E5%89%87%E5%8C%96+%E5%9B%9E%E5%B8%B0%E3%83%A2%E3%83%87%E3%83%AB)
- 赤池情報量規準 (https://www.google.com/search?q=%E8%B5%A4%E6%B1%A0%E6%83%85%E5%A0%B1%E9%87%8F%E8%A6%8F%E6%BA%96)
- WAIC ベイズ (https://www.google.com/search?q=WAIC+%E3%83%99%E3%82%A4%E3%82%BA)

3.2 信頼区間と予測区間に対応するP値函数のプロット

In [22]:

```julia
function plot_linreg_pvalue_functions(x, y, F...;
        α = 0.05, xstars=nothing, ystars=nothing, kwargs...)
    n = length(x)
    r = length(F)
    X = [fⱼ(xᵢ) for xᵢ in x, fⱼ in F] # design matrix
    β̂ = X \ y # equivalent to β̂ = X*(X'X)\X'y
    ŷ = X * β̂ # orthogonal projection of y onto X𝕣
    ŝ = norm(y - ŷ)/√(n - r) # ŝ² is the unbiased estimator of σ²
    @show n r α
    m = @show quantile(TDist(n - r), 1-α/2)
    @show β̂ ŝ

    f̂(xstar) = sum(β̂ⱼ * fⱼ(xstar) for (β̂ⱼ, fⱼ) in zip(β̂, F))
    f(xstar) = [fⱼ(xstar) for fⱼ in F]
    invXX = inv(X'X)
    g(xstar) = ŝ * √(    f(xstar)' * invXX * f(xstar)) # for CI
    h(xstar) = ŝ * √(1 + f(xstar)' * invXX * f(xstar)) # for PI
    tdist = TDist(n - r)
    G(xstar, ystar) = 2ccdf(tdist, abs(ystar - f̂(xstar))/g(xstar)) # pval of CI
    H(xstar, ystar) = 2ccdf(tdist, abs(ystar - f̂(xstar))/h(xstar)) # pval of PI

    if isnothing(xstars)
        a, b = extrema(x)
        a, b = a - 0.05(b-a), b + 0.05(b-a)
        xstars = range(a, b, 400)
    end
    a, b = extrema(xstars)
    if isnothing(ystars)
        c, d = extrema(y)
        c, d = c - 0.1m*(d-c), d + 0.1m*(d-c)
        ystars = range(c, d, 400)
    end
    c, d = extrema(ystars)

    P = plot(; legend=:topleft, colorbar=false)
    heatmap!(xstars, ystars, sqrt∘G)
    scatter!(x, y; label="", c=:cyan, msc=1)
    plot!(xstar -> f̂(xstar) - m*g(xstar), a, b;
        label="$(100(1-α))% CI", c=:orange, ls=:dash)
    plot!(xstar -> f̂(xstar) + m*g(xstar), a, b;
        label="", c=:orange, ls=:dash)
    plot!(; xlim=(a, b), ylim=(c, d))
    title!("√P-value functon of CI")

    Q = plot(; legend=:topleft, colorbar=false)
    heatmap!(xstars, ystars, sqrt∘H)
    scatter!(x, y; label="", c=:cyan, msc=1)
    plot!(xstar -> f̂(xstar) - m*h(xstar), a, b;
        label="$(100(1-α))% PI", c=:lightgreen, ls=:dash)
    plot!(xstar -> f̂(xstar) + m*h(xstar), a, b;
        label="", c=:lightgreen, ls=:dash)
    plot!(; xlim=(a, b), ylim=(c, d))
    title!("√P-value function of PI")

    plot(P, Q; size=(800, 300), layout=(1, 2), kwargs...)
end
```

Out[22]:

```
plot_linreg_pvalue_functions (generic function with 1 method)
```

3.2.1 信頼区間と予測区間に対応するP値函数のテストプロット

(回帰函数の値または回帰直線上の値の)信頼区間(confience interval)を CI と略し, 予測区間(prediction interval)を PI と略すことにする.

In [23]:

```
Random.seed!(464937345105963)
x = 0:20
e = rand(Normal(0, 5), length(x))
y = @. 1 + x + e
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:1)...;
    ystars=range(-17, 37, 400))
```

```
n = 21
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.093024054408309
β̂ = [-2.4436068097922625, 1.2636653169073329]
ŝ = 6.1883934314865146
```

Out[23]:

3.2.2 回帰直線の信頼区間と予測区間に対応するP値函数

In [24]:
```
Random.seed!(4649373)
n = 40
β₀, β₁ = β = [-1, 2]
@show β
f_true(x) = β₀ + β₁*x
x = rand(Normal(1, 1), n)
y = f_true.(x) + rand(Normal(0, 2), n)

plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:0)...;
    ystars=range(-8, 9, 400))
```

```
β = [-1, 2]
n = 40
r = 1
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0226909200367604
β̂ = [0.7224983356350365]
ŝ = 2.745840198360994
```

Out[24]:

In [25]:
```
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:1)...;
    ystars=range(-8, 9, 400))
```

```
n = 40
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0243941639119694
β̂ = [-0.8651568001715194, 1.7549089886753282]
ŝ = 2.222022308287274
```

Out[25]:

3.2.3 回帰直線の信頼区間に対応するP値函数の動画

In [26]:

```julia
function plot_linreg_confint_pvalfunc_3d(x, y, F...;
        α = 0.05, camera=(60, 60), xstars=nothing, ystars=nothing, kwargs...)
    n = length(x)
    r = length(F)
    X = [fⱼ(xᵢ) for xᵢ in x, fⱼ in F] # design matrix
    β̂ = X \ y # equivalent to β̂ = X*(X'X)\X'y
    ŷ = X * β̂ # orthogonal projection of y onto X⟨R⟩ᵀ
    ŝ = norm(y - ŷ)/√(n - r) # ŝ² is the unbiased estimator of σ²
    tdist = TDist(n - r)
    m = quantile(tdist, 1 - α/2)

    f̂(xstar) = sum(β̂ⱼ * fⱼ(xstar) for (β̂ⱼ, fⱼ) in zip(β̂, F))
    f(xstar) = [fⱼ(xstar) for fⱼ in F]
    invXX = inv(X'X)
    g(xstar) = ŝ * √(f(xstar)' * invXX * f(xstar))
    G(xstar, ystar) = 2ccdf(tdist, abs(ystar - f̂(xstar))/g(xstar))

    if isnothing(xstars)
        a, b = extrema(x)
        a, b = a - 0.05(b-a), b + 0.05(b-a)
        xstars = range(a, b, 400)
    end
    a, b = extrema(xstars)
    if isnothing(ystars)
        c, d = extrema(y)
        c, d = c - 0.1m*(d-c), d + 0.1m*(d-c)
        ystars = range(c, d, 400)
    end
    c, d = extrema(ystars)

    P = plot(; colorbar=false)
    surface!(xstars, ystars, G; camera, alpha=0.9)
    scatter3d!(x, y, zeros(length(x)); label="", c=:cyan, msc=:auto)
    plot!(size=(600, 500))
    plot!(; kwargs...)
end

Random.seed!(4649373)
n = 40
β₀, β₁ = β = [-1, 2]
@show β
f_true(x) = β₀ + β₁*x
x = rand(Normal(1, 1), n)
y = f_true.(x) + rand(Normal(0, 2), n)

plot_linreg_confint_pvalfunc_3d(x, y, (x -> x^k for k in 0:1)...)
```

```
β = [-1, 2]
```

Out[26]:

In [27]:

```
anim = @animate for t in 0:5:359
    plot_linreg_confint_pvalfunc_3d(x, y, (x→x^k for k in 0:1)...; camera=(t, 60))
end
gif(anim, "images/anim_linreg_confint_pvalfunc_3d.gif")
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\Statistics\2022\images\anim_linreg_confint_pvalfunc_3d.gif
@ Plots D:\.julia\packages\Plots\MzlNY\src\animation.jl:130
```

Out[27]:

PDFファイルではこのアニメーションは動かない. 動いている様子を見たい人は以下のリンク先を参照せよ.

- [線形回帰の信頼区間に対応するP値函数 (https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_linreg_confint_pvalfunc_3d.gif)](https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_linreg_confint_pvalfunc_3d.gif)

3.2.4 多項式回帰の信頼区間と予測区間に対応するP値函数

In [28]:

```
Random.seed!(4649373)

n = 30
f_true(x) = sin(x)
x = range(-π, π, n)
y = f_true.(x) + rand(Normal(0, 0.3), n)
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:0)...;
    ystars=range(-2.5, 2.5, 400))
```

```
n = 30
r = 1
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0452296421327034
β̂ = [-0.012921569974331694]
ŝ = 0.7781229227974805
```

Out[28]:

In [29]:

```
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:1)...;
    ystars=range(-2.5, 2.5, 400))
```

```
n = 30
r = 2
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0484071417952445
β̂ = [-0.012921569974331675, 0.25146544548096655]
ŝ = 0.6235650870768796
```

Out[29]:

In [30]:
```
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:3)...;
    ystars=range(-2.5, 2.5, 400))
```

```
n = 30
r = 4
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0555294386428726
β̂ = [-0.004868037850388715, 0.9055404522516881, -0.0022900460140340695, -0.10348024677024828]
ŝ = 0.3018043730527076
```

Out[30]:

In [31]:
```
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:5)...;
    ystars=range(-2.5, 2.5, 400))
```

```
n = 30
r = 6
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.0638985616280254
β̂ = [-0.026428141042331734, 1.0071582696511716, 0.0182525867063003, -0.14879998365691066, -0.002
2800938849420445, 0.003892016571707278]
ŝ = 0.3078837919499401
```

Out[31]:

In [32]:

```
plot_linreg_pvalue_functions(x, y, (x -> x^k for k in 0:19)...;
    ystars=range(-2.5, 2.5, 400))
```

```
n = 30
r = 20
α = 0.05
quantile(TDist(n - r), 1 - α / 2) = 2.228138851986274
β̂ = [0.14247028140681808, 1.2914469265448518, -2.862708649041329, -3.1780625037772205, 7.5180553
80487802, 7.790792095545583, -8.165629013870868, -8.17395305360277, 4.666047726772347, 4.3453609
37853182, -1.5269201557827485, -1.3012484041405847, 0.2944652989355421, 0.22929508226073686, -0.
03296579977722806, -0.023587239919145065, 0.001976582640073381, 0.0013103573390256058, -4.899517
106916385e-5, -3.036313618793969e-5]
ŝ = 0.24567191751997788
```

Out[32]:

4 ロジスティック回帰

In [33]:

```julia
"""ロジスティック回帰のunlink函数"""
p_(xᵢ, β) = logistic(β[1] + β[2]*xᵢ)

"""ロジスティック回帰のunlink函数の導函数"""
function dp_(xᵢ, β)
    pᵢ = logistic(β[1] + β[2]*xᵢ)
    pᵢ * (1 - pᵢ)
end

"""ロジスティック回帰モデルの尤度函数の-1倍"""
function logistic_negloglik(y, x, β)
    -sum(logpdf(Bernoulli(p_(xᵢ, β)), yᵢ) for (xᵢ, yᵢ) in zip(x, y))
end

"""ロジスティック回帰のスコア統計量A"""
function score_A(y, x, β)
    sum(p_(xᵢ, β) - yᵢ for (xᵢ, yᵢ) in zip(x, y))
end

"""ロジスティック回帰のスコア統計量B"""
function score_B(y, x, β)
    sum(xᵢ * (p_(xᵢ, β) - yᵢ) for (xᵢ, yᵢ) in zip(x, y))
end

"""ロジスティック回帰のフィッシャー情報量行列の成分a"""
function fisherinfo_a(x, β)
    sum(dp_(xᵢ, β) for xᵢ in x)
end

"""ロジスティック回帰のフィッシャー情報量行列の成分b"""
function fisherinfo_b(x, β)
    sum(xᵢ * dp_(xᵢ, β) for xᵢ in x)
end

"""ロジスティック回帰のフィッシャー情報量行列の成分c"""
function fisherinfo_c(x, β)
    sum(xᵢ^2 * dp_(xᵢ, β) for xᵢ in x)
end

"""ロジスティック回帰のフィッシャー情報量行列のすべての成分"""
function fisherinfo(x, β)
    a = fisherinfo_a(x, β)
    b = fisherinfo_b(x, β)
    c = fisherinfo_c(x, β)
    a, b, c
end

"""ロジスティック回帰モデルの最尤法"""
function logistic_mle(y, x; alg=LBFGS())
    f(β) = logistic_negloglik(y, x, β)
    o = optimize(f, MVector(0.0, 0.0), alg)
    β̂ = o.minimizer
end

"""ロジスティック回帰モデルの乱数"""
function logistic_rand(x, β)
    rand.(Bernoulli.(p_.(x, Ref(β))))
end
```

Out[33]: logistic_rand

In [34]:

```julia
"""β̂とSEhatの計算"""
function betahat_sehat_star(y, x, xstar; alg=LBFGS())
    β̂ = logistic_mle(y, x; alg)
    â, b̂, ĉ = fisherinfo(x, β̂)
    SEhat = √max(0, safediv(ĉ - 2b̂*xstar + â*xstar^2, â*ĉ - b̂^2))
    β̂, SEhat
end

"""β̂とSEhatとẑの計算"""
function betahat_sehat_zhat_star(y, x, xstar, tstar; alg=LBFGS())
    β̂, SEhat = betahat_sehat_star(y, x, xstar; alg)
    ẑ = safediv(β̂[1] + β̂[2]*xstar - tstar, SEhat)
    β̂, SEhat, ẑ
end

"""ロジスティック回帰のMonte Carloシミュレーション(xstar版)"""
function sim_logistic_regression_star(;
        β=[-4, 2], xstar=1, n=200,
        x=range(-β[1]/β[2]-2, -β[1]/β[2]+2, n),
        L=10^4, alg=LBFGS()
    )
    betahat = Vector{MVector{2, Float64}}(undef, L)
    sehat = Vector{Float64}(undef, L)
    zhat = Vector{Float64}(undef, L)
    tstar = β[1] + β[2]*xstar
    @threads for i in 1:L
        y = logistic_rand(x, β)
        β̂, SEhat, ẑ = betahat_sehat_zhat_star(y, x, xstar, tstar; alg)
        betahat[i] = β̂
        sehat[i] = SEhat
        zhat[i] = ẑ
    end
    betahat, sehat, zhat
end
```

Out[34]: sim_logistic_regression_star

In [35]:

```julia
"""tstarに関するWald型P値函数"""
function pvalue_tstar_wald(y, x, xstar, tstar; alg=LBFGS())
    ẑ = betahat_sehat_zhat_star(y, x, xstar, tstar; alg)[3]
    2ccdf(Normal(), abs(ẑ))
end

"""pstarに関するWald型P値函数"""
function pvalue_pstar_wald(y, x, xstar, pstar; alg=LBFGS())
    tstar = logit(pstar)
    pvalue_tstar_wald(y, x, xstar, tstar; alg)
end

"""tstarに関する信頼区間"""
function confint_tstar_wald(y, x, xstar; α = 0.05, alg=LBFGS())
    z = quantile(Normal(), 1-α/2)
    β̂, SEhat = betahat_sehat_star(y, x, xstar; alg)
    m = β̂[1] + β̂[2]*xstar
    m - z*SEhat, m + z*SEhat
end

"""pstarに関する信頼区間"""
function confint_pstar_wald(y, x, xstar; α = 0.05, alg=LBFGS())
    tL, tU = confint_tstar_wald(y, x, xstar; α, alg)
    logistic(tL), logistic(tU)
end
```

Out[35]: confint_pstar_wald

In [36]:
```julia
function betahat_sehat_beta1(y, x; alg=LBFGS())
    β̂ = logistic_mle(y, x; alg)
    â, b̂, ĉ = fisherinfo(x, β̂)
    SEhat = √max(0, safediv(â, â*ĉ - b̂^2))
    β̂, SEhat
end

function betahat_sehat_zhat_beta1(y, x, β₁; alg=LBFGS())
    β̂, SEhat = betahat_sehat_beta1(y, x; alg)
    β̂₁ = β̂[2]
    zhat = safediv(β̂₁ - β₁, SEhat)
    β̂, SEhat, zhat
end

"""ロジスティック回帰のMonte Carloシミュレーション(β₁版)"""
function sim_logistic_regression_beta1(;
        β=[-4, 2], n=200,
        x=range(-β[1]/β[2]-2, -β[1]/β[2]+2, n),
        L=10^4, alg=LBFGS()
    )
    betahat = Vector{MVector{2, Float64}}(undef, L)
    sehat = Vector{Float64}(undef, L)
    zhat = Vector{Float64}(undef, L)
    @threads for i in 1:L
        y = logistic_rand(x, β)
        β̂, SEhat, ẑ = betahat_sehat_zhat_beta1(y, x, β[2]; alg)
        betahat[i] = β̂
        sehat[i] = SEhat
        zhat[i] = ẑ
    end
    betahat, sehat, zhat
end
```

Out[36]: sim_logistic_regression_beta1

In [37]:
```julia
"""β₁のWald型のP値"""
function pvalue_beta1_wald(y, x, β₁; alg=LBFGS())
    zhat = betahat_sehat_zhat_beta1(y, x, β₁; alg)[3]
    2ccdf(Normal(), abs(zhat))
end

"""β₁のWald型信頼区間"""
function confint_beta1_wald(y, x; α=0.05, alg=LBFGS())
    z = quantile(Normal(), 1-α/2)
    β̂, SEhat = betahat_sehat_beta1(y, x; alg)
    β̂₁ = β̂[2]
    [β̂₁ - z*SEhat, β̂₁ + z*SEhat]
end
```

Out[37]: confint_beta1_wald

4.1 ロジスティック函数とロジット函数

ロジスティック函数 を次のように定める:

$$\mathrm{logistic}(t) = \frac{1}{1+e^{-t}} \quad (t \in \mathbb{R})$$

ロジスティック函数 $f(t) = \mathrm{logistic}(t)$ は微分方程式

$$f'(t) = f(t)(1-f(t)), \quad f(0) = \frac{1}{2}$$

を満たしており(自分で確認せよ), この条件を満たす函数 $f(t)$ はロジスティック函数になる.

ロジスティック函数の微分が出て来る計算は上の微分方程式を使うと楽にできる.

ロジスティック函数は狭義単調増加函数で, $t \to -\infty$ のとき 0 に収束し, $t \to \infty$ で 1 に収束する.

ロジスティック函数 $p = \mathrm{logistic}(t)$ $(t \in \mathbb{R})$ の値域は $0 < p < 1$ になる.

ロジスティック函数 $p = \mathrm{logistic}(t)$ の逆函数は **ロジット函数** と呼ばれている. ロジット函数は対数オッズ函数であり, 次のように表される:

$$\mathrm{logit}(p) = \log \frac{p}{1-p} \quad (0 < t < 1).$$

In [38]:

```
P1 = plot(logistic, -8, 8;
    label="", title = "p = logistic(t)",
    xguide="t", yguide="p", legend=:topleft,
    xtick=-10:2:10, ytick=0:0.1:1)
P2 = plot(logit, 0, 1;
    label="", title="t = logit(p)",
    ytick=-10:2:10, xtick=0:0.1:1,
    xguide="p", yguide="t", legend=:topleft,
    ylim=(-8.4, 8.4), bottommargin=4Plots.mm)
plot(P1, P2; size=(600, 300), tickfontsize=7)
```

Out[38]:

4.2 ロジスティック回帰のデータの形

データは以下の形で得られると仮定する:

$$(x_1, y_1), \ldots, (x_n, y_n) \in \mathbb{R} \times \{1, 0\}.$$

x_i 達は実数で y_i 達は 1 または 0 であるとする.

このデータは各 x_i ごとに x_i の値に応じである確率で y_i の値がランダムに 1 または 0 の値になるのように解釈できると仮定する.

この形のデータを2つのベクトルで表す:

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{R}^n, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \in \{1, 0\}^n.$$

4.3 ロジスティック回帰でのリンク函数

ベクトル値の回帰係数パラメータ

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \in \mathbb{R}^2$$

を考え, 確率パラメータ達 $0 < p_i < 1$ が次のように表されていると仮定する:

$$p_i = \mathrm{logistic}(\beta_0 + \beta_1 x_i).$$

すなわち,

$$\mathrm{logit}(p_i) = \log \frac{p_i}{1-p_i} = \beta_0 + \beta_1 x_i.$$

このときロジット函数を **リンク函数** (link function)と呼び, ロジスティック函数を **アンリンク函数** (unlink function)と呼ぶことがある.

注意: p_i 達のリンク函数として対数函数を使った場合には

$$\log p_i = \beta_0 + \beta_1 x_i$$

による回帰を考え, リンク函数として恒等函数を使った場合には

$$p_i = \beta_0 + \beta_1 x_i$$

による回帰を考えることになる. どれを使うのが適切であるかは目的によって変わる. **注意終**

4.4 ロジスティック回帰の統計モデル

前節までの記号の下で, 統計モデルとして $y \in \{1, 0\}^n$ に関する以下の確率質量函数を採用する:

$$P(y|x, \beta) = \prod_{i=1}^{n} \left(p_i^{y_i}(1-p_i)^{1-y_i}\right) = \prod_{i=1}^{n} \left(\mathrm{logistic}(\beta_0 + \beta_1 x_i)^{y_i}(1 - \mathrm{logistic}(\beta_0 + \beta_1 x_i))^{1-y_i}\right)$$

このモデルの確率分布を $\mathrm{LogisticModel}(x, \beta)$ と書くと,

$$\mathrm{LogisticModel}(x, \beta) = \prod_{i=1}^{n} \mathrm{Bernoulli}(\mathrm{logistic}(\beta_0 + \beta_1 x_i)).$$

これは最も簡単な場合でもっと複雑なモデルを考えることもできる. (例えば $\beta_0 + \beta_1 x_i$ の部分をもっと複雑にできる).

$Y = [Y_i]_{i=1}^n$ をこの統計モデルに従う確率変数であるとする.

このとき, 各 Y_i の値は次のようにしてランダムに決まると考えることができる:

(1) x_i の値に対する確率パラメータ p_i の値が $p_i = \mathrm{logistic}(\beta_0 + \beta_1 x_i)$ で決まる.

(2) Y_i の値はランダムに決まり, 確率 p_i で 1 になり, 確率 $1 - p_i$ で 0 になる.

x_i 達から $Y_i = 1, 0$ 達の値を確率的に決定する確率パラメータ達を

$$p_i = \mathrm{logistic}(\beta_0 + \beta_1 x_i) \quad \left(\iff \mathrm{logit}(p_i) = \log \frac{p_i}{1-p_i} = \beta_0 + \beta_1 x_i\right)$$

の形式で推定することが, **ロジスティック回帰** と呼ばれている.

In [39]:

```
# ロジスティック回帰モデルのサンプル(データ)の生成
#Random.seed!(4649373)
β₀, β₁ = β = [-4, 2]
n = 51
x = range(-β₀/β₁-2, -β₀/β₁+2, n)
#y = logistic_rand(x, β)
y = Bool[
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
    1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1,
    1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1]

# モデルとデータの視覚化
scatter(x, y; label="sample", msc=:auto, alpha=0.7)
plot!(xstar -> logistic(β[1] + β[2]*xstar);
    label="p = logistic($(β₀)+$(β₁)x)", c=:blue, ls=:dot)
plot!(ytick=0:0.1:1, yguide="y, p", xguide="x")
title!("size-$n sample of logistic model with β = $β")
plot!(legend=:outertopright, size=(800, 250),
    leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[39]:

4.5 最尤法

データの数値 $x \in \mathbb{R}^n$, $y \in \{1,0\}^n$ に対して, 前節で定義した統計モデルの尤度函数の対数の -1 倍 $L = L(\beta)$ は次の形になる:

$$L = L(\beta) = -\log p(y|x,\beta) = -\sum_{i=1}^{n} \left(y_i \log p_i + (1-y_i)\log(1-p_i)\right).$$

ここで,

$$p_i = \mathrm{logistic}(\beta_0 + \beta_1 x_i) \quad (i = 1, \ldots, n).$$

対数尤度函数の -1 倍 $L(\beta)$ を最小化する β を $\hat{\beta} = [\hat{\beta}_0, \hat{\beta}_1]^T$ と書き, **最尤法の解** (さいゆうほうの解)と呼ぶことにする.

特別な場合を除いて, 最尤法の計算はコンピュータによる最適化の数値計算によって行うことになる. (その手法も色々ある.)

In [40]:

```
# ロジスティック回帰の実行例
# 上の方で定義されている logistic_mle(y, x) 函数の定義の中で,
# Optim.jl パッケージの optimize 函数が使われている.
β̂₀, β̂₁ = β̂ = logistic_mle(y, x)
@show β₀, β₁
@show β̂₀, β̂₁

# ロジスティック回帰の実行結果の視覚化
scatter(x, y; label="sample", msc=:auto, alpha=0.7)
plot!(xstar -> logistic(β[1] + β[2]*xstar);
    label="p = logistic($(β₀)+$(β₁)x)", c=:blue, ls=:dot)
plot!(xstar -> logistic(β̂₀ + β̂₁*xstar); label="MLE", c=2)
plot!(ytick=0:0.1:1, yguide="y, p", xguide="x")
title!("size-$n sample of logistic model with β = $β")
plot!(legend=:outertopright, size=(800, 250))
```

```
(β₀, β₁) = (-4, 2)
(β̂₀, β̂₁) = (-3.3771836845450944, 1.5218694635151835)
```

Out[40]:

4.6 スコア統計量とFisher情報量行列

この節では $y = [y_i]_{i=1}^n$ はロジスティック回帰の統計モデルに従う確率変数であるとする. (「ロジスティック回帰の統計モデル」の節ではこれを Y と書いていたが, 前節の記号をそのまま使いたいので, $y = [y_i]_{i=1}^n$ と書くことにする. 一種の手抜き.)

以下では, ロジスティック函数が満たす微分方程式から得られる次の公式を自由に用いる:

$$\frac{\partial p_i}{\partial \beta_0} = p_i(1-p_i), \quad \frac{\partial p_i}{\partial \beta_1} = x_i p_i (1-p_i).$$

スコア統計量 $A = A(\beta)$, $B = B(\beta)$ を次のように定める:

$$A = A(\beta) = \frac{\partial L}{\partial \beta_0} = -\sum_{i=1}^{n}(y_i(1-p_i) - (1-y_i)p_i) = \sum_{i=1}^{n}(p_i - y_i),$$
$$B = B(\beta) = \frac{\partial L}{\partial \beta_1} = -\sum_{i=1}^{n}(x_i y_i(1-p_i) - x_i(1-y_i)p_i) = \sum_{i=1}^{n} x_i(p_i - y_i).$$

β は $p_i = \mathrm{logistic}(\beta_0 + \beta_1 x_i)$ の中に含まれている.

y_i は確率 p_i で 1 になり, 確率 $1-p_i$ で 0 になるので, $E[A] = E[B] = 0$ となる.

最尤法の解 $\hat{\beta}$ は $L = L(\beta)$ を最小化するので,

$$A(\hat{\beta}) = B(\hat{\beta}) = 0$$

を満たしている. ベクトル値確率変数 $U = U(\beta)$ を次のように定める:

$$U = U(\beta) = \begin{bmatrix} A(\beta) \\ B(\beta) \end{bmatrix}$$

この U をも **スコア統計量** と呼ぶことにする.

Fisher情報量行列 の成分 $a = a(\beta)$, $b = b(\beta)$, $c = c(\beta)$ を以下のように定める:

$$\begin{aligned} a = a(\beta) &= \frac{\partial^2 L}{\partial \beta_0^2} = \sum_{i=1}^n p_i(1-p_i), \\ b = b(\beta) &= \frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} = \sum_{i=1}^n x_i p_i(1-p_i), \\ c = c(\beta) &= \frac{\partial^2 L}{\partial \beta_1^2} = \sum_{i=1}^n x_i^2 p_i(1-p_i). \end{aligned}$$

$E[(y_i - p_i)^2] = E[y_i^2] - E[y_i]^2 = E[y_i] - E[y_i]^2 = p_i - p_i^2 = p_i(1-p_i)$ を使った直接的な計算によって, a, c はそれぞれスコア統計量 A, B の分散に等しく, b はスコア統計量 A, B の共分散に等しいことを示せる(実はこのようなことは一般的に成立している):

$$a = E[A^2], \quad b = E[AB], \quad c = E[B^2].$$

Fisher情報量行列 $I(\beta)$ を次のように定める:

$$I(\beta) = \begin{bmatrix} a(\beta) & b(\beta) \\ b(\beta) & c(\beta) \end{bmatrix}.$$

Fisher情報量行列 $I(\beta)$ はスコア統計量 $U(\beta) = [A(\beta), B(\beta)]^T$ の分散共分散行列になっている.

注意: Fisher情報量行列を $I(\beta)$ と単位行列を混同しないこと!

注意: 「Fisher情報量行列」のような何か非常に高級で難しそうな名前が付いているという理由で「これは難しいものだ」と判断してはいけない. 上の場合には, その成分は対数尤度函数の -1 倍 $L = L(\beta)$ を2回偏微分しただけのものに過ぎず, $L = L(\beta)$ を1回偏微分して得られるスコア統計量の分散共分散行列を与える量になっている行列であるに過ぎない. 付けられた名前の権威に負けて恐れを持ってはいけない. 恐れを持たずに数学的な内容だけを見ればよい.

注意: Fisher情報量行列の一般的な場合の定義では, 成分を定義するときには2回偏微分した後に期待値を取る. 上でそうなっていない理由は期待値を取る前の2階の偏微分の段階ですでに定数になってしまっているからである.

4.7 問題: 一般の場合のスコア統計量とFisher情報量行列

一般のパラメータ $\theta = (\theta_1, \ldots, \theta_d)$ を持つ統計モデル $p(y|\theta)$ とそれに従う確率変数 y が与えられていて, y の対数尤度函数の -1 倍を $L(\theta) = -\log p(y|\theta)$ と書くとき(これは確率変数 y に依存するので確率変数になる), **スコア統計量** $U = [U_i]_{i=1}^n$ が

$$U_i = \frac{\partial}{\partial \theta_i} L(\theta)$$

と定義され, **Fisher情報量行列** $I(\theta)$ の (i,j) 成分 $I_{ij}(\theta)$ が

$$I_{ij}(\theta) = E\left[\frac{\partial^2}{\partial \theta_i \partial \theta_j} L(\theta)\right]$$

と定義される. ここで $E[\,]$ は確率変数 y に関する期待値を取る操作である. 次が成立することを示せ:

$$E[U_i] = 0, \quad E[U_i U_j] = I_{ij}(\theta).$$

解答例: $E[U_i] = 0$ を示そう. $p(y|\theta)$ の θ_i に関する偏導函数を $p_{\theta_i}(y|\theta)$ と書くと,

$$\begin{aligned} E[U_i] = E\left[\frac{\partial}{\partial \theta_i} L(\theta)\right] &= -E\left[\frac{\partial}{\partial \theta_i} \log p(y|\theta)\right] = -E\left[\frac{p_{\theta_i}(y|\theta)}{p(y|\theta)}\right] \\ &= -\int \frac{p_{\theta_i}(y|\theta)}{p(y|\theta)}\, p(y|\theta)\, dy = -\int p_{\theta_i}(y|\theta)\, dy \\ &= -\frac{\partial}{\partial \theta_i} \int p(y|\theta)\, dy = -\frac{\partial}{\partial \theta_i} 1 = 0. \end{aligned}$$

$I_{ij}(\theta) = E[U_i U_j]$ を示そう. $p(y|\theta)$ の2階の偏導函数を $p_{\theta_i\theta_j}(y|\theta)$ と書くと, 上と同様にして,

$$E\left[\frac{p_{\theta_i\theta_j}(y|\theta)}{p(y|\theta)}\right] = \int p_{\theta_i\theta_j}(y|\theta)\,dy = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\int p(y|\theta)\,dy = 0.$$

これを使うと,

$$\begin{aligned}
I_{ij}(\theta) &= E\left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}L(\theta)\right] = -E\left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}\log p(y|\theta)\right] \\
&= -E\left[\frac{p_{\theta_i\theta_j}(y|\theta)}{p(y|\theta)} - \frac{p_{\theta_i}(y|\theta)p_{\theta_j}(y|\theta)}{p(y|\theta)^2}\right] \\
&= E\left[\frac{p_{\theta_i}(y|\theta)}{p(y|\theta)^2}\frac{p_{\theta_j}(y|\theta)}{p(y|\theta)^2}\right] = E[U_i U_j].
\end{aligned}$$

解答終

注意: 前節の最後の注意で「Fisher情報量行列という名前に権威を感じて恐れを持つな」と述べたが, ここまで来れた人は数理統計学の面白い部分まで十分に到達できていると考えられるので, 自分自身を誇って良いと思われる.

4.8 βの最尤推定量の分布の正規分布近似

スコア統計量 $U(\beta) = [A(\beta), B(\beta)]^T$ は次のように多変量正規分布に近似的に従っていると仮定する:

$$U(\beta) = \begin{bmatrix} A(\beta) \\ B(\beta) \end{bmatrix} \sim \mathrm{MvNormal}(0, I(\beta)),\ \text{approximately}.$$

最尤法の解で得られる β の推定量 $\hat{\beta}$ を使って, 統計量 $\hat{a}, \hat{b}, \hat{c}$ を次のように定める:

$$\hat{a} = a(\hat{\beta}), \quad \hat{b} = b(\hat{\beta}), \quad \hat{c} = c(\hat{\beta}).$$

このとき, $A(\hat{\beta}) = B(\hat{\beta}) = 0$ でかつ

$$a = \frac{\partial A}{\partial\beta_0}, \quad b = \frac{\partial A}{\partial\beta_1} = \frac{\partial B}{\partial\beta_0}, \quad c = \frac{\partial B}{\partial\beta_1}$$

なので, 次の一次近似が成立している(1次までのTaylor展開):

$$\begin{aligned}
A(\beta) &\approx A(\hat{\beta}) + \frac{\partial A}{\partial\beta_0}(\hat{\beta})(\beta_0 - \hat{\beta}_0) + \frac{\partial A}{\partial\beta_1}(\hat{\beta})(\beta_1 - \hat{\beta}_1) = \hat{a}(\beta_0 - \hat{\beta}_0) + \hat{b}(\beta_1 - \hat{\beta}_1), \\
B(\beta) &\approx B(\hat{\beta}) + \frac{\partial B}{\partial\beta_0}(\hat{\beta})(\beta_0 - \hat{\beta}_0) + \frac{\partial B}{\partial\beta_1}(\hat{\beta})(\beta_1 - \hat{\beta}_1) = \hat{b}(\beta_0 - \hat{\beta}_0) + \hat{c}(\beta_1 - \hat{\beta}_1).
\end{aligned}$$

すなわち,

$$\begin{bmatrix} A \\ B \end{bmatrix} \approx \begin{bmatrix} \hat{a} & \hat{b} \\ \hat{b} & \hat{c} \end{bmatrix} \begin{bmatrix} \beta_0 - \hat{\beta}_0 \\ \beta_1 - \hat{\beta}_1 \end{bmatrix}.$$

ゆえに,

$$\begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} \approx \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} - \begin{bmatrix} \hat{a} & \hat{b} \\ \hat{b} & \hat{c} \end{bmatrix}^{-1} \begin{bmatrix} A \\ B \end{bmatrix}.$$

これより,

$$\hat{\beta} \sim \mathrm{MvNormal}(\beta,\ I(\hat{\beta})^{-1}I(\beta)I(\hat{\beta})^{-1}),\ \text{approximately}.$$

さらに次の近似も成立していると仮定する:

$$I(\hat{\beta}) \approx I(\beta).$$

この仮定のもとで

$$\hat{\beta} \sim \mathrm{MvNormal}(\beta,\ I(\hat{\beta})^{-1}),\ \text{approximately}.$$

In [41]:

```
# ẑの分布が標準正規分布で近似されることの確認
# β̂の分布の視覚化
PP, QQ = [], []
for n in (100, 400)
    @time betahat, sehat, zhat = sim_logistic_regression_star(; n)
    P = stephist(zhat; norm=true, label="zhat")
    plot!(Normal(), -5, 5; label="Normal(0,1)")
    title!("n = $n")
    push!(PP, P)
    Q = scatter(first.(betahat), last.(betahat);
        label="betahat", msc=:auto, alpha=0.5, ms=1)
    title!("n = $n")
    plot!(xlim=(-8, -1), ylim=(0.5, 4.5), xtick=-10:10)
    push!(QQ, Q)
end
plot(PP...; size=(800, 250), layout=(1, 2)) ▷ display
plot(QQ...; size=(800, 250), layout=(1, 2)) ▷ display
```

```
3.044214 seconds (6.43 M allocations: 316.230 MiB, 4.31% gc time, 74.25% compilation time)
3.537139 seconds (2.89 M allocations: 168.000 MiB)
```

4.9 ロジスティック回帰における β₀+β₁x に関するWald型のP値函数と信頼区間

一般にベクトル値確率変数 $V = [V_i]_{i=1}^n \sim \mathrm{MvNormal}(\mu, \Sigma)$ と $c \in \mathbb{R}^n$ について,

$$
\begin{aligned}
E[V^T c] &= \mu^T c, \\
\mathrm{var}(V^T c) &= E[(V^T c - \mu^T c)^2] = E[((V-\mu)^T c)^2] = E[((V-\mu)^T c)^T ((V-\mu)^T c)] \\
&= E[(c^T (V-\mu))(V-\mu)^T c] = c^T E[(V-\mu))(V-\mu)^T] c = c^T \Sigma c.
\end{aligned}
$$

$x_* \in \mathbb{R}$ を任意に取って, この結果を $V = \hat{\beta}$, $c = [1, x_*]^T$ と

$$
\Sigma = I(\beta)^{-1} = \frac{1}{ac - b^2} \begin{bmatrix} c & -b \\ -b & a \end{bmatrix}
$$

に適用すると,

$$
V^T c = \hat{\beta}_0 + \hat{\beta}_1 x_*, \quad c^T \Sigma c = \frac{c - 2bx_* + ax_*^2}{ac - b^2}
$$

なので,

$$\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*} = \sqrt{\frac{\hat{c} - 2\hat{b}x_* + \hat{a}x_*^2}{\hat{a}\hat{c} - \hat{b}^2}}$$

とおくと,

$$\hat{\beta}_0 + \hat{\beta}_1 x_* \sim \mathrm{Normal}\Big(\beta_0 + \beta_1 x_*,\ \widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*}\Big),\ \text{approximately}.$$

以上の結果を使うと,「x_* に対応する y_* の値が 1 になる確率が $p_* = \mathrm{logistic}(\beta_0 + \beta_1 x_*)$ である」という仮説のWald型のP値を次のように定めることができる:

$$\mathrm{pvalue}_{\mathrm{Wald}}(y|x, x_*, \beta) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), \hat{z}(x_*, \beta)).$$

ここで,

$$\hat{z}(x_*, \beta) = \frac{(\hat{\beta}_0 + \hat{\beta}_1 x_*) - (\beta_0 + \beta_1 x_*)}{\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*}}.$$

有意水準 $0 < \alpha < 1$ を任意に取り, $z_{\alpha/2}$ を次のように定める:

$$z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2).$$

Wald型P値函数には $t_* = \beta_0 + \beta_1 x_*$ に関する次の信頼区間が対応している:

$$\mathrm{confint}_{\mathrm{Wald}}^{\beta_0+\beta_1 x_*}(y|x, x_*, \alpha) = \left[\hat{\beta}_0 + \hat{\beta}_1 x_* - z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*},\ \hat{\beta}_0 + \hat{\beta}_1 x_* + z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*}\right].$$

これを $p_* = \mathrm{logistic}(\beta_0 + \beta_1 x_*)$ に関する信頼区間に書き直すと,

$$\begin{aligned}
&\mathrm{confint}_{\mathrm{Wald}}^{p_*}(y|x, x_*, \alpha) \\
&= \left[\mathrm{logistic}\Big(\hat{\beta}_0 + \hat{\beta}_1 x_* - z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*}\Big),\ \mathrm{logistic}\Big(\hat{\beta}_0 + \hat{\beta}_1 x_* + z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_0+\hat{\beta}_1 x_*}\Big)\right].
\end{aligned}$$

In [42]:

```
# ロジスティック回帰モデルのサンプル(データ)の生成
#Random.seed!(4649373)
β₀, β₁ = β = [-4, 2]
n = 51
x = range(-β₀/β₁-2, -β₀/β₁+2, n)
#y = logistic_rand(x, β)
y = Bool[
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
    1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1,
    1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1]

# ロジスティック回帰
β̂₀, β̂₁ = β̂ = logistic_mle(y, x)
@show β₀, β₁
@show β̂₀, β̂₁
println()

# P値の例
for pstar in 0.2:0.1:0.6
    @eval @show pvalue_pstar_wald(y, x, 2, $pstar)
end

# 95%信頼区間の視覚化
α = 0.05
scatter(x, (yᵢ -> yᵢ==0 ? -0.05 : 1.05).(y); label="sample", msc=:auto, alpha=0.7)
plot!(xstar -> logistic(β₀ + β₁*xstar);
    label="true model", c=:blue, ls=:dot)
plot!(xstar -> logistic(β̂₀ + β̂₁*xstar); label="MLE", c=2)
plot!(xstar -> confint_pstar_wald(y, x, xstar; α)[1];
    fillrange = xstar -> confint_pstar_wald(y, x, xstar; α)[2],
    label="$(100(1-α))% CI", c=:red, ls=:dash, fillalpha=0.2)
plot!(xstar -> confint_pstar_wald(y, x, xstar; α)[2];
    label="", c=:red, ls=:dash)
plot!(ytick=0:0.1:1, yguide="p", xguide="x")
title!("size-$n sample of logistic model with β = $β")
plot!(legend=:outertopright, size=(800, 250),
    leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

```
(β₀, β₁) = (-4, 2)
(β̂₀, β̂₁) = (-3.3771836845450944, 1.5218694635151835)

pvalue_pstar_wald(y, x, 2, 0.2) = 0.004572848447696335
pvalue_pstar_wald(y, x, 2, 0.3) = 0.1663647854267721
pvalue_pstar_wald(y, x, 2, 0.4) = 0.8461954258378921
pvalue_pstar_wald(y, x, 2, 0.5) = 0.3691434652302312
pvalue_pstar_wald(y, x, 2, 0.6) = 0.046576295012251176
```

Out[42]:

In [43]:

```
# 99%信頼区間の視覚化
α = 0.01
scatter(x, (yᵢ → yᵢ==0 ? -0.05 : 1.05).(y); label="sample", msc=:auto, alpha=0.7)
plot!(xstar → logistic(β₀ + β₁*xstar);
    label="true model", c=:blue, ls=:dot)
plot!(xstar → logistic(β̂₀ + β̂₁*xstar); label="MLE", c=2)
plot!(xstar → confint_pstar_wald(y, x, xstar; α)[1];
    fillrange = xstar → confint_pstar_wald(y, x, xstar; α)[2],
    label="$(100(1-α))% CI", c=:red, ls=:dash, fillalpha=0.2)
plot!(xstar → confint_pstar_wald(y, x, xstar; α)[2];
    label="", c=:red, ls=:dash)
plot!(ytick=0:0.1:1, yguide="p", xguide="x")
title!("size-$n sample of logistic model with β = $β")
plot!(legend=:outertopright, size=(800, 250),
    leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[43]:

In [44]:

```
# P値函数の視覚化

xstars = range(0, 4, 400)
pstars = range(0, 1, 200)
heatmap(xstars, pstars, (xstar, pstar) → pvalue_pstar_wald(y, x, xstar, pstar);
    clim=(-0.1, 1), colorbar=false)
scatter!(x, (yᵢ → yᵢ==0 ? -0.05 : 1.05).(y); label="", msc=:auto, alpha=0.7, c=1)
plot!(ytick=0:0.1:1, yguide="p", xguide="x")
plot!(xlim=(-0.1, 4.1))
plot!(size=(630, 270), leftmargin=4Plots.mm, bottommargin=4Plots.mm)
```

Out[44]:

横軸の値 x_* ごとにパラメータ値 p_* にP値を対応させるP値函数が決まっている.

上のヒートマップで明るい所ほどP値が大きい.

P値が「暗過ぎない部分」が信頼区間になっている.

4.10 ロジスティック回帰における β_1 に関するWald型のP値函数と信頼区間

$\beta_0 + \beta_1 x_*$ に関するP値函数と信頼区間で $x_* = 0$ の場合を考えれば β_0 に関するP値函数と信頼区間が得られる. 以下では β_1 に関するP値函数と信頼区間を構成しよう.

この節でも次の近似が使えると仮定する:

$$\hat{\beta} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} \sim \mathrm{MvNormal}(\beta,\ I(\hat{\beta})^{-1}),\ \text{approximately.}$$

このとき,

$$\widehat{\mathrm{SE}}_{\hat{\beta}_1} = \sqrt{\frac{\hat{a}}{\hat{a}\hat{c} - \hat{b}^2}} = \left(I(\hat{\beta})^{-1} \text{ の } (2,2) \text{ 成分の平方根} \right)$$

とおくと,

$$\hat{\beta}_1 \sim \mathrm{Normal}\left(\beta_1, \widehat{\mathrm{SE}}_{\hat{\beta}_1}\right),\ \text{approximately.}$$

これにより,「一次の項の係数が β_1 である」という仮説のWald型のP値を次のように定めることができる:

$$\mathrm{pvalue}_{\mathrm{Wald}}(y|x, \beta_1) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |\hat{z}(\beta_1)|)).$$

ここで,

$$\hat{z}(\beta_1) = \frac{\hat{\beta}_1 - \beta_1}{\widehat{\mathrm{SE}}_{\hat{\beta}_1}}.$$

これに対応する β_1 に関する信頼区間は次のようになる:

$$\mathrm{confint}^{\beta_1}_{\mathrm{Wald}}(y|x, \alpha) = \left[\hat{\beta}_1 - z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_1},\ \hat{\beta}_1 + z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_1}\right].$$

ここで, $z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2)$.

In [45]:

```
# ẑ(β₁)が近似的に標準正規分布に従うことの確認
β₀, β₁ = β = [-4, 2]
PP = []
for n in (100, 400)
    @time betahat, sehat, zhat = sim_logistic_regression_beta1(; β, n)
    P = stephist(zhat; norm=true, label="zhat")
    plot!(Normal(), -5, 5; label="Normal(0,1)")
    title!("n = $n")
    push!(PP, P)
end
plot(PP...; size=(800, 250), layout=(1, 2))
```

```
1.152886 seconds (3.00 M allocations: 133.845 MiB, 15.16% compilation time)
3.255637 seconds (2.89 M allocations: 168.034 MiB)
```

Out[45]:

In [46]:

```
# ロジスティック回帰モデルのサンプル(データ)の生成
#Random.seed!(4649373)
β₀, β₁ = β = [-4, 2]
n = 51
x = range(-β₀/β₁-2, -β₀/β₁+2, n)
#y = logistic_rand(x, β)
y = Bool[
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
    1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1,
    1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1]
@show n

# ロジスティック回帰
β̂₀, β̂₁ = β̂ = logistic_mle(y, x)
@show β₀, β₁
@show β̂₀, β̂₁

# P値の例
@show pvalue_beta1_wald(y, x, β₁)

# 信頼区間
@show ci = confint_beta1_wald(y, x)

# P値函数と信頼区間の視覚化
xlim = confint_beta1_wald(y, x; α=0.001)
plot(β₁ -> pvalue_beta1_wald(y, x, β₁), xlim...; label="")
plot!(ytick=0:0.1:1, yguide="P-value", xguide="β₁")
plot!(ci, fill(0.05, 2); label="95% CI")
vline!([β̂₁]; label="MLE of β₁", ls=:dash) # maximum likelihood estimate of β₁
vline!([β₁]; label="true β₁", ls=:dot)
title!("n = $n")
```

```
n = 51
(β₀, β₁) = (-4, 2)
(β̂₀, β̂₁) = (-3.3771836845450944, 1.5218694635151835)
pvalue_beta1_wald(y, x, β₁) = 0.23754555186875032
ci = confint_beta1_wald(y, x) = [0.7284702769932621, 2.3152686500371047]
```

Out[46]:

青の線は β_1 のP値函数であり, 橙色の線分は β_1 の 95% 信頼区間である.

信頼区間はP値函数を有意水準の高さで切断すると得られる.

緑の縦破線は β_1 の最尤推定値であり, 紫色の縦点線はテストデータを生成するために使ったパラメータの β_1 である.

注意: 「真のパラメータ値」(true parameter value)の意味は「現実における真の値」という意味ではなく, 「テストデータを生成するために使ったパラメータの値」という意味であることが多い点に注意を払わなければいけない.

In [47]:

```
# ロジスティック回帰モデルのサンプル(データ)の生成
#Random.seed!(4649373837)
β₀, β₁ = β = [-4, 2]
n = 51
x = range(-β₀/β₁-2, -β₀/β₁+2, n)
#y = logistic_rand(x, β)
y = Bool[
    0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
    1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1,
    1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1]
@show n

# ロジスティック回帰
β̂₀, β̂₁ = β̂ = logistic_mle(y, x)
@show β₀, β₁
@show β̂₀, β̂₁

# P値の例
@show pvalue_beta1_wald(y, x, β₁)

# 信頼区間
@show ci = confint_beta1_wald(y, x)

# P値函数と信頼区間の視覚化
xlim = confint_beta1_wald(y, x; α=0.001)
plot(β₁ -> pvalue_beta1_wald(y, x, β₁), xlim...; label="")
plot!(ytick=0:0.1:1, yguide="P-value", xguide="β₁")
plot!(ci, fill(0.05, 2); label="95% CI")
vline!([β̂₁]; label="MLE of β₁", ls=:dash) # maximum likelihood estimate of β₁
vline!([β₁]; label="true β₁", ls=:dot)
title!("n = $n")
```

```
n = 51
(β₀, β₁) = (-4, 2)
(β̂₀, β̂₁) = (-2.5752705372489446, 1.2582835733068818)
pvalue_beta1_wald(y, x, β₁) = 0.03492292470138703
ci = confint_beta1_wald(y, x) = [0.5690637954332737, 1.9475033511804898]
```

Out[47]:

上の数値例では, データを生成するために使ったパラメータ値の $\beta_1 = 1$ が 95% 信頼区間に含まれていない.

4.11 ロジスティック回帰における β₁ に関するWald型のP値函数と信頼区間の動画

In [48]:

```julia
# ロジスティック回帰モデルのサンプル(データ)の生成
Random.seed!(46493735964)
β₀, β₁ = β = [-4, 2]
n = 1000
x = rand(Uniform(-β₀/β₁-2, -β₀/β₁+2), n)
y = logistic_rand(x, β)

# P値函数と信頼区間の動画
ran = [fill(50, 20); 50:100; 100:2:200; 200:4:400; 600:8:n; fill(n, 20)]
@time anim = @animate for m in ran
    @views X, Y = x[1:m], y[1:m]
    β̂₀, β̂₁ = β̂ = logistic_mle(Y, X)
    ci = confint_beta1_wald(Y, X)
    plot(β₁ -> pvalue_beta1_wald(Y, X, β₁), 0, 5; label="")
    plot!(yguide="P-value", xguide="β₁")
    plot!(ytick=0:0.1:1, ylim=(-0.02, 1.02))
    plot!(ci, fill(0.05, 2); label="95% CI")
    vline!([β̂₁]; label="MLE of β₁", ls=:dash) # maximum likelihood estimate of β₁
    vline!([β₁]; label="true β₁", ls=:dot)
    title!("n = $m")
end

gif(anim, "images/anim_logisticreg_beta1_confint_pvalfunc.gif")
```

```
49.771310 seconds (20.80 M allocations: 832.833 MiB, 0.48% gc time, 1.67% compilation time)
```

```
┌ Info: Saved animation to
│   fn = D:\OneDrive\work\Statistics\2022\images\anim_logisticreg_beta1_confint_pvalfunc.gif
└ @ Plots D:\.julia\packages\Plots\MzlNY\src\animation.jl:130
```

Out[48]:

PDFファイルではこのアニメーションは動かない. 動いている様子を見たい人は以下のリンク先を参照せよ.

- [ロジスティック回帰のβ_1の信頼区間に対応するP値函数](https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_logisticreg_beta1_confint_pvalfunc.gif) (https://github.com/genkuroki/Statistics/blob/master/2022/images/anim_logisticreg_beta1_confint_pvalfunc.gif)

5 x_i 達の値も1または0の場合のロジスティック回帰

x_i 達の値も1または0の場合のロジスティック回帰によって,

- [「検定と信頼区間: 比率の比較」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb) (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)

におけるオッズ比パラメータに関する検定や信頼区間に関する結果を再現できることを説明しよう.

すなわち, ロジスティック回帰は「比率の比較」を含んでいる.

このことは, 線形回帰が「平均の推定」を含んでいることに似ている.

以下の説明は **スコア統計量** や **Fisher情報量行列** などを使う統計分析の方法への入門的解説にもなっている.

5.1 x_i 達の値も1または0の場合にロジスティック回帰モデルは2つの二項分布モデルに等しい

$x = [x_i]_{i=1}^n \in \{1,0\}^n$ であると仮定する.

$X, Y \in \{1,0\}$ について, n_{XY} を $(x_i, y_i) = (X, Y)$ となる i の個数と定め, n_X を $x_i = X$ となる i の個数と定める. これによって次の 2×2 の分割表が得られる:

	$Y=1$	$Y=0$	
$X=1$	n_{11}	n_{10}	n_1
$X=0$	n_{01}	n_{00}	n_0

$$\left(\sum_{X,Y} n_{X,Y} = \sum_X n_X = n\right).$$

行列 N を次のように定める:

$$N = \begin{bmatrix} n_{11} & n_{10} \\ n_{01} & n_{00} \end{bmatrix}$$

さらに, $p = p(\beta)$ と $q = q(\beta)$ を次のように定める:

$$p = p(\beta) = \mathrm{logistic}(\beta_0 + \beta_1), \quad q = q(\beta) = \mathrm{logistic}(\beta_0).$$

これは

$$\log \frac{p}{1-p} = \beta_0 + \beta_1, \quad \log \frac{q}{1-q} = \beta_0$$

と同値なので,

$$\beta_1 = \log \frac{p(1-q)}{(1-p)q} = (\text{パラメータの対数オッズ比}).$$

ロジスティック回帰の統計モデルの確率質量函数は N に関する次の確率質量函数に書き直される:

$$P(N|n_1, n_0, p, q) = \binom{n_1}{n_{11}} p^{n_{11}} (1-p)^{n_{10}} \cdot \binom{n_0}{n_{01}} q^{n_{01}} (1-q)^{n_{00}}.$$

要するにこの節で扱っている最も簡単な場合のロジスティック回帰モデルは2つの二項分布モデルを含んでいる.

5.2 x_i 達の値も1または0の場合のスコア統計量とFisher情報行行列

このとき, スコア統計量は次のようになる:

$$\begin{aligned} A &= A(\beta) = -n_{11}(1-p) + n_{10}p - n_{01}(1-q) + n_{00}q = n_1 p - n_{11} + n_0 q - n_{01}, \\ B &= B(\beta) = -n_{11}(1-p) + n_{10}p = n_1 p - n_{11}. \end{aligned}$$

最尤法の解 $\hat{\beta}$ に対応する $\hat{p} = p(\hat{\beta})$, $\hat{q} = q(\hat{\beta})$ は $A = 0$, $B = 0$ の解になっているので,

$$\hat{p} = \frac{n_{11}}{n_1} = \frac{n_{11}}{n_{11} + n_{10}}, \quad \hat{q} = \frac{n_{01}}{n_0} = \frac{n_{01}}{n_{01} + n_{00}}.$$

Fisher情報量行列の成分達は次のようになる:

$$\begin{aligned} a &= a(\beta) = n_1 p(1-p) + n_0 q(1-q), \\ b &= b(\beta) = n_1 p(1-p), \\ c &= c(\beta) = n_1 p(1-p). \end{aligned}$$

ゆえに, $\hat{a} = a(\hat{\beta})$, $\hat{b} = b(\hat{\beta})$, $\hat{c} = c(\hat{\beta})$ は次のようになる:

$$\hat{a} = \frac{n_{11} n_{10}}{n_1} + \frac{n_{01} n_{00}}{n_0}, \quad \hat{b} = \hat{c} = \frac{n_{11} n_{10}}{n_1}.$$

5.3 x_i 達の値も1または0の場合のWald型のP値函数と信頼区間

したがって,

$$\widehat{\mathrm{SE}}_{\hat{\beta}_1} = \sqrt{\frac{\hat{a}}{\hat{a}\hat{c} - \hat{b}^2}} = \sqrt{\frac{1}{n_{11}} + \frac{1}{n_{10}} + \frac{1}{n_{01}} + \frac{1}{n_{00}}}.$$

ゆえに

$$\hat{z}(\beta_1) = \frac{\hat{\beta}_1 - \beta_1}{\widehat{\mathrm{SE}}_{\hat{\beta}_1}}.$$

の分母の具体系がわかったので, パラメータの対数オッズ比が β_1 であるという仮説のWald型P値函数

$$\mathrm{pvalue}_{\mathrm{Wald}}(N|\beta_1) = 2(1 - \mathrm{cdf}(\mathrm{Normal}(0,1), |\hat{z}(\beta_1)|))$$

や対数オッズ比 β_1 に関するWald型信頼区間

$$\mathrm{confint}^{\beta_1}_{\mathrm{Wald}}(N|\alpha) = \left[\hat{\beta}_1 - z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_1},\ \hat{\beta}_1 + z_{\alpha/2}\widehat{\mathrm{SE}}_{\hat{\beta}_1}\right]$$

の計算の仕方がよく分かったことになる.

以上の結果は

- [「検定と信頼区間: 比率の比較」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)

における「Wald版のオッズ比に関するP値と信頼区間」の再現になっている.

5.4 x_i 達の値も1または0の場合のWilson型のP値函数と信頼区間

5.4.1 A=0で定まる条件付き確率分布の正規分布近似

この節ではスコア統計量の分布の正規分布近似に戻って考え直す:

$$U(\beta) = \begin{bmatrix} A(\beta) \\ B(\beta) \end{bmatrix} \sim \mathrm{MvNormal}(0, I(\beta)),\ \text{approximately}.$$

$I(\beta)$ の成分は $a(\beta), b(\beta), c(\beta)$ と表されているのであった.

回帰係数パラメータの片方の β_1 (対数オッズ比パラメータ)を任意に固定して, β_0 のみを動かして得られる最尤法(尤度最大化)の解を $\tilde{\beta}_0$ と書き,

$$\tilde{\beta} = \begin{bmatrix} \tilde{\beta}_0 \\ \beta_1 \end{bmatrix}$$

とおく. このとき, $\tilde{\beta}_0$ は, 与えられた β_1 に対する方程式

$$A = A(\beta) = A\left(\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}\right) = 0$$

の解になる. ゆえに, $U(\tilde{\beta})$ が従う分布は $U(\beta)$ が従う分布を $A(\beta) = 0$ の場合に制限して得られる条件付き確率分布になる. このことから, 次が成立することがわかる:

$$B(\tilde{\beta}) \sim \mathrm{Normal}\left(0, \sqrt{\frac{\tilde{a}\tilde{c} - \tilde{b}^2}{\tilde{a}}}\right),\ \text{approximately}.$$

ここで, $\tilde{a}, \tilde{b}, \tilde{c}$ は $\beta = \tilde{\beta}$ のときの a, b, c である:

$$\tilde{a} = a(\tilde{\beta}), \quad \tilde{b} = a(\tilde{\beta}), \quad \tilde{c} = a(\tilde{\beta}).$$

$\tilde{a}/(\tilde{a}\tilde{c} - \tilde{b}^2)$ が $I(\tilde{\beta})^{-1}$ の $(2,2)$ 成分に等しいことに注意せよ. (2変量正規分布の条件付き確率分布について自分で計算してみて, そうなる理由を確認せよ.)

この結果を有用にするためには $\tilde{\beta}_0$ を求め, $\tilde{a}, \tilde{b}, \tilde{c}, B(\tilde{\beta})$ の具体的な形を決定する必要がある. 以下ではそれを実行しよう.

5.4.2 与えられた対数オッズ比パラメータの値 β_1 に対する β_0 の推定量に関する公式

公式: $\tilde{\beta}_0$ と $\tilde{a}, \tilde{b}, \tilde{c}, B(\tilde{\beta})$ の具体的な形は次のようになる:

$$
\begin{aligned}
&\hat{\beta}_0 = \log \frac{n_{01} + \tilde{\delta}}{n_{00} - \tilde{\delta}}, \\
&\tilde{a} = \frac{(n_{11} - \tilde{\delta})(n_{10} + \tilde{\delta})}{n_1} + \frac{(n_{01} + \tilde{\delta})(n_{00} - \tilde{\delta})}{n_0}, \\
&\tilde{b} = \tilde{c} = \frac{(n_{11} - \tilde{\delta})(n_{10} + \tilde{\delta})}{n_1}, \\
&B(\tilde{\beta}) = -\tilde{\delta}.
\end{aligned}
$$

ただし, $\tilde{\delta}$ は

$$
\omega = \exp(\beta_1), \quad A = 1 - \omega, \quad B = n_{11} + n_{00} + \omega(n_{10} + n_{01}), \quad C = n_{11}n_{00} - \omega n_{01}n_{10}
$$

とおいて, 次のように定義される:

$$
\tilde{\delta} = \frac{2C}{B + \sqrt{B^2 - 4AC}}.
$$

証明: $A(\beta)$ の具体的な形は

$$
\begin{aligned}
&A(\beta) = -n_{11}(1-p) + n_{10}p - n_{01}(1-q) + n_{00}q = n_1 p - n_{11} + n_0 q - n_{01}, \\
&\frac{q}{1-q} = \exp(\beta_0), \quad \frac{p(1-q)}{(1-p)q} = \exp(\beta_1)
\end{aligned}
$$

だったので, 与えられた β_1 に対する β_0 に関する方程式 $A(\beta) = 0$ は次と同値である:

$$
\delta := n_{11} - n_1 p = -(n_{01} - n_0 q).
$$

このとき,

$$
p = \frac{n_{11} - \delta}{n_1}, \quad 1 - p = \frac{n_{10} + \delta}{n_1}, \quad q = \frac{n_{01} + \delta}{n_0}, \quad 1 - q = \frac{n_{00} - \delta}{n_0}
$$

なので, 条件 $((p(1-q))/((1-p)q)) = \exp(\beta_1)$ は次のように書き直される:

$$
\frac{(n_{11} - \delta)(n_{00} - \delta)}{(n_{10} + \delta)(n_{00} + \delta)} = \exp(\beta_1).
$$

これを δ に関する2次方程式に直してから解くと, その解は

$$
\tilde{\delta} = \frac{2C}{B + \sqrt{B^2 - 4AC}}.
$$

ここで,

$$
\omega = \exp(\beta_1), \quad A = 1 - \omega, \quad B = n_{11} + n_{00} + \omega(n_{10} + n_{01}), \quad C = n_{11}n_{00} - \omega n_{01}n_{10}.
$$

$\tilde{p}, \tilde{q}$ を次のように定める:

$$
\tilde{p} = \frac{n_{11} - \tilde{\delta}}{n_1}, \quad \tilde{q} = \frac{n_{01} + \tilde{\delta}}{n_0}.
$$

このとき, β_0 に関する方程式 $A(q) = 0$ は次のように解ける:

$$
\hat{\beta}_0 = \log \frac{\tilde{q}}{1 - \tilde{q}} = \log \frac{n_{01} + \tilde{\delta}}{n_{00} - \tilde{\delta}}.
$$

そして, このとき,

$$
\begin{aligned}
&\tilde{a} = n_1 \tilde{p}(1 - \tilde{p}) + n_0 \tilde{q}(1 - \tilde{q}) = \frac{(n_{11} - \tilde{\delta})(n_{10} + \tilde{\delta})}{n_1} + \frac{(n_{01} + \tilde{\delta})(n_{00} - \tilde{\delta})}{n_0}, \\
&\tilde{b} = \tilde{c} = n_1 \tilde{p}(1 - \tilde{p}) = \frac{(n_{11} - \tilde{\delta})(n_{10} + \tilde{\delta})}{n_1}, \\
&B(\tilde{\beta}) = n_1 \tilde{p} - n_{11} = -\tilde{\delta}.
\end{aligned}
$$

これで $B(\tilde{\beta})$ の具体的な形がわかった. **証明終**

5.4.3 対数オッズ比パラメータ β_1 に関するWilson型のP値函数と信頼区間の構成

以上の結果と $n_{11}+n_{10}=n_1, n_{01}+n_{00}=n_0$ より,

$$
\begin{aligned}
\frac{\tilde{a}}{\tilde{a}\tilde{c}-\tilde{b}^2} &= \frac{\dfrac{(n_{11}-\tilde{\delta})(n_{10}+\tilde{\delta})}{n_1}+\dfrac{(n_{01}+\tilde{\delta})(n_{00}-\tilde{\delta})}{n_0}}{\dfrac{(n_{11}-\tilde{\delta})(n_{10}+\tilde{\delta})}{n_1}\dfrac{(n_{01}+\tilde{\delta})(n_{00}-\tilde{\delta})}{n_0}} \\
&= \frac{n_1}{(n_{11}-\tilde{\delta})(n_{10}+\tilde{\delta})}+\frac{n_1}{(n_{01}+\tilde{\delta})(n_{00}-\tilde{\delta})} \\
&= \frac{1}{n_{11}-\tilde{\delta}}+\frac{1}{n_{10}+\tilde{\delta}}+\frac{1}{n_{01}+\tilde{\delta}}+\frac{1}{n_{00}-\tilde{\delta}}.
\end{aligned}
$$

したがって,

$$
B(\tilde{\beta}) \sim \mathrm{Normal}\left(0, \sqrt{\frac{\tilde{a}\tilde{c}-\tilde{b}^2}{\tilde{a}}}\right), \text{ approximately.}
$$

という結果は次のように書き直される:

$$
z(\beta_1) := \tilde{\delta}\,\widehat{\mathrm{SE}}_{\tilde{\delta}}^{-1} \sim \mathrm{Normal}(0,1), \text{ approximately.}
$$

ここで,

$$
\widehat{\mathrm{SE}}_{\tilde{\delta}}^{-1} = \sqrt{\frac{\tilde{a}}{\tilde{a}\tilde{c}-\tilde{b}^2}} = \sqrt{\frac{1}{n_{11}-\tilde{\delta}}+\frac{1}{n_{10}+\tilde{\delta}}+\frac{1}{n_{01}+\tilde{\delta}}+\frac{1}{n_{00}-\tilde{\delta}}}.
$$

さらにこれは, 次と同値である:

$$
\chi^2(\beta_1) := z(\beta_1)^2 = \tilde{\delta}^2\,\widehat{\mathrm{SE}}_{\tilde{\delta}}^{-2} \sim \mathrm{Chisq}(1), \text{ approximately.}
$$

これより, 「対数オッズ比パラメータは β_1 である」という仮説のWilson型の(スコア検定の)P値函数が以下のように定義される:

$$
\mathrm{pvalue}_{\mathrm{Wilson}}(N|\beta_1) = 2(1-\mathrm{cdf}(\mathrm{Normal}(0,1), |z(\beta_1)|) = \mathrm{cdf}(\mathrm{Chisq}(1), \chi^2(\beta_1)).
$$

さらに, Wilson型(スコア検定に付随する)信頼区間が次のように得られる:

$$
\mathrm{confint}_{\mathrm{Wilson}}^{\beta_1}(N|\alpha) = \{\, \beta_1 \in \mathbb{R} \mid \mathrm{pvalue}_{\mathrm{Wilson}}(N|\beta_1) \geq \alpha \,\}.
$$

以上の結果は

- [「検定と信頼区間: 比率の比較」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)

における「Pearsonのχ²検定版のオッズ比に関するP値と信頼区間」の再現になっている. 信頼区間の計算法のより詳しい説明や計算の例もそちらにある.

5.5 x_i 達の値も1または0の場合にロジスティック回帰の一般化の役に立ち方

$x_i = (X_i, k) \in \{1,0\} \times \{1,\dots,K\}$ が $X_i = 1,0$ と $k = 1,2,\dots,K$ の組である場合への以上の話の一般化は, K 個の統計分析の結果をまとめあげる **メタアナリシス** などでも役に立っている.

例えば, K 個の統計分析に関する共通オッズ比を推定するためのMantel-Haenszel統計量の話やMantel-Haenszel検定の話をインターネットで検索してみよ. 共通オッズ比のMantel-Haenszel検定はロジスティック回帰を使った検定と本質的に同値になっている.

- [共通オッズ比 Mantel-Haenszel統計量 (https://www.google.com/search?q=%E5%85%B1%E9%80%9A%E3%82%AA%E3%83%83%E3%82%BA%E6%AF%94+Mantel-Haenszel%E7%B5%B1%E8%A8%88%E9%87%8F)](https://www.google.com/search?q=%E5%85%B1%E9%80%9A%E3%82%AA%E3%83%83%E3%82%BA%E6%AF%94+Mantel-Haenszel%E7%B5%B1%E8%A8%88%E9%87%8F)
- [Mantel-Haenszel検定 (https://www.google.com/search?q=Mantel-Haenszel%E6%A4%9C%E5%AE%9A)](https://www.google.com/search?q=Mantel-Haenszel%E6%A4%9C%E5%AE%9A)

検索すれば, x_i 達が離散的な値を取る場合のロジスティック回帰は医療統計でよく使われていることがわかるだろう.

このように, ロジスティック回帰は機械学習の文脈で解説されることが多いが, 医療統計でも使われている.

将来, 機械学習の技術を利用しようと思っている人も, それとは毛色が違う医療統計についても学んでおけば, アイデアの幅が広がり, オリジナルな仕事をできるかもしれない.

In [ ]:

```
1
```

まとめ

- 黒木玄
- 2022-07-20〜2022-07-23

このノートではJulia言語 (https://julialang.org/)を使用している:

- Julia言語のインストールの仕方の一例 (https://nbviewer.org/github/genkuroki/msfd28/blob/master/install.ipynb)

自明な誤りを見つけたら, 自分で訂正して読んで欲しい. 大文字と小文字の混同や書き直しが不完全な場合や符号のミスは非常によくある.

このノートに書いてある式を文字通りにそのまま読んで正しいと思ってしまうとひどい目に会う可能性が高い. しかし, 数が使われている文献には大抵の場合に文字通りに読むと間違っている式や主張が書いてあるので, 内容を理解した上で訂正しながら読んで利用しなければいけない. 実践的に数学を使う状況では他人が書いた式をそのまま信じていけない.

このノートの内容よりもさらに詳しいノートを自分で作ると勉強になるだろう. 膨大な時間を取られることになるが, このノートの内容に関係することで飯を食っていく可能性がある人にはそのためにかけた時間は無駄にならないと思われる.

目次

- 1 二項分布モデルでのClopper-Pearsonの信頼区間
  - 1.1 Clopper-Pearsonの信頼区間を例に信頼区間の解釈の仕方について説明
  - 1.2 二項分布モデルのClopper-Pearsonの信頼区間の効率的な計算の仕方
  - 1.3 二項分布モデルのBayes統計
  - 1.4 P値とBayes統計の関係
  - 1.5 復習: 二項分布とベータ分布の関係の証明
    - 1.5.1 両辺を p で微分して確認する方法
    - 1.5.2 ベータ分布の一様分布の順序統計量の分布として解釈を使う方法
- 2 P値函数について
  - 2.1 P値はモデルのパラメータ値とデータの数値の整合性の指標の1つ
  - 2.2 P値は信頼区間を考えたいすべてのパラメータ値について定義されている
  - 2.3 分割表の場合のP値函数
  - 2.4 P値函数から信頼区間が定義される
  - 2.5 2×2の分割表のP値と信頼区間とP値函数の例
  - 2.6 2×2の分割表での検定ではどれを使うべきか
  - 2.7 P値函数と最尤法の関係
  - 2.8 以上のまとめの図
  - 2.9 おまけ: Bayes統計の方法を使った場合
- 3 Welchの t 検定について
  - 3.1 Welchの t 検定のP値と信頼区間の定義
  - 3.2 Welchの t 検定のP値や信頼区間の計算例
    - 3.2.1 WolframAlphaによるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例
    - 3.2.2 Julia言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例
- 4 数学的な補足: 大数の法則と中心極限定理について
  - 4.1 二項分布の大数の法則
  - 4.2 二項分布の中心極限定理
  - 4.3 他の分布の場合
    - 4.3.1 例: Poisson分布
    - 4.3.2 例: Gamma分布
  - 4.4 標本分布の場合
  - 4.5 再掲: 大数の法則と中心極限定理のイメージ
  - 4.6 統計学の基礎になる確率論の三種の神器
  - 4.7 Kullback-Leibler情報量に関するSanovの定理の数値例

In [1]:

```julia
ENV["LINES"], ENV["COLUMNS"] = 100, 100
using Base.Threads
using BenchmarkTools
using DataFrames
using Distributions
using LinearAlgebra
using Memoization
using Optim
using Printf
using QuadGK
using RCall
@rimport stats as R
using Random
Random.seed!(4649373)
using Roots
using SpecialFunctions
using StaticArrays
using StatsBase
using StatsFuns
using StatsPlots
default(fmt = :png, size = (400, 250),
    titlefontsize = 10, guidefontsize=9, plot_titlefontsize = 12)
using SymPy
```

In [2]:

```julia
# Override the Base.show definition of SymPy.jl:
# https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.

@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
    print(io, as_markdown("\\displaystyle " *
            sympy.latex(x, mode="plain", fold_short_frac=false)))
end
@eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
    function toeqnarray(x::Vector{Sym})
        a = join(["\\displaystyle " *
                sympy.latex(x[i]) for i in 1:length(x)], "\\\\")
        """\\left[ \\begin{array}{r}$a\\end{array} \\right]"""
    end
    function toeqnarray(x::AbstractArray{Sym,2})
        sz = size(x)
        a = join([join("\\displaystyle " .* map(sympy.latex, x[i,:]), "&")
                for i in 1:sz[1]], "\\\\")
        "\\left[ \\begin{array}{" * repeat("r",sz[2]) * "}" * a * "\\end{array}\\right]"
    end
    print(io, as_markdown(toeqnarray(x)))
end
```

In [3]:

```julia
safemul(x, y) = x == 0 ? x : isinf(x) ? typeof(x)(Inf) : x*y
safediv(x, y) = x == 0 ? x : isinf(y) ? zero(y) : x/y

x ≲ y = x < y || x ≈ y

mypdf(dist, x) = pdf(dist, x)
mypdf(dist::DiscreteUnivariateDistribution, x) = pdf(dist, round(Int, x))

distname(dist::Distribution) = replace(string(dist), r"{.*}" => "")
myskewness(dist) = skewness(dist)
mykurtosis(dist) = kurtosis(dist)
function standardized_moment(dist::ContinuousUnivariateDistribution, m)
    μ, σ = mean(dist), std(dist)
    quadgk(x -> (x - μ)^m * pdf(dist, x), extrema(dist)...)[1] / σ^m
end
myskewness(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 3)
mykurtosis(dist::MixtureModel{Univariate, Continuous}) =
    standardized_moment(dist, 4) - 3
```

Out[3]:

```
mykurtosis (generic function with 2 methods)
```

In [4]:
```julia
function logtick(; xlim=(0.03, 30))
    xmin, xmax = xlim
    a = floor(Int, log10(xmin))
    b = ceil(Int, log10(xmax))
    nums =     [1, 2, 3, 4, 5, 6, 7, 8, 9]
    mask = Bool[1, 1, 0, 0, 1, 0, 0, 0, 0]

    logtick = foldl(vcat, ([10.0^k*x for x in nums if xmin ≤ 10.0^k*x ≤ xmax] for k in a:b))
    logticklabel_a = foldl(vcat,
        ([mask[i] ? string(round(10.0^k*x; digits=-k)) : ""
                for (i, x) in enumerate(nums) if xmin ≤ 10.0^k*x ≤ xmax]
            for k in a:-1))
    logticklabel_b = foldl(vcat,
        ([mask[i] ? string(10^k*x) : ""
                for (i, x) in enumerate(nums) if xmin ≤ 10.0^k*x ≤ xmax]
            for k in 0:b))
    logticklabel = vcat(logticklabel_a, logticklabel_b)
    (logtick, logticklabel)
end

# logtick()
```

Out[4]: logtick (generic function with 1 method)

1 二項分布モデルでのClopper-Pearsonの信頼区間

1.1 Clopper-Pearsonの信頼区間を例に信頼区間の解釈の仕方について説明

「n 回中 k 回当たりが出た」「n 人中 k 人が重症化」「n 人中 k 人が商品を購入」などの型のデータを扱うための統計モデルとして, 成功確率パラメータ p を持つ二項分布

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

を使う状況を考える. このとき, 仮説 $p = p_0$ のP値(データの数値とモデル+パラメータ値の整合性の指標の1つ)を次のように定義できるのであった:

$$\mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0) = \min \begin{pmatrix} 1 \\ 2\,\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k) \\ 2(1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k-1)) \end{pmatrix}$$

これをClopper-PearsonのP値と呼ぶことにしていた. このP値は以下の3つのうちの最小値として定義されている:

- 1
- 仮説 $p = p_0$ の下での二項分布で n 回中の当たりの回数が k 回以下の確率の2倍
- 仮説 $p = p_0$ の下での二項分布で n 回中の当たりの回数が k 回以上の確率の2倍

本当はモデル内で仮想的に生成されたデータの値 i (n 回中の当たりの回数)が期待値 np_0 からデータの数値 k 以上離れる確率をP値としたいのだが(P値の定義は仮説下のモデル内でデータの数値以上に極端な値が生成される確率またはその近似値であった), 反対側の確率を自然に決めることができないので, 2倍して反対側の確率も足し上げたことにしている. (これは1つの処方箋に過ぎず, 別の方法もあるのであった.)

これに対応する信頼度 $1 - \alpha$ の信頼区間の定義は次の通り:

$$\mathrm{confint}_{\mathrm{CP}}(k|n, \alpha) = \{\, p_0 \in [0, 1] \mid \mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0) \geq \alpha \,\}.$$

仮説 $p = p_0$ のP値 $\mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0)$ はデータの数値「n 回中 k 回」とモデルのパラメータに関する仮説 $p = p_0$ の整合性の指標として使われるのであった.

信頼区間を定義している条件 $\mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0) \geq \alpha$ はその整合性の指標の値がある閾値 α (この閾値は有意水準と呼ばれる)以上になっているという条件になっている.

通常, 有意水準 α は目的に合わせて小さな値を採用し, 検定の手続きではP値が α 未満になるパラメータ値 $p = p_0$ を棄却する.

信頼度 $1 - \alpha$ 信頼区間は有意水準 α で棄却されないパラメータ値 $p = p_0$ 全体の集合になっている.

「棄却されないこと」は「判断を保留すること」を意味する.

信頼区間に含まれるパラメータ値 $p = p_0$ については判断を保留することになるが, もしも信頼区間が十分に狭くなっていれば, その外側にある大部分のパラメータ値は検定の手続きによって棄却されており, 可能性として考慮すればよいのは, 十分に狭くなっている信頼区間に含まれるパラメータ値だけになっているので, 科学的に十分に意味のある結論を出すことができる場合が出て来る.

ただし, 信頼区間を使って科学的に信頼できる結論を出すためには, データの取得法が信頼できるか, 採用した統計モデルは妥当であるか, その他見落としている重要な問題はないか, などの多くの問題をクリアする必要がある.

機械的に信頼区間を計算して科学的な御墨付きが得られたような態度を取ることは誤りなので注意して欲しい.

1.2 二項分布モデルのClopper-Pearsonの信頼区間の効率的な計算の仕方

Clopper-Pearsonの信頼区間は定義をうまく変形すると以下のように書き直される:

$$\mathrm{confint}_{\mathrm{CP}}(k|n,\alpha) = [p_L, p_U].$$

ここで, p_L, p_U は次の条件で特徴付けられる値である:

$$\begin{aligned} 1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_L), k-1) &= \alpha/2, \\ \mathrm{cdf}(\mathrm{Binomial}(n, p_U), k) &= \alpha/2. \end{aligned}$$

すなわち, 小さい方の p_L は $p = p_L$ の二項分布内で n 回中の当たりの回数が k 以上になる確率が $\alpha/2$ になるという条件で特徴付けられ, 大きい方の p_U は $p = p_U$ の二項分布内で n 回中当たりの回数が k 回以下になる確率が $\alpha/2$ になるという条件で特徴付けられる.

問題はこの p_L, p_U をどのように計算するかである.

これには二項分布とベータ分布の間の関係を使う驚くべき方法がある.

二項分布の累積分布函数(cumulative distribution function, cdf)の定義は

$$\mathrm{cdf}(\mathrm{Binomial}(n,p),k) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}$$

である. 場合によっては万単位の個数の値を足し上げる計算になる.

しかし, そういう和を取る計算をコンピュータで素朴に行うのはコンピュータ資源の無駄遣いになってしまう.

なぜならば, すでに説明したように非常にうまい方法があるからである.

二項分布とベータ分布の関係:

$$\begin{aligned} 1 - \mathrm{cdf}(\mathrm{Binomial}(n,p),k-1) &= \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p), \\ \mathrm{cdf}(\mathrm{Binomial}(n,p),k) &= 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p). \end{aligned}$$

この前者と後者の公式は同値な公式になっており, 片方からもう一方が導かれる.

以下でこれの前者を $n = 20$, $k = 7$ の場合に数値的に確認してみよう.

In [5]:
```
n, k = 20, 7
plot(p → ccdf(Binomial(n, p), k-1), 0, 1; label="ccdf(Binomial(n, p), k-1)")
plot!(p → cdf(Beta(k, n-k+1), p); label="cdf(Beta(k, n-k+1), p)", ls=:dash)
title!("n = $n, k = $k", xguide="p", legend=:bottomright)
```

Out[5]:

2つのグラフがぴったり重なり合っている. (ccdf は $1 - \mathrm{cdf}$ の意味である.)

ベータ分布の累積分布函数 $\mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p)$ は正則化された不完全ベータ函数という名前がついている基本特殊函数になっており, その p に関する逆函数(分位点函数, quantile function)もコンピュータの基本特殊函数ライブラリの中で効率的に実装されている. だから,

$$1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_L), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p_L) = \alpha/2$$

を満たす p_L は, ベータ分布の分位点函数(quantile function)を用いて,

$$p_L = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2)$$

を使えばコンピュータで効率的に計算できる. 同様に

$$\mathrm{cdf}(\mathrm{Binomial}(n, p_U), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p_U) = \alpha/2$$

を満たす p_U は

$$p_U = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2)$$

を使えばコンピュータで効率的に計算できる.

まとめ: 二項分布モデルのでClopper-Pearsonの信頼区間 $[p_L, p_U]$ は

$$p_L = \mathrm{quantile}(\mathrm{Beta}(k, n-k+1), \alpha/2), \quad p_U = \mathrm{quantile}(\mathrm{Beta}(k+1, n-k), 1-\alpha/2)$$

と計算できる.

二項分布とベータ分布の間の関係が, コンピュータ上に実装された基本特殊函数ライブラリを通して, 二項分布モデルの場合の信頼区間の効率的な計算に役に立っている!

- 異なる確率分布に間の関係に関する数学
- 不完全ベータ函数などの基本特殊函数に関する数学
- それをコンピュータで実装するプログラミングの技術

などの多くの技術を組み合わせることによって, Clopper-Pearsonの信頼区間のシンプルで効率的な計算法が実現されている!

このようなことから, **このClopper-Pearsonの信頼区間についてこのように理解することは, 高等教育を受けた人達にとっての重要な教養になり得る** と思われる.

例えば, $n = 20$, $k = 7$, $\alpha = 0.05$ の場合の信頼度 $1-\alpha$ のClopper-Pearsonの信頼区間は次のように計算される.

In [6]:
```
n, k, α = 20, 7, 0.05
[quantile(Beta(k, n-k+1), α/2), quantile(Beta(k+1, n-k), 1-α/2)] ▷ println
```

```
[0.1539092047845412, 0.5921885345328282]
```

R言語を使って計算した結果も同じになっている:

In [7]:
```
rcopy(R"""binom.test(7, 20, p=0.05)""")[:conf_int] ▷ println
```

```
[0.15390920478454115, 0.5921885345328282]
```

WolframAlphaでも以下のようにすれば容易に計算できる.

$p_L \to$ `quantile(BetaDistribution(7, 20-7+1), 0.025)` → [実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%287%2C+20-7%2B1%29%2C+0.025%29)] → 0.153909

$p_U \to$ `quantile(BetaDistribution(7+1, 20-7), 0.975)` → [実行 (https://www.wolframalpha.com/input?i=quantile%28BetaDistribution%287%2B1%2C+20-7%29%2C+0.975%29)] → 0.592189

1.3 二項分布モデルのBayes統計

二項分布とベータ分布の関係は通常の検定とBayes統計の関係を理解するためにも役に立つ.

二項分布のベイズ統計では先に用意した二項分布の確率質量函数 $P(k|n,p)$ だけではなく, パラメータ p に関する確率密度函数 $\varphi(p)$ を任意に与え(目的に合わせて適切に事前分布を決めることになるが, 理論的には任意でよい), (k,p) に関する同時分布 (joint distribution)

$$P(k,p|n) = P(k|n,p)\varphi(p), \quad \sum_{k=0}^{n} \int_0^1 P(k,p|n)\,dp = 1$$

を考える. そして, 「n 回中 k 回」の形のデータの数値が与えられたとき, このモデルの同時確率分布を, データと同じ数値が生成されたという条件で制限して得られるパラメータ p に関する条件付き確率分布 $\varphi(p|n,k)$ を考える:

$$\varphi(p|n,k) = \frac{P(k|n,p)\varphi(p)}{\int_0^1 P(k|n,p)\varphi(p)\,dp}.$$

このとき, $\varphi(p)$ を **事前分布** (prioir)と呼び, $\varphi(p|n,k)$ を **事後分布** (posterior)と呼ぶ.

条件付き確率分布については

- [「条件付き確率分布, 尤度, 推定, 記述統計」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20

に解説がある. 数学的には

- 確率＝全体に対する部分の割合
- 条件付き確率＝部分に対する部分の割合

のように考えると直観的に理解し易いと思う. 上の事後分布の公式を **ベイズの定理** と呼ぶこともあるが, ベイズの定理を知らなくても, 条件付き確率分布について理解していれば困らない.

事後分布 $\varphi(p|n,k)$ の $p=p_0$ での値 $\varphi(p=p_0|n,k)$ はデータの数値 $p=p_0$ とデータの数値 n,k の統計モデル $P(k,p|n) = P(k|n,p)\varphi(p)$ の下での整合性の指標として利用できる. (P値とはまた別のモデルとデータの整合性の指標が得られたことになる.)

さて, 例によって, 問題はどのように事後分布と呼ばれる条件付き確率分布を計算するかである.

事後分布の公式を見れば分母が積分

$$Z(k|n) := \int_0^1 P(k|n,p)\varphi(p)\,dp$$

になっていて, その積分の部分の計算が大変そうなことがわかる. (事後分布の分母を **周辺尤度** (marginal likelihood)と呼ぶことがある. 他にも様々な呼び方があるが, 呼び方自体は重要ではない.)

二項分布の場合には, 事前分布としてベータ分布を採用すると, 事後分布もベータ分布になり, 事後分布の計算が著しく単純化される. この事実をベータ分布は二項分布の **共役事前分布** (conjugate prior)であるという.

以下では $\varphi(p)$ が $\mathrm{Beta}(a,b)$ $(a,b>0)$の確率密度函数であると仮定する:

$$\varphi(p) = \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)} \quad (0<p<1).$$

このとき, 事後分布の分母は次のように計算される:

$$\begin{aligned}
Z(k|n) &= \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \frac{p^{a-1}(1-p)^{b-1}}{B(a,b)}\,dp \\
&= \binom{n}{k} \frac{1}{B(a,b)} \int_0^1 p^{a+k-1}(1-p)^{b+n-k}\,dp \\
&= \binom{n}{k} \frac{B(a+k,b+n-k)}{B(a,b)}.
\end{aligned}$$

ゆえに, 事後分布は次のようになる:

$$\begin{aligned}
\varphi(p|n,k) &= \frac{P(k|n,p)\varphi(p)}{Z(k|n)} \\
&= \binom{n}{k} \frac{1}{B(a,b)} p^{a+k-1}(1-p)^{b+n-k} \times \left(\binom{n}{k} \frac{B(a+k,b+n-k)}{B(a,b)} \right)^{-1} \\
&= \frac{p^{a+k-1}(1-p)^{b+n-k}}{B(a+k,b+n-k)}.
\end{aligned}$$

以上によって, 事前分布が共役事前分布 $\mathrm{Beta}(a,b)$ のとき, データ「n 回中 k 回」から得られる事後分布は $\mathrm{Beta}(a+k,b+n-k)$ になることがわかった.

$a,b>0$ と仮定していたが, $a=0$ または $b=0$ の場合にも, $a+k$ と $b+n-k$ が共に正になるならば, 事後分布は $\mathrm{Beta}(a+k,b+n-k)$ はうまく定義されている. このような場合のうまく定義されていない事前分布は **improper事前分布** (improper prior)と呼ばれている.

この節の事柄については

- 「例：ベータ函数と二項分布の関係とその応用」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/07-1%20Relationship%20between%20beta%20and%20binomial%20distributions.ipynb)

の最後の方にも解説がある.

1.4 P値とBayes統計の関係

二項分布とベータ分布の累積分布函数の間には以下の関係があるのであった:

$$
\begin{aligned}
&1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p_0), \\
&\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k) = 1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p_0).
\end{aligned}
$$

一般に分布 D について累積分布函数 $\mathrm{cdf}(D, x)$ は分布 D に従う確率変数 X が x 以下になる確率を意味することに注意せよ.

ゆえに上の公式の前者の両辺は以下のような解釈を持つ:

- $1 - \mathrm{cdf}(\mathrm{Binomial}(n, p_0), k-1)$ (分布 $\mathrm{Binomial}(n, p_0)$ で k 以上になる確率)は仮説 $p \le p_0$ に関する片側検定のP値である.
- $\mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p_0)$ はimproper事前分布 $\mathrm{Beta}(0, 1)$ の下での事後分布 $\mathrm{Beta}(k, n-k+1)$ において仮説 $p \le p_0$ が成立する確率である.

このように, 仮説 $p \le p_0$ の片側検定のP値はBayes統計での事後分布において仮説 $p \le p_0$ が成立する確率に一致する. 同様に,

- $\mathrm{cdf}(\mathrm{Binomial}(n, p_0), k)$ (分布 $\mathrm{Binomial}(n, p_0)$ で k 以下になる確率)は仮説 $p \ge p_0$ に関する片側検定のP値である.
- $1 - \mathrm{cdf}(\mathrm{Beta}(k+1, n-k), p_0)$ はimproper事前分布 $\mathrm{Beta}(1, 0)$ の下での事後分布 $\mathrm{Beta}(k+1, n-k+1)$ において仮説 $p \ge p_0$ が成立する確率である.

ゆえに, 仮説 $p \ge p_0$ の片側検定のP値はBayes統計での事後分布において仮説 $p \ge p_0$ が成立する確率に一致する.

このように, 数学的にはP値とBayes統計のあいだにexactな関係が付けられる場合もある.

P値を使う統計学とBayes統計は決して水と油ではない.

以上によって, 片側検定のP値については, Bayes統計でのexactな解釈が得られたことになる.

しかし, 通常使われるP値は両側検定のP値(Clopper-Pearson型のP値では片側検定のP値の2倍を両側検定のP値として採用する)とBayes統計の関係はどうなっているのだろうか?

Clopper-Pearson型のP値のBayes統計での類似物を構成するには, 上の片側検定の場合に関する結果を利用すればよい.

そのときに問題になることは, 片側検定の向きを変えると, 使用する(improper)事前分布が $\mathrm{Beta}(0, 1)$ と $\mathrm{Beta}(1, 0)$ のあいだで変化してしまうことである.

そこで, 両側検定のP値のBayes統計によるexactな解釈を作ることはあきらめて, 近似的な関係を得ることを目標にしよう.

そのためには事前分布として $\mathrm{Beta}(0, 1)$ と $\mathrm{Beta}(1, 0)$ の中間に位置する $\mathrm{Beta}(1/2, 1/2)$ を採用すると良さそうであると予想され, 実際にそうであることが計算によって確認可能である.

$\mathrm{Beta}(1/2, 1/2)$ は二項分布の **Jeffreys事前分布** と呼ばれる.

他の選択肢として, 事前分布として平坦事前分布 $\mathrm{Beta}(1, 1)$ を採用することも考えられる. ($\mathrm{Beta}(1, 1)$ の確率密度函数は $\varphi(p) = 1$ $(0 < p < 1)$ となるので, **平坦事前分布** または **一様事前分布** と呼んだりする.)

事前分布として平坦事前分布を採用すると, 両側検定のP値との対応において, Jeffreys事前分布を採用した場合よりも誤差が増えるが, k と $n-k$ が大きければ, $\mathrm{Beta}(k+1/2, n-k+1/2)$ と $\mathrm{Beta}(k+1, n-k+1)$ の違いは小さくなるので, 実践的な応用の場面では問題でなくなる.

以下では, 事前分布が $\mathrm{Beta}(a, a)$ である場合を考える.

このとき, 事後分布は $\mathrm{Beta}(k+a, n-k+a)$ になる.

この事後分布における仮説 $p = p_0$ の両側検定に関するClopper-Pearson型のP値函数の類似物は次のように定義される:

$$
\mathrm{pvalue}_{\mathrm{Bayes}}(k|n, p = p_0) = \min \begin{pmatrix} 1 \\ 2\,\mathrm{cdf}(\mathrm{Beta}(k+a, n-k+a), p_0) \\ 2(1 - \mathrm{cdf}(\mathrm{Beta}(k+a, n-k+a), p_0)) \end{pmatrix}.
$$

これは, 二項分布に関するClopper-Pearson型のP値をそのままベータ分布に一般化しただけの定義になっている.

さらに, 通常のP値として, Clopper-Pearson型のP値だけではなく, 正規分布近似(中心極限定理)を使ったWilson型のP値函数も考えることにしよう:

$$\mathrm{pvalue}_{\mathrm{Wilson}}(k|n, p = p_0) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{|k - np_0|}{\sqrt{np_0(1-p_0)}}\right)\right).$$

このとき, 次が成立している.

結論: k と $n-k$ が大きければ, 以上の3つのP値 $\mathrm{pvalue}_{\mathrm{CP}}(k|n, p = p_0)$, $\mathrm{pvalue}_{\mathrm{Bayes}}(k|n, p = p_0)$, $\mathrm{pvalue}_{\mathrm{Wilson}}(k|n, p = p_0)$ は近似的に一致する.

要するに, 両側検定の通常使われるP値についても, Bayes統計での近似的な解釈が存在する.

これを $n = 100$, $k = 30$ の場合に数値的に確認してみよう.

In [8]:

```julia
# 上に書いてある定義の通りのP値の実装

function pvalue_cp(k, n, p)
    bin = Binomial(n, p)
    min(1, 2cdf(bin, k), 2ccdf(bin, k-1))
end

function pvalue_wilson(k, n, p)
    z = (k - n*p)/√(n*p*(1-p))
    2ccdf(Normal(0, 1), abs(z))
end

function pvalue_bayes(k, n, p; a=1/2)
    beta = Beta(k+a, n-k+a)
    min(1, 2cdf(beta, p), 2ccdf(beta, p))
end

# 後で使うための別のP値の定義

function pvalue_sterne(k, n, p)
    bin = Binomial(n, p)
    # 次はnaiveな定義で計算効率は非常に悪い
    sum(pdf(bin, i) for i in support(bin) if pdf(bin, i) ≲ pdf(bin, k))
end

function pvalue_wald(k, n, p)
    phat = k/n
    z = (k - n*p)/√(n*phat*(1-phat))
    2ccdf(Normal(0, 1), abs(z))
end
```

Out[8]:

```
pvalue_wald (generic function with 1 method)
```

In [9]:

```
n, k, a = 100, 30, 1/2
xlim = (0.1, 0.5)
P1 = plot(p -> pvalue_cp(k, n, p), xlim...; label="Clopper-Pearson")
P2 = plot(p -> pvalue_wilson(k, n, p), xlim...; label="Wilson", c=2, ls=:dash)
P3 = plot(p -> pvalue_bayes(k, n, p; a), xlim...; label="Bayes(a=$a)", c=3, ls=:dashdot)
P4 = plot(p -> pvalue_cp(k, n, p), xlim...; label="Clopper-Pearson")
plot!(p -> pvalue_wilson(k, n, p), xlim...; label="Wilson", c=2, ls=:dash)
plot!(p -> pvalue_bayes(k, n, p; a), xlim...; label="Bayes(a=$a)", c=3, ls=:dashdot)

plot(P1, P2, P3, P4; size=(800, 500), plot_title="data: n = $n, k = $k")
```

Out[9]:

3つのP値函数が近似的に一致しており, 特にJeffreys事前分布でのBayes版のP値函数は正規分布近似を使って定義されたWilson版のP値函数と非常によく一致している.

In [10]:

```
n, k, a = 100, 30, 1
xlim = (0.1, 0.5)
P1 = plot(p -> pvalue_cp(k, n, p), xlim...; label="Clopper-Pearson")
P2 = plot(p -> pvalue_wilson(k, n, p), xlim...; label="Wilson", c=2, ls=:dash)
P3 = plot(p -> pvalue_bayes(k, n, p; a), xlim...; label="Bayes(a=$a)", c=3, ls=:dashdot)
P4 = plot(p -> pvalue_cp(k, n, p), xlim...; label="Clopper-Pearson")
plot!(p -> pvalue_wilson(k, n, p), xlim...; label="Wilson", c=2, ls=:dash)
plot!(p -> pvalue_bayes(k, n, p; a), xlim...; label="Bayes(a=$a)", c=3, ls=:dashdot)

plot(P1, P2, P3, P4; size=(800, 500), plot_title="data: n = $n, k = $k")
```

Out[10]:

平坦事前分布の場合にも, Bayes版のP値函数はWilson型のP値函数とよく一致している.

以上のグラフを見ても, P値を使う統計学とBayes統計が水と油だと考えることはバカげているように思われる.

1.5 復習: 二項分布とベータ分布の関係の証明

以上を理解できた人は二項分布とベータ分布の累積分布函数の関係を完璧に理解しておくことの価値は大きいと感じると思われる.

以下では, 再度, 二項分布とベータ分布の累積分布函数の関係を証明しておく.

大事なことは何度でも証明しておいた方がよい.

1.5.1 両辺を p で微分して確認する方法

証明したい公式

$$1 - \mathrm{cdf}(\mathrm{Binomial}(n, p), k-1) = \mathrm{cdf}(\mathrm{Beta}(k, n-k+1), p) \quad (k = 1, 2, \ldots, n)$$

の両辺を具体的に書き下すと以下のようになる:

$$\sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)} \quad (k = 1, 2, \ldots, n). \tag{$*$}$$

$p = 0$ のとき, $k \geq 1$ より$(*)$の左辺は 0 になり, $(*)$の右辺も 0 になる.

$(*)$の左辺を p で微分すると,

$$\begin{aligned}
\frac{\partial}{\partial p}((*)\text{の左辺}) &= \sum_{i\geq k}\frac{n!}{i!(n-i)!}ip^{i-1}(1-p)^{n-i} - \sum_{i\geq k}\frac{n!}{i!(n-i)!}(n-i)p^{i}(1-p)^{n-i-1} \\
&= \sum_{i\geq k}\frac{n!}{(i-1)!(n-i)!}p^{i-1}(1-p)^{n-i} - \sum_{i\geq k}\frac{n!}{i!(n-i-1)!}p^{i}(1-p)^{n-i-1} \\
&= \sum_{i\geq k}\frac{n!}{(i-1)!(n-i)!}p^{i-1}(1-p)^{n-i} - \sum_{i\geq k+1}\frac{n!}{(i-1)!(n-i)!}p^{i-1}(1-p)^{n-i} \\
&\frac{n!}{(k-1)!(n-k)!}p^{k-1}(1-p)^{n-k}.
\end{aligned}$$

3つめの等号は2つめの和の項のインデックスの i を $i-1$ で置き換えることによって得られる.

一方, $\Gamma(i+1) = i!$ $(i = 0, 1, 2, \ldots)$より,

$$\frac{1}{B(k, n-k+1)} = \frac{\Gamma(n+1)}{\Gamma(k)\Gamma(n-k+1)} = \frac{n!}{(k-1)!(n-k)!}$$

なので, $(*)$の右辺を p で微分すると,

$$\frac{\partial}{\partial p}((*)\text{の右辺}) = \frac{n!}{(k-1)!(n-k)!}p^{k-1}(1-p)^{n-k}.$$

$(*)$の両辺は $p = 0$ のとき互いに等しく, 導函数も互いに等しい. ゆえに$(*)$の左辺と右辺は等しい.

1.5.2 ベータ分布の一様分布の順序統計量の分布として解釈を使う方法

前節での$(*)$の証明法では公式$(*)$が成立する理由はよくわからないので, この節では一様分布の順序統計量としてベータ分布が解釈できることを使った照明を紹介しよう.

$T_1, \ldots, T_n$ は 0 から 1 のあいだの n 個の独立な一様乱数であるとし, その中で k 番目に小さなものを $T_{(k)}$ と書く($T_{(k)}$ を一様分布の順序統計量と呼ぶ).

このとき, $T_{(k)} \leq p$ となることと, $T_1, \ldots, T_n$ の中に p 以下のものが k 個以上あることは同値である. 各々の T_i が p 以下になる確率は p なので, $T_1, \ldots, T_n$ の中に含まれる p 以下のものの個数は二項分布 $\mathrm{Binomial}(n, p)$ に従う. したがって,

$$(T_{(k)} \leq p\text{となる確率}) = (\text{二項分布 }\mathrm{Binomial}(n,p)\text{ において }k\text{ 以上になる確率}) = \sum_{i\geq k}\binom{n}{i}p^i(1-p)^{n-i}.$$

一方, $t \leq T_{(k)} \leq t + dt$ となる確率は

$$\begin{aligned}
&(t \leq T_{(k)} \leq t + dt\text{となる確率}) \\
&= (T_{(1)} \leq \cdots \leq T_{(k-1)} \leq T_{(k)} \leq t + dt\text{ かつ }t \leq T_{(k)} \leq T_{(k+1)} \leq \cdots \leq T_{(n)}\text{ となる確率}) \\
&\approx (T_{(1)} \leq \cdots \leq T_{(k-1)} \leq t \leq T_{(k)} \leq t + dt \leq T_{(k+1)} \leq \cdots \leq T_{(n)}\text{ となる確率}) \\
&= (T_1, \ldots, T_n\text{ のうち }k-1\text{ 個が }t\text{ 以下で, 1個が }[t, t+dt]\text{ に含まれ, }n-k\text{ 個が }t+dt\text{ 以上の確率}) \\
&= \frac{n!}{(k-1)!1!(n-k)!}t^{k-1}\,dt\,(1-t)^{n-k}
\end{aligned}$$

と近似され, 誤差は dt より高次の微小量になる. $T_{(1)} \leq T_{(2)} \leq \cdots \leq T_{(n)}$ が T_i 達を小さな順序に並べたものになっていることおよび, $n!/((k-1)!1!(n-k)!)$ が n 個の $T_1, \ldots, T_n$ を $k-1$ 個, 1 個, $n-k$ 個にグループ分けする方法全体の個数(多項係数の特別な場合)になっていることに注意せよ. そして,

$$\frac{n!}{(k-1)!1!(n-k)!} = \frac{\Gamma(n+1)}{\Gamma(k)\Gamma(n-k+1)} = \frac{1}{B(k, n-k+1)}$$

なので, $T_{(k)}$ が従う分布の確率密度函数はベータ分布 $\mathrm{Beta}(k, n-k+1)$ の密度函数

$$p(t) = \frac{t^{k-1}(1-t)^{n-k}}{B(k, n-k+1)}$$

に一致することがわかる. したがって,

$$(T_{(k)} \leq p\text{となる確率}) = \int_0^p p(t)\,dt = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)}.$$

これを上で示した結果と比較することによって次が得られる:

$$\sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} = \frac{\int_0^p t^{k-1}(1-t)^{n-k}\,dt}{B(k, n-k+1)}.$$

2 P値函数について

P値函数に関する考え方の簡潔な説明については次の論文を参照せよ:

- Valentin Amrhein and Sander Greenland, Discuss practical importance of results based on interval estimates and p-value functions, not only on point estimates and null p-values, Journal of Information Technology, First Published June 3, 2022. [doi (https://doi.org/10.1177%2F02683962221105904)]

2.1 P値はモデルのパラメータ値とデータの数値の整合性の指標の1つ

我々は, データの数値の生成のされ方(未知の法則)をパラメータ付きの確率分布でモデル化し, それを統計モデルと呼ぶのであった.

例えば, 「n 回中 k 回当たりが出た」の形のデータの生成のされ方の統計モデルの代表例は二項分布モデル $\mathrm{Binomial}(n, p)$ であり, 当たりが出る確率は成功確率パラメータ p でモデル化される.

以下では, データの数値 x の生成のされ方をパラメータ θ を持つ統計モデル $M(\theta)$ でモデル化している状況を考える.

そしてさらに, データの数値 x に関する仮説 $\theta = \theta_0$ のP値 $\mathrm{pvalue}(x|\theta = \theta_0)$ が適切に定義されているという一般的な状況も考える.

P値は, データの数値 x と仮説 $\theta = \theta_0$ の下での統計モデル $M(\theta = \theta_0)$ の整合性の指標の1つである.

ここでの整合性は英語では compatibility (両立性)や consititency (無矛盾性)という意味である.

例えば, 二項分布モデルの場合には, 「100 回中 40 回当たりが出た」というデータが得られたとき, 仮説 $p = 1/2$ のP値(Clopper-Pearsonの信頼区間を与えるP値)は約 5.7% になる.

その 5.7% というP値は「100 回中 40 回当たりが出た」というデータと「当たりが出る確率は $p = 1/2$ である」という仮説の整合性の指標として使われることになる.

ただし, **統計モデルの妥当性に注意を払う必要がある.**

P値の定義の仕方によって, その値は微妙に異なった値になる.

そういう細かな違いに結果が影響されないような頑健な議論を目指すべきである.

In [11]:
```
@show k, n, p = 40, 100, 1/2
@show pvalue_cp(k, n, p)
@show pvalue_sterne(k, n, p)
@show pvalue_wilson(k, n, p)
@show pvalue_wald(k, n, p);
```

```
(k, n, p) = (40, 100, 1 / 2) = (40, 100, 0.5)
pvalue_cp(k, n, p) = 0.05688793364098078
pvalue_sterne(k, n, p) = 0.05688793364098067
pvalue_wilson(k, n, p) = 0.04550026389635841
pvalue_wald(k, n, p) = 0.041226833337163676
```

2.2 P値は信頼区間を考えたいすべてのパラメータ値について定義されている

二項分布モデルの場合には仮説 $p = 1/2$ に限らず, 0 から 1 のあいだの任意の数値 p_0 に関する仮説 $p = p_0$ についてP値が定義されている.

それによって, 「100 回中 40 回当たりが出た」というデータが得られたとき, 仮説 $p = p_0$ における p_0 を動かしながらP値を計算することによって, $p = 1/2$ の場合以外についても仮説 $p = p_0$ と「100 回中 40 回当たりが出た」というデータの整合性達がどうなっているかを知ることができる.

例えば, 仮説 $p = 0.3$ とデータの数値の整合性を知ることができる.

In [12]:
```
@show k, n, p = 40, 100, 0.3
@show pvalue_cp(k, n, p)
@show pvalue_sterne(k, n, p)
@show pvalue_wilson(k, n, p)
@show pvalue_wald(k, n, p);
```

```
(k, n, p) = (40, 100, 0.3) = (40, 100, 0.3)
pvalue_cp(k, n, p) = 0.04197715200784929
pvalue_sterne(k, n, p) = 0.03745142924579381
pvalue_wilson(k, n, p) = 0.029096331741252188
pvalue_wald(k, n, p) = 0.041226833337163676
```

一般に, パラメータ値 θ_0 に対して, データ x に関する仮説 $\theta = \theta_0$ のP値を対応させる函数 $\theta_0 \mapsto \mathrm{pvalue}(x|\theta = \theta_0)$ を **P値函数** (P-value function)と呼ぶ.

P値函数は, 統計モデルについて, データの数値とパラメータ値のあいだの整合性(相性)の様子全体の情報を持っている重要な函数である.

以下では, 二項分布モデルにおける4種のP値函数のグラフをプロットしてみよう.

In [13]:
```
k, n = 40, 100
xlim = (0.2, 0.6)
P1 = plot(p -> pvalue_cp(k, n, p), xlim...;
    label="", title="Clopper-Pearson (n = $n, k = $k)")
P2 = plot(p -> pvalue_sterne(k, n, p), xlim...;
    label="", title="Sterne (n = $n, k = $k)", c=2)
P3 = plot(p -> pvalue_wald(k, n, p), xlim...;
    label="", title="Wald (n = $n, k = $k)", c=3)
P4 = plot(p -> pvalue_wilson(k, n, p), xlim...;
    label="", title="Wilson (n = $n, k = $k)", c=4)
plot(P1, P2, P3, P4; xguide="p = p₀", yguide="P-value", size=(800, 500))
```

Out[13]:

近似的にはどれもほぼ同じ形をしている.

2.3 分割表の場合のP値函数

「Aでは $m = a + b$ 回中 a 回当たりが出て, Bでは $n = c + d$ 回中 c 回当たりが出た」の形式のデータを

	当たり	はずれ	合計
A	a	b	m
B	c	d	n

のように 2×2 の分割表で表すのであった.

このようなデータの生成のされ方の統計モデルとして, 2つの二項分布モデル $\mathrm{Binomial}(m, p) \times \mathrm{Binomial}(n, q)$ を採用しよう.

このモデル内で, a の値は二項分布 $\mathrm{Binomial}(m, p)$ に従ってランダムに決まり, c の値はそれとは独立に二項分布 $\mathrm{Binomial}(n, q)$ に従ってランダムに決まっていると考える. ($m = a + b$ と $n = c + d$ は固定されているので, a, c から b, d が自動的に決まる.)

応用時に重要になるのは, p と q の違いである.

入門的な多くの教科書では非常に残念なことに, 仮説 $p = q$ のP値の定義しか書かれておらず, p と q の違いの大きさを表す指標の信頼区間の定義が書かれていない.

これが前節で, **P値は信頼区間を考えたいすべてのパラメータ値について定義されている** と強調したくなった理由である.

しかし, 我々はすでに

- [「検定と信頼区間: 比率の比較」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)

で p と q の違いを表す3つの指標についてP値と信頼区間を定義している:

- オッズ比パラメータ $\mathrm{OR} = \dfrac{p/(1-p)}{q/(1-q)} = \dfrac{p(1-q)}{(1-p)q}$,
- リスク比パラメータ $\mathrm{RR} = \dfrac{p}{q}$,
- リスク差パラメータ $\mathrm{RD} = p - q$.

これは, このノート群が入門的な教科書の多くと一線を画する点である.

この点に関しては, Rothmanさん達の有名な疫学の教科書

- Rothman, Lash, and Greenland, Modern Epidemiology [Googleで検索 (https://www.google.com/search?q=Rothman+Lash+Greenland+Modern+Epidemiology)]

が詳しい. 講義動画

- 佐藤俊哉, 臨床研究者のための生物統計学「回帰モデルと傾向スコア」2019年2月21日 [YouTube (https://youtu.be/cOHN444kBlo)]

も参照せよ. 8:30以降の「変換しない場合」がリスク差パラメータを p と q の違いの指標として採用した場合に対応しており, 10:30以降の「対数変換の場合」がリスク比パラメータを p と q の違いの指標として採用した場合に対応しており, 12:00以降の「ロジット変換の場合」がオッズ比パラメータ(正確にはその対数)を p と q の違いの指標として採用した場合に対応している. 「ロジット変換の場合」はロジスティック回帰の話になっている. ただし, 複数の分割表のデータを扱うより複雑な場合を扱っている.

注意: 実際にはオッズ比パラメータの代わりに対数オッズ比パラメータ $\log \mathrm{OR}$ を扱うことが多い. その場合はロジスティック・モデルの特別な場合になっている. **注意終**

2.4 P値函数から信頼区間が定義される

P値が信頼区間を考えたいすべてのパラメータ値について定義されていることの利点は, 言うまでもないことだが, 信頼区間を定義できることである.

一般に, データ x の生成のされ方に関する統計モデル $M(\theta)$ に関するP値函数 $\mathrm{pvalue}(x|\theta = \theta_0)$ が与えられているとき, パラメータ θ に関する信頼度 $1 - \alpha$ の信頼区間 ($100(1-\alpha)\%$ 信頼区間, confidence interval)は次のように定義される:

$$\mathrm{confint}^{\theta}(x|\alpha) = \{\, \theta_0 \mid \mathrm{pvalue}(x|\theta = \theta_0) \geq \alpha \,\}.$$

すなわち, データの数値 x について仮説 $\theta = \theta_0$ のP値が α 以上になるような θ_0 全体の集合が信頼区間になる.

ここで使われている閾値 α は **有意水準** (significance level)と呼ばれる.

$1 - \alpha$ は **信頼度** (信頼水準, confidence level)と呼ばれる.

コンピュータで計算するときには, $\mathrm{pvalue}(x|\theta = \theta_0) = \alpha$ を満たす θ_0 の値を2分法やNewton法などで計算すれば, 一般的な場合にも信頼区間を概ね計算できる.

しかし, それだとバグも発生し易くなるし, 計算効率も悪くなりがちなので, 信頼区間を直接できるシンプルな公式がある場合にはそちらを使うことになる.

2.5 2×2の分割表のP値と信頼区間とP値函数の例

以下では2×2の分割表のデータの数値

	当たり	はずれ
A	$a = 30$	$b = 70$
B	$c = 20$	$d = 80$

に関するP値と信頼区間の例を示そう.

簡単のためオッズ比パラメータ OR を扱う場合(実際には対数オッズ比パラメータ $\log \mathrm{OR}$ を扱う場合)のWald型のP値函数のみを扱う.

$a + b = m$, $c + d = n$ が十分に大きければ, 仮説 $\mathrm{OR} = \dfrac{p/(1-p)}{q/(1-q)} = \omega$ の統計モデル内で, モデル内での仮想的データ a, b, c, d の数値の対数オッズ比

$$\log \widehat{\mathrm{OR}} = \log \frac{ad}{bc}$$

が次のように近似的に正規分布に従うことを示せる:

$$\log \widehat{\mathrm{OR}} \sim \mathrm{Normal}\left(\log \omega, \widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right), \text{ approximately.}$$

ここで,

$$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}} = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$$

この近似については

- [「検定と信頼区間: 比率の比較」のノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/11%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20proportions.ipynb)

の「Wald版のオッズ比に関するP値と信頼区間の定義」に関する節を参照せよ.

これによって, 以下のように仮説 $\mathrm{OR} = \omega$ のP値と OR の信頼区間が定義できる:

$$\begin{aligned}
\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d|\, \mathrm{OR} = \omega) &= 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{\left|\log \widehat{\mathrm{OR}} - \log \omega\right|}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}}\right)\right), \\
\mathrm{confint}^{\mathrm{OR}}_{\mathrm{Wald}}(a, b, c, d|\alpha) &= \left[\exp\left(-z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right)\widehat{\mathrm{OR}},\ \exp\left(z_{\alpha/2}\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}\right)\widehat{\mathrm{OR}}\right].
\end{aligned}$$

ここで, $z_{\alpha/2} = \mathrm{quantile}(\mathrm{Normal}(0,1), 1 - \alpha/2)$.

注意: $\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d|\, \mathrm{OR} = \omega)$ の定義の右辺は, 標準正規分布内で

$$\frac{\log \widehat{\mathrm{OR}} - \log \omega}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}}$$

の絶対値以上の(データの数値以上極端な)数値が生成される確率の2倍になっている.

In [14]:

```julia
# 上の定義通りのP値函数と信頼区間函数の実装

oddsratiohat(a, b, c, d) = safediv(a*d, b*c)
stderrhat_logoddsratiohat(a, b, c, d) = √(1/a + 1/b + 1/c + 1/d)

function pvalue_or_wald(a, b, c, d; ω=1)
    logORhat = log(oddsratiohat(a, b, c, d))
    SEhat_logORhat = stderrhat_logoddsratiohat(a, b, c, d)
    2ccdf(Normal(0, 1), safediv(abs(logORhat - log(ω)), SEhat_logORhat))
end

function confint_or_wald(a, b, c, d; α=0.05)
    z = quantile(Normal(), 1-α/2)
    ORhat = oddsratiohat(a, b, c, d)
    SEhat_logORhat = stderrhat_logoddsratiohat(a, b, c, d)
    [safemul(exp(-z*SEhat_logORhat), ORhat), safemul(exp(z*SEhat_logORhat), ORhat)]
end
```

Out[14]: confint_or_wald (generic function with 1 method)

In [15]:

```julia
# グラフのプロット用の函数
# この手の函数は計算用の函数よりも複雑になりがち

function plot_pvalue_function_of_or_wald(;
        a=30, b=70, c=20, d=80, ω = 1.0, α = 0.05,
        xlim = confint_or_wald(a, b, c, d; α=1e-3)
    )
    @show α
    @show ω
    @show z = quantile(Normal(0, 1), 1 - α/2)
    println()
    @show ORhat = oddsratiohat(a, b, c, d)
    @show exp(z *stderrhat_logoddsratiohat(a, b, c, d))
    println()
    @show P_value = pvalue_or_wald(a, b, c, d; ω)
    @show CI = confint_or_wald(a, b, c, d; α)

    plot(ω -> pvalue_or_wald(a, b, c, d; ω), xlim...;
        label="P-value function (Wald)", c=:black)
    vline!([ORhat]; label="ORhat = $(round(ORhat; digits=5))", ls=:dash, c=2)
    plot!(CI, fill(α, 2); label="$(100(1-α))% confidence interval", lw=2, c=3)
    scatter!([ω], [α]; label="significance level α = $(100α)%", c=3, msc=:auto)
    vline!([ω]; label="", c=:black, lw=0.5)
    scatter!([ω], [P_value]; label="P-value of OR = $ω", c=:red, msc=:auto)
    plot!(xguide="OR = ω     (log scale)", yguide="P-value")
    plot!(xscale=:log, xtick=logtick(; xlim), ytick=0:0.05:1)
    title!("a, b, c, d = $a, $b, $c, $d")
    plot!(size=(720, 350), bottommargin=4Plots.mm)
end
```

Out[15]: plot_pvalue_function_of_or_wald (generic function with 1 method)

In [16]:
```
a, b, c, d = 30, 70, 20, 80
graph_of_pvalue_function =
    plot_pvalue_function_of_or_wald(; a, b, c, d, ω = 1.0, α = 0.05)
```

```
α = 0.05
ω = 1.0
z = quantile(Normal(0, 1), 1 - α / 2) = 1.9599639845400576

ORhat = oddsratiohat(a, b, c, d) = 1.7142857142857142
exp(z * stderrhat_logoddsratiohat(a, b, c, d)) = 1.9163037024703367

P_value = pvalue_or_wald(a, b, c, d; ω) = 0.10432099005636014
CI = confint_or_wald(a, b, c, d; α) = [0.8945793467266185, 3.28509206137772]
```

Out[16]:

仮説 $\mathrm{OR} = 1$ すなわち「AとBで当たりが出る確率は同じ」という仮説のP値は 10.4% 程度になっている.

これが意味するところは以下の通り. データの数値

	当たり	はずれ
A	$a = 30$	$b = 70$
B	$c = 20$	$d = 80$

を見ると, Aの方がBよりも当たりが出る確率が高そうに見えるが, 統計モデル内ではAとBで当たりが出る確率が同じであったとしても, モデル内でこのデータの数値以上に偏った値が生じる確率は約 10.4% もある.

さらに, 上のグラフを見ると, 仮説 $\mathrm{OR} = 4$ のP値は 1% 程度に見えるが, 実際に計算してもその程度の値になる.

In [17]:
```
pval4 = pvalue_or_wald(a, b, c, d; ω=4)
```

Out[17]:
```
0.010670206411228012
```

WolframAlphaでも仮説 $\mathrm{OR} = 4$ のP値を計算してみよう.

$\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}$ → `sqrt(1/a + 1/b + 1/c + 1/d) where a=30.0, b=70, c=20, d=80` → [実行 (https://www.wolframalpha.com/input?i=sqrt%281%2Fa+%2B+1%2Fb+%2B+1%2Fc+%2B+1%2Fd%29+where+a%3D30.0%2C+b%3D70%2C+c%3D20%2C+d%3D80)] → 0.331842

$\log \widehat{\mathrm{OR}} - \log \omega \to$ `log((a d)/(b c)) - log(4) where a=30.0, b=70, c=20, d=80` → [実行 (https://www.wolframalpha.com/input?i=log%28%28a+d%29%2F%28b+c%29%29+-+log%284%29+where+a%3D30.0%2C+b%3D70%2C+c%3D20%2C+d%3D80)] → -0.847298

$(\log \widehat{\mathrm{OR}} - \log \omega)/\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}} \to$ `0.847298/0.331842` → [実行 (https://www.wolframalpha.com/input?i=0.847298%2F0.331842)] → 2.55332

P値 → `2(1 - cdf(NormalDistribution(0,1), 2.55332))` → [実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28NormalDistribution%280%2C1%29%2C+2.55332%29%29)] → 0.0106701

In [18]:
```
# WolframAlphaで求めたP値のJuliaで求めたP値に対する相対誤差が小さいことの確認
0.0106701/pval4 - 1
```

Out[18]:
```
-9.972743160724384e-6
```

WolframAlphaでも OR の 95% 信頼区間を求めてみよう.

$z_{\alpha/2} \to$ `quantile(Normal(0,1), 0.975)` → [実行 (https://www.wolframalpha.com/input?i=quantile%28Normal%280%2C1%29%2C+0.975%29)] → 1.95996

信頼区間 → `{exp(-1.95996*0.331842) (a d)/(b c),  exp(1.95996*0.331842) (a d)/(b c)} where a=30.0, b=70, c=20, d=80` → [実行 (https://www.wolframalpha.com/input?i=%7Bexp%28-1.95996*0.331842%29+%28a+d%29%2F%28b+c%29%2C++exp%281.95996*0.331842%29+%28a+d%29%2F%28b+c%29%7D+w
→ {0.89458, 3.28509}

In [19]:
```
# WolframAlphaで求めた信頼区間のJuliaで求めた信頼区間に対する相対誤差が小さいことの確認
[0.89458, 3.28509] ./ confint_or_wald(a, b, c, d; α=0.05) .- 1
```

Out[19]:
```
2-element Vector{Float64}:
  7.302576165990615e-7
 -6.274946581230623e-7
```

2.6 2×2の分割表での検定ではどれを使うべきか

以上においては計算の仕方がシンプルだという理由でWald型のP値函数と信頼区間を扱ったが, 標本サイズが小さいときに誤差が大きくなるという欠点がある. 他にも以下の選択肢がある:

(1) Pearsonの χ^2 検定 (ただし扱う仮説を $\mathrm{OR} = \omega$ と $\mathrm{RR} = \rho$ の場合に一般化しておくこと, 連続性補正はかけない)

(2) Fisher検定 (ただし扱う仮説を $\mathrm{OR} = \omega$ の場合に一般化しておくこと)

(3) p と q の違いの指標としてそれらの差 $p - q$ を扱う場合には Zou-Donner 2004 の方法が使える. (Julia言語による実装例 (https://github.com/genkuroki/public/blob/main/0033/probability%20of%20alpha%20error%20of%20Zou-Donner.ipynb))

それぞれ長所と短所があるので目的に合わせて適当に使い分ければよい. 標本サイズが十分に大きい場合にはWald型のP値函数と信頼区間も十分な精度を持ち, 問題なく使える.

注意: よく使われているのは, Pearsonの χ^2 検定とFisher検定である. 標本サイズが小さい場合にはFisher検定を使えと解説されている場合もあるようだが, 実際に色々計算するとそれは誤りだと分かる. 実際には標本サイズが小さなとき, Fisher検定には過剰にP値が大きくなり過ぎるという欠点がある. しかし, その欠点は「第一種の過誤の確率が常に有意水準以下である」というFisher検定の優れた性質の裏返しなので, どの条件を重視するかによって, (Yatesの連続性補正をかけない)Pearsonの χ^2 検定と使い分けるとよいと思われる. 次の解説も参考になる:

- 朝倉こう子, 濱﨑俊光, 連載 第3回　医学データの統計解析の基本　2 つの割合の比較　[PDF (https://www.jstage.jst.go.jp/article/dds/30/3/30_235/_pdf)]

第一種の過誤の確率を確実に名目有意水準以下にしたければFisher検定を使う方がよいし, 平均的により高い検出力を得たいならば(連続補正されていない)Pearsonの χ^2 検定を使えばよいだろう.

2.7 P値函数と最尤法の関係

2×2の分割表のデータの数値

	当たり	はずれ
A	$a = 30$	$b = 70$
B	$c = 20$	$d = 80$

に関するP値函数などは1つのグラフの中に以下のようにプロットされるのであった.

In [20]:
```
graph_of_pvalue_function
```

Out[20]:

このグラフにおけるデータの数値のオッズ比

$$\widehat{\mathrm{OR}} = \frac{ad}{bc} = \frac{30 \cdot 80}{70 \cdot 20} = \frac{12}{7} \approx 1.7142857142857142$$

の値でP値函数が最大値の 1 になっていることに注意せよ. P値函数の定義

$$\mathrm{pvalue}_{\mathrm{Wald}}(a, b, c, d \mid \mathrm{OR} = \omega) = 2\left(1 - \mathrm{cdf}\left(\mathrm{Normal}(0,1), \frac{\left|\log \widehat{\mathrm{OR}} - \log \omega\right|}{\widehat{\mathrm{SE}}_{\log \widehat{\mathrm{OR}}}}\right)\right)$$

を見れば, $\omega = \widehat{\mathrm{OR}}$ のとき最大値 1 になることはすぐにわかる.

実はこのデータの数値のオッズ比 $\mathrm{OR} = (ad)/(bc)$ は2つの二項分布モデルにおける最尤法(さいゆうほう)の解

$$\hat{p} = \frac{a}{a+b}, \quad \hat{q} = \frac{c}{c+d} \quad \left(\text{このとき } 1 - \hat{p} = \frac{b}{a+b}, \quad 1 - \hat{q} = \frac{d}{c+d}\right)$$

のオッズ比になっている:

$$\frac{\hat{p}(1-\hat{q})}{(1-\hat{p})\hat{q}} = \frac{ad}{bc} = \widehat{\mathrm{OR}}.$$

この場合に限らず, **多くの場合にP値函数が最大になるパラメータ値は最尤法の解に対応する値になっている.**

少なくとも, 近似的にはP値函数を最大化するパラメータ値は最尤法の解に対応する値だと考えてよい.

データの数値 x に関する仮説 $\theta = \theta_0$ のP値は, 統計モデルを前提にしたときの仮説 $\theta = \theta_0$ とデータの数値 x の整合性の指標の1つであった.

ゆえに, 以上で述べたことは, **最尤法の解に対応するパラメータ値はデータの数値との整合性が最も高いパラメータ値である** とみなせる.

最尤法による点推定にはこのような解釈がある.

注意・警告: 統計学を実践的に使う場合には, 最尤法による点推定の結果のみを報告せずに, 信頼区間などの区間推定の結果も報告する習慣になっている. 点推定の値は単に統計モデルの前提の下でデータの数値と最も整合性が高いパラメータ値を計算したにすぎず, 確率的な揺らぎが考慮されていない. 確率的揺らぎの影響に配慮した区間推定の結果も報告しないと, 点推定の値がどこまで正確な推定値だと考えられるかが全くわからなくなってしまう.

注意: 尤度(ゆうど, likelihood)については

- 「条件付き確率分布, 尤度, 推定, 記述統計」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/06%20Conditional%20distribution%2C%20likelihood%2C%20

における「尤度 (ゆうど)と推定」の節を参照せよ. 尤度の定義は **統計モデル内でデータと同じ数値が生成される確率もしくは確率の密度** であり, モデルのデータの数値への適合度の指標としては使えるが, 「もっともらしさ」の指標としては不適切であることに注意せよ.

2.8 以上のまとめの図

以下の図のように, P値函数全体の形を見れば多くのことを一目で確認できる.

仮説 $\theta = 0$ のP値のみ, 点推定値のみ, 信頼区間のみをP値函数のグラフ抜きにばらばらに描くと以下のようになり, 全体の関係がどうなっているかがひどく分かり難くなる.

注意: Bayes統計における事後分布も以上で説明したP値函数とほぼ同じ使い方をできる. P値函数について十分に理解しておけば, Bayes統計について学習するときにも役に立つだろう.

2.9 おまけ: Bayes統計の方法を使った場合

このノート群の中ではできるだけBayes統計の方法には触れないように気を使ったが, 何も触れないわけには行かなかった. 現代においては, P値函数を使う方法だけではなく, Bayes統計の方法も「統計学の道具箱」の中に入れておいた方がよいと思われる.

2つの二項分布モデル $\mathrm{Binomial}(m, p) \times \mathrm{Binomial}(n, q)$ の事前分布として, 2つのJeffreya事前分布 $\mathrm{Beta}(1/2, 1/2) \times \mathrm{Beta}(1/2, 1/2)$ を採用する.

このとき, 2×2の分割表のデータの数値

	当たり	はずれ
A	$a = 30$	$b = 70$
B	$c = 20$	$d = 80$

から得られる事後分布は, $\mathrm{Beta}(30 + 1/2, 70 + 1/2) \times \mathrm{Beta}(20 + 1/2, 80 + 1/2)$ になる.

In [21]:
```
a, b, c, d = 30, 70, 20, 80
posterior = product_distribution([Beta(a+1/2, b+1/2), Beta(c+1/2, d+1/2)])
```

Out[21]:
```
Product{Continuous, Beta{Float64}, Vector{Beta{Float64}}}(v=Beta{Float64}[Beta{Float64}(α=30.5,
β=70.5), Beta{Float64}(α=20.5, β=80.5)])
```

事後分布のサイズ100万のサンプルを作る.

In [22]:
```
sample_of_posterior = rand(posterior, 10^6)
```

Out[22]:
```
2×1000000 Matrix{Float64}:
 0.324564  0.249517  0.279684  0.293979  0.26615   …  0.264104  0.267631  0.360962  0.289852
 0.222686  0.16415   0.198842  0.255289  0.244109     0.193823  0.222817  0.218     0.202151
```

事後分布でのオッズ比のサンプルを構成する.

In [23]:
```
@views p, q = sample_of_posterior[1,:], sample_of_posterior[2,:]
sample_of_posterior_odds_ratio = @. (p*(1-q))/((1-p)*q)
sample_of_posterior_odds_ratio'
```

Out[23]:
```
1×1000000 adjoint(::Vector{Float64}) with eltype Float64:
 1.67733  1.69297  1.56442  1.21465  1.12304  …  2.65816  1.49274  1.27462  2.02621  1.61092
```

オッズ比の点推定値として, 事後分布での中央値を採用しよう.

In [24]:
```
ORhat_posterior_median = median(sample_of_posterior_odds_ratio)
```

Out[24]:
```
1.7117354764696162
```

In [25]:
```
ORhat = oddsratiohat(a, b, c, d) # 最尤推定値
```

Out[25]:
```
1.7142857142857142
```

最尤推定値の ORhat との違いは小さい.

In [26]:
```
ORhat_posterior_median / ORhat - 1
```

Out[26]:
```
-0.0014876387260571722
```

オッズ比の区間推定として, 事後分布での 2.5% と 97.5% 分位点の間を採用しよう.

これを 95% 信用区間(確信区間, credible interval)と呼ぶ.

In [27]:
```
credible_interval = quantile.(Ref(sample_of_posterior_odds_ratio), [0.025, 0.975])
print(credible_interval)
```

```
[0.8986357079398595, 3.3064452661370107]
```

In [28]:
```
confidence_interval = confint_or_wald(a, b, c, d; α=0.05)
print(confidence_interval)
```

```
[0.8945793467266185, 3.28509206137772]
```

信頼区間(confidence interval)と信用区間(credible interval)の違いは小さい.

In [29]:
```
credible_interval ./ confidence_interval .- 1 ▷ print
```

```
[0.004534378340036227, 0.0065000323766681856]
```

オッズ比パラメータが 1 という仮説のP値のBayes的代替物として, 事後分布で 1 以下になる確率と 1 以上になる確率の小さい方の2倍(と 1 の小さい方)を採用する. (これはClopper-Pearson型のP値函数の定義の類似になっている.)

In [30]:
```
ecdf_OR = ecdf(sample_of_posterior_odds_ratio)
pvalue_or_bayesian = ω → min(1, 2ecdf_OR(ω), 2(1 - ecdf_OR(ω)))
P_value_Bayesian = pvalue_or_bayesian(1)
```

Out[30]:
```
0.101846
```

In [31]:
```
Pvalue_Wald = pvalue_or_wald(a, b, c, d; ω=1)
```

Out[31]: 0.10432099005636014

通常のP値とそのBayes的類似の違いは小さい.

In [32]:
```
P_value_Bayesian/Pvalue_Wald - 1
```

Out[32]: -0.023724756207001096

以上をまとめてプロットしてみよう.

In [33]:
```
ORs = sample_of_posterior_odds_ratio
CI = credible_interval
α = 0.05

xlim = quantile.(Ref(ORs), (1e-4, 1-1e-4))
bin = range(xlim..., 201)
h = Plots._make_hist((ORs,), bin, normed=true)
y = h.weights

plot(bin, y; seriestype=:stepbins)
stephist(ORs; norm=true, bin,
    label="posterior of odds ratio")
vline!([ORhat_posterior_median];
    label="median of posterior = $(round(ORhat_posterior_median; digits=5))", ls=:dash, c=2)
plot!(xguide="OR = ω", yguide="probability density")
plot!(xtick=0:0.5:10, ytick=0:0.05:1)
title!("data: a, b, c, d = $a, $b, $c, $d,     prior = Jeffreys")
plot!(size=(720, 350))

imin = findlast(i -> ecdf_OR(bin[i]) < 0.025, eachindex(bin))
imax = findlast(i -> ecdf_OR(bin[i]) > 0.975, reverse(eachindex(bin)))
@show ecdf_OR(bin[imin]), ecdf_OR(bin[imax])

plot!(CI, fill(α, 2); label="$(100(1-α))% credible interval (Bayesian CI)", lw=2, c=3)
plot!(bin[begin:imin], y[begin:imin-1]; seriestype=:stepbins,
    label="probability = 2.5%", c=4, fillrange=0, alpha=0.5)
plot!(bin[imax:end], y[imax:end]; seriestype=:stepbins,
    label="probability = 2.5%", c=5, fillrange=0, alpha=0.5)
```

```
(ecdf_OR(bin[imin]), ecdf_OR(bin[imax])) = (0.024634, 0.975766)
```

Out[33]:

In [34]:

```
plot(bin, y; seriestype=:stepbins)
stephist(ORs; norm=true, bin,
    label="posterior of odds ratio")
vline!([ORhat_posterior_median];
    label="median of posterior = $(round(ORhat_posterior_median; digits=5))", ls=:dash, c=2)
plot!(xguide="OR = ω", yguide="probability density")
plot!(xtick=0:0.5:10, ytick=0:0.05:1)
title!("data: a, b, c, d = $a, $b, $c, $d,     prior = Jeffreys")
plot!(size=(720, 350))

inull = findlast(i -> bin[i] < 1, eachindex(bin)) + 1

plot!(bin[begin:inull], y[begin:inull-1]; seriestype=:stepbins,
    label="half of P-value = $(round(P_value_Bayesian; digits=5)) / 2", c=6, fillrange=0, alph
```

Out[34]:

通常のP値函数(Wald)とBayes的なP値函数の類似物のグラフを同時に描くとほぼぴったり重なる.

In [35]:

```
a, b, c, d = 30, 70, 20, 80
plot(ω → pvalue_or_wald(a, b, c, d; ω);
    label="ordinary P-value function (Wald)")
plot!(ω → pvalue_or_bayesian(ω);
    label="Bayesian analogue of P-value function", ls=:dash)
plot!(xguide="OR = ω", yguide="P-value")
plot!(xtick=0:0.5:10, ytick=0:0.05:1)
title!("data: a, b, c, d = $a, $b, $c, $d,     prior = Jeffreys")
plot!(size=(720, 350))
```

Out[35]:

サンプルサイズがもっと小さいな場合にはここまでぴったり一致しないが, この場合には非常によく一致している.

この一致を見てしまった人が, 「ベイズ統計とP値を使う統計学は水と油である」のような考え方をすることは難しいように思われる.

以上によって, 通常のP値を使った統計分析をBayes統計の方法でどのようにすれば置き換えることができるかがわかった.

Bayes統計の方法では事前分布(prior)という新しい道具を使えるので, 別の新しいことも可能になる.

次の論文では以上と本質的に同じ方法をオッズ比ではなくリスク比の場合に適用して, Bayes統計の方法を使って統計分析を行っている:

- https://www.nejm.org/doi/full/10.1056/NEJMoa2115869 (https://www.nejm.org/doi/full/10.1056/NEJMoa2115869)

この論文ではBayes統計の方法を使っているが, 通常のP値を使う方法でも実質的に同じ結果が得られる.

3 Welchの t 検定について

2つの群の平均の差に関するWelchの t 検定に付随するP値と信頼区間については

- 「検定と信頼区間: 平均の比較」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb)

で非常に詳しく解説したが, 手続きが複雑なので以下で再度まとめておくことにする.

3.1 Welchの t 検定のP値と信頼区間の定義

データは別の母集団Aから抽出されたサイズ m の標本 $x_1, \dots, x_m$ と別の母集団Bから抽出されたサイズ n の標本 $y_1, \dots, y_n$ であると仮定する.

母集団Aからのサイズ m の標本についてその標本平均が従う分布は中心極限定理によって正規分布で十分に近似されていると仮定する. 母集団Bとサイズ n についても同様であると仮定しておく.

母集団Aの(真の)平均を μ_x と書き, 母集団Bの(真の)平均を μ_y と書く.

標本 $x_1, \dots, x_m$ の標本平均を $\bar{x}$ と書き, 標本 $y_1, \dots, y_n$ の標本平均を $\bar{y}$ と書く.

母集団Aから一意的に決まっている定数 μ_x と標本を取り直すごとに確率的に変化する標本平均 $\bar{x}$ を混同しないように注意せよ. μ_y と $\bar{y}$ についても同様に混同しないようにして欲しい.

さらに, 標本 $x_1, \ldots, x_m$ の不偏分散を s_x^2 と書き, 標本 $y_1, \ldots, y_n$ の不偏分散を s_y^2 と書く:

$$s_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2, \quad s_y^2 = \frac{1}{n-1}\sum_{j=1}^{n}(y_j - \bar{y})^2.$$

これらを, 母集団Aの(真の)分散 σ_x^2 と母集団Bの(真の)分散 σ_y^2 と混同しないように注意せよ.

以上の状況で, 任意に与えられた実数 $\Delta\mu$ に対して, $\mu_x - \mu_y = \Delta\mu$ という仮説のP値を定義しよう.

まず, t 統計量 $t = t(\Delta\mu)$ を次のように定義する:

$$t = t(\Delta\mu) = \frac{(\bar{x} - \bar{y}) - \Delta\mu}{\sqrt{\dfrac{s_x^2}{m} + \dfrac{s_y^2}{n}}}.$$

さらに, 自由度と呼ばれる統計量 ν を次のように定義する:

$$\nu = \frac{\left(s_x^2/m + s_y^2/n\right)^2}{\dfrac{(s_x^2/m)^2}{m-1} + \dfrac{(s_y^2/n)^2}{n-1}}.$$

そして, P値を次のように定義する:

$$\mathrm{pvalue}_{\mathrm{Welch}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \mu_x - \mu_y = \Delta\mu) = 2(1 - \mathrm{cdf}(\mathrm{TDist}(\nu), |t(\Delta\mu)|)).$$

ここで, $\mathrm{cdf}(\mathrm{TDist}(\nu), x)$ は自由度 ν の t 分布の累積分布函数(cumulative distribution function, cdf)である.

以上に対応する平均の差 $\mu_x - \mu_y$ に関する信頼度 $1 - \alpha$ の信頼区間は次のようになる:

$$\begin{aligned}&\mathrm{confint}_{\mathrm{Welch}}(\bar{x}, \bar{y}, s_x^2, s_y^2 | m, n, \alpha)\\ &= \left[\bar{x} - \bar{y} - t_{\nu,\alpha/2}\sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}},\ \bar{x} - \bar{y} + t_{\nu,\alpha/2}\sqrt{\frac{s_x^2}{m} + \frac{s_y^2}{n}}\right].\end{aligned}$$

ここで, $t_{\nu,\alpha/2} = \mathrm{quantile}(\mathrm{TDist}(\nu), 1 - \alpha/2)$. ただし, $\mathrm{quantile}(\mathrm{TDist}(\nu), p)$ は自由度 ν の t 分布の分位点函数(quantile function, 累積分布函数の逆函数)である.

注意: 自由度 ν をそのように定義する理由については

- [「検定と信頼区間: 平均の比較」のノート](https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb) (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb)

の「Welchの t 検定で使う自由度の式の導出」の節を参照せよ. さらにその「Studentの t 検定とWelchの t 検定で使用する t 分布の自由度の比較」の節では

$$(m + n - 2) - \nu = \frac{\left(s_x^2/(m(m-1)) - s_y^2/(n(n-1))\right)^2}{\dfrac{(s_x^2/(m(m-1)))^2}{n-1} + \dfrac{(s_y^2/(n(n-1)))^2}{m-1}} \geq 0$$

となることも注意している. これより $\nu \leq m + n - 2$ となっている.

In [36]:

```
@vars s²_x s²_y m n ν
ν = (s²_x/m + s²_y/n)^2/((s²_x/m)^2/(m-1) + (s²_y/n)^2/(n-1))
lhs = (m+n-2) - ν
rhsnum = (s²_x/(m*(m-1)) - s²_y/(n*(n-1)))^2
rhsden = (s²_x/(m*(m-1)))^2/(n-1) + (s²_y/(n*(n-1)))^2/(m-1)
@show (lhs - rhsnum/rhsden).simplify()
Eq(lhs, rhsnum/rhsden)
```

```
(lhs - rhsnum / rhsden).simplify() = 0
```

Out[36]:

$$m + n - \frac{\left(\frac{s^2{}_y}{n} + \frac{s^2{}_x}{m}\right)^2}{\frac{s^2{}_y^2}{n^2(n-1)} + \frac{s^2{}_x^2}{m^2(m-1)}} - 2 = \frac{\left(-\frac{s^2{}_y}{n(n-1)} + \frac{s^2{}_x}{m(m-1)}\right)^2}{\frac{s^2{}_y^2}{n^2(m-1)(n-1)^2} + \frac{s^2{}_x^2}{m^2(m-1)^2(n-1)}}$$

注意: $X_1, \ldots, X_n$ は分散 σ^2 を持つ分布の標本で, $\bar{X}$ はその標本平均であるとする. このとき, 不偏分散を計算するときに n ではなく, $n-1$ で割る理由は単に

$$E\left[\sum_{i=1}^{n}(X_i - \bar{X})^2\right] = (n-1)\sigma^2$$

となるからである. 両辺を n ではなく, $n-1$ で割らないと σ^2 が期待値として得られない. このような事実に関するより深い理解は **線形回帰が直交射影そのものであること** を学べば得られる. 線形回帰に一般化された場合の不偏分散は, n 次元空間の中の r 次元部分空間の直交補空間の次元 $n-r$ で割ることによって得られる. 詳しい説明は

- 「回帰 (regression)」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/14%20Regression.ipynb)

の「βとσ²の不偏推定量」の節にある. 通常の標本の不偏分散は部分空間が1次元の $\{\,(t, t, \ldots, t) \mid t \in \mathbb{R}\,\}$ の場合になっている.

3.2 Welchの t 検定のP値や信頼区間の計算例

以下の例は

- 「検定と信頼区間: 平均の比較」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb#Welch%E3%81%AE-t-%E6%A4%9C%E5%AE%9A%E3%81%AEP%E5%80%A4%E3%81%A8%E4%BF%A1%E9%A0%BC%E5%8C%BA%E9%96%

における

- 必修問題: Welchの t 検定のP値と信頼区間の計算 (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/12%20Hypothesis%20testing%20and%20confidence%20inter%20Two%20means.ipynb#%E5%BF%85%E4%BF%AE%E5%95%8F%E9%A1%8C:-Welch%E3%81%AE-t-%E6%A4%9C%E5%AE%9A%E3%81%AEP%E5%80%A4%E3%81%A8%E4%BF%A1%E9%A0%BC%E5%8C%BA%E9%96%

の再掲である.

サイズが `m = 20`, `n = 30` のデータ

```
x = [19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9,
     11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6]

y = [44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7,
     74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7,
     124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4]
```

について以下を求めよ.

(1) 仮説 $\mu_x - \mu_y = 0$ のP値.

(2) 仮説 $\mu_x - \mu_y = -30$ のP値.

(3) Welchの t 検定に付随する 95% 信頼区間.

3.2.1 WolframAlphaによるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例

WoldramAlphaで標本の標本平均は `mean {a, b, c, …}` のように入力すれば計算できる.

WoldramAlphaで標本の不偏分散は `var {a, b, c, …}` のように入力すれば計算できる.

注意: 統計学の慣習では, 標本の分散は n ではなく $n-1$ で割る方の不偏分散を意味することが多く, ソフトウェア側でもその慣習に従っていることが多い. WoldramAlphaやJuliaやRの `var` はデフォルトで不偏分散を計算するが, Pythonのnumpyの `np.var` はそうではないので注意せよ. `np.var(x, ddof=1)` のように `ddof=1` が必要になる. 詳しくは公式ドキュメントの

- https://numpy.org/doc/stable/reference/generated/numpy.var.html (https://numpy.org/doc/stable/reference/generated/numpy.var.html)

を参照せよ. コンピュータで標本の分散の値を計算する場合にはそのための函数の仕様を確認しておいた方がよい. **注意終**

$\bar{x} \to$ `mean {19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9, 11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6}` → 実行 (https://www.wolframalpha.com/input?i=mean+%7B19.2%2C+22.7%2C+7.8%2C+138.5%2C+70.5%2C+44.3%2C+84.0%2C+35.6%2C+72.4%2C+23.9%2C+11.7%2C+26
→ 59.445

$s_x^2 \to$ `var {19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9, 11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6}` → 実行 (https://www.wolframalpha.com/input?i=var+%7B19.2%2C+22.7%2C+7.8%2C+138.5%2C+70.5%2C+44.3%2C+84.0%2C+35.6%2C+72.4%2C+23.9%2C+11.7%2C+26.6
→ 1448.48

$\bar{y} \to$ `mean {44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7, 74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7, 124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4}` → 実行 (https://www.wolframalpha.com/input?i=mean+%7B44.3%2C+66.9%2C+62.9%2C+78.4%2C+71.2%2C+32.5%2C+111.4%2C+38.2%2C+68.2%2C+50.7%2C+74.5%2C+4
→ 67.5433

$s_y^2 \to$ `var {44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7, 74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7, 124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4}` → 実行 (https://www.wolframalpha.com/input?i=var+%7B44.3%2C+66.9%2C+62.9%2C+78.4%2C+71.2%2C+32.5%2C+111.4%2C+38.2%2C+68.2%2C+50.7%2C+74.5%2C+46.2
→ 479.481

以下において，`x`，`v`，`y`，`w` はそれぞれ $\bar{x}, s_x^2, \bar{y}, s_y^2$ を意味する.

自由度 $\nu \to$ `(v/m+w/n)^2/((v/m)^2/(m-1)+(w/n)^2/(n-1)) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28v%2Fm%2Bw%2Fn%29%5E2%2F%28%28v%2Fm%29%5E2%2F%28m-1%29%2B%28w%2Fn%29%5E2%2F%28n-1%29%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C+y%3D67.5433%2C+w%3D479.481%7
→ 27.4358

(1)

仮説 $\mu_x - \mu_y = 0$ の t 値 → `(x-y-0)/sqrt(v/m+w/n) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y-0%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C+y%3D
→ -0.861294

仮説 $\mu_x - \mu_y = 0$ のP値 → `2(1 - cdf(TDistribution(27.4358), 0.861294))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2827.4358%29%2C+0.861294%29%29) → 0.396541

(2)

仮説 $\mu_x - \mu_y = -30$ の t 値 → `(x-y+30)/sqrt(v/m+w/n) where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%28x-y%2B30%29%2Fsqrt%28v%2Fm%2Bw%2Fn%29+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2C
→ 2.32935

仮説 $\mu_x - \mu_y = -30$ のP値 → `2(1 - cdf(TDistribution(27.4358), 2.32935))` → 実行 (https://www.wolframalpha.com/input?i=2%281+-+cdf%28TDistribution%2827.4358%29%2C+2.32935%29%29) → 0.0274389

(3)

$\alpha = 0.05$ のときの $t_{\nu,\alpha/2} \to$ `quantile(TDistribution(27.4358), 0.975)` → 実行 (https://www.wolframalpha.com/input?i=quantile%28TDistribution%2827.4358%29%2C+0.975%29) → 2.05031

$\mu_x - \mu_y$ の 95% 信頼区間 → `{x-y-2.05031*sqrt(v/m+w/n), x-y+2.05031*sqrt(v/m+w/n)} where {m=20, x=59.445, v=1448.48, n=30, y=67.5433, w=479.481}` → 実行 (https://www.wolframalpha.com/input?i=%7Bx-y-2.05031*sqrt%28v%2Fm%2Bw%2Fn%29%2C+x-y%2B2.05031*sqrt%28v%2Fm%2Bw%2Fn%29%7D+where+%7Bm%3D20%2C+x%3D59.445%2C+v%3D1448.48%2C+n%3D30%2
→ {-27.3763, 11.1797}

3.2.2 Julia言語によるWelchの t 検定のP値と信頼区間の計算の必修問題の解答例

まず, 定義通りにJulia言語のコードで必要な函数を書く.

Julia言語対応環境において

- `x̄` は `x\bar` と入力してタブキーを押せば入力できる.
- `sx²` は `sx\^2` と入力してタブキーを押せば入力できる.
- `Δ` は `\Delta` と入力してタブキーを押せば入力できる.
- `ν` は `\nu` と入力してタブキーを押せば入力できる.
- `√` は `\sqrt` と入力してタブキーを押せば入力できる.

ユニコードの文字をプログラムのコードとして使いたくない人は `x̄` や `sx²` の代わりに `xbar` , `sx2` のように書いても全然問題ない.

以下ではほとんど数式通りにコードが書かれている

In [37]:

```julia
# 確率分布を扱う場合には常に以下を前もって実行しておく.

using Distributions

# t値の計算の仕方 (デフォルトで Δμ = 0 にしておく)

function tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    (x̄ - ȳ - Δμ) / √(sx²/m + sy²/n)
end

function tvalue_welch(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
end

# 自由度 ν の計算の仕方

function degree_of_freedom_welch(m, sx², n, sy²)
    (sx²/m + sy²/n)^2 / ((sx²/m)^2/(m-1) + (sy²/n)^2/(n-1))
end

function degree_of_freedom_welch(x, y)
    m, sx² = length(x), var(x)
    n, sy² = length(y), var(y)
    degree_of_freedom_welch(m, sx², n, sy²)
end

# P値の計算の仕方

function pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ=0)
    t = tvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
    ν = degree_of_freedom_welch(m, sx², n, sy²)
    2ccdf(TDist(ν), abs(t))
end

function pvalue_welch(x, y; Δμ=0)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    pvalue_welch(m, x̄, sx², n, ȳ, sy²; Δμ)
end

# 信頼区間の計算の仕方 (デフォルトでは α = 0.05 にしておく)

function confint_welch(m, x̄, sx², n, ȳ, sy²; α=0.05)
    ν = degree_of_freedom_welch(m, sx², n, sy²)
    c = quantile(TDist(ν), 1-α/2)
    SEhat = √(sx²/m + sy²/n)
    [x̄-ȳ-c*SEhat, x̄-ȳ+c*SEhat]
end

function confint_welch(x, y; α=0.05)
    m, x̄, sx² = length(x), mean(x), var(x)
    n, ȳ, sy² = length(y), mean(y), var(y)
    confint_welch(m, x̄, sx², n, ȳ, sy²; α)
end
```

Out[37]:

```
confint_welch (generic function with 2 methods)
```

以上のコードを何も考えずにコピー&ペーストして使っても勉強にならない.

可能ならば全部自分で書き直してみるべきである.

そのためには, 出て来たビルトイン函数のすべてについて公式ドキュメントを参照した方がよい.

Juliaの場合には公式ドキュメントはほぼ `?` で表示されるドキュメントに等しい.

Julia対応環境では例えば `?TDist` と入力すれば `TDist` のドキュメントを読むことができる.

In [38]:
```
?TDist
```

search: **TDist** **tdist**pdf **tdist**cdf **tdist**ccdf **tdist**logpdf **tdist**logcdf **tdist**invcdf

Out[38]:

\begin{verbatim}
TDist(v)
\end{verbatim}

The \emph{Students T distribution} with \texttt{v} degrees of freedom has probability density function

$$f(x;\nu) = \frac{1}{\sqrt{\nu}B(1/2,\nu/2)}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}$$

\begin{verbatim}
TDist(d) # t-distribution with v degrees of freedom

params(d) # Get the parameters, i.e. (v,)
dof(d) # Get the degrees of freedom, i.e. v
\end{verbatim}

External links \href{https://en.wikipedia.org/wiki/Student%27s_t-distribution}{Student's T distribution on Wikipedia}

In [39]:
```
# x, y に標本の値を代入する.
x = [19.2, 22.7, 7.8, 138.5, 70.5, 44.3, 84.0, 35.6, 72.4, 23.9,
    11.7, 26.6, 73.8, 118.3, 54.2, 57.6, 40.5, 117.4, 102.3, 67.6];
y = [44.3, 66.9, 62.9, 78.4, 71.2, 32.5, 111.4, 38.2, 68.2, 50.7,
    74.5, 46.2, 65.7, 58.7, 42.5, 57.4, 63.0, 67.9, 72.1, 117.7,
    124.1, 48.9, 91.8, 80.8, 60.2, 76.8, 76.3, 59.9, 70.7, 46.4];
```

In [40]:
```
# (1)
pval1 = pvalue_welch(x, y; Δμ = 0)
```

Out[40]: 0.39653998689489345

統計学関係のコードを自分で書いた場合には, 何らかの方法で実装を間違っていないことを確認した方がよい. そのための1つの方法は別の方法で計算して確認することである. 例えばR言語で計算し直すと次のようになる.

In [41]:
```
@rput x y
pval1_R = rcopy(R"""t.test(x, y, mu = 0)""")[:p_value]
```

Out[41]: 0.3965399868948938

浮動小数点計算で生じる微小な誤差を除いてよく一致している.

In [42]:
```
pval1 - pval1_R
```

Out[42]: -3.3306690738754696e-16

注意: Julia言語から[R言語 (https://cran.r-project.org/)](https://cran.r-project.org/)を使うためには, 自分のパソコンにR言語一式をインストールし, さらにJulia言語の側で

- [https://github.com/JuliaInterop/RCall.jl (https://github.com/JuliaInterop/RCall.jl)](https://github.com/JuliaInterop/RCall.jl)

をインストールしておく必要がある.

統計学を勉強するためには, 自分のパソコンで Julia, Python, R の3つを使えるようにしておくと非常に便利である. しかし, そのためにはまた別のスキルが必要になる. 数年計画で身に付けるようにしておくとよいと思われる. トラブルに出会ったら, 表示されたエラーメッセージをインターネットで検索してみるとよい. コンピュータやプログラミングでは英語で書かれたトラブルの解決法を読むことは普通であり, そのために英語にも慣れておくことが必要になる. 結局のところ, **統計学を十分にマスターするために, コンピュータの使い方と技術英語もマスターする必要が生じる.** しかし, 統計学を含めたそれら諸々のスキルを身に付けた人達は我々の社会の中で少数派であり, 学生時代のうちに身に付けることに成功した人達は相当に自信を持ってよいと思う. 繰り返しになるが, 数年計画で無理せずに身に付ければ十分である. あせる必要はない. 就職した後にも勉強を続けることが普通である. **注意終**

WolframAlphaで求めた値の相対誤差が十分に小さいことの確認.

In [43]:
```
0.396541/pval1 - 1
```
Out[43]: 2.5548624098536976e-6

In [44]:
```
# (2)
pval2 = pvalue_welch(x, y; Δμ = -30)
```
Out[44]: 0.027439073239531347

R言語で計算し直すと次のようになる.

In [45]:
```
@rput x y
pval2_R = rcopy(R"""t.test(x, y, mu = -30)""")[:p_value]
```
Out[45]: 0.027439073239531344

In [46]:
```
pval2 - pval2_R
```
Out[46]: 3.469446951953614e-18

WolframAlphaで求めた値の相対誤差が十分に小さいことの確認.

In [47]:
```
0.0274389/pval2 - 1
```
Out[47]: -6.313607235797214e-6

In [48]:
```
# (3)
ci3 = confint_welch(x, y; α = 0.05)
print(ci3)
```
```
[-27.376321534615272, 11.179654867948585]
```

R言語で計算し直すと次のようになる.

In [49]:
```
@rput x y
ci3_R = rcopy(R"""t.test(x, y, conf.level = 0.95)""")[:conf_int]
print(ci3_R)
```
```
[-27.376321534615265, 11.179654867948592]
```

In [50]:
```
ci3 - ci3_R ▷ print
```
```
[-7.105427357601002e-15, -7.105427357601002e-15]
```

WolframAlphaで求めた値の相対誤差が十分に小さいことの確認.

In [51]:
```
[-27.3763, 11.1797] ./ ci3 .- 1 ▷ print
```
```
[-7.866146386481532e-7, 4.036980743027385e-6]
```

Julia言語では要素ごとの演算(broadcast)にするためにはこのように演算記号に . を付ける.

せっかくなのでP値函数もプロットしておこう.

In [52]:

```
@show xlim = confint_welch(x, y; α=1e-3)
plot(Δμ → pvalue_welch(x, y; Δμ), xlim...; label="P-value function", c=:black)
vline!([mean(x) - mean(y)]; label="Δμ = x̄ - ȳ", ls=:dash, c=2)
plot!(ci3, fill(0.05, 2); label="95% conf. int.", lw=2, c=3)
scatter!([0], [0.05]; label="sig. level α = 5%", c=3, msc=:auto)
vline!([0]; label="", lw=0.5, c=:black)
scatter!([0], [pval1]; label="P-value of Δμ = 0", c=:red, msc=:auto)
plot!(xguide="μ_x - μ_y = Δμ", yguide="P-value")
plot!(ytick=0:0.05:1)
plot!(size=(640, 350))
```

```
xlim = confint_welch(x, y; α = 0.001) = [-42.723955601987406, 26.52728893532072]
```

Out[52]:

これと以下を比較せよ.

上のWelchの t 検定でのP値函数のグラフでは, パラメータ θ は2つの群の平均の差を意味する $\Delta\mu$ になっている.

そして, 点推定値は $\Delta\mu = \bar{x} - \bar{y}$ になっている. (この点推定値は正規分布の標本分布モデルにおける最尤推定値にもなっている.)

母集団からの標本の無作為抽出(random samplping, ランダム・サンプリング)を行うごとに変化するデータの数値の確率的揺らぎも考慮すると, 標本を取得した2つの母集団の平均の差 $\mu_x - \mu_y$ (これの値は未知)がぴったり点推定値 $\bar{x} - \bar{y}$ になっているとは考えられないことに注意せよ.

データの数値と $\mu_x - \mu_y = \Delta\mu$ という仮説の整合性がP値函数のグラフを見ればわかる.

大雑把に, $\Delta\mu$ の値が -43 未満だったり, 27 より大きいとき, $\mu_x - \mu_y = \Delta\mu$ という仮説とデータの整合性はほぼないように見える. (この判断は有意水準を $\alpha = 0.1\%$ とした場合に相当している.)

さらに大雑把に閾値 $\alpha = 5\%$ で切断した場合の信頼区間は大体 $[-27, 11]$ でそれよりは狭くなっている.

4 数学的な補足: 大数の法則と中心極限定理について

以下で $f(x)$ は有界な連続函数であると仮定する.

4.1 二項分布の大数の法則

二項分布

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

に関する大数の法則は, n が大きなとき, k/n の分布が p の近くに集中することを意味する. これは

$$\lim_{n\to\infty} \sum_{k=0}^{n} f\left(\frac{k}{n}\right) \binom{n}{k} p^k (1-p)^{n-k} = f(p)$$

が成立することを意味している.

注意: この結果の厳密な直接的証明は

- 高木貞治『解析概論』岩波書店

の「§78. Weierstrassの定理」の節にある. そこの式(8)を見よ. すなわち, 閉区間上の連続函数が多項式函数で一様近似できるというWeierstrassの定理は二項分布の大数の法則(の精密化)から得られる. この方法は Serge Bernstein による. **注意終**

$f(x) = \cos(\pi x)$, $p = 1/3$ の場合について, 二項分布の大数の法則を数値的に確認してみよう.

In [53]:

```
f(x) = cospi(x)
n, p = 10^6, 1/3
@show bin = Binomial(n, p)
@show LHS = sum(f(k/n)*pdf(bin, k) for k in support(bin))
@show RHS = f(p)
@show LHS/RHS - 1;
```

```
bin = Binomial(n, p) = Binomial{Float64}(n=1000000, p=0.3333333333333333)
LHS = sum((f(k / n) * pdf(bin, k) for k = support(bin))) = 0.49999945169316323
RHS = f(p) = 0.5
LHS / RHS - 1 = -1.096613673534641e-6
```

In [54]:
```
f(x) = cospi(x)
ns, p = 10 .^ (1:0.5:4), 1/3
LHSs = [(n = round(Int, n); bin = Binomial(n, p);
        sum(f(k/n)*pdf(bin, k) for k in support(bin))) for n in ns]
RHS = f(p)
plot(ns, LHSs; xscale=:log10, xtick=ns, marker=:o,
    label="LHS", legend=:right)
hline!([RHS]; label="RHS")
```

Out[54]:

In [55]:
```
plot((plot(x -> pdf(Binomial(10^k, 1/3), round(Int, x)), 0, 1.05*10^k;
            label="", title="Binomial($(10^k), 1/3)") for k in 2:5)...;
    size=(800, 500), layout=(2,2))
```

Out[55]:

4.2 二項分布の中心極限定理

二項分布

$$P(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k} \quad (k = 0, 1, \ldots, n)$$

に関する大数の法則は, n が大きなとき, k の分布が二項分布と同じ平均 np と分散 $np(1-p)$ を持つ正規分布で近似されることを意味する. すなわち, n が大きなとき,

$$z = \frac{k - np}{\sqrt{np(1-p)}}$$

の分布が標準正規分布で近似されることを意味する. このことは次が成立することを意味する:

$$\lim_{n\to\infty}\sum_{k=0}^{n} f\left(\frac{k-np}{\sqrt{np(1-p)}}\right)\binom{n}{k}p^k(1-p)^{n-k} = \int_{-\infty}^{\infty} f(z)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz.$$

$f(x)=\cos(x)$, $p=1/3$ の場合について, 二項分布の中心極限定理を数値的に確認してみよう.

In [56]:
```
f(x) = cos(x)
n, p = 10^6, 1/3
@show bin = Binomial(n, p)
@show LHS = sum(f((k-n*p)/√(n*p*(1-p)))*pdf(bin, k) for k in support(bin))
g(z) = f(z) * pdf(Normal(0, 1), z)
@show RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
@show LHS/RHS - 1;
```

```
bin = Binomial(n, p) = Binomial{Float64}(n=1000000, p=0.3333333333333333)
LHS = sum((f((k - n * p) / √(n * p * (1 - p))) * pdf(bin, k) for k = support(bin))) = 0.60653061
7597576
RHS = (quadgk(g, -Inf, Inf))[1] = 0.606530659712633
LHS / RHS - 1 = -6.94359902908559e-8
```

実は次が成立している:

$$\int_{-\infty}^{\infty}\cos(tz)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz = \int_{-\infty}^{\infty} e^{itz}\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz = E\left[e^{itZ}\right] = e^{-t^2/2}.$$

ここで Z は標準正規分布に従う確率変数である.

In [57]:
```
@vars z t real=true
g_sym(z) = cos(t*z) * exp(-z^2/2)/√(2Sym(π))
expr = sympy.Integral(g_sym(z), (z, -oo, oo))
sol_exact = expr.doit()
Eq(expr, sol_exact)
```

Out[57]:

$$\int_{-\infty}^{\infty}\frac{\sqrt{2}e^{-\frac{z^2}{2}}\cos(tz)}{2\sqrt{\pi}}\,dz = e^{-\frac{t^2}{2}}$$

In [58]:
```
@show sol_numeriacal = float(sol_exact(t⇒1))
@show RHS/sol_numeriacal - 1;
```

```
sol_numeriacal = float(sol_exact(t ⇒ 1)) = 0.6065306597126334
RHS / sol_numeriacal - 1 = -7.771561172376096e-16
```

In [59]:
```
f(x) = cos(x)
ns, p = 10 .^ (1:0.5:4), 1/3
LHSs = [(n = round(Int, n); bin = Binomial(n, p);
        sum(f((k-n*p)/√(n*p*(1-p)))*pdf(bin, k) for k in support(bin))) for n in ns]
g(z) = f(z) * pdf(Normal(0, 1), z)
RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
plot(ns, LHSs; xscale=:log10, xtick=ns, marker=:o,
    label="LHS", legend=:right)
hline!([RHS]; label="RHS")
```

Out[59]:

In [60]:

```
plot((plot(x -> pdf(Binomial(n, 1/3), round(Int, x)),
            round(Int, n/3 - 5√(n*1/3*2/3)),
            round(Int, n/3 + 5√(n*1/3*2/3));
            label="", title="Binomial($n, 1/3)") for n in (3, 10, 100, 1000))...;
    size=(800, 500), layout=(2,2))
```

Out[60]:

4.3 他の分布の場合

正規分布で近似される他の分布の場合にも同様の結果が成立している.

4.3.1 例: Poisson分布

$$P(k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!} \quad (k = 0, 1, 2, \ldots)$$

の平均と分散は共に λ であり, λ が大きなとき同じ平均と分散を持つ正規分布で近似されるので,

$$\lim_{\lambda\to\infty} \sum_{k=0}^{\infty} f\left(\frac{k-\lambda}{\sqrt{\lambda}}\right) \frac{e^{-\lambda}\lambda^k}{k!} = \int_{-\infty}^{\infty} f(z) \frac{e^{-z^2/2}}{\sqrt{2\pi}}\, dz.$$

In [61]:

```
f(x) = cos(x)
λ = 10^6
@show poi = Poisson(λ)
m, s = mean(poi), std(poi)
supp = max(0, round(Int, m-6s)):round(Int, m+6s)
@show LHS = sum(f((k-λ)/√λ) * pdf(poi, k) for k in supp)
g(z) = f(z) * pdf(Normal(0, 1), z)
@show RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
@show LHS/RHS - 1;
```

```
poi = Poisson(λ) = Poisson{Float64}(λ=1.0e6)
LHS = sum((f((k - λ) / √λ) * pdf(poi, k) for k = supp)) = 0.6065306741542528
RHS = (quadgk(g, -Inf, Inf))[1] = 0.606530659712633
LHS / RHS - 1 = 2.3810205806995555e-8
```

In [62]:

```
f(x) = cos(x)
λs = 10 .^ (1:0.5:4)
LHSs = [(poi = Poisson(λ);
        (m, s) = (mean(poi), std(poi));
        supp = max(0, round(Int, m-6s)):round(Int, m+6s);
        sum(f((k-λ)/√λ) * pdf(poi, k) for k in supp)) for λ in λs]
g(z) = f(z) * pdf(Normal(0, 1), z)
RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
plot(λs, LHSs; xscale=:log10, xtick=λs, marker=:o,
    label="LHS", legend=:right)
hline!([RHS]; label="RHS")
```

Out[62]:

In [63]:

```
plot((plot(x -> pdf(Poisson(λ), round(Int, x)),
            round(Int, λ - 5√(λ)),
            round(Int, λ + 5√(λ));
            label="", title="Poisson($λ)") for λ in (3, 10, 100, 1000))...;
    size=(800, 500), layout=(2,2))
```

Out[63]:

4.3.2 例: Gamma分布

$$p(x|\alpha,\theta) = \frac{e^{-x/\theta}x^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}$$

は 平均 $\alpha\theta$, 分散 $\alpha\theta^2$ を持ち, α が大きなとき同じ平均と分散を持つ正規分布で近似されるので,

$$\lim_{\lambda\to\infty}\int_0^\infty f\left(\frac{x-\alpha\lambda}{\sqrt{\alpha\theta^2}}\right)\frac{e^{-x/\theta}x^{\alpha-1}}{\theta^\alpha\Gamma(\alpha)}\,dx=\int_{-\infty}^{\infty}f(z)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz.$$

In [64]:

```
f(x) = cos(x)
α, θ = 10^6, 2
@show gam = Gamma(α, θ)
F(x) = f((x - α*θ)/√(α*θ^2)) * pdf(gam, x)
m, s = mean(gam), std(gam)
a, b = m-5s, m+5s
@show LHS = quadgk(F, a, b)[1]
g(z) = f(z) * pdf(Normal(0, 1), z)
@show RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
@show LHS/RHS - 1;
```

```
gam = Gamma(α, θ) = Gamma{Float64}(α=1.0e6, θ=2.0)
LHS = (quadgk(F, a, b))[1] = 0.6065305209355965
RHS = (quadgk(g, -Inf, Inf))[1] = 0.606530659712633
LHS / RHS - 1 = -2.2880465200270095e-7
```

In [65]:

```
f(x) = cos(x)
αs, θ = 10 .^ (1:0.5:4), 2
LHSs = [(gam = Gamma(α, θ);
        F(x) = f((x - α*θ)/√(α*θ^2)) * pdf(gam, x);
        (m, s) = (mean(gam), std(gam));
        (a, b) = (m-5s, m+5s);
        quadgk(F, a, b)[1]) for α in αs]
g(z) = f(z) * pdf(Normal(0, 1), z)
RHS = quadgk(g, -Inf, Inf)[1] # using QuadGK すると使える数値積分函数
plot(αs, LHSs; xscale=:log10, xtick=αs, marker=:o,
    label="LHS", legend=:right)
hline!([RHS]; label="RHS")
```

Out[65]:

In [66]:
```
plot((plot(x -> pdf(Gamma(α, θ), x), α*θ - 5√(α*θ^2), α*θ + 5√(α*θ^2);
            label="", title="Gamma($α, θ)") for α in (3, 10, 100, 1000))...;
    size=(800, 500), layout=(2,2))
```

Out[66]:

4.4 標本分布の場合

平均 μ, 分散 σ^2 を持つ分布 D の標本分布 D^n に従う確率変数 $(X_1, \ldots, X_n)$ について,

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$$

は平均 μ, 分散 σ^2/n を持ち, 大数の法則より, n が大きなとき μ の近くにその分布は集中する. このことは次が成立することを意味している:

$$\lim_{n\to\infty} E[f(\bar{X}_n)] = f(\mu).$$

さらに, 中心極限定理より, $\bar{X}_n$ はn が大きなとき同じ平均と分散を持つ正規分布に近似的に従う. すなわち,

$$Z_n = \frac{\bar{X}_n - \mu}{\sqrt{\sigma^2/n}} = \frac{\sqrt{n}\,(\bar{X}_n - \mu)}{\sigma} = \frac{1}{\sqrt{n}\,\sigma}\sum_{i=1}^{n}(X_i - \mu)$$

は n が大きなとき近似的に標準正規分布に従う. このことは次が成立することを意味している:

$$\lim_{n\to\infty} E[f(Z_n)] = \int_{-\infty}^{\infty} f(z)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz.$$

例えば, 平均 μ, 分散 σ^2 の分布 D が確率密度函数 $p(x)$ を持つとき,

$$\lim_{n\to\infty}\int\cdots\int f\left(\frac{x_1+\cdots+x_n}{n}\right)p(x_1)\cdots p(x_n)\,dx_1\cdots dx_n = f(\mu),$$
$$\lim_{n\to\infty}\int\cdots\int f\left(\frac{(x_1-\mu)+\cdots+(x_n-\mu)}{\sqrt{n}\,\sigma}\right)p(x_1)\cdots p(x_n)\,dx_1\cdots dx_n = \int_{-\infty}^{\infty} f(z)\frac{e^{-z^2/2}}{\sqrt{2\pi}}\,dz.$$

ここで $\int$ は適切な範囲の定積分を意味し, 各 n ごとに n 重の積分を考えている.

このような n 重積分の $n\to\infty$ での極限公式は非常に複雑な形に見えるが, 大数の法則と中心極限定理を理解していれば直観的に当然そうなるべき結果だと理解できる.

4.5 再掲: 大数の法則と中心極限定理のイメージ

以下の図は

- 「大数の法則と中心極限定理」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)

にあった図の再掲である.

確率統計に関する入門的教科書の中には, 大数の法則や中心極限定理について説明するために, 確率変数や確率分布に関する様々な種類の収束性の定義を説明しているものが多い. (確率収束, 概収束, 法収束(法則収束, 分布収束)など. 他にも色々ある.)

この一連のノート群においては, そのような説明を意図的に避けた.

なぜならば, 統計学を実践的に使うことが目標の人達は以下の図のイメージを大事した方がよいように思われるからである.

数学的に正確な理解を目指す場合であっても, 最初のうちは以下の図のようなイメージを大事にして, むしろこのイメージを元に, 確率変数や確率分布の収束に関する数学的に適切な定義について, 自分で考えるようにした方が良いように思われる.

教科書に書いてある天下り的な定義から出発しようとすると数学の理解は苦しくなることが多い.

自分の力で適切な定義を作るために十分な「論理的スキル」と「健全な直観」の組み合わせの習得を目指す方が楽に理解できる可能性が増えるだろう.

$\boxed{\text{大数の法則と中心極限定理のイメージ}}$

$\mu = E[X_i]$, $\sigma = \sqrt{E[(X_i-\mu)^2]}$ とおく.

$X_1, X_2, X_3, \ldots$ は独立同分布確率変数列であるとし, $\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ とおく.

平均 μ, 標準偏差 $\frac{\sigma}{\sqrt{n}} \to 0$

サイズ n の標本の平均 $\overline{X}_n$ の分布の $n \to \infty$ での様子:

$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ の分布は $n \to \infty$ で $\mu = E[X_i]$ に集中する.

$\overline{X}_n$ と μ の差を $\sqrt{n}$ 倍して拡大して見るとこうなる: $Y_n = \sqrt{n}(\overline{X}_n - \mu) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(X_i - \mu)$.

平均 0, 標準偏差 σ

$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ と μ の差の大きさはだいたい $\frac{\sigma}{\sqrt{n}}$ の大きさであり,

差を $\sqrt{n}$ 倍拡大して分布を見ると標準偏差 σ の正規分布になっている.

さらに σ 分の1倍するとこうなる: $Z_n = \frac{\sqrt{n}}{\sigma}(\overline{X}_n - \mu) = \frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}\frac{X_i - \mu}{\sigma}$

平均 0, 標準偏差 1

以上のように, 大数の法則によって1点に集中する標本平均の分布を

$\sqrt{n}$ 倍 (もしくは $\sigma/\sqrt{n}$ 分の1倍) することによってより精密に見ると中心極限定理が得られる.

4.6 統計学の基礎になる確率論の三種の神器

これは筆者の個人的な意見に過ぎないのだが, 統計学の基礎になる確率論の三種の神器は次の3つである:

- 大数の法則
- 中心極限定理
- Kullback-Leibler情報量に関するSanovの定理

この一連のノート群で3つ目の **Kullback-Leibler情報量に関するSanovの定理** にはほとんど触れることができなかった.

Kullback-Leibler情報量のSanovの定理については

- 「Bernoulli試行と関連確率分布」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/01%20Bernoulli%20trial%20and%20related%20distributions.ip

の「問題: Kullback-Leibler情報量とGibbsの情報不等式」と

- 「大数の法則と中心極限定理」に関するノート (https://nbviewer.org/github/genkuroki/Statistics/blob/master/2022/05%20Central%20limit%20theorem.ipynb)

の「注意: Kullback-Leibler情報量とSanovの定理との関係」で簡単に触れただけで終わった.

Kullback-Leibler情報量のSanovの定理は **大偏差原理** (large deviation principle)の特別な場合になっているので, 三種の神器の3番目は「大偏差原理」に一般化しておいてもよい. 大偏差原理は統計力学的な意味でのエントロピーや除法理論における情報量の概念を扱う話だと思ってよい.

Kullback-Leibler情報量のSanovの定理は赤池情報量規準(AIC)などの情報量規準の基礎になる.

Sanovの定理の易しい解説が以下の場所にある:

- Kullback-Leibler情報量とSanovの定理 (https://genkuroki.github.io/documents/20160616KullbackLeibler.pdf)

確率論で役に立つ解析学については次のノートも参考になるかもしれない:

- ガンマ分布の中心極限定理とStirlingの公式 (https://genkuroki.github.io/documents/20160501StirlingFormula.pdf)

ついでに紹介しておくと, ガンマ函数やベータ函数などについては次の場所にあるノート群が詳しい:

- 「微分積分学」に関するノート群 (https://github.com/genkuroki/Calculus)

さらに, 図書館で次の文献も参照できれば, 統計学の基礎になる確率論の三種の神器について理解を深め易いと思われる:

- 高橋陽一郎, 確率論の広がり, 数学のたのしみ, no.8 (1998) pp.26-35 (関連情報を検索 (https://www.google.com/search?q=%E9%AB%98%E6%A9%8B%E9%99%BD%E4%B8%80%E9%83%8E+%E7%A2%BA%E7%8E%87%E8%AB%96%E3%81

面白いことはたくさんある(ありすぎる).

あせらずに楽しみながら勉強して欲しいと思います.

4.7 Kullback-Leibler情報量に関するSanovの定理の数値例

二項分布の場合のSanovの定理の1つの形: $n \to \infty$, $k \sim np$ のとき

$$\lim_{n\to\infty} \frac{-1}{n} \log\left(\binom{n}{k} q^k (1-q)^{n-k}\right) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}.$$

この公式の右辺はKullback-Leibler情報量の最も簡単な場合になっている. これは

$$\begin{aligned}&(\text{二項分布 } \mathrm{Binomial}(n,q) \text{ で } np \text{ に近い値 } k \text{ が生成される確率})\\ &= \exp\left(-n\left(p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}\right) + o(n)\right).\end{aligned}$$

と書き直される. これは, 二項分布 $\mathrm{Binomial}(n,q)$ で np に近い値 k が生成される確率の大きさがほぼKullback-Leibler情報量で決まっていることを意味している.

注意: 上の結果は, 階乗のStirlingの近似公式を使うと簡単に証明できるし, 高校数学で習ったはずに区分求積法を使っても容易に証明できる. **注意終**

上の結果を数値的に確認してみよう.

In [67]:
```
n, p, q = 10^16, 1/3, 1/2
@show LHS = (-1/n)*logpdf(Binomial(n, q), round(Int, n*p))
@show RHS = p*log(p/q) + (1-p)*log((1-p)/(1-q));
```

```
LHS = (-1 / n) * logpdf(Binomial(n, q), round(Int, n * p)) = 0.05663301226513299
RHS = p * log(p / q) + (1 - p) * log((1 - p) / (1 - q)) = 0.056633012265132565
```

In [68]:

```
ns, p, q = 10 .^ (1:0.5:4), 1/3, 1/2
LHSs = [(-1/n)*logpdf(Binomial(round(Int, n), q), round(Int, n*p)) for n in ns]
RHS = p*log(p/q) + (1-p)*log((1-p)/(1-q))
plot(ns, LHSs; xscale=:log10, xtick=ns, marker=:o, label="LHS")
hline!([RHS]; label="RHS")
```

Out[68]:

In [ ]:

```
```