

高校数学の話題2

- 黒木玄 (Gen Kuroki)
- 2022-08-30
- Copyright 2022 Gen Kuroki
- License: MIT <https://opensource.org/licenses/MIT> (<https://opensource.org/licenses/MIT>)
- Repository: <https://github.com/genkuroki/HighSchoolMath> (<https://github.com/genkuroki/HighSchoolMath>)

このノートでは高校の数学の教科書にあるような話題を扱い, その数学的背景について解説する.

タイポや自明な誤りは自分で訂正して読むこと. 本質的な誤りがあれば著者に教えて欲しい.

このノートの想定読者は大学である程度を数学を学んで人で高校で習った数学について見直したい人達である.

このノートは未完成であり, 後で新たな話題を追加する可能性がある.

目次

- ▼ [1 ポリアの壺](#)
 - [1.1 問題: \$n\$ 回目に赤玉が出る確率](#)
 - [1.2 解答例: \$n\$ 回目に赤玉が出る確率の漸化式による導出](#)
 - [1.3 解答例: \$n\$ 回目に赤玉が出る確率の漸化式を使わない導出](#)
 - [1.4 ポリアの壺の大学入試問題の例: 名古屋大学2007年文系前期3\(b\)](#)
 - [1.5 問題: 名古屋大学2007年文系前期3\(b\)の一般化](#)
 - [1.6 解答例: \$P\(k|n, a, b\)\$ の漸化式による導出](#)
 - [1.7 解答例: \$P\(k|n, a, b\)\$ の漸化式を使わない導出](#)
- ▼ [2 ポリアの壺と二項分布や超幾何分布の関係](#)
 - [2.1 ベータ二項分布, 二項分布, 超幾何分布](#)
 - [2.2 ベータ二項分布, 二項分布, 超幾何分布の統一的理解 \(ポリア分布\)](#)
 - [2.3 問題: 二項分布がポリア分布の極限になっていること](#)
 - ▼ [2.4 解答例: 二項分布がポリア分布の極限になっていること](#)
 - [2.4.1 二項分布, ベータ二項分布, 超幾何分布の同時プロット\(素朴な方法\)](#)
 - [2.4.2 二項分布, ベータ二項分布, 超幾何分布の同時プロット\(1つの関数にまとめる方法\)](#)
 - [2.5 問題: ポリア分布の期待値と分散](#)
 - [2.6 解答例: ポリア分布の期待値と分散](#)
- ▼ [3 ポリアの壺試行の別の解釈](#)
 - [3.1 ベータ分布, ベータ関数, ガンマ関数](#)
 - [3.2 成功確率がベータ分布に従ってランダムに決まっているようなベルヌーイ試行](#)
 - [3.3 余談: 生まれつきの才能か, 努力の結果か?](#)
 - [3.4 問題: ポリアの壺試行で大数の法則が成立することの確認](#)
 - ▼ [3.5 解答例: ポリアの壺試行の側で大数の法則が成立することの確認](#)
 - [3.5.1 ポリアの壺試行の側で大数の法則が成立することの確認](#)
 - [3.5.2 ポリアの壺試行での大数の法則の収束先の分布がベータ分布になることの確認](#)

```
In [1]: 1 using Distributions
        2 using StatsPlots
        3 default(fmt=:png, size=(400, 250), titlefontsize=10,
        4           tickfontsize=6, guidefontsize=9, plot_titlefontsize=12)
        5 using Random
        6 Random.seed!(4649373)
```

Out[1]: TaskLocalRNG()

```
In [2]: 1 # SymPy.jl はjuliaを起動した後に以下のようにすると良いだろう。
2 #
3 # julia> ENV["Python"] = ""
4 # julia> ]
5 # pkg> add SymPy
6 # pkg> バックスペースを押す。
7 # julia> using SymPy
8 # Python環境がJulia環境の中にインストールされる。
9
10 using SymPy
11
12 # Override the Base.show definition of SymPy.jl:
13 # https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/types.
14
15 @eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
16     print(io, as_markdown("\displaystyle " * sympy.latex(x, mode="plain", fold_short_frac=false)))
17 end
18 @eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
19     function toeqnarray(x::Vector{Sym})
20         a = join(["\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\")
21         """\left[ \begin{array}{r} $a \end{array} \right]"""
22     end
23     function toeqnarray(x::AbstractArray{Sym,2})
24         sz = size(x)
25         a = join([join(["\displaystyle " * map(sympy.latex, x[i,:]), "&") for i in 1:sz[1]],
26             "\left[ \begin{array}{"} * repeat("r",sz[2]) * "}" * a * "\end{array}\right]")
27     end
28     print(io, as_markdown(toeqnarray(x)))
29 end
```

1 ポリアの壺

a, b は正の整数であると仮定する.

注意: 実際には a, b が 0 以上の実数であり, 少なくともどちらか片方が正であれば十分であるが, 壺から玉を取り出す設定で説明するためにこのように仮定しておく.

初期状態では壺の中に赤玉が a 個, 白玉が b 個入っていると仮定する.

その初期状態から出発して, 以下の試行を繰り返す:

(1) 壺から玉を無作為に取り出して, 取り出された玉の色を記録する.

(2) 取り出した玉と同じ色の玉を壺に **2個** 戻す.

この試行を1回行うごとに壺の中の玉の個数は1個ずつ増える.

これを以下では **ポリアの壺試行** (Pólya' urn trial)もしくは **ポリアの壺過程** (Pólya' urn process)と呼ぶことにする.

1.1 問題 : n 回目に赤玉が出る確率

ポリアの壺のついでには次の問題が定番である.

ポリアの壺試行において, n 回目に取り出した玉の色が赤である確率を求めよ.

この問題について, 答えが分かるまでまたは1時間以上考えてから以下の節を読むと理解が深まるだろう.

1.2 解答例 : n 回目に赤玉が出る確率の漸化式による導出

ポイント: 漸化式を作るときには, 「 n 回試行してから 1 回試行する」と考えるのではなく, 「1 回試行してから, n 回試行する」と考えるとよい. 1 回目の試行に関する確率計算は自明であるが, n 回の試行後の壺の状態に関する確率の計算は非自明である.

初期状態で壺の中に赤玉が a 個, 白玉が b 個入っている場合のポリアの壺試行を考える.

n 回目に取り出した玉の色が赤である確率を $p_n(a, b)$ と書こう.

1 回目に取り出した玉の色が赤である確率は $a/(a + b)$ である:

$$p_1(a, b) = \frac{a}{a + b}.$$

$n + 1$ 回目に取り出した玉の色が赤である確率は

- 1回目に赤玉が取り出される確率と
壺の初期状態が赤玉 $a + 1$ 個, 白玉 b 個のポリアの壺試行の n 回目で赤玉が取り出される確率の積
- 1回目に白玉が取り出される確率と
壺の初期状態が赤玉 a 個, 白玉 $b + 1$ 個のポリアの壺試行の n 回目で赤玉が取り出される確率の積

の和に等しいので,

$$p_{n+1}(a, b) = \frac{a}{a+b} p_n(a+1, b) + \frac{b}{a+b} p_n(a, b+1).$$

ゆえに,

$$\begin{aligned} p_2(a, b) &= \frac{a}{a+b} p_1(a+1, b) + \frac{b}{a+b} p_1(a, b+1) \\ &= \frac{a}{a+b} \frac{a+1}{a+b+1} + \frac{b}{a+b} \frac{a}{a+b+1} = \frac{a}{a+b} \\ p_3(a, b) &= \frac{a}{a+b} p_2(a+1, b) + \frac{b}{a+b} p_2(a, b+1) \\ &= \frac{a}{a+b} \frac{a+1}{a+b+1} + \frac{b}{a+b} \frac{a}{a+b+1} = \frac{a}{a+b} \\ &\dots \dots \dots \end{aligned}$$

このように次々に $p_n(a, b) = a/(a+b)$ であることが帰納的に示される.

実際, $p_n(a, b) = a/(a+b)$ と仮定すると,

$$\begin{aligned} p_{n+1}(a, b) &= \frac{a}{a+b} p_n(a+1, b) + \frac{b}{a+b} p_n(a, b+1) \\ &= \frac{a}{a+b} \frac{a+1}{a+b+1} + \frac{b}{a+b} \frac{a}{a+b+1} = \frac{a}{a+b}. \end{aligned}$$

答え: $p_n(a, b) = \frac{a}{a+b}$.

1.3 解答例：n回目に赤玉が出る確率の漸化式を使わない導出

ポリアの壺試行の代わりに以下のような試行を考える.

c は正の整数であると仮定する.

初期状態では 1 から c までの番号が付けられた c 種類の玉が1個ずつ壺の中に入っていると仮定する.

初期状態では壺の中に c 個の玉が入っていることになる.

その初期状態から出発して, 以下の試行を繰り返す:

- (1) 壺から玉を無作為に取り出して, 取り出された玉の番号を記録する.
- (2) 取り出した玉と同じ番号が付けられた玉を壺に **2個** 戻す.

この試行を1回行うごとに壺の中の玉の個数は1個ずつ増える.

この試行は玉に付けられた c 種類の番号について対称である.

ゆえに, i が 1 から c までの番号のどれかだとすると, n 回目に番号 i の玉が取り出される確率は $1/c$ になる.

ポリアの壺試行は, 1 番から a 番の玉が赤色で, $a + 1$ 番から $a + b = c$ 番の玉が白色の場合における以上の試行と同等だと考えられる.

ゆえに, n 回目の試行で 1 から a の番号が付けられた赤玉が取り出される確率は $a/c = a/(a+b)$ になる.

1.4 ポリアの壺の大学入試問題の例：名古屋大学2007年文系前期3(b)

袋の中に赤と白の玉が1個ずつ入っている。「この袋から1個取り出して戻し, 出た玉と同じ色の玉を袋の中に1個追加する」という操作を N 回繰り返した後, 赤の玉が袋の中に m 個ある確率を $p_N(m)$ とする.

- (1) $p_3(m)$ を求めよ.
- (2) 一般の N に対し, $p_N(m)$ を求めよ.

[illegible]

答えは次の通り:

$$p_N(m) = \begin{cases} 1/(N+1) & (m = 1, 2, \dots, N+1) \\ 0 & (\text{otherwise}) \end{cases}$$

答えの具体的な対応は $P(k|N, 1, 1) = p_N(1+k)$ で与えられる.

$$P(1|1, a, b) = \frac{a}{a+b}, \quad P(0|1, a, b) = \frac{b}{a+b}.$$
$$P(k|n+1, a, b) = \frac{a}{a+b} P(k-1|n, a+1, b) + \frac{b}{a+b} P(k|n, a, b+1).$$

この漸化式を使ってがんばって計算すると以下のようになることがわかる:

$$P(2|2, a, b) = \frac{a(a+1)}{(a+b)(a+b+1)},$$

$$P(1|2, a, b) = 2 \frac{ab}{(a+b)(a+b+1)},$$

$$P(0|2, a, b) = \frac{b(b+1)}{(a+b)(a+b+1)},$$

$$P(3|3, a, b) = \frac{a(a+1)(a+2)}{(a+b)(a+b+1)(a+b+2)},$$

$$P(2|3, a, b) = 3 \frac{a(a+1)b}{(a+b)(a+b+1)(a+b+2)},$$

$$P(1|3, a, b) = 3 \frac{ab(b+1)}{(a+b)(a+b+1)(a+b+2)(a+b+2)},$$

$$P(0|3, a, b) = \frac{b(b+1)(b+2)}{(a+b)(a+b+1)(a+b+2)},$$

$$P(4|4, a, b) = \frac{a(a+1)(a+2)(a+3)}{(a+b)(a+b+1)(a+b+2)(a+b+3)},$$

$$P(3|4, a, b) = 4 \frac{a(a+1)(a+2)b}{(a+b)(a+b+1)(a+b+2)(a+b+3)},$$

$$P(2|4, a, b) = 6 \frac{a(a+1)b(b+1)}{(a+b)(a+b+1)(a+b+2)(a+b+3)},$$

$$P(1|4, a, b) = 4 \frac{ab(b+1)(b+2)}{(a+b)(a+b+1)(a+b+2)(a+b+2)(a+b+3)},$$

$$P(0|4, a, b) = \frac{b(b+1)(b+2)(b+3)}{(a+b)(a+b+1)(a+b+2)(a+b+3)}.$$

実際に自分でも以上の結果を確認すると、以下の証明の方針が自明に見えて来るはずである。

これから、次が成立することが予想される:

$$P(k|n, a, b) = \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)}.$$

ここで $\binom{n}{k}$ は二項係数である:

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

二項係数は次の漸化式(Pascalの三角形)を満たしている:

$$\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k}.$$

上の予想を帰納法で証明しよう。これが n について成立している仮定する。上で示した漸化式を使うと、

$$\begin{aligned} P(k|n+1, a, b) &= \frac{a}{a+b} P(k-1|n, a+1, b) + \frac{b}{a+b} P(k|n, a, b+1) \\ &= \frac{a}{a+b} \binom{n}{k-1} \frac{(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k)}{(a+b+1) \cdots (a+b+n)} \\ &\quad + \frac{b}{a+b} \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) \cdot (b+1) \cdots (b+n-k)}{(a+b+1) \cdots (a+b+n)} \\ &= \left(\binom{n}{k-1} + \binom{n}{k} \right) \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k)}{(a+b)(a+b+1) \cdots (a+b+n)} \\ &= \binom{n+1}{k} \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k)}{(a+b)(a+b+1) \cdots (a+b+n)}. \end{aligned}$$

ゆえに n を $n+1$ に置き換えた場合の上の予想も成立することがわかる。

数学的帰納法より、すべての正の整数 $n = 1, 2, 3, \dots$ について上の予想が成立することが示された。

結論: $P(k|n, a, b)$ は次のようになる:

$$P(k|n, a, b) = \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)}.$$

注意: 例えば, $n = N$, $a = b = 1$, $k = 0, 1, \dots, N$ のとき,

$$P(k|N, 1, 1) = \frac{N!}{k!(N-k)!} \frac{1 \cdot 2 \cdots k \cdot 1 \cdot 2 \cdots (N-k)}{2 \cdot 3 \cdots (N+1)} = \frac{1}{N+1}.$$

これはちょうど名古屋大学2007年文系前期3(b)の解答を与える.

1.7 解答例: $P(k|n, a, b)$ の漸化式を使わない導出

ポリアの壺試行において, i 回目に赤玉が出たら $X_i = 1$ とおき, 白玉が出たら $X_i = 0$ とおく.

1 と 0 の列 x_1, x_2, \dots, x_n について, $X_1 = x_1, X_2 = x_2, \dots, X_n = x_n$ となる確率を次のように書くことにする:

$$P(x_1, x_2, \dots, x_n | a, b).$$

例えば, $n = 7$, $x_1 x_2 \cdots x_7 = 0110010$ のとき, $P(x_1, \dots, x_n | a, b)$ は試行回数 $n = 7$ のポリアの壺試行によって, 玉が「白赤赤白白赤白」の順で出る確率になる.

この $P(x_1, x_2, \dots, x_n | a, b)$ はポリアの壺試行における確率を最も詳細に記述している.

特に n 回中赤玉が k 回出る確率 $P(k|n, a, b)$ は X_1, \dots, X_n の中に 1 が k 個含まれる確率になるので,

$$P(k|n, a, b) = \sum_{x_1 + \cdots + x_n = k} P(x_1, \dots, x_n | a, b).$$

つまり, $P(x_1, x_2, \dots, x_n | a, b)$ を求める問題は, 以上で扱っている問題のさらなる一般化になっている.

$P(x_1, x_2, \dots, x_n | a, b)$ を求めよう.

例えば, $n = 7$, $x_1 x_2 \cdots x_7 = 0110010$ (赤玉が 3 回, 白玉が 4 回) のとき,

$$\begin{aligned} P(x_1, x_2, \dots, x_7 | a, b) &= \frac{b}{a+b} \frac{a}{a+b+1} \frac{a+1}{a+b+2} \frac{b+1}{a+b+3} \frac{b+2}{a+b+4} \frac{a+2}{a+b+5} \frac{b+3}{a+b+6} \\ &= \frac{a(a+1)(a+2) \cdot b(b+1)(b+2)(b+3)}{(a+b)(a+b+1) \cdots (a+b+6)}. \end{aligned}$$

最初の i 回の試行で赤玉が出た回数を $r_i = x_1 + \cdots + x_i$ と書き, 白玉が出た回数を $w_i = i - r_i$ と書く.

壺の中に赤玉が $a + r_{i-1}$ 個, 白玉が $b + w_{i-1}$ 個入っているときに無作為に取り出した玉が赤玉になる確率は $(a + r_{i-1}) / (a + b + i - 1)$ になり, 白玉になる確率は $(b + w_{i-1}) / (a + b + i - 1)$ になる.

それらの確率の積が $P(x_1, x_2, \dots, x_n | a, b)$ になるので,

$$P(x_1, x_2, \dots, x_n | a, b) = \frac{a^{x_1} b^{1-x_1}}{a+b} \frac{(a+r_1)^{x_2} (b+w_1)^{1-x_2}}{a+b+1} \cdots \frac{(a+r_{n-1})^{x_n} (b+w_{n-1})^{1-x_n}}{a+b+n-1}$$

$A^{x_i} B^{1-x_i}$ が $x_i = 1$ のときに A , $x_i = 0$ のときに B になることを使った.

例えば, $n = 7$, $x_1 x_2 \cdots x_7 = 0110010$ (赤玉が 3 回, 白玉が 4 回) のとき,

$$\begin{aligned} &a^{x_1} (a+r_1)^{x_2} \cdots (a+r_{n-1})^{x_n} \\ &= a^0 a^1 (a+1)^1 (a+2)^0 (a+2)^1 (a+3)^0 \\ &= a(a+1)(a+2), \\ &b^{1-x_1} (b+w_1)^{1-x_2} \cdots (b+w_{n-1})^{1-x_n} \\ &= b^1 (b+1)^0 (b+1)^1 (b+2)^1 (b+3)^0 (b+3)^1 \\ &= b(b+1)(b+2)(b+3). \end{aligned}$$

$a^{x_1} (a+r_1)^{x_2} \cdots (a+r_{n-1})^{x_n}$ の中の $x_i = 0$ に対応する因子は 1 になり, $x_i = 1$ が現れるたびに r_i が 1 増え, $b^{1-x_1} (b+w_1)^{1-x_2} \cdots (b+w_{n-1})^{1-x_n}$ についても同様のことが言える.

これより, $k = r_n$ とおくと (n 回中赤玉が k 回出た場合を考えると), 次のようになることがわかる:

$$\begin{aligned} a^{x_1} (a+r_1)^{x_2} \cdots (a+r_{n-1})^{x_n} &= a(a+1) \cdots (a+k-1), \\ b^{1-x_1} (b+w_1)^{1-x_2} \cdots (b+w_{n-1})^{1-x_n} &= b(b+1) \cdots (b+n-k-1). \end{aligned}$$

したがって, $x_1, \dots, x_n \in \{1, 0\}$, $x_1 + \cdots + x_n = k$ のとき,

$$P(x_1, x_2, \dots, x_n | a, b) = \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)}.$$

特に $P(x_1, x_2, \dots, x_n | a, b)$ は n 回中に赤玉が出た回数 $x_1 + \cdots + x_n = k$ だけで決まり, 赤玉が出る順序にはよらない.

$P(k|n, a, b)$ は $r_n = k$ の場合の $P(x_1, x_2, \dots, x_n|a, b)$ の x_1, \dots, x_n の中から 1 になる k 個を選ぶ組み合わせの数倍になっている。

ゆえに,

$$P(k|n, a, b) = \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)}.$$

ポイント: n 回中 k 回赤玉が出る確率を求める問題を単に解くだけでなく、**より野心的に**,

- ポリアの壺試行における確率を完全に記述している $P(x_1, x_2, \dots, x_n|a, b)$ を求めてしまおう

と考え、さらに、**より謙虚に**,

- $x_1 x_2 \cdots x_7 = 0110010$ の特別な場合について考える

と、 $P(x_1, x_2, \dots, x_n|a, b)$ が x_1, x_2, \dots, x_n を並べる順序によらないこともすぐにわかる。

そこまでたどりつけば以上の議論の全体を作り上げることは易しい。

数学の問題を楽に解く極意は、より一般的な場合を考えることと、特殊な場合を計算してみることの2つにある。

2 ポリアの壺と二項分布や超幾何分布の関係

以下の話は大学レベルの数学を含む。

高校生にそのような話を教える必要はないが、教える側は教養として知っておくべきことについて説明する。

一般に、高校生に数学を教える人は、高校数学の範囲内について詳しいだけでは教養が足りず、高校数学の範囲を超えた数学的教養を持っている必要がある。

高校数学の話題を「高校数学」(鍵括弧付き)という独特で特殊な「既存分野」の話題として消費してしまうのはもったいなく、数学教育的にも好ましいことではないと思われる。

「高校数学」という独特の「既存分野」(特に「受験数学」)のネタとしてポリアの壺を消費してしまうと、ポリアの壺が統計モデリングにおいて役に立つ考え方になっているというように一切気付かずに終わってしまう可能性がある。

2.1 ベータ二項分布, 二項分布, 超幾何分布

初期状態では壺の中に赤玉が a 個, 白玉が b 個入っていると仮定する。

(1) 前節の結果によって,

- 壺から取り出した玉と同じ色の玉を2個壺に戻す。(毎回壺の中の玉の個数は1個ずつ増える.)

というルールの **ポリアの壺** における n 回の試行で赤玉が k 回出る確率は

$$P_{\text{BetaBinom}}(k|n, a, b) = \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)}.$$

になることがわかっている。

これによって定まる k の確率分布は **ベータ二項分布** (Beta-binomial distribution) と呼ばれている。

なぜならば、この確率の式は次のように表されるからである:

$$\begin{aligned} \int_0^1 \binom{n}{k} p^k (1-p)^{n-k} \frac{p^{a-1} (1-p)^{b-1}}{B(a, b)} dp &= \binom{n}{k} \frac{1}{B(a, b)} \int_0^1 p^{a+k-1} (1-p)^{b+n-k-1} dp \\ &= \binom{n}{k} \frac{1}{B(a, b)} B(a+k, b+n-k) \\ &= \binom{n}{k} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \frac{\Gamma(a+k)\Gamma(b+n-k)}{\Gamma(a+b+n)} \\ &= \binom{n}{k} \frac{a(a+1) \cdots (a+k-1) \cdot b(b+1) \cdots (b+n-k-1)}{(a+b)(a+b+1) \cdots (a+b+n-1)} \\ &= P_{\text{BetaBinom}}(k|n, a, b). \end{aligned}$$

下の方にある「ポリアの壺試行の別の解釈」の節により詳しい説明がある。

ベータ二項分布についてはWikipediaでの説明を参照せよ:

- https://en.wikipedia.org/wiki/Beta-binomial_distribution (https://en.wikipedia.org/wiki/Beta-binomial_distribution)

ベータ二項分布という考え方をすると、 a, b が整数でなくても、 a, b が正の実数ならば、上の式が 0 以上の整数に関する離散確率分布を定めることがわかる。

注意: $k > n$ のとき、 $\binom{n}{k} = \frac{n(n-1)\cdots(n-k+1)}{k!}$ の右辺の分子が 0 になるので、 $\binom{n}{k} = 0$ となることに注意せよ。

(2) 試行のルールを

- 壺から取り出した玉をそのまま壺に戻す。(毎回壺の中の玉の個数は変わらない。)

にすると、 n 回の試行で赤玉が k 回出る確率は次の **二項分布** (binomial distribution) の確率になる:

$$P_{\text{Binomial}}(k|n, p) = \binom{n}{k} p^k (1-p)^{n-k}, \quad p = \frac{a}{a+b}.$$

これは次のようにも書ける:

$$P_{\text{Binomial}}\left(k \mid n, \frac{a}{a+b}\right) = \binom{n}{k} \frac{a^k b^{n-k}}{(a+b)^n}.$$

(3) 試行のルールを

- 壺から取り出した玉を壺に戻さない。(毎回壺の中の玉の個数は1個ずつ減る。)

にすると、 n 回の試行で赤玉が k 回出る確率は次の **超幾何分布** (hypergeometric distribution) の確率になる:

$$\begin{aligned} P_{\text{Hypergeom}}(k|n, a, b) &= \binom{n}{k} \frac{a(a-1)\cdots(a-(k-1)) \cdot b(b-1)\cdots(b-(n-k-1))}{(a+b)(a+b-1)\cdots(a+b-(n-1))} \\ &= \binom{a}{k} \binom{b}{n-k} / \binom{a+b}{n}. \end{aligned}$$

$n > a+b$ となることは不可能なことに注意せよ。

この確率の公式の1段目の右辺はポリアの壺の場合と同様の方法で得られる。

2段目の右辺は以下のようにして得られる。

壺に取り出した玉を戻さないという設定で n 回中に赤玉が k 回出る確率は次に等しい:

- 壺から n 個の玉を無作為にまとめて取り出したとき、その中に赤玉が k 個含まれる確率。

$a+b$ 個の玉が入っている壺から n 個取り出す組み合わせ全体の個数が確率の分母になる。

その n 個の中に赤玉が k 個含まれている場合の組み合わせ全体の個数は a 個の赤玉から k 個選ぶ組み合わせの個数と b 個の白玉から $n-k$ 個選ぶ組み合わせの個数の積に等しい。これが確率の分子になる。

このことから上の公式の2段目の右辺の式で求めたい確率が得られることがわかる。

さらに、1段目の右辺と2段目の右辺が等しいことは以下のようにして確認される:

$$\begin{aligned} (1\text{段目}) &= \frac{n!}{k!(n-k)!} \frac{a!}{(a-k)!} \frac{b!}{(b-(n-k))!} \frac{(a+b-n)!}{(a+b)!} \\ &= \frac{a!}{k!(a-k)!} \frac{b!}{(n-k)!(b-(n-k))!} \frac{n!(a+b-n)!}{(a+b)!} = (2\text{段目}). \end{aligned}$$

2.2 ベータ二項分布, 二項分布, 超幾何分布の統一的理解 (ポリア分布)

n は正の整数であるとし、0 以上の整数 k について、 $P(k|n, a, b, d)$ を次のように定める:

$$P(k|n, a, b, d) = \binom{n}{k} \frac{a(a+d)\cdots(a+(k-1)d) \cdot b(b+d)\cdots(b+(n-k-1)d)}{(a+b)(a+b+d)\cdots(a+b+(n-1)d)}.$$

この式は n, a, b が正の整数で、 d が -1 以上の整数のとき、0 以上の整数 k に関する離散確率分布を定める。

さらに、この式は a, b, d が整数ではなく、正の実数であっても、0 以上の整数 k に関する確率分布を定める。

この離散分布は **ポリア分布** (Pólya distribution) と呼ばれている。

- https://encyclopediaofmath.org/wiki/P%C3%B3lya_distribution (https://encyclopediaofmath.org/wiki/P%C3%B3lya_distribution)

このとき,

$$\begin{aligned}P_{\text{BetaBinom}}(k|n, a, b) &= P(k|n, a, b, 1), \\P_{\text{Binomial}}(k|n, a/(a+b)) &= P(k|n, a, b, 0), \\P_{\text{Hypergeom}}(k|n, a/(a+b)) &= P(k|n, a, b, -1).\end{aligned}$$

すなわち, $d = 1, 0, -1$ の場合のポリア分布がそれぞれベータ二項分布, 二項分布, 超幾何分布になっている.

コメント: 二項分布は高校でも習う.

袋の中に赤玉が a 個, 白玉が b 個入っているときに, 一度に n 個を無作為に取り出したときに, その中に k 個赤玉が含まれる確率の問題も高校まで扱う.

標本の抽出の仕方では, 復元抽出の場合の分布が二項分布になっている.

標本の抽出の仕方では, 非復元抽出の場合の分布が超幾何分布になっている.

この意味で二項分布と超幾何分布は母集団からの標本抽出について考えるときにも基本的である.

さらに, ポリアの壺に関係した問題は大学入試の定番の問題になっている.

これらについて教える側は, これらの確率分布が互いに無関係ではないことを認識しておくことが必要だと思われる.

2.3 問題 : 二項分布がポリア分布の極限になっていること

$0 < p < 1$ について, 次が成立することを示せ:

$$\lim_{L \rightarrow \infty} P(k|n, Lp, L(1-p), d) = P_{\text{Binomial}}(k|n, p).$$

もしも可能ならば, さらに, a, b が大きなとき, ベータ二項分布 $P_{\text{BetaBinom}}(k|n, a, b)$ と超幾何分布 $P_{\text{Hypergeom}}(k|n, a, b)$ が二項分布 $P_{\text{Binomial}}(k|n, a/(a+b))$ で近似されることをコンピュータで視覚化せよ. (コンピュータに不慣れなら, 以下の視覚化の結果を見るだけでもよい.)

2.4 解答例 : 二項分布がポリア分布の極限になっていること

$a = Lp, b = L(1-p)$ のとき $a + b = L$ なので, $L \rightarrow \infty$ とすると,

$$\begin{aligned}&P(k|n, Lp, L(1-p), d) \\&= \binom{n}{k} \frac{Lp(Lp+d) \cdots (Lp+(k-1)d) \cdot L(1-p)(L(1-p)+d) \cdots (L(1-p)+(n-k-1)d)}{L(L+d) \cdots (L+(n-1)d)} \\&= \binom{n}{k} \frac{p(p+d/L) \cdots (p+(k-1)d/L) \cdot (1-p)((1-p)+d/L) \cdots ((1-p)+(n-k-1)d/L)}{1(1+d/L) \cdots (1+(n-1)d/L)} \\&\rightarrow \binom{n}{k} p^k (1-p)^{n-k} = P_{\text{Binomial}}(k|n, p)\end{aligned}$$

コンピュータによる視覚化は以下の通り.

2.4.1 二項分布, ベータ二項分布, 超幾何分布の同時プロット(素朴な方法)

以下を [Download Julia \(https://julialang.org/downloads/\)](https://julialang.org/downloads/) からダウンロードしてインストールした Julia 言語を実行した状態で `julia>` プロンプトの画面に貼り付ければ, そのまま実行できるはずである.

[Distributions.jl \(https://github.com/JuliaStats/Distributions.jl\)](https://github.com/JuliaStats/Distributions.jl), [StatsPlots.jl \(https://github.com/JuliaPlots/StatsPlots.jl\)](https://github.com/JuliaPlots/StatsPlots.jl) (ほぼ [Plots.jl \(https://github.com/JuliaPlots/Plots.jl\)](https://github.com/JuliaPlots/Plots.jl)) の使い方はそれらのドキュメントを見ればわかる:

- <https://juliastats.org/Distributions.jl/stable/> (<https://juliastats.org/Distributions.jl/stable/>)
- <https://docs.juliaplots.org/stable/> (<https://docs.juliaplots.org/stable/>)

さらに次の検索結果も利用せよ:

- <https://www.google.com/search?q=Plots.jl> (<https://www.google.com/search?q=Plots.jl>)

```
In [3]: 1 # 使用パッケージの読み込み (少し時間がかかる)
        2 using Distributions, StatsPlots
```

```
In [4]: 1 # プロットの仕方のデフォルトの設定を変更
2 default(fmt=:png, size=(400, 250), titlefontsize=10,
3         tickfontsize=6, guidefontsize=9, plot_titlefontsize=12)
```

```
In [5]: 1 # 離散分布の確率質量関数を連続的に拡張
2 pmf(dist, x) = pdf(dist, round(Int, x))
```

Out[5]: pmf (generic function with 1 method)

```
In [6]: 1 # 分布のパラメータの設定
2 n, a, b = 20, 60, 140
3 p = a/(a+b)
```

Out[6]: 0.3

```
In [7]: 1 # 分布の設定
2 # 上から順に二項分布, ベータ二項分布, 超幾何分布
3 bin = Binomial(n, p)
4 bb = BetaBinomial(n, a, b)
5 hg = Hypergeometric(a, b, n)
```

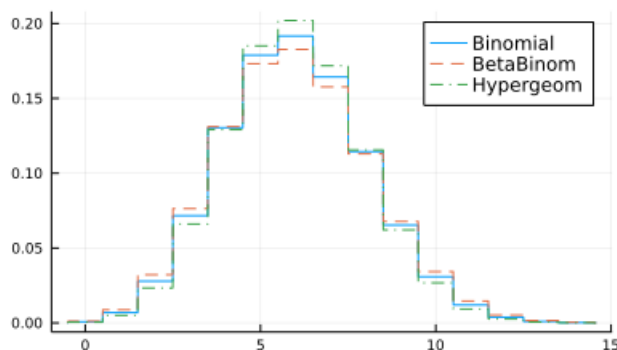
Out[7]: Hypergeometric(ns=60, nf=140, n=20)

```
In [8]: 1 # プロットする範囲
2 μ, σ = mean(bb), std(bb)
3 xmin, xmax = max(-0.51, μ-4σ), min(n+0.51, μ+4σ)
```

Out[8]: (-0.51, 14.576260187632494)

```
In [9]: 1 # グラフのプロット
2 plot(x → pmf(bin, x), xmin, xmax; label="Binomial")
3 plot!(x → pmf(bb, x), xmin, xmax; label="BetaBinom", ls=:dash)
4 plot!(x → pmf(hg, x), xmin, xmax; label="Hypergeom", ls=:dashdot)
5 title!("n=$n, a=$a, b=$b")
```

Out[9]: n=20, a=60, b=140



2.4.2 二項分布, ベータ二項分布, 超幾何分布の同時プロット(1つの関数にまとめる方法)

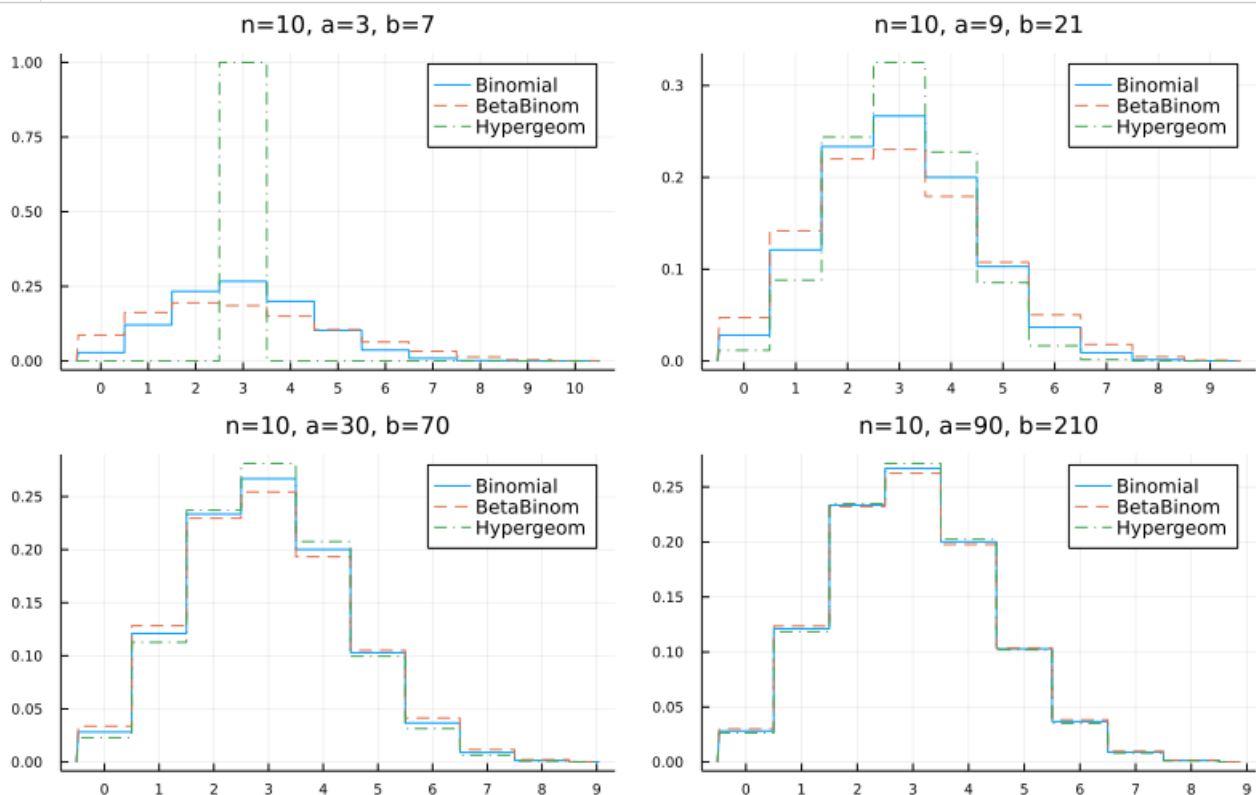
```
In [10]: 1 function plot_polya(n, a, b; F=Bool[1,1,1], xtick=0:n, kwargs...)
2     n ≤ a + b || (F[3] = false)
3     p = a/(a+b)
4     bin = Binomial(n, p)
5     bb = BetaBinomial(n, a, b)
6     F[3] && (hg = Hypergeometric(a, b, n))
7     μ, σ = mean(bb), std(bb)
8     xmin, xmax = max(-0.51, μ-4σ), min(n+0.51, μ+4σ)
9     plot()
10    F[1] && plot!(x → pmf(bin, x), xmin, xmax; label="Binomial", ls=:solid, c=1)
11    F[2] && plot!(x → pmf(bb, x), xmin, xmax; label="BetaBinom", ls=:dash, c=2)
12    F[3] && plot!(x → pmf(hg, x), xmin, xmax; label="Hypergeom", ls=:dashdot, c=3)
13    title!("n=$n, a=$a, b=$b")
14    plot!(; xtick, kwargs...)
15 end
```

Out[10]: plot_polya (generic function with 1 method)

超幾何分布は $n \leq a + b$ の場合にのみ表示される。

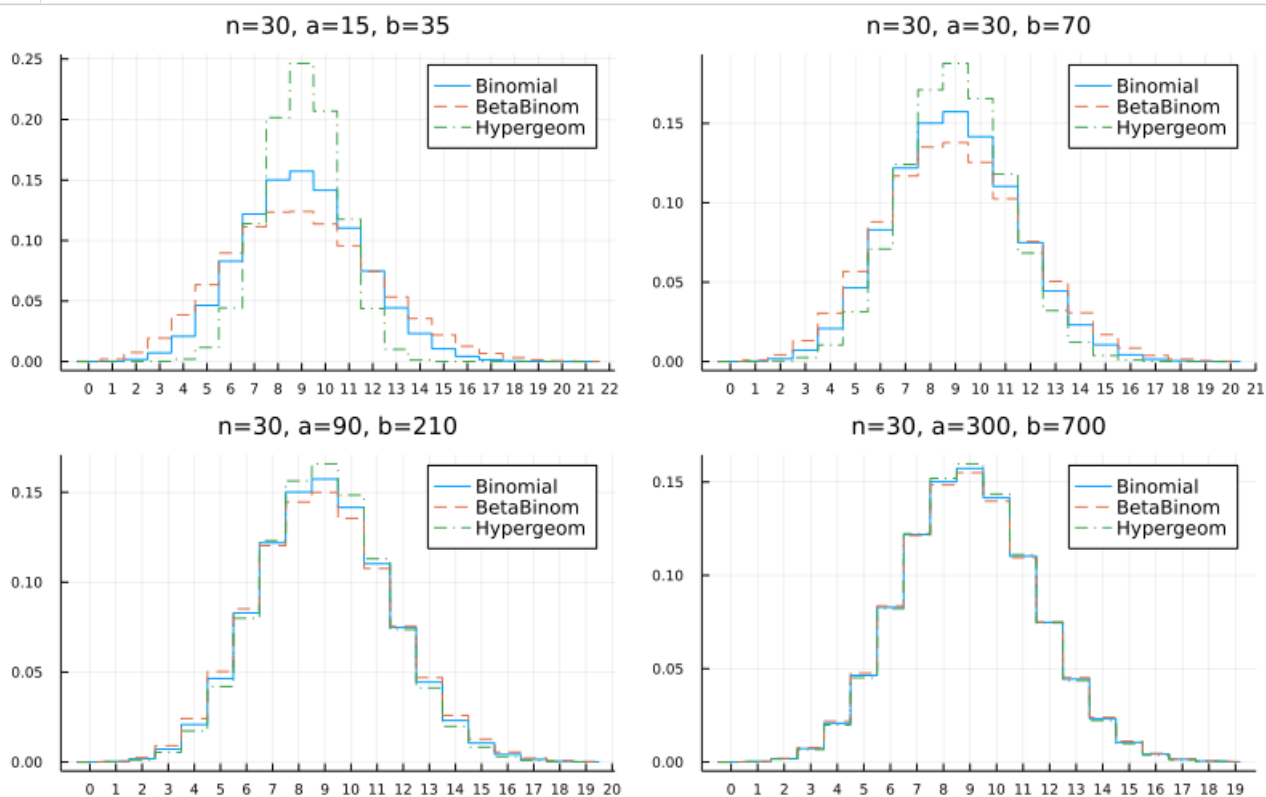
```
In [11]: 1 plot((plot_polya(10, 3m, 7m) for m in (1, 3, 10, 30))...;
2         size=(800, 500), layout=(2, 2))
```

Out[11]:



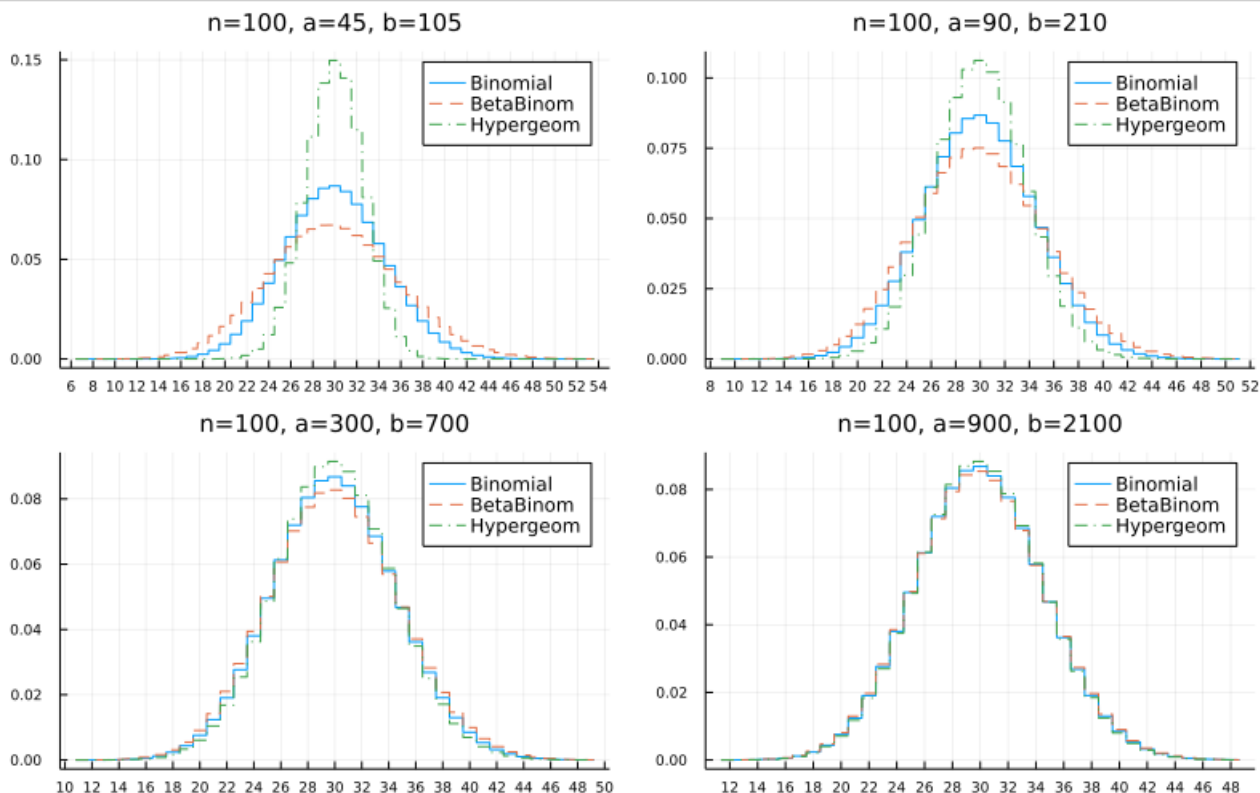
```
In [12]: 1 plot((plot_polya(30, 3m, 7m) for m in (5, 10, 30, 100))...;
2         size=(800, 500), layout=(2, 2))
```

Out[12]:



```
In [13]: 1 plot((plot_polya(100, 3m, 7m) for m in (15, 30, 100, 300))...;
2         size=(800, 500), layout=(2, 2), xtick=0:2:100)
```

Out[13]:



2.5 問題：ポリア分布の期待値と分散

ポリア分布の期待値

$$\mu = \sum_k k P(k|n, a, b, d)$$

と分散

$$\sigma^2 = \sum_k (k - \mu)^2 P(k|n, a, b, d) = \sum_k k^2 P(k|n, a, b, d) - \mu^2$$

がそれぞれ

$$\mu = \frac{na}{a+b}, \quad \sigma^2 = \frac{nab(a+b+nd)}{(a+b)^2(a+b+d)}$$

になることを示せ.

注意: この結果の $d = 1, 0, -1$ の場合より, それぞれベータ二項分布(ポリアの壺の場合), 二項分布, 超幾何分布の分散も求まったことになる. すなわち,

- $d = 1$ の場合: $\sigma^2 = \frac{nab(a+b+n)}{(a+b)^2(a+b+1)}$ (ベータ二項分布(ポリアの壺の場合)の分散),
- $d = 0$ の場合: $\sigma^2 = \frac{nab}{(a+b)^2}$ (二項分布の分散),
- $d = -1$ の場合: $\sigma^2 = \frac{nab(a+b-n)}{(a+b)^2(a+b-1)}$ (超幾何分布の分散).

このとき, $n = 1, 2, 3, \dots$ について

$$\frac{a+b-n}{a+b-1} \leq 1 \leq \frac{a+b+n}{a+b+1}$$

が成立しているので、

- ベータ二項分布の分散は二項分布以上になり、
- 超幾何分布の分散は二項分布以下になる

こともわかる。

ポリアの壺から出て来るベータ二項分布は二項分布的な現象で二項分布よりも分散が大きくなるような場合の統計モデルとして使われている。

2.6 解答例：ポリア分布の期待値と分散

$m = 0, 1, \dots, n$ であると仮定する。

ポリア分布における確率の総和が 1 であるという公式

$$\sum_k P(k|n, a, b, d) = \sum_k \binom{n}{k} \frac{a(a+d) \cdots (a+(k-1)d) \cdot b(b+d) \cdots (b+(n-k-1)d)}{(a+b)(a+b+d) \cdots (a+b+(n-1)d)} = 1$$

の n, a をそれぞれ $n-m, a+md$ で置き換え、二項係数中の k を $k-m$ で置き換えた場合を以下で使用する。

さらに、

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

から

$$k(k-1) \cdots (k-(m-1)) \binom{n}{k} = n(n-1) \cdots (n-(m-1)) \binom{n-m}{k-m}$$

であることも使用する。ただし、 $k-m < 0$ または $k-m > n-m$ のとき、 $\binom{n-m}{k-m} = 0$ であると約束しておく。

以上の準備を使うと、

$$\begin{aligned} & \sum_k k(k-1) \cdots (k-(m-1)) P(k|n, a, b, d) \\ &= \sum_k k(k-1) \cdots (k-(m-1)) \binom{n}{k} \frac{a(a+d) \cdots (a+(k-1)d) \cdot b(b+d) \cdots (b+(n-k-1)d)}{(a+b)(a+b+d) \cdots (a+b+(n-1)d)} \\ &= \sum_k n(n-1) \cdots (n-(m-1)) \binom{n-m}{k-m} \frac{a(a+d) \cdots (a+(k-1)d) \cdot b(b+d) \cdots (b+(n-k-1)d)}{(a+b)(a+b+d) \cdots (a+b+(n-1)d)} \\ &= n(n-1) \cdots (n-(m-1)) \frac{a(a+d) \cdots (a+(m-1)d)}{(a+b)(a+b+d) \cdots (a+b+(m-1)d)} \\ &\quad \times \sum_k \binom{n-m}{k-m} \frac{(a+md)(a+(m+1)d) \cdots (a+(k-1)d) \cdot b(b+d) \cdots (b+(n-k-1)d)}{(a+b+md)(a+b+(m+1)d) \cdots (a+b+(n-1)d)} \\ &= n(n-1) \cdots (n-(m-1)) \frac{a(a+d) \cdots (a+(m-1)d}{(a+b)(a+b+d) \cdots (a+b+(m-1)d)}. \end{aligned}$$

特に $m = 1, 2$ の場合より、

$$\begin{aligned} \mu &= \sum_k k P(k|n, a, b, d) = n \frac{a}{a+b} = \frac{na}{a+b}, \\ \sum_k k(k-1) P(k|n, a, b, d) &= n(n-1) \frac{a(a+d)}{(a+b)(a+b+d)}. \end{aligned}$$

ゆえに、

$$\begin{aligned}
\sigma^2 &= \sum_k k^2 P(k|n, a, b, d) - \mu^2 \\
&= \sum_k (k(k-1) + k) P(k|n, a, b, d) - \mu^2 \\
&= n(n-1) \frac{a(a+d)}{(a+b)(a+b+d)} + \mu - \mu^2 \\
&= \frac{nab(a+b+nd)}{(a+b)^2(a+b+d)}.
\end{aligned}$$

この計算の最後のステップについては以下の数式処理による計算を見よ.

```
In [14]: 1 using SymPy
2 @vars n a b d
3 expr = n*(n-1)*a*(a+d)/((a+b)*(a+b+d)) + n*a/(a+b) - (n*a/(a+b))^2
4 Eq(expr, expr.factor())
```

```
Out[14]: -\frac{a^2 n^2}{(a+b)^2} + \frac{an(a+d)(n-1)}{(a+b)(a+b+d)} + \frac{an}{a+b} = \frac{abn(a+b+dn)}{(a+b)^2(a+b+d)}
```

多項式や有理関数(=有理式)の因数分解は数式処理ソフトを使うとよい.

人間には不可能そうに見える因数分解もやってくれる場合がある.

3 ポリアの壺試行の別の解釈

3.1 ベータ分布, ベータ関数, ガンマ関数

a, b がともに正の実数のとき, 次の確率密度関数によって $0 < p < 1$ に関する連続的な確率分布が定められる:

$$f(p|a, b) = \frac{p^{a-1}(1-p)^{b-1}}{B(a, b)} \quad (0 < p < 1).$$

このようにして定まる確率分布をパラメータ (a, b) の **ベータ分布** と呼ぶ.

ただし, $B(a, b)$ は次のように定義される **ベータ関数** である:

$$B(a, b) = \int_0^1 p^{a-1}(1-p)^{b-1} dp \quad (a > 0, b > 0).$$

ベータ関数はガンマ関数で表せることがよく知られている(大学1年で習ったはず):

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}.$$

ガンマ関数 $\Gamma(s)$ は

$$\Gamma(s) = \int_0^\infty e^{-x} x^{s-1} dx \quad (s > 0)$$

と定義され,

$$\Gamma(s+1) = s\Gamma(s), \quad \Gamma(1) = 1, \quad \Gamma(1/2) = \sqrt{\pi}$$

などの公式を満たすこともよく知られている(大学1年で習ったはず).

この最初の2つの式より, $n = 0, 1, 2, 3, \dots$ のとき,

$$\begin{aligned}
\Gamma(n+1) &= n\Gamma(n) = n(n-1)\Gamma(n-1) \\
&= \dots = n(n-1)\dots 2 \cdot 1\Gamma(1) = n!
\end{aligned}$$

つまり, $\Gamma(s)$ は階乗 $\Gamma(n+1) = n!$ の一般化になっている.

上の3つの公式の3番目はGauss積分の公式に一致する: $x = t^2$ とおくと, $dx = 2t dt$ なので,

$$\Gamma(1/2) = \int_0^\infty e^{-x} x^{1/2} \frac{dx}{x} = 2 \int_0^\infty e^{-t^2} dt = \int_{-\infty}^\infty e^{-t^2} dt = \sqrt{\pi}.$$

これより, ガンマ関数はGauss積分の一般化にもなっていることがわかる.

要するに、ガンマ関数は階乗とGauss積分の両方の一般化になっている!

以下では、ガンマ関数について次が成立することが使われる: $k = 0, 1, 2, \dots$ について

$$\begin{aligned}\Gamma(s+k) &= (s+k-1)\Gamma(s+k-1) = (s+k-1)(s+k-2)\Gamma(s+k-2) \\ &= \dots = (s+k-1)(s+k-2)\dots(s+1)s\Gamma(s)\end{aligned}$$

となるので、

$$\frac{\Gamma(s+k)}{\Gamma(s)} = s(s+1)\dots(s+k-1).$$

3.2 成功確率がベータ分布に従ってランダムに決まっているようなベルヌーイ試行

次のようにして、1 と 0 の長さ n の列をランダムに生成することを考える.

(1) パラメータ (a, b) のベータ分布に従って、 $0 < p < 1$ をランダムに生成する.

(2) 確率 p で 1 を確率 $1-p$ で 0 をランダムかつ独立に n 回生成する(Bernoulli試行).

このとき、1 と 0 の列 x_1, \dots, x_n に対して、このように生成された 1 と 0 の列が x_1, \dots, x_n に等しくなる確率 $Q(x_1, \dots, x_n | a, b)$ は次のように計算される: $x_1 + \dots + x_n = k$ とおくと、

$$\begin{aligned}Q(x_1, \dots, x_n | a, b) &= \int_0^1 \prod_{i=1}^n (p^{x_i} (1-p)^{1-x_i}) \frac{p^{a-1} (1-p)^{b-1}}{B(a, b)} dp \\ &= \frac{1}{B(a, b)} \int_0^1 p^{a+k-1} (1-p)^{b+n-k-1} dp \\ &= \frac{1}{B(a, b)} B(a+k, b+n-k) \\ &= \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \frac{\Gamma(a+k)\Gamma(b+n-k)}{\Gamma(a+b+n)} \\ &= \frac{a(a+1)\dots(a+k-1) \cdot b(b+1)\dots(b+n-k+1)}{(a+b)(a+b+1)\dots(a+b+n-1)} \\ &= P(x_1, x_2, \dots, x_n | a, b).\end{aligned}$$

要するに、成功確率パラメータ p がベータ分布に従ってランダムに決まっているようなベルヌーイ試行は、ポリアの壺試行と同等である.

3.3 余談：生まれつきの才能か、努力の結果か？

以上によって、1 と 0 のランダムな列を生成する2つの方法は確率的に同等であることがわかった.

以下、 a, b は正の実数であると仮定する.

ポリアの壺試行 (a, b が整数でない場合への一般化)

(1) 初期状態は $(a_0, b_0) = (a, b)$ であるとする.

(2) すでに 1 または 0 がランダムに i 回生成されているときに以下を n 回実行する:

- 確率 $\frac{a_i}{a_i + b_i}$ で 1 を生成し、 $(a_{i+1}, b_{i+1}) = (a_i + 1, b_i)$ とおく.
- 確率 $\frac{b_i}{a_i + b_i}$ で 0 を生成し、 $(a_{i+1}, b_{i+1}) = (a_i, b_i + 1)$ とおく.

成功確率がベータ分布に従ってランダムに決められたベルヌーイ試行

(1) パラメータ (a, b) のベータ分布に従ってランダムに $0 < p < 1$ を生成する.

(2) 確率 p で 1 を確率 $1-p$ で 0 をランダムかつ独立に n 回生成する.

これら2つの方法で生成された 1 と 0 のランダムな列は確率的に区別できない.

これは以下のように考えると少し奇妙に感じられることでもある.

以下では 1 が出ることを「成功」と考え、0 が出ることを「失敗」と考える.

ポリアの壺試行では

- 成功するたびに次に成功する確率が高まり、失敗するたびに次に失敗する確率が高まる.

つまり、ポリアの壺試行では、成功失敗の相乗効果が働いていると考えられる。

成功確率がベータ分布に従ってランダムに決められたベルヌーイ試行では

- 最初に成功確率 p がベータ分布に従ってランダムに決められている。
- それ以後は最初にランダムに決めた成功確率 p に従って、成功と失敗を繰り返す。

そして、試行を沢山繰り返すと、大数の法則より、成功の割合は最初にランダムに決めた値 p に近づくことになる。

その p は **生まれつきの才能** のように解釈される。

この2つが確率的に同等であることから、ポリアの壺試行でも、試行回数を大きくする極限で、成功の割合はある一定の値 p に近づくことになる。

そして、成功の割合が近づく先の値 p が従う分布はパラメータ (a, b) のベータ分布になることもわかる。

ポリアの壺試行におけるその p は、成功と失敗の試行錯誤によってたどりついた **努力の結果** だと解釈される。

要するに、ポリアの壺試行と成功確率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が確率的に同等であるということは、生まれつきの才能としての p と努力の結果としての p が成功失敗の列を眺めただけでは区別できないことを意味している。

3.4 問題：ポリアの壺試行で大数の法則が成立することの確認

コンピュータによる計算で以下を行え。

(1) ポリアの壺試行において、試行回数 n を大きくすると、赤玉が出る回数 k の割合 k/n が一定の値 p に近づくことを確認せよ。

(2) その近づく先の値 p の分布がパラメータ (a, b) のベータ分布に従うことを確認せよ。

コンピュータの操作にまだ不慣れならば、以下の解答例をすぐに見てもよい。

これから数年かけて同じようなことをできるようになれば十分である。

3.5 解答例：ポリアの壺試行の側で大数の法則が成立することの確認

3.5.1 ポリアの壺試行の側で大数の法則が成立することの確認

```
In [15]: 1 # 使用パッケージの読み込み（少し時間がかかる）
        2 using Distributions, StatsPlots
```

```
In [16]: 1 # プロットの仕方のデフォルトの設定を変更
        2 default(fmt=:png, size=(400, 250), titlefontsize=10,
        3         tickfontsize=6, guidefontsize=9, plot_titlefontsize=12)
```

```
In [17]: 1 # 離散分布の確率質量関数を連続的に拡張
        2 pmf(dist, x) = pdf(dist, round(Int, x))
```

Out[17]: pmf (generic function with 1 method)

```
In [18]: 1 # ポリアの壺試行の結果(1と0の長さnの列)を返す。
        2 function rand_polya_urn(n, a, b; X = Vector{Int}(undef, n))
        3     for i in 1:n
        4         if rand() ≤ a/(a+b)
        5             X[i] = 1 # 赤玉が出た（成功した）
        6             a, b = a+1, b
        7         else
        8             X[i] = 0 # 白玉が出た（失敗した）
        9             a, b = a, b+1
        10        end
        11    end
        12    X
        13 end
```

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


```
In [19]: 1 using Random
2 Random.seed!(4649373)
3 @show rand_polya_urn(10, 3, 7);
4 @show rand_polya_urn(10, 3, 7);
5 @show rand_polya_urn(10, 3, 7);
6 @show rand_polya_urn(10, 3, 7);
7 @show rand_polya_urn(10, 3, 7);
8 @show rand_polya_urn(10, 3, 7);
9 @show rand_polya_urn(10, 3, 7);
10 @show rand_polya_urn(10, 3, 7);
11 @show rand_polya_urn(10, 3, 7);
12 @show rand_polya_urn(10, 3, 7);
```

```
rand_polya_urn(10, 3, 7) = [0, 1, 1, 0, 0, 0, 0, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [0, 1, 0, 1, 0, 0, 0, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [1, 1, 1, 0, 0, 0, 1, 0, 0, 1]
rand_polya_urn(10, 3, 7) = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [1, 0, 0, 1, 0, 0, 1, 0, 1, 0]
rand_polya_urn(10, 3, 7) = [0, 0, 0, 0, 0, 0, 1, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [0, 0, 0, 1, 0, 0, 1, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
rand_polya_urn(10, 3, 7) = [0, 0, 0, 1, 0, 1, 0, 1, 0, 0]
rand_polya_urn(10, 3, 7) = [1, 0, 0, 0, 0, 0, 0, 0, 0, 1]
```

```
In [20]: 1 # ポリアの壺試行の結果をプロット
2 # 横軸は試行回数で縦軸は1の割合
3 # kwargs は keyword arguments (キーワード引数達)の略.
4 function plot_polya_urn(a, b; n = 10^4, e=0.1, kwargs...)
5     X = rand_polya_urn(n, a, b)
6     prob = cumsum!(X, X) ./ (1:n)
7     ylim = prob[end] - e, prob[end] + e
8     plot(prob; label="")
9     plot!(; ylim, kwargs...)
10 end
```

Out[20]: plot_polya_urn (generic function with 1 method)

Julia言語の環境で ?cumsum! のように入力すればその解説を読むことができる。

```
In [21]: 1 ?cumsum!
```

search: cumsum! cumsum ContinuousMatrixDistribution

Out[21]: cumsum!(B, A; dims::Integer)

Cumulative sum of A along the dimension dims, storing the result in B. See also [cumsum](#) [.\(@ref\)](#).

問題: ?cumsum の結果も表示させてみよ。

例えば, $a = [1, 2, 3, 4, 5]$ のとき $\text{cumsum!}(a, a)$ によって a とその戻り値は $[1, 3, 6, 10, 15]$ になる。

```
In [22]: 1 a = [1, 2, 3, 4, 5]
2 @show a
3 cs = cumsum!(a, a)
4 @show a
5 @show cs;
```

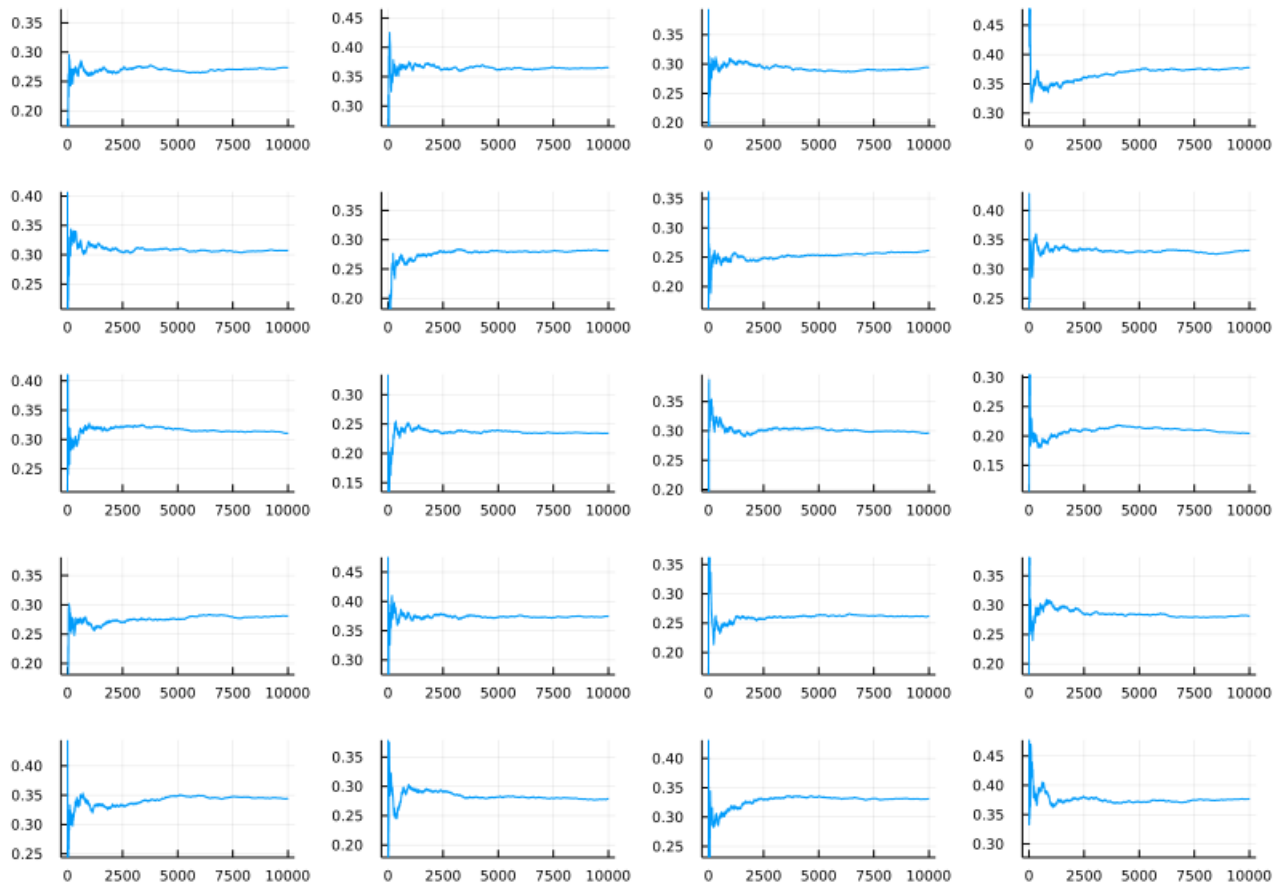
```
a = [1, 2, 3, 4, 5]
a = [1, 3, 6, 10, 15]
cs = [1, 3, 6, 10, 15]
```

```
In [23]: 1 # ポリアの壺試行の結果を5×4=20個プロット
2 function plot_polya_urns(a, b; n = 10^4, kwargs...)
3     PP = [plot_polya_urn(30, 70; n) for _ in 1:20]
4     plot(PP...; layout=(5, 4), size=(800, 600))
5     plot!(plot_title="Polya's urn trials: n=$n, a=$a, b=$b")
6     plot!(); kwargs...)
7 end
```

Out[23]: plot_polya_urns (generic function with 1 method)

In [24]: 1 plot_polya_urns(30, 70)

Out[24]: Polyá's urn trials: n=10000, a=30, b=70



確かに n を大きくすると、成功割合は一定の値に近付いているように見える。

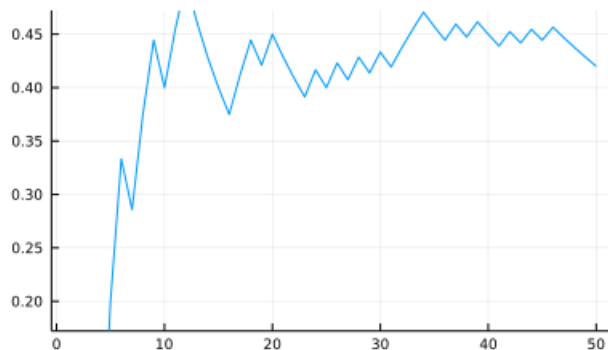
```
In [25]: 1 # 動画の作成
2 function anim_polya_urn(a, b; n = 10^4, e=0.15, fps=20, kwargs...)
3     X = rand_polya_urn(n, a, b)
4     prob = cumsum(X, X) ./ (1:n)
5     ylim = prob[end] - e, prob[end] + e
6     s = n ÷ 200
7     anim = @animate for t in [fill(s, fps); s:s:n; fill(n, fps)]
8         plot(view(prob, 1:t); label="")
9         title!("Polyá's urn trial: n=$n, a=$a, b=$b")
10        plot!(; ylim)
11    end
12    gif(anim, "anim_polya_urn.gif"; fps)
13 end
```

Out[25]: anim_polya_urn (generic function with 1 method)

```
In [26]: 1 using Random
2 Random.seed!(5963)
3 anim_polya_urn(30, 70)
```

```
Info: Saved animation to
  fn = D:\OneDrive\work\HighSchoolMath\anim_polya_urn.gif
@ Plots D:\.julia\packages\Plots\lW9ll\src\animation.jl:137
```

Out[26]: Polyá's urn trial: $n=10000$, $a=30$, $b=70$



上の動画はpdf版では動かない. pdf版の閲覧者は動画を見るためには以下のリンク先を参照せよ:

- https://github.com/genkuroki/HighSchoolMath/blob/master/anim_polya_urn.gif
(https://github.com/genkuroki/HighSchoolMath/blob/master/anim_polya_urn.gif)

3.5.2 ポリアの壺試行での大数の法則の収束先の分布がベータ分布になることの確認

```
In [27]: 1 # スレッド 並列化が有効になっていることの確認
2 # 値が2以上ならば有効になっている
3 # スレッド 並列化を有効にする方法については
4 # https://docs.julialang.org/en/v1/manual/multi-threading/#man-multithreading
5 # を参照せよ.
6 Threads.nthreads()
```

Out[27]: 12

```
In [28]: 1 # 私のパソコンでは次のように環境変数が設定されている.
2 ENV["JULIA_NUM_THREADS"]
```

Out[28]: "12"

```
In [29]: 1 # Monte Carlo シミュレーション
2 # ポリアの壺試行を L 回行って、最終的な成功割合を記録する.
3 # lln は law of large numbers の略
4 function mc_polya_urn_lln(a, b; n = 10^4, L = 10^5)
5     prob = Vector{Float64}(undef, L)
6     tmp = [Vector{Int}(undef, n) for _ in 1:Threads.nthreads()]
7     # Threads.@threads for ... end でスレッド 並列化.
8     # ループの内側は互いに独立に実行できればいけない.
9     Threads.@threads for i in 1:L
10         X = rand_polya_urn(n, a, b; X = tmp[Threads.threadid()])
11         prob[i] = sum(X)/n
12     end
13     prob
14 end
```

Out[29]: mc_polya_urn_lln (generic function with 1 method)

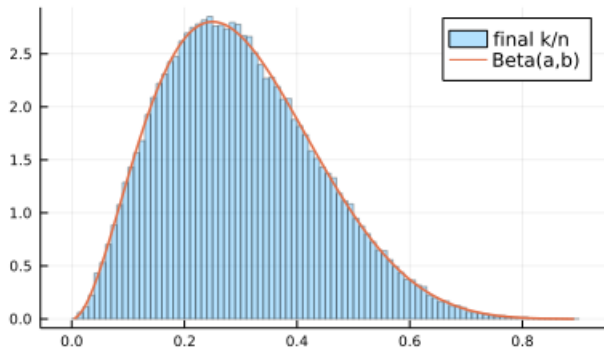
```
In [30]: 1 # ポリアの壺試行での大数の法則の収束先の分布をプロットする
2 function plot_polya_urn_lln(a, b; n = 10^4, L = 10^5, bin = 100, kwargs...)
3     prob = @time mc_polya_urn_lln(a, b; n, L)
4     xlim = extrema(prob)
5     histogram(prob; norm=true, alpha=0.3, label="final k/n", bin)
6     plot!(Beta(a, b), xlim...; label="Beta(a,b)", lw=1.5)
7     title!("n=$n, a=$a, b=$b (L=$L)")
8     plot!(; kwargs...)
9 end
```

Out[30]: plot_polya_urn_lln (generic function with 1 method)

```
In [31]: 1 plot_polya_urn_lln(3, 7)
```

0.696587 seconds (140.77 k allocations: 8.745 MiB, 9.44% compilation time: 40% of which was re compilation)

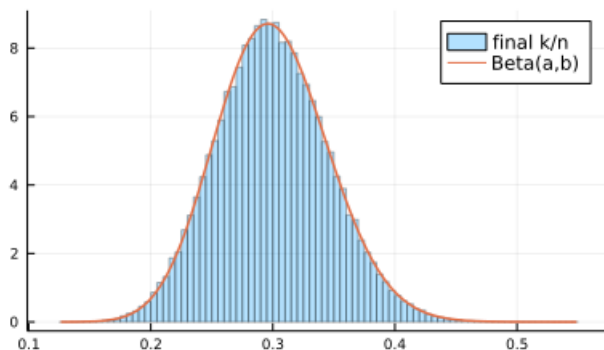
Out[31]: n=10000, a=3, b=7 (L=100000)



```
In [32]: 1 plot_polya_urn_lln(30, 70)
```

0.679218 seconds (105 allocations: 1.688 MiB)

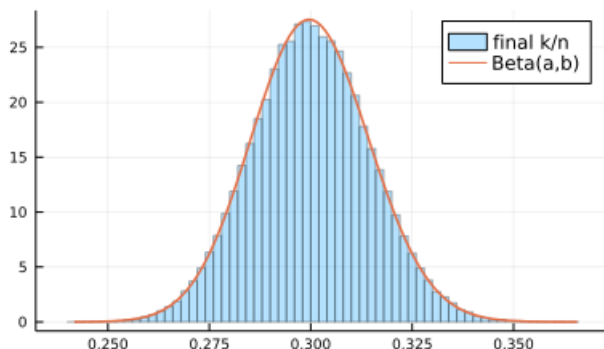
Out[32]: n=10000, a=30, b=70 (L=100000)



```
In [33]: 1 plot_polya_urn_lln(300, 700; n=10^5)
```

6.937067 seconds (108 allocations: 9.927 MiB)

Out[33]: n=100000, a=300, b=700 (L=100000)



以上のように、ポリアの壺試行での成功割合の収束先の分布が確かにベータ分布になっていることが数値的に確認された。

そうなることはすでに証明されていることなので、こうなるのは当たり前のことではある。

しかし, $n \rightarrow \infty$ における数学的結果が, コンピュータで計算できる有限の n においてどれだけの近似精度で成立しているかは実際に計算させないと分からないことが多い.

だから, 当然そうなるはずの結果であっても, コンピュータで確認しておくことにも数学的に価値がある.

そして, 以上のようにうまく行っていることがわかるきれいなグラフをプロットできると, 納得感が深まるという(自分に対する)教育的なメリットもある.

我々は人間なので, 純粋に論理的に納得するだけではなく, 気持ちの上でも納得できるように工夫することは大事なことである.

追記: 数学を教える側が数学を楽しんで勉強しているところを見せるのはとても大事なことだと思います。

In []:

1