Linear_regression

Masayuki Sakai

yyyy/mm/dd

Contents

1 最小二乗法

ここでは，n 個の対になる観測値，$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$ と $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^T \in \mathbb{R}^n$ について最小二乗法により線形モデルである

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \varepsilon_i \sim N(0, \sigma^2)$$

のパラメーター β_0, β_1 を求める．

これは，直線から上下の距離 $|y_i - \beta_0 - \beta_1 x_i|$ の二乗和 L を最小にする $\boldsymbol{\beta} = (\beta_0, \beta_1)^T \in \mathbb{R}^2$ を求める時に，L の偏微分を用いる．まず

$$L := \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

として，これを β_0, β_i の関数と見てそれぞれの偏微分を求めると

$$\frac{\partial L}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) = 0 \frac{\partial L}{\partial \beta_1} = -2 \sum_{i=1}^{n} x_i (y_i - \beta_0 - \beta_1 x_i) = 0$$

となる．この連立方程式を解くと

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

を得る．ただし $\sum_{i=1}^{n} (x_i - \bar{x})^2 \neq 0$ である．これは，共変量 $\boldsymbol{x}$ の値が全て同じことを意味しており，そのような変数はなんら情報を持たないためこのケースは実務上無視して良い．また，$\bar{x} = \sum x_i / n, \bar{y} = \sum y_i / n$ である．

ここで，$\boldsymbol{x}' = \boldsymbol{x} - \bar{x}, \boldsymbol{y}' = \boldsymbol{y} - \bar{y}$ とすると，$\bar{y}' = \bar{x}' = 0$ で $\bar{\beta}_0 = 0$，$\hat{\beta}_1 = \sum x_i y_i / \sum x_i^2$ となり計算が簡単になる．加減による傾きの影響は無いので，$\hat{\beta}_1$ の推定はこのままで問題はないが，切片項 β_0 はもとのスケールで計算する必要がある．

1.1 実装

ここでは，シンプルに $\boldsymbol{x}, \boldsymbol{y}$ という 2 つのベクトルを受け取った場合に，$\hat{\beta}_0, \hat{\beta}_1$ を返す関数を書く．

```
lsm <- function(x, y){
  if(length(unique(x)) == 1){
    stop("x has only 1 unique values")
  }
  # estimation
  bar_x <- mean(x); bar_y <- mean(y)
  hat_b1 <- sum((x-bar_x)*(y-bar_y)) / sum((x-bar_x)^2)
  hat_b0 <- bar_y - hat_b1 * bar_x

  # return
  return(list(b0 = hat_b0, b1 = hat_b1))
}
```

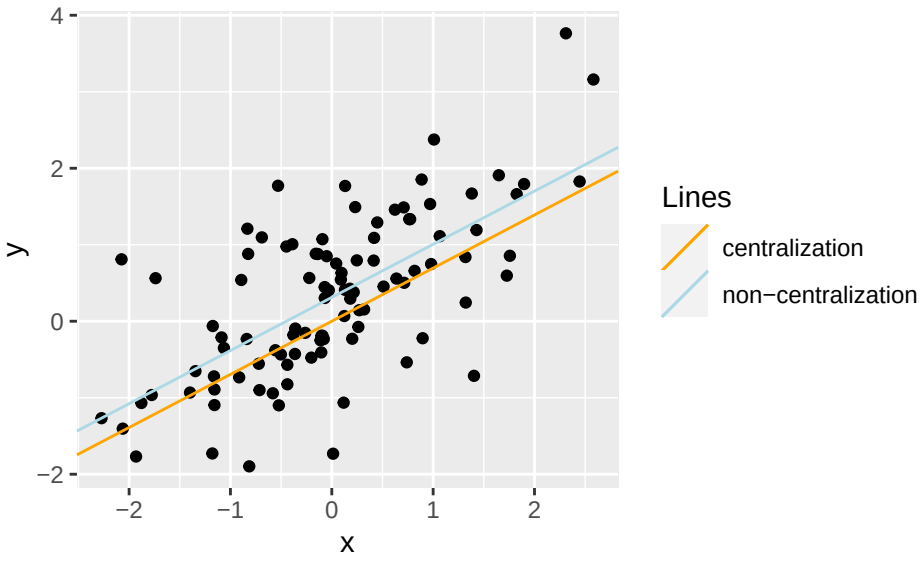
```
set.seed(100)
beta <- c(0.3, 0.8)
n <- 100
x <- c(rep(1,n), rnorm(n)) %>% matrix(nrow=100, byrow=FALSE)
y <- x %*% beta + rnorm(n)

fit <- lsm(x=x[,2], y=y[,1])
fit_centered <- lsm(x=x[,2]-mean(x[,2]), y=y-mean(y))

print(fit)
```

```
## $b0
## [1] 0.3114477
##
## $b1
## [1] 0.6946326
```

```
print(fit_centered)
```

```
## $b0
## [1] 2.193677e-17
##
## $b1
## [1] 0.6946326
```

```
data.frame(x=x[,2], y=y) %>%
  ggplot(aes(x=x, y=y)) +
  geom_point() +
  geom_abline(
    aes(slope = fit$b1,
        intercept = fit$b0,
        color = 'non-centralization'),
    show.legend = TRUE) +
  geom_abline(
    aes(slope = fit_centered$b1,
        intercept = fit_centered$b0,
        color = 'centralization'),
    show.legend = TRUE) +
  labs(color="Lines") +
  scale_color_manual(values = c('orange', 'lightblue'))
```



2　重回帰モデル

$$y := \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad X := \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}, \quad \boldsymbol{\beta} := \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

とすると，

$$L = \|y - X\boldsymbol{\beta}\|^2$$

とかけて，

$$\nabla L := \begin{bmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \end{bmatrix} = -2X^T(y - X\boldsymbol{\beta})$$

と表すことができる．ここから共変量の数を増やして

$$X := \begin{bmatrix} 1 & x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{bmatrix} \quad \boldsymbol{beta} := \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}$$

とする．このように拡張してもやることは変わらず，

$$-2X^T(y - X\boldsymbol{\beta}) = \mathbf{0}$$

を解く．結果として，

$$\hat{\boldsymbol{\beta}} = (X^TX)^{-1}X^Ty$$

を得る．ただし，$(X^TX)^{-1}$ は存在するとする．

2.1　実装

```
lsm <- function(X, y){
  res <- solve(t(X)%*%X) %*% t(X) %*% y
  return(res %>% as.vector)
}
```

```
n <- 100
p <- 3
set.seed(888); beta <- rnorm(4)
x <- matrix(1, nrow=n, ncol=p+1)
x[,-1] <- rnorm(100*p)
set.seed(666);y <- x %*% beta + rnorm(n)

fit <- lsm(X = x, y = y)
data.frame(beta = beta, fit = fit, diff = beta - fit)
```

```
##         beta        fit         diff
## 1 -1.9513433 -2.0154629  0.064119522
## 2 -1.5443662 -1.5412662 -0.003099945
## 3  0.7298327  0.7150430  0.014789653
## 4 -0.2775818 -0.3555181  0.077936312
```

2.2 考察

2.2.1 $(X^TX)^{-1}$ の存在

次の場合は $(X^TX)^{-1}$ が存在しない．
1. $N < p+1$ 2. X の異なる 2 列が等しい

2.2.2 関連する補題

Lemma 1. 正方行列 $A \in \mathbb{R}^{n\times n}$ について以下は同値である．*1.* A が正則 *2.* $\mathrm{rank}(A) = n$ *3.* $\det(A) \neq 0$

Lemma 2. 正方行列 A, B について，次が成り立つ．*1.* $\det(AB) = \det(A)\det(B)$ *2.* $\det(A^T) = \det(A)$

Lemma 3. $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times l}$ として，以下が成立．

$$\mathrm{rank}(AB) \leq \min\{\mathrm{rank}(A), \mathrm{rank}(B)\}$$
$$\mathrm{rank}(A^T) = \mathrm{rank}(A) \leq \min\{m, n\}$$

Lemma 4. V, W をそれぞれ $\mathbb{R}^n, \mathbb{R}^m$ の部分空間として，行列 $A \in \mathbb{R}^{m\times n}$ による線形写像 $V \to W$ の像と核は，それぞれ V, W の部分空間であって，それらの次元の和は n になる．また，その次元は，A の階数に一致する．

2.2.3 実務上の仮定

以降では，$X \in \mathbb{R}^{N\times(p+1)}$ の階数は $p+1$ であることを仮定する．$p = 1$ の場合は $\mathrm{rank(X)} = 2 = \mathrm{p} + 1$ と同値．

3 β の分布

被説明変数 $\boldsymbol{y} \in \mathbb{R}^N$ が共変量 $X \in \mathbb{R}^{N\times(p+1)}$ とその係数 $\boldsymbol{\beta} \in \mathbb{R}^{p+1}$ と誤差項 $\boldsymbol{\varepsilon} \sim N(\boldsymbol{\mu}, \sigma^2 I)$ によって下記のような関係にあると仮定する．

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

ここで，前項のように $\boldsymbol{\beta}$ と推定された $\hat{\boldsymbol{\beta}}$ は異なるものであることに注意しよう．あくまで $\boldsymbol{\beta}$ の値は未知である．いま，あらためて $\hat{\boldsymbol{\beta}}$ は次のように表せることを確認する．

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= (X^TX)^{-1}X^Ty \\ &= (X^TX)^{-1}X^T(X\boldsymbol{\beta} + \boldsymbol{\varepsilon}) \\ &= \boldsymbol{\beta} + (X^TX)^{-1}\boldsymbol{\varepsilon}\end{aligned}$$

となり，$\hat{\boldsymbol{\beta}}$ は $\boldsymbol{\varepsilon}$ によって決まることがわかる．$\boldsymbol{\varepsilon}$ が従う分布はその平均を $E[\boldsymbol{\varepsilon}] = \mathbf{0}$ と仮定していることから，$\hat{\boldsymbol{\beta}}$ の分布の平均は $\boldsymbol{\beta}$ となる．このように，ある推定量の平均が真値に一致する時，その推定量を不

偏推定量 (unbiased estimator) と呼ぶ．また，$\hat{\boldsymbol{\beta}}$ の分散を考えよう．それぞれの要素の分散 $V(\hat{\beta}_i)$ は定義通り $E(\hat{\beta}_i - \beta_i)^2$ である．また，それぞれの要素間の共分散 $\sigma_{i,j} = \mathrm{Cov}(\hat{\beta}_i, \hat{\beta}_j), (i \neq j)$ と定めれば，$\hat{\boldsymbol{\beta}}$ の共分散行列は，次のようになる．

$$\begin{aligned}
V(\hat{\boldsymbol{\beta}}) &= E[(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T] \\
&= E[(\boldsymbol{\beta} + (X^TX)^{-1}X^T\boldsymbol{\varepsilon} - \boldsymbol{beta})(\boldsymbol{\beta} + (X^TX)^{-1}X^T\boldsymbol{\varepsilon} - \boldsymbol{\beta})^T] \\
&= E[(X^TX)^{-1}X^T\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T(X^TX)^{-1}X^T] \\
&= (X^TX)^{-1}E[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T]X(X^TX)^{-1} \\
&= \sigma^2(X^TX)^{-1}
\end{aligned}$$

となる．これより $\hat{\boldsymbol{\beta}}$ の分布は下記のようになることがわかった．

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2(X^TX)^{-1})$$

4 二乗誤差の分布：Residual sum of squares

ここでは，$L = \|\boldsymbol{y} - X\hat{\boldsymbol{\beta}}\|^2$ の分布を導出する．まず $H := X(X^TX)^{-1}X^T$ として，その性質を確認していく．

以下を確認しなさい．

$$\begin{aligned}
H^2 &= H \\
(I-H)^2 &= I - H \\
HX &= X \\
\hat{\boldsymbol{y}} &= H\boldsymbol{y} \\
\boldsymbol{y} - \hat{\boldsymbol{y}} &= (I-H)\varepsilon
\end{aligned}$$

また，$H, I-H$ については次の補題が成り立つ．

Lemma 5. *$H, I-H$ の固有値は，$0, 1$ のみであり，H の固有値 1 と $I-H$ の固有値 0 の固有空間の次元は $p+1$，H の固有値 0 と $I-H$ の固有値 1 の固有空間の次元は $N-p-1$ となる．*

証明は後述する．

ここで RSS を次のように定義する．

$$\begin{aligned}
RSS &:= \|\boldsymbol{y} - \hat{\boldsymbol{y}}\| \\
&= [(I-H)\varepsilon]^T[(I-H)\varepsilon] \\
&= \varepsilon^T(I-H)^2\varepsilon \\
&= \varepsilon^T(I-H)\varepsilon
\end{aligned}$$

Lemma 6. 対称行列 A は，ある直交行列 P を用いて，$P^{-1}AP$ という，A の固有値が対角成分となる対角行列を作ることができる．

この補題により，実対称行列 $I-H$ はある直交行列 P によって対角行列 $P(I-H)P^T$ とできて，この固有値は $N-p-1$ 個が 1，その他は 0 なので対角成を次のように並び替えることができる．

$$P(I-H)P^T = \mathrm{diag}(1, \ldots, 0, \ldots, 0)$$

また，$\boldsymbol{v} = P\boldsymbol{\varepsilon} \in \mathbb{R}^N$ とすれば $\boldsymbol{\varepsilon} = P^T\boldsymbol{v}$ であり，

$$
\begin{aligned}
RSS &= \varepsilon^T (I - H)\varepsilon \\
&= [v_1, \ldots, v_{N-p-1}, v_{N-p}, \ldots, v_N]
\begin{bmatrix}
1 & 0 & \cdots & \cdots & \cdots & 0 \\
0 & \ddots & 0 & \cdots & \cdots & \vdots \\
\vdots & 0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & 0 & 0 & \cdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \cdots & \cdots & 0
\end{bmatrix}
\begin{bmatrix}
v_1 \\ \vdots \\ v_{N-p-1} \\ v_{N-p} \\ \vdots \\ v_N
\end{bmatrix} \\
&= \sum_{i=1}^{N-p-1} v_i^2
\end{aligned}
$$

とできる．また $E[P\varepsilon] = 0$ より，$\boldsymbol{w} = (v_1, v_2, \ldots, v_{N-p-1})$ である $\boldsymbol{w}$ に対して $E[\boldsymbol{w}] = 0$ が成り立つ．また，

$$
\begin{aligned}
E[\boldsymbol{v}\boldsymbol{v}^T] &= E[P\varepsilon(P\varepsilon)^T] \\
&= PE[\varepsilon\varepsilon^T]P^T \\
&= P\sigma^2\tilde{I}P^T \\
&= \sigma^2\tilde{I}
\end{aligned}
$$

が成り立つ．ここで $\tilde{I}$ は対角成分の最初から $N-p-1$ 個目までが 1 でそれ以降が 0 であるような対角行列である．

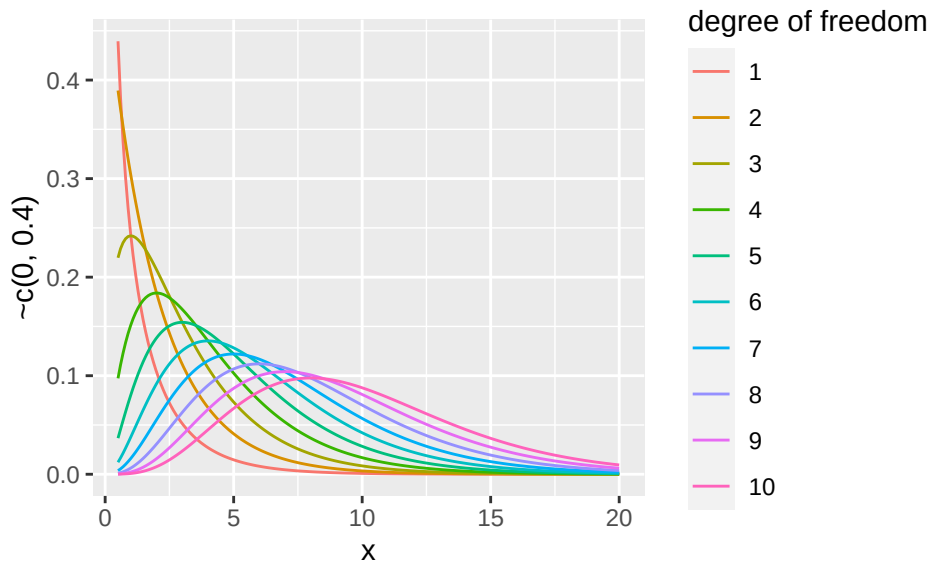
すなわち，$E[\boldsymbol{w}\boldsymbol{w}^T] = \sigma^2 I$ である．ここで，正規分布においては各変量の独立性と，それらの共分散行列が対角行列であることは同値なので，

$$
\frac{RSS}{\sigma^2} \sim \chi^2_{N-p-1}
$$

となる．ここで χ^2_m は自由度 m のカイ二乗分布を表す．これは，標準正規分布に従う m 個の独立な確率変数の二乗和が従う分布である．

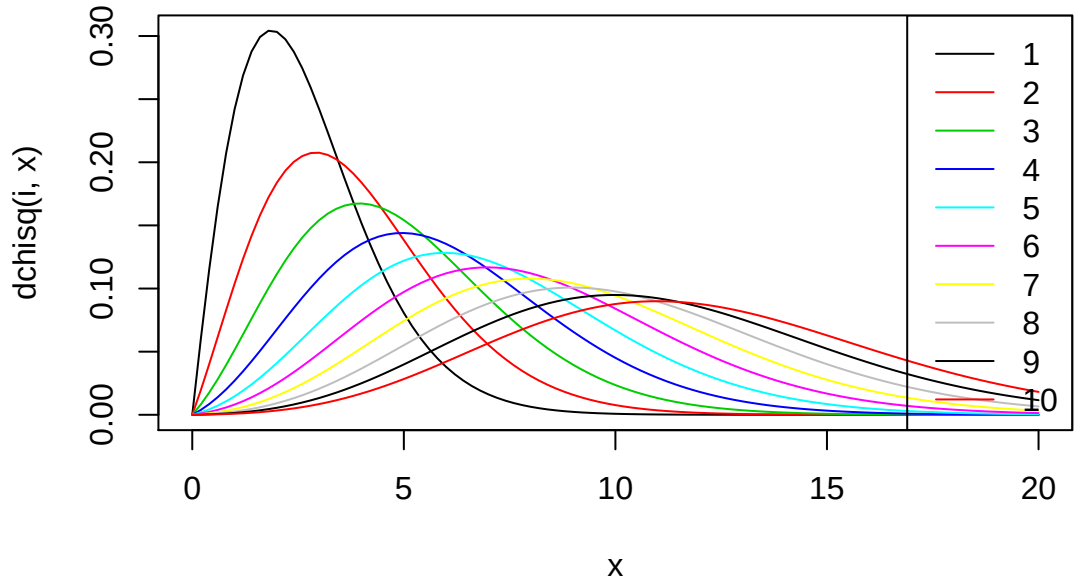
4.1 実験

```
x <- seq(0.5, 20, length.out = 200)
df_values <- c(1:10)
names(df_values) <- 1:10
df <- map_dfc(.x = as.list(df_values), .f = function(df){dchisq(x,df=df)}) %>%
  mutate(x=x) %>%
  pivot_longer(-x) %>%
  mutate(name = as.integer(name)) %>%
  ggplot(aes(x=x, y=value, group=name, color=as.factor(name))) +
    geom_line() +
    scale_y_continuous(aes(limits = c(0,0.4))) +
    labs(
      # title = TeX("PDFs of $\\chi^2$ distribution with parameters."),
      color ="degree of freedom"
      ) -> g

plot(g)
```



書籍にあったやつはこっち.

```
for(i in 1:10){
  curve(
    dchisq(i,x), 0, 20, col=i,
    add=ifelse(i==1, FALSE, TRUE),
    ann=ifelse(i==1,TRUE,FALSE)
  )
}
legend("topright", legend=1:10, col=1:10, lty=1)
```



5 $\hat{\beta}_j$ の仮説検定

回帰モデルの興味の 1 つは，推定した係数 $\hat{\boldsymbol{\beta}} = 0$ かどうかである．つまり，興味のある変数 $\boldsymbol{y}$ に対して影響の有無を確かめたい．

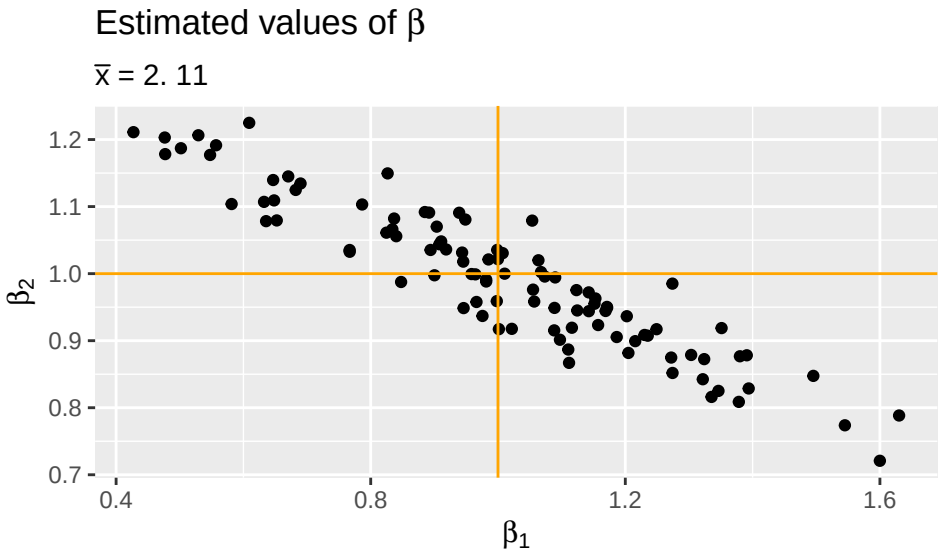
そのために，「もし $\hat{\boldsymbol{\beta}} = 0$ であったなら」という仮定のもと $\hat{\boldsymbol{\beta}}$ から計算できる統計量の分布に対する実際の推定値の値の関係を判断材料とする．

具体的には，$\hat{\boldsymbol{\beta}}$ が自由度 $N - p - 1$ の t 分布に従うことを確認し，仮説検定を構成する．

5.1 誤差による推定値のばらつき

```
set.seed(1)
n <- 100
x <- cbind(1, rnorm(n,2,1))
hat_beta <- matrix(0, ncol=2, nrow=100)
for(i in 1:100){
  y <- apply(cbind(x, rnorm(n,0,1)), 1, sum)
  fit <- lsm(X=x,y=y)
  hat_beta[i,1] <- fit[1]
  hat_beta[i,2] <- fit[2]
}
```

```
as.data.frame(hat_beta) %>%
  ggplot(aes(x=V1, y=V2)) +
  geom_point() +
  geom_vline(xintercept = 1, color = 'orange') +
  geom_hline(yintercept = 1, color = 'orange') +
  labs(title = TeX("Estimated values of $\\beta$"),
       subtitle = TeX(str_interp(
         "$\\bar{x}$ = $[.2f]{value}",
         list(value = mean(x[,2]))
       ),
       y = TeX("$\\beta_2$"),
       x = TeX("$\\beta_1$")
     )
```



単純なモデル $y_i = 1 + x_i + \varepsilon$，ただし $x_i \sim N(2,1), \varepsilon_i \sim N(0,1)$ としたモデルを考える．$n = 1, \ldots, 100$ として，x_i を一回サンプリングして固定する．その後

- ε_i をサンプリングして y_i を計算したのち，y_i, x_i から β_0, β_1 を推定する

という実験を 100 回繰り返した結果を図示した．モデルの定義より $\beta_0 = 1, \beta_1 = 1$ であるが，推定値は観測の際に含まれる誤差項の影響を受けて各回でばらついていることがわかる．

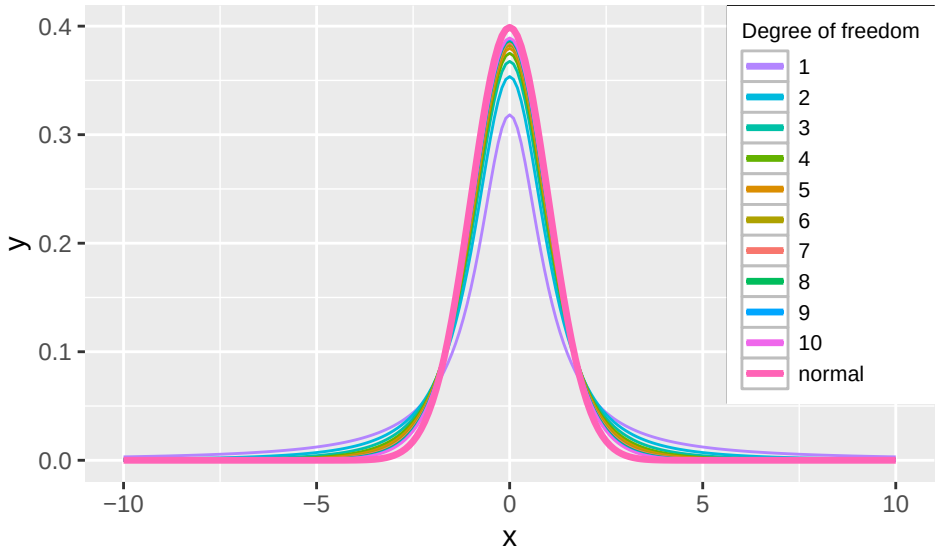
5.2 t 分布

t 分布の形状を確認しよう．いま，２つの独立な確率変数 $U \sim N(0,1)$ と $V \sim \chi^2_m$ を考える．この時 $T := U/\sqrt{V/m}$ という確率変数 T が従う分布が自由度 m の t 分布になる．

```
# ref:https://qiita.com/hoxo_b/items/c569da6dbf568032e04a
ggColorHue <- function(n, l=65) {
  hues <- seq(15, 375, length=n+1)
  hcl(h=hues, l=l, c=100)[1:n]
}
```

```
# warning が出るが無視
df_values <- 1:10
cols <- ggColorHue(n=10)

ggplot(data.frame(x=c(-10, 10)), aes(x)) +
  mapply(
    function(col, df) {
      stat_function(aes(color=col),fun = dt, args = list(df = df), n = 201)
    },
    df = df_values, col = cols) +
  stat_function(aes(color = 'normal'),
                fun = dnorm, args = list(mean=0, sd=1),
                n = 201, size = 1.2) +
  labs(color = "Degree of freedom") +
```

```
  scale_color_hue(
    breaks = c(cols, 'normal'),
    labels = c(1:10, 'normal')
  ) +
  theme(
    legend.title = element_text(color = "black", size = 8),
    legend.text = element_text(color = "black", size = 8),
    legend.position = c(1, 1), legend.justification = c(1, 1),
    legend.box.background = element_rect(size=0.1),
    legend.key = element_rect(fill = "white", colour = "gray", size=0.5),
    legend.key.height = unit(0.8,"line")
  )
```



太い線が $N(0,1)$ の密度関数である．自由度が大きくなると $N(0,1)$ の密度関数の形に近づいていることがわかる．

6 仮説検定

有意水準 $\alpha = 0.01, 0.05$ として検定統計量 t によって検定を行う．ここで帰無仮説は $\beta_j = 0$ である．帰無仮説が成立する下では $t \sim t_{N-p-1}$ となる．

いま，ε の標準偏差 σ の推定量を次のように構成する．

$$\hat{\sigma} := \sqrt{\frac{RSS}{N-p-1}}$$

また，$\hat{\beta}_j$ の標準偏差を

$$SE(\hat{\beta}_j) := \hat{\sigma}\sqrt{B_j}$$

とする．ただし，B_j は $(X^TX)^{-1}$ の j 番目の対角成分である．$p = 1$ のケースを考える．この時

$$
\begin{aligned}
(X^TX) &= \begin{bmatrix} 1 & \cdots & 1 \\ x_1 & \cdots & x_N \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \\
&= \begin{bmatrix} N & \sum_{i=1}^N x_i \\ \sum_{i=1}^N x_i & \sum_{i=1}^N x_i^2 \end{bmatrix} \\
&= \frac{1}{N} \begin{bmatrix} 1 & \bar{x} \\ \bar{x} & \frac{1}{N}\sum_{i=1}^N x_i^2 \end{bmatrix}
\end{aligned}
$$

となる．この逆行列は公式より，以下のようになる．

$$
(X^TX)^{-1} = \frac{1}{\sum_{i=1}^N (x_i - \bar{x})^2} \begin{bmatrix} \frac{1}{N}\sum_{i=1}^N x_i^2 & -\bar{x} \\ -\bar{x} & 1 \end{bmatrix}
$$

定義により，$B_1 = (\sum_{i=1}^N x_i^2)/(N\sum_{i=1}^N (x_i - \bar{x})^2), B_2 = 1/\sum_{i=1}^N (x_i - \bar{x})^2$ となる．$SE(\hat{\beta}_j) := \hat{\sigma}\sqrt{B_j}$ より，$B_j\sigma^2$ の推定値として $B_j\hat{\sigma}^2$ を採用する．

先ほどの実験によると，$\hat{\beta}_1, \hat{\beta}_2$ は負の相関を持っているように見える．これは $(X^TX)^{-1}$ の $(1,2),(2,1)$ 要素が $\bar{x} = 2.11$ であることによる．

ここからは，

$$
t = \frac{\hat{\beta}_j - \beta}{SE(\hat{\beta}_j)} \sim t_{N-P-1}
$$

が成り立つことを示す．まず，

$$
\begin{aligned}
U &:= \frac{\hat{\beta}_j - \beta}{\sqrt{B_j}\sigma} \sim N(0,1) \\
V &:= \frac{RSS}{\sigma^2} \sim \chi^2_{N-p-1}
\end{aligned}
$$

として，U, V が独立であることを示す。

$$
(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\boldsymbol{y} - \hat{\boldsymbol{y}})^T = (X^TX)^{-1}X^T\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T(I - H)
$$

より

$$
\begin{aligned}
E[(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\boldsymbol{y} - \hat{\boldsymbol{y}})^T] &= \sigma^2 IE[(X^TX)^{-1}X^T(I - H)] \\
&= \sigma^2 IE[(X^TX)^{-1}X^T - (X^TX)^{-1}X^TH] \\
&= 0
\end{aligned}
$$

となる．ここで，$HX = X \Leftrightarrow X^TH = X$ を用いた．また，$\boldsymbol{y} - \bar{\boldsymbol{y}} = (I - H)\boldsymbol{\varepsilon}$ は正規分布に従うこと，$\hat{\boldsymbol{\beta}}$ が正規分布に従うことから $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$ も正規分布に従うので，正規分布の性質から，$(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\boldsymbol{y} - \hat{\boldsymbol{y}})^T$ は互いに独立であることがわかる．

U は $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$ の関数, V は RSS，すなわち $\boldsymbol{y} - \hat{\boldsymbol{y}}$ の関数であるので，上記の理由により U, V は互いに独立であることが導かれた．

6.1 実験：検定の統計量の手計算と関数の結果比較

帰無仮説 $H_0 : \beta_j = 0$，対立仮説 $H_1 : \beta_j \neq 0$ という仮説検定を行う．$p = 1$ として，H_0 が成り立つ下での

$$t = \frac{\hat{\beta}_i - 0}{SE(\hat{\beta}_i)} \sim t_{N-p-1}$$

という性質を利用して，R の線形モデルの推定関数 lm の値を考察する．

```
# sample data
set.seed(100)
N <- 100
x <- rnorm(N)
y <- rnorm(N)
p <- 1

# estimate beta
beta_1 <- sum((x-mean(x))*(y-mean(y))) / sum((x-mean(x))^2)
beta_0 <- mean(y) - beta_1*mean(x)
beta   <- c(beta_0, beta_1)

# calc statistics
X <- cbind(1, x)
B <- diag(solve(t(X) %*% X))

RSS <- sum((y - beta_0 - beta_1*x)^2)
SE <- sqrt(RSS/(N-p-1)) * sqrt(B)

t <- beta/SE
p <- 2 * (1 - pt(abs(t), N-2))

# print results
results <- data.frame(
  beta = beta,
  SE = SE,
  t_value = t,
  p_value = p
)

row.names(results) <- c(0,1)

results
```

```
##          beta         SE    t_value   p_value
## 0  0.01144773 0.07929094  0.1443762 0.8854999
## 1 -0.10536744 0.07807314 -1.3495991 0.1802540
```

デフォルト関数の結果と比較すると一致していることがわかる．

```
summary(lm(y~x))$coef %>% data.frame()
```

```
##                Estimate Std..Error    t.value  Pr...t..
## (Intercept)  0.01144773 0.07929094  0.1443762 0.8854999
## x           -0.10536744 0.07807314 -1.3495991 0.1802540
```

6.2 実験：$\hat{\beta}$ の分布

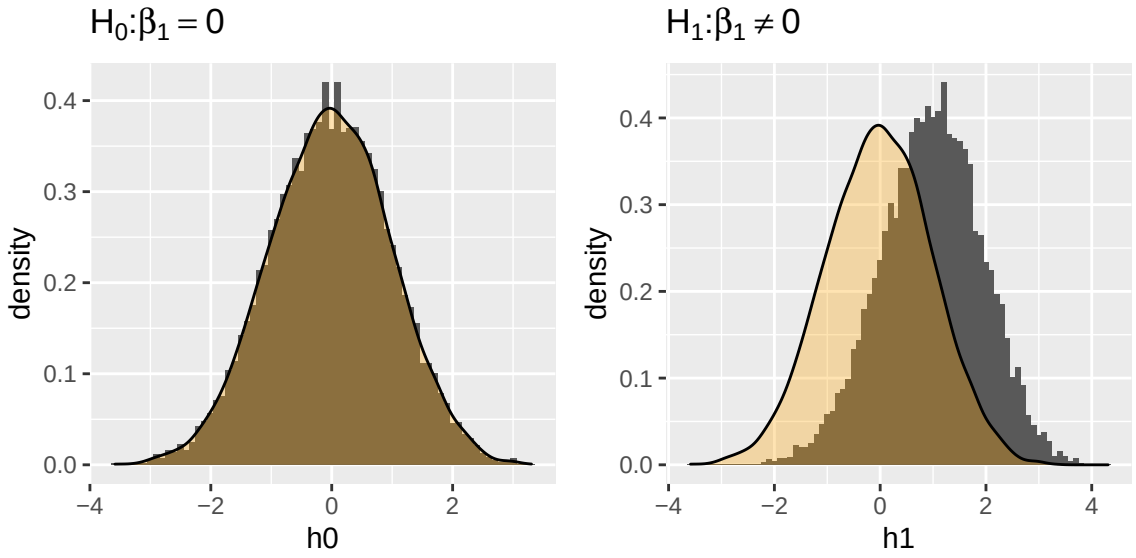
上記の要領で $\hat{\beta}$ の推定を繰りかえし，$\hat{\beta}_1/SE(\hat{\beta}_1)$ のヒストグラムを描く．

```
lsm <- function(X, y){
  res <- solve(t(X)%*%X) %*% t(X) %*% y
  return(res %>% as.vector)
}

estimate_beta <- function(x, y){
  X <- cbind(1, x)

  # estimate beta
  beta <- lsm(X, y)

  # calc statistics
  B <- diag(solve(t(X) %*% X))
  RSS <- sum((y - beta_0 - beta_1*x)^2)
  SE <- sqrt(RSS/(N-p-1)) * sqrt(B)
  t <- beta/SE
  p <- 2 * (1 - pt(abs(t), N-2))

  # return
  res <- list(beta=beta, SE=SE, t=t, p=p)
  return(res)
}
```

```
set.seed(100)
N <- 100
r <- 10000
results <- matrix(NA, nrow = 10000, ncol = 2)

for(i in 1:r){
  x  <- rnorm(N)
  y1 <- rnorm(N)
  y2 <- 0.1*x + rnorm(100)
  results[i,] <- c(estimate_beta(x, y1)$t[2], estimate_beta(x, y2)$t[2])
}

results <- results %>% as.data.frame()
colnames(results) <- c("h0", "h1")
```

```
results %>%
  ggplot(aes(x=h0, y=..density..)) +
  geom_histogram(binwidth = .1, position = "identity") +
  geom_density(fill='orange', alpha=0.3, show.legend = FALSE) +
  labs(title = TeX("$H_0:\\beta_1 = 0$")) -> p1

results %>%
  ggplot(aes(x=h1, y=..density..)) +
  geom_histogram(binwidth = .1, position = "identity") +
  geom_density(aes(x=h0, y=..density..), fill = 'orange',
               alpha=0.3, show.legend = FALSE) +
```

```
  labs(title = TeX("$H_1:\\beta_1 \\neq 0$")) -> p2

gridExtra::grid.arrange(p1, p2, ncol=2, widths = c(2,2))
```



7　決定係数と共線型性の検出

まず

$$
\begin{aligned}
RSS &:= \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|^2 = \|(I-H)\boldsymbol{y}\|^2 \\
ESS &:= \|\hat{\boldsymbol{y}} - \bar{\boldsymbol{y}}\|^2 = \|(H-W)\boldsymbol{y}\|^2 \\
TSS &:= \|\boldsymbol{y} - \bar{\boldsymbol{y}}\|^2 = \|(I-W)\boldsymbol{y}\|^2
\end{aligned}
$$

ここで，$W \in \mathbb{R}^{N\times N}, Wi,j = 1/N, i,j = 1,\ldots,N$ とする．また，$TSS = RSS + ESS$ が成り立つ．この時，$HX = X$ より，$HW = W$ が成立する．これより $(I-H)(H-W) = H-W-H^2+HW = H-W-H+W = 0$ が成り立つ．特に，

$$
R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}
$$

を決定係数 (coefficient of determination) という．この時，R^2 は $p = 1$ の場合，標本相関係数 $\hat{p}$ の二乗に一致する．

$$
\hat{p} := \frac{(\boldsymbol{x} - \bar{x})^T(\boldsymbol{y} - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})}}
$$

$$
\begin{aligned}
\frac{ESS}{TSS} &= \frac{\hat{\beta}_1 \|\boldsymbol{x} - \bar{x}\|^2}{\|\boldsymbol{y} - \bar{y}\|^2} \\
&= \left\{ \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^N (x_i - \bar{x})^2} \right\}^2 \frac{\sum_{i=1}^N (x_i - \bar{x})^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \\
&= \frac{\left[\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})\right]^2}{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})^2} \\
&= \hat{p}^2
\end{aligned}
$$

また，これに対して，

$$
1 - \frac{RSS/(N-p-1)}{TSS/(N-1)}
$$

という指標も使われ，これを調整済み決定係数 (adjusted coefficient of determination) という．これは RSS, TSS をそれぞれの自由度で割ったものである．説明変数が多くなる (p の値が大きくなる) 場合，決定係数は大きくなるが，調整済み決定係数は小さくなるように補正される．意味合いとして，説明変数を多くすることにペナルティを課すことでおよそ目的変数に関係の無い説明変数を加えることによる過学習を是正することが期待されている．

7.1 決定係数の計算

```
calc_r2 <- function(x,y,beta){
  # x is covariate matrix that include 1-column
  # y is target variable
  yhat <- x %*% beta
  ess  <- sum((yhat-mean(y))^2)
  tss  <- sum((y-mean(y))^2)
  rss  <- tss - ess
  r2   <- ess/tss
  adjtd_r2 <- 1 - {rss/(length(y)-ncol(x))} / {tss/(length(y)-1)}
  return(list(r2=r2, adjtd_r2=adjtd_r2))
}
```

```
set.seed(100)
N <- 100
x  <- cbind(1, rnorm(N))
y <- rnorm(N)
beta <- lsm(x, y)
str_interp("r2 = $[.3f]{r2}, adjusted_r2 = $[.3f]{adjtd_r2}",
           calc_r2(x, y, beta)) %>%
  print
```

```
## [1] "r2 = 0.018, adjusted_r2 = 0.008"
```

```
res <- summary(lm(y~x[,2]))
str_interp("r2 = $[.3f]{r2}, adjusted_r2 = $[.3f]{ar2}",
           list(r2 = res$r.squared, ar2 = res$adj.r.squared)) %>%
  print
```

```
## [1] "r2 = 0.018, adjusted_r2 = 0.008"
```

また，$VIF := 1/(1 - R^2_{X_j|X_{-j}})$ という指標があり，これは説明変数同士に冗長なものが無いかどうかを測る尺度として用いられる．ここで $R^2_{X_j|X_{-j}}$ は目的変数に j 番目の説明変数を，説明変数として j 番目を除いた全ての説明変数としたモデルにおける R^2 である．

```
calc_vif <- function(x){
  res <- rep(NA, ncol(x)-1)
  for(i in 2:ncol(x)){
    beta <- lsm(X=x[,-i], y=x[,i])
    res[i-1] <- 1/(1-calc_r2(x[,-i], x[,i], beta)$r2)
  }
  return(res)
}
```

```
require(MASS)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
x <- cbind(1, Boston) %>% as.matrix()
res <- calc_vif(x)
print(res)
```

```
##  [1] 1.831537 2.352186 3.992503 1.095223 4.586920 2.260374 3.100843
##  [8] 4.396007 7.808198 9.205542 1.993016 1.381463 3.581585 3.855684
```

8 信頼区間と予測区間

ここからは $\hat{\boldsymbol{\beta}}$ の分布を利用して信頼区間について考えていく．t の従う分布の確率密度関数を f とおいた時，

$$\frac{\alpha}{2} = \int_t^\infty f(u)du$$

となる t を $t_{N-p-1}(\alpha/2)$ とすると，$\hat{\beta}_j$ の信頼区間として，

$$\beta_i = \hat{\beta}_i \pm t_{N-p-1}(\alpha/2)SE(\hat{\beta}_i),\ i = 0, 1, \ldots, p$$

が得られる．

$\boldsymbol{x}_*\hat{\boldsymbol{\beta}}$ を，推定に用いた $\boldsymbol{x}$ とは異なる点 $\boldsymbol{x}_* \in \mathbb{R}^{p+1}$ とおく．この時，$\boldsymbol{x}_*\hat{\boldsymbol{\beta}}$ の平均

$$E[\boldsymbol{x}_*\hat{\boldsymbol{\beta}}] = \boldsymbol{x}_* E[\hat{\boldsymbol{\beta}}]$$

であり，定数行列 $A \in \mathbb{R}^{n\times m}$ と確率変数ベクトル $X \in \mathbb{R}^n$ について，$V(AX) = AV(X)A^T$ が成り立つことから，

$$\begin{aligned} V[\boldsymbol{x}_*\hat{\boldsymbol{\beta}}] &= \boldsymbol{x}_* V(\hat{\boldsymbol{\beta}}) {\boldsymbol{x}_*}^T \\ &= \sigma^2 \boldsymbol{x}_* (X^T X)^{-1} \boldsymbol{x}_*^T \end{aligned}$$

を得る．ここで，$\hat{\sigma}$ と $SE(\boldsymbol{x}_*\hat{\boldsymbol{\beta}})$ をそれぞれ

$$\begin{aligned} \hat{\sigma} &:= \sqrt{RSS/(N-p-1)} \\ SE(\boldsymbol{x}_*\hat{\boldsymbol{\beta}}) &:= \hat{\sigma}\sqrt{\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*^T} \end{aligned}$$

として，

$$\begin{aligned} C &:= \frac{\boldsymbol{x}_*\hat{\boldsymbol{\beta}} - \boldsymbol{x}_*\boldsymbol{\beta}}{SE(\boldsymbol{x}_*\hat{\boldsymbol{\beta}})} \\ &= \frac{\boldsymbol{x}_*\hat{\boldsymbol{\beta}} - \boldsymbol{x}_*\boldsymbol{\beta}}{\sigma\sqrt{\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*^T}} \Bigg/ \sqrt{\frac{RSS}{\sigma^2} \Bigg/ (N-p-1)} \\ &\sim t_{N-p-1} \end{aligned}$$

が得られる．$\frac{\boldsymbol{x}_*\hat{\boldsymbol{\beta}} - \boldsymbol{x}_*\boldsymbol{\beta}}{\sigma\sqrt{\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*^T}}$ は $\boldsymbol{x}_*\hat{\boldsymbol{\beta}}$ の平均を引いた物をその標準偏差で割っている形になっているので標準正規分布に従う．分母の部分は $RSS/\sigma^2 \sim \chi^2_{N-p-1}$ であり，$RSS, \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$ が独立であることがわかっているので，t_{N-p-1} に従うことがわかる．

また，観測誤差 $\boldsymbol{\varepsilon}$ を考慮した信頼区間を予測区間 $prediction interval$ と呼ぶことが多い．この場合においても分散が評価できて，その場合は単純に観測誤差 $\boldsymbol{\varepsilon}$ の分散 σ^2 を足すだけである．この場合は

$$\begin{aligned} P &:= \frac{\boldsymbol{x}_*\hat{\boldsymbol{\beta}} - \boldsymbol{y}_*}{\sigma(1+\sqrt{\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*})} \Bigg/ \sqrt{\frac{RSS}{\sigma} \Bigg/ (N-p-1)} \\ &\sim t_{N-p-1} \end{aligned}$$

が成り立つ．

以上より，棄却域を確率 α として定めると信頼区間は

$$\boldsymbol{x}_*\boldsymbol{\beta} = \boldsymbol{x}_*\hat{\boldsymbol{\beta}} \pm t_{N-p-1}(\alpha/2)\hat{\sigma}\sqrt{\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*^T}$$

となる．予測区間は

$$\boldsymbol{y}_* = \boldsymbol{x}_*\hat{\boldsymbol{\beta}} \pm t_{N-p-1}(\alpha/2)\hat{\sigma}\sqrt{1+\boldsymbol{x}_*(X^TX)^{-1}\boldsymbol{x}_*^T}$$

として得られる．

8.1 実験

```r
est_beta <- function(x,y){
  res <- solve(t(x)%*%x) %*% t(x) %*% y
  return(res %>% as.vector)
}

calc_rss <- function(x,y){
  betahat <- est_beta(x,y)
  yhat <- x %*% betahat
  ess  <- sum((yhat-mean(y))^2)
  tss  <- sum((y-mean(y))^2)
  rss  <- tss - ess
  return(list(tss=tss, rss=rss, ess=ess))
}

est_sigma <- function(x,y,params){
  N <- params$N; p <- params$p
  rss <- calc_rss(x,y)$rss
  res <- sqrt(rss/(n-p-1))
  return(res)
}

est_conf_interval <- function(xast, x, y, params){
  N <- params$N; p <- params$p; alpha <- params$alpha
  betahat  <- est_beta(x,y)
  inv_xtx  <- solve(t(x)%*%x)
  sigmahat <- est_sigma(x,y,params)
  cent     <- xast %*% betahat
  interval <- apply(xast, 1, function(newx){
   qt(p=1-alpha/2, df=N-p-1) * sigmahat * sqrt({newx %*% inv_xtx} %*% newx)
  })
  res <- data.frame(
    conf_lower = cent - interval,
    conf_cent = cent,
    conf_upper = cent + interval)
  return(res)
}

est_pred_interval <- function(xast, x, y, params){
  N <- params$N; p <- params$p; alpha <- params$alpha
  betahat  <- est_beta(x,y)
  inv_xtx  <- solve(t(x)%*%x)
  sigmahat <- est_sigma(x,y,params)
  cent     <- xast %*% betahat
  interval <- apply(xast, 1, function(newx){
   qt(p=1-alpha/2, df=N-p-1) * sigmahat * sqrt(1 + {newx %*% inv_xtx} %*% newx)
  })
  res <- data.frame(
    pred_lower = cent - interval,
    pred_cent = cent,
    pred_upper = cent + interval)
  return(res)
}
```

```r
set.seed(200)
params <- list(
  N = 100, p = 1,
  beta = c(1,1),
  eps = list(mean=0, sd=1),
  alpha = 0.05
)
x    <- cbind(1, matrix(rnorm(params$N*params$p), ncol=params$p))
xast <- cbind(1, seq(from=-10, to=10, length.out = 300))
eps  <- rnorm(mean = params$eps$mean, sd = params$eps$sd, n = params$N)
y    <- X %*% params$beta + eps

upper_bounds <- bind_cols(
    x=xast[,2],
    est_conf_interval(xast = xast, x = X,y = y,params = params) %>%
      dplyr::select(conf_cent, conf_upper),
    est_pred_interval(xast = xast, x = X,y = y,params = params) %>%
      dplyr::select(pred_upper)
  ) %>%
  rename(
    cent = conf_cent,
    conf = conf_upper,
    pred = pred_upper
  ) %>%
  pivot_longer(
    cols = c("conf", "pred"),
    names_to = "interval",
    values_to = "upper_value"
  )

lower_bounds <- bind_cols(
    x=xast[,2],
    est_conf_interval(xast = xast, x = X,y = y,params = params) %>%
      dplyr::select(conf_cent, conf_lower),
    est_pred_interval(xast = xast, x = X,y = y,params = params) %>%
      dplyr::select(pred_lower)
  ) %>%
  rename(
    cent = conf_cent,
    conf = conf_lower,
    pred = pred_lower
  ) %>%
  pivot_longer(
    cols = c("conf", "pred"),
    names_to = "interval",
    values_to = "lower_value"
  )

intervals <- left_join(upper_bounds,
                       lower_bounds,
                       by = c("x", "cent", "interval"))
```

```
intervals %>%
  ggplot(aes(x=x, y=cent)) +
  geom_ribbon(
    aes(ymin=lower_value,
        ymax=upper_value,
        group = interval,
        color = interval,
        linetype = interval),
    alpha = 0) +
  geom_line()
```