

ビジネスに役立つデータ分析 （入門編）

瓜生真也（徳島大学デザイン型AI教育研究センター）

諸注意

資料置き場: https://github.com/uribo/cue2022aw_r104

投影するプレゼンテーション、ソースコードを置いています
(来週分は来週更新)

Rコードと実行環境

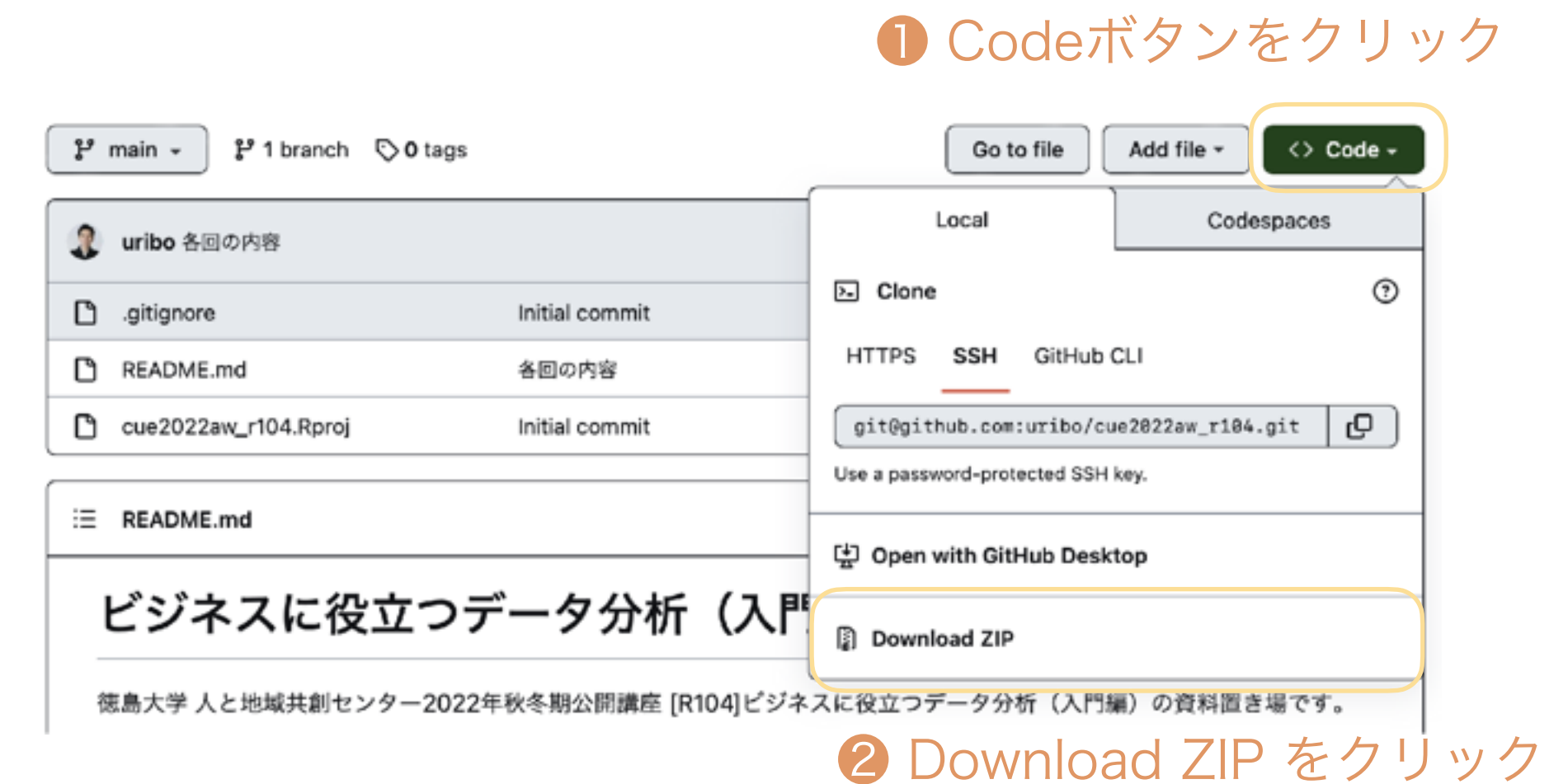
```
# 気温とアイスの相関係数を求める
cor(df_icecream_temperature$temperature_average_c,
    df_icecream_temperature$value)
#> [1] 0.9144466
```



Rでの実行コマンドであることを示します

https://mybinder.org/v2/gh/uribo/cue2022aw_r104/main?urlpath=rstudio

から必要なパッケージ、データ、ソースコードを含んだRStudioが起動します



第三週: データ分析入門（1）

瓜生真也（徳島大学デザイン型AI教育研究センター）

講座の内容

第一週

第二週

第三週

データサイエンス入門 (1)

回帰モデル

線形単回帰モデル

線形重回帰モデル

分類モデル

第四週

第五週

データ分析の目的（本丸） ... 未知のデータへの予測

既存のデータとデータの関係性を説明する**モデル**により、
未知のデータが得られた場合の予測を行う



データ分析の手法

体の一部の部位から他の部位のサイズを推定

体の部位から種名を判定

① 回帰モデル

② 分類モデル

回帰モデル

線形単回帰モデル

2つの変数 (x, y) の間に $y = f(x)$ という関係を表すモデルを当てはめる

y 目的変数（被説明変数、従属変数、出力変数）... モデルによって表現される値

x 説明変数（独立変数、入力変数）

一つの目的変数に対して、一つの説明変数での回帰を単回帰モデル

複数の説明変数での回帰を重回帰モデルと呼ぶ

線形回帰モデルの当てはめは `lm()` 関数で行います

```
lm_res <-
```

```
# 「アイスの売り上げ = f(気温)」のモデルを考える
```

```
lm(ice ~ temperature_average_c, data = df_ice_weather)
```



回帰直線と回帰係数

2つの変数 (x, y) の関係を説明する 回帰直線 $y = ax + b$ （一次関数）を考える

回帰係数

a 切片 (x が0のときの y の値)

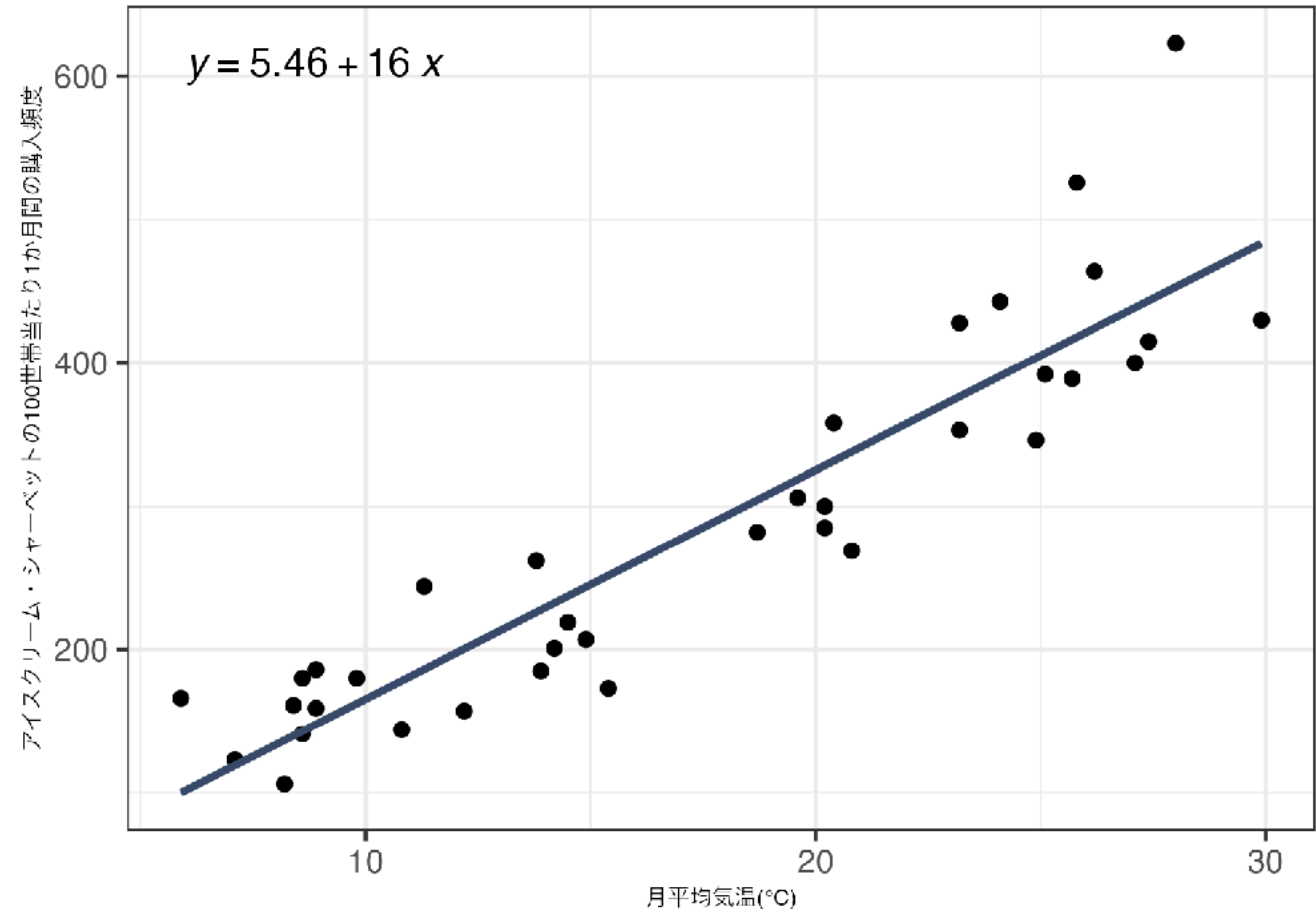
b 傾き

a b モデルの係数またはパラメータと呼ぶ

$$a = \beta_0$$

$$b = \beta_1$$

気温とアイスクリーム・シャーベットの売上の関係



回帰直線

「切片aと傾きbからなる回帰直線上のxの値によってyの値が決まる」と考える

平均気温が高くなればアイスの売り上げが上がる？

気温が1℃上がればアイスの購入頻度が16増える

回帰直線の係数と気温からアイスの売り上げを予測する

アイスの売り上げ = $5.46 + 16.0 \times 30$ (気温)

```
broom::tidy(lm_res)
#> # A tibble: 2 × 5
#>   term                estimate std.error statistic    p.value
#>   <chr>                <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)          5.46      22.8      0.239 8.12e- 1
#> 2 temperature_average_c 16.0       1.21     13.2 6.49e-15
```



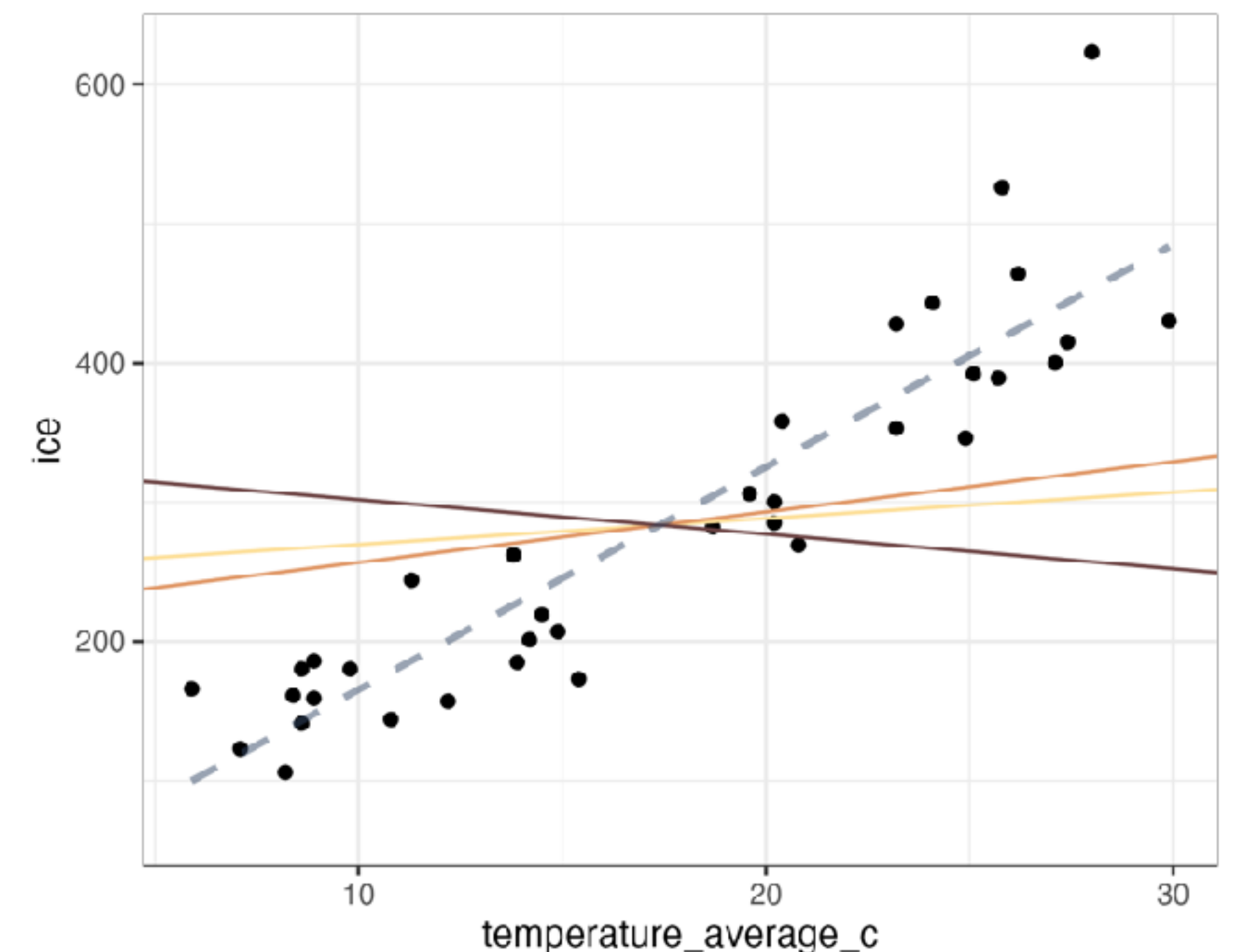
回帰係数を直接得る

```
coef(lm_res)
#>      (Intercept) temperature_average_c
#>      5.463889      15.987059
```

回帰係数

切片と傾きが異なれば回帰直線の形も異なる

2つの変数の関係を最も説明できる回帰直線とは？



さまざまな回帰係数からなる回帰直線

最小二乗法

回帰直線から残差（residual, 観測値 y から予測値 \hat{y} のズレ… 誤差）を求める

予測値 \hat{y} は回帰直線の係数から推定する

$$\hat{y}_i = 5.46 + 16x_i$$

$$x_1 = 7.1$$

$$\hat{y}_1 = 5.46 + 16 \times 7.1$$

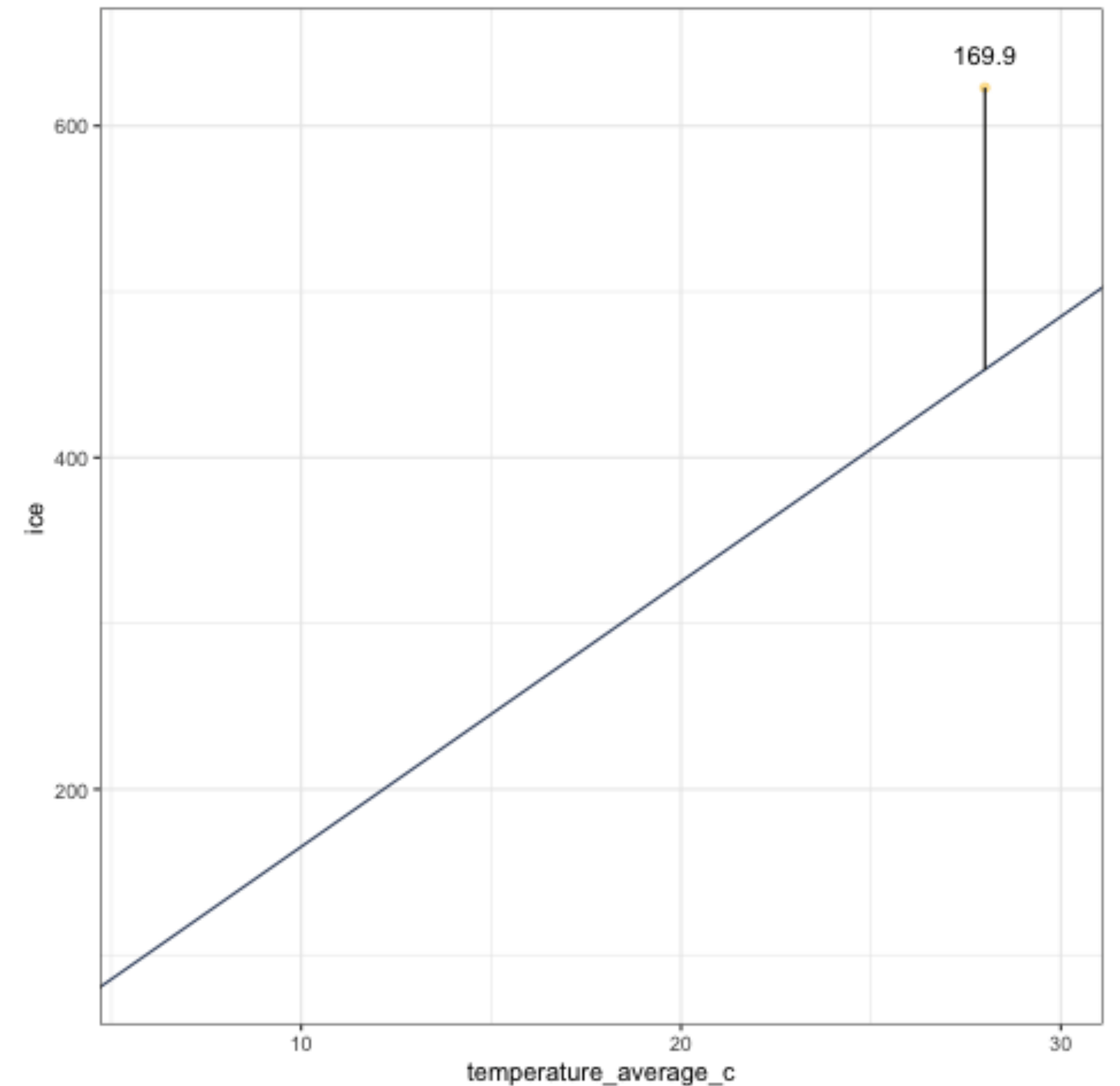
$$\hat{y}_1 = 118.9$$

$$residual_1 = y_1 - \hat{y}_i$$

$$y_1 = 123$$

$$123 - 118.9$$


残差平方和（各残差を二乗した結果を合計する）が
最小となる定数項（傾きと切片）を求める



残差平方和を最小にすることにより当てはめた回帰直線と気温。縦の棒が残差を示す。

残差、残差平方和を求める


残差



```
# y = ax+bを考えると、残差は y - (a + b * x) で求められる
df_ice_weather$ice[1] - ((coef(lm_res)[[1]] + coef(lm_res)[[2]] *
df_ice_weather$temperature_average_c[1]))
#> [1] 4.027995
# 同じ値はlm()関数の実行結果に residuals として記録される
lm_res$residuals[1]
#> 1
#> 4.027995

# すべての y に対する残差
lm_res$residuals
#> 1 2 3 4 5 6 7
#> 4.027995 -30.557770 -34.124122 -18.276239 -28.402473 51.636351 108.069999
#> ... 省略
```

残差平方和



```
# 残差平方和は残差の二乗を合計することで求められる
sum(lm_res$residuals^2)
#> [1] 93769.75
# lm()関数の結果に対してdeviance()を実行することでも計算される
deviance(lm_res)
#> [1] 93769.75
```

$$RSS = \sum (y_i - \hat{y}_i)^2$$

RSS: Residual sum of squares


回帰式に含まれる誤差

回帰式には必ず誤差 ϵ が存在する

観測値 y も x の値、モデルの係数（切片と傾き）と誤差項 ϵ により得られる

$$y = \beta_0 + \beta_1 x + \epsilon$$

```
df_ice_weather$ice[1]
#> [1] 123
unnname(coef(lm_res)[[1]] + coef(lm_res)[[2]] * df_ice_weather$temperature_average_c[1] +
lm_res$residuals[1])
#> [1] 123
```



→誤差項の存在により、真の回帰直線（切片と傾き）が得られていたとしても
 x から y を完璧に予測することはできない

決定係数 R^2

モデルの当てはまりの度合いを評価する指標

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

RSS (残差平方和) おさらい

TSS: total sum of squares (全平方和)

$$RSS = \sum (y_i - \hat{y}_i)^2$$

$$TSS = \sum (y_i - \bar{y})^2$$

```
# 1 - 残差平方和 / 全平方和
1 - (deviance(lm_res) / sum((df_ice_weather$ice - mean(df_ice_weather$ice))^2))
#> [1] 0.8362125
# lm()関数の結果に対してsummary()を適用し、r.squared で参照する
summary(lm_res)$r.squared
#> [1] 0.8362125
```



通常0~1の間の値となり、1に近いほど回帰の当てはまりが良いことを示す

lm()関数の結果を要約する



```
lm_res <-  
  lm(ice ~ temperature_average_c, data = df_ice_weather)  
summary(lm_res)  
#>  
#> Call:  
#> lm(formula = ice ~ temperature_average_c, data = df_ice_weather)  
#>  
#> Residuals:  
#>      Min       1Q   Median       3Q      Max  
#> -78.67 -34.76 -16.51  36.20 169.90  
#>  
#> Coefficients:  
#>              Estimate Std. Error t value Pr(>|t|)  
#> (Intercept)         5.464     22.840   0.239   0.812  
#> temperature_average_c 15.987      1.213  13.175 6.49e-15 ***  
#> ---  
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
#>  
#> Residual standard error: 52.52 on 34 degrees of freedom  
#> Multiple R-squared:  0.8362, Adjusted R-squared:  0.8314  
#> F-statistic: 173.6 on 1 and 34 DF,  p-value: 6.491e-15
```

回帰式

残差の四分位数

回帰係数、標準偏差、有意性検定の結果

残差標準誤差

自由度

決定係数

自由度調整済み決定係数

決定係数の分布

線形重回帰モデル

複数の説明変数により目的変数を予測するモデル

複数の説明変数を同時にモデルへ投入することで、
目的変数に対する他の影響を調整した個々の変数の影響を見る

× 説明変数が異なる単回帰モデルを複数用意する

$$y = \beta_0 + \beta_1 x + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon$$

説明変数がp個のときの重回帰モデル

回帰係数の推定は最小二乗法により行う

データへの当てはまりは、決定係数や赤池情報量基準(AIC)で評価する

線形重回帰モデル

アイスの売り上げを予測するモデルに「気温」と「降水量」の2つの説明変数を考える



```
# 「アイスの売り上げ = 気温 + 降水量」のモデルを考える
lm_res <-
  lm(ice ~ temperature_average_c + precipitation_sum_mm, data = df_ice_weather)
broom::tidy(lm_res) # 降水量よりも気温の効果が大きい、降水量の効果は有意ではない
#> # A tibble: 3 × 5
#>   term                estimate std.error statistic    p.value
#>   <chr>                <dbl>      <dbl>      <dbl>    <dbl>
#> 1 (Intercept)          5.29        23.2         0.228 8.21e- 1
#> 2 temperature_average_c 16.1         1.49        10.8 2.30e-12
#> 3 precipitation_sum_mm -0.0166       0.105        -0.158 8.75e- 1
```

単回帰モデルと同様、回帰係数が得られる … 重回帰モデルではこれを偏回帰係数と呼ぶ

ある偏回帰係数は、それ以外の説明変数を固定した際に、その説明変数が1増加するとyがどれだけ増加・減少するかを示す

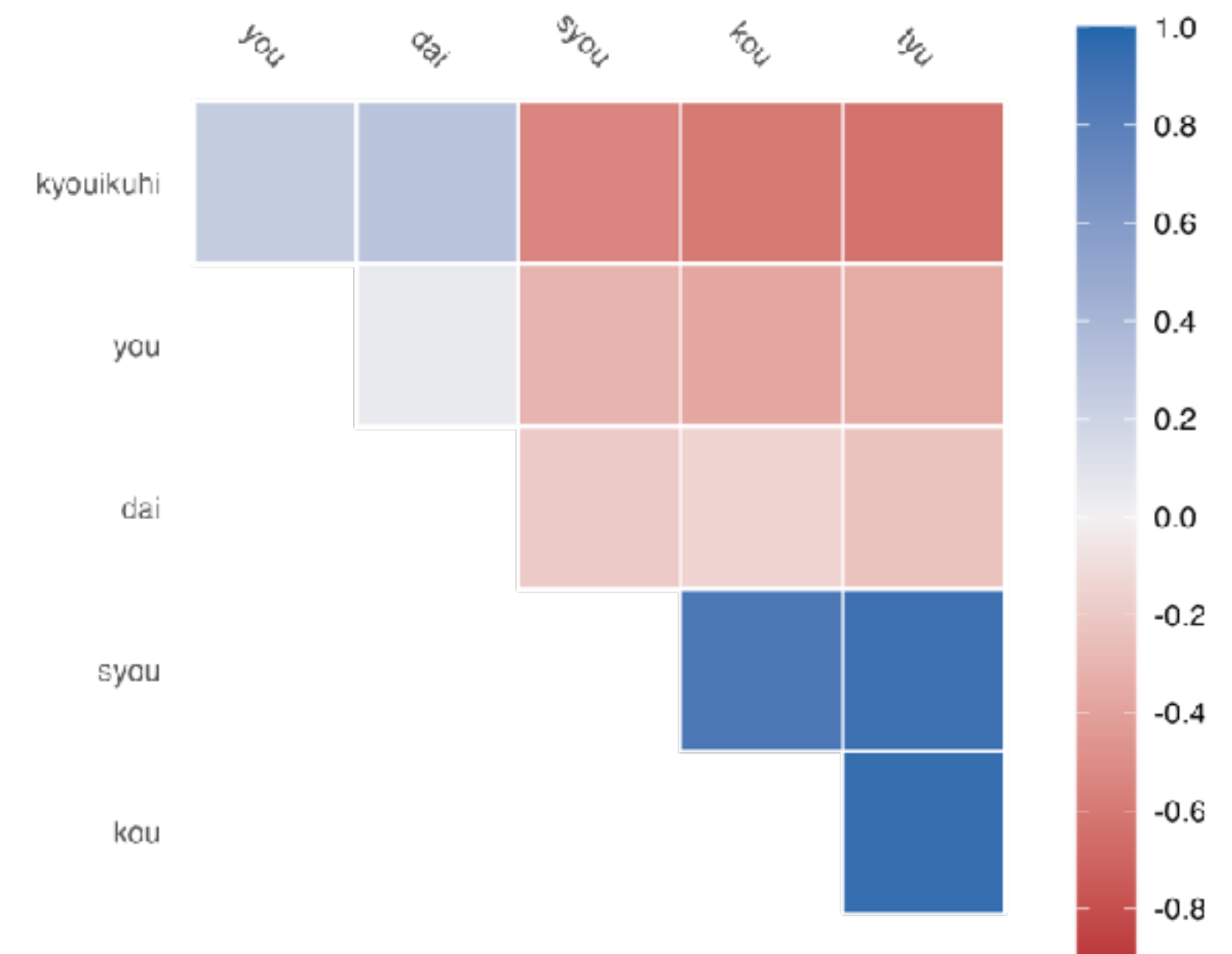
多重共線性

説明変数の間で強い相関があるものをモデルに取り入れることで引き起こされる問題

```
# 都道府県別の教育費と各学校（幼稚園から大学）の人口1万人あたり教員数
df_ssdse_b2019
#> # A tibble: 47 × 6
#>   kyouikuhi  you  syou  tyu  kou  dai
#>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1      8848  9.36  36.5  22.2  19.7  12.7
#> 2      5095  5.25  37.5  24.3  24.0  11.1
#> 3     10722  5.47  39.3  24.1  24.4  11.4
#> 4     10218 10.2   34.3  21.0  19.8  21.7
#> 5      9193  4.06  33.6  23.0  21.6  10.6
#> 6     12377  7.06  36.7  21.6  23.3   9.85
#> 7      8375 10.2   36.9  23.3  21.9   8.29
#> 8     13836  7.13  33.7  20.6  20.0  11.1
#> 9     10706  6.55  36.3  21.3  18.8  21.2
#> 10      6493  7.00  35.3  20.9  19.2   9.13
#> # ... with 37 more rows
```



```
# 中学校教員数は小学校教員数、高校教員数と強い相関関係をもつ
corrr::autoplot(
  corrr::correlate(df_ssdse_b2019))
```



多重共線性

3つまたはそれ以上の変数が共線性（予測変数間に強い相関）をもつと多重共線性を引き起こす

多重共線性が起こると推定結果も解釈もすべて誤りとなる可能性がある

推定結果が不安定… 回帰係数の値が極端に大きくなる、係数の符号の逆転



```
# Rのモデル式の指定では、 . を使うと目的変数に与えた変数以外のすべての変数を指定することになる
lm_full_res <-
  lm(kyouikuhi ~ ., data = df_ssdse_b2019)
broom::tidy(lm_full_res)
#> # A tibble: 6 × 5
#>   term          estimate std.error statistic    p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)  23397.    4774.     4.90 0.0000154
#> 2 you           51.3     233.     0.220 0.827
#> 3 syou          161.     234.     0.689 0.495
#> 4 tyu          -775.     526.    -1.47 0.148
#> 5 kou          -144.     402.    -0.358 0.722
#> 6 dai           95.1     65.5     1.45 0.154
```

多重共線性への対処

回帰から問題のある変数のうち1つを取り除く

分散拡大要因 (Variance Inflation Factor: VIF)の値が5または10を上回る変数がないか確認する

$$\text{VIF} = \frac{1}{1 - R^2}$$

```
# 中学校教員数_人口1万人あたり(tyu)が VIF 10を超える
car::vif(lm_full_res)
#>           you           syou           tyu           kou           dai
#>  1.160125  6.098338 13.911387  8.865920  1.086794

# 中学校教員数、高校教員数を除いたモデルを検討
lm_res <-
  lm(kyouikuh1 ~ you + syou + dai, data = df_ssdse_b2019)
```



多重共線性への対処

```
# 小学校教員数の係数が有意 p < 0.05 となった（フルモデルでは p = 0.495）  
# 小学校の教員数が減ると教育費は上がる？  
broom::tidy(lm_res)  
#> # A tibble: 4 × 5  
#>   term          estimate std.error statistic    p.value  
#>   <chr>          <dbl>      <dbl>      <dbl>    <dbl>  
#> 1 (Intercept)  22314.    5004.        4.46 0.0000582  
#> 2 you           170.     239.        0.711 0.481  
#> 3 syou          -381.     107.       -3.57 0.000886  
#> 4 dai           111.     67.5        1.65 0.107
```



変数の標準化と標準偏回帰係数

重回帰モデルでは、単位の異なる説明変数からなるモデルを扱う場合に
係数の比較が困難になる



```
lm_res <-  
  lm(ice ~ temperature_average_c + precipitation_sum_mm, data = df_ice_weather)  
broom::tidy(lm_res)  
#> # A tibble: 3 × 5  
#>   term                estimate std.error statistic    p.value  
#>   <chr>                <dbl>      <dbl>      <dbl>    <dbl>  
#> 1 (Intercept)          5.29        23.2         0.228 8.21e- 1  
#> 2 temperature_average_c 16.1         1.49        10.8   2.30e-12  
#> 3 precipitation_sum_mm -0.0166       0.105        -0.158 8.75e- 1
```

気温が1°Cと降水量1mmの変化を同じ物差しで評価できない

→説明変数および目的変数をそれぞれ標準化した重回帰モデルを適用する

平均0、標準偏差1に変換

変数の標準化と標準偏回帰係数

```
x <-  
  scale(df_ice_weather$precipitation_sum_mm)  
round(mean(x, na.rm = TRUE)); sd(x, na.rm = TRUE)  
#> [1] 0  
#> [1] 1
```



```
df_ice_weather_scaled <-  
  df_ice_weather |>  
  mutate(across(.cols = c(ice, precipitation_sum_mm, temperature_average_c),  
    .fns = ~ c(scale(.x))))  
  
# 標準化したデータを使って重回帰モデルを行う  
lm_res_scaled <-  
  lm(ice ~ temperature_average_c + precipitation_sum_mm, data = df_ice_weather_scaled)  
broom::tidy(lm_res_scaled)  
#> # A tibble: 3 × 5  
#>   term                estimate std.error statistic    p.value  
#>   <chr>                <dbl>      <dbl>      <dbl>    <dbl>  
#> 1 (Intercept)        -4.66e-16    0.0694  -6.71e-15  1.00e+ 0  
#> 2 temperature_average_c  9.22e- 1    0.0854   1.08e+ 1  2.30e-12  
#> 3 precipitation_sum_mm  -1.35e- 2    0.0854  -1.58e- 1  8.75e- 1
```



標準化によりすべての係数が同じ物差しとして評価できる（標準偏回帰係数）

分類モデル

分類: 質的な目的変数を予測する

分類問題の例

メールボックスが受信するメールがスパムであるか、そうでないものか

過去の授業出席状況や小テストの結果から、ある学生が試験に合格するかどうか

さまざまな部位の計測により製品の状態を3段階で評価する際、ある製品はどの段階に割り当てられるか

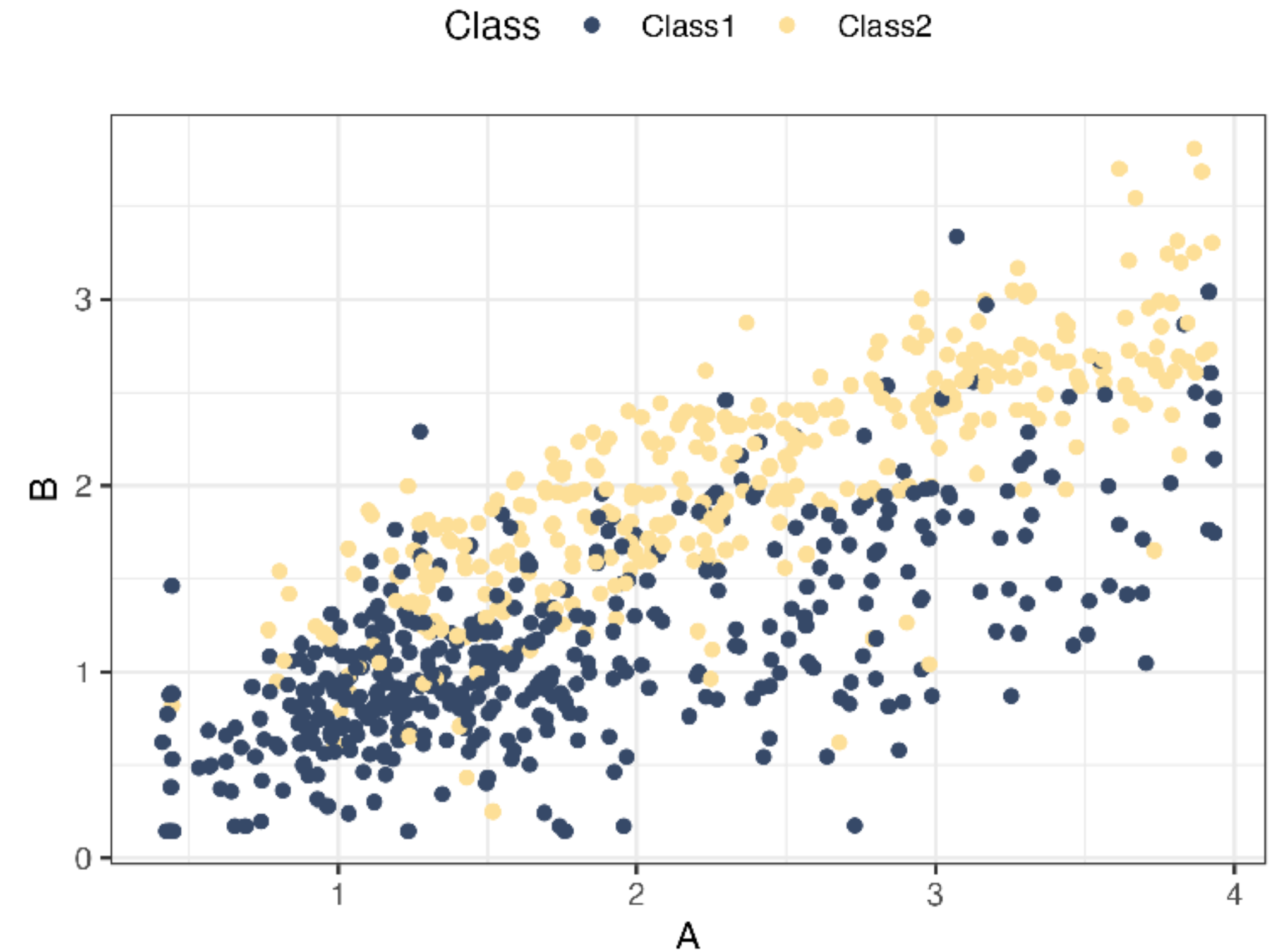
質的な目的変数を線形回帰モデルで扱うのは必ずしも適切ではない

データ分割

モデルの学習・評価のためにデータを分割する



```
library(tidymodels)
data("two_class_dat", package = "modeldata")
two_class_split <-
  initial_split(two_class_dat, strata = "Class")
two_class_split
#> <Training/Testing/Total>
#> <592/199/791>
# 学習データ
train_two_class <-
  training(two_class_split)
# 評価データ
test_two_class <-
  testing(two_class_split)
```



ロジスティック回帰

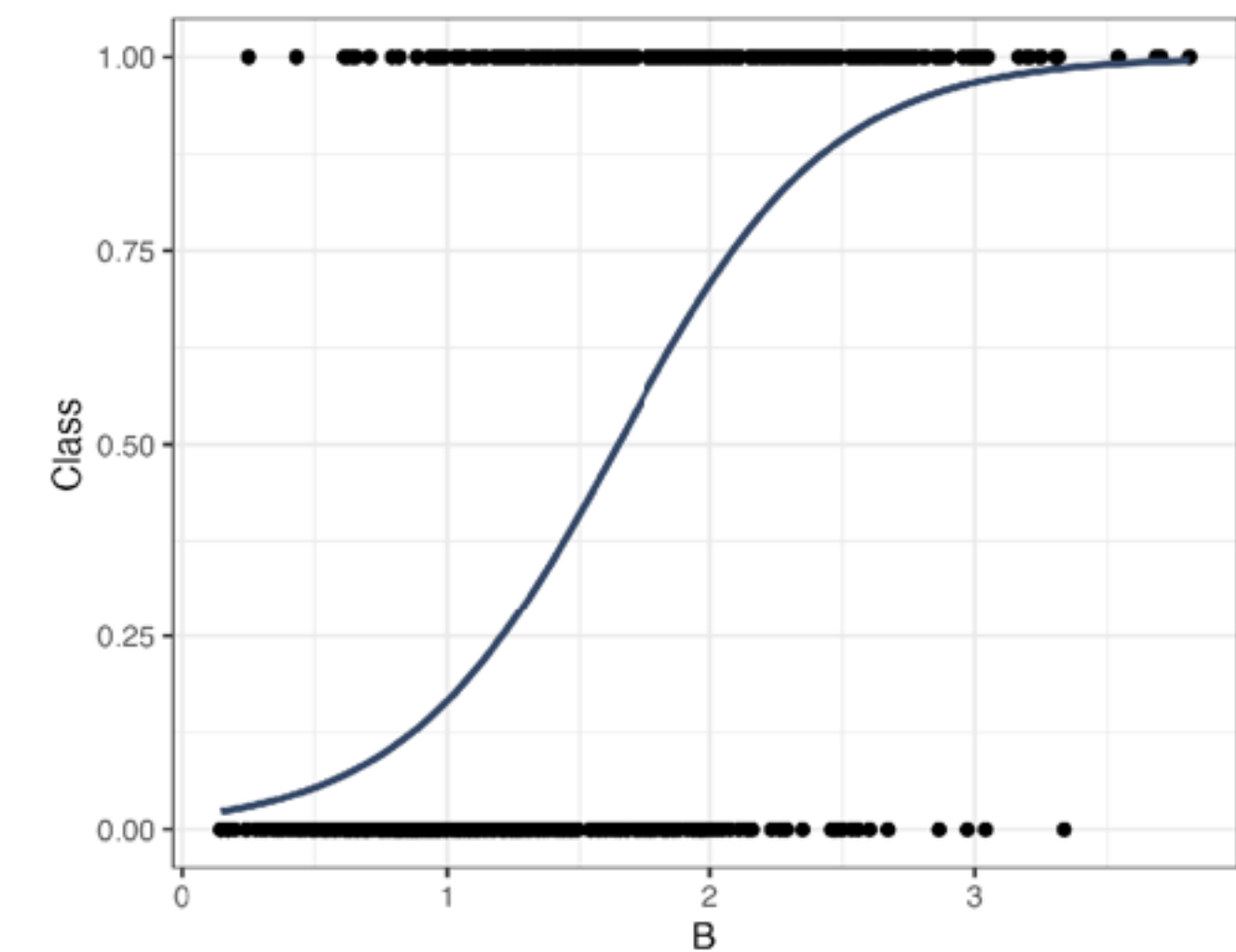
2値データを目的変数とする分析

目的変数を直接モデル化せず、Yが特定のカテゴリに属する確率をモデル化する

目的変数の値が二項分布（コインの裏表のような2つの値を取る）に従うと仮定

ロジット関数による変換を行い、回帰係数を最尤法で解くことで推定する

```
glm(Class ~ A + B, data = train_two_class, family = binomial) |>
  broom::tidy()
#> # A tibble: 3 × 5
#>   term          estimate std.error statistic    p.value
#>   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
#> 1 (Intercept)   -3.82     0.343    -11.1 8.30e-29
#> 2 A             -1.22     0.217     -5.61 2.00e- 8
#> 3 B              3.83     0.337     11.4 5.35e-30
```

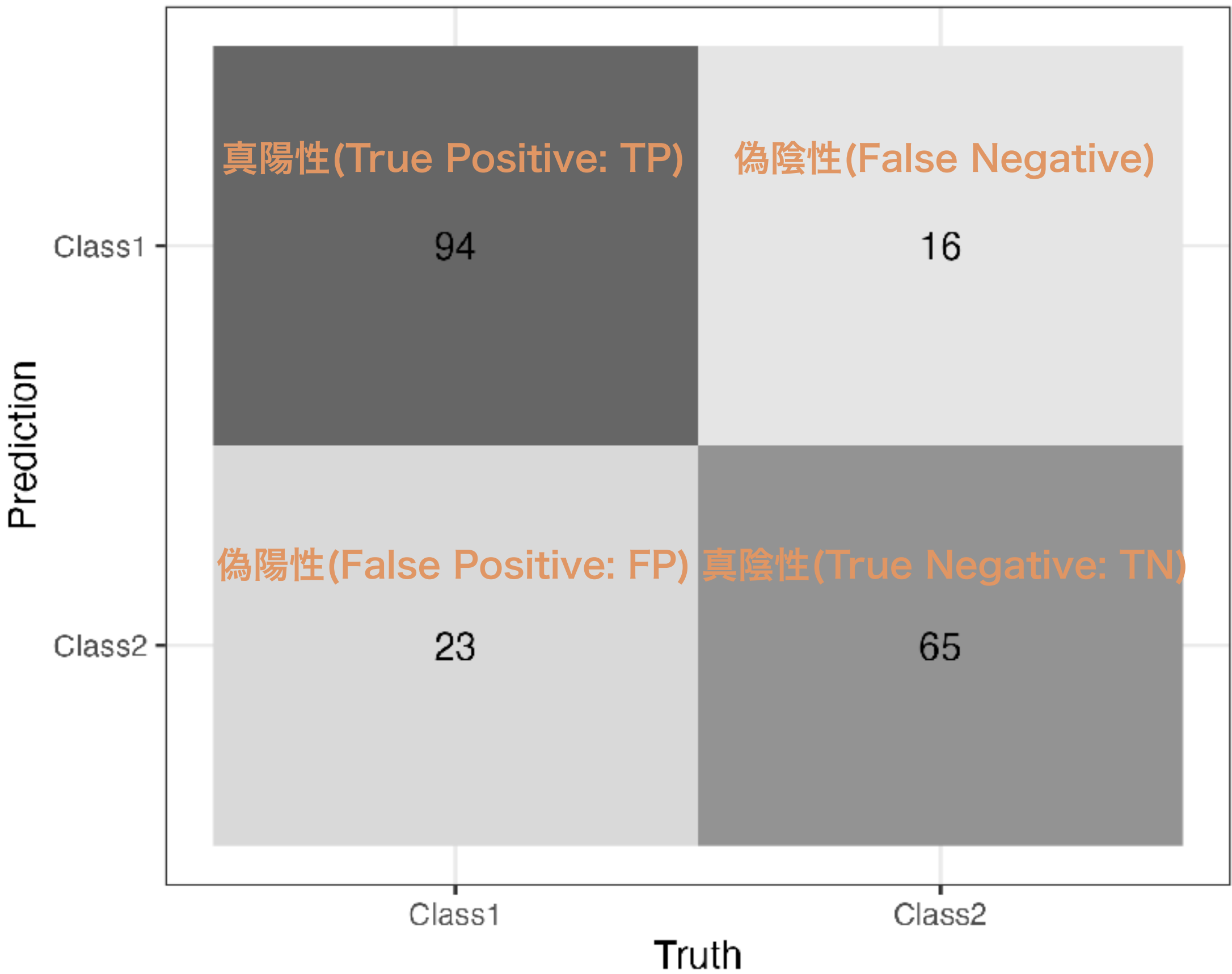


分類モデルの評価

学習したモデルが評価データに対してどの程度の精度を発揮できるか

混同行列… 分類器の4つの予測の個数を報告する正方行列

真と偽は「予測が当たったかどうか」を、陽性と陰性は「予測されたクラス」を表す



分類モデルの評価

分類モデルの場合、**正解率 (accuracy)**や**適合率 (precision)**、**再現率 (recall)**などの評価指標を利用する

正解率… 予測した値（正や負）が実際の値と一致する割合

適合率… 正と予測したデータのうち、実際に正であるものの割合

再現率… 実際に正であるもののうち、正であると予測されたものの割合

```
lr_spec <-  
  logistic_reg(mode = "classification",  
                 engine = "glm")  
lr_fitted <-  
  lr_spec |>  
  fit(Class ~ A + B, data = train_two_class)
```



```
multi_metric <-  
  metric_set(accuracy, precision, recall)  
augment(lr_fitted, new_data = test_two_class) |>  
  multi_metric(truth = Class, estimate = .pred_class)  
#> # A tibble: 3 × 3  
#>   .metric      .estimator .estimate  
#>   <chr>      <chr>      <dbl>  
#> 1 accuracy  binary      0.794  
#> 2 precision binary      0.8  
#> 3 recall   binary      0.836
```



モデルの性能評価のためには交差検証法を用いて検証用のデータを用意することが一般的

参考文献・URL

G.James, D.Witten, T.Hastie, R.Tibshirani 著 落海浩、首藤信通 訳 (2018).
Rによる統計的学習入門 (朝倉書店) ISBN: 978-4-254-12224-4

有賀友紀、大橋俊介 (2021). RとPythonで学ぶ実践的データサイエンス&機械学習 増補改訂版 (技術評論社)
ISBN: 978-4-297-12022-1

岡崎直観 「機会学習帳」 <https://chokkan.github.io/mlnote/>