統計学第 3 講/第 4 講

後藤　晶　　akiragoto@meiji.ac.jp

目次

目次

前回の復習

統計学やデータサイエンスを学ぶ意義を考える

統計学とは？

- データサイエンス：データから有用な情報・知識を引き出したり，新たな価値あるデータを創造するための基本な考え方
  - 平均・分散を算出したり，全体的な傾向を把握するためにヒストグラムを作るのは有用な情報・知識を引き出すため．
  - プログラムを組んで，新たに価値あるものを作る．
- 統計学を学ぶと何がどうなる？
  -「データ」に基づいた思考方法が身につく
  -「見せかけの類似性」に騙されなくなる．
  - 日常生活・ビジネスへの応用可能性が広がる

日常生活・ビジネスへの応用可能性が広がる

- ビジネスにおいて，データを利用し，企業活動を改善・開拓するのに必要な 3 要素
- データを効率的に収集・処理する
  - 企業活動におけるデータは莫大であり，このようなデータを処理するためには情報技術の利用が必要
- データを適切に取り扱い，妥当かつ汎用的な成果を残す
  - データ全体から，その傾向の妥当性を検討する．
- データをビジネスの枠組みの中にうまく組み込む
  - 効率的に処理をし，有用な結論を導き出してビジネスへの応用をしていく．

RStudio を使うための準備

- RStudio で使うディレクトリを決める
  - 初回のみ準備が必要
- ".R" ファイルを作成する
  - 毎回準備が必要
- 毎回「USB メモリ」を持ってくるようにしてください．

RStudio で使うディレクトリを決める

- 最初に RStudio で使うディレクトリ（フォルダ）を決めます．この時，【フォルダまでの間に 2 バイト文字（日本語）やスペースが入らないように気をつけて下さい】．
  - 使えない文字：あいうえおかきくけこ「」□ 今日の天気は晴れ！１２３４５６７８９０
  - 使える文字：abcdefghijkelmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890
- また，windows でも mac でもこの作業は変わりません．

".Rproj" ファイルを作成する

- 毎回，授業の際に R で作業をする際には，指定した作業用フォルダにある ".Rproj" ファイルから開きます．
- 作業フォルダを決める（必要に応じてフォルダを作成する）
- RStudio を起動する．
- 右上の現在いるフォルダを示している文字をクリックする．
  - 場合によっては "(None)" と表示されているかもしれません．
- "New Project..." をクリックする．
- "Existing Directory" をクリックする．
- "Browse..." をクリックして，先程決めた作業フォルダまでたどり着く
- "Open" をクリックする．
- "Create Project" をクリックする．
- 完了

".R" ファイルを作成する

- 授業中に作成したファイルを保存するために，".R" ファイルを作成する．
  - これは毎回行います．
- 左上にある白い四角の左上に緑のプラスが書いてあるヤツをクリックする．
- "R Script" をクリックする．
- "Untitled 1" と書かれたファイルができる．
- "ctrl + s" を押すと保存ができる．

  - Mac で作業している場合は “cmd + s”

- ファイル名として “今日の日付 _ 学籍番号” を設定する．
  - ex.“20201013_ 学籍番号.R” とする
  - 学籍番号を入れてもらうのは，後ほど提出してもらうため．
- “Save” をクリックする．
- また，このファイルに書いた数式は “ctrl + Enter”（Mac の場合は “cmd + Enter”）でその行の計算を R に読み込ませることができます．

実際に計算をしてみよう

加減乗除

```
123 + 123   # 足し算
```

```
## [1] 246
```

```
123 - 123    # 引き算
```

```
## [1] 0
```

```
23 * 123     # 掛け算
```

```
## [1] 2829
```

```
123 / 123    # 割り算
```

```
## [1] 1
```

```
123 ^ 2   # 累乗
```

```
## [1] 15129
```

その他の計算

```
sqrt(144)# 平方根
```

```
## [1] 12
```

```
1234 %/% 123   #整数商：割り算をした時の整数部分
```

```
## [1] 10
```

```r
1234 %% 123   # 剰余：割り算のあまり
```

```
## [1] 4
```

記述統計量とは

平均値・分散・標準偏差とは？

- 平均値：全てのデータを足して割ったもの．一般的に代表値（データ全体を表している数値）として扱われる．
- 分散：平均値とそれぞれの値の差を求めて 2 乗して，合計したものをデータの個数で割ったもの．データの散らばり具合を示す数値であり，分散が大きければ大きいほど，データが散らばっていることを示す．
  - σ^2 という記号で表される．
  - $(\ ) = \Sigma\{(\ ) - (\ \ )\}^2/(\ )$
- 標準偏差：分散の平方根．通常の長さのばらつきを評価する際には同じ単位で理解したほうがわかりやすいために用いる．
  - σ という記号で表される．

その他，重要な指標

- 最小値：そのデータの中で最も小さい値
- 第一四分位数（25% パーセンタイル値）：最小値と中央値の間の中央値
- 中央値（第二四分位数）：データを大きい (小さい) 順に並べたとき，真ん中の値のこと (median)．外れ値がある時に代表値として用いられる．
  - 奇数の場合：ちょうど真ん中が存在する．
  - 偶数の場合：真ん中の数字 2 つの平均値を中央値とする．
- 最頻値：データの中で最も多く出てくる値のこと (mode)．因子データの際に代表値として使われる．
- 第三四分位数（75% パーセンタイル値）：中央値と最大値の間の中央値
- 最大値：そのデータの中で最も大きい数
- 以下の 2 つは参考までに．
  - 平均偏差：「平均からの偏差」の絶対値の平均
  - 範囲：最大値から最小値の間．引き算で求められる．

平均値の計算

- 7 人の学生の体重が 50, 60, 85, 70, 80, 67, 66kg であったする．これらの学生の体重の平均値を求めよ．
- 平均値：全てのデータを足して割ったもの．一般的に代表値（データ全体を表している数値）として扱われる．

平均値の計算

- 7 人の学生の体重が 50, 60, 85, 70, 80, 67, 66kg であったする．これらの学生の体重の平均値を求めよ．

```
# 平均値 =(それぞれのデータの値の合計)/(データの個数)
(50+60+85+70+80+67+66)/7
```

```
## [1] 68.28571
```

オブジェクト指向

- 「オブジェクト」: データやモデル式などを入れる「何でも箱」
  - R ではモデル式，データなどをオブジェクトに入れて考える
  - 数式やデータをいちいち書くのは大変...
  - オブジェクトに入れることを「代入する」と言う

オブジェクトに入れて計算する：

```
x <- 5 #x というオブジェクトに 5 を代入する
x #x の値を出力する
```

```
## [1] 5
```

```
(y <- 3) #( ) に挟むと，一発で結果も出してくれる．
```

```
## [1] 3
```

```
x + y
```

```
## [1] 8
```

```
x * y
```

```
## [1] 15
```

```
x - y
```

```
## [1] 2
```

データセットを作ろう

7 人の学生の体重が 50, 60, 85, 70, 80, 67, 66kg であったする．このデータを変数名 “weight” に代入する．

```r
weight<-c(50, 60, 85, 70, 80, 67, 66)
```

演習問題 3：
同じ 7 人の学生の身長が 155, 164, 182, 165, 177, 177, 172cm であったとする．このデータを変数名 “height” に代入せよ．

R における「関数」とは？

- 関数：頻繁に用いられるデータ操作方法や，標準的な統計計算をまとめてオブジェクトにしたもの．正式には「関数オブジェクト」
  - 簡単に計算できるように，先人たちがまとめたものだと理解すれば良い．
  - これを用いることで，簡単に計算ができる．

記述統計量を色々出してみる．

```r
sum(weight)/7  #sum() 関数で合計を算出できる．
```

```
## [1] 68.28571
```

```r
sum(weight)/length(weight)  #length() 関数でデータの個数を数える．
```

```
## [1] 68.28571
```

```r
mean(weight)  # 実は mean() という関数を使うと一発で出てしまう．
```

```
## [1] 68.28571
```

```r
median(weight)  # 中央値は median() という関数で出せる．
```

```
## [1] 67
```

```r
table(weight)  # 最頻値は table() という関数を使って探し出す．
```

```
## weight
## 50 60 66 67 70 80 85
##  1  1  1  1  1  1  1
```

```r
# ちなみに，"weight" の中に最頻値は存在していない．(全てが最頻値 =1)
```

演習問題 4：

変数名 “height” の合計・個数・平均値・中央値・最頻値を求めよ．

体重の記述統計量をまとめて算出する.

```
summary(weight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   50.00   63.00   67.00   68.29   75.00   85.00
```

- 左から順番に「最小値，第 1 四分位数，中央値，平均値，第 3 四分位数，最大値」を示しています.

演習問題 5：

変数名 “height” の最小値・第 1 四分位数・中央値・平均値・第 3 四分位数・最大値を求めよ.

第 3 講：データの可視化

今回の目標：

- RStudio の準備をしよう.
- R を使った基本的な計算方法に慣れよう.
- 分散・標準偏差を手計算で算出してみよう.

分散と標準偏差を手計算で算出してみよう

分散を算出する

- 定義通りの算出方法

$$\sigma^2 = \Sigma((\ ) - (\quad))^2/(\ \ )$$

- 簡便に算出する際に用いられる数式

$$\sigma^2 = \Sigma((\ )^2/(\ \ )) - (\Sigma((\ )/(\ \ ))^2)$$

* 「2 乗の平均」-「平均の 2 乗」

- ここでは，定義通りの数式から分母と分子に分けて話を進めていきましょう.

分子を計算する

体重の平均値をオブジェクトに入れる

- “mean_weight” というオブジェクトを作って，体重の平均値を入れます.

```
mean_weight <- mean(weight)
mean_weight
```

```
## [1] 68.28571
```

平均からの偏差を求めて，オブジェクトに入れる

*(データの値)-(weight の平均値) をして「平均からの偏差」を求めます．結果は “hensa_weight” に代入します．

```
hensa_weight <- weight - mean_weight
hensa_weight
```

```
## [1] -18.285714  -8.285714  16.714286   1.714286  11.714286  -1.285714  -2.285714
```

「平均からの偏差」を 2 乗する

*「平均からの偏差」を 2 乗します．“hensa_weight2” というオブジェクトを作って代入をしましょう．- 2 乗しないとで全部足すと，数字は 0 になります．- ただし，小数点以下を四捨五入しているので，ここでは完璧に 0 にはなりませんが，限りなく 0 に近くなります．

```
hensa_weight2 <- hensa_weight^2
hensa_weight2
```

```
## [1] 334.367347  68.653061 279.367347   2.938776 137.224490   1.653061   5.224490
```

「平均からの偏差の 2 乗」を全部足してオブジェクトに入れる

- これらの 5 つの値を合計した「平均からの偏差の二乗和」を求めます．
  – “sum_hensa_weight2” という名前にしましょう．これで分子は完成です．

```
sum_hensa_weight2<-sum(hensa_weight2)
sum_hensa_weight2
```

```
## [1] 829.4286
```

分母を計算する

データの個数を数えてオブジェクトに入れる

- 今度は分母を算出します．分母はデータ数です，“length_weight” というオブジェクトに代入しましょう．

```
length_weight<-length(weight)
length_weight
```

```
## [1] 7
```

最後の計算

分散の算出

- 分散は「平均からの偏差の二乗和」/「データ数」ですから，以下の通りに求められます．
  - 分散は “vari_weight” というオブジェクトに入れましょう

```
vari_weight<-sum_hensa_weight2/length_weight
vari_weight
```

```
## [1] 118.4898
```

標準偏差を算出する

標準偏差を算出する

$$\sigma^2 = \Sigma(( )^2/(\ )) - (\Sigma(( )/(\ ))^2)$$

$$\sigma = \sqrt{\sigma^2}$$

- 標準偏差は分散の平方根です．
  - 平方根を求める関数は “sqrt()” であり，“hyohen_weight” というオブジェクトに入れてあげます．

ルートをとります

```
hyohen_weight <- sqrt(vari_weight)
hyohen_weight
```

```
## [1] 10.8853
```

分散の別解

$$\sigma^2 = \Sigma(( )^2/(\ )) - (\Sigma(( )/(\ ))^2)$$

```r
sum(weight^2/length(weight))-sum(mean(weight)^2)
```

```
## [1] 118.4898
```

演習問題 1：

- “weight” と同様に，変数名 “height” について分散・標準偏差を計算してください．
  - ヒント：ちょっと書き換えるだけですぐいけます．
  - “ctrl + f”(mac では “cmd + f”) で置換ができます．
  - 基本方針は「極力手抜きをしましょう」です．

関数を使って分散と標準偏差を算出する．

- 分散と標準偏差はよく算出します．当然 R では関数が用意されています．

```r
var(weight) # 分散
```

```
## [1] 138.2381
```

```r
sd(weight) # 標準偏差
```

```
## [1] 11.75747
```

- 関数で求められるのは「不偏分散・不偏標準偏差」
- 手計算で求めたのは「標本分散・標本標準偏差」

2 種類の分散と標準偏差

2 種類の分散と標準偏差 ??

「不偏分散・不偏標準偏差」と「標本分散・標本標準偏差」というものが出てきました．この話を理解するためには「母集団」と「標本」という話を理解する必要があります．ここでは簡単に，その 2 つの違いについてお話したいと思います．

私達が何かのデータを取る時は，全ての物事のデータを集めることが必ずしもできるとは限りません．例えば，「本学大学生 1 年生全員を対象としたアンケート」を実施すれば全てのデータを集めることができるかもしれませんが，「日本国民全てを対象としたアンケート」を集計するのは非常に困難です．

例えば，大学 1 年生の意見を調査することを目的として，1 年生全員のデータをそのまま用いる分には問題ないのですが，「日本国民全てを対象としたアンケート」を実施するのはコストの面から考えても現実的ではありません．そのために，全体（母集団）の中から一部を取り出して（標本，サンプル），全体の意見・傾向を「推定」するという手法がとられるようになりました．

このような「推定」という手法を取る時に，“データ数” のままで分析するよりも “データ数-1” で計算してあげたほうがよりよい推定ができる，ということで “データ数-1” をするようになりました．

本当はもう少し細かな数学的な議論があるのですが，入り込むと帰って来れなくなるのでここまでにしておこうと思います．とりあえず，これからは「不偏分散・不偏標準偏差」が使われることが多い，とだけ覚えておいて下さい．

興味のある方はコチラをご参照ください．

分散が一致することを確認する．

```
var(weight) # 分散
```

```
## [1] 138.2381
```

```
huhen_vari_weight<-sum_hensa_weight2/(length_weight-1)
huhen_vari_weight
```

```
## [1] 138.2381
```

```
sd(weight) # 標準偏差
```

```
## [1] 11.75747
```

```
huhen_hyohen_weight<-sqrt(huhen_vari_weight)
huhen_hyohen_weight
```

```
## [1] 11.75747
```

演習問題 2：

以下のデータ 3 つ数値の平均値、偏差平方和、分散，標準偏差を求めてください．なお，分散および標準偏差については不偏分散・不偏標準偏差でかまわない．

- 「2, 3, 3, 4」なお，これらのデータは「data1」というオブジェクトに入れて算出すること
- 「1, 1, 1, 9」なお，これらのデータは「data2」というオブジェクトに入れて算出すること
- 「43.6, 45.2, 45.4, 45.8, 47.2, 47.8, 48.2, 48.7, 48.8, 48.9, 49.0, 49.0, 49.4」なお，これらのデータは「data3」というオブジェクトに入れて算出すること

第 4 講：実証分析の手続き

関数とパッケージ

R においてよく使う計算式は**関数**として用意されています．必要な関数は**パッケージ**をインストールすることで，適宜追加することができます．

- パッケージ：機能を拡張するもの．
  - 研究者など，たくさんの開発者が自身の研究上・仕事上のニーズに応じて拡張パッケージを用意している．
  - これを使えば様々な分析や操作が便利になる．
  - 必要に応じて，様々なパッケージを追加していくイメージ

以下の段取りを踏むことで利用可能となる．

1. パッケージをインストールする．
2. パッケージを読み込む．

1. については，一度行うだけで良い．
2 については必要に応じて適宜実施する．

以下にはその一例を示す．

```
install.packages(“dplyr”, dependencies = TRUE)
```
パッケージをインストールする．一度実行するだけで良い．

```
library(dplyr)
```
パッケージを読み込む，使うときには必ず入力する

他のデータを読み込む

今は皆さんに手入力でデータを打ち込んで貰いました．今度は，皆さんには “csv ファイル” からデータを読み込んでもらおうと思います．R の標準のデータ形式以外の他の形式のファイルを読み込むことを「インポート」と言います．

RStudio を使ってもらうと，次の手順でデータを読み込むことができます．

- “Import Dataset” をクリックする．
  - “From Text (readr)…” をクリックする．
  - 何かをインストールするように案内されたら，素直にインストールする．
- “Browse” をクリックする
  - 読み込みたいデータを選んで “Open” をクリックする．

  - データに併せて，クリックしていく．
    * 今回の場合は “First Row as Names” にチェックを入れる．これは 1 行目が各行のデータ名を示しているためである．
- “Import” をクリックしてデータを読み込む．
- 完了

下のコンソールには 3 つのコードが書かれます．1 番目のコードは “readr” というパッケージを使うように，という指示をしています．2 番目のコードは “データを読み込んで，こんな名前にしておいて下さい” を示しており，3 番目のコードは “読み込んだデータを表示して下さい” を示している．

なお，このコード（特に上の 2 つ）は “>” を取り除いて上の “.R” ファイルに保存しておくと，次回以降便利です．

```
library(readr)
# パッケージreadrを使う
dataset <- read_csv("~/hogehoge/dataset.csv")
# datasetを読み込む
```

- R のコード内で “#” と書くとコメントアウト（コードとして扱わず，メモとして使える）

なお，この “hogehoge” は読み込んだデータを保存した場所を示しており，人によって異なるので注意してください．

データのダウンロード

- 「データの説明」ページにある「こちらからダウンロードしてください」というところから，csv ファイルをダウンロードしてください．

注意：この授業で取り扱うデータについて

このデータはゴトウが実施した 1926 人分のデータのうち，ランダムに選んだ 963 人分のデータです．まだ，データの中身は「データの概要」に記載してあるので，そちらを参考にしてください．

読み込んだデータの記述統計量を算出します．ここでは人々の主観的幸福度について記述統計量を算出します．

主観的幸福度とは：

主観的幸福度とは人が感じている幸福度を示したものです．ここでは「現在，あなたはどの程度幸せですか？「とても幸せ」を 10 点，「とても不幸せ」を 0 点とすると，何点くらいになると思いますか？」として尋ねたものです．

それでは，記述統計量を出してみましょう．特に，複数列あるデータの場合は $ を使って，「データセットの中のこのデータ列について平均値を出して下さい」というように指定してあげます．

- データはコチラからダウンロードできます.

平均・分散・標準偏差・度数など.

```
library(readr)
exdataset <- read_csv("../data/exdataset.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   F_SEX = col_character(),
##   F_GEN_2 = col_character(),
##   F_GEN = col_character(),
##   F_FGR = col_character(),
##   F_INK = col_character(),
##   F_INS = col_character(),
##   F_TAN = col_character(),
##   ARE = col_character(),
##   MAR = col_character(),
##   CHI = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
library(ggplot2)
```

```
# データを読み込めたら，以下をコピペしてください.
# いろいろなものの順番を理解しやすいように並べ替えます.

## Reordering exdataset$ARE
exdataset$ARE <- factor(exdataset$ARE, levels=c("Kanto", "Hokkaido", "Tohoku", "Chubu", "Kinki", "Ch

## Reordering exdataset$MAR
exdataset$MAR <- factor(exdataset$MAR, levels=c("NotMarried", "Married"))

## Reordering exdataset$CHI
exdataset$CHI <- factor(exdataset$CHI, levels=c("NoChild", "Child"))
```

記述統計量をコードで算出する

平均値を算出してみる.
主観的幸福度 (SUB_HAP) の平均値

```
mean(exdataset$SUB_HAP)
```

```
## [1] 6.002077
```

分散を算出してみる.
主観的幸福度 (SUB_HAP) の分散

```
var(exdataset$SUB_HAP)
```

```
## [1] 5.503114
```

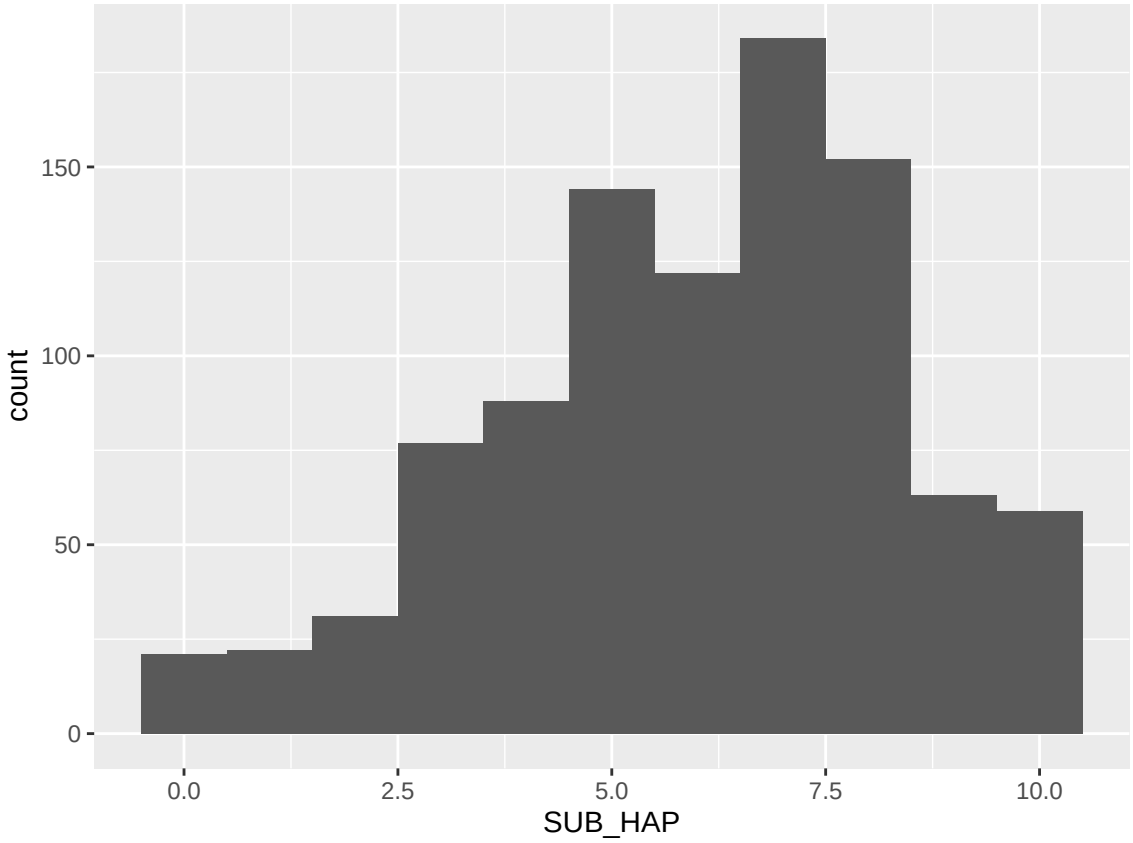
標準偏差を算出してみる.
主観的幸福度 (SUB_HAP) の標準偏差

```
sd(exdataset$SUB_HAP)
```

```
## [1] 2.345872
```

主観的幸福度 (SUB_HAP) のヒストグラム

```
g <- ggplot(exdataset, aes(x = SUB_HAP)) + geom_histogram(binwidth = 1.0)
g
```



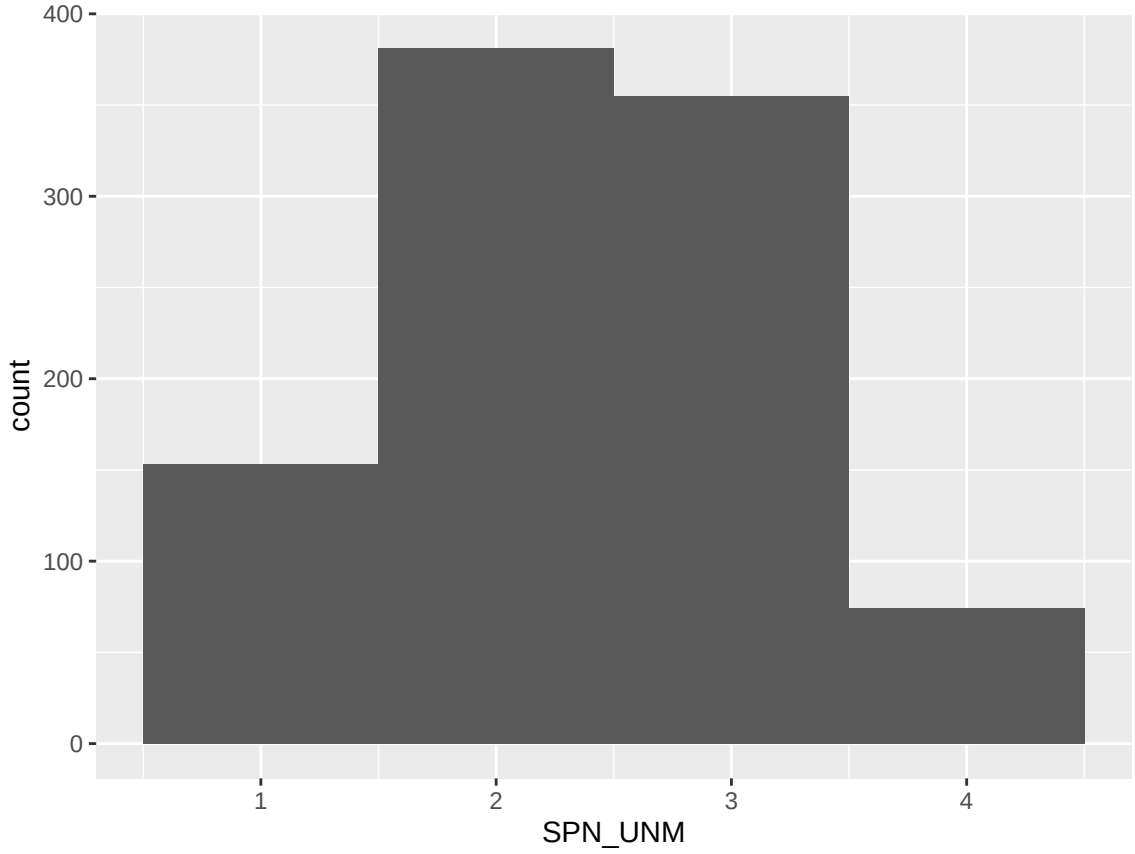
運命 (SPN_UNM) の頻度を数えてみる.

```
table(exdataset$SPN_UNM)
```

```
##
##   1   2   3   4
## 153 381 355  74
```

ついで運命 (SPN_UNM) のにヒストグラムも作ってみよう

```
g <- ggplot(exdataset, aes(x = SPN_UNM)) + geom_histogram(binwidth = 1.0)
g
```



世代の頻度を数えてみる.

```
table(exdataset$F_GEN)
```

```
##
## 10dai 20dai 30dai 40dai 50dai 60dai 70dai
##     8   140   361   358    77    18     1
```

- 次回は「クリック」だけでグラフを作成する方法を紹介します.

```
install.packages("ggplotgui")
# 初回だけ必要，ggplotguiをインストールする
library(ggplotgui)
# パッケージggplotguiを使う
```

再現可能性 (Reproducibility) の重要性

再現可能性に関する様々な議論と定義（Kulkarni, 2017）

Goodman による定義（Goodman et.al, 2016）:

- **方法の再現可能性** (Methods reproducibility)：反復可能性にもっとも近い．研究方法とデータに関する十分な情報が提供され、同じ手順を反復できるようになっていることを意味する．
- 結果の再現可能性 (Results reproducibility)：「方法の再現可能性」と密接に関連している．「元の実験と可能な限り同じ手順で，独立した実験を実施し，同じ結果を得ること」を意味する．
- 推論の再現可能性 (Inferential reproducibility)：先の 2 つの再現可能性とは異なる。別の研究から同じ推論が導かれることもあれば，同じデータから別の結果が推測されることもある．このため，推論の再現可能性とは「独立した再現実験もしくは元の研究の再分析から，質的に類似した結果を導くこと」を意味する．

Stodden による定義（Stodden, 2014）:

- 実証的再現可能性 (Empirical reproducibility)：物理的に実験を繰り返して実証する必要なすべての情報が提供されていることを意味する．この定義は，グッドマン氏の「方法の再現可能性」の定義に近い．
- **計算／統計的再現可能性** (Computational and statistical reproducibility)：研究における計算結果や分析結果を再び行うために欠かせないリソースが提供されていることを意味する．

Baker による定義（Baker, 2016）:

- **分析的反復** (Analytic replication)：単に元データを再分析して結果を再現すること．
- 直接的反復 (Direct replication)：元の実験と同じ条件，材料，方法を利用しようとすること．
- 体系的反復 (Systematic replication)：異なる実験条件で結果を再現しようとすること．（例えば，異なる細胞株やマウス株で実験を行うことなど．）
- 概念的反復 (Conceptual replication)：ある概念の一般的な正当性を示そうとすること．異なる有機体を使用する場合も含まれる．

再現可能なデータ分析とレポート作成のメリット（高橋, 2018）

信頼性の向上

- データ解析：得たデータを分析結果やグラフに変換すること
- 同じデータからいつでもどこでも誰でも同じ結果を得られる必要がある
  - 皆さんの分析結果がゴトウが授業でやっている結果と一致しなかったら不安になりませんか？
- 分析が再現できることは，その研究の信頼性が高いことを示している．

- 統計処理はあくまでも「プロセス」なので，決まった形式が存在している．同じ分析結果を出力するための技術は身につける必要がある．

間違いの検証

- 人間の作業には何らかの間違いが発生しがち．
- 特に，分析過程でコードのどこかに間違いが存在することがある．
- 再現可能なデータ分析を行うことで，間違いを探すことができる．
- **間違いは罪ではないが，「どこで間違ったかわからなくする」のは罪である．**
  - 間違ったことを責めるのではなく，どこに原因があるのかを探す & 見つけられることが重要．

作業効率の向上

- 作業の大半を自動化できており，作業時間を減少することができる．
- 間違いの検証にかかる時間も大幅に減少することが可能となる．
- 本当は R を使うと同じコードを使いまわしできる．
  - 必要に応じて過去に使ったコードを使う必要がある．

作業を進める際には以下のことを気をつけましょう．

- データソースを手で加工，整形していないか
  - どんなことをやったかわからなくなりがちなので，極力データは RStudio の上で加工するようにしましょう．
  - とはいえ，最初はいきなりこれも厳しいか．
- コピペを行っていないか
  - R のコードを R スクリプトにコピペする作業は除く
- コンソールに直接コマンドを入力していないか
  - R スクリプトを作成する際の動作確認やインストールするためのコマンドはコンソールに直接入力して良い