

## 第 3 章 最頻値

### 3 最頻値

#### 3.1 必要なライブラリをインストールする

下の行は console で実行すると良い。

```
# console
install.packages("tidyverse")
```

#### 3.2 必要なライブラリを読み込む

```
# code
library(tidyverse)
library(systemfonts)
library(ragg)
```

#### 3.3 最頻値が何かを知る（テキスト 3.1）

##### 3.3.1 最頻値とは

「データを大きい順に並べた際、同じ値のデータが複数あることは珍しくありません。このとき、もっとも個数の多いデータが最頻値」

##### 3.3.2 データフレームの作成

R には最頻値を求める関数はない。

社員の出張回数を出してみる。

```
# code
df_trip_counts <- tibble(
  staff_name = c("A", "B", "C", "D", "E", "F"),
  trip_counts = c(0, 6, 0, 0, 0, 0)
)
df_trip_counts
```

```
## # A tibble: 6 x 2
##   staff_name trip_counts
##   <chr>         <dbl>
## 1 A             0
## 2 B             6
## 3 C             0
## 4 D             0
## 5 E             0
## 6 F             0
```

変数の型を確認しましょう。

```
# code
class(df_trip_counts)

## [1] "tbl_df"      "tbl"        "data.frame"
```

tibble データフレームであることが分かります。

```
# code
class(df_trip_counts$staff_name)
```

```
## [1] "character"
```

character 文字列なのが分かります。

```
# code
class(df_trip_counts$trip_counts)
```

```
## [1] "numeric"
```

numeric 数値なのが分かります。

### 3.3.3 平均を求めておく

```
# code
mean(df_trip_counts$trip_counts)
```

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

## 3.4 最頻値を求める（テキスト 3.2）

第2章で作成した「営業車使用年数.xlsx」のデータを使用する。

### 3.4.1 CSV ファイル

CSV ファイル名の例: zenki\_chap\_02\_01\_utf8.csv

CSV ファイルの中身

```
car_id,year_nen
1,11
2,1
3,10
(以下省略)
```

### 3.4.2 データフレームの作成

```
# code
df_eigyousha <- read_csv("zenki_chap_03_01_utf8.csv")

## Rows: 16 Columns: 2
## -- Column specification -----
## Delimiter: ","
## dbl (2): car_id, year_nen
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

glimpse(df_eigyousha)

## Rows: 16
## Columns: 2
## $ car_id    <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
## $ year_nen  <dbl> 11, 1, 10, 12, 2, 3, 10, 11, 2, 1, 12, 2, 11, 12, 8, 11
```

## 3.5 count() を使った最頻値の求め方

最頻値の求め方には以下のようなものがある。

- count() 関数で各値の出現回数を集計
- slice\_max() 関数で最も出現回数の多い行を抽出
- pull() 関数で最頻値を取得

```
# code
```

```
df_eigyousha |>  
  count(year_nen) |>  
  slice_max(n) |>  
  pull(year_nen)
```

```
## [1] 11
```

なお、各関数の実行結果を表示しておく。

```
# code
```

```
df_eigyousha |>  
  count(year_nen)
```

```
## # A tibble: 7 x 2  
##   year_nen      n  
##   <dbl> <int>  
## 1      1      2  
## 2      2      3  
## 3      3      1  
## 4      8      1  
## 5     10      2  
## 6     11      4  
## 7     12      3
```

```
# code
```

```
df_eigyousha |>  
  count(year_nen) |>  
  slice_max(n)
```

```
## # A tibble: 1 x 2  
##   year_nen      n  
##   <dbl> <int>  
## 1     11      4
```

## 3.6 章末問題

### 3.6.1 問題

以下のデータは、ある観光期におけるレストラン（11 店舗）のメニュー数を示したものです。最頻値を求めてください。

### 3.6.2 解答例

#### 3.6.3 データフレームの作成

```
# code
df_menu <- tibble(
  menu = c(15, 9, 10, 24, 20, 18, 8, 35, 27, 10, 13)
)
glimpse(df_menu)
```

```
## Rows: 11
```

```
## Columns: 1
```

```
## $ menu <dbl> 15, 9, 10, 24, 20, 18, 8, 35, 27, 10, 13
```

#### 3.6.4 最頻値の計算

```
# code
df_menu |>
  count(menu) |>
  slice_max(n) |>
  pull(menu)
```

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

答え: 10

### 3.7 テキストと参考文献

#### 3.7.1 テキスト

- 玄場公規, 湊宣明, 豊田裕貴, 2016, 『Excel で学ぶビジネスデータ分析の基礎ビジネス統計スペシャリスト・エクセル分析ベーシック対応』, オデッセイコミュニケーションズ.

#### 3.7.2 参考文献

- ウィットカム, 2024, 『R ではじめるデータサイエンス第2版』, オライリー・ジャパン.