

## 第 4 章 レンジ

### 4 レンジ

#### 4.1 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")
```

#### 4.2 必要なライブラリを読み込む

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

#### 4.3 レンジとは何かを知る (テキスト 4.1)

「レンジ (range) とは範囲という意味です。その名のとおり、データが分布している範囲です。レンジはデータの最大値から最小値を引いて求めることができます。」

#### 4.4 各月のクレーム件数 (表 4.1)

各月のクレーム件数を CSV ファイルにしてアップロードする。

ファイル名の例: zenki\_chap\_04\_01\_utf8.csv

CSV ファイルの中身は以下の通り。

```
month,claim_count  
1月,30  
2月,20  
3月,40
```

4月,90

(以下省略)

read\_csv() で読み込む。

```
# code
```

```
df_claim <- read_csv("zenki_chap_04_01_utf8.csv")
```

```
## Rows: 12 Columns: 2
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (1): month
```

```
## dbl (1): claim_count
```

```
##
```

```
## 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.
```

```
df_claim
```

```
## # A tibble: 12 x 2
```

```
##   month claim_count
```

```
##   <chr>      <dbl>
```

```
## 1 1月          30
```

```
## 2 2月          20
```

```
## 3 3月          40
```

```
## 4 4月          90
```

```
## 5 5月         100
```

```
## 6 6月          60
```

```
## 7 7月          20
```

```
## 8 8月          70
```

```
## 9 9月          20
```

```
## 10 10月         30
```

```
## 11 11月         10
```

```
## 12 12月         20
```

データフレームの中身を確認する。

```
# code
```

```
glimpse(df_claim)
```

```
## Rows: 12
```

```
## Columns: 2
```

```
## $ month      <chr> "1月", "2月", "3月", "4月", "5月", "6月", "7月", "8月", "9月", "10月~
```

```
## $ claim_count <dbl> 30, 20, 40, 90, 100, 60, 20, 70, 20, 30, 10, 20
```

変数の型を確認する

```
# code
class(df_claim)
```

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

```
# code
class(df_claim$month)
```

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

```
# code
class(df_claim$claim_count)
```

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

## 4.5 降順

各月のクレーム件数を降順にしてみる。

`arrange()` 関数と `desc()` 関数を組み合わせて用いる。

- `arrange()`: データフレームを指定した列の値に基づいて並び替える。(ウィツカム 2024: 38-39)
- `desc()`: 引数の列を降順に指定する。(ウィツカム 2024: 39)

```
# code
df_claim |>
  arrange(desc(claim_count))
```

```
## # A tibble: 12 x 2
##   month claim_count
##   <chr>      <dbl>
## 1 5月         100
## 2 4月          90
## 3 8月          70
## 4 6月          60
## 5 3月          40
## 6 1月          30
## 7 10月         30
## 8 2月          20
## 9 7月          20
## 10 9月         20
```

```
## 11 12月      20
## 12 11月      10
```

claim\_count の降順にデータの並べ替えが行われたのがわかる。

「最大値は 5 月の「100」、最小値は 11 月の「10」です。つまり、このデータのレンジは以下の式で「90」と求めることができます。」

$$100 - 10 = 90$$

R で一応計算しておく。

```
# code
100 - 10
```

```
## [1] 90
```

## 4.6 レンジを求める (4.2)

### 4.6.1 CSV ファイルの作成

Excel で各月の受注数の CSV ファイルを作成する。

CSV ファイル名の例: zenki\_chap\_04\_02\_utf8.csv

CSV ファイルの中身

```
month,order_count
1月,340
2月,400
3月,560
(以下省略)
```

### 4.6.2 CSV ファイルのアップロード

RStudio にファイルをアップロードする。

### 4.6.3 CSV ファイルの読み込み

read\_csv() で読み込む。

```
# code
df_count <- read_csv("zenki_chap_04_02_utf8.csv")
```

```
## Rows: 12 Columns: 2
```

```
## -- Column specification -----
```

```
## Delimiter: ","
## chr (1): month
## dbl (1): order_count
##
## 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.
df_count
```

```
## # A tibble: 12 x 2
##   month order_count
##   <chr>      <dbl>
## 1 1月         340
## 2 2月         400
## 3 3月         560
## 4 4月         550
## 5 5月         480
## 6 6月         320
## 7 7月         610
## 8 8月         590
## 9 9月         380
## 10 10月        620
## 11 11月        650
## 12 12月        550
```

## 4.7 レンジを求める

### 4.7.1 最大値と最小値からレンジを求める

- `max()`: 最大値を求める関数

```
# code
max(df_count$order_count)
```

```
## [1] 650
```

- `min()`: 最小値を求める関数

```
# code
min(df_count$order_count)
```

```
## [1] 320
```

レンジを求める。最大値から最小値を引く。

```
# code
max(df_count$order_count) - min(df_count$order_count)
```

```
## [1] 330
```

#### 4.7.2 range() 関数を使うやり方

R には最大値と最小値を一度に計算する関数 `range()` がある。

- `range()`: 値列の最小値と最大値をベクトルとして返す
- `diff()`: 差分を計算する

```
# code
range(df_count$order_count)
```

```
## [1] 320 650
```

```
# code
range(df_count$order_count) |>
  diff()
```

```
## [1] 330
```

#### 4.7.3 summarize() 関数を使う方法

```
# code
df_count |>
  summarize(
    count_min = min(order_count),
    count_max = max(order_count),
    count_range = count_max - count_min
  )
```

```
## # A tibble: 1 x 3
##   count_min count_max count_range
##   <dbl>     <dbl>     <dbl>
## 1      320      650         330
```

## 4.8 章末問題

### 4.8.1 問題

「以下のデータは、ある店の1年間の来店者数を表しています。このデータのレンジを求めてください。」

#### 4.8.2 解答例

CSV ファイルを作成する。

CSV ファイル名の例: zenki\_chap\_04\_shoumatu.csv

CSV ファイルの中身

```
month,visitor_count
1,560
2,320
3,624
(以下省略)
```

CSV ファイルの読み込み。

```
# code
df_visitor <- read_csv("zenki_chap_04_shoumatu.csv")

## Rows: 12 Columns: 2
## -- Column specification -----
## Delimiter: ","
## dbl (2): month, visitor_count
##
## 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.

df_visitor

## # A tibble: 12 x 2
##   month visitor_count
##   <dbl>         <dbl>
## 1     1             560
## 2     2             320
## 3     3             624
## 4     4             670
## 5     5             770
## 6     6             650
## 7     7             890
## 8     8             220
## 9     9             456
## 10    10            560
```



```
## 11      11      534
## 12      12      290
```

```
# code
```

```
max(df_visitor$visitor_count) - min(df_visitor$visitor_count)
```

```
## [1] 670
```

又は

```
# code
```

```
range(df_visitor$visitor_count) |>
  diff()
```

```
## [1] 670
```

## 4.9 テキストと参考文献

### 4.9.1 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.

### 4.9.2 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.