

自分で触ってよくわかる

# 変数作成の話:

個々の変数へのアクセスと新規作成

神戸市立医療センター中央市民病院  
臨床研究推進センター

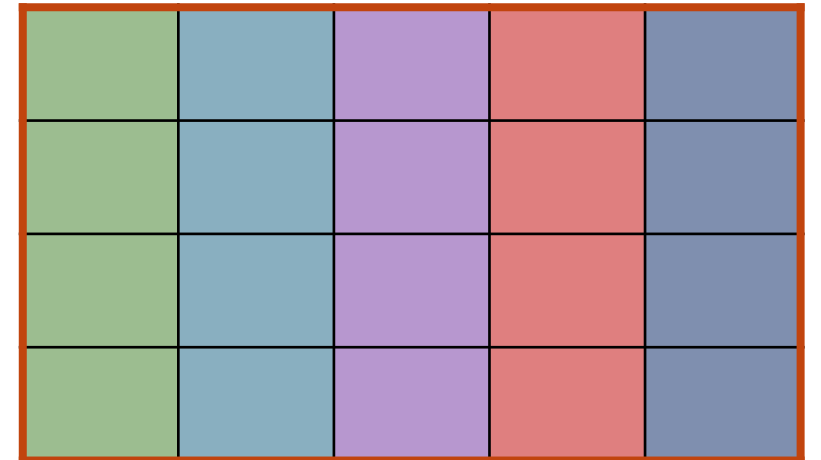
宮越 千智

# 今回の学習目標

- ✓ データフレームに含まれるある変数だけを抽出することができる
- ✓ 他の変数の値をもとにして新しい変数を作成することができる

# データフレームは変数の集合

- **変数**: 同じ型の値が1次元に並んだデータ構造
  - ✓ 「同じ型」→ 数値と文字列が混ざっているのはダメ
  - ✓ Rではベクトル(vector)と呼ぶ
  - ✓ Python(pandas)ではシリーズ(Series)と呼ぶ
- **データフレーム**: 同じ長さの変数をまとめた2次元のデータ構造
  - ✓ Rではdata.frameと書く
  - ✓ Python(pandas)ではDataFrameと書く
- **行列とデータフレームの比較**:
  - ✓ 両方とも2次元構造
  - ✓ データフレームは変数ごとにデータ型が異なってよい
  - ✓ 行列は全て同じデータ型でなければならない



# 変数へのアクセス

- データフレームからある変数のみを取り出す方法はいくつかある  
(以下、汎用性が高い方法を紹介)
- 取り出した変数に関数やメソッドを適用することができる

R(標準)	Python(pandas)
df\$変数名	df[ '変数名' ]
例: mean(pbc\$age)	例: df[ 'age' ].mean()

Rを使いたい人 

# Rを使いたい人: 変数同士の計算から新しい変数を作成する

1. 10人分の身長(m)・体重(kg)を含んだサンプルデータを作成する

```
data <- data.frame(  
  weight = c(70, 80, 60, 90, 75),  
  height = c(1.75, 1.80, 1.65, 1.90, 1.70)  
)
```

2. BMIを計算し、新しい変数として追加する

- ✓ BMI = 体重(kg)/身長(m)<sup>2</sup>
- ✓ 変数名はBMIとします

```
data$BMI <- data$weight/(data$height)^2
```

別法: tidyverseパッケージのmutate( )関数を用いる

```
data <- data %>%  
  mutate(BMI = weight/(height^2))
```

Rを使いたい人:


## 他の変数の値によって新しい変数の取る値を決める

1. survialパッケージのpbcデータが使える状態にしておく
2. if\_else( )関数あるいはcase\_when( )関数で条件と対応する値を指定して、mutate( )関数で新しい変数を作成する

```
pbc %>% mutate(新しい変数名 = if_else(条件式, 真の場合の値, 偽の場合の値))  
pbc %>% mutate(新しい変数名 = case_when(条件式1 ~ 真の場合の値,  
                                         条件式2 ~ 真の場合の値,  
                                         ...  
                                         TRUE ~ どの条件にも該当しない場合の値))
```

✓ 例: 年齢が65歳以上の場合「Yes」、65歳未満の場合「No」を取る、age\_over65という名前の変数を作成する

```
pbc %>% mutate(age_over65 = if_else(age>=65, "Yes", "No"))
```

Pythonを使いたい人 





# Pythonを使いたい人: 変数同士の計算から新しい変数を作成する

1. 10人分の身長(m)・体重(kg)を含んだサンプルデータを作成する

```
data = {  
    'weight': [70, 80, 60, 90, 75],  
    'height': [1.75, 1.80, 1.65, 1.90, 1.70]  
}  
df = pd.DataFrame(data)
```

2. BMIを計算し、新しい変数として追加する

- ✓ BMI = 体重(kg)/身長(m)<sup>2</sup>
- ✓ 変数名はBMIとします

```
df['BMI'] = df['weight']/(df['height']**2)
```



# Pythonを使いたい人: 他の変数の値によって新しい変数の取る値を決める

1. survialパッケージのpbcデータが使える状態にしておく

```
import pandas as pd
import statsmodels.api as sm
dataset = sm.datasets.get_rdataset("pbc", "survival")
df = dataset.data
```

2. numpyの.where( )メソッドを使う

✓ 例: 年齢が65歳以上の場合「Yes」、65歳未満の場合「No」を取る、age\_over65という名前の変数を作成する

```
import numpy as np
df['age_over60'] = np.where(df['age']>=65, 'Yes', 'No')
```

# 課題8：変数作成

- Rのsurvivalパッケージにあるpbcデータについて、臨床スコアをscoreという変数名で新しく作成してみましょう
  - ✓ 以下の点数の合計点を臨床スコアとする

検査項目	変数名	基準	点数
アルブミン	albumin	$\geq 3$	0点
		$< 3$	1点
ビリルビン	bili	$< 2$	0点
		$2 \leq, < 5$	1点
		$\geq 5$	2点

# 今回のまとめ

- ✓ データを2次加工して得られる変数は手入力するのではなく、収集されたデータから作成するようにしましょう  
(その方が手間もミスも少なくなります)
- ✓ 条件分岐式は長くなることがあるので、適宜改行して可読性を保ちましょう