

自分で触ってよくわかる
変数と関数の話:
コードを書いて実行してみる

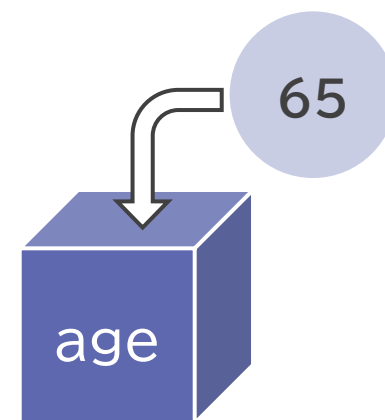
神戸市立医療センター中央市民病院
臨床研究推進センター

宮越 千智

変数(variable)とは

- データを一時的に保存するための**名前付きの記憶場所**

- ✓ 中身が変わっても同じ名前で呼ぶことができる(再利用性)
- ✓ 中身が何を意味しているのか分かりやすい(可読性の向上)



- 保存するデータは単一の値に限らない

例1: スカラー型	単一の要素	<code>age = 65</code> <code>name = “田中”</code>
例2: ベクトル型	要素が1列に並んだもの	<code>ages = [65, 74, 58]</code> <code>names = [“田中”, “佐藤”, “鈴木”]</code>
例3: マトリックス型	要素が縦横に並んだもの	<code>data = [ages, names]</code> * <code>ages, names</code> は例3のもの

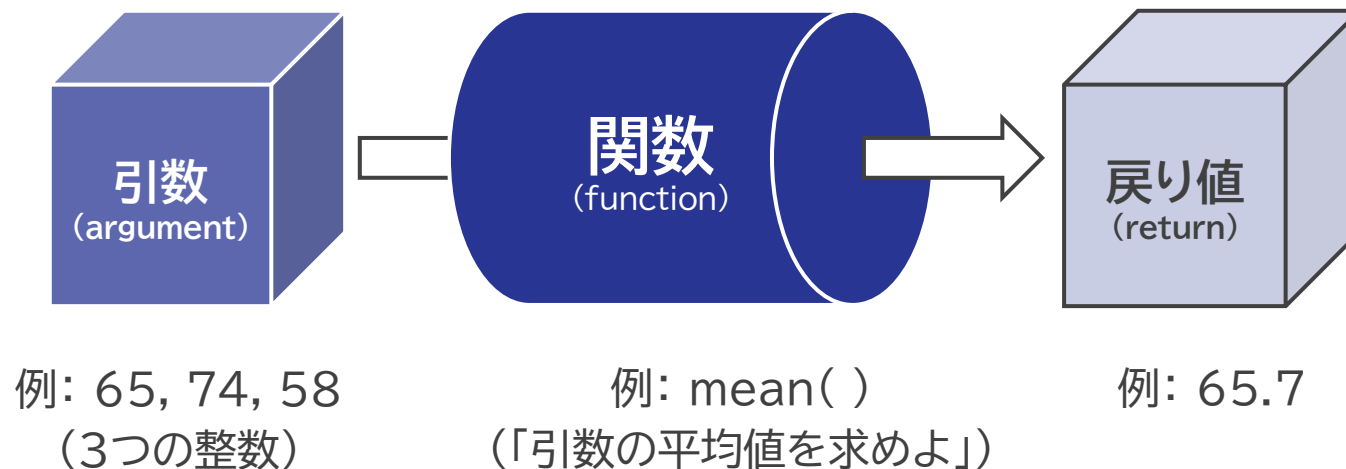
*「型」の名称はプログラム言語や使用している拡張ライブラリによって多少変わります

復習: 要素 機械の視点でみた変数の型

変数の型	説明	例
文字列型	文字の並びを値にとる 数字を文字列として認識させることもある	M, F, “1”
整数型	整数のみ	-1, 0, 1
浮動小数点数型 (実数型)	小数もOK	1.0, 3.14
ブール型	真か偽かどちらかの値をとる 論理式の真偽を示すために使う	True, False
日付・時間型	日付や時間を表す 期間の計算が可能	2024-02-25 02:50:15

関数(function)とは

- データを受け取り、定められた処理を実行して、その結果を返す命令のこと
 - ✓ 引数: 関数に渡されるデータ。関数によって引数に取れる型の制約がある場合がある。
 - ✓ 戻り値: 出力される結果。これを他の関数の引数として再利用することも可能。



- 関数は最初から用意されているもの以外に、自分で定義(宣言)することも可能



Pythonを使いたい人:

クラス・オブジェクト・属性・メソッド

- Pythonでは関数のほかに「メソッド」という方法で操作することがある

クラス (class)	オブジェクトの仕様書、あるいは設計図や鋳型のようなもの そのクラスが持っている属性や、適用できるメソッドが定義されている	例:「ヒト」
属性 (attribute)	オブジェクトの特性や状態	例:「年齢」
メソッド (method)	オブジェクトに行うことができる操作 オブジェクトが行うことができる行動	例:「挨拶する」
オブジェクト (object)	クラスをもとにして作られた個々の実物	例:「神戸太郎」

- 対象のオブジェクト名の後にピリオド(.)をつけて属性を取り出したり、メソッドを適用したりする
 - ✓ 例: 神戸太郎.年齢 → 45歳(オブジェクトの属性の取り出し)
 - ✓ 例: 神戸太郎.挨拶 → 「神戸太郎さん、おはようございます」(オブジェクトに対する操作)

Rを使いたい人 

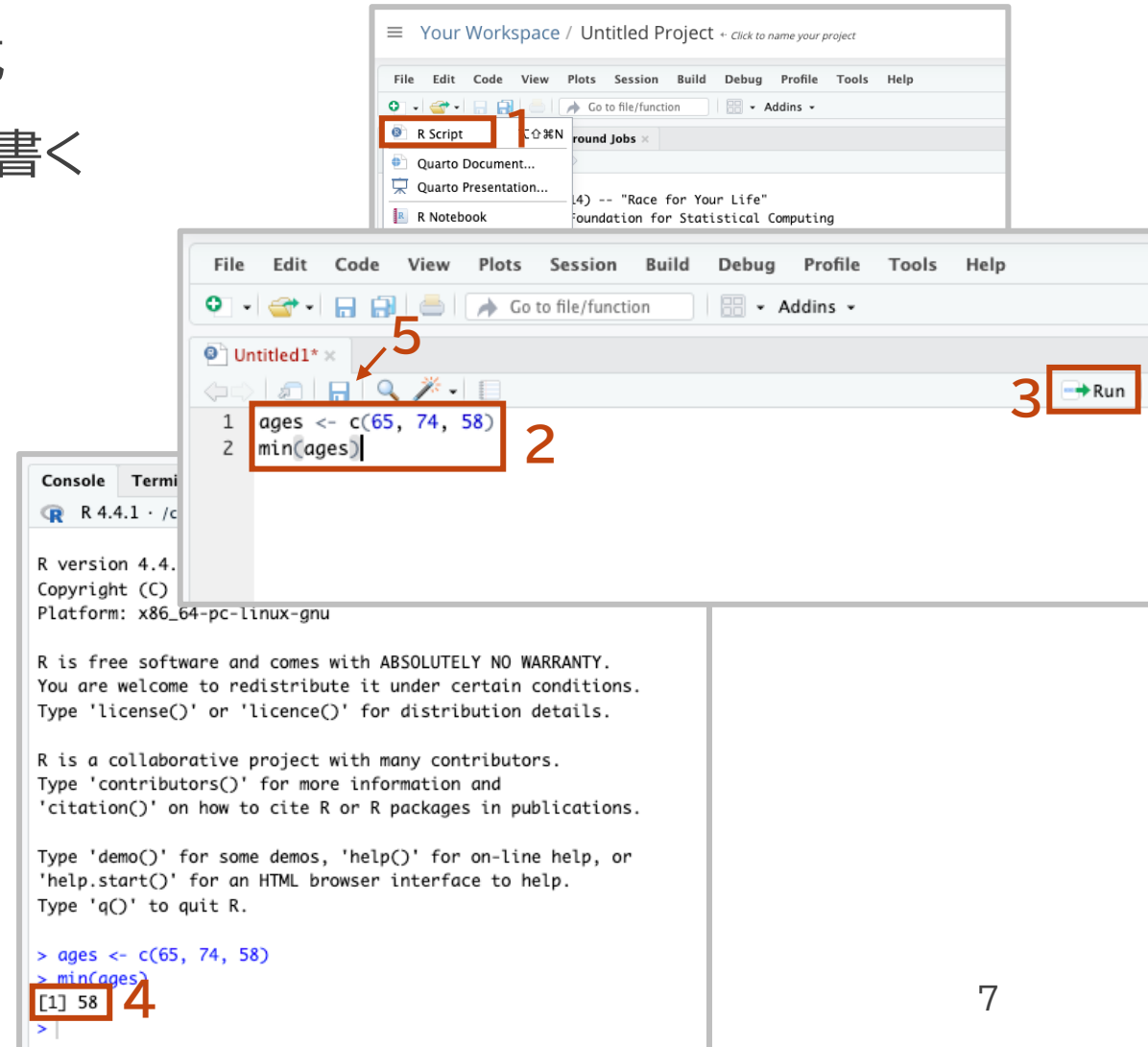
Rを使いたい人:

RStudioで変数を保存し、関数を実行する

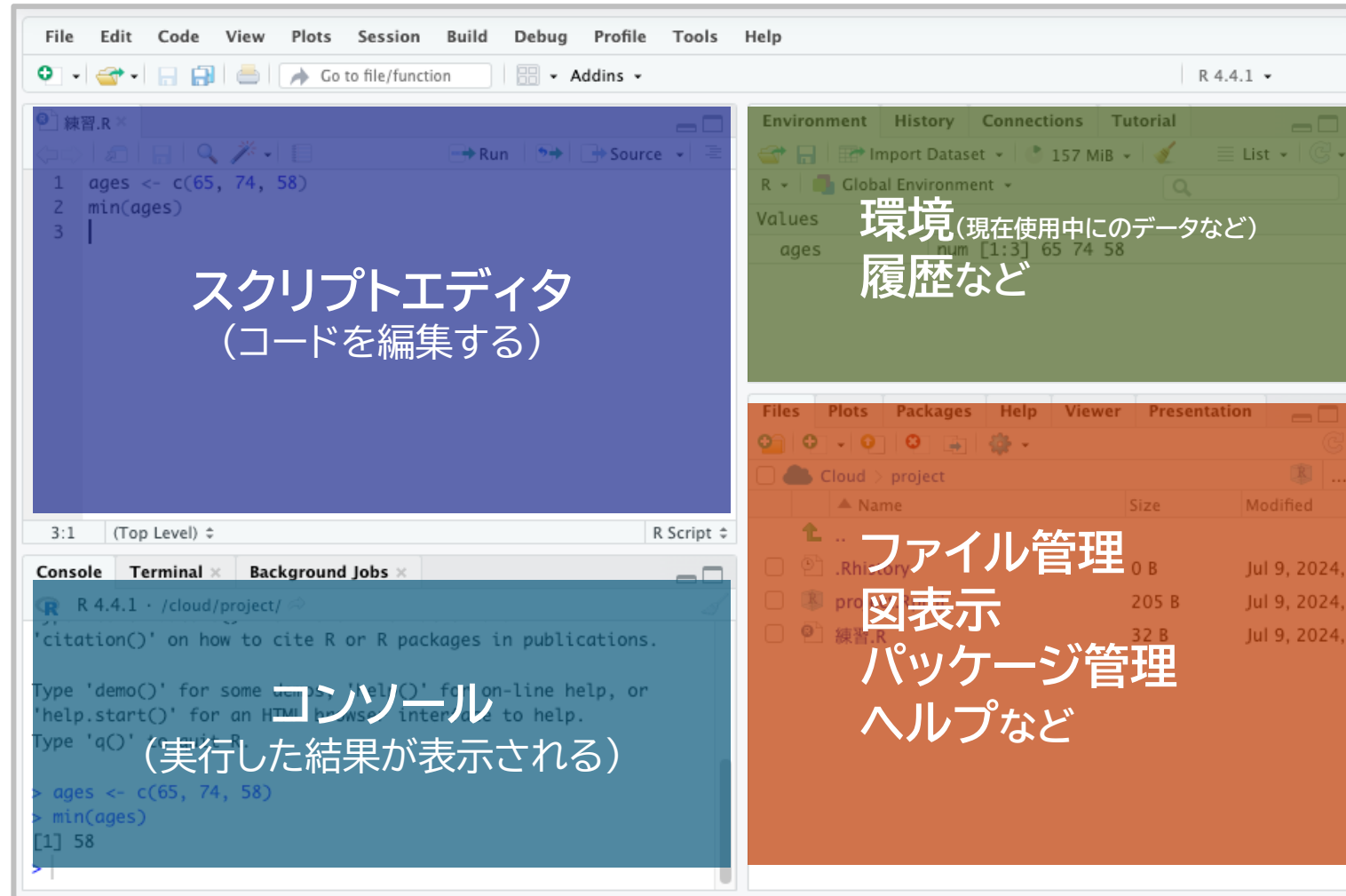
1. 左上の[緑+] から [R Script] を新規作成
2. 新しく出現したRスクリプトのタブにコードを書く
 (“<-”は←の意味。“=”でもOK)

```
ages <- c(65, 74, 58)
min(ages)
```

3. コードを選択して [Run] を押す
 - ✓ Macショートカットキー: ⌘ + ↵
 - ✓ Windowsショートカットキー: Ctrl + ↵
4. コンソールタブに表示された結果を確認する
5. 次回も使いたいときは保存しておく



Rを使いたい人: RStudioの操作画面



復習:

変数名の付け方

- 簡潔で、中身が分かりやすい名前をつける

- ✓ 例: `age2`(2値化?バージョンが2?) → `age_cat`(カテゴリー化した年齢)

- 接頭辞、接尾辞を工夫すると一括操作がしやすい

- ✓ 例: `date_adm`, `date_event`(`date_`を使って日付を表す変数を一括で選択できる)


- 2値変数はどちらに1を割り当てているかが分かれると便利

- ✓ 例: `sex`(男女どちらが1?) → `female`(女性=1と想像しやすい)

- 統一感があると読みやすい

- ✓ キャメルケース(`dateAdm`など):最初は小文字で、あとの単語の先頭は大文字にする

- ✓ スネークケース(`date_adm`):アンダースコアで単語を区切る

Pythonを使いたい人 



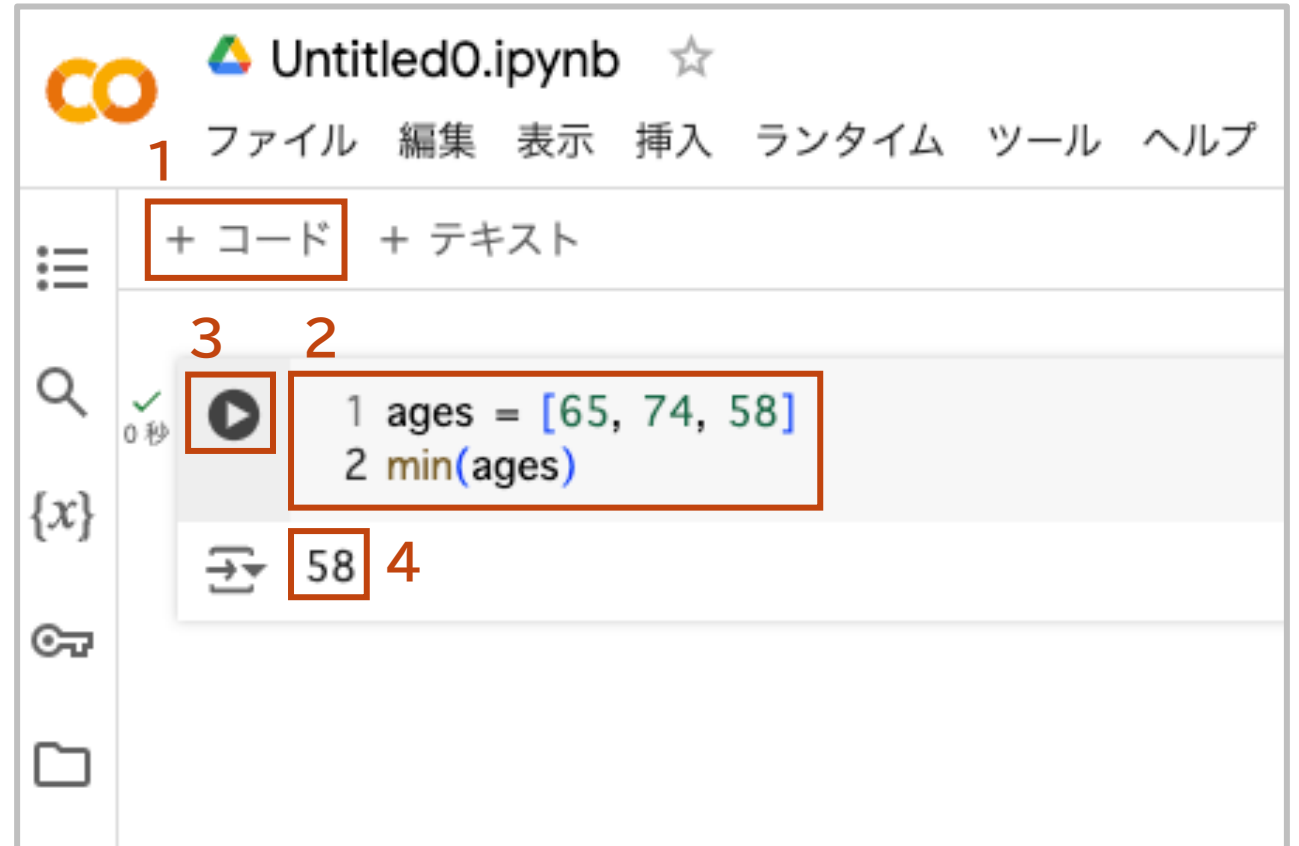
Pythonを使いたい人:

Google Colabで変数を保存し、関数を実行する

1. コードブロックを作成する(最初から1つある)
2. コードブロックにコードを書く

```
ages = [65, 74, 58]  
min(ages)
```

3. 三角の実行ボタンを押す
 - ✓ Macショートカットキー: ⌘ + ↵
 - ✓ Windowsショートカットキー: Ctrl + ↵
4. 表示された結果を確認する



課題1：関数を実行する

- 5つの整数 1,2,3,5,8 を“numbers”という変数名で保存してください
- max()という関数を使って、“numbers”の最大値を求めてください

困ったら生成AIに教えてもらおう

Chat GPT(まずは3.5でもOK)

ChatGPT 3.5

5つの整数 1,2,3,5,8 を “numbers” という変数名で保存するRコードを教えてください。

以下のようR言語で変数`numbers`に整数のリストを保存することができます：

```
numbers <- c(1, 2, 3, 5, 8)
```

これにより、`numbers`という変数に整数のリストが保存されます。

Gemini

Gemini

5つの整数 1,2,3,5,8 を “numbers” という変数名で保存するPythonコードを教えてください。

Python

```
numbers = [1, 2, 3, 5, 8]
```

コードは[注意](#)してご使用ください。

このコードは、以下の内容を実行します。

- `numbers` という名前の変数を宣言します。
- `numbers` 変数に、5つの整数 1, 2, 3, 5, 8 を要素とするリストを代入します。

このコードを実行すると、`numbers` 変数には以下のリストが保存されます。

```
[1, 2, 3, 5, 8]
```

このリストは、0番目の要素が1、1番目の要素が2、...、4番目の要素が8というように、5つの要素を

今回のまとめ

- ✓ ショートカットキーを覚えると作業が捗ります
- ✓ 作文はできなくても構文読解はできるレベルを目指しましょう
(作文は生成AIが助けてくれます)
- ✓ どのような目的で書いたコードなのか振り返って分かるように
コメントをつけておきましょう
(#から行の終わりまではコメントとして無視されます)

```
# 解析手順1: 最小値を求める  
ages <- c(65, 74, 58) #3人の年齢  
min(ages) #最小値を求める関数
```