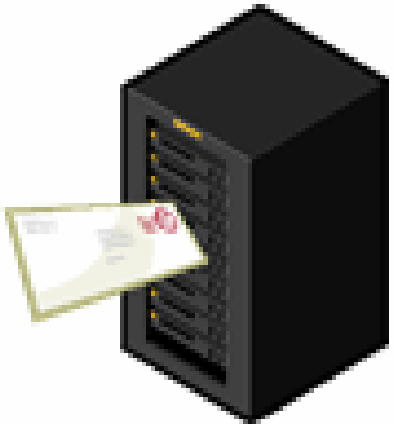
Rで学ぶプログラミングの基礎の基礎

(3) 変数とベクトル

プログラミングについて

- プログラミングとは
  - 人間がコンピュータに命令をすること
  - R の場合は「ユーザーが R のコマンドをひとつひとつ記述する」作業のこと ⇒ 関数定義！
- R におけるプログラミングのための道具は・・・
  - 変数，ベクトル，関数定義
  - 条件分岐（if）
  - くり返し（for）



本日のメニュー

1. 変数と代入
2. ベクトル
3. ベクトル操作

変数と代入

- 変数とは，値や計算結果を保存する箱のこと
- R では「変数名 <- 計算式」で計算結果を保存出来る
  これを「変数」への「代入」という（ "<-" は代入記号）

```
> x <- 1 + 2    # x という変数に 1 + 2 の結果を代入
```

- 保存した値は変数名を入力することで再表示される

```
> x
[1] 3
```

変数と代入

- 変数 を使ってさらに計算を行うことも出来る

```
> x^2
[1] 9
```

- 変数を使った計算結果を，さらに変数に代入することも可

```
> y <- x^2          # y という変数に x^2 = 9を代入
```

- 丸括弧を使うと，値の保存と表示が同時に実行できる

```
> ( y <- x^2 )
[1] 9
```

変数の命名規則

- 大文字と小文字を区別する → x と X は別物
- 変数名にはローマ字や数字を使うことができるが，
  変数名の先頭を数字にすることはできない
  → 例えば「1A」という変数名はエラーとなる
- 以下に挙げる名前は（R の処理系によって）先に予約
  されている予約語なので，変数名として使用できない
  break else FALSE for function if in Inf
  NA NaN next NULL repeat TRUE while

演習問題①

1. 変数 x に 2 を代入した後，3 + x^2 を計算してください
2. 次の文を実行すると，変数 a と変数 b には何が入っているでしょうか

```
> a <- b <- -1.5
```

3. 2. で生成した変数 a について，変数 a の整数部分を求める trunc(a)と変数 a の切り下げを行う floor(a) を実行して下さい
   → trunc(a) と floor(a) の結果の違いは何でしょうか
4. スーパーのレジに商品を持って行き，100 円を払った時のおつりを貰うときの「50 円玉の枚数」「10 円玉の枚数」「5 円玉の枚数」「1 円玉の枚数」をそれぞれ変数 M50，M10，M05，M01 に代入するプログラムを作成してください

変数を使った計算例

```
> x <- 1      # x に 1 を代入する
> y <- 2      # y に 2 を代入する
> x + y       # x の中身と y の中身の和
[1] 3
> x <- 4      # 変数に値を代入し直すと古い値は破棄され
> x           # 新しい値で上書きされる
[1] 4
```

本日のメニュー

1. 変数と代入
2. ベクトル
3. ベクトル操作

ベクトル

- ある 5 人の体重について，体重の合計値を求める

```
> x1 <- (55 + 60 + 65 + 70 + 75)
> x1
[1] 325
```

- 同じ 5 人の体重について平均値を求める

```
> x2 <- (55 + 60 + 65 + 70 + 75) / 5
> x2
[1] 65
```

- 同じ 5 人の体重について，今度は分散を求め・・・
（同じデータなのだから，手間を省く方法は無いの？？）

ベクトル

- 1 つの変数に 1 つの値を代入することが出来たように，
  1 つの変数に複数の値を代入することも出来る

```
> x <- c(55, 60, 65, 70, 75)
```

- 複数の値が代入された変数を「ベクトル」とよぶ
  ※ 1 つの値しか代入されていない変数も「ベクトル」

```
> x
[1] 55 60 65 70 75
```

ベクトル

- R には「ベクトル」用の関数が多数用意されている

```
> sum(x)
[1] 325
```

- ベクトルに入っているデータ数（要素数）を計算する場合は関数 length() を使用する

```
> length(x)
[1] 5
```

ベクトル

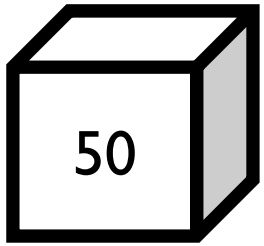
R で用意されているベクトル用関数の一部

関　数	cor(x)	max(x)	mean(x)	median(x)	min(x)
意　味	相関係数	最大値	平均値	中央値	最小値
関　数	prod(x)	summary(x)	sd(x)	sum(x)	var(x)
意　味	総積	要約統計量	標準偏差	総和	不偏分散

変数とベクトルの違い

- 変数は箱が 1 個

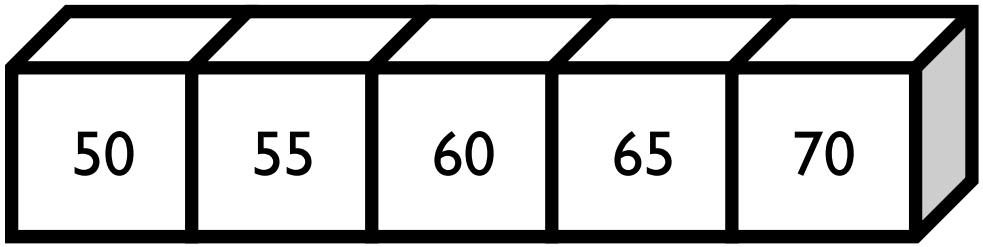
変数 a:



- ベクトルは箱が 1 個以上

→ 「ベクトルの箱が 1 つバージョン」が「変数」

ベクトル x:



本日のメニュー

1. 変数と代入
2. ベクトル
3. ベクトル操作

ベクトル操作

- ベクトルの中の数値を取り出す場合は [] を使用する

```
> x[2]            # 変数 x の 2 番目の値を取り出す
[1] 60
```

- ベクトルとベクトルを結合する場合は以下の様にする

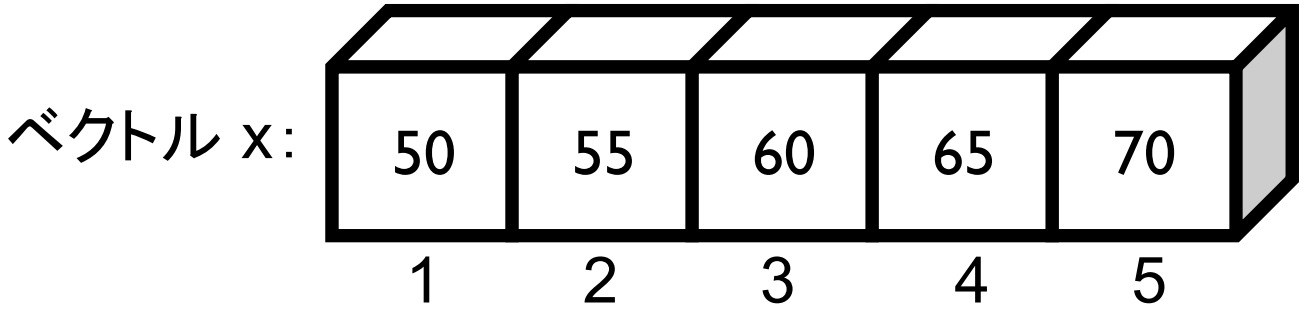
```
> c(x, 80)        # c(ベクトル, ベクトル)
[1] 55 60 65 70 75 80
```

- ベクトルの中の数値を書き換える場合は以下の様にする

```
> x[2] <- 99      # 変数 x の 2 番目の値を 99 に修正
> x
[1] 55 99 65 70 75
```

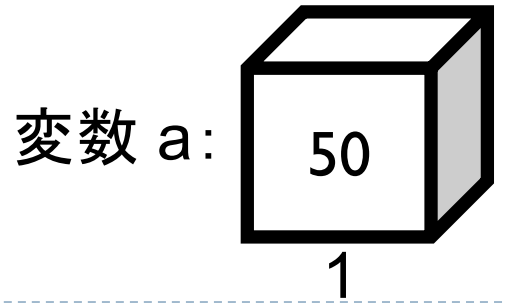
変数とベクトルの違い

- ベクトル x について，x[1] の値は 50



- 変数 a について，a[1] の値は 50

→「変数」は「ベクトルの箱が 1 つしかないバージョン」



ベクトル操作

規則性のあるベクトルを生成する関数

関数・コマンド	機能
1:5	1, 2, 3, 4, 5 を生成する
5:1	5, 4, 3, 2, 1 を生成する
seq(1, 5, by=2)	1 から 5 まで 2 ずつ増加する等差数列を生成する
rep(1, 3)	1 を 3 個繰り返した数列を生成する
rep(c(1,2,3), 3)	(1,2,3) を 3 個繰り返した数列を生成する

演習問題②

1. 「0.8-4」の結果を変数 x に代入して下さい
2. 整数部分 trunc(x) と切り下げ floor(x) の違いは何でしょう?
3. 以下の 5 つのデータを変数 y に格納して下さい

1	2	3	4	5

4. 変数 y の値と変数 Y（大文字）の値を表示して下さい
5. 3. で作成した変数 y の平均と標準偏差を求めて下さい
6. 変数 y の後ろに，変数 x を結合し，結果を変数 z に格納して下さい
7. 6. で作成した変数 z の 6 番目の値を表示して下さい

本日のメニュー

1. 変数と代入
2. ベクトル
3. ベクトル操作

おまけ

- [Tab] キーを押すことで補完機能が働く

```
> sq [Tab]
```

途中まで入力して [Tab] キーを押すと残りのスペルを補完してくれる

```
> sqrt
```

- 左下のコンソール画面で [↑] や [↓] を押すことでも今までに実行したコマンドの履歴が辿れる

```
> [↑]    （1つ前に実行したコマンド）が表示される
```

おまけ

- 記号 # を使ってコメントを付けることができる

```
> log(2)     # この文章は無視されます ↵
```

- 記号 ; で区切ることで複数の数式を同時に実行出来る

```
> log(2); log(3); log(4) ↵
[1] 0.6931472    [1] 1.098612    [1] 1.386294
```

- 関数 help() 以外にも以下のヘルプ検索方法がある

```
> help.search("log")     #  log の機能が含まれる関数
> apropos("log")         # "log"という文字列が含まれる関数
```

おまけ

- コマンド入力の途中で ↵ を押してしまうと・・・

```
> log(2 ↵
+
```

- ↑の状態になっても気にせずに続きを入力する
  （コマンド入力を中断する場合は Esc を押す）

このスライドの内容が書籍になっています

- 舟尾 暢男 『 R で学ぶプログラミングの基礎の基礎（カットシステム）』

  「R と RStudio のインストール」「R の基礎」「変数とベクトル」
  「種々のベクトル」「関数の作成」「条件分岐」「くり返し」
  「複雑なくり返し」「グラフ作成の概要」「問題集」「簡単なゲーム作成」

※ 本資料の演習問題の解答も載っています

Rで学ぶプログラミングの基礎の基礎

終