

データサイエンスへの誘い

第2回: プログラミング入門

瓜生真也 (デザイン型AI教育研究センター・助教)

先週の振り返り

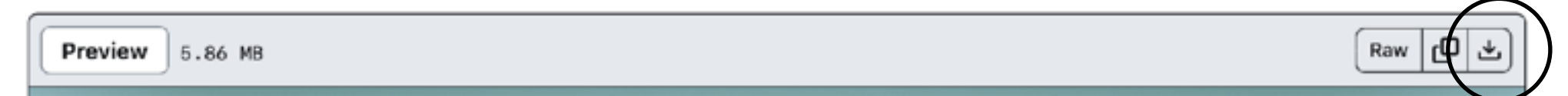
講義内容（予定）

講義に関する資料（スライド、補足資料等）を  GitHubに置いておきます

 <https://github.com/uribo/INNV1250>



ダウンロード可能



1. ガイダンス、データサイエンスとは何か
2. プログラミング基礎
3. 再現可能性
4. データ処理の手法
5. データの要約
6. データの可視化
7. データと確率
8. データからの推論
9. 複数のデータを比較する
10. 統計のウソ
11. 統計的モデリング
12. 統計的学習
13. さまざまなデータサイエンスの手法
14. 機械学習と人工知能（AI）
15. 期末試験
16. 振り返りと統括

今日の目標

JupyterとRの基本操作を学ぶ

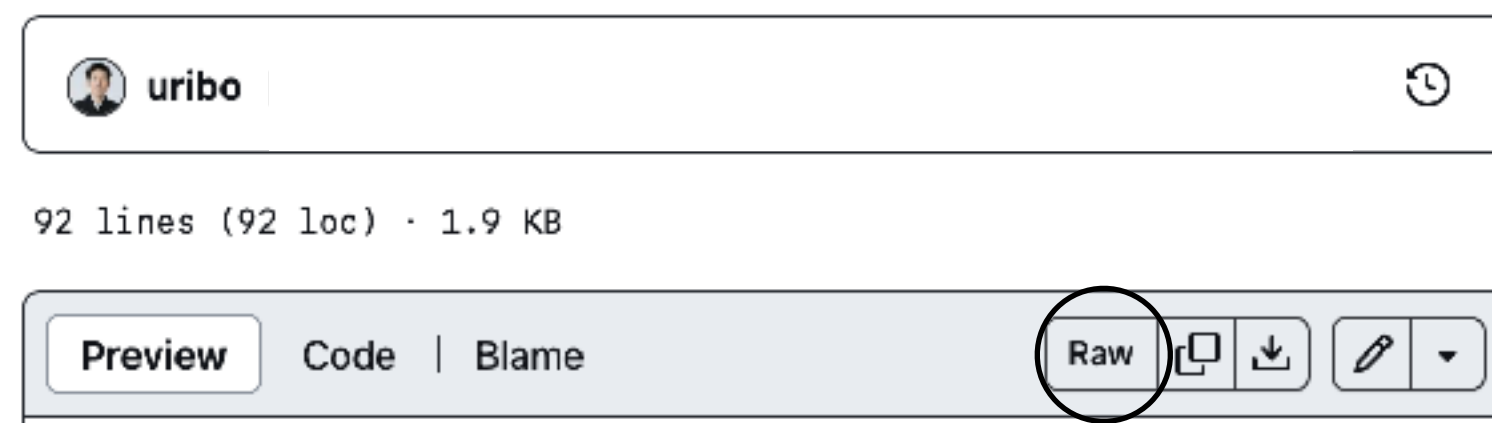
【課題】 Rの基本操作の実行

提出期限: 来週の講義開始前まで

manabaのレポートとして提出してください

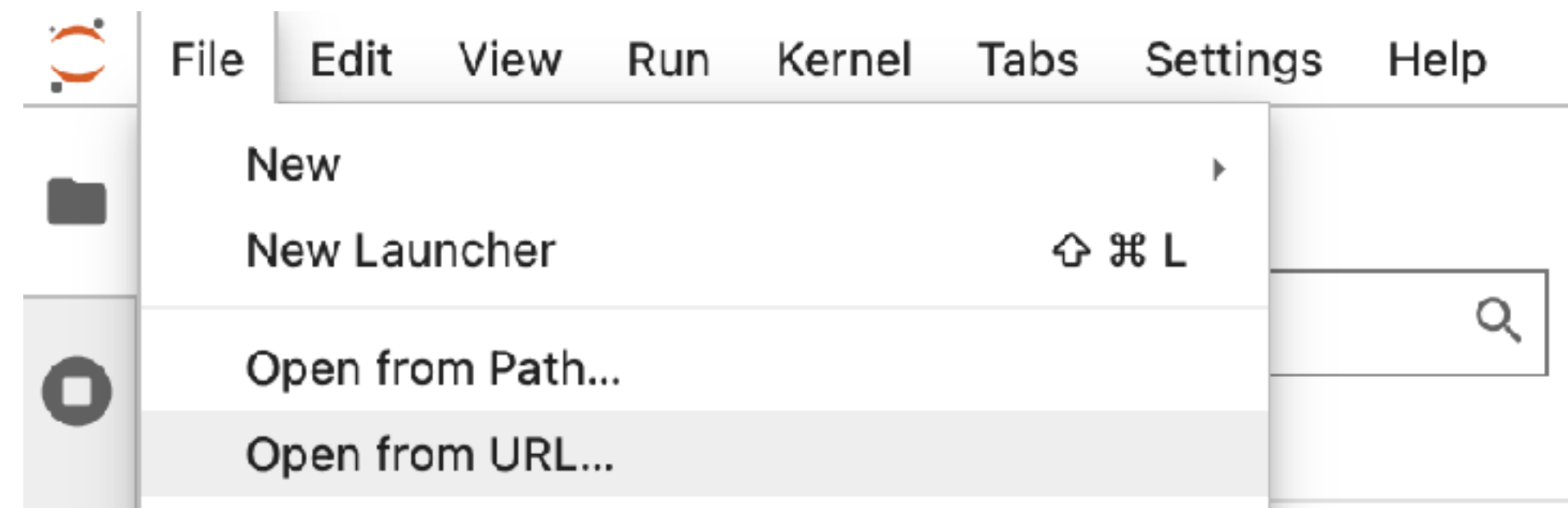
GitHubから0201_r_intro.ipynbをJupyterHubにアップロードして記載

week02/0201_r_intro.ipynb

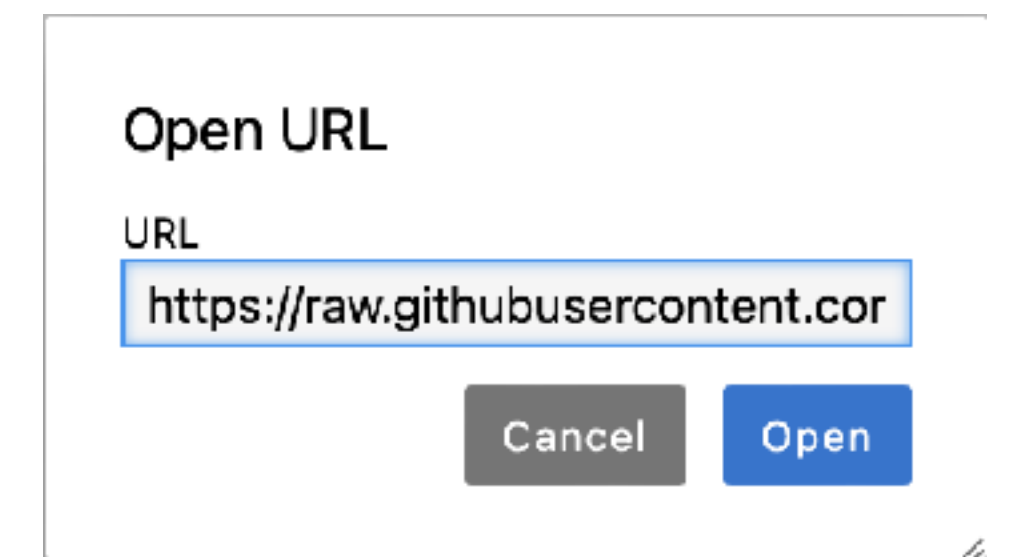


Rawをクリックして表示先のURLをコピー

JupyterHubのサーバを起動、メニューのFileから
“Open from URL…” を選択



コピーしたURLを貼り付け



注意: ファイル名は英数字のみにすること

日本語（漢字、片仮名、平仮名）、全角英数字、スペース、記号等は使わない

ファイルをダウンロードしても開けなくても問題ない（気にしない）

内容の確認、編集はJupyterHub上で行う

プログラミング

コピーから月面着陸までを可能にする

かなり端折ったプログラミングの流れ



このあたりの詳しい内容は「情報科学入門」で！

言葉の整理

プログラム(program) コンピュータが特定の作業を行うための一連の命令（の集まり）

命令（プログラム）をコンピュータに与えることで機能する

一つ一つの命令は非常に単純な処理を実行する

多くの場合、複数の命令を組み合わせて、より複雑で実用的なプログラムを作成する→アプリケーション、ソフトウェア

プログラミング

プログラムを作成・編集する過程や活動

デバッグ（エラーの検出と修正）を含む

プログラミング言語

プログラムを実装するために使用される言語

多様な言語… C、Python、R、Swift

選択する言語は目的の要件によって異なる

プログラマー

プログラムを開発・編集・デバッグする人々

プログラムの内容に応じて職種がわかる… アプリケーションエンジニア、ゲーム開発者、データサイエンティストなど

スクリプト(script)ないしコード(code)

コードはプログラムを構成するプログラミング言語で記述された一連の命令や文を指す

スクリプトはコードの断片

退屈なことはプログラムに任せよう

例えば…ウェブページのURLをもとにQRコードを作りたい
そういうウェブサービス・ツールはいくつか存在する

対象のURLが1つならいいかも 対象のURLをコピーして貼り付け、生成 😊

URLが1000種類あったら？ 1000回もやってられない 😓



プログラミングでできることの例

データ分析、機械学習・深層学習、ウェブスクレイピング、アプリケーション開発

プログラミング言語ごとに得意・不得意、向き不向きがある

プログラムとAIの力で時間短縮

戦略1: プログラミングを学ぶ

まずはここからはじめよう

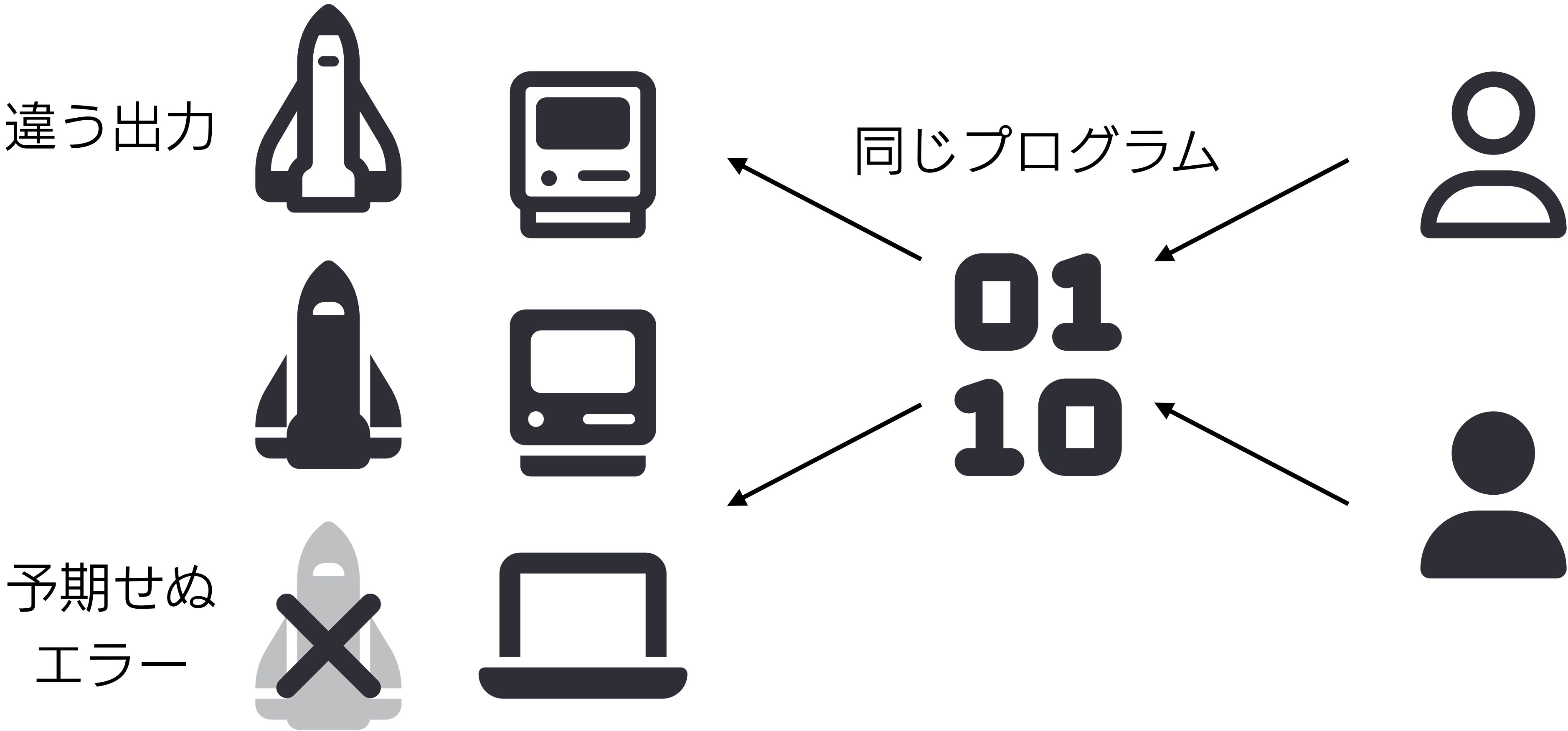
戦略2: AIの力を借りる

後半で解説

実行環境の違いに注意

プログラム実行時に利用されるコンピュータのハードウェア・ソフトウェアの環境
人も機械もさまざま

異なる結果



Jupyter

Project Jupyter

<https://jupyter.org/>

科学的研究における再現性とインタラクティブ性を高めるためのプロジェクト

Jupyter Notebook

コードと実行結果、またそれらを補足する文章（画像等を含む）からなるファイルを作成するためのツール
ノートブックファイルはJSON形式でメタデータ等を記録した .ipynb として扱われる→テキストファイル

Python, R, Juliaを始め、40以上の言語に対応

ウェブアプリケーションとしてウェブブラウザ上で動作する

Google Colaboratory (Colab)… Googleが提供するノートブック環境

Binder

Kaggle Notebooks

JupyterLab

Jupyter Notebookのための優れた実行環境

タブ機能による複数ファイルの同時編集やファイル操作が柔軟に可能

→Jupyter Notebook単体で使うよりも操作性に優れる



Jupyter Notebook

The screenshot shows a Jupyter Notebook interface in a web browser. The address bar indicates the URL is `localhost:8888/lab/tree/week2/qrcode.ipynb`. The notebook has a single tab titled `qrcode.ipynb`. The kernel is `Python 3 (ipykernel)`. The notebook content is in Japanese and includes a title cell, two text cells, a code cell with a terminal command, and three code cells for QR code generation.

ウェブサイトのURLからQRコードを作成する

まずはプログラムの実行に必要なPythonライブラリをインストールします。QRコードの作成には、`qrcode`と`pillow`の2つのライブラリを使用します。pipコマンドでライブラリをインストールしましょう。

Jupyternotebookでは、コードセルの行頭に `!` を付けた内容をコマンドとして処理します。すなわち、次のコードセルの内容はターミナルで `pip install qrcode[pil]` `pillow` を実行したときと同じです。

なお、次のセルはコードセルではなくテキストセル扱いになっているので、実際にこのノートブック上でインストールを行う際はセルの種類を `Raw` から `Code` に変換する必要があります。

```
! pip install qrcode[pil] pillow
```

1つのURLからQRコードの生成

pipコマンドにより2つのライブラリのインストールが終わったらPythonでのQRコード生成を行います。さきほどインストールしたライブラリを `import` によって読み込みましょう。こうすることで `qrcode` や `Image` ライブラリの機能が利用可能となります。

```
[1]: # QRコードの生成に必要なライブラリを読み込む
import qrcode
from PIL import Image
```

```
[2]: # QRコードにエンコードするテキスト
data = "https://github.com/uribo/exeai"
```

```
[3]: # QRコードの設定
```

At the bottom of the interface, the status bar shows `Simple` mode, `0` lines, `2` cells, `Python 3 (ipykernel) | Idle`, `Mode: Edit`, `Ln 1, Col 26`, `qrcode.ipynb`, and `0` notifications.

Rのいろは

https://speakerdeck.com/s_uryu



Uryu Shinya

s_uryu

7 Followings

25 Followers

54 Stars

Edit my profile

Decks

『R言語入門』をクリック

統・再現性・協力
人為的過誤を防ぎ、未来へ進む策
瓜生真也



20230427@skye-RH106@オンライン

uribo u_ribo

統・再現性・協力: 人為的過誤を防ぎ、未来へ...

s_uryu ☆ 1 110

2022-12-17 統計数理研究所共同研究集会「データ解析環境Rの整備と利用」

Rによる大規模データの処理

瓜生真也 (徳島大学デザイン型AI教育研究センター)
@u_ribo

https://github.com/uribo/cus2022aw_r104

Rによる大規模データの処理

s_uryu ☆ 2 800

R言語入門

R-4.2.2(2022年11月版)

瓜生真也

https://github.com/uribo/cus2022aw_r104

@u_ribo

R言語入門 (R-4.2.2 2022年11月版) / introdu...


s_uryu ☆ 2 1.4k

AIによるプログラミング支援


分からないことをChatGPTに聞いてみよう

グラフの作成方法

Model: Default (GPT-3.5)

 Rのggplot2パッケージを使ってワッフルチャートを作成したいです。コードを示してください



 ggplot2パッケージを使用してワッフルチャートを作成するためには、先にビジュアル化するデータセットを準備する必要があります。ワッフルチャートは、単純なカウントデータをグリッド状の正方形で表現する方法で、通常、割合やパーセンテージなどを示すために使用されます。

以下は、ggplot2パッケージを使用してワッフルチャートを作成するための基本的なコードです。

R


Copy code

```
# 必要なライブラリを読み込む
library(ggplot2)
library(waffle)


# ワッフルチャート用のデータを作成する
```

Regenerate response

パッケージのインストール方法

 macを使っています。rstanパッケージがインストールできずに困っています。Rのバージョンは4.2.1を使っています。



 macOS上でrstanパッケージをインストールするには、まずRtoolsとXcodeのインストールが必要です。また、Rバージョン4.2.x以降を使用する場合は、C++14が必要です。これを確認するには、ターミナルから以下のコマンドを実行してください。

bash

Copy code

```
echo 'int main(){}' | g++ -std=c++14 -xc++ -
```

もし以下のようなエラーが表示された場合は、C++14がサポートされていない可能性があります。

csharp

Copy code

```
error: invalid argument 'std=c++14' not allowed with 'C++11'
```

Regenerate response

【推奨】 GitHubの活用

GitHub by Microsoft

プログラミングのソースコード置き場、開発者のためのプラットフォーム
バージョン管理システムGitを用いたコード共有
学生であれば有料プランを無料で利用可能（教育プログラム、要申請）

GitHub Copilot

コメント、関数名等からコードの入力を自動補完するAIツール
VSCodeの機能拡張としても利用可能
↑ 上記の教育プログラムに加入で無料

【推奨】GitHubの活用

tidyverse.R U ●



tidyverse.R > ...

```
1  # 入力の際に表示されるグレーの文字は
2  # GitHub Copilotによる入力補完です
3  # このファイルはGitHub Copilotによって自動生成されました
4  library(tidyverse)
5  df <- read_csv("https://raw.githubusercontent.com/rstudio/c
6
7  ggplot(df, aes(x = reorder(country, -lifeExp), y = lifeExp))
   geom_col() +
   coord_flip() +
   labs(x = NULL, y = "Life Expectancy (years)", title = "Li
```

main*+



0 0 1

スペース: 4

UTF-8

LF

R

