

アドヴァンストコース  
S-4 情報科学・数理科学  
を用いた医学研究

1限: 環境構築・計算演習

医学医療系 助教 矢野貴大

# 注意事項

- 講義はZoom
- 文字ベースの質問はTeams
  - 2021年度 アドヴァンストコース S-4  
「情報科学・数理科学を用いた医学研究」
  - 実習中にわからないことや詰まったりした場合は、  
「質問(なんでも)」チャンネルへ投稿
    - 担当時間以外の教員の誰かが対応
  - 気軽に投稿してください
- 画面共有・音声通話を伴う質問はZoomのブレイクアウト
  - ホストが教員と学生を誘導

# アドヴァンストコース S-4: 9/7 (火)

- 1限: 環境構築・計算演習 (矢野 貴大)
  - 8:40~9:55
- 2限: 数理データ解析 (讃岐勝)
  - 10:10~11:25
- 3限: ゲノムデータ解析 (尾崎遼)
  - 12:15~13:30
- 4限: 医療データ解析 (1) (久米 慶太郎)
  - 13:45~15:00
- 5限: 医療データ解析 (2) (香川璃奈)
  - 15:15~16:30

# 講義資料

- 講義内容に関する資料が置かれているリポジトリ
  - <https://github.com/bioinfo-tsukuba/AdvancedCourse2021>
- 1限目の資料
  - <https://github.com/bioinfo-tsukuba/AdvancedCourse2021/tree/main/1>

# 1限の内容

- 講義
  - Pythonについて
    - プログラミング言語
    - 実行環境
  - Google Colaboratory (Colab)
    - 概要
    - ログイン方法
    - ノートブック
    - Google Driveのマウント
  - Markdownについて
- 実習

# プログラミング言語

- コンピュータに計算(仕事)を行わせるプログラムを書くための人工言語
  - 例: C, C++, C#, Java, Perl, Python, Ruby, R, Julia, ...
- 機械学習やデータサイエンスではPythonが使用されることが多い
  - 統計解析の分野ではRも人気

# なぜPythonなのか

- ライブラリが充実している
  - 機械学習: TensorFlow, PyTorch
  - データ分析: pandas
  - 数値計算: NumPy, SciPy
  - ...
- インタプリタ型言語である
  - 書いたプログラムをすぐ実行可能

# 実行環境(1/3)

- REPL (Read-Eval-Print Loop)
  - 対話的にコードを実行できる環境
    - (ユーザーが書いた)コードを読み(Read)
    - 読んだコードを実行(Eval)し
    - 実行した結果を表示(Print)する
    - 上記を繰り返す(Loop)
  - 大きなプログラムを書いたり実行することには向かない
    - あくまでコード片の実行に留める
- (REPL実行デモ)



# 実行環境(2/3)

- テキストエディタとインタプリタ
  - テキストエディタでコードを書き、インタプリタで書いたコードを実行する
- テキストエディタ
  - VSCode, Vim, Emacs, ...
  - プログラミング言語での開発を想定して作られている
    - 構文の強調(シンタックスハイライト)、入力補完、...
- (デモ)
  - テキストエディタとインタプリタを用いた実行
  - コンパイル型言語との違い(Python vs C++)

# 実行環境(3/3)

- Jupyter notebook
  - コード片と文章を「ノートブック」にまとめることができる
    - 実行単位がコードブロックになる
      - デバッグが容易
      - 開発効率向上
  - 予めインスタンスを自分のPCなどに立てておき、ブラウザで接続して使用する
- (実行デモ)

# Google Colaboratory (Colab)

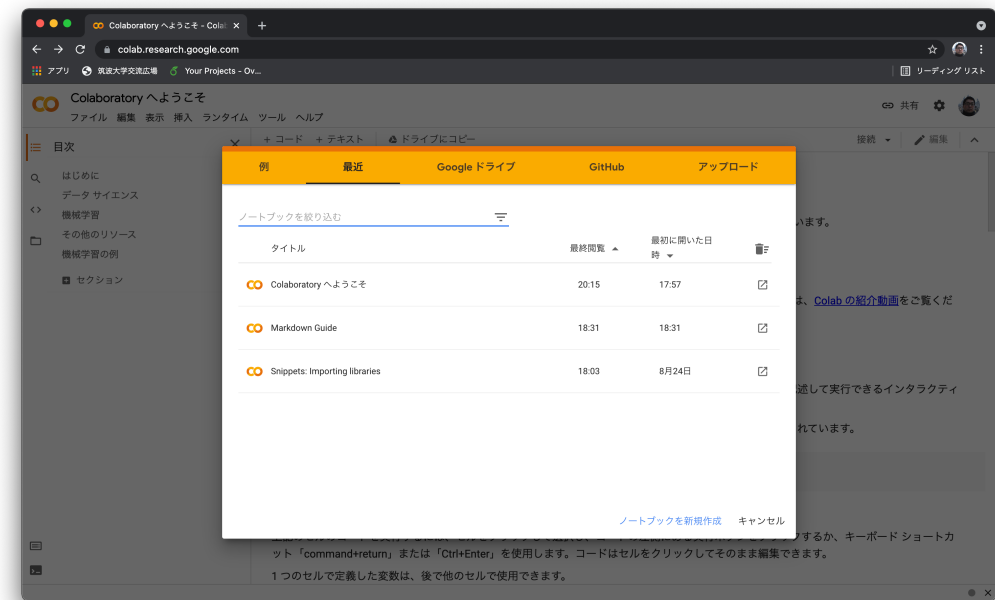
- (簡単に言うと)Jupyter notebookをセットアップすることなく使用することができるGoogleのサービス
- メリット
  - Jupyter notebookのセットアップが不要
  - Googleの各サービスとの連携
    - ドライブ、スプレッドシート、...

# アクセス・ログイン

- ブラウザで以下のURLへアクセス

<https://colab.research.google.com/>

- 右上のログインをクリックしてGoogleアカウントでログイン(右上)
  - 既にGoogleアカウントでログインしている(Chromeの)場合、ログインの操作は不要(右下)



# ノートブックの構成

ノートブックは「コードセル」と「テキストセル」から構成される。

- コードセル
  - プログラムのコード片
  - コード片の実行結果
- テキストセル
  - コード片の説明などのテキスト
  - Markdownで装飾等が可能(後述)

## テキストセル

### Colaboratory とは

Colaboratory (略称: Colab) は、ブラウザから Python を記述、実行できるサービスです。次の特長を備えています。

- 環境構築が不要
- GPU への無料アクセス
- 簡単に共有

Colab は、学生からデータサイエンティスト、AI リサーチャーまで、皆さんの作業を効率化します。詳しくは、[Colab の紹介動画](#)をご覧ください。下のリンクからすぐに試してみることもできます。

### はじめに

ご覧になっているこのドキュメントは静的なウェブページではなく、**Colab ノートブック**という、コードを記述して実行できるインタラクティブな環境です。

たとえば次の**コードセル**には、値を計算して変数に保存し、結果を出力する短い Python スクリプトが記述されています。

```
[ ] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day

86400
```

コード片

実行結果

上記のセルのコードを実行するには、セルをクリックして選択し、コードの左側にある実行ボタンをクリックするか、キーボードショートカット「command+return」または「Ctrl+Enter」を使用します。コードはセルをクリックしてそのまま編集できます。

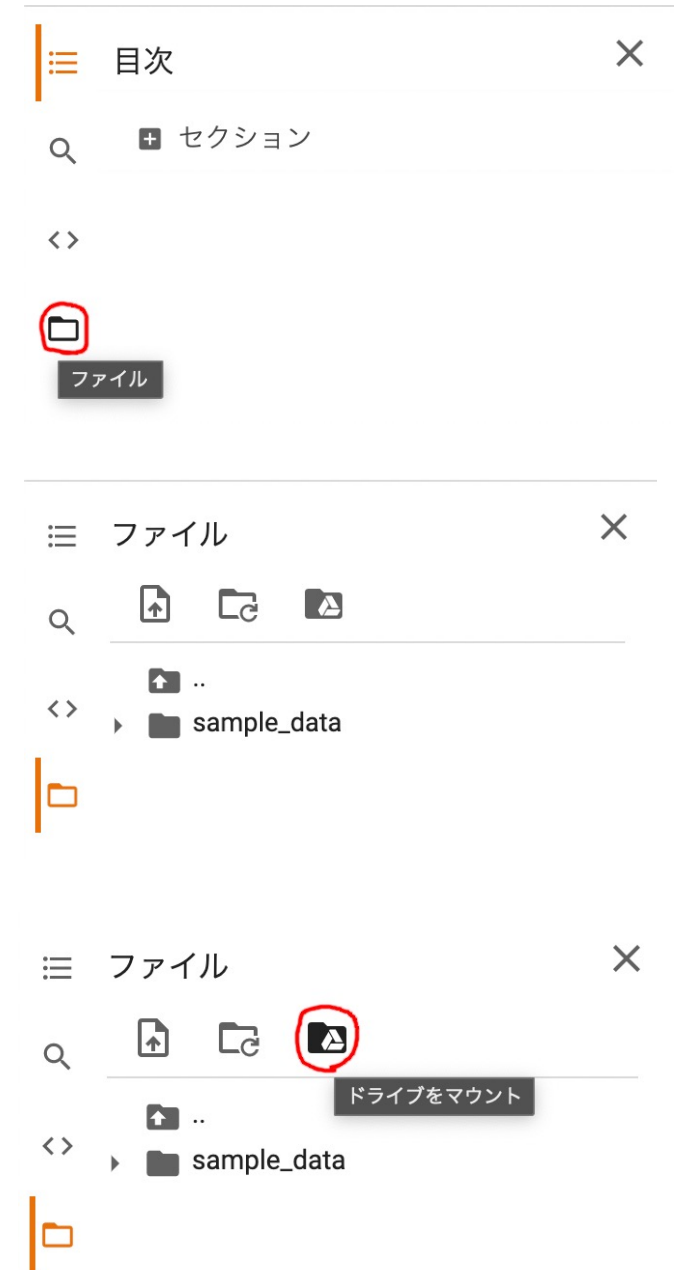
1 つのセルで定義した変数は、後で他のセルで使用できます。

# Google Driveのマウント(1/4)

Google Drive上のファイルをマウントして使用することができます

## 手順

1. マウントしたいノートブックを開いた状態で、左側のフォルダのマークをクリックする(右上)
2. インスタンスに接続されていない場合は数秒待つ(右中)
3. 上部にある「ドライブをマウント」(3つあるアイコンの一番右)をクリックする(右下)



# Google Driveのマウント(2/4)

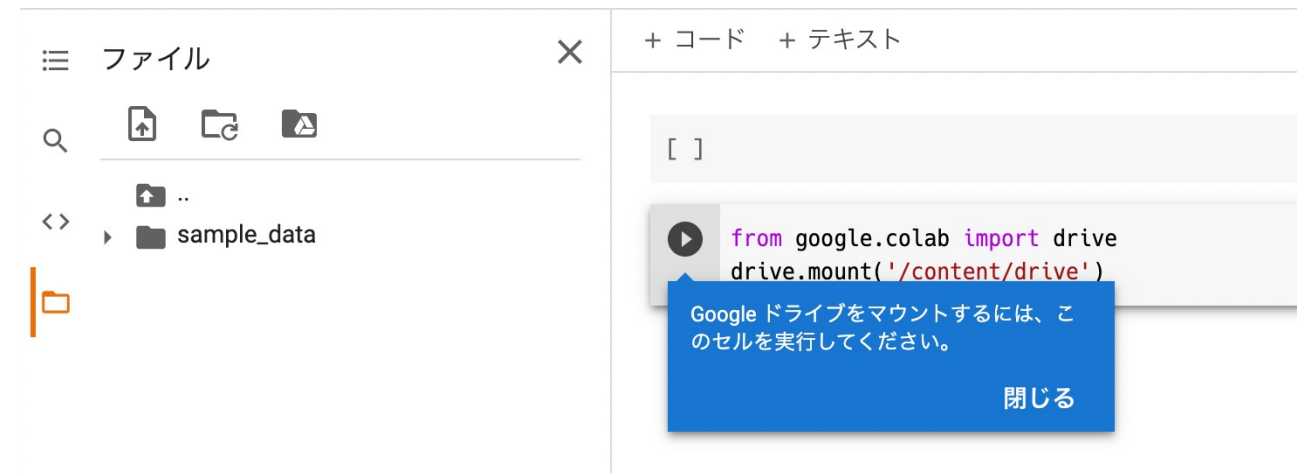
4. 「スキップ」を選択する  
(右上)

※2回目以降は

「Google ドライブに接続」  
でOK(のはず)

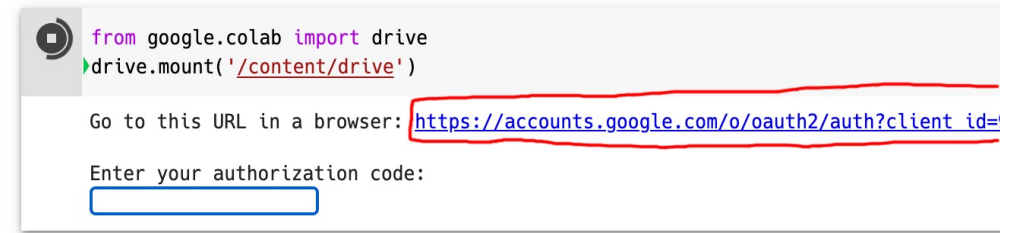


5. マウントするためのコード  
片が出てくるので、ポップ  
アップを閉じて実行する(右  
下)



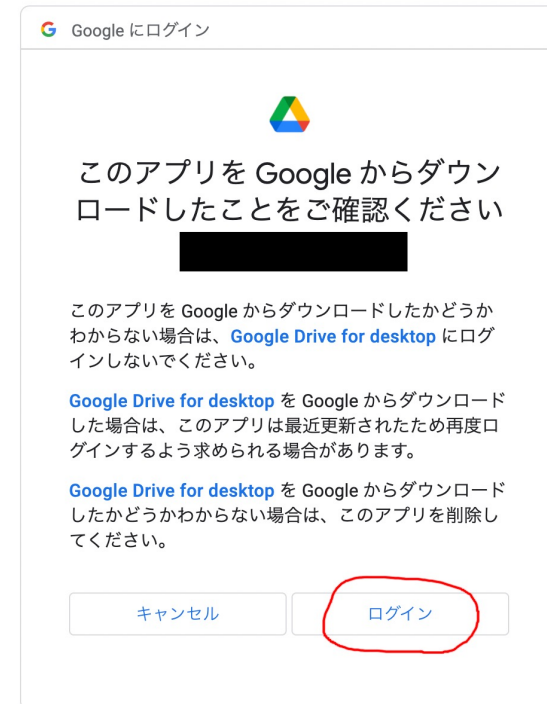
# Google Driveのマウント(3/4)

6. 認証のためのURLが出てくるのでクリックする(右上)



7. 別窓が開いたら今ログインしているGoogleアカウントを選択する

8. Google Driveの認可画面が出てくるので「ログイン」をクリックする(右下)

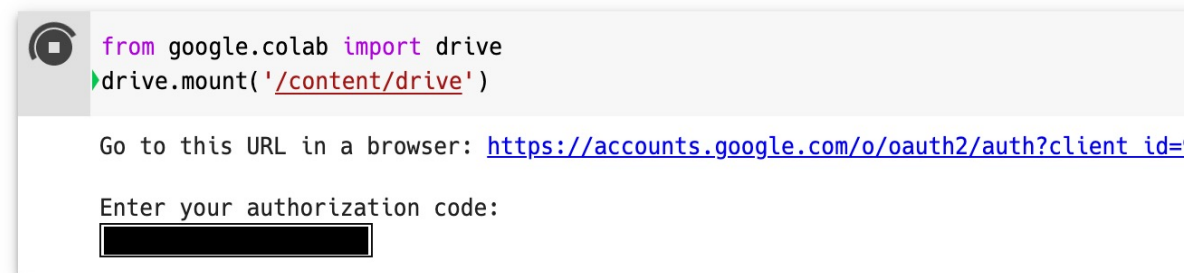




# Google Driveのマウント(4/4)

9. 出てくる認証コードを元のウィンドウのコード片のテキストボックスにペーストしてEnterキーを押す(左→右上)

10. Mounted at /content/drive と表示されたらOK(右下)



```
from google.colab import drive
drive.mount('/content/drive')
```

Go to this URL in a browser: [https://accounts.google.com/o/oauth2/auth?client\\_id=](https://accounts.google.com/o/oauth2/auth?client_id=)

Enter your authorization code:

[Redacted code input field]



```
from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

# Markdown

- マークアップ言語の一種
    - 視覚表現や文章構造を指定する
    - 他のマークアップ言語の例: HTML, XML, TeX, MathML, SVG, ...
  - リッチな表現
    - 太字、斜体、下線、取り消し線、リンクなどのインライン要素
    - セクション、表、箇条書きなどブロックレベル要素
- 素のテキストよりわかりやすいノートブックに

実例はColabのトップページや実習のノートブックなどを参考にする。

# 実習準備: ノートブックのコピー

- (デモで実演)
- 参考: 去年度の講義の1限目の資料
  - [https://github.com/bioinfo-tsukuba/AdvancedCourse2020/blob/master/01/01\\_%E3%82%B7%E3%82%B9%E3%83%86%E3%83%A0%E5%85%A5%E9%96%80%EF%BC%88%Ef%BC%91%Ef%BC%89.pdf](https://github.com/bioinfo-tsukuba/AdvancedCourse2020/blob/master/01/01_%E3%82%B7%E3%82%B9%E3%83%86%E3%83%A0%E5%85%A5%E9%96%80%EF%BC%88%Ef%BC%91%Ef%BC%89.pdf)
  - 『GitHubにあるノートブックをコピーする』
    - 枠に入力するリポジトリを今年の「bioinfo-tsukuba/AdvancedCourse2021」に変更する