ae-8.Python と Google Colaboratoryの活用

(AI 演習) (全15回)

https://www.kkaneko.jp/ai/ae/index.html

金子邦彦





Google Colaboratoryで段階的に学ぶプログラ ミングで実践力向上

② Pythonの多様な機能習得でデータ分析からAI 開発まで視野拡大





- 1. 実践重視: AIツールを実際に使用し、機能に慣れる
- 2. **エラーを恐れない**:実行においては、エラーの発生の可能性がある.エラーを恐れず,むしろ学習の一部として 捉えるポジティブさが大切.
- 3. 段階的学習:基礎から応用へと段階的に学習を進め, AI の可能性を前向きに捉える



アウトライン

- 1. Google Colaboratory の基礎
- 2. Google アカウントの取得
- 3. Python プログラミングの基礎
- 4. Python の Pandas データフレー ム
- 5. Python の主な機能

8-1. Google Colaboratory の基礎

Google Colaboratory (Colab)





- ・Google が提供
- ・クラウドベースの Python ノートブック(プログラム,実行結果,実行結果,説明文,グラフなどを1つにまとめたもの)
- Pythonプログラミング(特に、人工知能、データサイエンス)の開発に適する
- ・本格的な利用(ノートブックの新規作成、コードセル・テキス トセルの追加,編集,実行)には,Google アカウントが必要

Google Colaboratory の全体画面





Web ブラウザの画面





- コードセル, テキストセルの2種類
- ・**コードセル**: Python プログラム, コマンド, 実行結果
- ・テキストセル:説明文,図

2種類のセルを組み合わせて使用



Google Colaboratory の基本的な使い方①



① ノートブックの新規作成

1. Web ブラウザでアクセス

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

2. 「**ログイン**」をクリック. Google アカウントでログイン



Google Colaboratory の基本的な使い方②



② 他の人のノートブックの利用

1. Web ブラウザでアクセス. 共有用のリンクを使用

例:<u>https://colab.research.google.com/drive/1MMhlrh08-0Byq-</u> <u>U1JITdDVwMe6dyUcaq?usp=drive_link</u> ※ リンクはノートブックごとに異なる.

2. ノートブックが表示される.



中身を編集したり、再実行するには、Google アカウン
 トでのログインが必要









コードセルの実行

files = ['a.png', 'b.png','c.png', '126.png', '127.png']

• 6. 顏検出

O

import sys import dlib import os import urllib.request import cy2

for f in files:

おけ

願検出は、写真やビデオの中の顔を検出すること、顔とそれ以外のオブジェクトを区別することも行う、顔検出の結果は、バウ ンディングボックスで得られるのが普通である.

次のプログラムは、Dlibを用いて、画像からの顔検出を行う。

- ・「dets = cnn_face_detector(img, 6)」・・・顔検出の実行
- ・「cv2.rectangle(disp, (d.rect.left(), d.rect.top()), (d.rect.right(), d.rect.bottom()), (255, 0, 0), 1)」・・・顔検出の結果を四 角形で表示

結果は、赤い四角で表示される.1,3,4,5番目の画像(a.png.c.png.126.png.127.png)からは、顔が検出される.2番目の画像 (b.png,手で顔を覆い隠したもの)からは顔が検出されない.少し隠れていたり、顔が傾いていても顔検出ができるが、大きく隠れていると 顔検出できない.

実行結果が長いので,スクロールして全体を確認すること.

import matplotlib.pyplot as plt

print("*** file: {} ***".format(f))

img = dlib.load_rgb_image(f)

謝辞: ここの Python プログラムは、Dlib に付属の cnn_face_detector.py を書き換えて使用している

cnn_face_detector = dlib.cnn_face_detection_model_v1('mmod_human_face_detector.dat')

コードセル

テキストセル

コードセル

C+ *** file: a.png ***

Number of faces detected: 1

Humber of faces detected: 0

コードセルはPython プログ ラム.

 各自の Google アカウントで ログインすれば、セルの変 更、コードセルの実行可能









編集前

編集後







- ・コードセルの追加:「+コード」
- ・テキストセルの追加: 「+ テキスト」







- コードセルの実行:実行ボタン(または Shift + Enter)
- ・コードセルの追加:「+ コード」
- ・テキストセルの追加: 「+ テキスト」

特徴的な機能



- CPU, GPU, TPU の利用が可能
 - ・「ランタイム」→「ランタイムのタイプを変更」で設定

| | ファイル 編集 表示 挿入 | ランタイム ツール ヘルプ | |
|----------------|---------------|----------------|------------------|
| := | 目次 | すべてのセルを実行 | Ctrl+F9 |
| ~ | | | |
| Q | はじめに | | |
| $\{x\}$ | データ サイエンス | | |
| С л | 機械学習 | | |
| ~ | その他のリソース | | |
| | 使用例 | | |
| | | | |
| | + セクション | ランタイムを接続解除して削除 | |
| | | ランタイムのタイプを変更 | $\sum_{i=1}^{n}$ |
| | | セッションの管理 | |
| | | リソースを表示 | |

| C | Colaboratory へようこそ | | | | |
|---------------|--------------------|----------------|---------|--|--|
| ~ | ファイル 編集 表示 挿入 | ランタイム ツール ヘルプ | | | |
| :≡ | 目次 | すべてのセルを実行 | Ctrl+F9 | | |
| | | | | | |
| Q | はじめに | | | | |
| { <i>x</i> } | データ サイエンス | | | | |
| 0 | 機械学習 | | | | |
| ~ | その他のリソース | | | | |
| | 使用例 | | | | |
| | | | | | |
| | + セクション | ランタイムを接続解除して削除 | | | |
| | | ランタイムのタイプを変更 | \Im | | |
| | | セッションの管理 | | | |
| | | リソースを表示 | | | |

CPU: コンピュータの頭脳. プログラム実行を行う.
GPU: 並列処理に特化した処理装置. 画像処理や AI の計算 を高速で実行.
TPU: Google製の AI専用プロセッサ.
※ GPU, TPU は, 使用制限あり(制限の緩和は有料版)

Google Colaboratory でうまく実行できない 場合

混雑しているときなどは、実行が止まり、再開しない(混 雑時の切断)場合もある

【その対処】

次の手順で,アクティブなセッションの停止を行い,その後 最初から実行をやり直す

- ・メニューで「ランタイム」, 「セッションの管理」と操作 する.
- ・アクティブなセッションの一覧が表示されるので、「終 了」をクリックして、すべてのアクティブなセッションを 終了する.

無料版と有料版の違い



| | 無料版 | 有料版 |
|--------------|----------|------------------------|
| 基本機能 | ОК | ОК |
| GPU, TPU の利用 | 使用制限あり | 使用制限は緩和 |
| 混雑時の切断 | 切断の可能性あり | 切断の可能性あり(可 能性はより低い) |
| RAM | 12.7 GB | 25 GB以上 |
| ディスク容量 | 107.7 GB | 178 GB以上 |

この授業では無料版の利用





- ・ 共同作業: ノートブックの共有, リアルタイムの共同編集
- Google ドライブ連携
- ファイルのアップロード、ダウンロード
- システムコマンド (!ls, !pwd, %cd など)

Google Colaboratory まとめ



- Google Colaboratory: クラウドベースのPythonノート
 ブック環境、AIやデータサイエンス開発に適する
- ・ノートブック:プログラム、実行結果、説明文を1つにま とめた作業環境
- ・コードセル:Pythonプログラムを記述・実行できるノート ブックの要素
- ・**テキストセル:説明文や図を記述**できるノートブックの構成要素
- ・GPU:画像処理やAI計算に特化した処理装置
- ・セッション管理: Colaboratoryの実行環境を制御する機能. 混雑時に使用(混雑時の切断では,アクティブなセッションの停止を行い,その後最初から実行をやり直す)

8-2. Google アカウント の取得





- Googleアカウント: GoogleのさまざまなWebサービス
 (Gmail、YouTube、Googleドライブなど)を利用するた
 めの個人アカウント
- ・メールアドレスは後から変更できないので、慎重に決める





- ・Google公式サイトにアクセスし「**アカウントを作成**」を選 択
- ・以下の情報を入力
 - 姓名
 - 希望するGmailアドレス
 - パスワード (8文字以上)
 - 生年月日
 - 電話番号(本人確認用)
- •利用規約とプライバシーポリシーに同意
- 確認コードを受け取り、入力して完了

各自でよく確認し、心配なことがある場合には取りやめて ください



① Google の Web ページ

を開く

https://www.google.com/





Gmail

画像

ログイン

③「**アカウントを作成す** る」をクリック



メールアドレスまたは電話番号

メールアドレスを忘れた場合

ご自分のパソコンでない場合は、InPrivate ウィンドウ を使用してログインしてください。 詳細





④ 右のような画面が出たとき
は「**自分用**」を選ぶ
※「ビジネスの管理用」は
Google の有料サービス
G Suite のこと(関係ない)



Google

Google アカウントの作成

YouTubeに移動

| ^y 王 金子 | 邦彦 | | | |
|----------------------------|------------------|--|--|--|
| メールアドレス | | | | |
| kaneko@fukuyama-u.ac.jp | | | | |
| | | | | |
| なります。 | すりていてのることの確認が必要に | | | |
| 代わりに Gmail アカウントを作成 | | | | |
| パスワード | 確認 | | | |
| ••••• | Ø | | | |
| | | | | |

代わりにログイン

277



⑤ 次の情報を登録する 姓,名 自分が希望するメールアドレス <ユーザー名> @gmail.com パスワード(2か所), 電話番号 生年月日,性別



Google

電話番号の確認

セキュリティ保護のため、本人であることを Google が確認する必要があります。6 桁の確認コードが記載 されているテキスト メッセージが Google から送信さ れます。



⑥ 本人確認のための電話番号 を入れ、「次へ」をクリック. ※ 電話番号は、スマートフォンの電話番号が便利

⑦その後、電話での指示に従う



Goodle の Web サービス

- Gmail
- Google Colaboratory
- Googleドライブ (15GB無料)
- Googleカレンダー
- YouTube
- Google Maps
- Google フォト
- Google ドキュメント, Google スプレッドシート, Google スライド, Google フォーム, Goolge Jamboard

Google アカウントまとめ



- Googleアカウント: Googleサービスを利用するための個
 人用アカウント. Google Colaboratory の利用でも有用
- ・Gmailアドレス:一度設定すると変更不可能
- Google アカウント作成での本人確認:電話番号を用いて 実在する人物かを確認





- Googleサービスはクラウドサービス.他の人が管理する
 サービスなので、秘密にしたい情報を扱わないことを推奨
- ・ID, パスワードは厳重管理. 貸し借りをしてはいけない.

この授業では, Google アカウントの取得を強制しないこと にしています.

Google アカウント取得者: 発展学習が一部可能





発展演習

これは、Google Colaboratory の ノートブックを作成する演習です。

Google アカウントの保持者の自 主的学習とします



① Google Colaboratory のWebページを開く https://colab.research.google.com

| CC | Colaboratory へようこそ ファイル 編集 表示 挿入 ランタイム | > ツール ヘルプ | ④ 共有 | ¢ | ログイン |
|----------------|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ≔ | 目次 × | + コード + テキスト 🍐 ドライブにコピー | 接続 🔻 | - / | ▲ 編集 ▲ |
| Q <> {x} | はじめに データ サイエンス 機械学習 その他のリソース | 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。 | | ∍ 🖍 | [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] |
| | 機械学習の例 ■ セクション | ・ 環境構築か不要 ・ 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の紹介動画をご覧ください。下のリンクからすぐに使ってみることもできます。 | 率化します。 | 詳しく | は、 <u>Colab</u> |
| - (- i | | 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 行できるインタラクティブな環境です。 | 1という、コー | -ドを | 記述して実 |
| | | 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 す。 | スクリプトカ | が記述さ | されていま |



② 「**ファイル**」で, 「ノートブックを新規作成」を選ぶ



③ Google アカウントでのログインが求められたときはログ インする

| Google へのログインが必要 | | |
|--------------------------------------|------|--|
| 続行するには、Google アカウントにログインしている必要があります。 | | |
| | ログイン | |



④コードセルの新規作成

PRO ファイル 編集 表示 挿入 ランタイム ツール ヘルプ メニューの「+コード」 ★ニューの「+コード」 ★ たクリック

⑤ コードセルに Python プログラムを入れる x = 100

x = 100



⑥ コードセルの新規作成

PRO ファイル 編集 表示 挿入 ランタイム ツール ヘルプ + コード + テキスト ▲ドライブにコピー ト ト をクリック

⑦ コードセルに Python プログラムを入れる





⑧ コードセルの新規作成

PRO ファイル 編集 表示 挿入 ランタイム ツール ヘルプ + コード + テキスト ▲ドライブにコピー ト ト をクリック

⑨ コードセルに Python プログラムを入れる

s = 0

for i in [1, 2, 3, 4, 5]: for i in [1, 2, 3, 4, 5] の直後に「:」

s = s + i

print(s)





⑩ コードセルを上から順に実行し,結果を確認



8-3. Python プログラミン グの基礎

コンピュータとプログラムの関係

- ・**コンピュータ**は, **プログラム**に 従って動作
- ・プログラムは、コンピュータに指 示を出し,所定の作業を遂行させ る
- この関係が, コンピュータシステム の基礎

100%" y1="0%" x2=" "#06101F" offset= "#1D304B" offset=

"450" rx="8" fill="

ght="96" viewBox="0 0 🤧

dient x1="87.565%" y1="

stop-color="#FFF" offset

-500%" y="-500%" width

ianBlur stdDeviation

et dy="16" in="5

atrix values=

adient>

stop-color="#FFF" stop-opac.



① **プログラム**の内容によって, **コ** ンピュータは<u>さまざまな作業を実行</u> できる

② **プログラム**を利用することで, 多くの作業を<u>自動化</u>できる

③ **プログラム**で行った作業を<u>いつ</u> <u>でも再現できる</u>.

④ **プログラム**は柔軟性があり,変 更により, <u>プログラムの動作を簡単</u> <u>に調整できる</u>







・Web開発

フロントエンド(HTML, CSS, JavaScript), バックエンド (Python, Django, Flask)

・データ分析

ビッグデータ処理,統計分析,データビジュアライゼーション (データの可視化)

・人工知能

自然言語処理, コンピュータビジョン(画像認識技術), 予測モ デリング

・ゲーム開発

2Dゲーム, 3Dゲーム, モバイルゲーム

IoT (Internet of Things)

センサーデータの収集と分析, スマートホームシステム

・サイバーセキュリティ

ネットワークセキュリティ, 暗号化技術

Python プログラミングの基礎の全体像



- ・プログラミングは、プログラムを作成する創造的な活動
- **プログラム**は, **コンピュータ**に指示を出し,所定の作 業を遂行させる
- Python は, 読みやすさ, 書きやすさ, 幅広い応用範 囲が特徴
 - a = [200, 400, 300] for i in a: print (i * 1.08)

Python 言語のプログラム

216.0 432.0 324.0

> **プログラム**の 実行結果





- ・変数:プログラム内で名前を付けて利用するオブジェ **クト**である。**値を保存**し、後から参照できる仕組み。
- ・代入:「<u>x = 100</u>」のように書くことで,<u>x という名前</u> **の変数に、値 100 が保存**される操作

$$\mathbf{x} = \mathbf{100}$$

- 保存された値は、プログラムの中で何度でも参照 することが可能
- 別の値で上書きすることも可能。





・オブジェクト:コンピュータでの操作や処理の対象と なるもの

| t.goto(0,100) | t オブジェクト | |
|---------------|-----------------------|--|
| | goto(0,100) メソッド | |
| | 間を「.」で 区切って いる | |

- ・メソッド: オブジェクトに属する機能や操作. オブジェ クトがもつ能力に相当.
- ・引数:メソッドが行う操作の詳細に関する情報.メ ソッド呼び出しのときに、引数を指定できる

例 t.goto(0,100)

引数は0と100

Python プログラムの書き方(コーディングス タイル)

- ・インデント(字下げ)によるブロックの区切り (通常「タブ」または「4つの半角スペース」)
- **コメント**:行の先頭に「#」.プログラムの説明を記述
- **空白行**: プログラムを読みやすくするために挿入可能

```
import turtle
```

例

```
for i in range(4):
turtle.forward(100)
turtle.right(90)
```

```
turtle.done()
```

空白行 } インデント(字下げ) 空白行



プログラミング学習の意義と目標

- ・論理的思考力や課題解決力の向上
- ・プログラムの作成と実行を通じて, コンピュータの動作原理を学び, 問題解決能力を高める
- ・自分の作ったプログラムが動作す る喜びと達成感







① プログラミングの基本概念の理解

② 創造的なプログラム作成の態度. 習慣

③ 問題解決能力

④自主的な学習態度と探求心

8-4. Python の Pandas データフレーム

Python の Pandas データフレーム



表形式のデータ

| | X | y | | |
|-------|---|---|--|--|
| 0 | 1 | 4 | | |
| 1 | 1 | 2 | | |
| 2 | 1 | 5 | | |
| 3 | 2 | 4 | | |
| 4 | 3 | 5 | | |
| 5 | 3 | 3 | | |
| | | | | |
| データ本体 | | | | |

Pythonの Pandas データフレームを組み立てる ログラム



データフレームの組み立て,結果は df に代入

df = pd.DataFrame([[1, 4], [1, 2], [1, 5], [2, 4], [3, 5], [3, 3]], columns=['x', 'y'])





- ・データ分析
- ・**データクリーニング**: 欠損データの処理や、データの変 換・正規化
- ・時系列データ:時間属性を含むデータを扱う機能
- データの統合: データの結合、マージ、グループ化

【柔軟性】

様々なデータ形式(CSV, Excel, リレーショナルデータベー ス等)をサポート

①簡単な散布図

必要なライブラリをインポート from sklearn.datasets import load_iris import matplotlib.pyplot as plt import pandas as pd

#新しいXとyの値 X = [1, 2, 3, 4, 5] y = [3, 4, 1, 3, 4]

Matplotlibを用いて結果をプロ、 plt.scatter(X, y, color='blue')

軸ラベルとタイトルを追加 plt.xlabel('X-axis') plt.ylabel('Y-axis') plt.title('Custom Data Plot')

グラフを表示 plt.show()









必要なライブラリをインポート from sklearn.datasets import load_iris import matplotlib.pyplot as plt import pandas as pd

irisデータセットをロード (pandas を使用) iris = load_iris() df = pd.DataFrame(iris.data, columns=iris.feature_names)

データの先頭と末尾を確認 print("データの先頭:") print(df.head()) print("データの末尾:") print(df.tail())

必要な列だけを選択 (DataFrame形式でスライス)
X = df[['sepal length (cm)']]
y = df['sepal width (cm)']

Matplotlibを用いて結果をプロット plt.scatter(X, y, color='blue')

軸ラベルとタイトルを追加 plt.xlabel('Sepal Length') plt.ylabel('Sepal Width') plt.title('Iris Dataset')

グラフを表示 plt.show()



53





・機械学習のうち1つ「**回帰**」を行う。ここでは、 データに最もよく適合する線を求めている

必要なライブラリをインポート

from sklearn.linear model import LinearRegression from sklearn.datasets import load iris import matplotlib.pyplot as plt import pandas as pd

irisデータセットをロード (pandas を使用) iris = load iris() df = pd.DataFrame(iris.data, columns=iris.feature names)

データの先頭と末尾を確認 print("データの先頭:") print(df.head()) print("データの末尾:") print(df.tail())

必要な列だけを選択(DataFrame形式でスライス) X = df[['sepal length (cm)']] y = df['sepal width (cm)']

#線形回帰モデル model = LinearRegression()

学習 model.fit(X, y)

#予測を行う y pred = model.predict(X)

Matplotlibを用いて結果をプロット plt.scatter(X, y, color='blue') plt.plot(X, y pred, color='red')

軸ラベルとタイトルを追加 plt.xlabel('Sepal Length') plt.ylabel('Sepal Width') plt.title('Linear Regression on Iris Dataset')



plt.show()



8-5. Python の主な機能

Pythonプログラミングの基礎から応用 - 19の実践の 的な機能①

【基本的な概念と操作】

1. プログラミングは「コンピュータへの指示書」として機能し、複雑な 作業を自動化できる

《プログラム》**小計 = 商品価格 * 個数**→計算の自動化

2. データ型の柔軟性: 文字列、数値(整数・小数)、真偽値など、様々 な種類のデータを扱える

《プログラム》名前 = "山田太郎" →文字列の使用

3. True/Falseを使った論理演算により、複雑な条件判断が可能

《プログラム》合格者数 = (scores >= 80).sum()

→点数が80以上のデータ数を数えるものだけの合計

4. f文字列(変数を文字列に埋め込む方法)を使用することで、テキストと 変数を組み合わせた出力が簡単に

《プログラム》print(f"名前: {名前}")→f文字列による変数の埋め込み例

Pythonプログラミングの基礎から応用 - 19の実践 的な機能②

【基本的なデータ構造とフロー制御】

5. リストを使用することで、複数のデータをまとめて効率的に管理・処理が可能

《プログラム》**果物 = ["りんご", "みかん"]** →リストによるデータのま とめ

6. 条件分岐(if文)により、状況に応じて異なる処理を自動的に実行で きる

《プログラム》if 点数 >= 80: →点数による合否判定

7. for文による繰り返し処理で、大量のデータを効率的に処理することが 可能

《プログラム》for 果物 in 果物リスト:→リストの要素を順番に処理

【プログラムの構造化と信頼性】

8. 関数を作成することで、複雑な処理を再利用可能な形でまとめることができる

《プログラム》def 挨拶する(名前):→関数定義と引数の例

Pythonプログラミングの基礎から応用 - 19の実践の 的な機能③

【データ処理の基本】

9. データの自動計算:売上計算など、数値データの処理を自動化できる

《プログラム》合計 = 小計 + 消費税 →売上計算の自動化

10. 数値の統計分析:平均、最大値、最低値などを簡単に計算可能

《プログラム》最高点 = max(点数リスト) →最高点を求める統計計算の 例

11. pandas(表計算用のデータ分析ツール)ライブラリを使用して、Excelのような表形式のデータ(DataFrame)を効率的に処理・分析できる

《プログラム》df = pd.DataFrame(data) → pandas表データ作成例

12. データの並び替えや絞り込みを、簡単に実現できる

《プログラム》sort_values('売上') →pandasによるデータ並び替えの例

Pythonプログラミングの基礎から応用 - 19の実践 的な機能④

【高度なデータ処理と分析】

13. groupby(データをグループ分けする機能)による集計で、カテゴリご との分析が容易

《プログラム》groupby('日付').agg →日付ごとの集計

14. rolling(連続するデータの計算方法)による平均の計算など、時系列 データの分析が可能

《プログラム》rolling(3).mean()→3日間の平均計算

15. 複数のデータを比較しグラフ化することで、傾向分析が容易に

《プログラム》daily_summary['売上', ['sum', 'mean']]

→日々の売上合計と平均を並べて比較

Pythonプログラミングの基礎から応用 - 19の実践 的な機能⑤

【高度な応用】

- 16. データの可視化: plt(グラフ描画ツール)を使って、グラフや図 表でデータを分かりやすく表現することが可能
 - 《プログラム》plt.plot(months, coffee) →折れ線グラフ作成

17. 商品シェアの計算やパーセンテージ分析など、ビジネス分析 に活用できる

《プログラム》plt.pie(sizes, labels) →円グラフによる比率表示

18. スクレイピング(Webページから情報を取得すること)機能により、Webページから自動的にデータを収集できる。soup(HTMLを解析するツール)を使用

《プログラム》title = soup.find('title').text →Webページのタイトルを自動で取得

19. インポート(外部のプログラムを取り込むこと)とfromによる外部ライブラリの取り込みで、プログラムの機能を拡張できる

《プログラム》import pandas as pd →外部ライブラリのイン ポート

Python の主なキーワード



- ・import, from: パッケージ(外部ライブラリなど)のイン ポート
- def: 関数定義
- ・return: 関数の戻り値
- if, else, elif: 条件分岐
- for, in: 繰り返し(in は繰り返し範囲)
- +, -, , /, ==, !=: 基本演算子
- True, False

すべてを一度に覚えるということではなく,基礎から応用へと段階的 に学習を進めましょう





演習 Python のプログラム例と 説明

① Google Colaboratory のページを開く



https://colab.research.google.com/drive/16epaQjvr38PJbblVff15e8eAHNB0WE0 Y?usp=sharing

② プログラムや説明や実行結果をよく読む 【**目的】**

- ・Python の多様な機能と可能性を知る
- ・「19の実践的な機能」を網羅した説明である





- Google Colaboratory : クラウドベースのPythonノート ブック環境で、GPUも使えて無料。プログラム、実行結果、 説明文を1つにまとめて開発できる
- Pandas データフレーム:表形式データを効率的に処理で きるPythonの機能。データの分析、統合などを柔軟に行え る
- ・プログラミングの段階的学習法:基礎から応用へと段階を 追って進み、エラーを恐れずに実践を重ねることで、確実 にスキルを習得できる
- Python の実践的機能:データ分析、機械学習、Web開発 など、現代のIT開発に必要な Python の実践的な機能

Google Colaboratoryで段階的に学ぶプログラント シングで実践力向上

Google Colaboratoryでは、クラウドベースの 環境で実際にPythonプログラムを動かしながら 学習できます。エラーを恐れずに試行錯誤し、 基礎から応用へと段階的に進める学習方針によ り、確実なスキル向上が実現できます。実践的 なAI開発スキルにもつながります。

② Pythonの多様な機能習得でデータ分析からAI 開発まで視野拡大

Pythonには、Pandasによるデータフレーム処理、グラフ作成、Webスクレイピングなど、現代のIT開発に必要な機能があります。これにより、Web開発、データ分析、AI開発、IoTなど、多様な応用分野に対応できる技術者としての視野が広がります。