

人工知能(Python言語)講座

(プログラム言語初心者)

校舎 AIP College

東京都新宿区西新宿 2 丁目 6 - 1

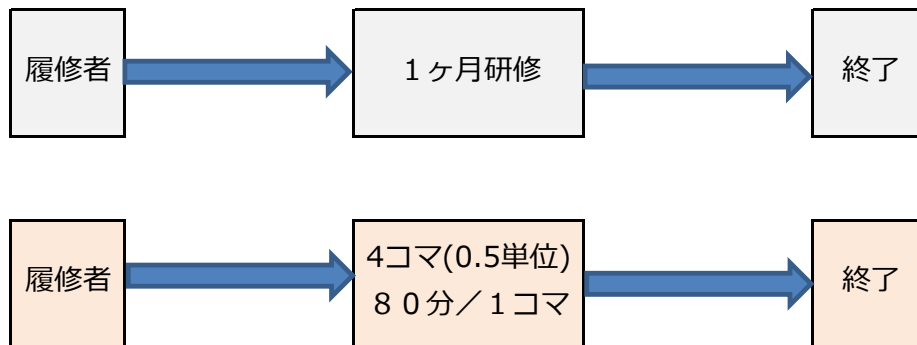
新宿住友ビルディング 19F

TEL 0800-800-0815

<https://aipcollege.com/>

Pythong言語講座

研修 :



履修経験:

(1) Lips、C言語
(2-1) C++, C#, Java
(2-2) VB, Php, Ruby
(3) Python

研修 :

研修の進め方

Python 計算時間 (PythonNumpyCalTime.py)

```
コマンドプロンプト
C:\PythonPG>python PythonNumpyCalTime.py

----- Numpy Python 計算速度 -----
----- (20*100000000) 個の積と加算 -----

-----Python Start          === 23-12-30
160
-----Python End & Numpy Start === 23-43-22

[[8]]
[[8]]
-----Numpy End            === 23-43-24

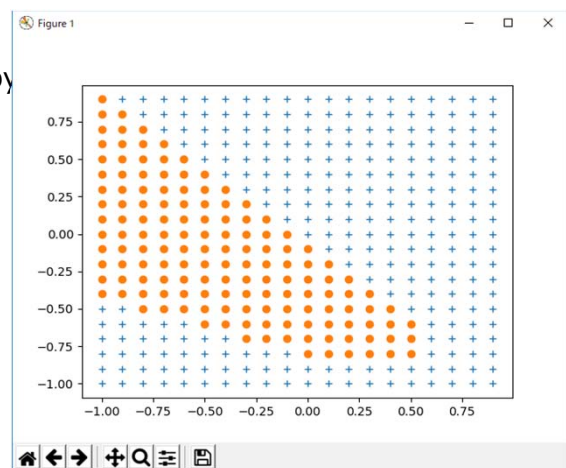
C:\PythonPG>
```

```
AAA.java
21  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
22  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
23  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
24  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
25  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
26  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
27
28  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
29  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
30  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
31  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
32  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
33  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }
34  for (int nn=0; nn<100000000; nn++){ y += X[nn] * W[nn]; }

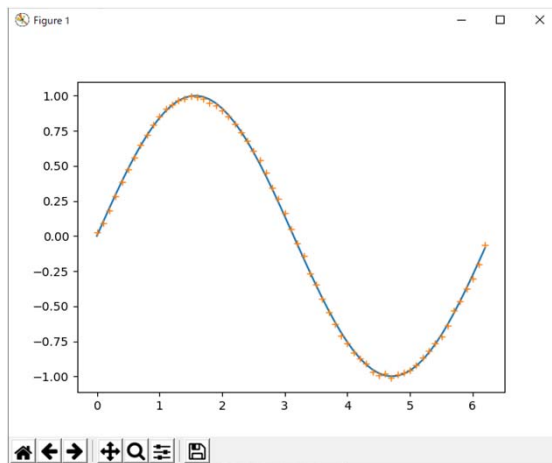
<終了> AAA [Java アプリケーション] C:\pleiades4.4\java\bin\javaw.exe (2019/10/09 23:47:15)
Wed Oct 09 23:47:17 JST 2019
Sum=>160
Wed Oct 09 23:47:19 JST 2019
```

サンプル :

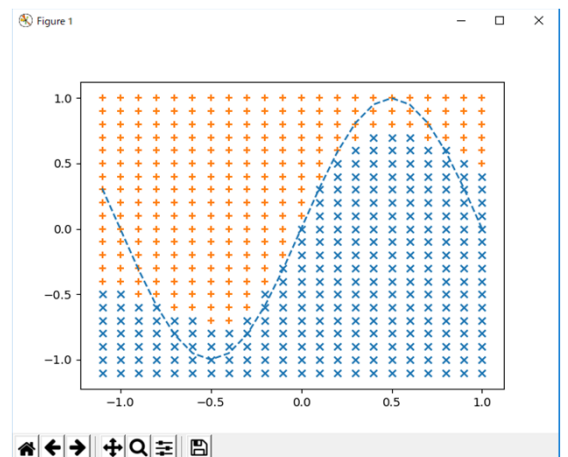
設問 : 実装 0 3 : Learning_Neu_0030.py



設問 : 実装 0 4 : Learning_Neu_0040.py



設問 : 実装 0 5 : Learning_Neu_0050.py



第0編 はじめに

1 本講義の全般説明

(A) Python言語講座

Python言語で使用するアプリ：

python(SQLite3)	言語
Django	Webアプリ
Numpy	科学技術ライブラリー（人工知能）
Matplotlib	xy座標の作図
MySQL	データベース
Sakuraエディト	エディター

本講座で使用するのは、最上項目Python言語のみである。
ただし、演習課題で詳細説明なしでMatplotlibを使う場合がある。

(B) 使用する教科書

やさしいPython、高橋麻奈著、SB Creative社出版

(C) 講義要領

授業は、講義と演習、課題演習、人工知能の実装で構成される。

(a) 講義
本講座は、本テキストに沿って学習する。 ただし、講義はプロジェクターを用いて重要な項目およびサンプルを説明する。
(b) 課題演習
主要な項目が終了する毎に理解度を深めるために、簡単な課題演習を行う。
(c) 総合演習
最後に、人工知能の課題演習として人工知能実装を行う。

(D) 講義内容

(a) 概要

原則として下記のような項目を実施するが、履修者の理解度に応じて柔軟に対応する。

第1回 基本文法と制御処理	<p>Python言語の基本的な文法を説明する。</p> <p>変数と式・演算子の基本</p> <p>制御処理としてif文, 論理演算子, for文, while文, 処理の流れの制御</p> <p>最後に、制御処理に関する課題演習を行う。</p>
第2回 コレクションと関数	<p>Python言語特有なコレクションと関数を説明する。</p> <p>リスト, 範囲(Range)スライス, タプル, ディクショナリの基本, セット</p> <p>関数の定義と呼び出し, 引数, 戻り値, 変数とスコープ</p> <p>最後に、リストと関数に関する課題演習を行う。</p>
第3回 クラスとクラスの応用	<p>オブジェクト指向プログラミングの中核をなすクラスを説明する。</p> <p>クラスの基本, コンストラクタ, クラス変数・クラスメソッド, カプセル化, 新しいクラス, モジュール, 標準ライブラリ</p> <p>クラスの拡張として、文字列クラス・ファイルクラス・日付クラス・例外クラスを説明し、 最後に、クラスのメソッド呼び出しに関する課題演習を実施する。</p>
第4回 Numpy, Matplotlib とニューラルネットワーク	<p>Numpy (数値演算) , Matplotlib(図)の概要、使い方を説明して、ニューラルネットワークを説明する。</p> <p>Numpy, Matplotlib</p> <p>本パーセプトロン・ニューラルネットワーク(Neural network)</p> <p>活性化関数</p> <p>最後に、ニューラルネットワークの演習を行う。</p>
第5回 人工知能（学習則）とその実装	<p>第4回に引き続き、バックプロパゲーションを説明する・</p> <p>活性化関数</p> <p>学習則</p> <p>学習則（偏微分）の数値演習</p> <p>最後に、人工知能の実装を行う。</p> <p>正弦波の回帰・分類問題（モデル化・損失関数・学習則）</p> <p>pattern認識に関する分類問題の実装例・アヤメの3品種の分類問題の実装例</p>

(b) カリキュラム

項目	内容	時間
第Ⅰ篇 基本文法と制御処理		8時間
1 基本文法	(1) 基礎事項	
2 制御処理	(2) 式と変数 (3) 演算子 (1) if 文 (2) for文 (3) while文 (4) 処理流れの制御 break, continue, pass	
3 特殊演算	範囲とスライス 課題演習	
第Ⅱ篇 コレクションと関数		
1 コレクション	(1) 概要 (2) リスト(List) (3) タプル(Tuple) (4) 辞書(連想配列)(Dictionary) (5) セット(Set)	
2 関数	(1) 概要 (2) 定義と呼び出し (3) 引数 (4) 戻り値 (5) 関数オブジェクト	
3 変数とスコープ	変数とスコープ 課題演習	
第Ⅲ篇 クラスと継承		
1 クラス	(1) クラスの基本 (2) コンストラクタ (3) クラス変数とクラスメソッド (4) カプセル化	
2 継承	(1) 継承(extends) (2) モジュール化(import)	
第Ⅳ篇 クラスの応用		
1 文字列	文字列と正規表現	
2 ファイル操作	ファイル操作	
3 例外処理	例外処理	
4 日付と時刻	日付と時刻 課題演習	
第Ⅴ・Ⅵ篇 Numpy と Matplotlib		
1 Numpy	数値計算用のライブラリー(行列の演算、特殊演算)	
2 Matplotlib	作図用のライブラリー	
第Ⅶ篇 人工知能(ニューラルネットワーク)		
1 ニューラル ネットワーク	ニューロンのモデル パーセプトロンとニューラルネットワーク 課題演習	
第Ⅷ篇 人工知能(バックプロパゲーション・学習則)		
1 人工知能	活性化関数 学習則 学習則の数値演習	
2 実装	正弦波の回帰・分類問題 (モデル化・損失関数・学習則) pattern認識に関する分類問題の実装例・アヤメの3品種の分類問題の実装例	

2 講義準備

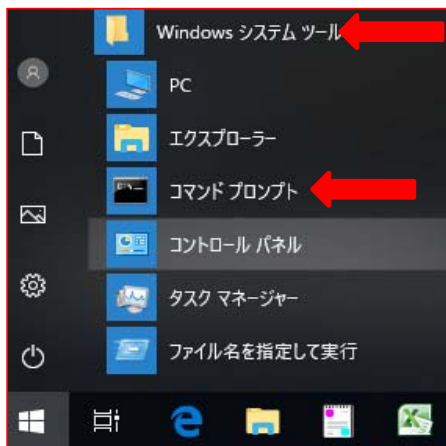
(1) インストール (済み)

python
Numpy
Matplotlib
sakura

(2) pythonの確認

(A) Pathの確認

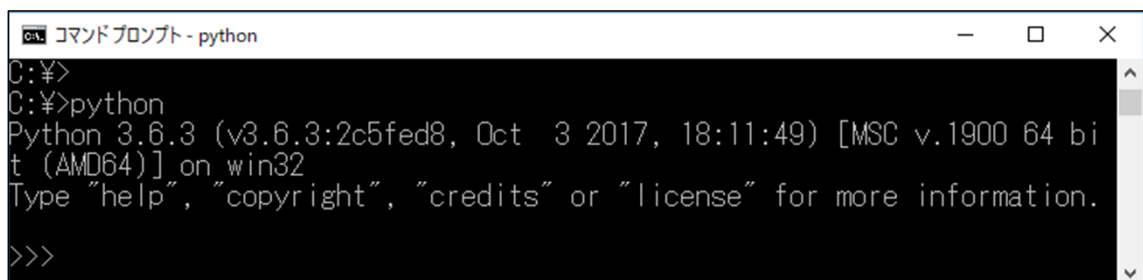
下記のようにしてコマンドプロンプトを起動する。



コマンドプロンプト画面で

```
C:¥> python
```

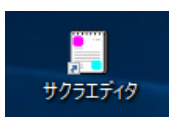
と入力すると、下記の画面が表示されたら、pythonがインストールされていることが分かる。



終了は「CTRL+Z」 + 「リターンキー」 または `exit()` です。

(3) Sakuraエディタの確認

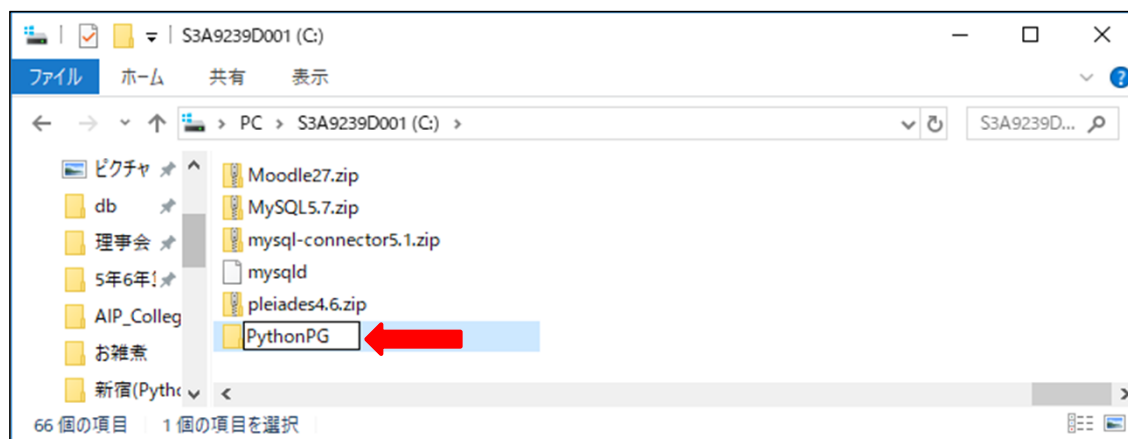
デスクトップに下記のショートカットがあることを確認しなさい。



(4) プログラムを格納するフォルダの作成

C¥:にプログラムを格納するつぎのフォルダを作成する。

[C:¥PythonPG]



以上で必要なソフトがインストールされており、pythonが使える状態であることが確認できた。

(★) NumPyのインストール方法

(A) インストール(既にインストールされている)

pip[C:¥Python3.6.3¥Lib¥site-packages¥pip]があることを確認する。

C:¥python3.6.3>pip install numpy

```

C:\Python3.6.3>pip install numpy
Installing collected packages: numpy
Successfully installed numpy-1.15.2
You are using pip version 9.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
C:\Python3.6.3>

```

(B) Numpyインストール確認

C:¥python3.6.3>python**>>>import numpy**

```

C:\Python3.6.3>python
Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 18:11:49) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy
>>>

```

Numpyの概要

インポート : `import numpy as np`

リストの作り方

<code>np.arange(0,5)</code>	0,1,2,3,4
<code>np.arange(0, 1, 0.1)</code>	0,0.1,0.2,...,0.9
<code>np.linspace(0,1,11)</code>	[0,1]を11等分する 0, 0.1, 0.2, ..., 0.9, 1.0

```
arr0 = np.array( [1,2,3 4,5,6, 7,8,9] )
arr0.reshape(3,3)    [1,2,3 4,5,6, 7,8,9]を3行3列に分解する
```

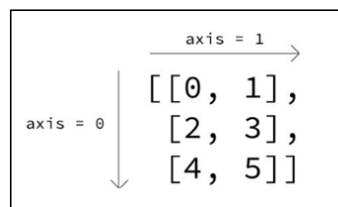
<code>aa = 1 + [2,3,4]</code>	<code>aa=[1+2, 1+3, 1+4]</code>
<code>bb = 10/[1,2,5]</code>	<code>bb=[10/1, 10/2, 10/5]</code>

行列の積 :

$$C = AB = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 10 & 1000 \\ 20 & 2000 \\ 30 & 3000 \end{bmatrix}$$

`C=np.dot(A,B)`

行列の行または列の和



列の和 :

`B=np.sum(A, axis=0)` [6,9]

行の和

`C=np.sum(A, axis=1)` $\begin{bmatrix} 1 \\ 5 \\ 9 \end{bmatrix}$

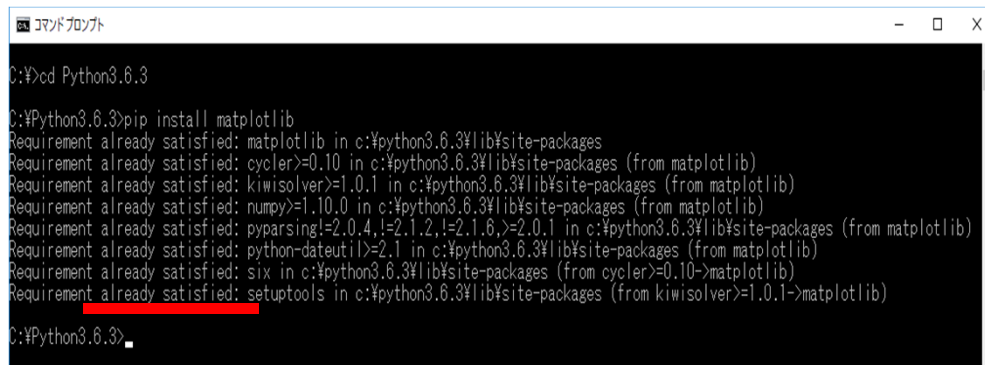
リストのシャッフル

`index_list = np.arange(n) # 0,1,2,3,...,n-1``np.random.shuffle(index_list) # n-10,5,12,3,..., m (ランダムになる)`

(★) Matplotlibのインストール方法

(A) インストール(既にインストールされている)

pip[C:\Python3.6.3\Lib\site-packages\pip]があることを確認する。

C:\python3.6.3>pip install matplotlib


```

C:\Python3.6.3>pip install matplotlib
Requirement already satisfied: matplotlib in c:\python3.6.3\lib\site-packages
Requirement already satisfied: cython>=0.10 in c:\python3.6.3\lib\site-packages (from matplotlib)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\python3.6.3\lib\site-packages (from matplotlib)
Requirement already satisfied: numpy>=1.10.0 in c:\python3.6.3\lib\site-packages (from matplotlib)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\python3.6.3\lib\site-packages (from matplotlib)
Requirement already satisfied: python-dateutil>=2.1 in c:\python3.6.3\lib\site-packages (from matplotlib)
Requirement already satisfied: six in c:\python3.6.3\lib\site-packages (from cython>=0.10->matplotlib)
Requirement already satisfied: setuptools in c:\python3.6.3\lib\site-packages (from kiwisolver>=1.0.1->matplotlib)
C:\Python3.6.3>

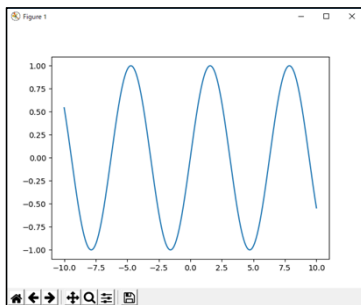
```

簡単正弦波を書いてみる。

```

C:\>python
>>>import numpy as np
>>>import matplotlib.pyplot as plt # pltとしてインポートされるのが慣例です。
>>>X = np.linspace(-10, 10, 1000) # 1000個がない場合は50個
>>>y = np.sin(X) # サインの値を計算する
>>>plt.plot(X, y) # これでプロットをする。plotで点と点同士をなめらかにつなぐ
>>>plt.show() # グラフの表示

```



Matplotlibの概要

インポート : import matplotlib.pyplot as plt

データ :

x=[?,?, ..., ?] x座標の値を格納する

y=[?,?, ..., ?] X座標に対応するY座標の値を格納する

タイトル

タイトル: plt.title('文字列')

Xラベル: plt.xlabel('文字列')

Yラベル: plt.ylabel('文字列')

描画 :

plt.plot(x, y, '+')

plt.show()