

NumPyとpandas

1ブロックずつ実行してみましょう

はコメントを表します

NumPy の利用

NumPyを使うと、配列・行列の計算が可能となります。

NumPyでは ndarray というデータ形式(numpy配列)を扱います。
これは、Rのベクトルや行列に相当します。

1次元の配列を作る方法を以下に示します。

```
In [ ]: import numpy as np      #numpyをnpという名前で読み込む

#numpy配列を作成 (リストから変換)
a = np.array([1, 2, 3])
A = np.array([4, 5, 6])

#Rのベクトルと同様に計算が可能
print("a+A", type(a+A), " : ", a+A)  #要素どうしの足し算
print("a*A", type(a*A), " : ", a*A)  #要素どうしの掛け算
print("a*5", type(a*5), " : ", a*5)  #5を掛ける
```

2次元の行列を作る方法を以下に示します。

```
In [ ]: #リストを2次元のnumpy配列 (行列) に変換
print("\n", np.array([[1,2,3], [4,5,6], [7,8,9]]))  #行列を作成

#.arange() でnumpy配列を作成
Nums = np.arange(4, 63, 2, dtype=np.int32)  #4から62まで間隔2で作成
print("\narray:\n", Nums)

Nums = np.insert(Nums, 0, 2)  #0番目に2を挿入
Nums = np.append(Nums, 64)  #最後に64を追加
print("array:\n", Nums)

Nums = np.reshape(Nums, (8, 4))  #8x4の行列に変換
print("\narray:\n", Nums)

print("\nshape: ", Nums.shape)  #.shapeはnumpy配列が持つメソッド
print("object type: ", type(Nums))  #関数type() で型を表示
```

添字を使って要素を参照できます。

```
In [ ]: print( Nums[2] )      #row 2 (2次元の場合は行が1つの要素)
print( Nums[2:4] )          #row 2 と 3
print( Nums[1][2] )         #row 1, column 2 (1番目の中の2番目)
print( Nums[1, 2] )         #row 2, column 1 (カンマで区切る方法)
print( Nums[:, 2] )         #全てのrow, column 2
print( Nums[2, :] )         #row 2, 全てのcolumns
```

pandas の利用

pandasを使うと、データフレームの利用が可能となります。

Rのデータフレームとほぼ同じと考えてよいでしょう。

データフレームを作る方法を以下に示します。

```
In [ ]: import pandas as pd    #pandasをpdという名前で読み込む
        from numpy import nan #numpyの欠損値機能を使う

#データフレームの作成
DFSize = pd.DataFrame({"cup"      : ["Kids", "Short", "Medium", "Tall", "Grand"],
                       "fl.oz"   : [7, 10, 14, 18, 24],
                       "USD"     : [nan, 2.45, 2.85, 3.25, 3.65] })

#列の参照
print( DFSize['cup'] )

#データフレームの列どうしの演算
DFSize['UnitPrice'] = DFSize['USD'] / DFSize['fl.oz']

print( "\nobject type:\n", type(DFSize) ) #型を表示
display( DFSize )                       #データフレームを表示
```

添字を使って要素を参照できます。
数字による参照では、`.iloc` を使います。

```
In [ ]: #添え字での参照 (.ilocを使う)
        print( "\n", DFSize.iloc[1, 2] )    #row 1, column 2
        print( "\n", DFSize.iloc[0:2, :] )  #row 0:2, 全てのcolumn
        print( "\n", DFSize.iloc[:, 2] )    #全てのrow, column 2
```

以下では、条件に該当する行を抽出しています。

```
In [ ]: #条件式での抽出
        print( "\n", DFSize[DFSize.cup == "Tall"] ) #Cupが"Tall"のものを抽出
        print( "\n", DFSize[DFSize.USD <= 3.0] )  #価格が3.0以下のものを抽出
```