データベース演習 （Advanced Database Exercise）

９．ガウス分布，クラスタリング

URL: http://www.kkaneko.jp/cc/dbenshu/index.html

**概要 Abstract**

今日の授業では、データの分類に関する演習を行う．

Today's class is exercise on data classification

■ **データフレーム**

Python では「テーブル」の形のデータのことを，「**データフレーム**」と呼ぶことが多い

　　リレーショナルデータベース：　**テーブル**

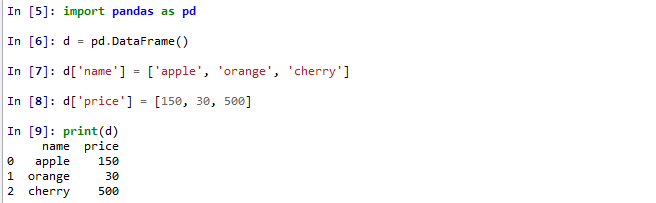
　　エクセル：**ワークシート**

　　Python： **データフレーム**

■ **pandas のデータフレームの列名 Column name of a pandas dataframe**

　pandas のデータフレームの列には、数字や文字列などの列名がある．下記の例では,name, price の2つの列名がある．

Each column of pandas dataframe has a numeric or string column name. In the following example, there are two column names: 'name' and 'price'.

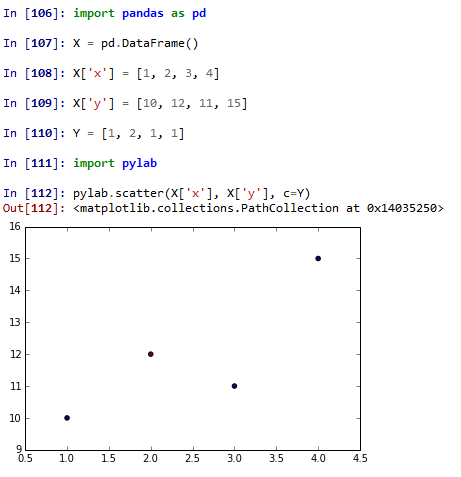
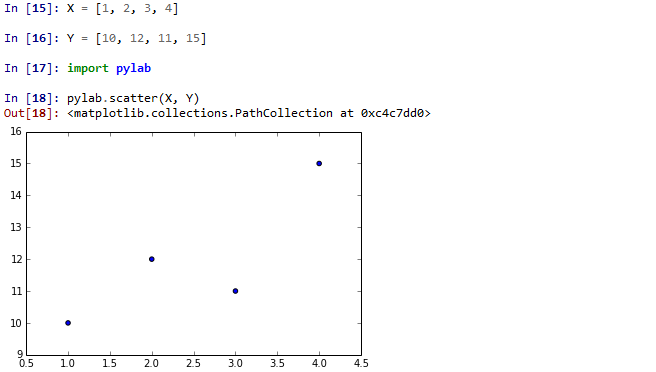


**■ Python の pylab パッケージを用いた散布図のプロット**

Python での散布図のプロットの例を下に示す．**pylab.scatter**を使う

Scatter plot using Python pylab package

　An example of scatter plot using Python is following (It uses **pylab.scatter**)

****

**■ Python の数値列 Python numeric interval**

Python では、区間 [s, t) の数値列（s で始まり、tの値を超えることが無い数値列）は、**arange** を用いて、次のように生成する．

　　A numeirc interval [s, t) including start but excluding stop can be generated as follows.



■ **メッシュグリッド meshgrid**

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 1 | 2 | 3 |

メッシュグリッド

|  |  |  |
| --- | --- | --- |
| 10 | 10 | 10 |
| 11 | 11 | 11 |

図. メッシュグリッドの例 A meshgrid example

2次元のメッシュグリッドは、2つの2次元配列からなる．

　1) x座標を格納した配列．各列には同じ値が入る

2) y座標を格納した配列．各行には同じ値が入る．

A two-dimensional mesh grid consists of two two-dimensional arrays.

1) One stores x-coodinate values. Each column is the same value.

2) Another stores y-coordinate values. Each line is the same value.

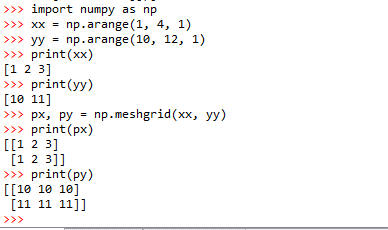
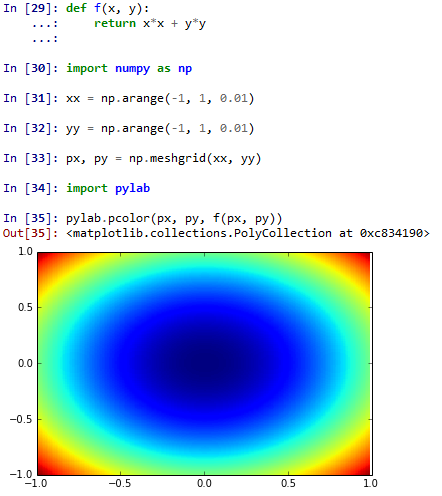
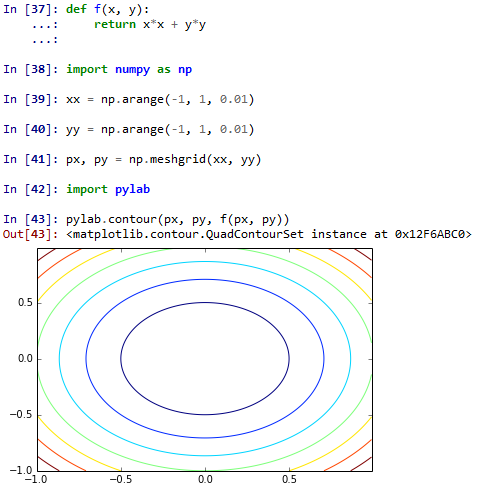


図 Python でのメッシュグリッド生成例 A example Python program to generate a meshgrid

**■ メッシュグリッドを用いたグラフ表示　Graph display using mesh grid**

メッシュグリッドはグラフ表示に役立つ



[

図．メッシュグリッドを用いた f(*x*, *y*) =*x*2 + *y*2 の表示 図. f(*x*, *y*) =*x*2 + *y*2 の等高線表示

Displaying f(*x*, *y*) =*x*2 + *y*2 using meshgrid　　　　　　　　　 Displaying contour of f(*x*, *y*) =*x*2 + *y*2

■ （参考）**メッシュグリッドを数値列に変換する操作**

import numpy as np

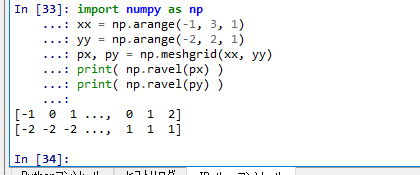
xx = np.arange(-1, 3, 1)

yy = np.arange(-2, 2, 1)

px, py = np.meshgrid(xx, yy)

print( np.ravel(px) )

print( np.ravel(py) )



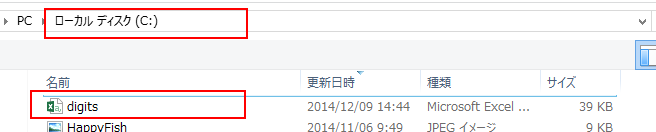
**演習 (Exercises)**

* **ステップ1 (Step 1)**

**課題：以下の手順で、CSVデータファイルを準備しなさい　(Prepare a CSV data file)**

USB メモリを使い、CSVデータファイル **digits.csv** を配る．各自、ファイルを **C:\digits.csv にコピー.**

A database file named '**digits.csv** ' will be distributed using a USB memory. Everyone is expected to copy the database file to '**C:\digits.csv**'.



今日の授業で使用するCSVデータファイルの出典と著作権表示

出典：http://archive.ics.uci.edu/ml/datasets/Pen-Based+Recognition+of+Handwritten+Digits   
===  
E. Alpaydin, Fevzi. Alimoglu   
Department of Computer Engineering   
Bogazici University, 80815 Istanbul Turkey   
alpaydin **'@'** boun.edu.tr

* **ステップ2 (Step 2)**

①　スタートメニューの「**Anaconda3 (64bit)**」の下の「**spyder**」を起動する．右下に IPython コンソール・ウインドウがある．

　　Launch “spyder” in the “Anaconda3 (64bit)” in a start menu. There is an IPython console windows in right lower.



**②** IPython コンソール・ウインドウで，**次の Python プログラムを実行させてみなさい**

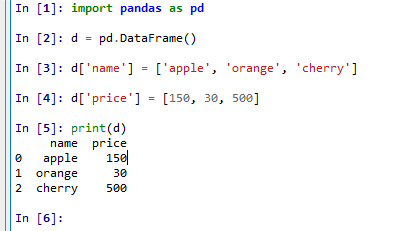
データフレームの演習である

import pandas as pd

d = pd.DataFrame()

d['name'] = ['apple', 'orange', 'cherry']

d['price'] = [150, 30, 500]

print(d)

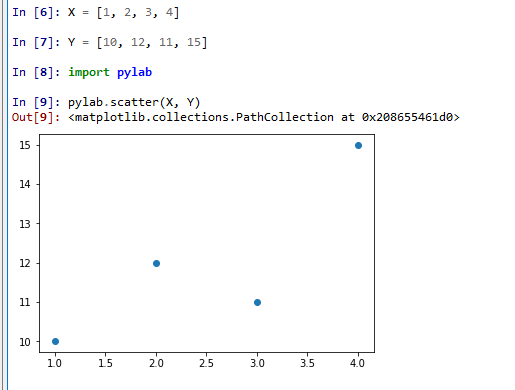
　　Python でのグラフの描画である

X = [1, 2, 3, 4]

Y = [10, 12, 11, 15]

import pylab

pylab.scatter(X, Y)



　　データフレームとグラフの描画

import pandas as pd

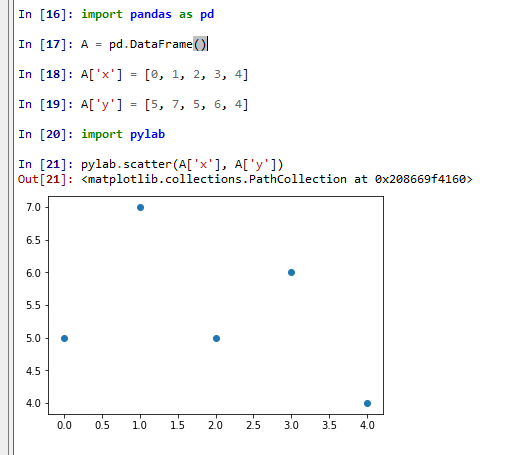
A = pd.DataFrame()

A['x'] = [0, 1, 2, 3, 4]

A['y'] = [5, 7, 5, 6, 4]

import pylab

pylab.scatter(A['x'], A['y'])



数値列

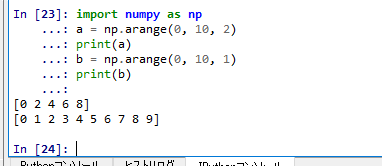
import numpy as np

a = np.arange(0, 10, 2)

print(a)

b = np.arange(0, 10, 1)

print(b)



メッシュグリッド

xx= np.arange(-1, 1, 0.1)

print(xx)

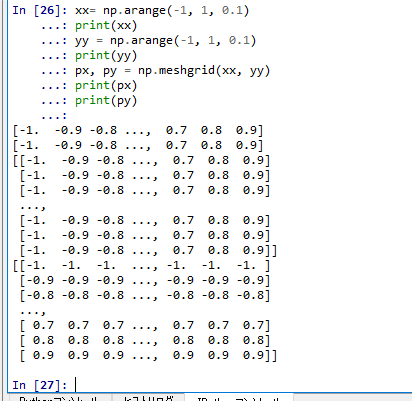
yy = np.arange(-1, 1, 0.1)

print(yy)

px, py = np.meshgrid(xx, yy)

print(px)

print(py)

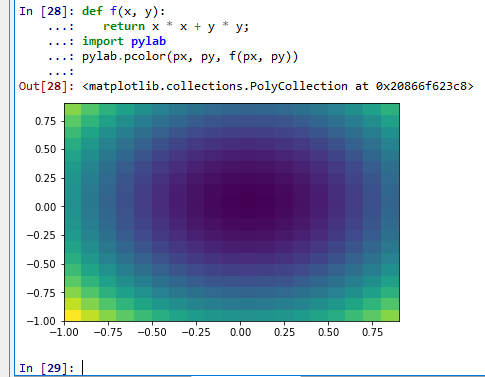


def f(x, y):

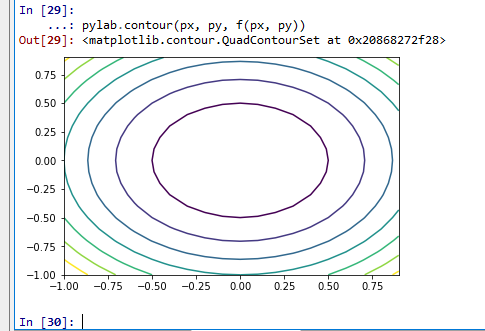
return **x \* x + y \* y**;

import pylab

pylab.pcolor(px, py, f(px, py))



pylab.contour(px, py, f(px, py))



* **ステップ３(Step ３)**

**(1) CSVファイル digits.csv を Pandasのデータフレームに格納する**

read a CSV file and store it into a Pandas data frame

1. まず CSVファイルのファイル名を確認する (Examine the correct CSV file name)



　以下、ファイル名が **C:\digits.csv** であるとして説明を続ける (In this exercise, the file name is c:\digits.csv)

2. データフレーム **d** に CSVファイルを読み込む. 列名は 0, x, y, c の4つである

　　(read the CSV file into a dataframe d. There are four column names: 0, x, y and c).

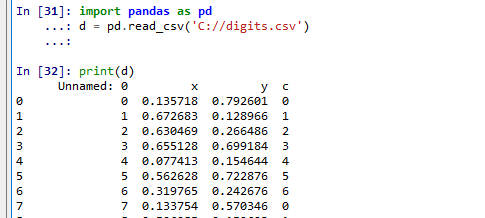
import pandas as pd

d = pd.read\_csv('**C://digits.csv**')

print(d)

* ファイル名 **C:\digits.csv**

Python で書くときは **C://digits.csv**

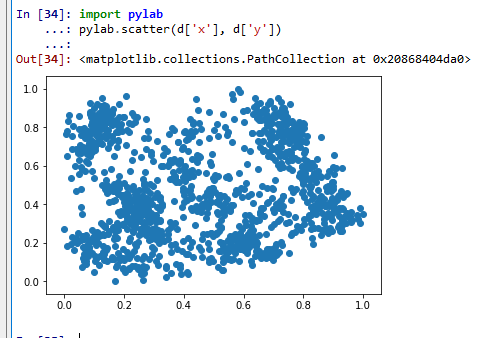


(2) データフレーム d のデータから**散布図**の作成

Exercises on displaying **a scatter plot** using database

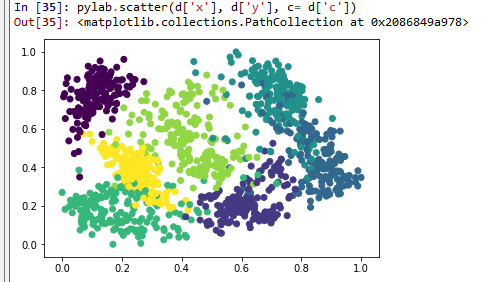
import pylab

pylab.scatter(d['x'], d['y'])



今度は c 列のデータを使って色を付ける．

pylab.scatter(d['x'], d['y'], c= d['c'])



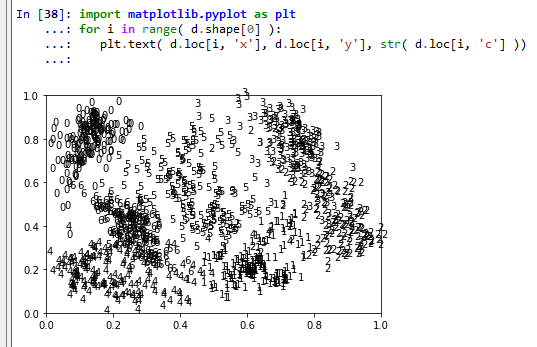
３．散布図の作成(色と番号付きでプロット)

　　　scatterplot with color and number

import matplotlib.pyplot as plt

for i in range( d.shape[0] ):

plt.text( d.loc[i, 'x'], d.loc[i, 'y'], str( d.loc[i, 'c'] ))



import matplotlib.pyplot as plt

for i in range( d.shape[0] ):

plt.text( d.loc[i, 'x'], d.loc[i, 'y'], str( d.loc[i, 'c'] ), **color=plt.cm.Set1(d.loc[i, 'c'])**)

