Alexander Fronimopoulos

Άσκηση: 1.1

Α) Εφόσον πρέπει να τρέξουμε το προηγούμενο παράδειγμα με άλλες πιθανότητες, αλλάζω τις πιθανότητες στον κώδικα και έχουμε τα εξής αποτελέσματα – τελικές πιθανότητες για p1: 0.0139 και για p2: 0.0314



Β) Το διάνυσμα χ ταξινομείται στη δεύτερη κλάση, επειδή η p2 είναι μεγαλύτερη.

Γ) Η a-priori είναι αρκετά μεγαλύτερη από την a-posteriori.

Άσκηση: 1.2

Α) κώδικας python:  
Για το πρώτο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

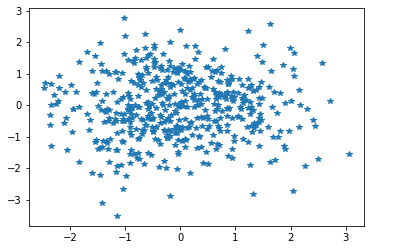
cov = [[1, 0], [0, 1]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το δεύτερο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

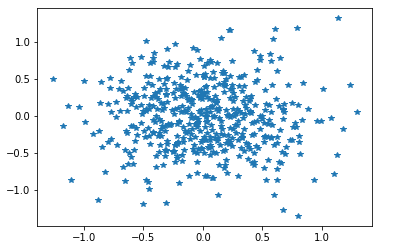
cov = [[0.2, 0], [0, 0.2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το τρίτο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

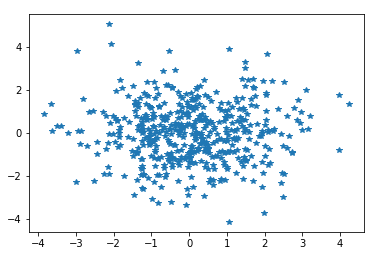
cov = [[2, 0], [0, 2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το τέταρτο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

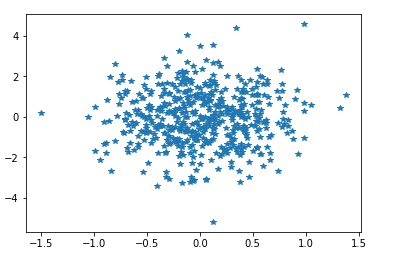
cov = [[0.2, 0], [0, 2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το πέμπτο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

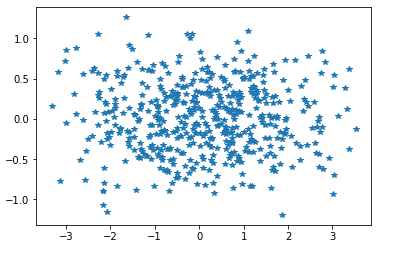
cov = [[2, 0], [0, 0.2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το έκτο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

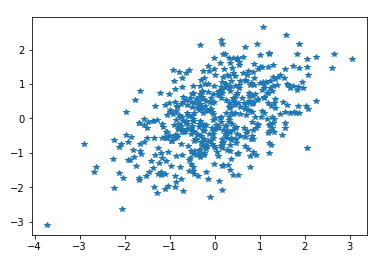
cov = [[1, 0.5], [0.5, 1]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το έβδομο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

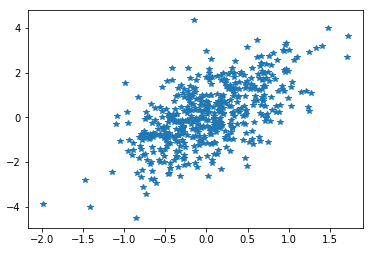
cov = [[0.3, 0.5], [0.5, 2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

plt.show()

Εικόνα:



Για το όγδοο σύνολο σ:

import matplotlib.pyplot as plt

import numpy as np

mean = [0, 0]

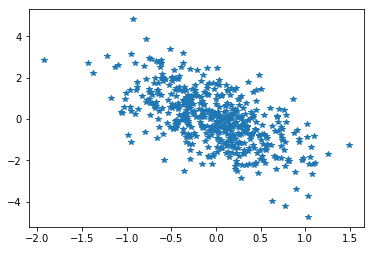
cov = [[0.3, -0.5], [-0.5, 2]]

x, y = np.random.multivariate\_normal(mean, cov, 500). T

plt.plot(x,y,'\*')

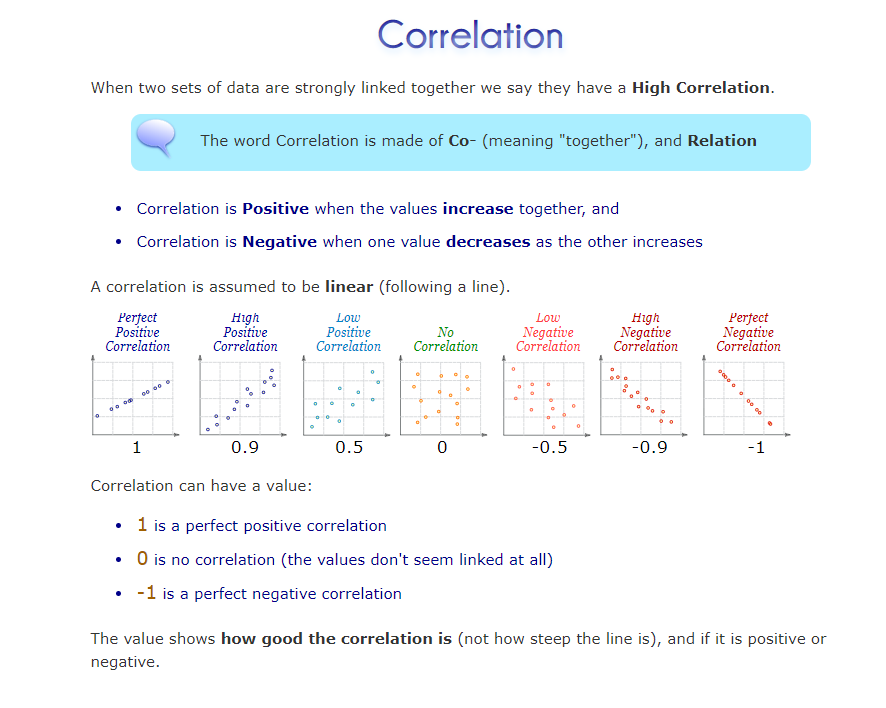
plt.show()

Εικόνα:



Β) Σχολιασμός:

Σχολιασμός:Ανάλογα που τείνουν τα σ12&σ21 (που είναι ίσα γιατί ο πίνακας είναι συμμετρικός) έχουμε και το αντίστοιχο correlation για τα δεδομένα, παρακάτω βλέπουμε που υπάρχει υψηλό και που χαμηλό correlation.



Άσκηση: 2

Α) Κώδικας:

import numpy as np  
import matplotlib.pyplot as plt   
import pandas as pd  
import seaborn as sns  
from sklearn.model\_selection import train\_test\_split  
from sklearn.preprocessing import StandardScaler  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.metrics import confusion\_matrix  
from sklearn import metrics

data = pd.read\_csv('/Users/Froni/OneDrive/Desktop/seeds\_dataset.txt')

data.head ()

kf = model\_selection.KFold(n\_splits=5, shuffle=False) means = [] for training,testing in kf.split(features):

knn.fit(features[training], target[training]) prediction = knn.predict(features[testing])

curmean = np.mean(prediction == target[testing]) means.append(curmean) print('Mean accuracy: {:.1%}'.format(np.mean(means)))

knn = KNeighborsClassifier(n\_neighbors=5)

model.fit(features,label)

predicted= model.predict([[0,2]]) # 0:Overcast, 2:Mild print(predicted)

knn.fit(X\_train, y\_train)

y\_pred = knn.predict(X\_test)

plt.show()