Άσκηση 1

Gaussian συνάρτηση πυκνότητας πιθανότητας (σ.π.π)

Η Gaussian συνάρτηση πυκνότητας πιθανότητας (σ.π.π) χρησιμοποιείται ευρύτατα στο πεδίο της Μηχανικής Μάθησης λόγω της μαθηματικής ανιχνευσιμότητας αλλά και λόγω του Κεντρικού Οριακού Θεωρήματος. Σε σχέση με το Κεντρικό Οριακό Θεώρημα, η σ.π.π. του αθροίσματος ενός αριθμού στατιστικά ανεξάρτητων τυχαίων μεταβλητών τείνουν να είναι Gaussian καθώς ο αριθμός των όρων τείνει στο άπειρο. Κάτι τέτοιο στην πράξη είναι αληθές για ένα μεγάλο αριθμό όρων.

Συχνά αναφερόμαστε σε μία Gaussian σ.π.π. ως την κανονική σ.π.π. και χρησιμοποιούμε τη σημειολογία $N(\boldsymbol{m}, \boldsymbol{S})$.

Η πολυδιάστατη Gaussian σ.π.π. έχει τη μορφή:

$$\mathbf{p(x)} = \frac{\mathbf{1}}{\mathbf{(2\pi)^{d/2}|S|^{1/2}}} \mathbf{exp}\left(-\frac{\mathbf{1}}{\mathbf{2}}\mathbf{(x-m)^T S^{-1}(x-m)}\right)$$

όπου $\mathbf{m} = \mathbf{E[x]}$ είναι το μέσο διάνυσμα και
S είναι ο πίνακας συνδιακύμανσης που ορίζεται ως: $\mathbf{S} = \mathbf{(x-m)^T(x-m)}$ και $|\boldsymbol{S}|$ είναι η ορίζουσα του **S**.
Με d συμβολίζουμε το πλήθος των διαστάσεων.

Για τη μονοδιάστατη περίπτωση όπου $\mathbf{x}\boldsymbol{\epsilon R}$**, η Gaussian σ.π.π.** μετατρέπεται στην:

$$\mathbf{p(x)} = \frac{\mathbf{1}}{\sqrt{\mathbf{2\pi}}\boldsymbol{\sigma}} \mathbf{exp}\left(-\frac{\mathbf{(x-m)^2}}{\mathbf{2\sigma^2}}\right)$$

Παράδειγμα 1. Να υπολογιστεί η τιμή της Gaussian σ.π.π. $N(m,S)$ στη θέση $x_1 = [0.2, 1.3]^T$ και $x_1 = [2.2, -1.3]^T$ όπου $m = [0,1]^T, S = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$

Λύση:

```
import numpy as np
#mean = (1, 2)
#cov = [[1, 0], [0, 1]]
#x = np.random.multivariate_normal(mean, cov, (3, 3))
#x.shape

# Mean vector and covariance matrix
mu = np.array([0., 1.])
Sigma = np.array([[ 1. , 0], [0,  1.]])

x1 =np.array([0.2, 1.3])
x2 =np.array([2.2, -1.3])

from scipy.stats import multivariate_normal
print(multivariate_normal.pdf(x1,mu, Sigma))
print(multivariate_normal.pdf(x2,mu, Sigma))
```

Οι τιμές που προκύπτουν είναι: 0.1491 και 0.001 αντίστοιχα.

Παράδειγμα 2. Σκεφτείτε το πρόβλημα της ταξινόμησης 2-κλάσεων στο δισδιάστατο χώρο, όπου τα δεδομένα και στις δύο κλάσεις ω_1, ω_2, κατανέμονται σύμφωνα με τις Gaussian κατανομές $N(m_1, S_1)$ και $N(m_2, S_2)$ αντίστοιχα. Έστω $m_1 = [1,1]^T,\ m_2 = [3,3]^T,\ S_1 = S_2 = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$.
Υποθέτοντας ότι $P(\omega_1) = P(\omega_2) = 1/2$, ταξινομήστε το $\mathbf{x} = [1.8, 1.8]^T$ στις κλάσεις ω_1ή ω_2.

Λύση:

```
P1=0.5
```

```
P2=0.5
mu1 = np.array([1., 1.])
mu2 = np.array([3., 3.])

Sigma=np.eye(2, dtype=int)
x=[1.8, 1.8]

from scipy.stats import multivariate_normal
print(P1*multivariate_normal.pdf(x,mu1, Sigma))
print(P2*multivariate_normal.pdf(x,mu2, Sigma))
```

Οι τιμές που προκύπτουν είναι προσεγγιστικά: 0.042 και 0.0189.

Ασκήσεις για παράδοση

Άσκηση 1.1 για παράδοση: Να επαναλάβετε την προηγούμενη άσκηση για πιθανότητες $P(\omega_1) = 1/6, P(\omega_2) = 5/6$. Ποιες είναι οι τελικές πιθανότητες; Που ταξινομείται το διάνυσμα **x**; Ερμηνεύστε τα αποτελέσματα της ταξινόμησης ως προς την εξάρτησή τους σε σχέση με τις a-priori πιθανότητες.

Άσκηση 1.2 για παράδοση. Να δημιουργήσετε N=500 διδιάστατα σημεία δεδομένων που κατανέμονται σύμφωνα με την Gaussian κατανομή $N(m, S)$ με μέσο $m = [0,0]^T$ και πίνακα συνδιακύμανσης $\begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix}$ για τις ακόλουθες περιπτώσεις (θυμηθείτε ότι ο πίνακας S είναι συμμετρικός) :

$\sigma_1^2 = \sigma_2^2 = 1, \sigma_{12} = 0$

$\sigma_1^2 = \sigma_2^2 = 0.2, \sigma_{12} = 0$

$\sigma_1^2 = \sigma_2^2 = 2, \sigma_{12} = 0$

$\sigma_1^2 = 0.2, \quad \sigma_2^2 = 2, \sigma_{12} = 0$

$\sigma_1^2 = 2, \sigma_2^2 = 0.2, \sigma_{12} = 0$

$\sigma_1^2 = \sigma_2^2 = 1, \sigma_{12} = 0.5$

$\sigma_1^2 = 0.3, \quad \sigma_2^2 = 2, \sigma_{12} = 0.5$

$\sigma_1^2 = 0.3, \quad \sigma_2^2 = 2, \sigma_{12} = -0.5$

Να εμφανίσετε κάθε σύνολο δεδομένων και να σχολιάσετε επί των συστάδων (clusters) που δημιουργούνται.
Ο σχολιασμός να αφορά το σχήμα και τη μορφή του, το εάν οι συνιστώσες είναι συσχετισμένες και πόσο καθώς και επί της διασποράς του.

Άσκηση 2 – Nearest Neighbor classification

Στη προκείμενη άσκηση εξετάζουμε ένα γεωργικό σύνολο δεδομένων. Αυτό το σύνολο δεδομένων αποτελείται από μετρήσεις σπόρων σίτου. Υπάρχουν επτά χαρακτηριστικά που υπάρχουν, τα οποία έχουν ως εξής:

- ☐ **Area (Εμβαδό) A**
- ☐ **Perimeter (Περίμετρος) P**
- ☐ **Compactness (Συμπαγότητα) $C = 4\pi A/P^2$**
- ☐ **Length of kernel (μήκος του πυρήνα)**
- ☐ **Width of kernel (Πλάτος του πυρήνα)**
- ☐ **Asymmetry coefficient (Συντελεστής ασυμμετρίας)**
- ☐ **Length of kernel groove (Μήκος αυλάκωσης του πυρήνα)**

Ένα παράδειγμα των 5 πρώτων γραμμών έχει ως εξής:

1 15.26,14.84,0.871,5.763,3.312,2.221,5.22,1
2 14.88,14.57,0.8811,5.554,3.333,1.018,4.956,1
3 14.29,14.09,0.905,5.291,3.337,2.699,4.825,1
4 13.84,13.94,0.8955,5.324,3.379,2.259,4.805,1
5 16.14,14.99,0.9034,5.658,3.562,1.355,5.175,1

Δηλαδή υπάρχει επιπλέον και η έξοδος.

- ☐ **Class (1, 2, 3)**

Υπάρχουν τρεις κατηγορίες που αντιστοιχούν σε τρεις ποικιλίες σίτου: Canadian, Koma και Rosa. Ο στόχος είναι να είναι δυνατή η ταξινόμηση του είδους με βάση αυτές τις μορφολογικές μετρήσεις. Σε αντίθεση με το σύνολο δεδομένων Iris, το οποίο συλλέχθηκε στη δεκαετία του 1930, αυτό είναι ένα πολύ πρόσφατο σύνολο δεδομένων και τα χαρακτηριστικά του υπολογίστηκαν αυτόματα από ψηφιακές εικόνες.
Με τον τρόπο αυτό μπορεί να εφαρμοστεί η αναγνώριση προτύπων εικόνας: μπορείτε να τραβήξετε εικόνες σε ψηφιακή μορφή, να υπολογίσετε μερικά σχετικά χαρακτηριστικά (features) από αυτές και να χρησιμοποιήσετε ένα γενικό σύστημα ταξινόμησης.
Το Πανεπιστήμιο της Καλιφόρνιας στο Irvine (UCI) διατηρεί ένα ηλεκτρονικό αποθετήριο των συνόλων δεδομένων Μηχανικής Μάθησης. Το σύνολο δεδομένων που μας ενδιαφέρει μπορεί να ληφθεί από εκεί. Το αποθετήριο διατίθεται στο διαδίκτυο στη διεύθυνση http://archive.ics.uci.edu/ml/.

Το συγκεκριμένο σύνολο δεδομένων μπορείτε να το βρείτε στη διεύθυνση:

http://archive.ics.uci.edu/ml/datasets/seeds

Μπορούμε να χρησιμοποιήσουμε cross-validation (of course) για να κοιτάξουμε τα δεδομένα μας. Το scikit-learn module μας βοηθά να κάνουμε διάφορα πράγματα:

```
kf = model_selection.KFold(n_splits=5, shuffle=False)
means = []
for training,testing in kf.split(features):
    # We learn a model for this fold with `fit` and then apply it to the
    # testing data with `predict`:
    knn.fit(features[training], target[training])
    prediction = knn.predict(features[testing])

    # np.mean on an array of booleans returns fraction
    # of correct decisions for this fold:
    curmean = np.mean(prediction == target[testing])
    means.append(curmean)
print('Mean accuracy: {:.1%}'.format(np.mean(means)))
```

Για να εισάγετε τον kNN ταξινομητή, μπορείτε να γράψετε:

```
from sklearn.neighbors import KNeighborsClassifier
```

Τώρα μπορούμε να δημιουργήσουμε ένα αντικείμενο object. Στον κατασκευαστή (θυμηθείτε την έννοια από τον αντικειμενοστρεφή), προσδιορίζουμε το πλήθος των γειτόνων:
Π.χ.

```
knn = KNeighborsClassifier(n_neighbors=1)
```

Αν δε καθοριστεί ο αριθμός των γειτόνων τότε αυτός τίθεται εξ' ορισμού ίσος με 5.

```
# Train the model using the training sets
model.fit(features,label)

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

Το ζητούμενο είναι:
Α) Να δημιουργηθούν τα μοντέλα για διάφορες τιμές του k.
Β) Στη συνέχεια να γίνει αξιολόγηση για όλες αυτές τις τιμές με τη βοήθεια της μετρικής acuracy.

Παράδειγμα κώδικα:

```
#Import knearest neighbors Classifier model
from sklearn.neighbors import KNeighborsClassifier

#Create KNN Classifier
knn = KNeighborsClassifier(n_neighbors=5)

#Train the model using the training sets
knn.fit(X_train, y_train)% το σημείο που πρέπει να καταλάβετε λίγο περισσότερο

#Predict the response for test dataset
y_pred = knn.predict(X_test)
```

Για να μελετήσει κανείς την ακρίβεια μπορεί να χρησιμοποιήσει τον εξής κώδικα:

```
#Import scikit-learn metrics module for accuracy calculation
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

Γ) Τα αποτελέσματα που θα προκύψουν σας δίνουν το classification rate (δηλ. το accuracy)
Να γίνει γραφική παράσταση της accuracy έναντι των διαφόρων τιμών του k στον αλγόριθμο kNN.

Σημείωση: Να παραδοθεί το πρόβλημα, η μελέτη του (δηλ. συνοπτικά τι κάνατε και τι αποτελέσματα προκύπτουν, συμπεράσματα, σχόλια κ.τ.λ) καθώς και ο κώδικάς του σε έγγραφο word ή pdf. Επίσης ο κώδικας να παραδοθεί και με ξεχωριστό τρόπο (όπως σας βολεύει).