Data Science et Machine Learning

TP sur l'Analyse Discriminante (AD)

A réaliser dans l'environnement Python avec les bibliothèques Scikit-Learn, Pandas et Matplotlib

1. Utilisez la bibliothèque Pandas pour extraire les data frames nécessaires à ce TP à partir du fichier Taille_Poids_Genre.xlsx joint à ce document.

   ```
   df = pd.read_excel('Taille_Poids_Genre.xlsx')
   ```

2. Construisez des tableaux Numpy pour les données X et le classement Y.

   ```
   XY = np.array(df)
   X = XY[:,0:p]
   Y = XY[:,p]
   ```

3. Construisez le tableau Z des données centrées réduites avec la fonction StandardScaler du module preprocessing de la bibliothèque Scikit-Learn.

   ```
   from sklearn.preprocessing import StandardScaler
   scaler = StandardScaler().fit(X)
   Z = scaler.transform(X)
   ```

4. Utilisez la fonction LinearDiscriminantAnalysis (LDA) du module discriminant_analysis de la bibliothèque Sci-kit learn pour réaliser une AFD sur ce même jeu de données.

   ```
   from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
   lda = LinearDiscriminantAnalysis().fit(Z,Y)
   ```

5. Récupérez les vecteurs u_k des axes discriminants à partir de l'attribut scalings_ de la LDA.

   ```
   U = lda.scalings_
   ```

6. Trouvez la projection F des individus sur avec la méthode transform de la LDA.

   ```
   F = lda.transform(Z)
   ```

7. Trouvez les moyennes des classes dans l'attribut means_ de la LDA et calculez leur projection sur les axes discriminants :

   ```
   Moyennes = lda.means_
   Moyennes_projetees = np.dot(Moyennes,U)
   ```

8. Construisez le dessin du nuage projeté avec Matplotlib.
9. Réalisez un classement des individus aux classes en précisant les scores correspondants en utilisant la méthode predict et la méthode predict_proba de la LDA.

   ```
   classement = lda.predict(Z)
   score = lda.predict_proba(Z)
   ```

10. Reconstruisez la figure du nuage projeté en mettant en vert les individus bien classés et en rouge les mal classés.
11. Calculer le pourcentage de bon classement.