SY09 Printemps 2022
TD/TP 09 — Analyse discriminante de données gaussiennes

1 Partie pratique

On souhaite comparer les performances de trois modèles d'analyse discriminante (analyse discriminante quadratique, analyse discriminante linéaire, et classifieur bayésien naïf sous hypothèse de normalité des classes) sur des jeux de données simulées.

Pour chacun des jeux de données, la distribution conditionnelle à chaque classe est gaussienne, les paramètres pouvant en revanche changer d'un jeu de données à l'autre.

On utilisera les instructions suivantes pour charger ces modèles.

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
from sklearn.naive_bayes import GaussianNB
```

1 Pour chacun des jeu de données bidimensionnels suivants :
- `data/SynthCross_n1000_p2.csv`,
- `data/SynthPara_n1000_p2.csv`,
- `data/SynthPlus_n1000_p2.csv`,

calculer les erreurs de validation croisée à 10 plis, comparer les performances des trois algorithmes et les interpréter.

Pour visualiser les erreurs conjointement, on pourra utiliser la fonction `sns.lineplot` après avoir généré un jeu de données regroupant les 10 erreurs de validation croisée pour chaque algorithme. À cette fin, on pourra d'abord définir un générateur `validation_errors` qui itère sur les modèles et génère pour chaque les 10 erreurs de validation croisée retournées par `cross_val_score`.

Pour expliquer ces résultats, on pourra s'appuyer sur la fonction `add_decision_boundary` définie au TP07. On pourra également définir une fonction `add_decision_boundaries` qui ajoute les frontières de décision pour chaque modèle.

2 Recommencer avec les jeux de données suivants :
- `data/SynthBlob_n1000_p300.csv.gz`,
- `data/SynthPara_n1000_p300.csv.gz`,
- `data/SynthPlus_n1000_p300.csv.gz`.

Les trois jeux de données étant de dimension > 2, on pourra utiliser la fonction `scatterplot_pca` du TP06 pour les visualiser.

3 Construire un jeu de données sur lequel le taux de bonne classification des trois modèles n'excède pas 60%.

Construire un classifieur simple « à la main » d'après ce que vous savez de ce jeu de données.

2 Partie théorique

Les jeux de données synthétiques utilisés au paragraphe 1 ont été obtenus par un processus génératif tel que décrit dans le TP08 :

1. tout d'abord, l'effectif n_1 de la classe ω_1 a été déterminé par tirage aléatoire suivant une loi binomiale $\mathcal{B}(n, \pi_1)$, avec $\pi_1 = 0.5$;
2. n_1 individus ont ensuite été générés dans la classe ω_1 suivant une loi normale bivariée $\mathcal{N}(\mu_1, \Sigma_1)$, et $n_2 = n - n_1$ dans la classe ω_2 suivant une loi normale bivariée $\mathcal{N}(\mu_2, \Sigma_2)$.

Questions.

4 Quelles sont les distributions marginales des variables X^1 et X^2 dans chaque classe ?

5 Calculer l'expression des courbes d'iso-densité dans la classe ω_k, en fonction de μ_k et Σ_k. À quoi correspondent ces courbes dans le cas général (Σ_k quelconque) ? Si Σ_k est diagonale, sphérique ?

6 Calculer l'expression de la frontière de décision de la règle de Bayes $\delta^\star$ dans le cas général.

7 Représenter cette frontière et comparer aux frontières obtenues avec les trois modèles d'analyse discriminante pour le jeu de données `SynthCross_n1000_p2.csv`, généré avec les paramètres suivants :

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \Sigma_1 = \frac{1}{2}\begin{pmatrix} 11 & 9 \\ 9 & 11 \end{pmatrix}, \quad \Sigma_2 = \frac{1}{2}\begin{pmatrix} 11 & -9 \\ -9 & 11 \end{pmatrix}.$$

Pour tracer une fonction implicite, on pourra s'inspirer du code suivant

```
# Les bornes
xlim = ax.get_xlim()
ylim = ax.get_ylim()

# Discrétisation et création du tableau des abscisses et des ordonnées
resolution = 1000
xx = np.linspace(xlim[0], xlim[1], resolution)
yy = np.linspace(ylim[0], ylim[1], resolution)
X, Y = np.meshgrid(xx, yy)

# Calcul de la fonction implicite
Z = ...

# Tracé de la ligne séparant Z > 0 de Z < 0
plt.contour(X, Y, Z, levels=[0])
plt.show()
```

8 Calculer l'expression de la frontière de décision de la règle de Bayes $\delta^\star$ lorsque Σ_1 et Σ_2 sont diagonales ($\Sigma_k = \text{diag}(\sigma_{k1}^2, \sigma_{k2}^2)$, pour $k = 1, 2$).

9 Représenter cette frontière et comparer aux frontières obtenues avec les trois modèles d'analyse discriminante pour le jeu de données `SynthPlus_n1000_p2.csv`, généré avec les paramètres suivants :

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \Sigma_1 = \begin{pmatrix} 15 & 0 \\ 0 & 1 \end{pmatrix}, \quad \Sigma_2 = \begin{pmatrix} 1 & 0 \\ 0 & 15 \end{pmatrix}.$$

10 Calculer la frontière de décision de la règle de Bayes $\delta^\star$ lorsque $\Sigma_1 = \Sigma_2 = \Sigma$.

11 Représenter cette frontière et comparer aux frontières obtenues avec les trois modèles d'analyse discriminante pour le jeu de données `SynthPara_n1000_p2.csv`, généré avec les paramètres suivants :

$$\mu_1 = \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \quad \Sigma = \Sigma_1 = \Sigma_2 = \frac{1}{2}\begin{pmatrix} 41 & 39 \\ 39 & 41 \end{pmatrix}.$$