Université de Thiès **Année Académique 2020-2021**
U.F.R SET **Master 1 Sciences de données**

Apprentissage Statistique

Travail individuel (pondération 10%).
à rendre avant 18 décembre 2022 à $23h59$

Exercice 1 Soient $X \in \mathbb{R}^{n\times(p+1)}$ la matrice contenant les données dont la $i^{\text{ème}}$ ligne est $(\boldsymbol{x}_i^{'}, 1)$ avec $\boldsymbol{x}_i^{'} = (x_1, \ldots, x_p)$ et $Y \in \mathbb{R}^n$ vecteur contenant les étiquettes y_i. L'estimateur des moindre carrés le vecteur

$$\begin{pmatrix}\hat{\alpha}\\ \hat{\beta}\end{pmatrix} = (X^TX)^{-1}X^TY = \min_{\alpha\in\mathbb{R}^p,\beta\in\mathbb{R}} \sum_{i=1}^{n}(y_i - (\langle\alpha, \boldsymbol{x}_i\rangle + \beta))^2$$

1. Programmez une fonction *regression*(X, Y) qui renvoie l'estimateur des moindre carrés.
   Utiliser votre fonction de régression sur le jeu de données Credit.csv (variable à expliquer *Balance* par toutes les autres variables du jeu de données). Comparez les vecteurs $\hat{\alpha}$ et $\hat{\beta}$ renvoyés par votre fonction avec les attributs *coef_* et *intercept_* d'un régresseur de type *linear_model.LinearRegression*.
   Quelques fonctions utiles : *dot*(), *transpose*(), *pinv*().
2. Écrire une fonction *regress*(X, α, β) qui renvoie le vecteur $\hat{Y}$ des étiquettes prédites tel que $\hat{y}_i = \langle\alpha, \boldsymbol{x}_i\rangle + \beta$
3. Calculer $\hat{\epsilon} = \|Y - \hat{Y}\|_2^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ l'erreur au sens des moindres carrés du régresseur appris sur l'ensemble du jeu de données Credit.
4. Dans certains cas, la matrice X^TX n'est pas inversible. Pour remédier à ce problème, on ajoute un ridge λI_{p+1} à cette matrice où I_{p+1} est la matrice identité d'ordre $p+1$.

   Cela correspond à une légère modification du problème d'optimisation qui pénalise la taille des coefficients. Le vecteur des moindre carrés généralisés est donné par :

$$\begin{pmatrix}\hat{\alpha}\\ \hat{\beta}\end{pmatrix} = (X^TX + \lambda I_{p+1})^{-1}X^TY = \min_{\alpha\in\mathbb{R}^p,\beta\in\mathbb{R}} \sum_{i=1}^{n}(y_i - (\langle\alpha, \boldsymbol{x}_i\rangle + \beta))^2 + \lambda\|\alpha\|_2^2$$

   (a) Programmez une fonction *ridge_regression*(X, Y, lambda) qui renvoie l'estimateur des moindre carrés généralisés. Comparez à nouveau les vecteurs $\hat{\alpha}$ et $\hat{\beta}$ obtenus pour le paramètre lambda = 1 sur le jeu de données Credit avec les attributs *coef_* et *intercept_* d'un régresseur de type *linear_model.Ridge*

   (b) Tracez l'évolution des coefficients du vecteur $\hat{\alpha}$ en fonction du paramètre de régularisation lambda pour des valeurs entre 0.001 et 1000. Quelles variables semblent le mieux expliquer la variable *Balance* ?

   (c) Trouvez par un moyen approprié la meilleure valeur pour le paramètre lambda. Apprenez ensuite un régresseur avec cette valeur sur l'ensemble du jeu de données Credit et calculez l'erreur au sens des moindres carrés sur ce même échantillon.

5. La formulation Lasso est une variante de la régression linéaire régularisée. La pénalisation du vecteur des coefficients se fait ici avec la norme $||_1$ a la place de la norme euclidienne $||_2$. Soit $\alpha \in \mathbb{R}^p$, $\|\alpha\|_1 = \sum_{i=1}^{p} |\alpha_i|$. Il s'ensuit des solutions dites parcimonieuses, c'est-à-dire que de nombreux coefficients sont égaux à zéro. Le problème d'optimisation s'écrit alors :

$$\min_{\alpha \in \mathbb{R}^p, \beta \in \mathbb{R}} \sum_{i=1}^{n} (y_i - (\langle \alpha, \boldsymbol{x}_i \rangle + \beta))^2 + \lambda \|\alpha\|_1$$

   (a) En utilisant la *classe linear_model.Lasso*, tracez l'évolution des coefficients du vecteur $\hat{\alpha}$ en fonction de la valeur du paramètre *lambda*. Quelles variables semblent le mieux expliquer la variable *Balance* du jeu de données ? Sont-elles les mêmes que celles trouvées à l'exercice précédent ? Comment se comportent les autres variables lorsque la valeur de *lambda* augmente ?

   (b) Trouvez par un moyen approprié la meilleure valeur pour le paramètre lambda. Apprenez ensuite un régresseur avec cette valeur sur l'ensemble du jeu de données *Credit* et calculez l'erreur au sens des moindres carrés sur ce même échantillon.

6. Quel modèle choisiriez vous entre la régression linéaire, la régression Ridge et la régression Lasso pour prédire la variable *Balance* du jeu de données ?

Exercice 2

1. Simuler des données (taille 500) de classification binaire et deux features.
2. Découper les données en données d'apprentissage et données de test.
3. Programmer l'algorithme de Gradient Descent sur la régression logisitique et faire la prédiction des données de test. Quel score avez obtenu ? Essayez d'améliorer ce score.