TELECOM ParisTech **HLMA308**
2019/2020 **Nicolas Verzerlen Joseph Salmon**

Projet final : Apprentissage statistique

Pour ce travail vous devez déposer un **unique** fichier au format `nom_prenom.ipynb` sur le site moodle du cours (si moodle ne l'accepte pas, zipper cet unique fichier en fichier .zip)

Vous devez charger votre fichier sur Moodle, avant le dimanche 03/11/2019, 23h55.

La note totale est sur **20** points répartis comme suit :

— qualité des réponses aux questions : **15** pts,

— qualité de rédaction, de présentation et d'orthographe : **2** pts,

— indentation, style PEP8, commentaires adaptés, etc. : **2** pts,

— absence de bug : **1** pt.

Les personnes qui n'auront pas rendu leur devoir avant la limite obtiendront **zéro**.

Rappel : aucun travail par mail ne sera accepté !

- Comparaisons de différentes méthodes de classification -

On va comparer dans cette partie les différentes méthodes sur la base de donnée obtenue comme dans le premier TP avec les commandes suivantes :

```
from sklearn.datasets import load_digits
digits = load_digits()
X, y = digits.data, digits.target
```

1) Proposer quelques élements de synthèse de la base de données (visualisation, résumé de statistiques élementaires, etc.)

On suivra maintenant le protocole expérimental suivant : couper les données en deux parties 75% pour l'apprentissage et 25% pour la validation (donner la taille des deux blocs choisis). Sur la partie d'apprentissage on entraînera les méthodes suivantes :

— Naive Bayes

— LDA

— Régression logistique

— QDA

— KNN (en prenant comme nombre de voisins $k = 1$)

— KNN (en choisissant k par validation croisée (V-fold) avec $V = 6$)

— une autre méthode de votre choix

On validera leur performance en donnant :

2) la proportion d'erreurs de classification faite sur la partie des données gardée pour la validation

3) le score F1.

4) Pour les méthodes mentionnées, proposer une synthèse sous forme de tableau ou de graphique, avec les renseignements suivants :

   — temps de calcul en seconde pris par chaque méthode pour la partie apprentissage (pour l'entraînement sur les 79% des données)

   — temps de calcul en seconde pris par chaque méthode pour la partie validation (sur les 21% restants)

   — pourcentage d'erreurs de classification de chaque méthode

   — le score F1.

5) On affichera les matrices de confusion associées : celles de la meilleure et de la pire des méthodes obtenues (au sens du nombre d'erreurs commises) parmi celles étudiées. Commentez vos résultats.

6) En s'inspirant de http://scikit-learn.org/stable/auto_examples/model_selection/plot_learning_curve.html afficher la courbe d'apprentissage (en : *learning curve*) pour la méthode KNN sur le même jeu de données [1].

7) Proposer un (cours) paragraphe synthétisant l'ensemble de vos expériences ci-dessus.

8) Reprendre tout l'analyse précédente sur un problème de classification mais pour une autre base de données, de votre choix, avec comme contrainte au moins $n \geq 150$ observations, $p \geq 10$ (nombres de variables explicatives), $q \geq 3$ (nombre de classes).

   Vous pourrez proposer les données de votre choix en automatisant leur téléchargement (depuis votre notebook) afin que le lancement du notebook soit executable sans avoir les données en local. En particulier, vous ne devrez pas déposer les données sur Moodle.

   Une piste (mais tout autre source est la bienvenue) pour trouver des donnés :

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

1. pour une version de `sklearn` > version 0.18, voir http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.learning_curve.html