

**Département informatique**

**MST : Systèmes intelligents et réseaux**

**Module : Reconnaissance des formes**

TP : KPPV & RB

***Préparé par :***

* ***Kawthar mssiaidi***
* ***Salma mekaoui***
* ***Hajar lazrak senhaji***

***Encadré par : Pr. Kharroubi Jamal***

***13/06/2021***

l’objectif

Dans ce TP on va implémenter K-NN sur un vrai jeu de données pour faire une classification multi-classes.

* Pour suivre ce TP il faut disposer de la bibliothèque **Sickit-Learn**.

On utilisera le célèbre jeu de données **MNIST.**

Scikit-Learn vient avec un ensemble de jeux de données prêt à l’emploi pour des fins d’expérimentations. Ces dataset sont regroupés dans le package sklearn.datasets

**Notre classificateur et dans : « sktlearn.neighbors »**

jeu de donne

MNIST est une base de données étiquetée propice pour un apprentissage supervisé. Dans l’image ci-dessus, pour chaque chiffre, on a sa représentation sous forme d’image ainsi que son étiquette. Par exemple, pour le dernier chiffre en bas à droit, l’étiquette vaut 9 vu qu’il s’agit du chiffre 9. La représentation de ces chiffres est normalisée à travers tout le jeu de données MNIST. Ainsi, chaque chiffre est codé dans un format 8 pixels \* 8 pixels. En plus, chaque pixel peut prendre une valeur de 0 à 255. Cette plage de valeurs représente le niveau de gris [Grayscale](https://fr.wikipedia.org/wiki/Niveau_de_gris). En d’autres terme, chaque représentation **d’une image est une matrice** de dimension .Le jeu de données MNIST présent par défaut dans la librairie [Scikit Learn](http://scikit-learn.org/stable/), comporte un sous-ensemble de la “vraie” base de données MNIST. Le sous-ensemble comporte  chiffres que nous diviserons par la suite en deux sous ensembles : d’entrainement et de test.

La réalisation:











