

Formation Machine Learning

Atelier Pratique AP-ML8

ALIASE

Atelier Pratique AP-ML8 : Exercice ML8.5

Objectif : Reprendre l'exercice de prediction des **prix des voitures**, et introduire la **selection de variables**

1) Créer 4 fonctions dans un fichier *module_Features_Selection.py* :

FS1(X,Y) : Feature selection utilisant *VarianceThreshold()* , retourne le nouveau dataset

FS2(X,Y) : Feature selection utilisant *SelectKBest()* , retourne le nouveau dataset

FS3(X,Y) : Feature selection utilisant *SelectFromModel()* , retourne le nouveau dataset

FS4(X,Y) : Feature selection utilisant *RFECV()* , retourne le nouveau dataset

2) Dans programme principal, créer une fonction *Features_Selection(X, Y, methode=i)* où **i** = 1, 2, 3 ou 4

si **i** == 1 alors appel FS1(X, Y)

si **i** == 2 alors appel FS2(X, Y)

si **i** == 3 alors appel FS3(X, Y)

si **i** == 4 alors appel FS4(X, Y)



SOLUTION

Atelier Pratique AP-ML8 : Exercice ML8.6.4

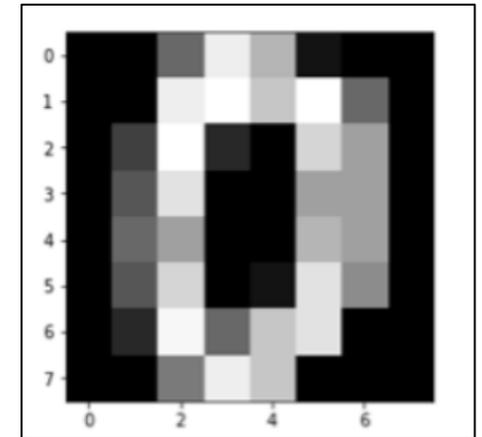
Objectif: Trouver le meilleur modèle KNN avec le Dataset des digits

1) importer une base de données de chiffre

```
digits = load_digits()
```

```
X = digits.data
```

```
Y = digits.target
```



2) Définir une fonction *meilleure_modele()* : retourne le meilleur modèle trouvé par GridSearchCV()

3) Créer un model KNN avec les paramètres par défaut : *model1*

4) Créer un model KNN avec les paramètres optimisés : *model2*

5) Evaluer les deux modèles sur *x_test*, *y_test* et afficher leurs scores



SOLUTION

Atelier Pratique AP-ML8 : Exercice ML8.6.5

Objectif : Trouver le meilleur modèle **SGDRegressor** avec le Dataset des **autos**

1) Charger le dataset à partir du fichier **autos.csv**, en remplaçant les "?" par **NaN**

2) Trouver le meilleur modèle **SGDRegressor** à l'aide de **GridSearchCV()**

en testant les paramètres **max_iter** et **eta0** avec les valeurs suivantes :

```
ITERATIONS = np.arange(1000, 9001, 1000)      # [ 1000, 2000, 3000, ..., 9000 ]
```

```
ETATO       = np.arange(0.001, 0.009, 0.001)  # [ 0.001, 0.002, 0.003, ..., 0.008 ]
```

3) Afficher les meilleurs paramètres obtenus

4) Evaluer le modèle sur **x_test**, **y_test**, et afficher le score

5) Visualiser les courbes de prédictions

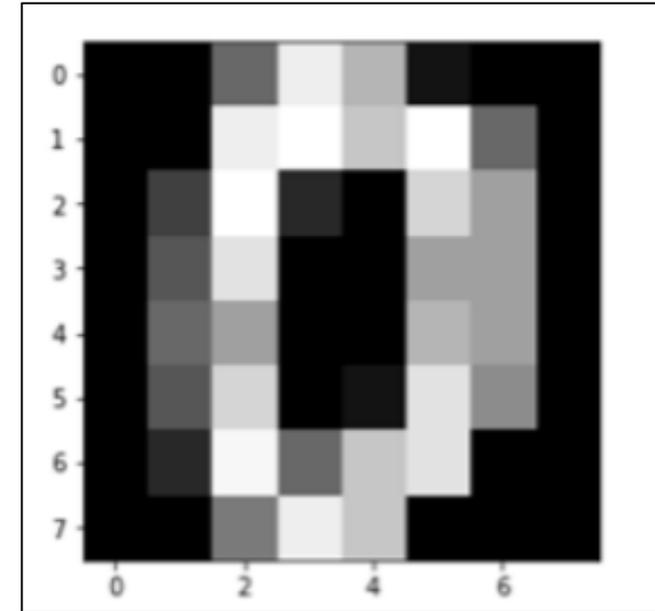


SOLUTION

Atelier Pratique AP-ML8 : Exercice ML8.7.2

Objectif : Exploiter la matrice de confusion. modèle KNN avec le Dataset des digits

- 1) Compléter l'exercice 8.6.4 en affichant la *matrice de confusion*
- 2) Avec une boucle for, compter le nombre d'erreurs de prédiction dans la matrice de confusion
- 3) Comparer avec le nombre d'erreurs obtenus avec celui obtenu via la fonction *Evaluate()*
- 4) Obtenir les scores sur Training Set et sur Validation Set via la fonction *learning_curve()* et tracer les courbes d'apprentissage



SOLUTION

Atelier Pratique AP-ML8 : Quiz

Quiz : Machine Learning, Objectif pédagogique 5

