

Formation Machine Learning

Atelier Pratique AP-ML8

ALIASE

Atelier Pratique AP-ML8 : Exercice ML8.6.4

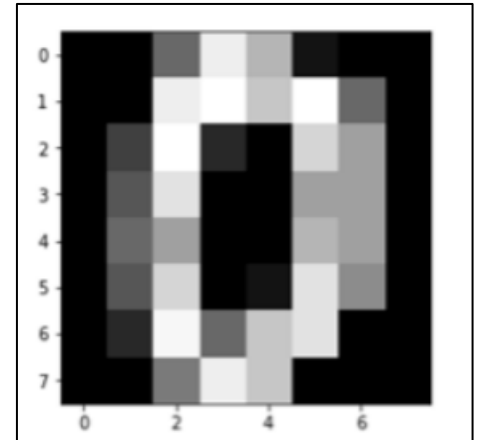
Objectif: Trouver les meilleurs paramètres du modèle KNN avec le Dataset des digits

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



Voir MLExo8.6.4.py

Atelier Pratique AP-ML8 : Exercice ML8.6.5

Objectif : Trouver les meilleurs paramètres du modèle `SGDRegressor` avec le Dataset des autos

1) Charger le dataset à partir du fichier `autos.xlsx`, 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]

`ETAT0 = 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

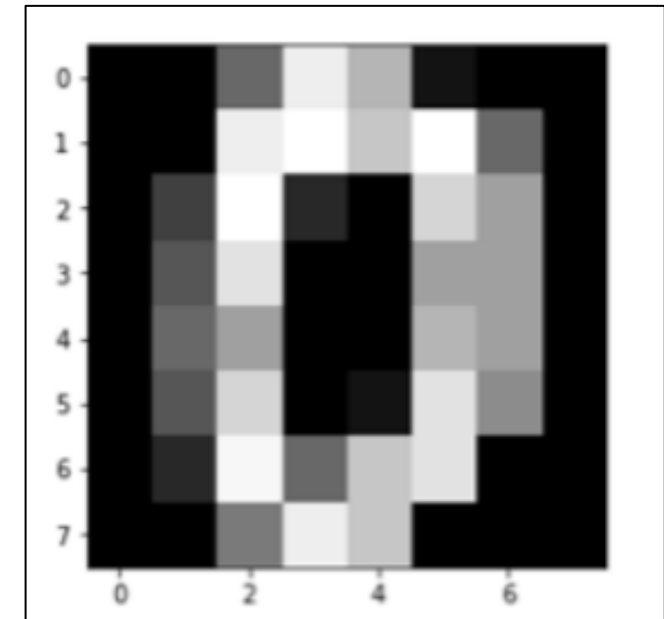


Voir MLExo8.6.5.py

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



Voir MLExo8.7.2.py