

Formation Machine Learning

Atelier Pratique AP-ML1

Atelier Pratique AP-ML1 : Exercice ML1.4.1

Objectif : *Prédire le prix des véhicules en fonction des caractéristiques techniques*

- Refaire le test de l'exercice ML1.4.1 avec la méthode-2 d'encodage qui consiste à utiliser le transformer *LabelEncoder()*
- dataset : *autos.xlsx*

Solution



Voir MLExo1.4.1.py

Atelier Pratique AP-ML1 : Exercice ML1.4.2 - A

Objectif : *Prédire le prix des véhicules en fonction des caractéristiques techniques*

- Refaire le test de l'exercice ML1.4.1 avec la méthode des *Moindre Carrés*.

Elle consiste à utiliser un modèle de type *LinearRegression*

- dataset : *autos.xlsx*

Solution



Voir MLExo1.4.2.py

Atelier Pratique AP-ML1 : Exercice ML1.4.2 - B

Objectif : *Prédire le prix des véhicules en fonction des caractéristiques techniques*

- Refaire le test de l'exercice ML1.4.1 avec les 3 méthodes de normalisation :
 - *MinMaxScaler()*
 - *StandardScaler()*
 - *RobustScaler()*
- dataset : *autos.xlsx*

Solution



Voir MLExo1.4.2.py

Atelier Pratique AP-ML1 : Exercice ML1.4.4

Objectif : *Prédire le prix des véhicules* en utilisant le modèle **SVR**

dataset : *autos.xlsx*

```
from sklearn.svm import SVR

# Création du modèle
model = SVR ( C=10000 )      # C = paramètre de Régularisation. Sera abordé dans le module 8.
                             # Plus il est grand, plus le modèle va minimiser l'erreur de prédiction
```

Solution



Voir MLExo1.4.4.py

Atelier Pratique AP-ML1 : Exercice ML1.6

Objectif : *prédire la progression du diabète chez les patients un an après les tests initiaux.*

Dataset :

```
from sklearn.datasets import load_diabetes  
  
diabetes = load_diabetes()  
  
X = diabetes.data  
  
Y = diabetes.target
```

Atelier Pratique AP-ML1 : Exercice ML1.6

Dataset :

Variables explicatives : (x1, x2, ..., x10)

- **Âge** : âge en années.
- **Sexe** : facteur biologique du sexe.
- **IMC** : indice de masse corporelle, qui donne une idée du poids corporel.
- **PA** : pression artérielle moyenne.
- **S1** : taux de cholestérol total (TC).
- **S2** : taux de lipoprotéines de basse densité (LDL).
- **S3** : taux de lipoprotéines de haute densité (HDL).
- **S4** : taux de cholestérol / HDL.
- **S5** : niveau de log de la triglycéride.
- **S6** : taux de glucose.

Variable Cible :

- **Progression** : mesure quantitative de la progression de la maladie diabétique un an après les indices de base.

Solution



Voir MLExo1.6.py

Atelier Pratique AP-ML1 : Exercice ML1.9

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

1) Utiliser la fonction *Features_Selection(X, Y, methode)* fournie dans *module_DPP.py* : # Data PreProcessing

methode=1 : Feature selection utilisant *VarianceThreshold()* , retourne le nouveau dataset

methode=2 : Feature selection utilisant *SelectKBest()* , retourne le nouveau dataset

methode=3 : Feature selection utilisant *SelectFromModel()* , retourne le nouveau dataset

methode=4 : Feature selection utilisant *RFECV()* , retourne le nouveau dataset

2) Dans programme principal, appeler la fonction *Features_Selection(X, Y, methode=i)* avec **i** = 1, 2, 3 ou 4

Solution



Voir MLExo1.9.py

Atelier Pratique AP-ML1 : Exercice ML1.10

Objectif :

Reprendre l'exercice de prediction des **prix des voitures** en :

- normalisant les **labels** également,
- et en présentant les résultats des prédictions après **dénormalisation** des réponses

Solution



Voir MLExo1.10.py

Atelier Pratique AP-ML1 : Exercice ML1.14

Objectif : Reprendre l'exercice de prediction des **prix des voitures** en :

1- analysant la colonne **labels** pour déterminer :

- . S'il est **recommandé de normaliser les labels ou pas**

- . Si oui, **quelle fonction de normalisation** (StandardScaler, MinMaxScaler, RobustScaler)

→ Utiliser la fonction **diagnostic_normalisation_Y()** fournie dans **module_DPP.py**

2- en présentant les résultats des prédictions après **dénormalisation** des réponses

Solution



Voir MLExo1.14.py

Atelier Pratique AP-ML1 : Exercice ML1.15

Objectif : Reprendre l'exercice de prediction des **prix des voitures** en :

1- analysant la colonne **labels** pour déterminer :

- . S'il est **recommandé de normaliser les labels ou pas**
- . Si oui, **quelle fonction de normalisation** (StandardScaler, MinMaxScaler, RobustScaler)
- Utiliser la fonction **diagnostic_normalisation_Y()** fournie dans **module_DPP.py**

2- analysant les **features** pour déterminer **quelle fonction de normalisation**

- Utiliser la fonction **diagnostic_normalisation_X()** fournie dans **module_DPP.py**

3- en présentant les résultats des prédictions après **dénormalisation** des réponses

