

Guide de projet de validation de: Algorithmique pour le BigData

Enseignante : Rakia JAZIRI

Réalisation : Nadia OUACIF

Sujet :

Prédire le cours d'une action en bourse CAC40

Données :

L'historique journalière d'une entreprise au cours des 5 ans,

Source :

CAC40 (Boursorama)

https://www.boursorama.com/bourse/actions/cotations/?quotation_az_filter%5Bmarket%5D=1rPCAC

Outils informatique utilisé : google colab, Pyspark

Les Bibliothèque utilisées :

```
import findspark
findspark.init()
from pyspark.sql import SparkSession
import pandas as pd
from pyspark.sql.functions import *
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.regression import LinearRegression
from pyspark.sql.functions import *
from pyspark.ml.evaluation import RegressionEvaluator
import matplotlib.pyplot as plt
import seaborn as sns
```

Les étapes de projet :

Etape 1 :

Configuration de l'environnement de travail, puisque moi j'utilise google colab, donc je dois installer mon environnement de travail ;

```
# innstall java
!apt-get install openjdk-8-jdk-headless -qq > /dev/null
```

```
# install spark (change the version number if needed)
!wget -q https://archive.apache.org/dist/spark/spark-3.0.0/spark-3.0.0-
bin-hadoop3.2.tgz
# unzip the spark file to the current folder
!tar xf spark-3.0.0-bin-hadoop3.2.tgz

!pip install -q findspark
!pip install library
!apt-get install library
```

Etape 2 :

Création des script.py chacuns contient sa fonction,

Mon projet à 4 script.py

1. utils1.py : pour instancier mon SparkSession
2. utils2.py : j'ai appelé toutes les bibliothèques que je vais utiliser dans ce projet
3. config_projet.py : dans ce script, j'ai appelée les deux premiers scripts, et il contient trois fonctions, une fonction qui appel Spark en appelant utils1.py, une fonction qui charge le dataset, et la fonction qui fait le traitement des donner, et l'apprentissage. Elle enregistre le résultat sous forme d'un fichier csv et la figure en format .png
4. execute.py : j'appelle config_projet.py, et y'a deux fonctions, une fonction qui exécute les fonctions de script config_projet.py lire le fichier de données et la fonction de prédiction ; la deuxième fonction et la fonction principale « main » il suffit de l'appeler et d'exécuter le code en utilisant la commande shelle suivante :

```
python execute.py arg1 arg2 avec arg1= chemin de fichier, arg2=
nom de l'entreprise choisi pour prédire son prix de fermeture dans
la bourse
```

5. J'exécute main on activant le terminal avec la commande `!bash`

```
python execute.py "/content/preprocessed_CAC40.csv", 'Hermès (Hermes
International'
```

Etape 3 :

Les résultats obtenus qui sont enregistrer automatique dans le fichier prediction_result.csv :

Attributes	Closing_Price	Date	prediction
[95.76,97.02,95.76]	97.02	2010-01-12	96.86789050397373
[96.09,96.09,96.09]	96.09	2010-02-16	96.11207268654255
[96.16,97.43,96.16]	97.43	2010-02-17	97.27649384919927
[96.34,96.34,96.34]	96.34	2010-02-10	96.36206383419177
[97.21,97.21,97.21]	97.21	2010-02-25	97.23203302801113
[97.52,97.52,97.52]	97.52	2010-02-05	97.5420220510962
[98.75,98.75,98.75]	98.75	2010-04-19	98.77197849753047
[99.29,99.29,99.29]	99.29	2010-02-02	99.31195937645282
[99.41,99.41,99.41]	99.41	2010-04-22	99.43195512732446
[99.8,99.8,99.8]	99.8	2010-05-06	99.82194131765726
[99.9,99.9,99.9]	99.9	2010-03-03	99.92193777671697
[101.35,101.35,10...	101.35	2010-03-04	101.37188643308255
[101.35,101.95,10...	101.95	2010-03-12	101.88893697228647
[101.8,101.8,101.8]	101.8	2010-05-04	101.8218704988512
[102.15,102.15,10...	102.15	2010-04-01	102.17185810556013
[102.25,102.25,10...	102.25	2010-03-16	102.27185456461982
[102.95,102.95,10...	102.95	2010-03-26	102.9718297780377
[103.4,103.4,103.4]	103.4	2010-03-08	103.42181384380633
[104.35,104.35,10...	104.35	2010-03-22	104.37178020487345
[104.55,104.55,10...	104.55	2010-03-24	104.57177312299284

RMSE w/ All Features: 1.083627

r2(All Features): 0.999949

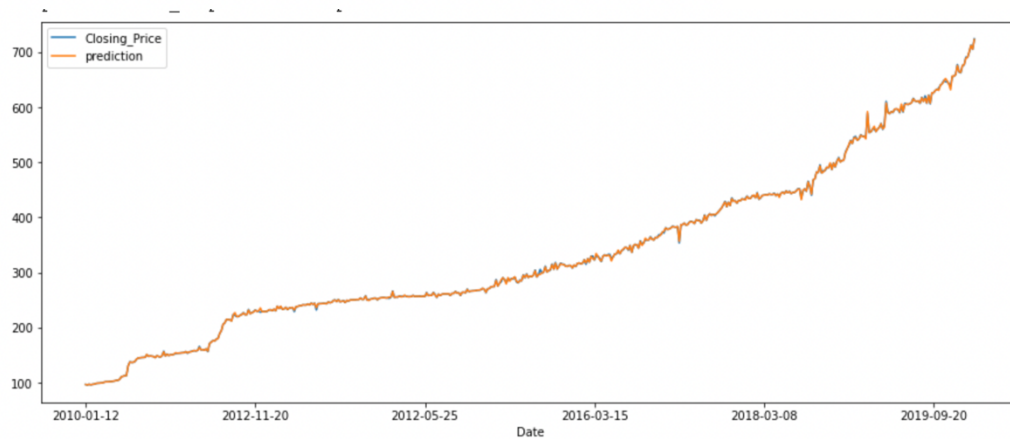


Fig : illustre le prix de fermeture(en bleu), et les prix de fermeture prédit (orange)