

Prévision des cours boursiers (actions) des entreprises du CAC 40.

➤ Documentation

La prévision du cours boursier est un sujet important pour les investisseurs. Elle est utilisée comme outil d'aide à la décision éclairant les investisseurs sur le meilleur moment pour acheter ou vendre des actions. La démarche repose sur l'hypothèse selon laquelle l'information fondamentale accessible au public dans le passé a des relations prédictives avec les rendements futurs des actions.

L'objectif du projet est de mettre en place un outil d'aide à la décision permettant de guider les décisions d'un investisseur.

Dans notre cas, on se concentre sur le cours boursier de l'entreprise CARREFOUR à l'aide des données du site Boursorama. Notre but est de récolter les données de la page grâce à notre outil.

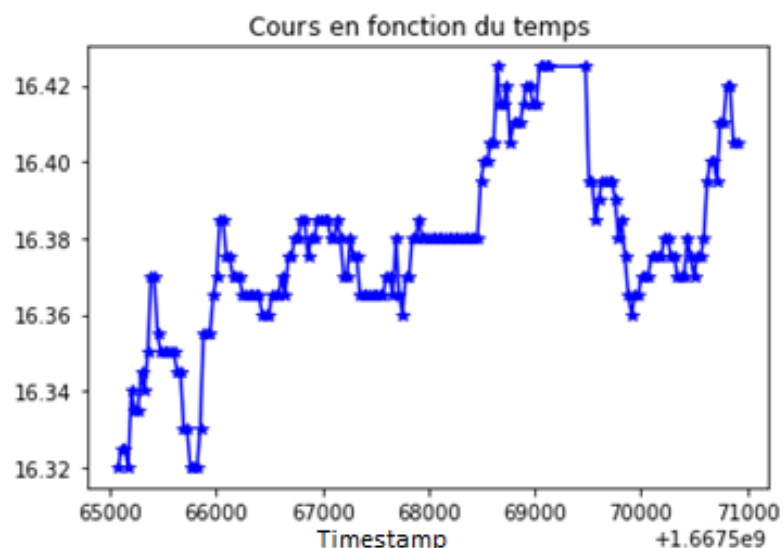
Pour cela il faut extraire les codes HTML de la page concernée, puis les passer en format texte. Ensuite à partir de ce texte, on cherche les balises qui nous intéressent. On veut le prix de l'action, qui est spécifié dans la balise `span` avec la class `"c-instrument c-instrument--last"`. C'est cette balise qui est mise à jour à chaque fois, donc on récupère le dernier cours boursier à chaque fois.

On utilise une fonction qui fait tourner notre boucle en fonction du temps réel. Ici on a choisi de faire tourner notre boucle jusqu'à avoir 180 valeurs récoltées, sachant que le site Boursorama met à jour ses données toutes les 30 secondes, on sait alors que pour avoir nos 180 valeurs le Crawler devra tourner pendant 1h30min.

On sait également que ces valeurs sont mises à jour uniquement durant les jours et les horaires d'ouverture de la bourse. C'est-à-dire hors week-end et jours fériés, entre 9h00 et 17h30. On a décidé de faire tourner notre Crawler **le jeudi 4 novembre aux alentours de 13h30**, on aura donc bien des données exploitables.

Le but ici étant de prédire à partir de données que l'on récolte, on va stocker les données que l'on obtient par notre Crawler dans un fichier CSV pour pouvoir travailler dessus sans avoir à le faire tourner à chaque fois. Attention, on aura donc les valeurs du 04/11/2022 entre 13h30 et 15h00, on va donc prédire à partir de valeurs fixées dans le temps. Si on veut prédire par exemple pour dans 3 mois, le mieux serait de faire tourner le Crawler à ce moment pour avoir des données récentes car on ne sait pas ce qu'il peut arriver entre-temps (crash market, etc.).

Voilà ce qu'on obtient :



On observe donc que le cours varie entre 16.32 et 16.42 durant le laps de temps sur lequel on a décidé de travailler. La valeur en abscisse est exprimée en Timestamp (nombre de secondes écoulées depuis le 1^{er} janvier 1970 à minuit).

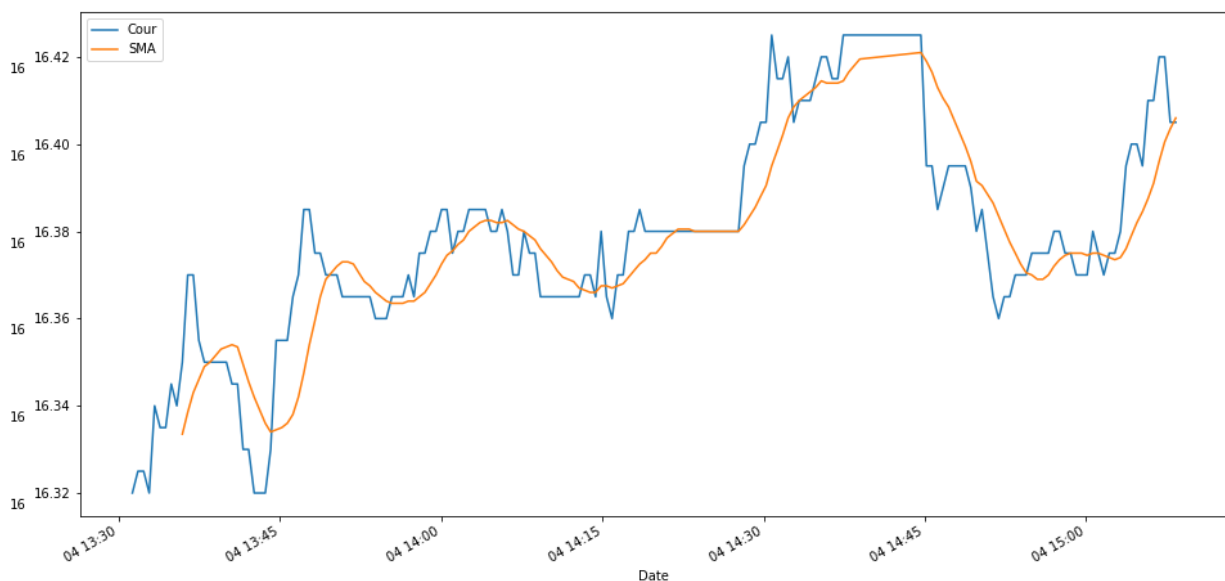
Voici le fichier CSV dans lequel on stocke les valeurs récoltées :

```
In [25]: df
Out[25]:
```

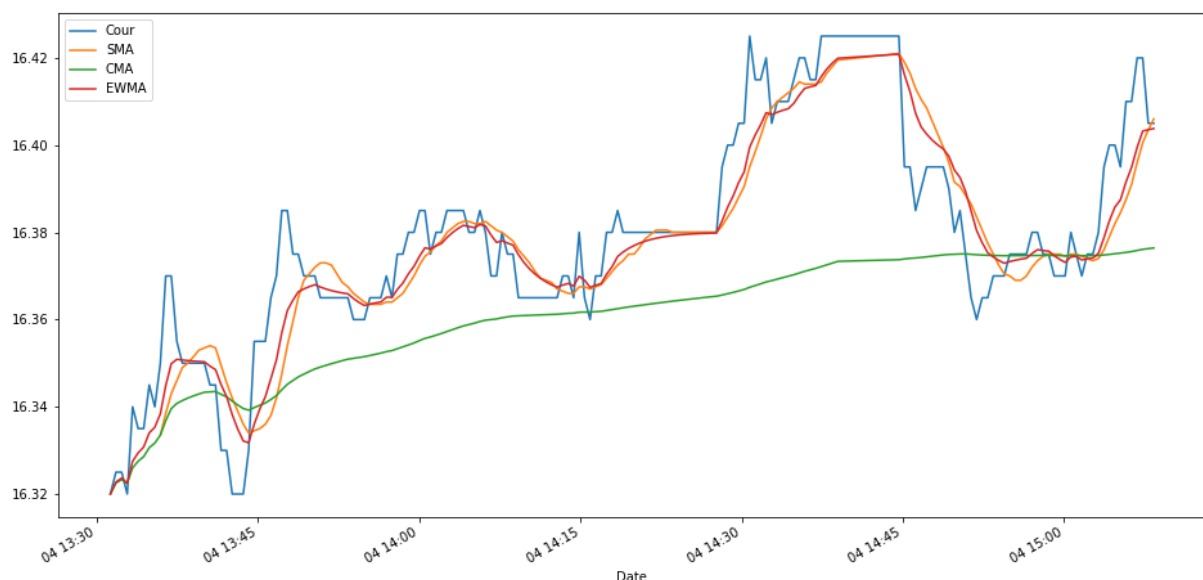
	Timestamp	Date	Cours
0	1.667565e+09	2022-11-04 13:31:16	16.320
1	1.667565e+09	2022-11-04 13:31:47	16.325
2	1.667565e+09	2022-11-04 13:32:19	16.325
3	1.667565e+09	2022-11-04 13:32:50	16.320
4	1.667565e+09	2022-11-04 13:33:20	16.340
..
175	1.667571e+09	2022-11-04 15:06:17	16.410
176	1.667571e+09	2022-11-04 15:06:48	16.420
177	1.667571e+09	2022-11-04 15:07:19	16.420
178	1.667571e+09	2022-11-04 15:07:50	16.405
179	1.667571e+09	2022-11-04 15:08:21	16.405

[180 rows x 3 columns]

Faisons quelques manipulations en calculant la moyenne mobile simple :



Même chose mais on y ajoute moyenne mobile cumulée et exponentielle :

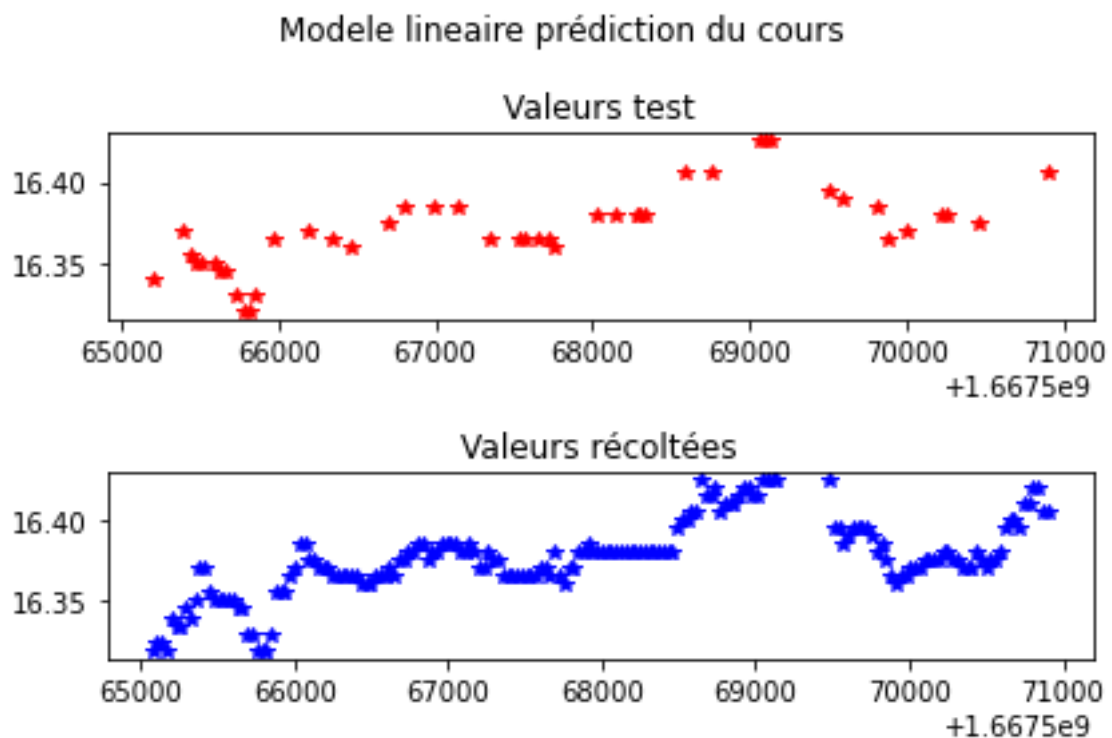


On observe que la meilleure façon est donc la moyenne mobile simple et exponentielle, on a la même dynamique que l'évolution du cours de l'action, donc une fiabilité assez précise.

On note qu'il faut tout de même un minimum de données à prendre en compte pour calculer les moyennes à chaque temps t , ici on a choisi de prendre 10 valeurs pour chaque moyenne. Donc le cours d'une action varie ici par rapport à ces 10 dernières valeurs soit les 5 dernières minutes qui se sont écoulées.

À l'aide des données récoltées par notre Crawler, on va créer un modèle pour faire des prévisions. Tout d'abord, on va essayer notre modèle en simulant les mêmes données que l'on a pour les comparer et observer la différence des deux.

On a donc ces deux graphiques :



On observe la même dynamique entre les valeurs tests et les valeurs que l'on a déjà. C'est bon signe, car cela veut dire qu'on s'approche assez bien de la réalité grâce à notre modèle de prédiction. On va tout de même vérifier mathématiquement si c'est correct, on fait donc la moyenne entre la différence au carré des valeurs réelles et celles prédites par notre modèle :

```
np.mean((y_test-y_pred)**2))
Out[25]:
Cour    0.00024
dtype: float64
```

Le résultat est très proche de 0, on peut donc dire que le modèle est assez fiable.

Pour la suite de la prédiction, on va supposer vouloir prédire les valeurs qu'on aurait pu avoir après l'arrêt du Crawler. Dans notre cas on s'est arrêté à 15h08min21sec, on va donc partir de ce temps et essayer de prédire sur les 5 prochaines minutes la variation du cours. On aura donc 10 valeurs que l'on va prédire grâce à notre modèle. Ces valeurs seront également stockées dans un fichier CSV afin de garder les valeurs. Voici les valeurs que l'on trouve :

```
In [32]: new_dp
Out[32]:
```

	Timestamp	Date	Prediction
0	1.667571e+09	2022-11-04 15:08:21	16.401012
1	1.667571e+09	2022-11-04 15:08:51	16.401265
2	1.667571e+09	2022-11-04 15:09:21	16.401518
3	1.667571e+09	2022-11-04 15:09:51	16.401771
4	1.667571e+09	2022-11-04 15:10:21	16.402024
5	1.667571e+09	2022-11-04 15:10:51	16.402276
6	1.667571e+09	2022-11-04 15:11:21	16.402529
7	1.667571e+09	2022-11-04 15:11:51	16.402782
8	1.667571e+09	2022-11-04 15:12:21	16.403035
9	1.667571e+09	2022-11-04 15:12:51	16.403288

Ici on observe que le cours reste assez stable dans une fenêtre de 5 minutes. Essayons maintenant sur une plus grande durée :

```
In [34]: new_dp
Out[34]:
```

	Timestamp	Date	Prediction
0	1.667571e+09	2022-11-04 15:08:21	16.402517
1	1.667571e+09	2022-11-04 15:08:51	16.402783
2	1.667571e+09	2022-11-04 15:09:21	16.403048
3	1.667571e+09	2022-11-04 15:09:51	16.403314
4	1.667571e+09	2022-11-04 15:10:21	16.403579
..
95	1.667574e+09	2022-11-04 15:55:51	16.427739
96	1.667574e+09	2022-11-04 15:56:21	16.428005
97	1.667574e+09	2022-11-04 15:56:51	16.428270
98	1.667574e+09	2022-11-04 15:57:21	16.428536
99	1.667574e+09	2022-11-04 15:57:51	16.428801

[100 rows x 3 columns]

D'après le modèle, on aurait donc une augmentation du cours dans la prochaine heure.

Ce qu'on observe sur le passé a bien une influence sur le futur, les valeurs ne sont pas aléatoires on a une ligne directrice tout de même. L'action du cours ne passe pas subitement d'une valeur à une autre on a une progression successive.

À partir des valeurs passées, par exemple ici les 180 données relevées pendant 1h30, on peut prédire ce qu'il pourrait se passer dans l'heure qui suit soit les 120 prochaines valeurs.

Evidemment il existe plusieurs méthodes et approches pour la prédiction, avec plus ou moins de fiabilité. En revanche, il y a toujours une marge de confiance et il ne faut pas s'y fier à 100% car ça reste de la prédiction et donc on ne peut pas réellement savoir exactement l'action du cours au cours du temps.

➤ Mon avis

J'ai trouvé ce projet intéressant car c'est un outil que l'on peut utiliser pour divers domaines et pas seulement pour étudier la variation du cours d'une action. C'est un outil très pratique et assez technique à créer, mais après de nombreuses recherches et de comparaison j'ai abouti à un résultat qui m'a satisfait.

Cependant par rapport à la prédiction j'ai eu un peu plus de mal car il y a différentes méthodes qui existent que j'ai essayé mais j'ai eu des résultats pas très concluant. Je me suis donc documenté autrement à travers des vidéos de démonstration et je me suis orientée vers celle que je vous présente. C'était quand même assez compliqué car il me fallait des valeurs récentes sinon on avait une prédiction non représentative de la réalité et donc pas assez fiable pour conclure. Cela reste tout de même abordable. Une approche par la méthode ARIMA aurait été meilleure mais je n'ai pas réussi à l'appliquer.

J'ai bien aimé travailler sur ce sujet, cela m'a permis d'en apprendre davantage sur la bourse et de maîtriser un outil de collecte très utile pour la suite de mon parcours.

➤ Sources

<https://www.boursorama.com/cours/1rPCA/>

https://www.cours-gratuit.com/tutoriel-python/tutoriel-python-comment-scrapper-un-site-en-python#_Toc60090618

<https://mrmint.fr/regression-lineaire-python-pratique>

<https://www.youtube.com/watch?v=ACi3aiJRIP8&t=4017s>

<https://www.askpython.com/python/examples/python-predict-function>

<https://docs.python.org/3/reference/index.html>

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

https://scikit-learn.org/stable/modules/linear_model.html

<https://www.geeksforgeeks.org/how-to-calculate-moving-average-in-a-pandas-dataframe/#:~:text=In%20Python%2C%20we%20can%20calculate,a%20parameter%20in%20the%20function%20>