Data pre-processing

Prédiction des cours des actions commerciales

Réalisé par :

Mohamed Bouibauan
Mohamed Maataoui
Jamal Rebii
Imane Achkaoukaou

Encadré par :

Pr. Mohamed Lazaar

Table des matières

1 Chapitre 1 : Contexte géneral du projet **5**
1.1 Présentation du sujet 5
1.2 Conduite du projet 5
1.3 Acquisition de données 6
1.4 Description des variables 6

2 Chapitre 2 : Outils et techniques utilisés **7**
2.1 Langage de programmation 7
2.1.1 Python
. 7
2.1.2 Jupiter
. 7
2.1.3 Google Colab
. 8
2.2 Les bibliothèques 9
2.2.1 NumPy 9
2.2.2 Matplotlib : 9
2.2.3 Pandas 9
2.2.4 Selenium 10
2.2.5 RequestS 10
2.2.6 Scikit-learn 10

3 Chapitre 3 :Data scraping et crawling **11**
3.1 Scraping 11
3.2 Crawling 11
3.3 Scrapinge Vs Crawling 12
3.4 Construction de la base de données (scraping et crawling) 13
3.4.1 Scrape the Trending stocks 14
3.4.2 Scrape the trending Stocks by Popularity 15
3.4.3 Scrape the trending Stocks by Sector 16
3.5 Dataset après la concaténation 17

4 Chapitre 4 :Présentation des algorithmes utilisés **18**
4.1 Support Vector Regression 18
4.1.1 SVR à Marge dure 18
4.1.2 Formulation primal 19
4.1.3 Formulation dual 19

4.1.4 SVR à Marge souple . . . 20
4.1.5 Formulation primal . . . 20
4.1.6 Formulation dual . . . 20
4.2 Linear regression . . . 21
4.2.1 Fonction hypothèse . . . 21
4.2.2 Algorithme . . . 22
4.3 Algoritmes de réduction de dimentionalité . . . 23
4.3.1 Sélection de caractéristiques . . . 23
4.3.1.1 Approches filtres . . . 23
4.3.1.1.1 Critère de Fisher . . . 23
4.3.1.1.2 Critère de Variance . . . 24
4.3.1.1.3 Critère d'information mutuelle . . . 24
4.3.1.2 Approches enveloppantes . . . 25
4.3.1.2.1 Sequential forward selection (SFS) . . . 26
4.3.1.2.2 Sequential backward selection (SBS) . . . 27
4.3.1.3 Approches integres . . . 28
4.3.1.3.1 Random forest . . . 28
4.3.2 Transformation de données . . . 30
4.3.2.1 ACP . . . 30

5 Chapitre 5 :Prétraitement de dataset 31
5.1 Compréhension de l'ensemble des donnéess . . . 31
5.2 Description des donnees . . . 31
5.3 La variation du prix en fonction du variable Date (2018-2022) . . . 32
5.4 La variation du prix avec d'autres variables . . . 33
5.5 La variation des variables entre eux . . . 34
5.6 Data cleaning . . . 35
5.6.1 Structuration du dataset après scraping . . . 35
5.7 Normalisation des données . . . 36
5.8 La corrrélation avant la réduction de dimentionalité . . . 36
5.9 La prédiction du prix avant la réduction de dimentionnalité 37
5.9.1 La prédiction avec Support Vector Machine SVR . . . 37
5.9.2 La prédiction avec Linear regression . . . 37
5.10 La réduction de dimentionalité et sélection des variables . . . 38
5.10.1 Sélection des caractéristiques . . . 38
5.10.1.1 Approches filtres . . . 38
5.10.1.1.1 Critère de Fisher . . . 38
5.10.1.1.2 Critère de variance . . . 39
5.10.1.1.3 Critère d'information mutuelle . . . 40
5.10.1.2 Approches enveloppantes . . . 41
5.10.1.2.1 SFS . . . 41
5.10.1.2.2 SBS . . . 42

5.10.1.3 Approches intégres 43
5.10.1.3.1 Random Forest 43
5.10.2 Transformation de données 44
5.10.2.1 ACP 44
5.10.3 Tableau de comparaison 45

Introduction

Le marché boursier est connu pour sa volatilité, son dynamisme et sa non linéarité. Cette instabilité a eu comme conséquence qu'une prévision précise des cours des actions commérciaux est extrêmement difficile, voire impossible, à cause de plusieurs facteurs tels que la situation politique, les conditions économiques mondiales, les événements innatendues et la performance financière d'une entreprise.

Cela signifie qu'il y a beaucoup de facteurs à prendre en considération et beaucoup de données à trier. Ainsi, les analystes financiers, les chercheurs et les scientifiques des données continuent d'explorer des techniques d'analyse pour détecter les tendances des marchés boursiers. Ce qui a donné naissance au concept du trading algorithmique, qui utilise des stratégies automatisées et préprogrammées pour exécuter des ordres.

Dans ce projet, nous allons se baser sur les données historiques extraites du fameux site web *"Ivesting"* pour prévoir les cours des actions des géants du marché mondial comme Apple, Google, Tesla et Amazon. Pour se faire, nous avons choisi d'adopter le plan suivant : premièrement, nous présenterons le contexte général du projet, l'environnement logiciel ainsi que les notions du data scraping et crawling, ensuite nous expliquerons les différents algorithmes utilisés dans la prévision et la réduction de dimensionnalité et finalement nous établirons les tableaux et les figures des résultats finaux.

1 Chapitre 1 : Contexte géneral du projet

1.1 Présentation du sujet

Le prix d'un titre financier varie en fonction de l'offre et de la demande sur les marchés. Ces deux facteurs fluctuent au gré de nombreux événements pouvant survenir de manière inattendue. Ils peuvent être propres à l'entreprise ou liés à son environnement.

Dans le premier cas, le cours d'une action peut s'élever après l'annonce d'une acquisition surprise, ou baisser si elle lance un profit warning.

Dans le second cas, le cours des actions d'une société cotée est susceptible d'augmenter lorsqu'un de ses concurrents connaît une baisse de ses parts, ou chuter si un pays adopte une réglementation qui n'est pas favorable à ses activités principales.

Certes, les investisseurs ne peuvent pas prévoir l'avenir. Cependant plusieurs indicateurs peuvent permettre de se faire une opinion sur le cours qu'une action peut d'atteindre dans les jours à venir.

1.2 Conduite du projet

La planification est une étape très importante qui permet d'assurer le bon déroulement du projet tout au long des phases de la réalisation du projet d'une façon à respecter les différentes contraintes.

Ce projet a été réalisé dans une durée de 20 jours en suivant les étapes suivantes :
-Confirmation du choix du sujet
-Documentation et collecte des informations nécessaires
-Suivi des cours nécessaires pour la réalisation du projet
-Conception du projet
-Développement de l'application
-Rédaction du rapport (tout au long du projet)

1.3 Acquisition de données

Nos tableaux de données ont été créés grâce au web scraping du site web "Investing". Ils s'agissent de 4 tableaux contenant plusieurs variables (date, price, open, close...) qui permettent de prédire le cours des actions des entreprises leaders dans le domaine technologique, on parle ici de Tesla, Apple, Google et Amazon.

1.4 Description des variables

Chacun de ces tableaux de données contient les 12 variables suivantes :
-**Date :** la date de la journée pendant laquelle cette observation a été faite
-**Price :** le cours auquel l'action a clôturé la veille
-**Open :** il s'agit des cours d'ouverture
-**High :** représente le cours le plus haut de la journée
-**Low :** représente le cours le plus bas de la journée
-**Vol :** il s'agit du nombre de titres, d'actions échangés, au jour et à l'heure de la consultation de la fiche de l'action
-**Change :** le taux de change est exprimée en pourcentage, par rapport au cours de clôture de la veille
-**coef**
-**state**
-**activity_in**
-**activity_out**
-**company_name :** le nom de l'entreprise propriétaire des actions

2 Chapitre 2 : Outils et techniques utilisés

Après avoir mis notre projet dans son contexte général, nous allons décrire, de façon précise, dans ce chapitre les étapes suivies ainsi que les outils utilisés pour mettre un modèle d'apprentissage automatique (régression) après les étapes de scraping et crawling

2.1 Langage de programmation

Nous avons choisi d'utiliser Python et ses différentes bibliothèques pour la réalisation de ce projet, Spider, Jupiter et Google Colab pour les environnements du travail.

2.1.1 Python

Python est le langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. -Python dispose d'un système de types dynamiques et d'une gestion automatique de la mémoire, ainsi que d'une bibliothèque standard vaste et complète.

2.1.2 Jupiter

Jupyter Notebook (anciennement IPython Notebooks) est un environnement de programmation interactif basé sur le Web permettant de créer des documents Jupyter Notebook. Le terme "notebook" peut faire référence à de nombreuses entités différentes, adaptées au contexte, telles que l'application web Jupyter, le serveur web

2.1.3 Google Colab

Google Colab ou Colaboratory est un service cloud, offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l'apprentissage automatique.

Réalisé par :

Mohamed Bouibauan
Mohamed Maataoui
Jamal Rebii
Imane Achkaoukaou

Encadré par :

Pr. Mohamed Lazaar

2.2 Les bibliothèques

2.2.1 NumPy

NumPy est une bibliothèque pour langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

2.2.2 Matplotlib :

Matplotlib est une bibliothèque du langage de programmation Python destinée à tracer et visualiser des données sous formes de graphiques. Elle peut être combinée avec les bibliothèques python de calcul scientifique NumPy et SciPy.

2.2.3 Pandas

Pnadas est une librairie python qui permet de manipuler facilement des données à analyser : manipuler des tableaux de données (colonnes) et d'individus (lignes).

2.2.4 Selenium

Selenium webDriver : un Framework web qui nous permet d'exécuter des tests multinavigateurs. Cet outil est utilisé pour automatiser les actions humaines comme cliquer sur un bouton, remplir un champ de texte, cocher une case...

2.2.5 RequestS

Requests : est aussi un Framework web permettant de 'parser' les éléments à partir une page html ;

2.2.6 Scikit-learn

Scikit-learn est un module Python pour l'apprentissage automatique construit au-dessus de SciPy et est distribué sous la licence 3-Clause BSD. Le projet a été lancé en 2007 par David Cournapeau dans le cadre du Google Summer of Code et, depuis, de nombreux volontaires ont contribué au projet.

3 Chapitre 3 :Data scraping et crawling

3.1 Scraping

Le web scraping est une technologie qui permet de récupérer de manière automatisée des données provenant du web et qui permet d'extraire des données et des informations qui sont présentes sur des sites webs et de les transformer en d'autres formats plus exploitables comme excel ou csv . . .
Le web scraping possède de multiples intérêts.Il est aussi utilisé dans divers secteurs dont on peut citer : -la surveillance des concurrents
-le suivi des tendances des produits
-la prise de décision en matière d'investissement
-la compréhension de l'orientation du marché et l'évaluation de la valeur des biens

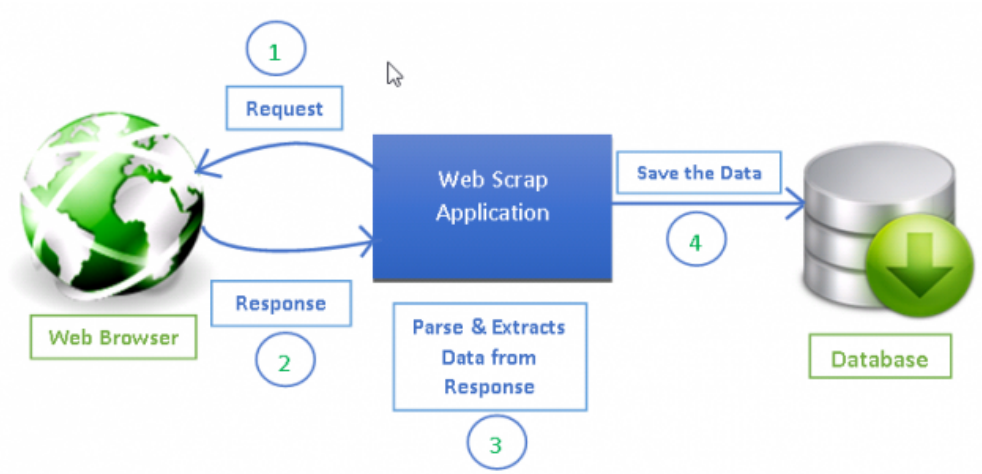


FIGURE 2 – Démarche de scraping

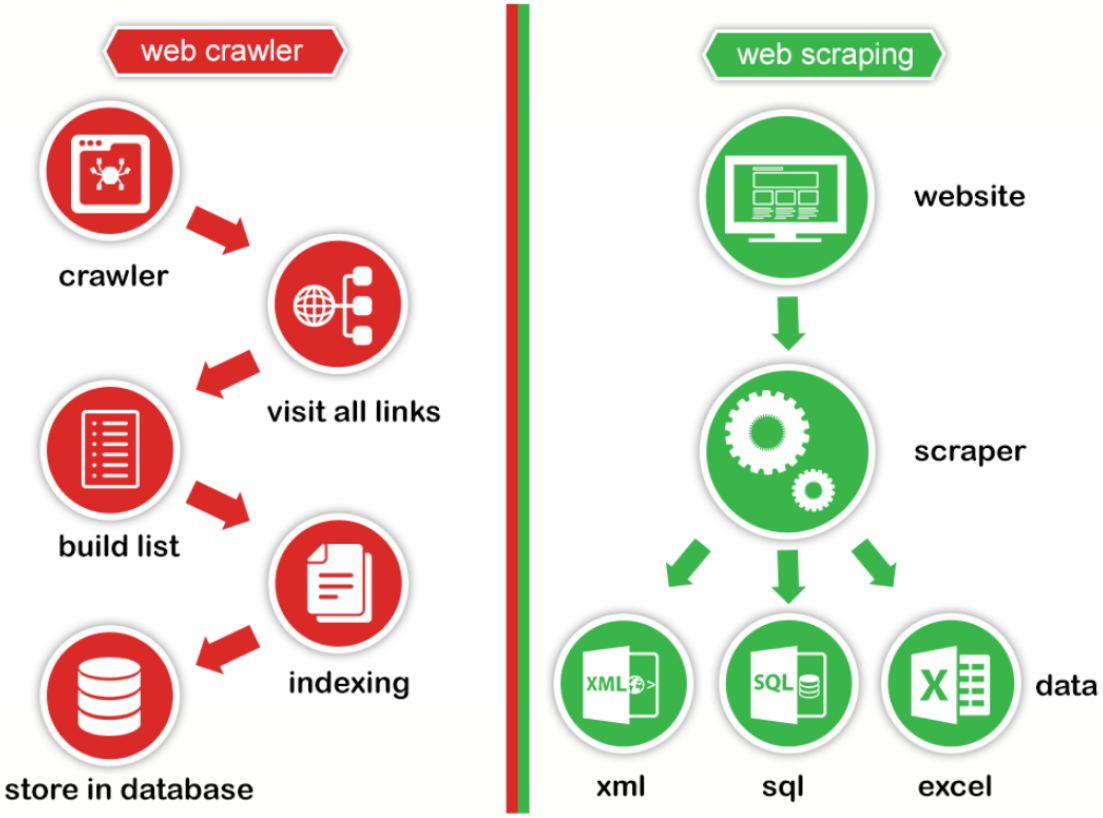
3.2 Crawling

Les termes de crawler, robot de crawl ou spider, désignent dans le monde de l'informatique un robot d'indexation. Concrètement, il s'agit d'un logiciel qui a comme principale mission l'exploration du Web afin d'analyser le contenu des documents visités et les stocker de manière organisée dans un index

3.3 Scrapinge Vs Crawling

-Le web scraping et le web crawling sont étroitement liés.

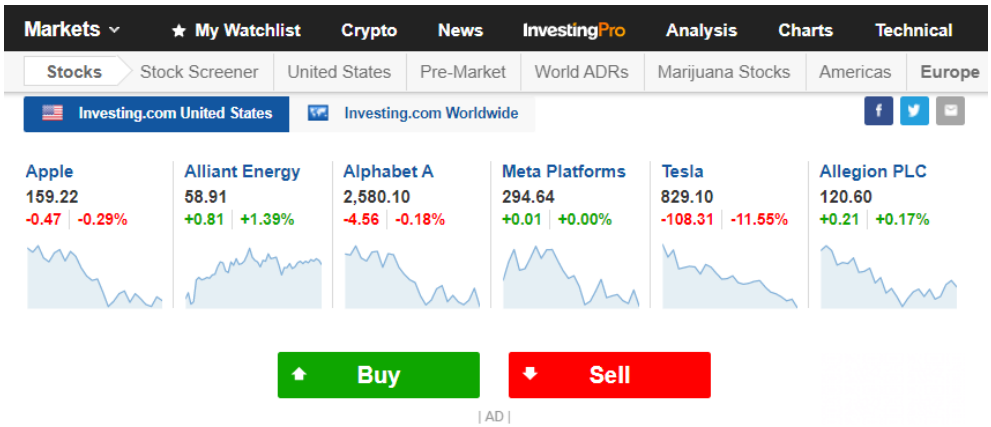
-Le web crawling indexe juste l'information en utilisant des robots, alors que le web scraping (alias l'extraction de donnée web) est une technique logiciel automatisée d'extraction d'information issue d'internet.



3.4 Construction de la base de données (scraping et crawling)

La récupération de données sur le Web est un aspect important de la construction des modèles d'apprentissage automatique. Comme nous le savons, les données sont en quelque sorte l'épine dorsale de toute tâche d'apprentissage automatique,Dans ce cas d'utilisation, nous allons récupérer les détails d'un site Web financier

link : www.investing.com.



Name	Last	High	Low	Chg.	Chg. %	Vol.	Time
Apple	159.22	163.84	158.28	-0.47	-0.29%	107.77M	15:59:59
Alliant Energy	58.91	59.37	58.10	+0.81	+1.39%	1.16M	
Alphabet A	2,580.10	2,653.04	2,578.65	-4.56	-0.18%	1.55M	15:59:59
Meta Platforms	294.64	301.71	294.26	+0.01	+0.00%	21.28M	15:59:59
Tesla	829.10	935.39	829.00	-108.31	-11.55%	47.61M	15:59:59
Allegion PLC	120.60	123.36	119.34	+0.21	+0.17%	602.36K	16:00:00
C3 Ai	22.54	24.25	22.41	-0.92	-3.92%	3.91M	15:59:59
Uipath	32.85	35.05	32.83	-1.25	-3.67%	3.88M	
Kinross Gold	5.310	5.500	5.310	-0.150	-2.75%	17.98M	15:59:59
Bit Digital	3.525	3.980	3.510	-0.375	-9.62%	5.21M	
Microsoft	299.84	307.30	297.93	+3.13	+1.05%	52.73M	15:59:59
McDonald's	248.74	252.46	245.25	-1.11	-0.44%	5.09M	15:59:59
First Majestic Sil...	9.55	10.08	9.54	-0.66	-6.46%	7.70M	16:00:00
Barrick Gold	18.71	19.15	18.66	-0.32	-1.68%	20.25M	15:59:59
Agnico Eagle Mi...	46.72	48.70	46.67	-1.82	-3.75%	3.17M	15:59:59
Suncor Energy	28.45	28.95	28.05	+0.56	+2.01%	9.09M	
Nevsun	4.440	4.470	4.440	0.000	0.00%	0	
Great Panther M...	0.2002	0.2100	0.2000	-0.0098	-4.67%	1.28M	16:00:00

3.4.1 Scrape the Trending stocks

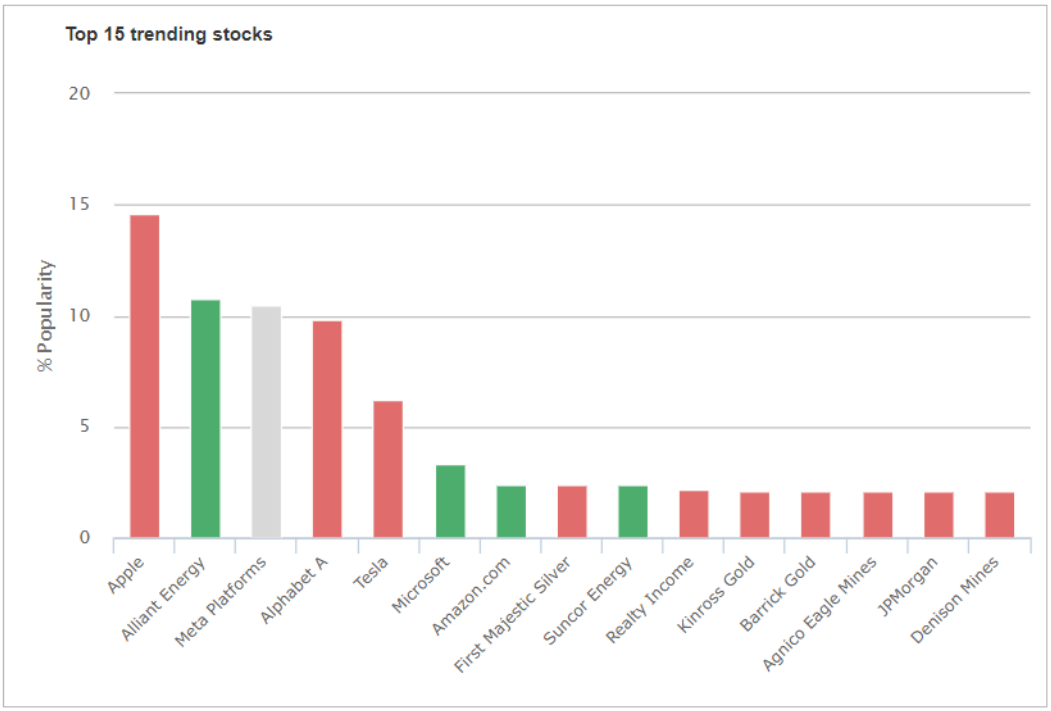
Dans le site Web, les actions en vogue sont présentes en haut de la page et il y a 6 blocs différents qui montrent quelles actions de la société sont en vogue actuellement. Après avoir obtenu les résultats du bloc ci-dessus, nous analysons d'abord les détails HTML, puis en accédant aux détails par élément, nous obtenons les détails. Nous pouvons trouver les détails de la classe pour accéder à l'élément dans le fichier html rendu.

Résultats

	Organization	Share_Value	Increase_Decrease	Pchange
0	Apple	159.22	-0.47	-0.29%
1	Alliant Energy	58.91	+0.81	+1.39%
2	Meta Platforms	294.64	+0.01	+0.00%
3	Alphabet A	2,580.10	-4.56	-0.18%
4	Tesla	829.10	-108.31	-11.55%
5	Microsoft	299.84	+3.13	+1.05%

3.4.2 Scrape the trending Stocks by Popularity

Les détails de la popularité des actions sont présentés dans un graphique à barres sur le site Web. Après une observation attentive, vous pouvez voir que les données du graphique ont été stockées dans une variable stockPopularityData. Nous avons accédé à la variable et obtenu tous les détails de celle-ci.

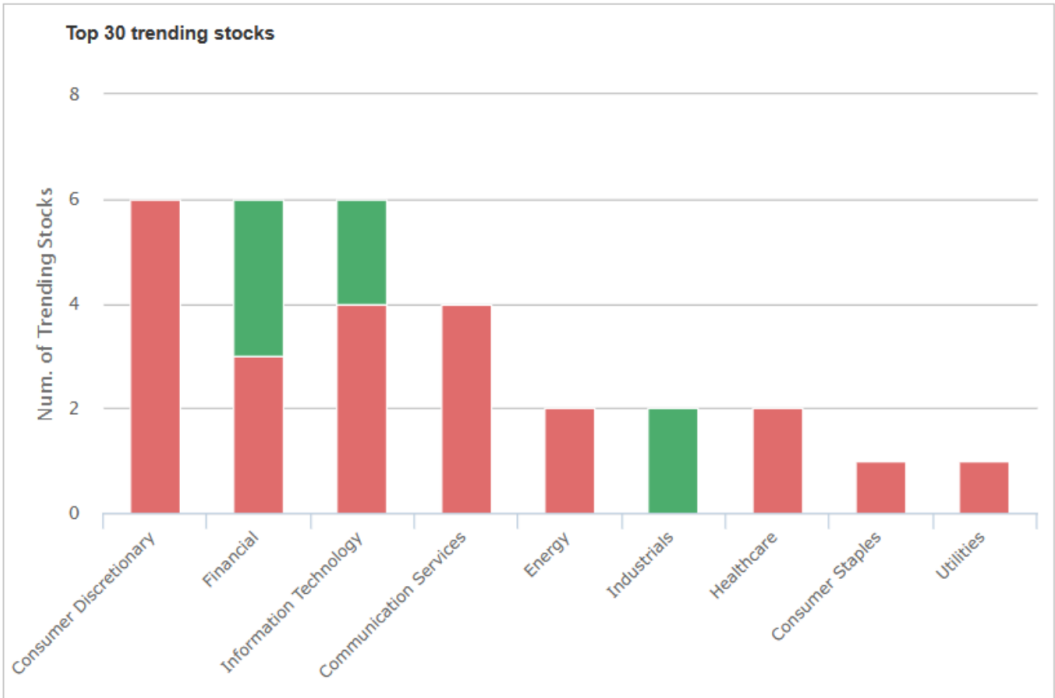


Résultats

	percentage	pair_name	pair_change	index	positive
0	15.742025	Apple	0.51	1	green
1	11.858530	Alphabet A	0.64	2	green
2	11.650485	Meta Platforms	1.66	3	green
3	3.814147	Tesla	1.75	4	green
4	2.843273	Amazon.com	0.57	5	green
5	2.773925	Pinterest	-0.42	6	red
6	2.635229	Molycorp	0.00	7	red

3.4.3 Scrape the trending Stocks by Sector

Comme pour notre deuxième tâche, les détails des actions de tendance sectorielle sont également présents dans une variable sectorPopularityData. De même, nous avons accès aux détails de la variable par laquelle nous avons obtenu nos données.



	sectorName	SectorPairs	Pair_change
0	Financial	267	-10.3400
1	Financial	15358	-0.1000
2	Communication Services	6369	17.8700
3	Communication Services	6378	2.0200
4	Communication Services	26490	5.4200
5	Communication Services	1127189	-0.1400
6	Information Technology	6408	0.8800
7	Information Technology	6497	3.6700

Ext ...

3.5 Dataset après la concaténation

Après les étapes de scraping et crawling et après la concaténation de tous les fichies, on trouve le dataset suivant avec 4112 lignes et 12 colonnes :

Date	Price	Open	High	Low	Vol.	Change %	coef	state	activity_in	activity_out	company_name
Dec 27, 2017	42.65	42.52	42.7	42.43	85.99M	0.02%	607.188563	2.0	459.012446	723.116596	APPLE
Dec 26, 2017	42.64	42.7	42.87	42.42	132.74M	-2.54%	912.234279	7.0	512.988709	419.767008	APPLE
Dec 22, 2017	43.75	43.67	43.86	43.62	65.40M	0.00%	525.357693	1.0	892.087271	274.688995	APPLE
Dec 21, 2017	43.75	43.54	44.01	43.52	83.80M	0.37%	134.447494	7.0	714.290558	214.932556	APPLE
Dec 20, 2017	43.59	43.72	43.85	43.31	93.90M	-0.09%	588.396452	NaN	517.921111	522.840917	APPLE
Dec 19, 2017	43.63	43.76	43.85	NaN	109.75M	-1.07%	758.425635	0.0	341.587978	29.517790	APPLE
Dec 18, 2017	44.1	43.72	44.3	43.72	117.68M	1.40%	494.513484	3.0	195.319583	69.321476	APPLE
Dec 15, 2017	43.49	43.41	43.54	43.12	160.68M	1.02%	134.476915	8.0	653.176660	83.086196	APPLE
Jan 14, 2022	2,789.61	2,739.97	2,814.73	2,739.97	1.45M	NaN	440.965533	3.0	604.720961	750.648842	GOOGLE
Jan 13, 2022	2,771.74	2,830.80	2,857.00	2,768.18	1.49M	-2.01%	964.955645	7.0	737.024029	114.104153	GOOGLE
Jan 12, 2022	2,828.61	2,823.00	2,852.16	2,813.89	1.28M	1.21%	716.311835	3.0	383.405551	290.800200	GOOGLE
Jan 11, 2022	2,794.72	2,760.14	2,804.32	2,733.84	1.21M	0.77%	507.215965	2.0	140.756690	893.501867	GOOGLE
Jan 10, 2022	2,773.39	2,701.56	2,776.39	2,663.29	2.20M	1.21%	5.046724	5.0	764.075383	57.522567	GOOGLE
Jan 07, 2022	2,740.34	2,762.91	2,768.97	2,715.33	1.49M	-0.53%	712.018573	1.0	564.190168	76.854707	GOOGLE

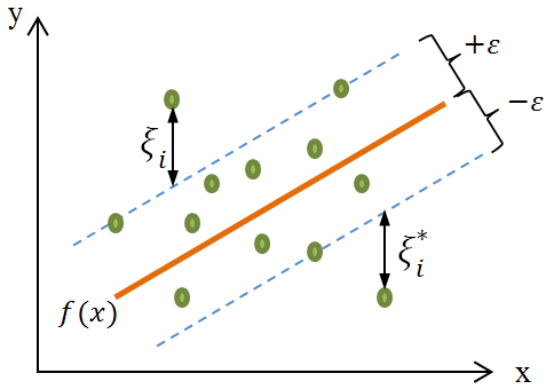
4 Chapitre 4 :Présentation des algorithmes utilisés

4.1 Support Vector Regression

Support Vector Regression (SVR) utilise le même principe que SVM, mais pour les problèmes de régression, le problème de la régression est de trouver une fonction qui rapproche le mappage d'un domaine d'entrée à des nombres réels sur la base d'un échantillon d'apprentissage. Dans ce chapitre, nous verrons comment utiliser les SVMs pour résoudre des problèmes de ré gression avec la formulation primale et duale des quelques cas SVR classiques.

4.1.1 SVR à Marge dure

Considérez ces deux lignes rouges comme limite de décision et la ligne verte comme hyperplan. Notre objectif, lorsque nous allons de l'avant avec la SVR, est de considérer essentiellement les points qui se trouvent à l'intérieur de la ligne de délimitation de décision. Notre meilleure ligne d'ajustementest l'hyperplan qui a un nombre maximum de points.



Si les points d'échantillonnage sont supposés dans une relation linéaire, alors la fonction de régression peut être écrite comme :

$$f(x) = wx + b \tag{1}$$

Où w est le vecteur orthogonal à l'hyperplan et b est le déplacement par rapport à l'origine.

4.1.2 Formulation primal

Le problème de résolution de la fonction de régression peut être transformé en problème d'optimisation suivant :

$$J(\omega) = \frac{1}{2}\omega^T\omega$$
$$\text{sujet } a : \begin{cases} y_k - \omega^T x_k - b \leq \varepsilon k = 1, \cdots, N \\ -y_k + \omega^T x_k + b \leq \varepsilon k = 1, \cdots, N \end{cases}$$

Où x_k et y_k indiquent respectivement les k ème données d'entrée et de sortie ; ε représente la précision fixe de l'approximation de la fonction. Pour obtenir la solution de ce problème, le Lagrangien est introduit

$$\begin{aligned} L(\omega, b, \alpha, \alpha^*) = & \frac{1}{2}\omega^T\omega \\ & - \sum_{k=1}^{N} \alpha_k \left(\varepsilon - y_k + \omega^T x_k + b\right) \\ & - \sum_{k=1}^{N} \alpha_k^* \left(\varepsilon + y_k - \omega^T x_k - b\right) \end{aligned}$$

4.1.3 Formulation dual

En annulant les dérivées partielles du lagrangien, selon les conditions de K−K−T, on obtient le point selle de la fonction lagrangienne :

$$\begin{cases} \frac{\partial L}{\partial \omega} = 0 \Rightarrow \omega^* = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) x_i \\ \frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0 \end{cases}$$

En remplaçant ω^* par son expression dans le lagrangien, on obtiendra le problème de programmation quadratique suivant :

$$\widehat{L}(\alpha, \alpha^*) = -\frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)\left(\alpha_j - \alpha_j^*\right) x_i^T \cdot x_j + \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) y_i$$

4.1.4 SVR à Marge souple

Pour une marge souple, nous introduisons deux variables ($\xi_i, \quad \xi_i^* \geq 0, \quad i = 1, \cdots, N.$)

4.1.5 Formulation primal

Donc la formulation primale devient :

$$\text{sujet } a : \begin{cases} J(\omega) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \\ y_k - \omega^T x_k - b \leq \varepsilon + \xi_k \quad k = 1, \cdots, N \\ -y_k + \omega^T x_k + b \leq \varepsilon + \xi_k^* \quad k = 1, \cdots, N \\ \xi_k, \xi_k^* \geq 0 \quad k = 1, \cdots, N \end{cases}$$

Pour obtenir la solution de ce problème, le Lagrangien est introduit

$$\begin{aligned} L(\omega, \alpha, \alpha^*, \beta, \beta^*, \xi, \xi^*) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \\ - \sum_{k-1}^{N}\alpha_k\left(\varepsilon + \xi_k - y_k + \omega^T x_k + b\right) - \sum_{k-1}^{N}\beta_k\xi_k \\ - \sum_{k=1}^{N}\alpha_k^*\left(\varepsilon + \xi_k^* + y_k - \omega^T x_k - b\right) - \sum_{k=1}^{N}\beta_k^*\xi_k^* \end{aligned}$$

4.1.6 Formulation dual

Fn annulant les dérivées partielles du lagrangien, selon les conditions de $K-K-T$, on obtient le point selle de la fonction lagrangienne :

$$\begin{cases} \frac{\partial L}{\partial \omega} = 0 \Rightarrow \omega^* = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\, x_i \\ \frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \\ \frac{\partial L}{\xi_i} = 0 \Rightarrow 0 \leq \alpha_i \leq C, \quad i = 1, \cdots, N \\ \frac{\partial L}{\xi_i^*} = 0 \Rightarrow 0 \leq \alpha_i^* \leq C, \quad i = 1, \cdots, N \end{cases}$$

En remplaçant ω^* par son expression dans le lagrangien, on obtiendra le problème de programmation quadratique suivant :

$$\widehat{L}(\alpha, \alpha^*) = -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \alpha_i^*)\left(\alpha_j - \alpha_j^*\right) x_i \cdot x_j + \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\, y_i$$

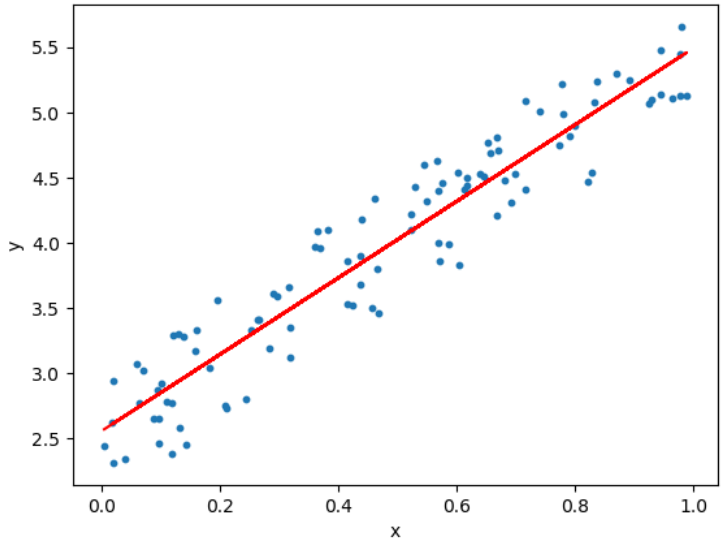
Avec

$$\sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \quad \text{et} \quad |\alpha_i - \alpha_i^*| \leq C, i = 1, \cdots, N$$

4.2 Linear regression

- On dispose d'un ensemble d'apprentissage contenant **m** observations pour lesquelles les valeurs des variables **x** et **y** sont déjà connues [m tuples $\left(x^{(i)}, y^{(i)}\right) i = | \ldots m]$
- On cherche à prédire la valeur d'une variable continue **y** en fonction d'une variable continue **x**

$$y = f(x) = ax + b$$



4.2.1 Fonction hypothèse

- Définir la fonction hypothèse **h**

$$h_\theta(x) = \theta_0 + \theta_1 x$$

où θ_0 et θ_1 sont les paramètres du modèle - Objectif : $\mathbf{h}_\theta(\mathbf{x})$ doit être une bonne approximation de la valeur réelle ***y***
-Minimiser J $\frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2 \Rightarrow$ fonction coût

4.2.2 Algorithme

Initialiser avec θ_j au hasard $(j = 0, 1)$

Répéter

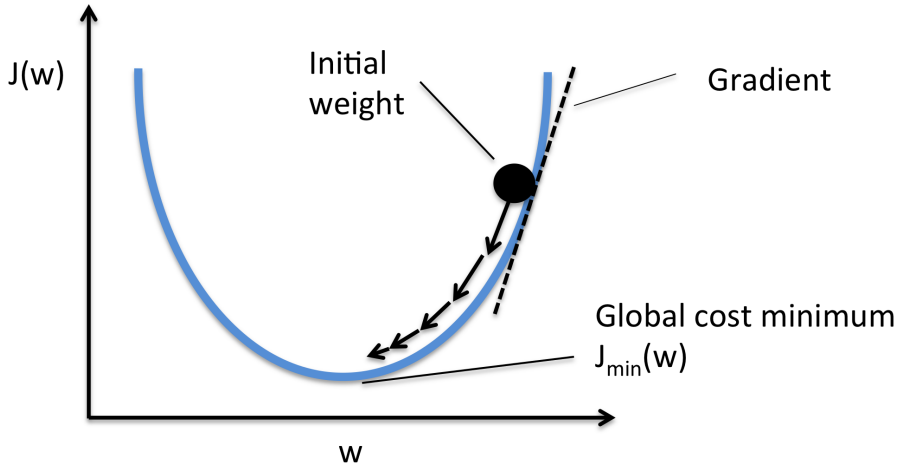
$$\theta_0 \leftarrow \theta_0 - \alpha \frac{\partial J(\theta_0, \theta_1)}{\partial \theta_0}$$

$$\theta_1 \leftarrow \theta_1 - \alpha \frac{\partial J(\theta_0, \theta_1)}{\partial \theta_1}$$

Jusqu'à convergence

$$\frac{\partial J(\theta_0, \theta_1)}{\partial \theta_0} = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right)$$

$$\frac{\partial J(\theta_0, \theta_1)}{\partial \theta_1} = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right) x^{(i)}$$



4.3 Algoritmes de réduction de dimentionalité

4.3.1 Sélection de caractéristiques

4.3.1.1 Approches filtres

Les méthodes de filtrage sont généralement utilisées comme étape de prétraitement. La sélection des fonctionnalités est indépendante de tout algorithme d'apprentissage automatique. Au lieu de cela, les caractéristiques sont sélectionnées sur la base de leurs scores dans divers tests statistiques pour leur corrélation avec la variable de résultat. Les caractéristiques de ces méthodes sont les suivantes :

- Ces méthodes reposent sur les caractéristiques des données (feature features)
- Ils n'utilisent pas d'algorithmes d'apprentissage automatique.
- Ce sont des modèles agnostiques.
- Ils ont tendance à être moins coûteux en calcul.
- Ils donnent généralement des performances de prédiction inférieures à celles des méthodes wrapper.
- Ils sont très bien adaptés pour un dépistage rapide et la suppression des fonctionnalités non pertinentes.

4.3.1.1.1 Critère de Fisher

La méthode ANOVA F-value estime le degré de linéarité entre la caractéristique d'entrée (c'est-à-dire le prédicteur) et la caractéristique de sortie. Une valeur F élevée indique un haut degré de linéarité et une faible valeur F indique un faible degré de linéarité. Le principal inconvénient de l'utilisation de la valeur F ANOVA est qu'elle ne capture que les relations linéaires entre les caractéristiques d'entrée et de sortie. En d'autres termes, aucune relation non linéaire ne peut être détectée par la valeur F.

4.3.1.1.2 Critère de Variance

La méthode du seuil de variance supprime les entités dont la variance est inférieure à une valeur limite prédéfinie. Il est basé sur la notion que les caractéristiques qui ne varient pas beaucoup en elles-mêmes ont un faible pouvoir prédictif. La principale faiblesse du seuil de variance est qu'il ne tient pas compte de la relation entre les caractéristiques d'entrée et la caractéristique de sortie.

Il convient de noter qu'avant d'effectuer le seuillage de la variance, toutes les caractéristiques doivent être standardisées afin qu'elles aient la même échelle.

4.3.1.1.3 Critère d'information mutuelle

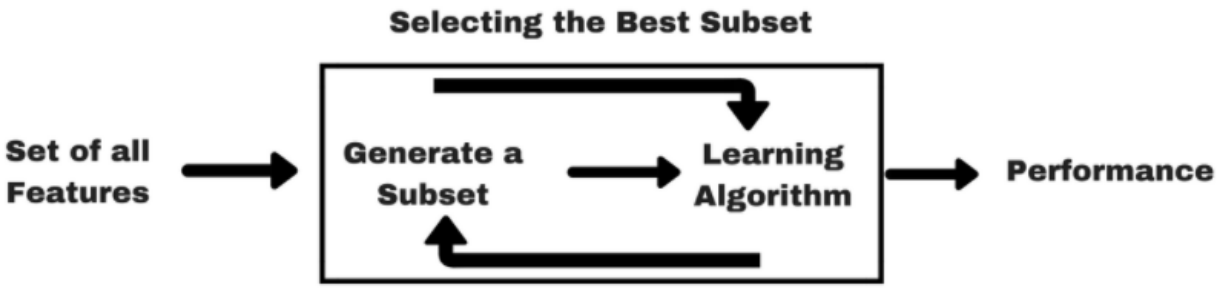
L'information mutuelle (IM) mesure la dépendance d'une variable à une autre en quantifiant la quantité d'informations obtenues sur une caractéristique, à travers l'autre caractéristique. MI est symétrique et non négatif, et vaut zéro si et seulement si les caractéristiques d'entrée et de sortie sont indépendantes. Contrairement à la valeur F ANOVA, les informations mutuelles peuvent capturer des relations non linéaires entre les caractéristiques d'entrée et de sortie.

4.3.1.2 Approches enveloppantes

Dans les méthodes wrapper, nous essayons d'utiliser un sous-ensemble de fonctionnalités et de former un modèle en les utilisant. Sur la base des inférences que nous tirons du modèle précédent, nous décidons d'ajouter ou de supprimer des fonctionnalités du sous-ensemble.
Le problème est essentiellement réduit à un problème de recherche. Ces méthodes sont généralement très coûteuses en temps de calcul.

Quelques exemples courants de méthodes wrapper sont

— Sélection vers l'avant,
— Élimination à rebours,
— Sélection exhaustive des fonctionnalités,
— Élimination des fonctionnalités récursives.
— Élimination récursive des fonctionnalités avec validation croisée



4.3.1.2.1 Sequential forward selection (SFS)

SFS trouve le meilleur sous-ensemble de fonctionnalités en ajoutant une fonctionnalité qui améliore le mieux le modèle à chaque itération.

Algorithme :

Sequential Forward Selection (SFS)

Input : $Y = \{y_1, y_2, \ldots, y_d\}$

- L'algorithme SFS prend les d-caractéristique définie comme input.

Output : $X_k = \{x_j \mid j = 1, 2, \ldots, k; x_j \in Y\}$, ou $k = (0, 1, 2, \ldots, d)$

- SFS retourne un subset de features ; le nombre de features selectionné k, ou $k < d$, doit etre spécifié a priori.

Initialisation : $X_0 = \emptyset, k = 0$

- On initialise l'algorithme avec un set vide $\emptyset$ ("null set") donc $k = 0$ (ou k est le nombre de subset).

Step 1 (Inclusion) :

$x^+ = \arg\max J(X_k + x)$, where $x \in Y - X_k$

$X_{k+1} = X_k + x^+$

$k = k + 1$

Go to Step 1

- Dans cette etape on ajoute un feature, x^+, a notre subset X_k.

- x^+est le feature qui maximise notre fonction de critère, c'est-à-dire le feature qui est associée a la meilleure performance du model s'il est ajouté à X_k.

- On repete jusqu'a terminaison.

terminaison : $k = p$

Les p feature desiré specifié a priori.

4.3.1.2.2 Sequential backward selection (SBS)

SBS est l'opposé de SFS. SBS commence par toutes les fonctionnalités et supprime la fonctionnalité qui a le moins d'importance pour le modèle à chaque itération.

Algorithme :

Sequential BackWard Selection (SBS)

Input : $Y = \{y_1, y_2, \ldots, y_d\}$

- L'algorithme SFS prend les d-caractéristique définie comme input.

Output : $X_k = \{x_j \mid j = 1, 2, \ldots, k; x_j \in Y\}$, ou $k = (0, 1, 2, \ldots, d)$

- SBS retourne un subset de features ; le nombre de features selectionné k, ou $k < d$, doit etre spécifié a priori.

Initialisation : $X_0 = Y, k = d$

- On initialise l'algorithme avec tout l'ensemble de features c'est a dire $k = d$.

Step 1 (Exclusion) :

$x^- = \arg\max J(X_k - x)$, where $x \in X_k$

$X_{k-1} = X_k - x^-$

$k = k - 1$

Go to Step 1

- Dans cette etape on elimine un feature, x^-du subset X_k.

- x^-est le feature qui maximise notre fonction de critère , c'est-à-dire la le feature qui est associée avec les meilleures performances du model s'il est supprimé de X_k.

- Nous répétons cette procédure jusqu'à ce que le critère de terminaison soit satisfait.

Termination : $k = p$

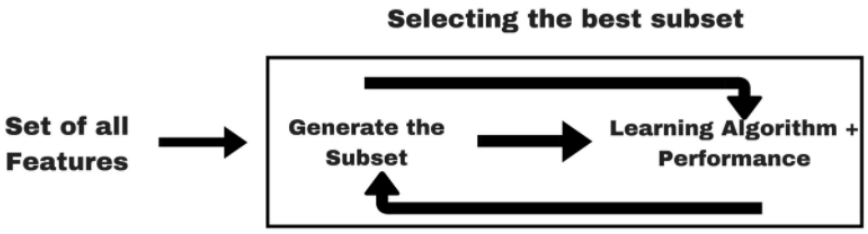
- On elimine des features du subset X_k jusqu'a ce que le subset est de taille k

(2)

4.3.1.3 Approches integres

Les méthodes intégrées combinent les points forts des méthodes de filtrage et d'encapsulation en tirant parti des algorithmes de la machine qui ont leur propre processus de sélection de fonctionnalités intégré. Ils intègrent une étape de sélection des fonctionnalités dans le cadre du processus de formation (c'est-à-dire que la sélection des fonctionnalités et le processus de formation sont effectués simultanément). Les méthodes intégrées ont généralement un processus plus efficace que les méthodes wrapper car elles éliminent le besoin de recycler chaque sous-ensemble de fonctionnalités examinées.

Les méthodes intégrées peuvent être expliquées à l'aide du graphique suivant :



4.3.1.3.1 Random forest

Les forêts aléatoires sont l'un des algorithmes d'apprentissage automatique les plus populaires. Ils ont autant de succès car ils offrent en général de bonnes performances prédictives, un faible surajustement et une interprétabilité facile. Cette interprétabilité est donnée par le fait qu'il est simple de déduire l'importance de chaque variable sur la décision de l'arbre. En d'autres termes, il est facile de calculer dans quelle mesure chaque variable contribue à la décision.

La sélection de fonctionnalités à l'aide de la forêt aléatoire relève de la catégorie des méthodes intégrées. Les méthodes embarquées combinent les qualités des méthodes de filtrage et d'encapsulation. Ils sont implémentés par des algorithmes qui ont leurs propres méthodes de sélection de fonctionnalités intégrées. **Certains des**

avantages des méthodes embarquées sont :

— Ils sont très précis.
— Ils généralisent mieux.
— Ils sont interprétables.

Fonctionement

Les forêts aléatoires se composent de 4 à 12 cents arbres de décision, chacun d'eux construit sur une extraction aléatoire des observations de l'ensemble de données et une extraction aléatoire des caractéristiques.
Tous les arbres ne voient pas toutes les caractéristiques ou toutes les observations, ce qui garantit que les arbres sont décorrélés et donc moins sujets au sur-ajustement. Chaque arbre est également une séquence de questions oui-non basées sur une seule ou une combinaison de fonctionnalités.
À chaque nœud (c'est-à-dire à chaque question), les trois divisent l'ensemble de données en 2 compartiments, chacun d'eux hébergeant des observations plus similaires entre elles et différentes de celles de l'autre compartiment. Par conséquent, l'importance de chaque caractéristique est dérivée de la "pureté" de chacun des seaux.

4.3.2 Transformation de données

4.3.2.1 ACP

L'analyse en composantes principales (ACP ou PCA en anglais pour principal component analysis), ou, selon le domaine d'application, transformation de Karhunen–Loève (KLT) ou transformation de Hotelling, est une méthode de la famille de l'analyse des données et plus généralement de la statistique multivariée, qui consiste à transformer des variables liées entre elles (dites « corrélées » en statistique) en nouvelles variables décorrélées les unes des autres. Ces nouvelles variables sont nommées « composantes principales » ou axes principaux. Elle permet au statisticien de résumer l'information en réduisant le nombre de variables.
On applique généralement une ACP sur un ensemble de N variables aléatoires X1, ..., XN

Cet échantillon de ces N variables aléatoires peut être structuré dans une matrice M, à K lignes et N colonnes.
Les principales etapes de l'ACP sont :
1-La normalistaion des données :
Pour resoudre le probleme des unités.
2-Le calcul de la matrice de corelation
3-Calculer les valeurs et vecteurs propres de cette matrice.
4-Choisir les composantes souhaités suivant les resulatas obtenu sur les valeurs propres.

5 Chapitre 5 :Prétraitement de dataset

5.1 Compréhension de l'ensemble des donnéess

Visualisation des 4 premières lignes :

Date	Price	Open	High	Low	Vol.	Change %	coef	state	activity_in	activity_out	company_name
Dec 27, 2017	42.65	42.52	42.7	42.43	85.99M	0.02%	607.188563	2.0	459.012446	723.116596	APPLE
Dec 26, 2017	42.64	42.7	42.87	42.42	132.74M	-2.54%	912.234279	7.0	512.988709	419.767008	APPLE
Dec 22, 2017	43.75	43.67	43.86	43.62	65.40M	0.00%	525.357693	1.0	892.087271	274.688995	APPLE
Dec 21, 2017	43.75	43.54	44.01	43.52	83.80M	0.37%	134.447494	7.0	714.290558	214.932556	APPLE
Dec 20, 2017	43.59	43.72	43.85	43.31	93.90M	-0.09%	588.396452	NaN	517.921111	522.840917	APPLE
Dec 19, 2017	43.63	43.76	43.85	NaN	109.75M	-1.07%	758.425635	0.0	341.587978	29.517790	APPLE
Dec 18, 2017	44.1	43.72	44.3	43.72	117.68M	1.40%	494.513484	3.0	195.319583	69.321476	APPLE
Dec 15, 2017	43.49	43.41	43.54	43.12	160.68M	1.02%	134.476915	8.0	653.176660	83.086196	APPLE
Jan 14, 2022	2,789.61	2,739.97	2,814.73	2,739.97	1.45M	NaN	440.965533	3.0	604.720961	750.648842	GOOGLE

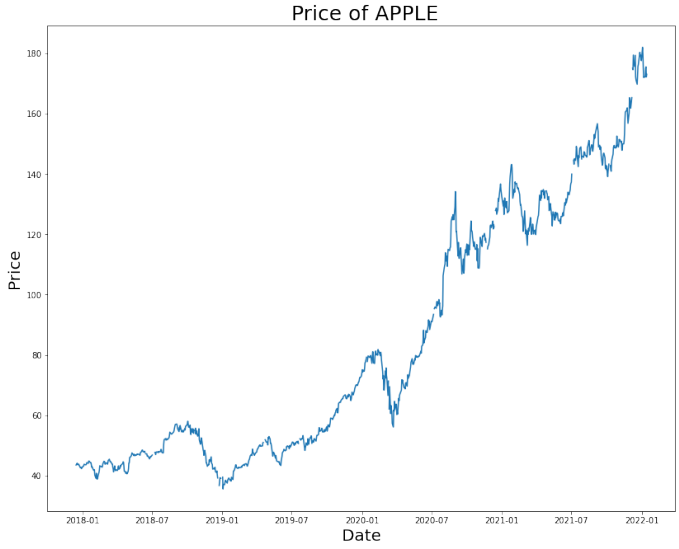
5.2 Description des donnees

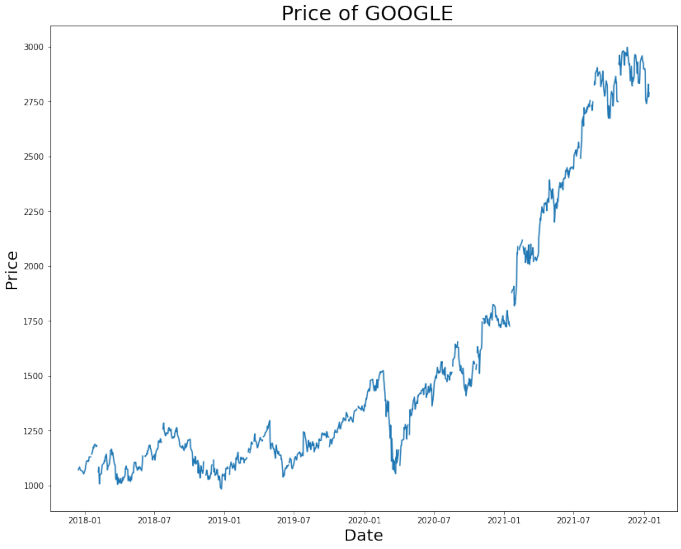
La méthode «describe» génère des statistiques descriptives (Min-Max Moyenne-Somme..) qui résument la tendance centrale, la dispersion et la forme de la distribution d'un ensemble de données, à l'exclusion des valeurs NaN

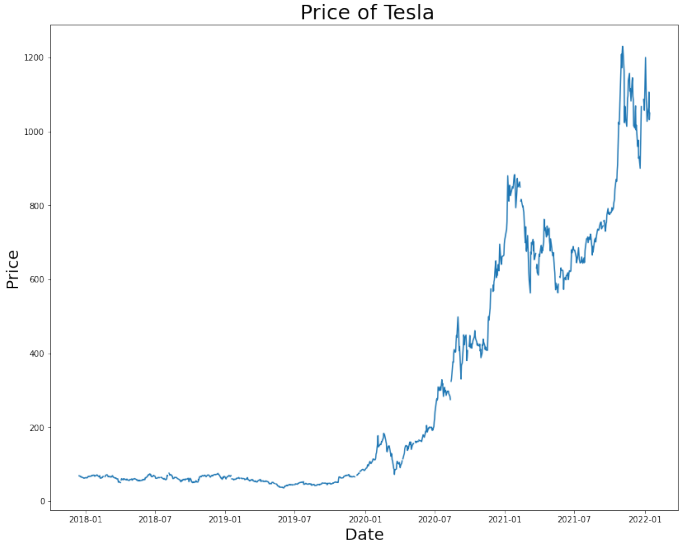
Date	Price	Open	High	Low	Vol.	Change %	coef	state	activity_in	activity_out	company_name
Dec 27, 2017	42.65	42.52	42.7	42.43	85.99M	0.02%	607.188563	2.0	459.012446	723.116596	APPLE
Dec 26, 2017	42.64	42.7	42.87	42.42	132.74M	-2.54%	912.234279	7.0	512.988709	419.767008	APPLE
Dec 22, 2017	43.75	43.67	43.86	43.62	65.40M	0.00%	525.357693	1.0	892.087271	274.688995	APPLE
Dec 21, 2017	43.75	43.54	44.01	43.52	83.80M	0.37%	134.447494	7.0	714.290558	214.932556	APPLE
Dec 20, 2017	43.59	43.72	43.85	43.31	93.90M	-0.09%	588.396452	NaN	517.921111	522.840917	APPLE
Dec 19, 2017	43.63	43.76	43.85	NaN	109.75M	-1.07%	758.425635	0.0	341.587978	29.517790	APPLE
Dec 18, 2017	44.1	43.72	44.3	43.72	117.68M	1.40%	494.513484	3.0	195.319583	69.321476	APPLE
Dec 15, 2017	43.49	43.41	43.54	43.12	160.68M	1.02%	134.476915	8.0	653.176660	83.086196	APPLE
Jan 14, 2022	2,789.61	2,739.97	2,814.73	2,739.97	1.45M	NaN	440.965533	3.0	604.720961	750.648842	GOOGLE

FIGURE 3 – Description

5.3 La variation du prix en fonction du variable Date (2018-2022)







5.4 La variation du prix avec d'autres variables

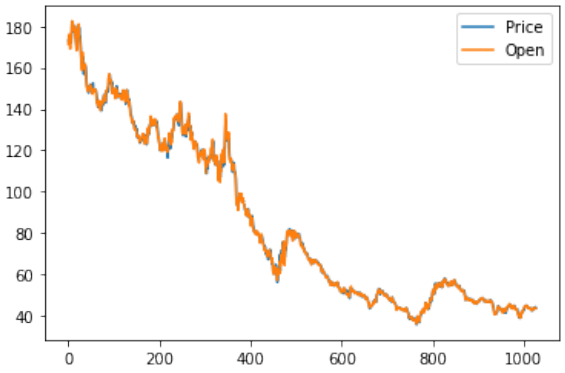


FIGURE 4 – le prix et variable open

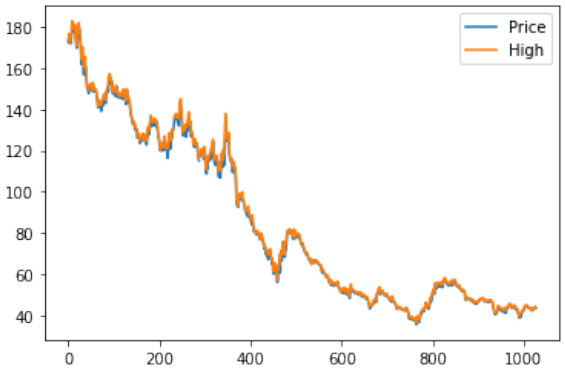


FIGURE 5 – le prix et variable High

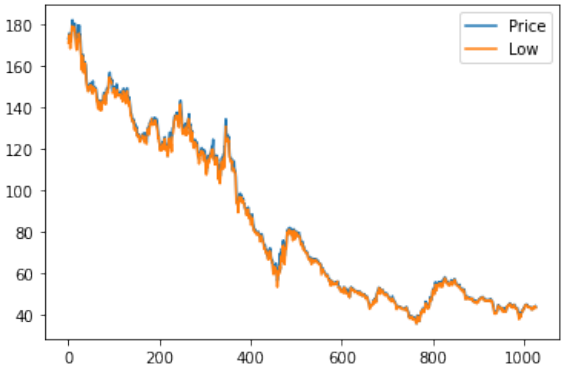


FIGURE 6 – le prix et variable Low

5.5 La variation des variables entre eux

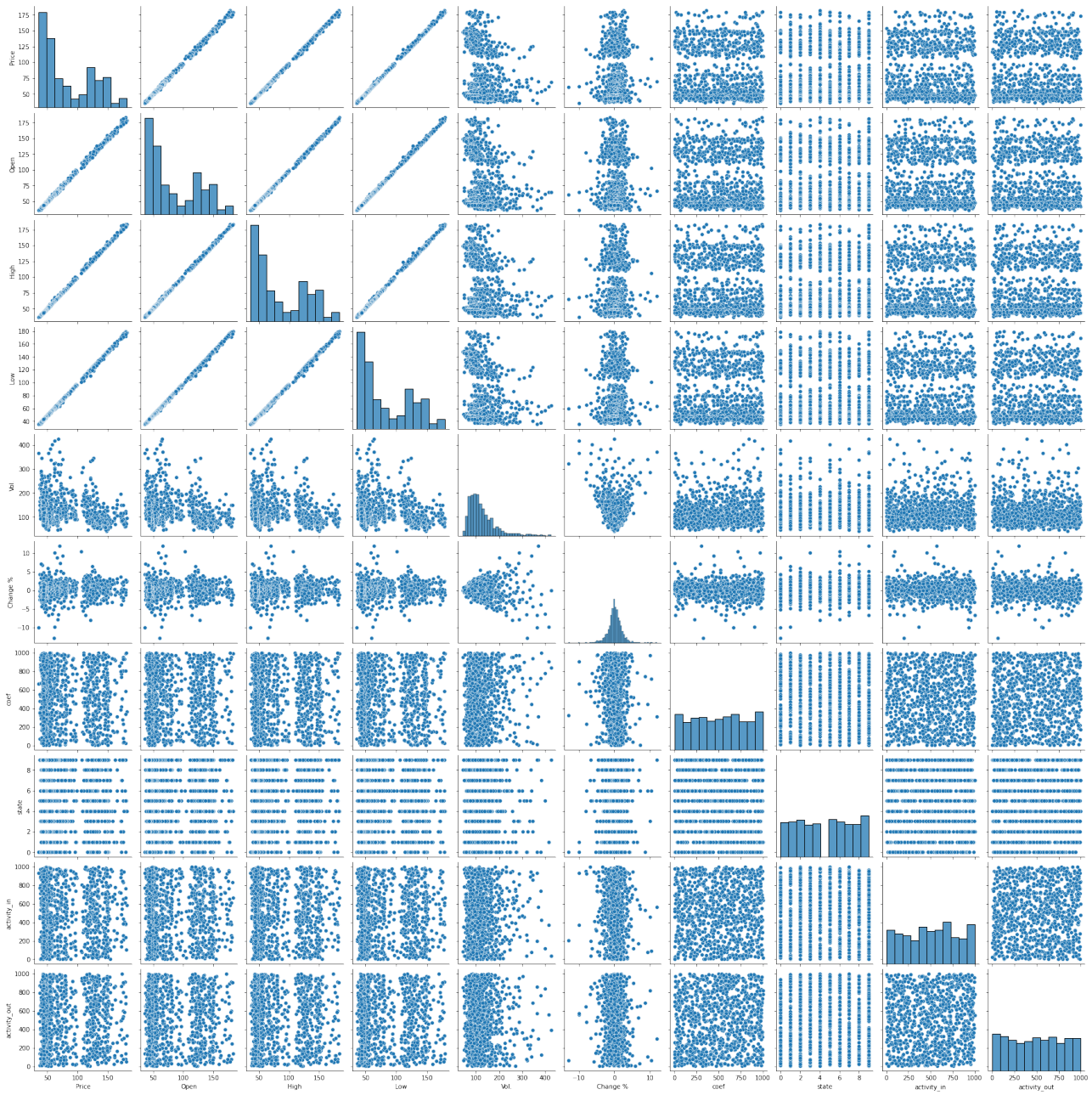


FIGURE 7 – La variation du variables

5.6 Data cleaning

Correction du problème des valeurs nulles : comme nous l'avons vu dans nos tableaux de données, nous avons plusieurs lignes avec la valeur 0,NaN dans certaines colonnes. ce que nous avons fait est de changer ces valeurs en moyenne de la colonne ,on a même tester la median et le mode mais la moyenne c'est lui qui donne la meilleur résultats

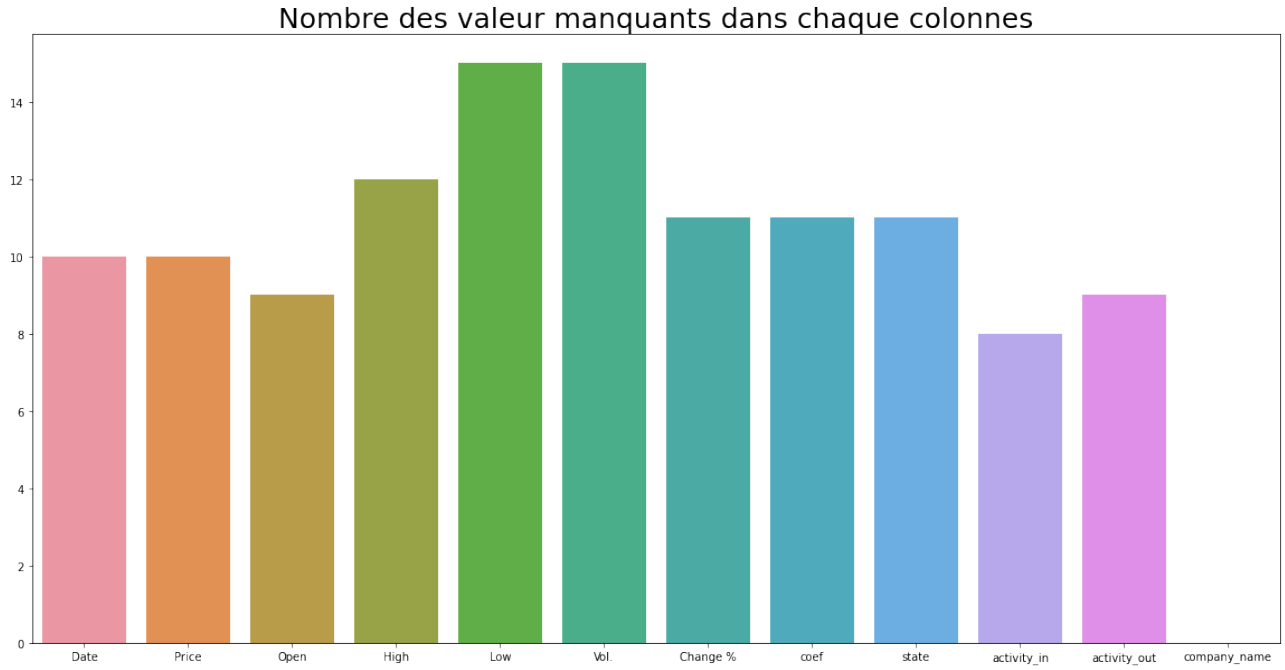


FIGURE 8 – Missing value

Remarque

==>Nous avons supprimé les lignes qui contiennent des valeurs manquantes au niveau de la variable Date car la Date est très importante pour prédicter le prix des actions comerciales

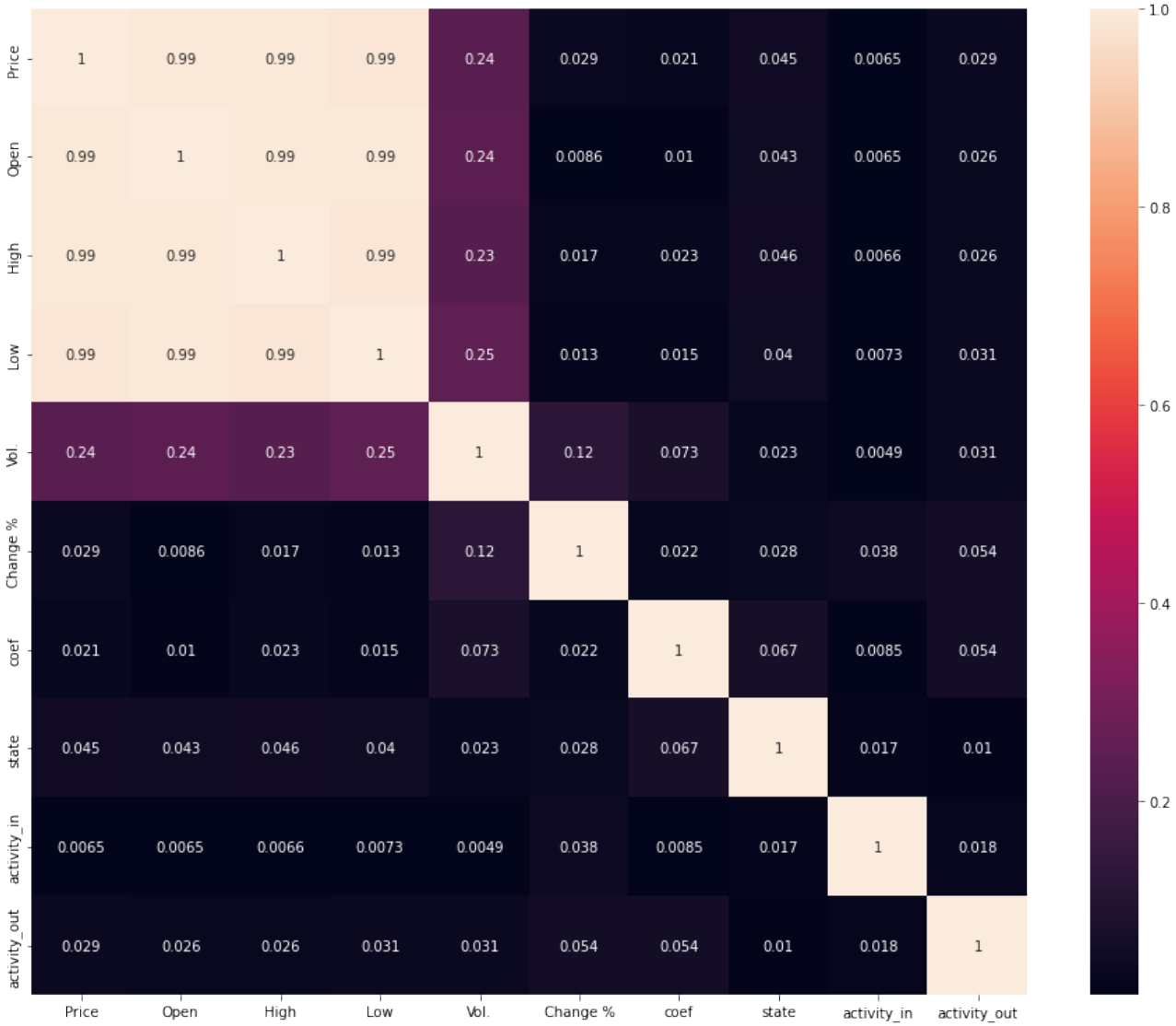
5.6.1 Structuration du dataset après scraping

A ce niveau on enlève les virgules et les symboles qui résulte de scraping et on a structuré le dataset sous forme de Dataframe

5.7 Normalisation des données

La normalisation consiste à remettre à l'échelle des attributs numériques à valeur réelle dans la plage 0 et 1 .
La plupart des algorithmes d'apprentissage automatique prévoient ou sont plus efficaces si les attributs des données ont la même échelle. Dans notre cas on a utilisé une normalisation standard grace à la fonction **StandardScaler** du package **sklearn**.

5.8 La corrrélation avant la réduction de dimentionalité



5.9 La prédiction du prix avant la réduction de dimentionnalité

mean absolute error :

$$- > \text{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n} \tag{3}$$

Mean Squared Error

$$- > \text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \tilde{y}_i)^2 \tag{4}$$

5.9.1 La prédiction avec Support Vector Machine SVR

Pour l'implementation de Support Vector Machine on a utilisé la fonction prédéfinie SVR() du package sklearn, ensuite on a entraine le modèle avec les donées d'entrainement par la fonction fit()

Mean Absolute Error Value is : 0.035860823087298106
Mean Squared Error Value is : 0.004029410534518302
Median Squared Error Value is : 0.020457652378688906

5.9.2 La prédiction avec Linear regression

Pour l'implementation de Linear regression on a utilisé la fonction prédéfinie LinearRegression() du package sklearn

Mean Absolute Error Value is : 0.0103032099001841 62
Mean Squared Error Value is : 0.0002485281157529967
Median Squared Error Value is : 0.006358866837261834

5.10 La réduction de dimentionalité et sélection des variables

5.10.1 Sélection des caractéristiques

5.10.1.1 Approches filtres

5.10.1.1.1 Critère de Fisher

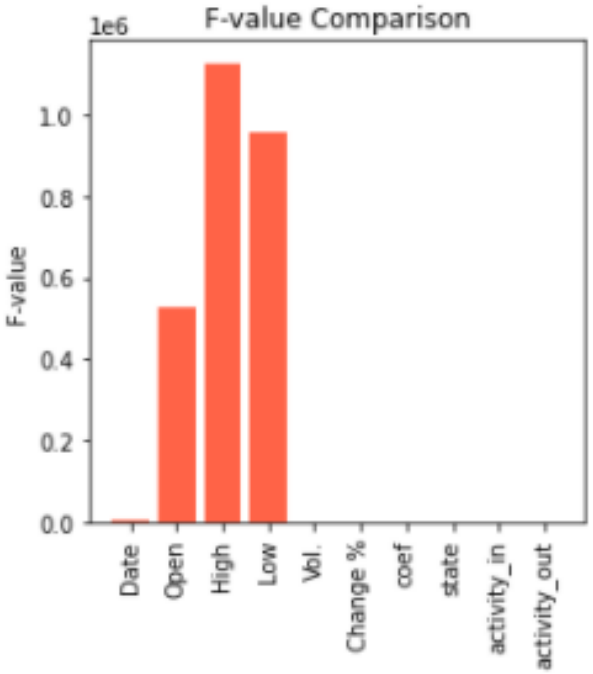


FIGURE 9 – F-test

==>On remarque que seul les variables open , High , Low sont a prendre en consideration en suivant ce critere .

5.10.1.1.2 Critère de variance

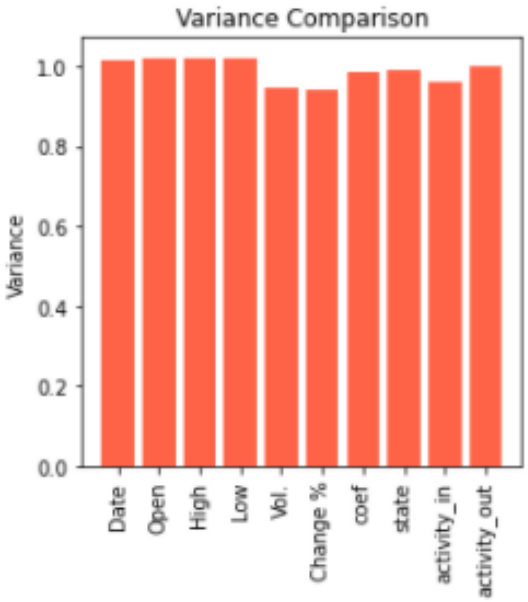


FIGURE 10 – Variance de chaque feature

Date : 1.015
Open :1.020
High :1.020
Low :1.021
Vol. :0.943
Change : 0.942
coef : 0.985
state :0.988
activity_in :0.959
activity_out : 0.999

==>**Tout les variance sont bonnes donc aucune feature ne sera eliminée par ce critere.**

5.10.1.1.3 Critère d'information mutuelle

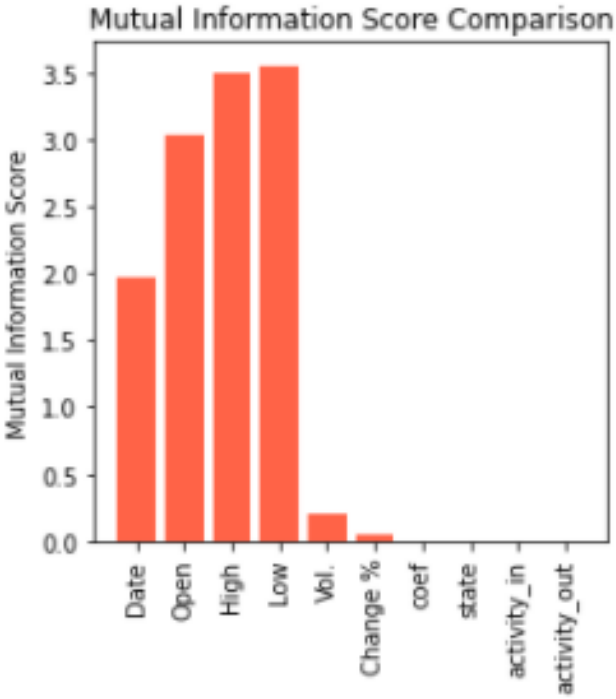


FIGURE 11 – Importance de chaque feature suivant le critere de l'Information mutuelle

On remarque que seul les variables date , open , High , Low sont a prendre en consideration en suivant ce critere .

5.10.1.2 Approches enveloppantes

5.10.1.2.1 SFS

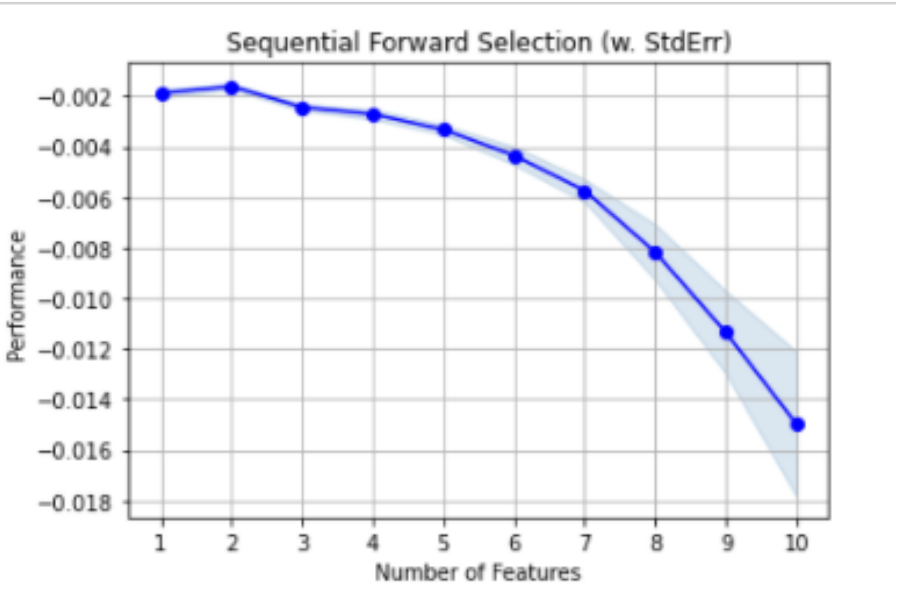


FIGURE 12 – Evolution de la performance du model selon le nombre de features(de 1 a 10)

Le SFS nous donne que le model le plus performant est celui avec les deux variable Open et LoW.

5.10.1.2.2 SBS

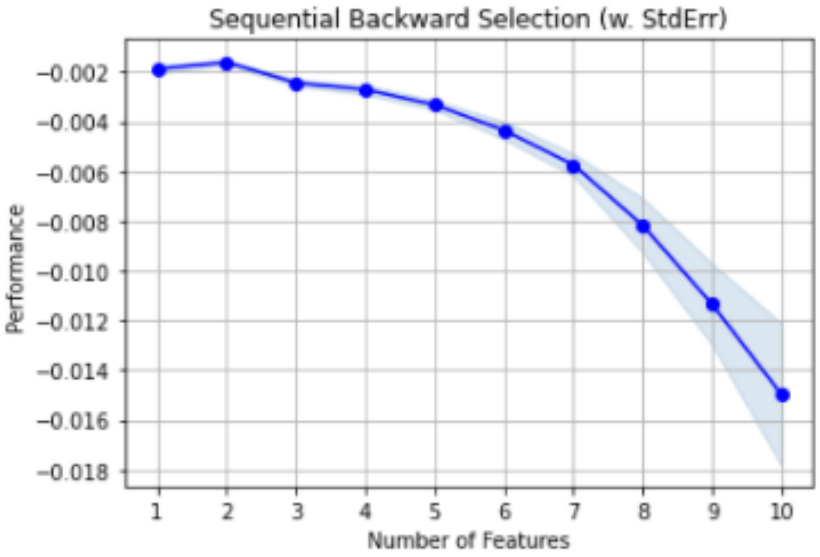


FIGURE 13 – Evolution de la performance du model selon le nombre de features(de 10 a 1)

De meme pour le SBS . Le model le plus performant est celui avec les deux variable Open et LoW.

5.10.1.3 Approches intégres

5.10.1.3.1 Random Forest

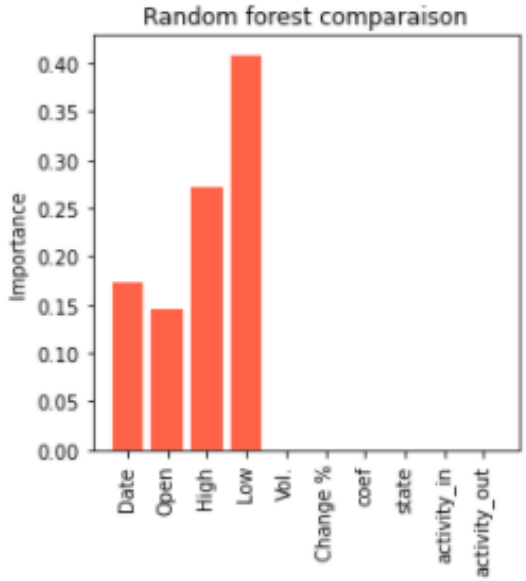


FIGURE 14 – Importance de chaque feature suivant l'approche random forest

Date :0.173
Open :0.145
High :0.271
Low :0.408
Vol. :3.673 e-05
Change : 8.044 e-05
coef :2.462 e-05
state :2.096 e-05
activity_in :3.251 e-05
activity_out :2.429 e-05
===>Les variable date , open , High , Low sortent avec la plus grosse importance .

5.10.2 Transformation de données

5.10.2.1 ACP

Apres avoir appliqué l'ACP sur notre jeu de données prétraité on a les resulats suivant :

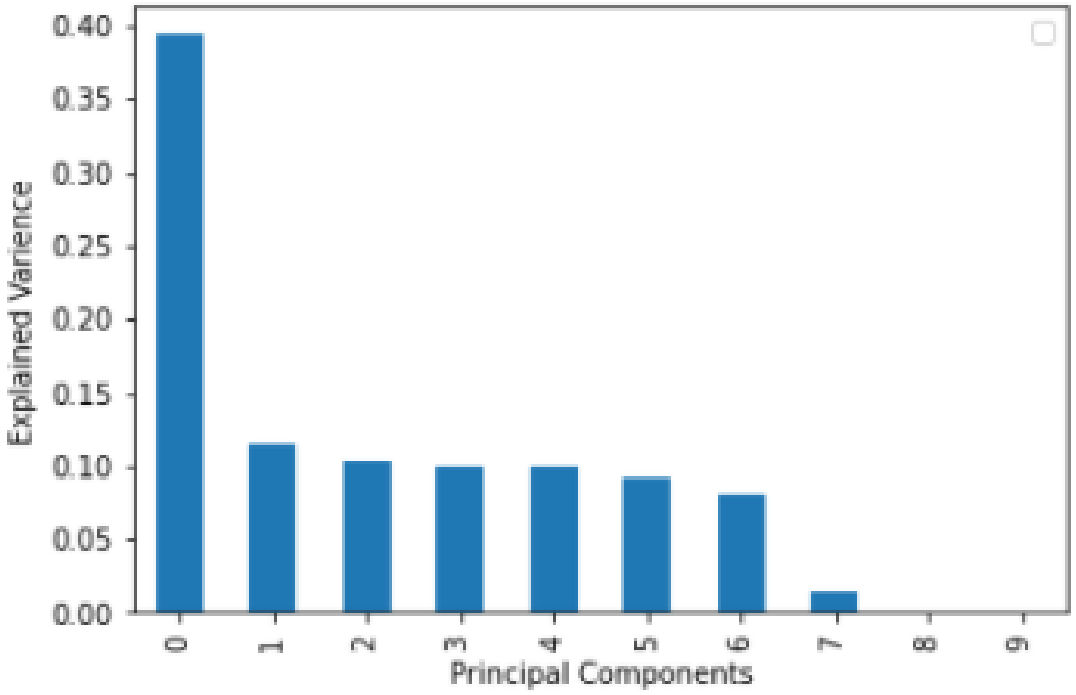


FIGURE 15 – Variance expliquée par composante

on va prendre les valeurs propres dont la variance expliquée est superieur a 0.05 donc les six premieres composantes.

5.10.3 Tableau de comparaison

		SVR	Linear regression	Methode
Sans reduction de dimensionalté	Precision : Mean SE : Median SE :	0.99577213 0.00402941 0.02045765	0.99973923 0.00024852 0.00635886	-
Critère d'information mutuelle	Precision : Mean SE : Median SE :	0.99954522 0.00043343 0.00806097	0.99959234 0.00043343 0.00806097	Approches filtres
Critère de Fisher	Precision : Mean SE : Median SE :	0.99953947 0.00043890 0.00754596	0.99959314 0.00043890 0.00754596	Approches filtres
SFS/SBS	Precision : Mean SE : Median SE :	0.99946716 0.00050782 0.00892904	0.99946724 0.00050782 0.00892904	Approches enveloppantes
Random Forest	Precision : Mean SE : Median SE :	0.99954522 0.00043343 0.00806097	0.99959234 0.00043343 0.00806097	Approches intégres
PCA	Precision : Mean SE : Median SE :	0.98567475 0.01365282 0.07597105	0.98575719 0.01365282 0.07597105	Transformation de données

FIGURE 16 – Tableau de comparaison

Conclusion

Ce projet nous a permis d'appliquer nos connaissances et nos acquis pendant le cours d'analyse avancée des données, nous avons pu appliquer et étudier les méthodes de réduction de dimensionnalité à notre tableau de données, nous avons également étudié et appliqué les différentes méthodes de régression qui permettent de prédire les cours des actions commerciales en se basant sur les valeurs historiques de plusieurs critères qui sont à l'origine des fluctuations de ces cours.