UNIVERSITÉ DE MONTPELLIER

PROJET TUTEURÉ

Prédiction de cours boursiers du CAC40

COMPTE RENDU

Présenté par :

Thibault CLÉMENT
Azhar KHAMIDULLINA

Encadrés par :

Gilles MICHEL
Stéphane MUSSARD

DU BDAP

Big Data, Data Sciences & Analyse des risques sous Python

FACULTÉ D'ÉCONOMIE
Université de Montpellier

01 Mars 2021 - 30 Juin 2021

Résumé

Ce compte rendu présente le travail effectué pour le projet tuteuré concernant la prédiction boursière d'actions du CAC40 par réseaux neuronaux. La conception passe par plusieurs étapes afin de présenter un projet en « full stack ». Le web scraping permet d'abord de récupérer les données nécessaires aux analyses, le data cleaning et features engineering permettent respectivement de nettoyer et transformer les données afin d'être traité, l'entraînement et test du modèle est effectué dans un réseau de neurone adapté pour traité les données temporelles, les résultats sont enfin présentés dans un dashboard disponible en ligne.

Mots clefs : Yahoo Finance, Keras Tensorflow, LSTM, NLP, Scikit-Learn, GridSearchCV

This report presents the work done for the tutored project concerning the stock market prediction of CAC40 shares by neural networks. The design goes through several steps in order to present a "full stack" project. The web scraping allows first to recover the data necessary for the analyses, the data cleaning and features engineering allow respectively to clean and transform the data in order to be treated, the training and test of the model is carried out in a neural network adapted to treat the temporal data, the results are finally presented in a dashboard available on line.

Keywords : Yahoo Finance, Keras Tensorflow, LSTM, NLP, Scikit-Learn, GridSearchCV

Table des matières

Introduction **4**

1 Choix des actions et Data Preparation **5**

1.1 Les actions . . . 5
1.2 Les matières premières . . . 6
1.3 La météo et cycle lunaire . . . 6
1.4 Les réseaux sociaux . . . 6
1.5 L'actualité . . . 7
1.6 Features Selection . . . 7
1.6.1 L'index . . . 7
1.6.2 Les variables y à prédire . . . 7
1.6.3 Les variables de la matrice X . . . 7
1.7 Data Cleaning . . . 8
1.7.1 Valeurs manquantes . . . 8

2 Modélisation LSTM pour la prédiction de série temporelle **9**

2.1 Features Engineering . . . 9
2.1.1 Standardisation des données . . . 9
2.1.2 Transformation en données exploitables . . . 9
2.2 Les modèles LSTM . . . 9
2.2.1 Les modèles LSTM univariés . . . 10
2.2.2 Les modèles LSTM multivariés . . . 11
2.3 Natural Language Processing (NLP) . . . 11
2.3.1 Textblob . . . 11
2.3.2 BERT . . . 12

3 Analyse des résultats **13**

3.1 Résultats et Gestion du portefeuille . . . 13
3.1.1 Fonction de perte . . . 13
3.1.2 Ratio de prédiction . . . 13
3.1.3 La stratégie boursière . . . 14
3.2 Résultats des prédictions . . . 14
3.2.1 Airbus . . . 14

3.2.2 Orange 15
3.2.3 L'Oréal 15
3.2.4 Renault 16
3.2.5 Total 17

4 Industrialisation du projet **18**
4.1 Structure du projet : *Cookiecutter* 18
4.2 Mise en production et hébergement web 18
4.2.1 JavaScript 18
4.2.2 Serveur Raspberry Pi 18
4.2.3 Le dashboard 19
4.2.4 Les évolutions de versions à venir 20

Conclusion **21**

Références **21**

Table des figures **23**

A Structure de Cookiecutter **24**

Introduction

Depuis les années 1970, le paysage financier n'a cessé de se transformer en terme de déréglementation, d'innovations, de désintermédiation [14]. Dès lors, les acteurs financiers sont confrontés à la complexité croissante de ces changements. L'étude des prédictions boursières et plus généralement de leurs tendances liées aux conjonctures qui en découlent a toujours été une problématique passionnante de ce secteur.

Dans cet optique, ce projet a pour finalité de prédire des cours boursiers du CAC40 défini à l'avance à l'aide d'un réseau de neurone et sur une période de temps passé de manière à pouvoir considérer son efficacité.

Pour y parvenir, le projet est en *full-stack*, utilisant un ensemble de technologies et méthodes permettant de présenter le projet de sa conception jusqu'à sa mise en production.

- La première partie présente le choix des actions à étudier, ainsi que leur extraction par *web scraping*. On présentera également l'étape de nettoyage des données nécessaire au bon traitement des données pour la modélisation.
- La seconde partie expose les différentes études et réflexions concernant le type de réseau de neurone, ainsi que les sous-types qui s'y réfèrent. On présentera également le feature engineering nécessaire au bon traitement des données par le réseau.
- La troisième partie présente l'analyse des résultats avec les stratégies de portefeuille sélectionné.
- Enfin, une dernière partie présentera les outils utilisés pour l'industrialisation et la mise en production des résultats.

Chapitre 1

Choix des actions et Data Preparation

Le CAC40 est le principal indice boursier de la Bourse de Paris. On y retrouve les 40 actions cotées en continu sur le premier marché parmi les 100 sociétés dont les échanges sont les plus abondants. Ces sociétés sont représentatives en principe des différentes branches d'activités de l'économie.

Dans un premier temps, on fixe une limite de 5 actions différentes à étudier, représentant chacune un secteur différent. On étudie ensuite différentes variables qui pourraient potentiellement jouer un rôle dans l'évolution des tendances des actions. On cite par exemple les cours des matières premières, les réseaux sociaux ou encore la météo.

1.1 Les actions

Voici la liste des actions conservées :

- **Airbus** est une société du secteur de l'aéronautique, spatiale et de la défense. Le chiffre d'affaire est constitué à 67% de commerce d'avion, 20% pour les systèmes de défense et aérospatiaux et 12% pour les hélicoptères civils et militaires [8].
- **Orange** est une société du secteur de la télécommunication et banque. Le chiffre d'affaire est constitué à 79% de prestation de service de télécommunications aux particuliers (téléphonie mobile, téléphonie fixe, accès internet), à 18% de prestation de services de télécommunications aux entreprises et de 3% de prestations de services aux opérateurs télécoms internationaux [10].
- **L'Oréal** est une société du secteur de cosmétique. Le chiffre d'affaire est constitué de la vente à 42% de produits cosmétiques grand public, 36% de produits cosmétiques de luxe, 11% de produits professionnels et 11% de produits cosmétiques actives [9].
- **Renault** est une société du secteur de l'automobile. Le chiffre d'affaire est constitué à 92% de vente de véhicule et 8% de prestations de services (location, entretien, etc.) [11].
- **TotalEnergies** est une société du secteur de l'industrie pétrolière et de gaz naturel. Son chiffre d'affaire est constitué principalement à 40% du raffinage et chimie, 45% pour la distribution de produit pétroliers et 12% de génération d'électricité [12].

L'idée est de récupérer des actions qui seraient liées à ces actions à prédire ou qui ont un lien avec leurs compositions.

1.2 Les matières premières

Les matières premières sont potentiellement intéressantes à intégrer dans le modèle. L'évolution de leur prix peut se répercuter ensuite sur l'évolution de l'action étudiée. Le cours de certaines des matières principales ont donc été scrapé : or, l'argent, le cuivre, le pétrole et le gaz naturel.

1.3 La météo et cycle lunaire

Les informations sur la météo ne semblent pas forcément pertinentes à première vue. Des données correspondantes à la région Parisienne sont scrapées sur l'ensemble de l'année 2020. Ces données étaient disponibles par le biais d'une étude sur le Covid-19, présentant entre autre les informations quotidiennes de 119 stations en France Métropolitaine pour les études de liens entre météorologie et Covid-19 [7]. On y trouve le niveau des précipitations, la température, le vent et le niveau d'humidité.

De manière plus expérimentale encore, on décide de récupérer les stades de cycle lunaire et notamment le pourcentage de visibilité de la lune avec les dates quotidiennes correspondantes. Aucune présomption particulière n'est attendue avec ces données mais certains journalistes financiers semblent penser qu'une corrélation est présente entre la bourse et les astres [6] ...

1.4 Les réseaux sociaux

Les réseaux sociaux représentent une mine d'information sur les caractéristiques et tendance d'une entreprise. Il serait particulièrement intéressant de les intégrer dans le modèle, mais nécessitent un traitement du langage naturel pour les rendre quantifiable par le modèle, par exemple avec une analyse des sentiments.

En terme de réseaux sociaux, le plus pratique semble être Twitter et de nombreux travaux ont été effectué à l'aide des données fourni par Twitter. Il existe plusieurs packages Python qui permettent d'utiliser l'API Twitter mais les résultats n'ont pas été satisfaisant, principalement dû aux limites de cette solution gratuite [5] :

- la connexion à l'API Twitter nécessite un compte enregistré et validé par une adresse e-mail et un numéro de téléphone.
- les données disponibles sont valables pour les 30 derniers jours à partir de la date de requête. L'accès à l'historique des données pour obtenir les tweets datant de 2020 nécessite un accès premium payant.

L'analyse par les tweets est donc avortée pour cette étude qui nécessite les données historisées de l'année 2020. Cependant, pour une analyse actualisé quotidiennement, il peut être intéressant d'y implémenter cette analyse.

1.5 L'actualité

La récupération d'information est tout de même primordiale dans le monde boursier pour prendre des décisions d'achat/vente sur les actions. Le site boursier.com présente plusieurs avantages en ce sens. Il fournit des informations sur les différentes actions, les cours en vigueur et possède également une section d'actualité, répertoriant les titres d'articles liés à l'action recherchée [4].

Il est décidé de récupérer l'ensemble des titres d'article de presse avec leur dates respectives afin d'y effectuer une analyse de sentiment, qui pourrait potentiellement aider à la prédiction des cours boursiers. Le site a mis en place une barrière pour limiter le web scraping avec `beautiful soup` et il faut utiliser un moyen détourné pour récupérer l'ensemble de ces informations. Les pages sont donc téléchargées au format HTML au préalable, avant d'être analysées avec le package `lxml` et d'en extraire les dates, ainsi que les titres. Le traitement pour l'exploitation de ces données est présenté dans la partie suivante.

Une piste d'amélioration serait d'effectuer une analyse de sentiment sur chaque article qui est référencé et qui pourrait ainsi fournir de meilleurs score à analyser.

1.6 Features Selection

L'ensemble des types de variables sélectionnées est présenté dans cette section.

1.6.1 L'index

La granulométrie de la base de donnée est la journée. Si la granulométrie des variables se trouve être plus fine qu'un pas de temps à la journée, la donnée est agrégée (souvent par une moyenne si cela a du sens) pour avoir dans la base de donnée une ligne par jour. L'index de la base de données est donc un timestamp de journée unique. Sur l'année 2020, on compte 250 jours ouvrés avec des cours boursiers disponibles. La base compte donc 250 lignes.

1.6.2 Les variables y à prédire

Les variables y (dépendantes ou endogènes) que l'on souhaite prédire ont été présentés dans la section précédente. Pour chacune de ces actions, on ne conserve que la **valeur des cours de sorties ajustées**.

1.6.3 Les variables de la matrice X

Les variables de la matrice X (indépendantes ou exogènes) que l'on utilise pour prédire la variables y sont constituées de plusieurs types de variables.

- Certaines de ces variables seront présentées dans la partie concernant le *Features Engineering*. On retient seulement dans cette partie que les premières variables explicatives sont les mêmes que les variables expliqués mais créées dans une fenêtre glissante pour permettre la prédiction. La structure de la base de données est donc un peu particulière pour l'utilisation du réseau de neurone de type LSTM. Pour ces variables, on ne retient que la valeur des cours de sorties ajustées.
- Une recherche est effectuée sur le lien entre les actions du CAC40 et les entreprises concurrentes présentes à l'étranger (notamment États-Unis, Chine ou pays de l'Europe). Il est très probable que le cours de ces entreprises, si leur poids est supérieur, se répercutent ensuite sur le cours des entreprises françaises et donc pourrait faciliter la prédiction.

1.7 Data Cleaning

1.7.1 Valeurs manquantes

Les valeurs manquantes sont comblées par la moyenne de la valeur précédente et la valeur suivante. Ces valeurs manquantes sont principalement dû à la fermeture des bourses françaises durant les jours fériés.

Chapitre 2

Modélisation LSTM pour la prédiction de série temporelle

2.1 Features Engineering

2.1.1 Standardisation des données

L'ensemble des variables sont traitées dans un pipeline et sont mises à l'échelle pour éviter des calculs intensifs. Il est recommandé de prendre la normalisation, en particulier lorsqu'on travaille sur un RNN avec une fonction sigmoïde dans la couche de sortie.

2.1.2 Transformation en données exploitables

Une structure de données spéciales est nécessaire pour entraîner le réseau de neurone LSTM sur une fenêtre glissante de jour. En effet, le LSTM apprend les séquences de cours d'action pour prédire le terme suivant. Par exemple, les réseaux de neurones ont principalement été entraînés sur une fenêtre glissante de 5 jours pour prédire le 6e jour. Pour ce faire, la base est remodelée afin de contenir autant de colonnes que de termes de la séquence. Si l'entraînement est réalisé sur une fenêtre glissante de 5 jours, la base contiendra donc 5 colonnes comprenant les cours d'action de t-5 jusqu'en t-1, pour essayer de prédire t.

Considérons la séquence temporelle de données suivante :

$$[10,\ 20,\ 30,\ 40,\ 50,\ 60,\ 70,\ 80,\ 90,\ 100]$$

Voici la matrice X et y correspondant au remodelage :

$$X = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 \\ 20 & 30 & 40 & 50 & 60 \\ 30 & 40 & 50 & 60 & 70 \end{pmatrix} \qquad y = \begin{pmatrix} 60 \\ 70 \\ 80 \end{pmatrix}$$

2.2 Les modèles LSTM

Les LSTM peuvent être utilisés pour modéliser des problèmes de prévision de séries temporelles comme la prévision de cours boursier dans le temps. Il en existe de plusieurs types et certains

de ceux qui ont été testé sont présentés ci-dessous [13]. Pour utiliser ce type de modèle, il est nécessaire de préparer la base de données au préalable pour qu'elle respecte une certaine structure. Cette préparation est présentée dans la partie précédente du *feature engineering*.

2.2.1 Les modèles LSTM univariés

On s'intéresse dans un premier temps aux modèles LSTM univariés, c'est-à-dire qui sont constitués d'une seule série d'observation. Dans notre cas, cela revient à prédire la valeur boursière d'une action uniquement à partir des valeurs boursière précédentes de cette même action.

Vanilla LSTM

Un Vanilla LSTM est un modèle LSTM qui a une seule couche cachée d'unités LSTM, et une couche de sortie utilisée pour faire une prédiction.

Les premiers tests de modélisation sur chacune des actions sont effectués avec ce type de LSTM très simple, pour avoir un avant goût de la puissance de ce type de réseaux neuronaux. L'apprentissage glissant est effectué sur 5 jours afin de prédire le 6e jour.

Aux premiers abords, les résultats sont époustouflants. Les courbes de prédictions sont très sensiblement liées aux courbes réelles. Cependant, les résultats étaient moyennement efficace et on remarque immédiatement un retard dans les prédictions. En effet, il y a comme un effet de rattrapage du réseau de neurone quand il mesure sa prédiction avec la prédiction réelle quand elle survient le jour suivant. Le réseau rattrape donc son retard sur la prédiction suivante, ce qui prend la forme d'une sorte de délai de décalage dans la prédiction.

Stacked LSTM

Une approche intéressante consiste à ajouter d'autres couches cachées de neurone dans le réseau. Ces modèles sont appelés stacked LSTM (ou LSTM empilée). Les résultats étaient plus intéressant et les prédictions du réseau étaient de meilleur qualité.

On rappelle que ces modèles sont toujours entraînés avec une seule série d'observation, ce sont des réseau LSTM univariés.

Bidirectional LSTM

Une autre approche intéressante est le LSTM bidirectionnel (bi-LSTM). Le principe est de permettre au modèle LSTM d'apprendre la séquence d'entrée à la fois en avant et en arrière, puis de concaténer les deux interprétations.

Il est apparu que les résultats s'amélioraient et le retard s'atténue encore un peu plus avec ce type de modèle.

CNN-LSTM

Une approche non explorée mais très intéressante est le CNN-LSTM. C'est un réseau neuronal convolutif développé pour travailler sur des données bidimensionnelles. L'idée serait d'utiliser ce type de réseau dans un modèle hybride avec un LSTM sur des sous-échantillons de séquence temporel.

2.2.2 Les modèles LSTM multivariés

Les données de séries chronologiques multivariées sont des données pour lesquelles il y a plus d'une observation pour chaque pas de temps. C'est particulièrement intéressant dans notre cas pour prédire les cours boursiers car nous disposons de multiples données qui représentent potentiellement des effets de causalité sur les fluctuations des cours. Il est alors indispensable d'intégrer dans notre modèle, en plus du cours d'actions lui-mème, certaines variables potentiellement corrélées.

On note qu'il est possible d'utiliser le LSTM multivariés avec tous les types de LSTM univariés cités précédemment.

2.3 Natural Language Processing (NLP)

L'idée est d'utiliser les avancées en NLP pour effectuer des analyses de sentiment sur l'actualité concernant les actions que l'on souhaite prédire.

2.3.1 Textblob

Pour effectuer un premier test dans le domaine de l'analyse de sentiment par NLP, la librairie *Textblob* semble tout à fait intéressante. Il est basé sur NTLK et pattern et permet de façon plutôt simple d'effectuer plusieurs types d'analyse, dont l'analyse de sentiment.

Le processus consiste à déterminer l'attitude ou l'émotion de l'auteur, c'est-à-dire si elle est positive, négative ou neutre. La fonction sentiment de *Textblob* renvoie deux propriétés, la polarité et la subjectivité. On estime ici que la subjectivité est faible puisque le corpus est tiré de l'actualité. On s'intéresse principalement à la polarité des sentiments. Elle est comprise dans un intervalle de [-1,1] où 1 signifie une déclaration positive, 0 est neutre et -1 une déclaration négative.

La librairie *Textblob* est déjà entraînée pour prédire la polarité des sentiments. On calcule donc directement les scores et on les intègre dans la base de données, pour effectuer les simulations de prédictions en fonction de différentes variables, dont le cours boursier et l'analyse de sentiment selon l'actualité de l'action.

2.3.2 BERT

Dans une autre mesure, les équipes de Google Research ont développé le modèle de langage BERT (Bidirectional Encoder Representations from Transformers) [2]. C'est également un réseau de neurone de type Bi-LSTM. Le modèle a été adapté à la langue française en 2019 avec les modèles CamemBERT et FlauBERT, entraîné sur plusieurs Go de littérature, permettant d'améliorer significativement les performances en traitement automatique des langues.

Dans la mesure où les modèles BERT sont plus entraînés que le modèle de la librairie *Textblob*, les résultats d'analyse de sentiment n'en seront que plus précis.

En partant de l'hypothèse que l'actualité est corrélée à la tendance haussière ou baissière du cours de l'action visée, la prédiction n'en serait que meilleurs. Ces travaux seront implémentés dans la prochaine version du projet.

Chapitre 3

Analyse des résultats

3.1 Résultats et Gestion du portefeuille

L'analyse des entraînements du modèle se fait sur deux axes.

3.1.1 Fonction de perte

Dans un premier temps, on observe les résultats de la fonction de perte de l'erreur quadratique moyenne (MSE). Ces résultats de la fonction diminuent au fur et à mesure que le modèle est entraîné et tendent vers 0. En effet, le modèle examine de nombreux exemples et tente de trouver un modèle qui minimise la perte. On parle de minimisation du risque empirique. Voici la formule du MSE [3] :

$$MSE = \frac{1}{N} \sum_{(x,y)\in D} \left(y - prediction(x)\right)^2$$

3.1.2 Ratio de prédiction

Dans un second temps, on observe directement les résultats financiers en appliquant la stratégie modélisée. L'entraînement du modèle est effectué sur les 9 premiers mois de l'année 2020 et la prédiction est effectuée sur les 3 derniers mois de cette même année. On observe donc des résultats pour 3 mois.

Ces résultats sont notamment composés du delta entre la valeur du portefeuille de départ (défini à 10 000 € pour chaque action) et la valeur du portefeuille à la fin des prédictions. On notera la conjoncture particulière liée à la crise du Covid-19 avec la dégringolade générale des cours boursiers pour le début de l'année 2020 et qui remontent progressivement vers la fin de l'année. Cela sous-entend que les investissements effectués avant la remontée générale des prix provoque également une hausse de la valeur du portefeuille. Les résultats vont donc potentiellement dans le sens de notre stratégie, puisque l'idée fondamentale est d'augmenter la valeur du portefeuille.

La second indicateur de notre stratégie boursière, et certainement le plus pertinent, est le ratio concernant les variations des cours d'actions. Ce ratio indique les prédictions qui se sont finalement réalisées dans la bonne tendance, c'est-à-dire à la hausse ou à la baisse. L'idéal est de faire mieux que le hasard, qui est estimé à un ratio de 50% (soit 50% de tendance bien prédit et 50% de tendance erronée).

3.1.3 La stratégie boursière

La stratégie boursière mise en place est extrêmement simple. Par rapport à un prix d'action actuelle (en date t) :

- on souhaite acheter le maximum d'action possible en date t si on prédit que le prix de cette même action va augmenter en $t+1$.
- inversement, si on prédit que le prix de cette action va baisser en $t+1$, alors on vend le maximum des actions possédées.

Il serait très intéressant d'ajouter des conditions de no-go, c'est-à-dire d'inaction, lorsque la prédiction effectuée indique un ratio d'évolution trop faible. Autrement dit, si le cours de l'action varie trop faiblement d'une période de temps à l'autre, aucune décision n'est exécutée.

On prend pour hypothèse que les ordres sont passés au niveau du marché et donc immédiatement effectués.

3.2 Résultats des prédictions

3.2.1 Airbus

L'action Airbus a subi plusieurs tests, effectué sur les hyper-paramètres. On a tout d'abord tenté d'ajouter plusieurs couches de neurones, puis changer la période de temps glissante entre 5 et 20 jours, ensuite tester avec des actions concurrentes françaises ou étrangère, et enfin en appliquant une analyse de sentiment sur l'actualité.

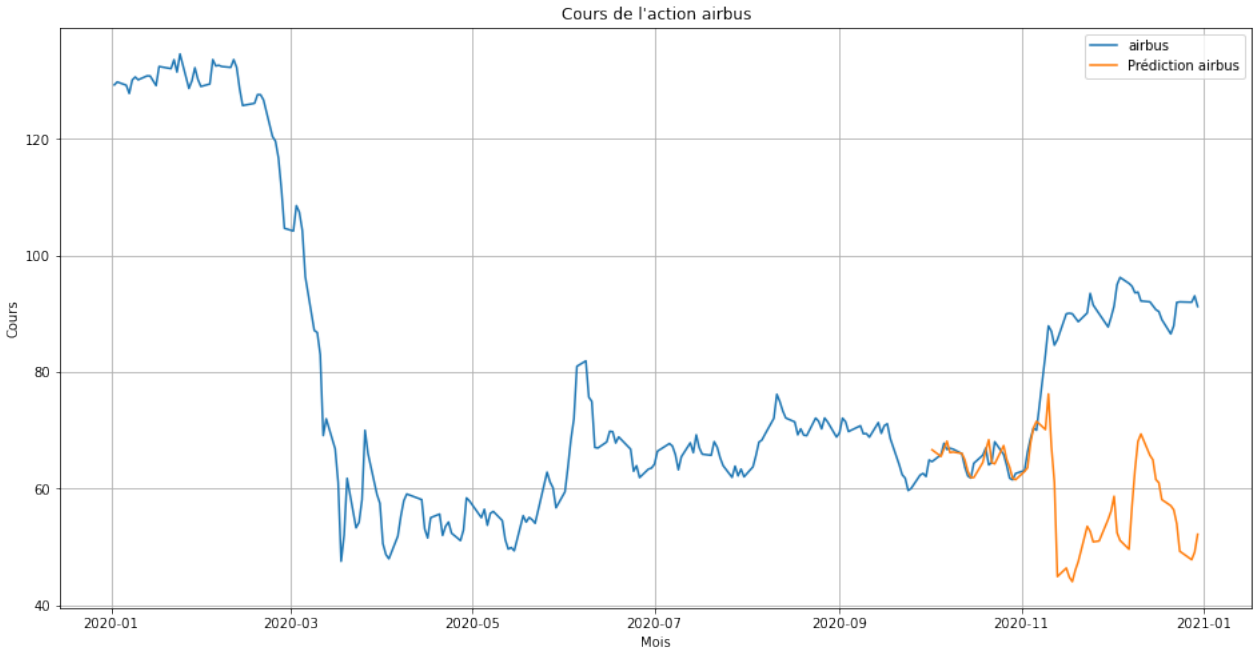


FIGURE 3.1 – Cours boursier d'Airbus

Assez étonnement, le meilleur modèle est resté le plus simple. Une seule couche de 50 neu-

rones, sans autre action. Le rendu présente une courbe assez peu représentative de l'action pour les 2 derniers mois de la prédiction. Malgré cela, le taux de prédiction a atteint plus de 67% de précision, ce qui est le meilleur score sur l'ensemble des prédictions effectuées. Partant d'un portefeuille à 10 000€, la somme totale récolté en comptant les frais de broker sont à 14 526.68€ pour seulement 6 transactions. Ce phénomène s'explique par simple analyse graphique : on remarque immédiatement que le cours de prédiction est inférieur aux cours réels pour les deux derniers mois, donc le robot n'achète pas d'actions et le cours continue de monter, accroissant la valeur du porte-feuille.

3.2.2 Orange

La prédiction du cours d'Orange est un cas intéressant de prédiction. La courbe est plutôt bien suivi, même si on peut déceler un léger petit retard de correction dans les prédictions. Le score de prédiction juste est de 57%, ce qui est plutôt honorable. L'action en l'état est de faible valeur, puisqu'elle tourne autour des 12€. On ne s'attend donc pas à obtenir un portefeuille explosif mais les gains sont quand même de presque 1000€ en 3 mois : 10 972.43€ pour 18 prédictions.

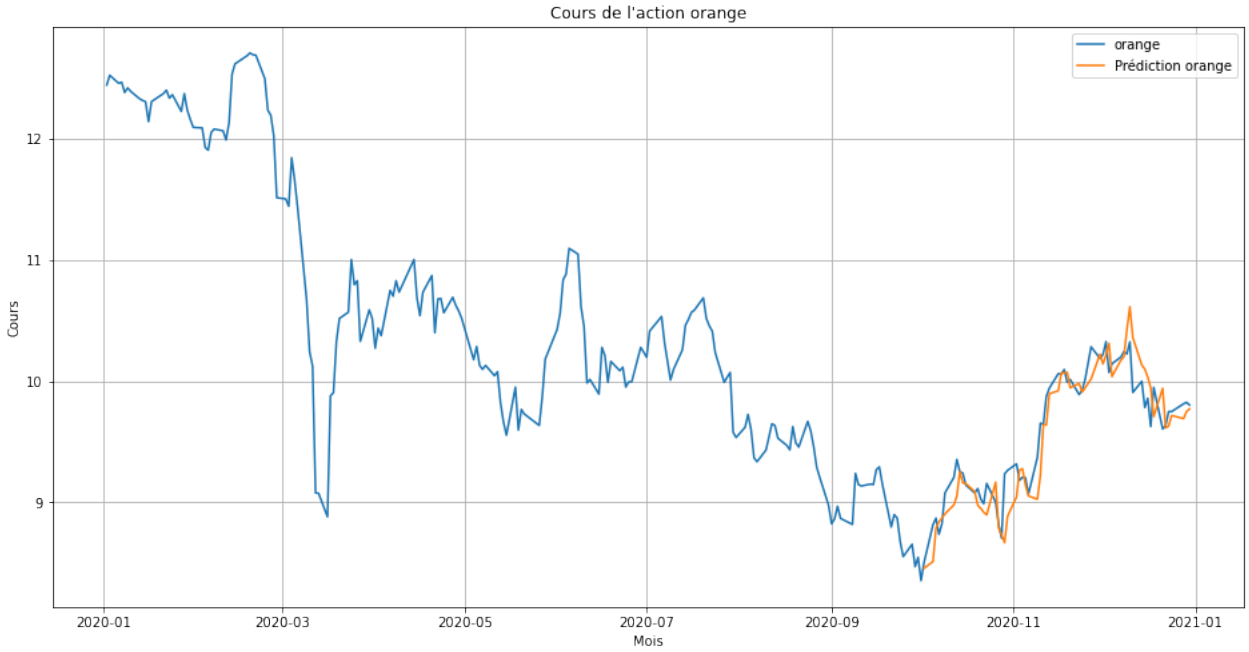


FIGURE 3.2 – Cours boursier d'Orange

3.2.3 L'Oréal

L'action L'Oréal a été de loin l'action la plus difficile à prédire pour obtenir des résultats corrects. Cela peut s'expliquer par la valeur de l'action qui est aux environs de 280€.

On atteint difficilement les 52% de valeurs correctement prédite, avec ou sans concurrents introduits dans le modèle. Le modèle dispose de deux couches cachées, ce qui a permis de dépasser les 50%. Les analyses de sentiments n'ont eu que des effets indésirables en augmentant la variance

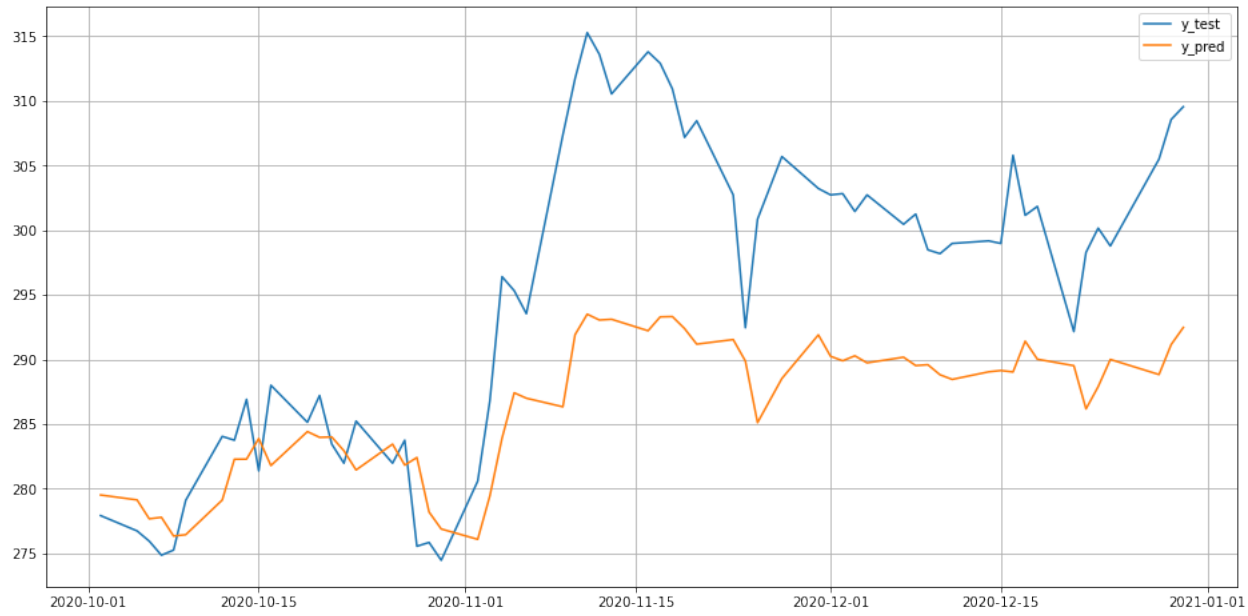


FIGURE 3.3 – Cours boursier de L'Oréal

fortement de la courbe de prédiction. Seulement 6 transactions ont été effectué pour une valeur de portefeuille finale de 10 638.99€.

3.2.4 Renault

L'action Renault était très intéressante à analyser car plusieurs tests de différents types ont été plutôt concluant. De manière intéressante, l'action de Renault se prédit plutôt bien quand elle est couplée avec l'action d'Aston Martin.

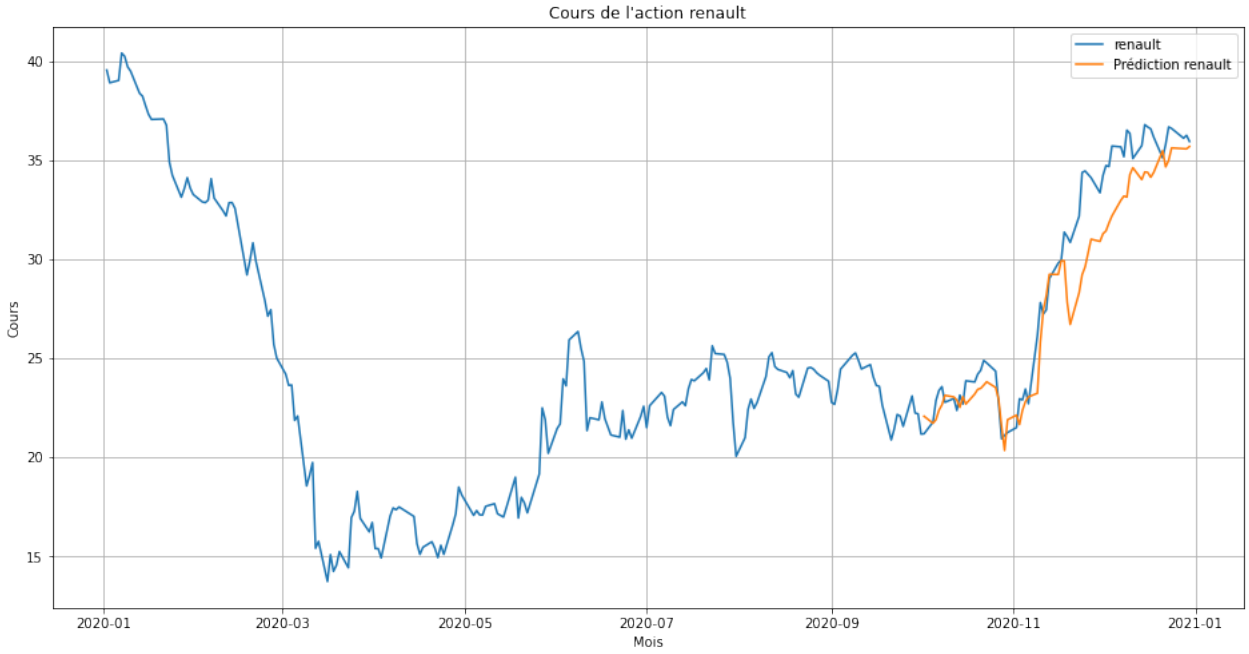


FIGURE 3.4 – Cours boursier de Renault

Le taux de précision est à 63% de justesse et le portefeuille a littéralement explosé, pour obtenir le plus haut niveau de portefeuille à 15 495.47€ pour seulement 12 transactions. Cela s'explique également par l'explosion du cours qui a presque doublé sur les trois derniers mois. Le réseau ayant été entraîné sur des données moins volatiles et majoritairement baissière, les prédictions se retrouvent pendant les deux derniers mois inférieur à aux données réelles.

3.2.5 Total

L'action TotalEnergie est l'une des plus intéressantes et nous permet de démontrer qu'un modèle avec de nombreux paramètres n'est pas forcément moins efficace. Ce modèle a été entraîné avec pas moins de 5 autres concurrents, à savoir : Exxon Mobil, Royal Dutch Shell, Chevron Corporation, BP et Petro China. En plus de cela, le modèle a également été entraîné avec l'analyse de sentiment des actualités de Total.



FIGURE 3.5 – Cours boursier de Total

Le résultat obtenu est de 61% de prédictions de tendance correcte. Enfin, en partant d'un portefeuille de 10 000€, la somme finale récoltée s'élève à 12 823.17€ pour 19 transactions effectuées.

On remarquera que le dernier cours prédit pour chacune des actions présentées est toujours toujours inférieurs au cours réel. Ainsi, le portefeuille est toujours composés d'argent réel sans aucune action.

Chapitre 4

Industrialisation du projet

4.1 Structure du projet : *Cookiecutter*

Dans l'optique d'industrialisation et de reproductibilité de ce projet, il est primordiale d'adopter une structure logique et raisonnablement normalisée. Pour cela, Cookiecutter propose une structure plutôt bien pensé [1]. Cette structure est disponible dans l'annexe 1 en fin de document pour plus de détails.

L'utilisation du notebook Jupyter était donc limité aux tests. Les modèles satisfaisants ont été exportés en format binaire .joblib afin d'être conservé et réexploité. Les fonctions d'entraînements ont été placé dans des fichiers de fonctions .py.

Les fichiers makefile, maketrain, etc. doivent encore être scripté avant de terminer l'industrialisation du projet et compléter le caractère de reproductibilité des résultats.

4.2 Mise en production et hébergement web

Pour finaliser l'objectif du projet en *full-stack*, il était primordiale de présenter un dashboard composé de graphiques et d'indicateurs clés (KPI) en ligne. Ce tableau de bord est disponible à l'adresse suivante : `https://datasoupe.fr/dashboard/`.

4.2.1 JavaScript

Le tableau de bord est créé quasi-exclusivement avec du JavaScript, qui a nécessité une formation. Le JavaScript permet ici de charger les différentes bases de données en .csv avec la librairie d3js. L'interface est rendu dynamique en offrant la possibilité de changement d'action par des boutons. Les graphiques sont générés par la librairie Chart.js.

4.2.2 Serveur Raspberry Pi

L'hébergement du dashboard est effectué sur un Raspberry Pi transformé en serveur avec nginx. Le dashboard a été intégré sur un site personnel statique développé à l'aide de Jekyll, ce qui a également nécessité une formation.

4.2.3 Le dashboard

Le tableau de bord se présente de la manière suivante. Dans un premier temps, des boutons sont définis pour sélectionner l'action voulu. Deux courbes sont présentées dans ce graphique : le cours de l'action réel sur l'année entière et sa prédiction sur les 3 derniers mois.

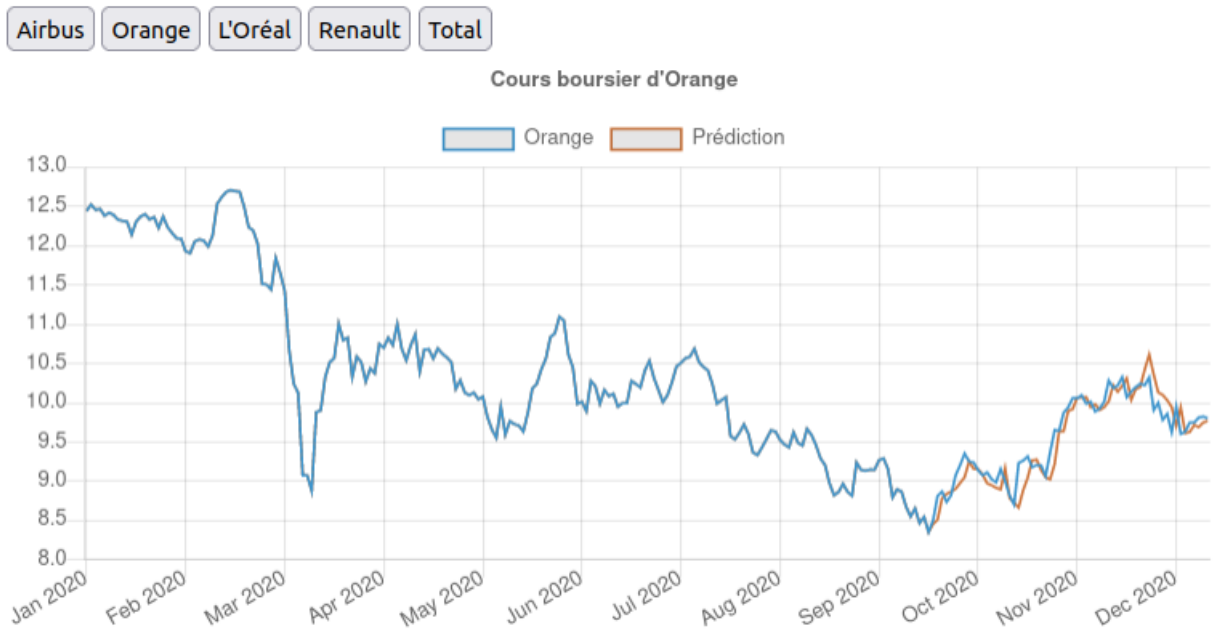


FIGURE 4.1 – Dashboard - Graphique

Ensuite, des KPI sont disposés afin de donner plus d'informations sur les résultats de la modélisation et de la stratégie adopté. On retrouve notamment :

- La valeur du portefeuille finale en fin d'année.
- Le nombre d'action présent dans le portefeuille à la fin de la période.
- Le nombre de transactions totales effectuées sur la période de test du modèle, donc sur les 3 derniers mois.
- Le ratio en pourcentage des prédictions qui se sont révélées correctes avec la prédiction.



FIGURE 4.2 – Dashboard - KPI

Enfin, à titre plus expérimentale, on trouve un camembert présentant la part de différentes stratégies à adopter selon leur degré de risques. On pourrait apparenté cela à une mesure de la variance dans les données mais rien n'est encore défini et les données présentées dans le camembert sont aléatoires. Elles n'ont donc aucun sens à l'heure du rendu de ce rapport.

On trouve à côté un tableau présentant l'ensemble des actualités scrapées avec les dates de publicationsconcernant l'action en question. La dynamique n'est pas encore au point car le tableau n'est pas encore variable selon l'action sélectionné. Cette amélioration est en cours et sera validé lors de la prochaine version du dashboard.

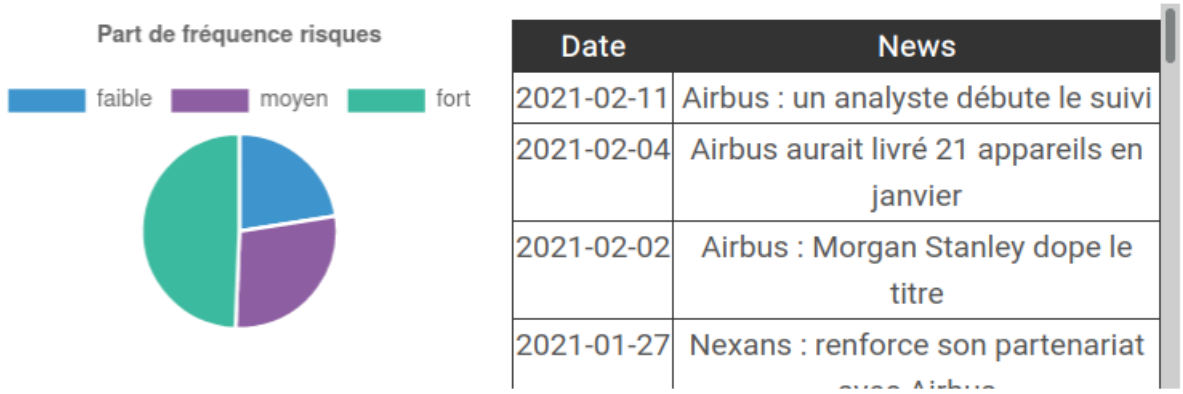


FIGURE 4.3 – Dashboard - News

4.2.4 Les évolutions de versions à venir

L'industrialisation de ce projet est une énorme plus-value dans le domaine technique et dans une perspective *full-stack*. La maîtrise totale du projet aide à prendre conscience des difficultés que l'on peut rencontrer dans une perspective DevOps.

Plusieurs améliorations sont prévus pour y implémentant de nouvelles fonctionnalités. Il serait notamment très intéressant de tester le modèle sur les données les plus récentes et d'observer les résultats. L'actualisation des données pourrait se faire de façon journalière avec la modélisation.

Conclusion

Ce rapport présente la création du début à la fin d'un projet *full-stack* sur la prédiction de cours boursiers du CAC40, en présentant les différents points de réflexion, les limites et les résultats. Ce projet était très formateur et permet de mettre un point d'honneur pour la conclusion du D.U. avec l'utilisation de plusieurs techniques apprises durant la formation.

Concernant les résultats, plusieurs remarques et pistes de réflexions sont envisagées : les graphiques avec une prédiction bien superposée ne sont pas forcément les meilleurs modèles. Les résultats de la fonction de perte ne sont pas non plus toujours révélateurs des bonnes performances du modèle. Tout doit être observé et analysé dans la totalité.

Une autre piste très intéressante concerne le type du modèle du réseau de neurone. Ces réseaux ont été entraînés pour des problèmes de régression, c'est-à-dire qu'on essayait de déterminer le cours précis. Ensuite, les résultats étaient analysés pour observer la tendance (haussière ou baissière) et la décision d'achat ou de vente étaient déterminés par ces résultats. L'autre approche consisterait à transformer ce problème de régression en problème de classification. Le réseau s'entraînerait directement avec la tendance du cours de l'action et prédirait la tendance, selon le cours d'autres actions, actualités, etc.

Enfin, il pourrait être intéressant de tester d'autres méthodes de *machine learning*, pour cette même problématique de régression et de classification. Des travaux ont par exemple été effectués sur des comparaisons de différents modèles et les SVM offriraient par exemple des résultats particulièrement intéressants.

Références

[1] *Cookiecutter Data Science* . https://drivendata.github.io/cookiecutter-data-science/#cookiecutter-data-science. Date d'accès : 2021-04-17.

[2] *Open Sourcing BERT : State-of-the-Art Pre-training for Natural Language Processing* . https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html. Date d'accès : 2021-06-24.

[3] *Plongée dans le ML : apprentissage et pertebookmark_border* . https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss?hl=fr. Date d'accès : 2021-06-29.

[4] *Boursier.com*. https://www.boursier.com/. Date d'accès : 2021-06-20.

[5] *Conditions applicables aux développeurs - Twitter*. https://developer.twitter.com/fr/developer-terms/agreement-and-policy. Date d'accès : 2021-06-20.

[6] *La lune prédirait les cours de la Bourse*. https://www.20min.ch/fr/story/la-lune-predirait-les-cours-de-la-bourse-162522100814.

[7] *Paramètres quotidiens de température, humidité, précipitations et force du vent de 119 stations en France Métropolitaine du 01/01/2020 au 21/04/2021*. https://www.data.gouv.fr/fr/datasets/donnees-quotidiennes-de-119-stations-en-france-metropolitaine-pour-les-etudes-de-li Date d'accès : 2021-04-24.

[8] *Profil Airbus*. https://www.boursier.com/actions/societe/profil/airbus-group-ex-eads-NL0000235190,FR.html. Date d'accès : 2021-06-19.

[9] *Profil L'Oréal*. https://www.boursier.com/actions/societe/profil/l-oreal-FR0000120321,FR.html. Date d'accès : 2021-06-26.

[10] *Profil Orange*. https://www.boursier.com/actions/societe/profil/orange-FR0000133308,FR.html. Date d'accès : 2021-06-26.

[11] *Profil Renault*. https://www.boursier.com/actions/societe/profil/renault-FR0000131906,FR.html. Date d'accès : 2021-06-26.

[12] *Profil TotalEnergies*. https://www.boursier.com/actions/societe/profil/totalenergies-ex-total-FR0000120271,FR.html. Date d'accès : 2021-06-19.

[13] J. BROWNLEE, *Deep learning for time series forecasting : predict the future with MLPs, CNNs and LSTMs in Python*, Machine Learning Mastery, 2018.

[14] G. DANIEL AND T. AMINE, *La Bourse*, La Découverte, 2006.

Table des figures

3.1 Cours boursier d'Airbus 14
3.2 Cours boursier d'Orange 15
3.3 Cours boursier de L'Oréal 16
3.4 Cours boursier de Renault 16
3.5 Cours boursier de Total 17

4.1 Dashboard - Graphique 19
4.2 Dashboard - KPI 19
4.3 Dashboard - News 20

Annexe A

Structure de Cookiecutter

```
Dashboard_repository
├── LICENSE
├── Makefile ..... Makefile with commands like 'make data' or 'make train'
├── README.md ..... The top-level README for developers using this project
├── data
│   ├── external ..... Data from third party sources
│   ├── interim ..... Intermediate data that has been transformed
│   ├── processed ..... The final, canonical data sets for modeling
│   └── raw ..... The original, immutable data dump
├── docs ..... A default Sphinx project; see sphinx-doc.org for details
├── models ..... Trained and serialized models, model predictions, or model summaries
├── notebooks ..... Jupyter notebooks. Naming convention is a number (for ordering), the creator's initials, and a short '-' delimited description, e.g. '1.0-jqp-initial-data-exploration
├── references ..... Data dictionaries, manuals, and all other explanatory materials
├── reports ..... Generated analysis as HTML, PDF, LaTeX, etc
│   └── figures ..... Generated graphics and figures to be used in reporting
├── requirements.txt ..... The requirements file for reproducing the analysis environment, e.g. generated with 'pip freeze > requirements.txt'
├── setup.py ..... Make this project pip installable with 'pip install -e'
├── src ..... Source code for use in this project
│   ├── __init__.py ..... Makes src a Python module
│   ├── data ..... Scripts to download or generate data
│   │   └── make_dataset.py
│   ├── features ..... Scripts to turn raw data into features for modeling
│   │   └── build_features.py
│   ├── models ..... Scripts to train models and then use trained models to make predictions
│   │   ├── predict_model.py
│   │   └── train_model.py
│   └── visualization ..... Scripts to create exploratory and results oriented visualizations
│       └── visualize.py
└── tox.ini ..... tox file with settings for running tox; see tox.readthedocs.io
```