LA CASA DE CRYPTO

Prediction du cours des cryptomonnaies

Peut-on prédire le cours des cryptomonnaies avec précision ?

***Chef de projet** : LOUNIS Thomas*
***Responsable technique** : BOUKHEMKHAM Mohamed*
***Responsable design** : SUKUMAR Kabilash*
***Responsable communication** : BIREM Ramy*
***Responsable qualité** : GHASAROSSIAN François*

Promo 2020

Sommaire

Introduction

Le projet transverse que nous devons réaliser en équipe de 5 cette année porte sur la conception d'une application utilisant une certaine Intelligence Artificielle. Ce projet en plus de nous faire travailler et découvrir plusieurs aspects techniques en programmation nous fait interroger sur l'intelligence artificielle, sa place dans une application concrète ainsi que son importance dans le futur de demain qui semble incontournable, et ce, quel que soit le domaine concerné.

Cet intérêt sur l'intelligence artificiel est même arrivé au plus haut niveau en France, puisque c'est le président français, Emmanuel Macron, qui s'exprime sur cette révolution technologique qui, en plus de révolutionner le monde d'aujourd'hui, va apporter d'immenses bénéfices à notre société. Avec la volonté d'investir 1,5 milliard d'euros d'ici 2022, la France essaye de devenir un acteur incontournable de l'IA.

Cependant cette révolution technologique ne peut se faire qu'avec la production de talents, formés dans les écoles d'ingénieurs du monde entier. Ce projet transverse permet donc en plus d'apercevoir le potentiel et la place de l'IA dans une application donnée, de nous intéresser à ce domaine très prometteur.

En ce qui concerne notre projet transverse, nous avons décider de nous lancer dans la création d'un programme pouvant prédire le cours des 3 plus grandes cryptomonnaies qui sont : le Bitcoin, le Dash, et le Litecoin dans un futur proche.

Présentation détaillée

Pour qui ?

Ce programme permettant de prédire le cours des cryptomonnaies s'adresse principalement aux traders (quelque soit leurs niveaux d'expérience). Les traders n'auront plus besoin de se soucier de tous les détails de l'analyse technique ce qui leur fait perdre beaucoup de temps mais pourront maintenant se focaliser sur des choses plus spécifiques grâce à notre outil de prédiction. Il s'adresse également aux traders novices qui viennent de découvrir le monde des crypto-monnaies et qui pourront se baser sur notre outil de prédiction pour mieux appréhender le monde du trading.

Nous avons également les investisseurs. En effet, de nos jours, de nombreuses personnes décident d'investir dans le long terme dans le bitcoin et ne plus y toucher (c'est-à-dire ne pas trader), un peu comme les personnes qui investissent dans l'or ou bien dans l'immobilier. Notre programme leur permettra de garder un œil sur leur investissement. En effet, ils pourront se projeter plus facilement dans le futur grâce a notre outil de prédiction et donc pouvoir anticiper de futures tendances.

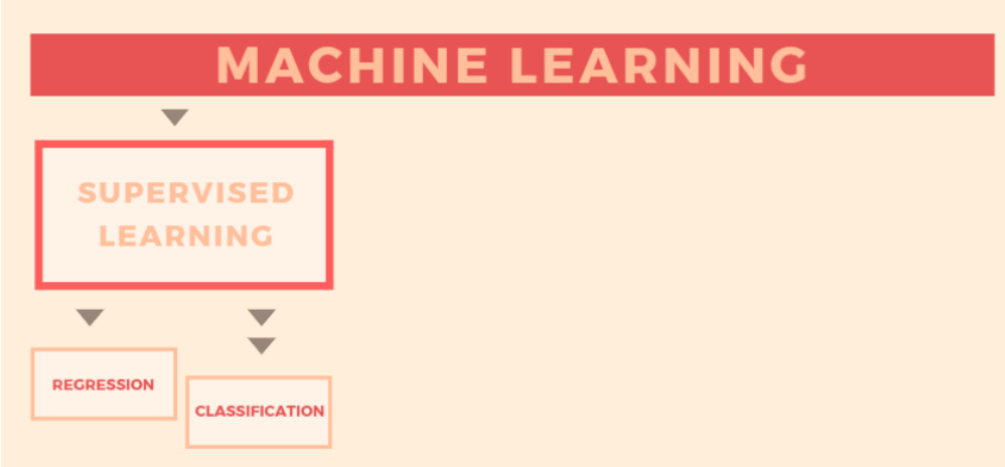
Prédire le cours du bitcoin, mais dans quel but ?

On ne va pas se le cacher, que ce soit trader ou bien investisseur le but premier d'investir dans les cryptomonnaies est de se faire du bénéfice. En quoi notre programme permettra de se faire du bénéfice et d'éviter les pertes ? Le principe est simple, nous pouvons prédire le prix du bitcoin en avance (avec un certain degré de précision), cela veut dire que nous avons un temps d'avance sur la tendance, il suffit juste d'acheter du bitcoin avant la prédiction d'une tendance haussière et vendre quand notre programme prédit une tendance baissière à venir.

Les différents aspects de l'IA utilisé

Pour créer une intelligence artificielle, nous allons utiliser différentes techniques regroupées sous une même discipline : l'apprentissage machine (ou machine Learning, en anglais). Nous nous attarderons sur l'aspect de l'apprentissage supervisé.

Actuellement, c'est la méthode la plus répandue en ce qui concerne le Machine Learning. Nous voulons transmettre cette capacité d'apprentissage par l'expérience à notre programme. Nous voulons que notre programme puisse trouver les règles associées à son existence par lui-même, que ce soit par généralisation, association ou inférence.



La régression est la capacité de notre machine à reconnaître les nombres et à les grouper pour former des prédictions. Grâce à la régression linéaire, la machine sera capable de prédire le cours des cryptomonnaies

La classification est la capacité de la machine à identifier les actions binaires (oui et non).

La variable pour prédire le cours des cryptomonnaies est son propre prix.

Cette variable de prix a deux modalités : une modalité en variable continue et une en variable catégorique. Le mode catégorique du prix n'a que deux valeurs : 1 si le prix a augmenté et 0 s'il a baissé.

Ces différentes modalités de la variable prix vont nous permettre d'utiliser différents modèles de machine Learning pour essayer de prédire leur valeur : modèles de régression avec le mode continue et modèles de classification avec la modalité catégorique.

Il faut savoir qu'il existe plusieurs méthodes d'apprentissage supervisé pour la classification. Par manque de temps et de connaissances, nous avons donc choisis d'utiliser un Automatic Machine Learning (AML) via une librairie du langage python qui s'appelle TPOT et qui permet de définir le meilleur modèle à utiliser pour un jeu de données donner (nous reviendrons en détail sur cette librairie ultérieurement). Pour la classification, les meilleures méthodes d'apprentissage supervisé pour nos différentes cryptomonnaies ont été :

- **Les arbres de décision** : c'est un outil d'aide a la décision qui va nous permettre de représenter les différents choix possibles sous la forme d'un arbre. Les différentes décisions possibles sont situées aux extrémités de l'arbre et sont atteints en fonction de décisions prises à chaque étape.

- **Méthode des *k* plus proches voisins :** Cette méthode est l'un des algorithmes de classification les plus élémentaires mais essentiels de l'apprentissage automatique. Il appartient au domaine de l'apprentissage supervisé. On dispose d'une base de données d'apprentissage constituée de *N* couples « entrée-sortie », Dans notre cas pour estimer la sortie, on retiendra la classe la plus représentée parmi les *k* sorties associées aux *k* entrées les plus proches de la nouvelle entrée *x*.

- **Gradient Boosting :** Il s'agit là encore d'une méthode d'agrégation de modèles. Plutôt que d'utiliser un seul modèle, nous en utilisons plusieurs que nous agrégeons ensuite pour obtenir un seul résultat. Le Boosting travaille de manière séquentielle. C'est-à-dire qu'il commence par construire un premier modèle qu'il va évaluer, puis à partir de ce modèle, chaque individu va être pondéré en fonction de la performance de la prédiction. Le but ici, est de donner un poids plus important aux individus pour lesquels la valeur a été mal prédite pour la construction du modèle suivant.

Le Bitcoin suivra donc la méthode des arbres de décision pour la prédiction en classification.
Le Litecoin suivra la méthode du Gradient Boosting pour la prédiction en classification.
Le Dash suivra la méthode des *k* plus proches voisins pour la prédiction en classification.

Les méthodes assignées aux cryptomonnaies ont été décidées par TPOT.

De même pour la régression, les méthodes assignés ont été le Gradient Boosting (version regressor) pour le Bitcoin, et extra trees regressor(pour arbres extrêmement aléatoires en francais) pour le Dash et le Litecoin. Les extra trees diffèrent des arbres de décision classiques dans la façon dont ils sont construits. Lorsque vous recherchez la meilleure répartition pour séparer les échantillons d'un nœud en deux groupes, des divisions aléatoires sont dessinées pour chacune des entités sélectionnées au hasard et la meilleure répartition parmi celles-ci est choisie. Quand max_features est mis à 1, cela revient à construire un arbre de décision totalement aléatoire.

Conception détaillée

Décomposition en sous problèmes

a) Le choix des données

Pour commencer nous devions choisir les différents paramètres d'entrée qui feront varier notre modèle et donc les prédictions.

Au départ nous nous sommes fixés comme objectif de choisir 2 ou 3 paramètres d'entrée parmi cette liste que nous avons établie :

- **Le prix de la cryptomonnaie** en question
- **L'évolution du nombre de followers** de la page Reddit (forum très connu dans le milieu) de la cryptomonnaie
- **Google trends** (évolution du nombre de recherches google de la cryptomonnaie)
- **Prix des monnaies refuges** (or, bronze …)
- **Sentiment twitter**
- **Le volume** (il reflète la force d'une cryptomonnaie, plus celle-ci est échangée plus le volume de transactions sera important. C'est l'indicateur le plus utile avec le prix pour faire du trading, la plupart des autres indicateurs sont en réalité basés sur des formules de relations entre le prix et le volume d'échanges. Par ailleurs, plus celui-ci est élevé moins le cours est susceptible d'être manipulé par des acteurs isolés et plus l'analyse technique est fiable) en euro de la cryptomonnaie en question
- **La moyenne des prix** sur les 7 derniers prix récupérer pour chaque cryptomonnaie

Au fur et a mesure de nos recherches nous nous sommes aperçus que pour construire un modèle fiable il nous fallait énormément de données. Et gérer autant de donner relève de l'impossible pour de simple étudiants en L3. Nous avons donc décider d'affiner nos paramètres d'entrée pour notre modèle et avons choisis : Le prix de la cryptomonnaie en question, les google trends, le volume et le sentiment twitter.

Le prix de la cryptomonnaie est fixé en Dollars.
Pour les google trends, le score peut aller de 0 pour aucune recherche a 100 pour une recherche top.

Le sentiment twitter quant a lui peut osciller entre -1 pour un sentiment totalement négatif, 0 pour un sentiment neutre et au maximum 1 pour un sentiment très positif.
Le volume est en bitcoin et représente le volume tradé.
La moyenne des prix sur les 7 derniers prix.

b) La récupération de données

Tout d'abord nous devons récupérer le prix des différentes cryptomonnaies et leur volume associé lors de la prise d'information. Il faut savoir qu'il existe des centaines de plate-forme d'échange permettant d'acheter, de vendre et de stocker des cryptomonnaies. La plateforme d'échange que nous avons choisi est « Poloniex » qui fournit une API qui permet de récupérer les données historiques d'une cryptomonnaie (son prix, son volume … au cours du temps) au format JSON. Nous avons choisi ce site en particulier car il ne fait pas de vérification anti robot pour leurs API contrairement à la plupart des plateformes d'échange. Ce qui nous a permis de récupérer les données (le prix, volume et aussi la moyenne des prix) à haute fréquence.
Pour ce faire nous avons créé un dossier Grabber de données qui permettra de grab toutes les données dont nous avons besoin. Nous avons implémenté les classes nécessaires dans chaque fichier poloniex.py, twitter.py …. Puis avons utilisé ces classes dans notre fichier DataGrabber.py pour récupérer nos données toute les 9 minutes (limite de recherche pour les API twitter et Google) et les stocker dans notre base de données.

c) Le stockage des données

Rien de plus simple qu'une base de données pour stocker nos données ! Pour notre moteur de base de données nous avons choisis SQLite car contrairement aux serveurs de bases de données traditionnels, comme MySQL ou PostgreSQL, sa particularité est de ne pas reproduire le schéma habituel client-serveur mais d'être directement intégrée aux programmes. L'intégralité de la base de données est stockée dans un fichier indépendant de la plateforme, ici appelé « cryptodata.db ».

En effet nous avons créé une base de données composée de 3 tables :

La première table nommée « crypto » comporte deux colonnes « symbol » et « name » qui stock respectivement le symbole et le nom complet de la cryptomonnaie (par exemple : BTC comme symbole et Bitcoin comme nom complet).
Notre deuxième table LogBook est un livre de sauvegarde des différentes données. Il contient 10 colonnes : l'ID, le symbole de la cryptomonnaie, la date de la capture des données, l'heure, le prix, la moyenne du prix, increased (1 si le prix a monté, 0 s'il a baissé par rapport à la dernière occurrence), volume qui est le volume de transaction, le score google trend et le sentiment twitter.
La dernière table quant à elle est une simple séquence du nombre d'occurrences de la table logbook.

Lors de chaque itération, c'est-à-dire lors de l'ajout d'une nouvelle ligne de données nous l'ajoutons directement a notre base de données en incrémentant simplement l'ID.

d) Création de nos modèles

Comme nous l'avons dit précédemment, nous allons utiliser la régression et la classification comme type d'apprentissage supervisé. La prochaine étape est de déterminer les algorithmes a utilisé pour la classification et la régression. Cette étape est particulièrement délicate et complexe et nécessite une certaine connaissance et maitrise du domaine de l'IA que nous n'avons malheureusement pas. Heureusement de nos jours il existe des librairies en python qui permettent d'automatiser la partie la plus fastidieuse de l'apprentissage automatique en explorant intelligemment des milliers de pipelines possibles pour trouver le meilleur pour vos données comme TPOT.

En effet, pour la classification nous laisserons TPOT choisir nos modèles pour nos 3 cryptomonnaie (BTC, LTC, DASH). Pareil pour la régression. TPOT est un outil d'optimisation de haut niveau très intéressant pour les pipelines d'apprentissage automatique utilisant le populaire paquetage Python, scikit-learn. En définissant vos données d'entrée et de sortie et en spécifiant vos ensembles d'entraînement et de test, TPOT utilise un algorithme génétique itératif pour sélectionner un modèle de sklearn et des hyperparamètres qui fonctionnent le mieux parmi ceux générés.

Partant de cette idée nous avons décidé de créer un script analytics.py

```
FONCTION MODEL_BUILDING(TYPE):
    SELECTIONNE TOUT LES CRYPTOMONNAIES DE LA TABLE CRYPTO
    POUR TOUT LES CRYPTOMONNAIES:
        SELECTIONNE LES DONNEES DE LA LOGBOOK CORRESPONDANTES A NOTRE CRYPTOMONNAIE
        NOTRE VARIABLE X <- MOYENNE DU PRIX, VOLUME, GOOGLE TREND, SENTIMENT TWITTER
        SI LE TYPE EST UN CLASSIFIEUR:
            NOTRE VARIABLE Y <- INCREASED
        SINON SI C'EST UNE REGRESSION:
            NOTRE VARIABLE Y <- PRIX
        FIN SI
        ON TRAINE LE MODEL AVEC COMME TAILLE DU TRAIN 80% DE TOUTE LES DONNEES ET TAILLE D'ESSAI 20% DE TOUTE LES DONNEES
        SI LE TYPE EST UN CLASSIFIER:
            LE MODELE SERA UN TPOT CLASSIFIEUR
        SINON SI C'EST UNE REGRESSION:
            LE MODELE SERA UN TPOT REGRESSOR
        FIN SI
        VERBOSITY DU MODELE <- 3 (COMBIEN D'INFO TPOT COMMUNIQUE PENDANT QU'IL TOURNE, 0 = aucune, 1 = le minimum, 2 = elevé, 3 = tout)
        ON ADAPTE LE MODELE
        SI LE TYPE EST UN CLASSIFIEUR:
            ON EXPORTE NOTRE MODELE AU FICHIER : CLASSIFIER_CRYPTO.py
        SINON SI C'EST UNE REGRESSION:
            ON EXPORTE NOTRE MODELE AU FICHIER : REGRESSOR_CRYPTO.py
        FIN SI
    FIN POUR


FONCTION MAIN
    ON LANCE EN PROCESSUS MULTIPLE LES DEUX TYPES DE MODELES POUR CHAQUE CRYPTO CE QUI VA UTILISER BEAUCOUP DE RESSOURCES MAIS SERA PLUS RAPIDE.
```

Le processus de création des différents modèles est une tache fastidieuse qui prends énormément de temps et de ressources. Pour pouvoir effectuer nos modèles le plus rapidement possibles nous avons louer un serveur dédié avec un processeur puissant avec suffisamment de RAM. Il faut compter en moyenne 2 heures pour la création de chaque modèle ce qui fait un totale d’environ 12 heures pour tous nos modèles. Je vous laisse imaginer le fait de laisser un ordinateur allumer 12 heures, c’est pour cela que nous avons décidé d’opter pour un serveur dédié.

Quels outils ont été utilisés ?

Pour mener à bien notre projet, nous avons décidé de développer notre application en **Python** qui est un langage très simple mais qui permet de construire des projets complexes comme le nôtre. Python nous a fait gagner beaucoup de temps. Pas de compilation, un typage très dynamique, une syntaxe succincte, un debugger intégré, un shell de tests et des stack traces très verbeuses. Ce langage a été inventé pour la productivité. De plus python donne accès à des milliers de librairies que ce soit celles fournit avec python même ou bien celles développées par des personnes extérieures.

Clairement, Python n'est pas vraiment au-dessus des autres langages pour l'intelligence artificielle. Mais il s'y prête bien, sa syntaxe concise et facile nous a permis d'avancer beaucoup plus vite dans notre projet en évitant les pertes de temps pour l'implémentation.

Il dispose aussi de quelques librairies spécialisées (une liste exhaustive ici : https://wiki.python.org/moin/PythonForArtificialIntelligence) en IA qui lui permettent de s'initier à cette discipline.

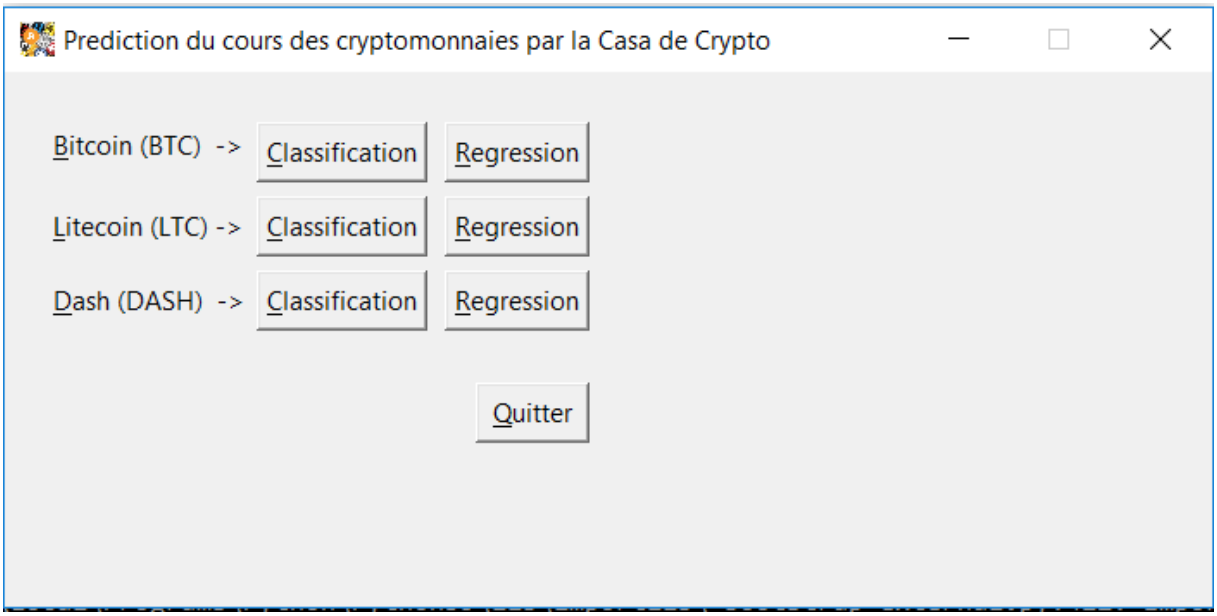
En outre, concernant le machine Learning, la problématique la plus difficile pour nous était de réussir à se procurer nos données comme par exemple le prix. Sur ce plan, Python est particulièrement bien outillé avec des librairies comme **Numpy** ou **Pandas**, par exemple qui nous ont permis de manier nos données avec une aisance déconcertante.

Enfin, concernant le domaine de l'apprentissage automatique, Python se distingue tout particulièrement en offrant une pléthore de librairies de très grande qualité, couvrant tous les types d'apprentissages disponibles. Pour notre projet, nous avons décidé d'utiliser une librairie qui est complète et simple d'utilisation : **Scikit-Learn** pour les algorithmes de machine Learning et TPOT pour l'automatisation.

Pour visualiser les données en sortie (les graphiques) nous allons utiliser la librairie **Matplotlib**.

Maquette

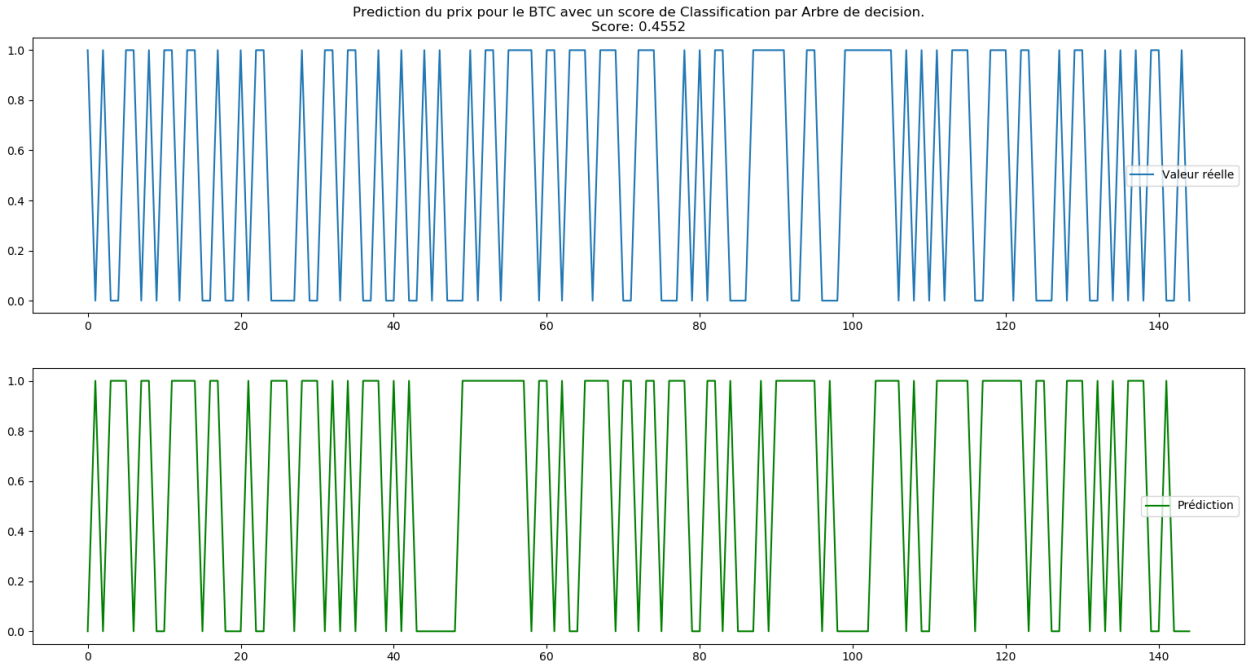
Voici notre fenêtre principale :



Nous pouvons choisir parmi nos 3 cryptomonnaies qui sont le BTC, le DASH et le LTC.
Nous pouvons également choisir le type d'apprentissage supervisé (Classification ou Regression)
Nous avons finalement un bouton quitter qui permet de fermer la fenêtre.

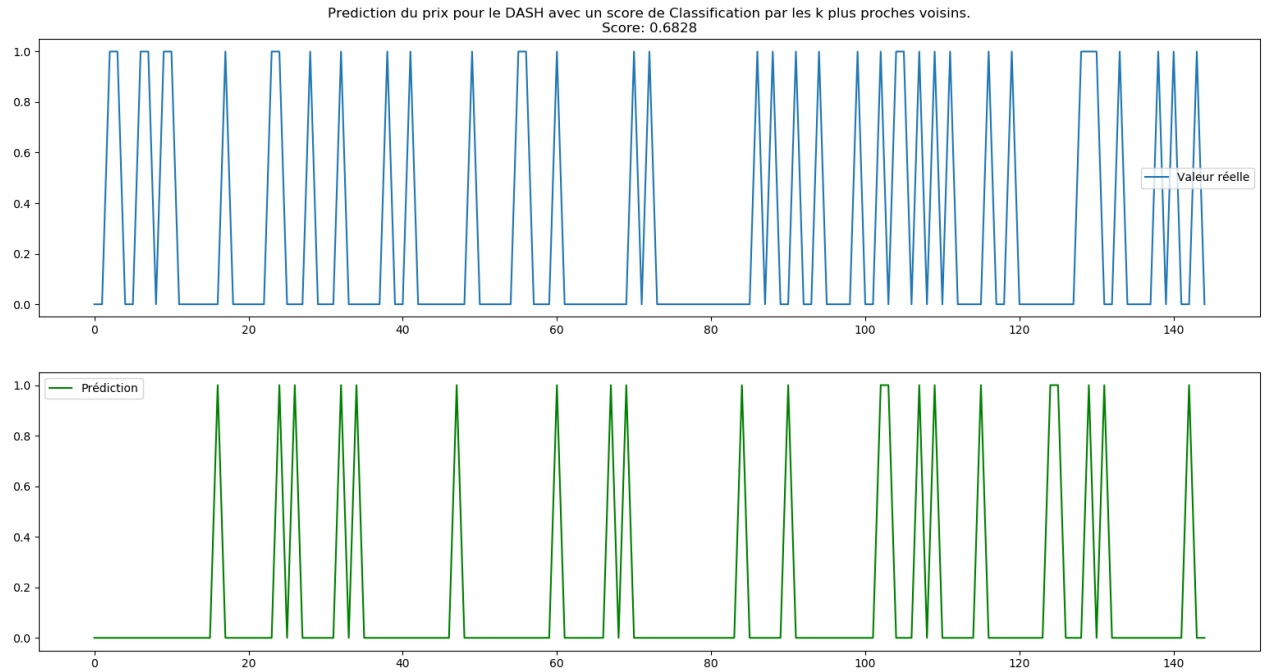
Voici les résultats obtenus pour **la classification** (score allant de 0 prédiction très médiocre, à 1 prédiction totalement identique) :

Pour le Bitcoin :



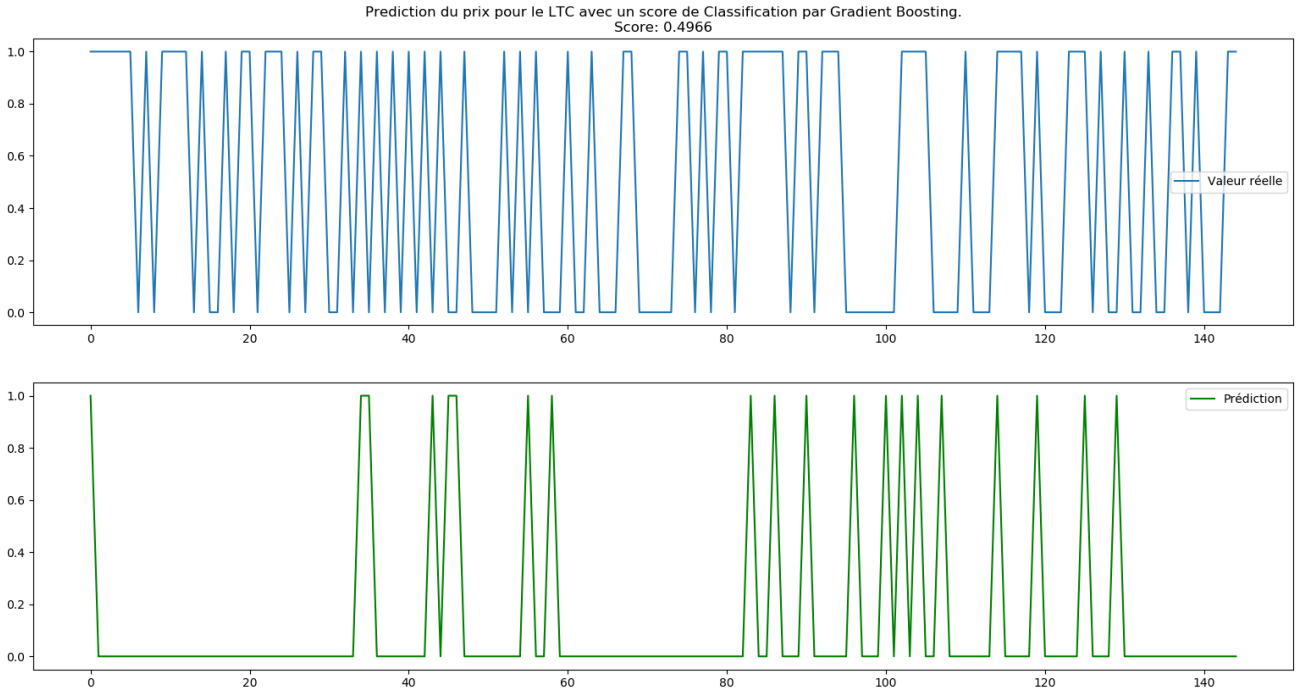
Nous avons un score de 0.4552 ce qui n'est même pas la moyenne (0.5), ce modèle ne prédis donc pas très bien le prix du bitcoin. L'abscisse est une echelle de temps et l'ordonné va de 0 (prix baisse) à 1(il augmente).

Pour le Dash :



On obtient un score de 0.6828 ce qui est plus que 0.5 (c'est déjà plus que du hasard !), il prédit donc avec perfection une montée ou une baisse à 68% ! Ce modèle est donc satisfaisant

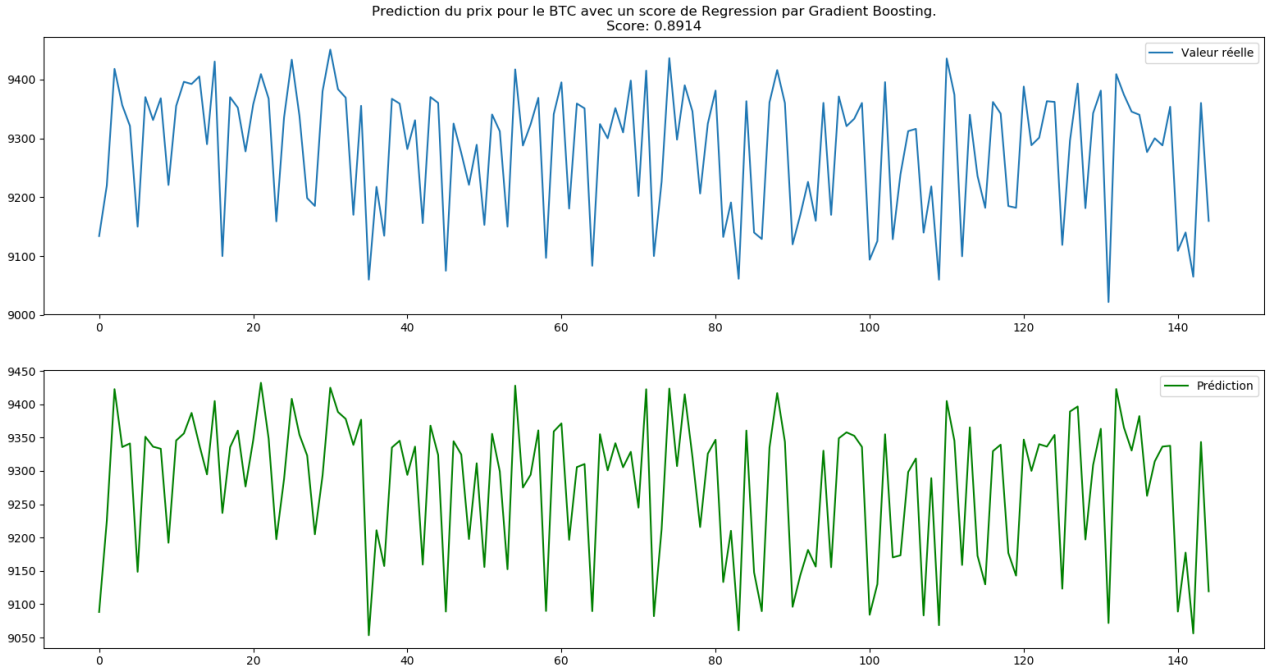
Pour le Litecoin :



On obtient un score de 0.49 ce qui veut dire que ce modèle prédit la bonne monté ou descente du Litecoin 49% du temps. Il faut donc mieux ce fier au hasard (qui fournit une prédiction de 50%) que de suivre ce modèle qui n'est pas très convaincant.

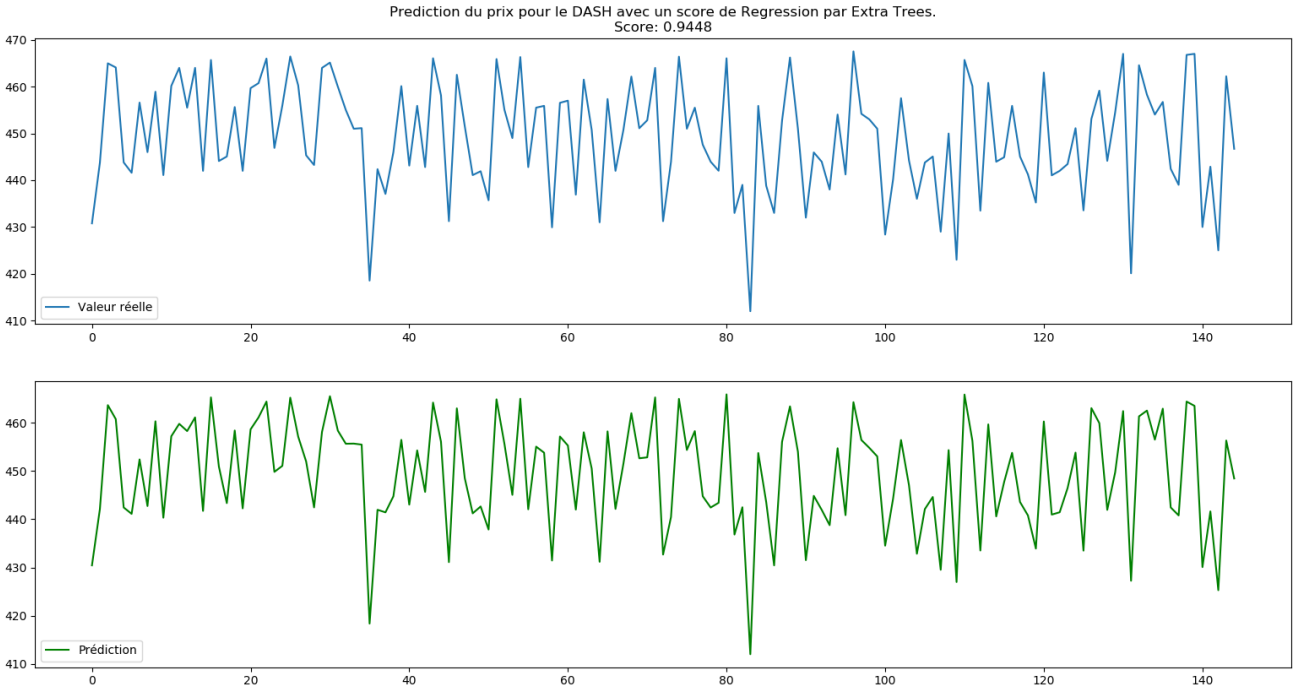
Passons maintenant à la régression et voyons voir les résultats obtenus :

Pour le Bitcoin :



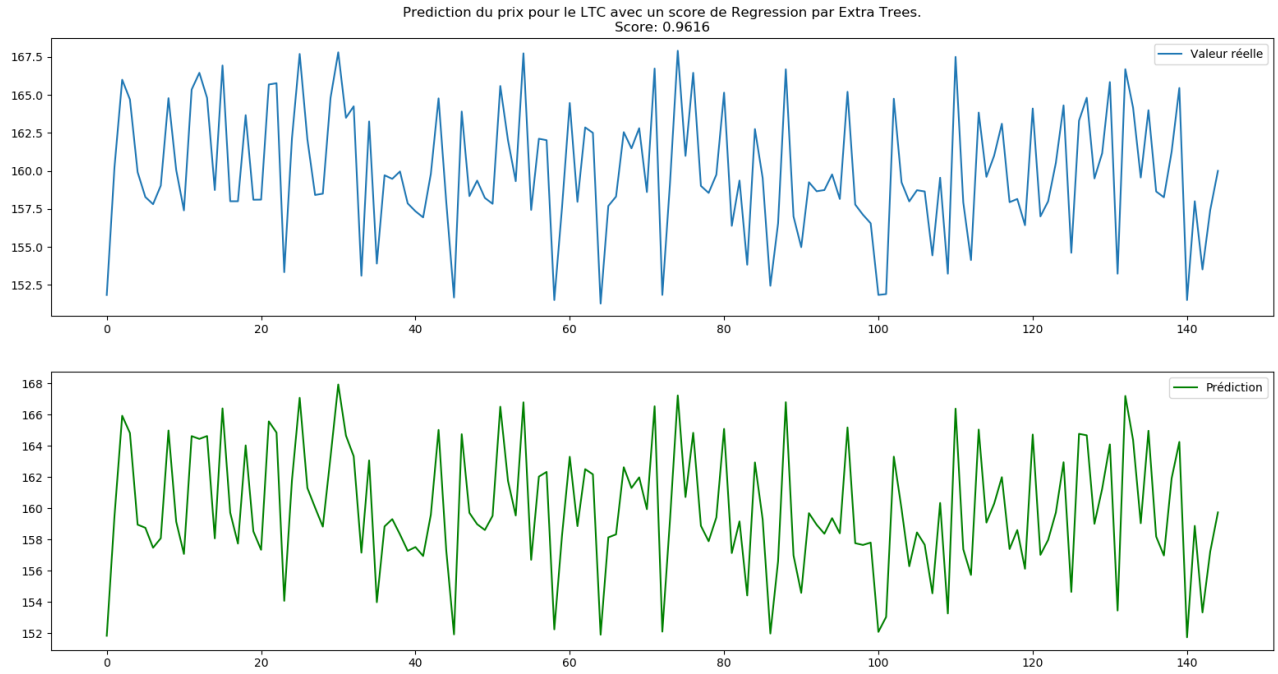
Nous avons un score plus qu'honorable de 0.8914 ce qui veut dire que notre modèle arrive à prédire les fluctuations du bitcoin avec 89% de justesse !

Pour le Dash :



Pour le Dash on obtient un score extraordinaire en régression de 0.94 ce qui veut dire que le modèle de régression en utilisant l'algorithme d'extra trees nous fournit une précision de 94% ! regardez par vous-même les deux courbes sont presque identiques.

Pour le Litecoin :



De même pour le Litecoin, une précision de 96% obtenue avec le même modèle que le Dash !

Conclusion

Pour conclure sur notre avancement, nous devons maintenant essayer de prédire le cours de ces cryptomonnaies en temps réelle. Nous avons trouvé les meilleurs modèles et tester chacun pour s'entrainer avec des valeurs passées. Le but maintenant est de les prédire dans le futur. Nous avons également pensé à affiner nos modèles en prenant en compte plus de données dans une prochaine version de l'application.

Grace a ce projet nous en avons appris énormément sur les enjeux de l'IA et comment implémenter celle-ci grâce à différents procédés. Nous avons également découvert un nouveau langage qui est le Python et qui offre de nombreuses possibilités.