POLITÉCNICA

INDUSTRIALES
ETSII | UPM

Quantitative study on the relationship between cryptocurrencies and social media flows

Guillaume Kunsch

Tutor :
José Manuel Mira McWilliams

Escuela Técnica Superior de Ingenieros Industriales
Universidad Politécnica de Madrid
Madrid, Spain

June 2021

Table of contents

Executive summary

The increasing technological development in recent years has brought a great revolution in finance and given birth to a new concept of money and economic transactions: cryptocurrencies. At the same time, the development of social networks and the spread of their use have significantly changed the way human societies communicate. This thesis focuses on this new financial scenario with the aim of finding a behavioral model between the price of cryptocurrencies and the flow of social networks.

A cryptocurrency is a digital or virtual currency created as an alternative medium of exchange to traditional money. Its operation is based on the blockchain, a public and decentralized digital ledger that records all transactions of a cryptocurrency between two users along with the date they were made. These transactions must be verified within the blockchain network before they are added. This verification is done by miners, that is, users who, with the help of computer equipment, compete to find the solution to a very complex mathematical problem before the others and seal the transactions generated in a block. As a reward for this service, they receive a certain share of the cryptocurrency they mine, as well as a commission paid by the user who initiates each transaction.

There is a wide variety of cryptocurrencies, each with its own characteristics and applications. The most accepted, known and valuable currency in the markets is Bitcoin, which was created in 2009. This currency is used to buy and sell other assets and as a store of value. Also, due to the growing interest in this new concept of money, it can be used as a means of payment in some cases, as if it were cash.

These cryptocurrencies are traded and bought by users through virtual platforms created by various companies that act as exchanges. The main exchanges are Binance, CoinBase, Kraken and Bittrex, all of which have the same goal: to generate maximum volume and therefore higher profits, as their revenue depends on exchange fees.

Historically, the emergence of cryptocurrencies is associated with crypto-anarchists who abhor anything centralized. They prefer peer-to-peer communication that doesn't go through a central server. The circulation of money is very centralized because money is stored in banks under the supervision of a central bank. Thus, banks have some power because they see most of the financial flows in circulation: this is what bothers crypto-anarchists.

Ultimately, cryptocurrencies are decentralized currencies that aim to bypass a central authority and thus transfer power to the collective force that controls transactions. Furthermore, the growing capabilities of tech giants have allowed humanity to communicate more and faster than ever before, without borders. Therefore, it makes sense to examine the relationship between social networks and cryptocurrencies. If cryptocurrencies are to be a currency by the people and for the people, can social media, which promote communication for all, have an impact on them? The objective will be to assess the relationship between the sentiment expressed on social networks and the positive or negative evolution of the cryptocurrency price. In the models we will use

positivity, volume of texts, objectivity, … as inputs and the output will be a binary categorical variable: a decrease or increase of the price of Bitcoin.

Method

This study focuses only on Bitcoin, as it is the most widely used cryptocurrency, both in terms of age and market share, which shows that it is accepted by the general public.

Similarly, social media posts will only be collected from Twitter. This is one of the platforms with the most active users, sending 500 million tweets per day. Twitter can therefore be a very rich source of data on people's feelings on almost any topic. With the ability to see when a tweet was published, one can also track how sentiment changes over time. This makes Twitter an excellent source for collecting textual data on a topic like cryptocurrencies.

Twitter data was collected using Tweepy. This is an open source framework written in Python that facilitates the collection of tweets from the Twitter API. It allows filtering based on hashtags or words and has proven to be an effective way to collect relevant data. The filter only includes the word « Bitcoin » - « BTC », the abbreviation of Bitcoin, is not included as it is more likely to be used for commercial purposes by robots.
The price of Bitcoin was found on the Binance exchange. It is a global cryptocurrency exchange platform that allows the exchange of more than 100 cryptocurrencies.

For sentiment analysis we used the VADER - Valence Aware Dictionary and sEntiment Reasoner library. It is a dictionary and rule-based sentiment analysis tool that is specifically adapted to the emotions expressed in social media. It is open source under the MIT license. VADER's emotion dictionary is sensitive to the polarity and intensity of emotions expressed in social media and is generally applicable to emotion analysis in other fields. It returns 4 new features for each tweet:
- *compound* score: this is the most useful metric to measure in a one-dimensional way the emotion of a given sentence. It is also useful to classify sentences as positive, neutral or negative.
- *pos*, *neu* and *neg* show the proportion of text in each category - so when using the float operation, all values must be equal to or close to 1. These are the most useful metrics for analyzing context and for showing how emotion is expressed in a particular sentence or embedded in rhetoric.

It is important to get rid of all automated tweets published by robots. This is a very difficult task and even the best platforms have difficulty identifying them. In this study, we tested two simple filters:
- URL-based: When we searched the dataset, it was clear that tweets sent by robots often contain a URL link - usually to a cryptocurrency trading platform. The idea is to get rid of all tweets that contain URLs.
- Based on the average number of tweets per user per day: the idea is to get rid of all tweets where this number is higher than 10: that is to say for user who posts more than 10 times a day on average. This figure is confirmed by the exploratory analysis of the dataset. The assumption behind this approach is that robots publish posts with high frequency. Bots can be programmed to update the bitcoin price every hour, for example. On the other hand, it is probably rare - but not impossible - for a real human to post more than ten times a day.

In addition, Twitter data was aggregated by different time periods - 1 day, 1 hour and 5 minutes - before being combined with Binance data into a single dataset.

Some feature engineering was conducted as well, so the final dataset is composed of the following features:

- d*ate_format* is the day - or datetime - from which the following features are related
- o*pen* is the price of Bitcoin at the opening of the trading session
- *close* is the price of Bitcoin at the end of the trading session
- *pos* is the average positive score of tweets during the session
- *neu* is the average neutral score of tweets during the session
- *neg* is the average negative score of tweets during the session
- *compound* is the average compound score of tweets during the session
- *variation_cat* indicates wether the price increases or decreases during the day
- *nb_of_tweets* indicates the number of tweets harvested through the session - it is not the real number of tweets on Bitcoin posted on the platform
- *variation* is the price variation along the time frame
- *variation_%* is price variation in % along the time frame
- *pos_variation%* is the variation in % of the average positive score of tweets
- *neu_variation%* is the variation in % of the average neutral score of tweets
- *neg_variation%* is the variation in % of the average negative score of tweets
- *compound_variation%* is the variation in % of the average compound score of tweets
- *nb_of_tweets_variation%* is the variation in % of the number of harvested tweets through the time frame

The variation of all scores is thought to be of interest because maybe more than the absolute sentiment measure it may be its variation that drives changes. However the variation computed is not the real one between 2 truly consecutive time frames but rather the one between 2 consecutive time frames in the dataset - that is one of the consequences that was faced by only resorting to the free version of Twitter API.

Let's bear in mind that such a dataset is obtained for each time frame and each way of filtering - URL or using the average number of tweets per user per day -, that is to say 3x2 = 6 datasets. Besides for each dataset we will try 3 difference algorithms : a model-based with Logistic Regression, an instance-based with k-Nearest Neighbors and an ensemble model with Random Forest.

Besides, for each combination of filter, time frame and algorithm a mix of 3 different sets of features will be tried:

- A model trained with all features: *pos, neg, compound, nb_of_tweets, pos_variation%, neg_variation%, compound_variation%, nb_of_tweets_variation%*
- A model trained only with absolute features: *pos, neg, compound, nb_of_tweets*
- A model trained only with variation features: *pos_variation%, neg_variation%, compound_variation%, nb_of_tweets_variation%*

In all models the output is the same : the increase or decrease of Bitcoin price. Globally, 2 filters x 3 time frames x 3 algorithms x 3 sets of features = 54 models will be tried in this study.

Results

54 models were tested, out of them 2 can be distinguished from the others for their results.

```
Accuracy: 65.00%
Precision: 76.19%
Recall: 64.00%
f1 score: 69.57%

array([[10,  5],
       [ 9, 16]])
```

```
Accuracy: 80.00%
Precision: 77.78%
Recall: 87.50%
f1 score: 82.35%

array([[5, 2],
       [1, 7]])
```

On the left column the kNN - 1 hour - trained on only variation features - filter on the average number of tweets per day per user, and on the right the kNN - 1day - trained on only absolute features - filter on URL. However a deeper analysis is required to replicate results and achieve higher outcomes and, if so, put models into production.
To give some context to these results they can be compared to the study by Lamon et al. (2015). They found that logistic regression was the most efficient way to classify tweets with an accuracy of 43.9% for price increases and 61.9% for price decreases.

Broadly speaking, considering all the models tested, it appears social media sentiments are promising indicators to relate to the evolution of the price of Bitcoin. However, we should not forget that this study is dealing with human behavior which is intrinsically random, unexpected (even if not entirely) and not always rational. As such it seems unlikely to be able to achieve results as accurate as in hard sciences based on natural laws. Nonetheless an accuracy of 75% - on a balanced and numerous dataset- could be achieved: it would show the usefulness of Twitter sentiments.

In addition to the models' results themselves, some other key takeaways can be derived from his study.

In general, it seems the 1 hour time frame is the best suited to aggregate data and derive relationships between sentiments on Twitter and the evolution of Bitcoin price. 1 day is too large considering all the events that could occur and impact the price, while 5 min is too short. Nonetheless, better results could be achieved on other time frames.

There is no set of features that performs best every time. We've seen in some cases a mix of all features perform better, while in others it was only the absolute ones or the variation ones. Besides it has appeared that the volume of tweets posted could be a more important input than sentiments. An hypothesis to explain it would be that sentiment analysis remains a hard task to perform, especially in social contexts where one can mean various things with one sentence and the massive amounts of bots on social media. More simply, an increase in the volume of tweets would be linked to an increase in the interest granted to Bitcoin - maybe because of other events that Twitter would only react to, and not influence in any way - such interest being translated as an increase in price.

Similarly there is no algorithm that works better every time, although kNN seems to be the one achieving the best results globally. Again, a deeper research would be needed. As a non-parametric model, that can detect patterns from data without any hypothesis

on the distribution it may be more equipped than Logistic Regression to achieve such a task.

The performances of the two filters seem to be very similar, even if the one on the average number of tweets per user per day may perform slightly better. A reason for this could be that the hypotheses behind it are more robust.

For further research some paths that weren't taken in this study could be followed.

Firstly, in addition to Twitter there are plenty of other sources, to cite a few : Reddit, Quora, Google Trends, dedicated forums that could be scraped, Telegram channels, … Maybe more than all tweets, only the tweets posted by a small group of influential people should be considered. Indeed, the frequent impacts in 2021 of some highly popular people such as Elon Musk put under limelight the tremendous power of opinion makers, maybe up to the point of market manipulation.
Secondly, sentiment analysis could be performed using different libraries than VADER. VADER is mainly designed for social media but not for finance or Bitcoin. A specific lexicon designed for that purpose could be better suited to perform the task.
Thirdly, a more complex approach toward robot filtering involving clustering could be undertaken.
Finally, we did not take into account the effects of future time horizons. For example, one hypothesis could be that Twitter sentiment influences the price of Bitcoin but with some lag of 2 hours, in which case we should not relate the price movement to the sentiment of the same time period but rather consider a lag - the best to be determined.

KEY WORDS:

Cryptocurrency, Blockchain, Bitcoin, Sentiment Analysis, Machine Learning, Twitter

UNESCO CODES:

120304, 120323, 120903, 530406

I. Cryptocurrencies

1. A quick history of currency

Before diving into what is cryptocurrency, some context on traditional currency is needed. This part doesn't aim to be exhaustive, but rather to give the most important trends of what lead to our current currency system.

The need for a currency appeared by the drawbacks of barter. In this kind of economy, people don't rely on an intermediate to exchange values but trade directly their goods. If you want to buy a cow you could convince a seller that his value is tantamount to the 2 horses you own and then conclude the deal. The main issue in this kind of economy is to find the adequate value between each item. Besides, these values must be integers.

To get rid of these issues, currencies were developed. Theirs first stains go back to the 2nd millennium BC. It wasn't coin as it can be used today, but more primitive objects lasting enough : it could be mere shells, rocks or even teeth. Gold being a scarce and solid metal, it became precious and took value : it ended up being the most used support for trading, mostly on coins. These coins were much more practical to exchange, but they had a defect : they were made of a precious metal, such as if you lost the coin the gold was lost as well.

To bypass this inconvenient characteristic a new system was developed : not based on intrinsic value, but on trust. Explaining the mechanisms for building such a system is outside the scope of this paper, it should only be noted that this system allows trade to be carried out by low value items such as mere papers - whose value only depends on what's written on it.

Thanks to the tremendous progress of science and technology, the support for exchange has been be totally digitalized. Now, western people are used to pay for something using a credit card, rather than coins or notes. However, our currency system is still based on trust. When you see marked 1,000€ on your online bank account it's just a line in a database in the bank's computer system. The money shown on a bank account is what the bank owes you. The 1,000€ displayed on your online account is therefore a €1,000 debt owed to you by the bank, but that doesn't mean it could pay it back to you right now.

Only the coins and banknotes in your pocket have value today - fiat money - although if the institution which guarantees the banknotes collapsed, the banknotes wouldn't have any value. On the other hand, what's displayed on a bank account is only a promise from the bank to you that it has this money - scriptural money. [1]

2. Where do cryptocurrencies come from ?

The early history of Bitcoin is linked to the anarchist current, and in particular to a technological offshoot of it, crypto anarchism. Today anarchy is often used in a negative way, but philosophically it is a society without a system of power such as authoritarian government, exploitative economy or dominant religion. It is the situation of a social environment where there are no power relations, no leaders, no central authority; a society where each person, group, community or environment is autonomous in his/her internal and external relations. There is always an organization, an order, a political power or even several, but not a single domination of a coercive nature [2]. Therefore it's a philosophy whose goal is to create a self-organized society, without hierarchy, and especially without any state.

Applying their ideal on computer science, anarchists wished to keep their privacy and cipher all their data, preventing any authorities to access them. Even if the Snowden Reveal took place after the creation of the first cryptocurrency, it could well explain their motives.

Edward Snowden's revelations began with a massive amounts of documents transmitted by former CIA agent and NSA consultant Edward Snowden to two journalists, Glenn Greenwald and Laura Poitras, and progressively made public from 2013 through several articles. They refer to the worldwide surveillance of the internet, but also mobile phones and other means of communication, mainly by the NSA.

The revelations have and continue to contribute bringing to the attention of the general public the extent of the intelligence collected by the American and British secret services. In particular, they have brought to light the PRISM program for gathering information online, the GENIE program for spying on computer equipment abroad, the spying on intercontinental telecommunications submarine cables and on international institutions such as the European Council in Brussels or the United Nations headquarters, as well as many practices within the agency to achieve its aims. [3]

The crypto anarchists abhor anything that is centralized. They prefer to use peer-to-peer communications, where computers communicate with each other without going through a central server. Money also moves very centrally, as it is held in banks under the authority of a central bank. The banks therefore have a certain power, since they see most of the financial flows circulate : this is what bothers crypto-anarchists.

3. Blockchain principles

In 2009, Satoshi Nakamoto published a pdf document explaining how he created a new currency - the Bitcoin - based on a new system: the Blockchain [4]. It's a technology for storing and transmitting information without a control governing body. Technically, it is a distributed database in which the information sent by users and the internal links within the database are checked and grouped at regular time intervals into blocks, thus forming a chain. The whole is secured by cryptography. Today the whole blockchain is

made up of approximately 100 Go. For detailed explanations, some additional definitions are provided.

Transactions:

Exchanges between users, stored in the blocks of the blockchains.

Blocs:

The various transactions registered are grouped into blocks. After recording recent transactions, a new block is generated and all transactions will be validated by the miners, who will analyze the complete history of the block chain. If the block is valid, it is time-stamped and added to the blockchain. The transactions it contains are then visible throughout the network. Once added to the chain, a block can no longer be - theoretically - modified or deleted, which guarantees the authenticity and security of the network. The system is built in such a way that, if one wanted to hack the blockchain, one would have to control more than 50% of the computing power of all the computers that mine.

Hash functions:

A hash function is a special function that, based on input data, calculates a digital fingerprint to quickly identify the initial data, in the same way that a signature is used to identify someone. The hash functions have a particularity: they only work in one direction. From the result of hashing some data you can't go back the initial data, but you could ensure the data has not been modified by hashing the data again: if the 2 outcomes are different, then the data have been corrupted .

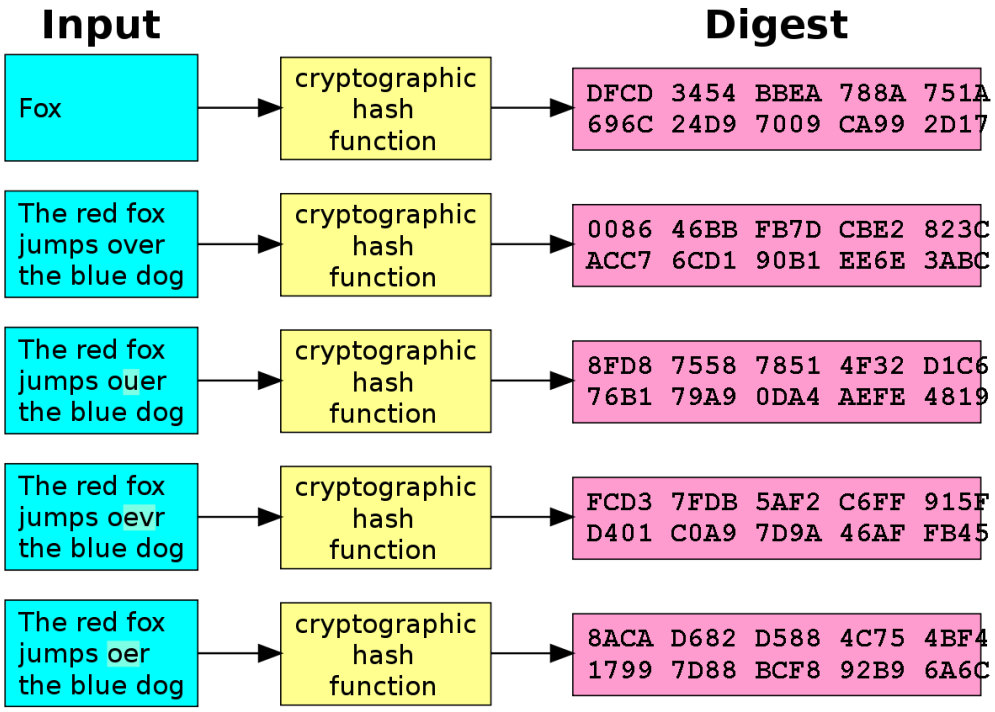

Figure 1. Explanation of hash function

Miners:

Miners are the people who perform mining of the blockchain. Bitcoin mining consists in confirming Bitcoin transactions and recording them on a distributed ledger in exchange of Bitcoins. It's the most important procedure in the entire Bitcoin network, as it secures the system, ensures that everyone is acting correctly and introduces new Bitcoins.

Mining computers try to solve a mathematical problem : specifically, they try to find the number that, when hashed, gives a number that starts with a long series of zeros. As explained above, hash functions only work in one way, so computers have to try many possibilities until they find a suitable answer. This is the proof of work.

This process is extremely time consuming. But when all the computers of the networks are trying to solve it, they can find a solution in 10 minutes. As a matter of fact , the difficulty is automatically adjusted according to the number of computers hashing, so that a new block is generated on average every 10 minutes. The computer which found the solution is rewarded in Bitcoins, *today* around 12,5 BTC - around 7,000 €.

But this process is also extremely resources consuming. According to Digiconomist, for Bitcoin only it would be 71.1 TWh/year on 1 July 2018, i.e. the energy produced for one year by six 1,300 MW nuclear reactors operating at full capacity or Chile's annual electricity consumption or 0.32% of world electricity consumption [9]. The electricity consumption of all the encryption systems would be double that of Bitcoin.

The difficulty of mining has led miners to group together in cooperatives - mining pools - to combine their computing resources and build new blocks more quickly. The remuneration corresponding to the constitution of each block is then distributed among the members, after deduction of fees, which smoothes their income and makes it less uncertain. By 2016, around ten of these cooperatives provided 95% of the blocks. Most of them are located in China (which accounts for most of the energy on the Bitcoin network), but also in the Czech Republic or Georgia.

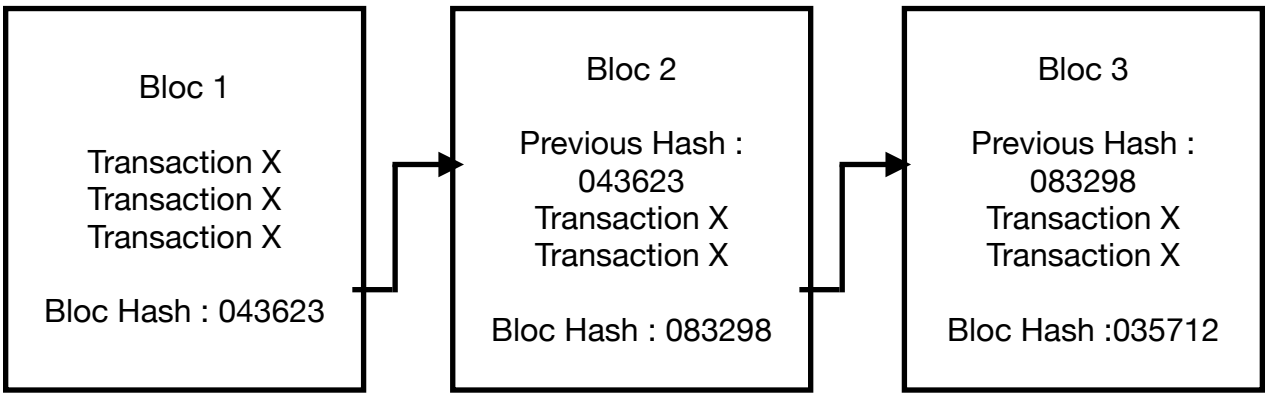

Figure 2. Explanation of the blockchain

4. Characteristics of cryptocurrencies

In addition of the traditional characteristics of currencies, such as a medium of exchange, divisibility or value reserve, cryptocurrencies have their own characteristics that make them different from other traditional currencies:

- they are decentralized: everyone on the network has a copy of the distributed ledger - the blockchain.

- they are borderless: in theory, you could pay with a crypto currency from anywhere in the world.

- they are trust less: this means that the network as a whole verifies and guarantees the correctness of the data without the need for a trusted source - normally this role is played by banks in any monetary transaction. As seen, the blockchain is conceived in a way that leaves no possibility of fraud in the system: the constant surveillance of the blockchain by all miners ensures the smooth functioning of the system. Miners are encouraged to monitor transactions through the proof-of-work - PoW - consensus algorithm.

- they are anonymous: even if the blockchain is public, the personal identity never appears on it. Instead each transaction occurs between 2 accounts, and you can't trace back an account to a natural or legal person. To further ensure their anonymity, most people who use cryptocurrencies use numerous accounts.

5. Advantages and drawbacks

Cryptocurrencies have different characteristics from traditional currencies. From these different characteristics, their own advantages and disadvantages are derived.

As for the positive aspects, it is important to distinguish:

- The near perfect security of the distributed system

- The independent governance production of crypto money. The state does not determine the value of the currency through the issuance of money. They have a controlled inflation, as the amount of coins to be created is determined in advance.

- They offer interesting solutions to issues faced by companies : features as multiple signature authorization and accounting transparency would allow to automate the workflow. Multi-signature means that several people have to sign a payment, which offers more security. And the very nature of a blockchain - where all transactions are public- improves company transparency.

- Their use is voluntary, as opposed to traditional coins which are imposed according to the area you are trading in.

- The user can choose the commission he/she pays. The commission depends on the size of the transaction and the network congestion, not the amount. In other words, if the user wants his transaction to be carried out as soon as possible, he will have to pay a high commission so that the miner can choose it from the first ones.

However, cryptocurrencies also have some disadvantages that have caused their inclusion in society to slow down:

- The regulation is unclear and different in each country. Their decentralization and lack of control and supervision by governments and banks, while being the main attraction of cryptocurrencies, can even create legal problems, such as non-payment of the corresponding taxes or money laundering.

- Securing Bitcoins requires basic knowledge of cybersecurity. While the network is virtually inaccessible, organizations and individual users are.

- Cryptocurrencies are usually very volatile. The value of crypto coins can fall or rise suddenly and abruptly, triggering large gains or losses for their users.

- An Internet access is needed for most cryptocurrencies to perform a transaction

- Cryptocurrencies' fundamental ideology runs counter to the most powerful institutions, governments, politics, banks, regulators and censorship, and is likely to meet a great deal of resistance before these actors tolerate or approve it.

6. Cryptocurrency landscape

Cryptocurrency is a singular term which encapsulates various currencies behind it. The most famous and widespread of all, as well as the first, is the Bitcoin. However its popularity - and volatility - opened a new era where plenty of new cryptocurrencies now exist: each one has its own algorithm and addresses a particular problem it aims to solve. It's worth emphasizing the most known :

- Ethereum: it is historically the second most popular cryptocurrency. Ethereum is actually the name of the blockchain platform and Ether is the name of the cryptocurrency. Ethereum is the blockchain platform for smart contracts. While Bitcoin is intended as an alternative to traditional fiat currencies, the purpose of Ether - besides being traded as an asset - is to pay for use of the Ethereum platform. It's known as a utility cryptocurrency.

- Ripple: it is another 'utility' coin. Its blockchain platform is set up to facilitate cross-border transfers of fiat currency more efficiently. Closely connected to and supported by a number of banks from its beginning, Ripple XRP is often regarded as the establishment cryptocurrency. The number of transfer services using Ripple's

platform has gradually grown over the years - with companies such as UBS, Santander or Crédit Agricole - and there is a genuine possibility that it will become part of the traditional financial system.

- Litecoin: it is another potential fiat alternative and a prominent rival for Bitcoin. Its creators hope Litecoin will eventually be used to pay for everyday goods and services. Litecoin has positioned itself as a more practical and technologically superior alternative to Bitcoin. Litecoin transactions can be confirmed by the P2P network significantly quicker than Bitcoin transactions. In theory, this could make Litecoin more attractive for merchants, but with real-life cryptocurrencies transactions still hugely limited, Bitcoin's more established brand keeps it well out in front as the fiat alternative cryptocurrency of choice.

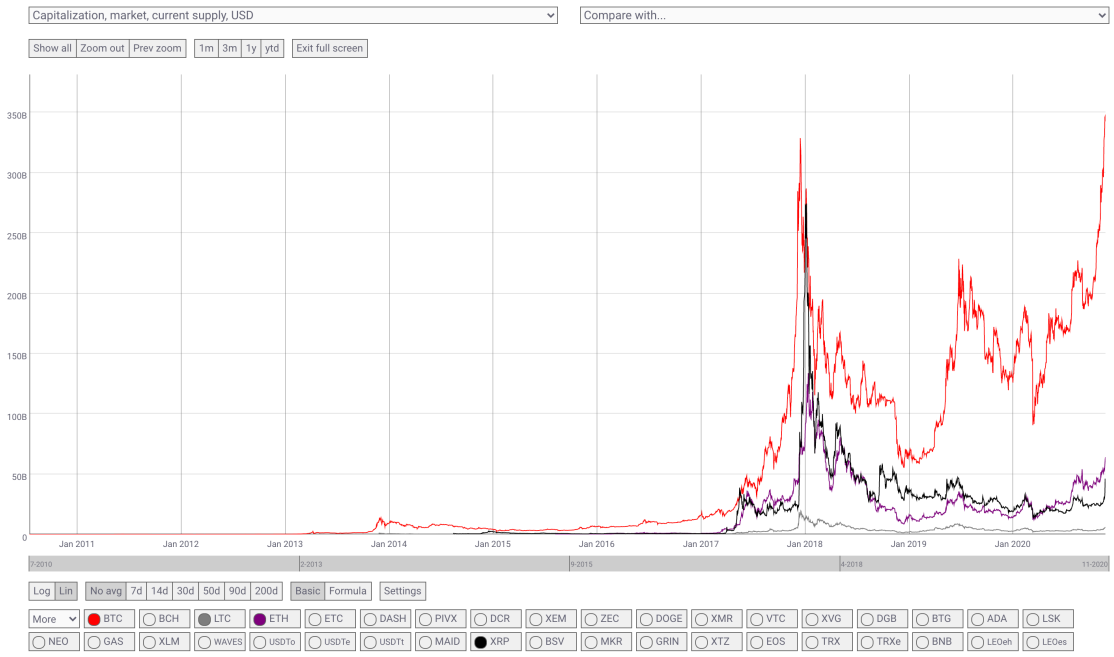

Figure 3. Capitalization of BTC, LTC, ETH and XRP as a function of time

The number of people who actively resort to crypto currency is growing exponentially. According to the Global Cryptocurrency Benchmarking Study the current number of unique active users of cryptocurrency wallets is estimated to be between 2.9 million and 5.8 million, and between 5.8 million and 11.5 million wallets are estimated to be currently active in 2017. As for cryptocurrencies applications, they can be divided into 3 usages.

Cryptocurrencies are used, above all, as a means of investment. The number of investments in these virtual currencies is increasing, as they are considered by a large number of people to be the best investment opportunity at present.

Crypto assets are also accepted as a form of payment both by some online merchants and by small local shops, restaurants and bars. They can be used to pay for hotels, flights, jewelry, applications... Even Apple has authorized at least ten different

currencies as a payment method in its App Store. It should be mentioned that the most accepted crypto-currency is Bitcoin, the use of the rest being not so widespread. But users can always exchange their cryptocurrencies for Bitcoin.

A last possible use of these virtual currencies is mining, seen as an investment as well as trade. The more computing power a miner has, the greater the chance of solving cryptographic problems, and therefore the greater the chance of receiving a reward and the corresponding transaction fee.

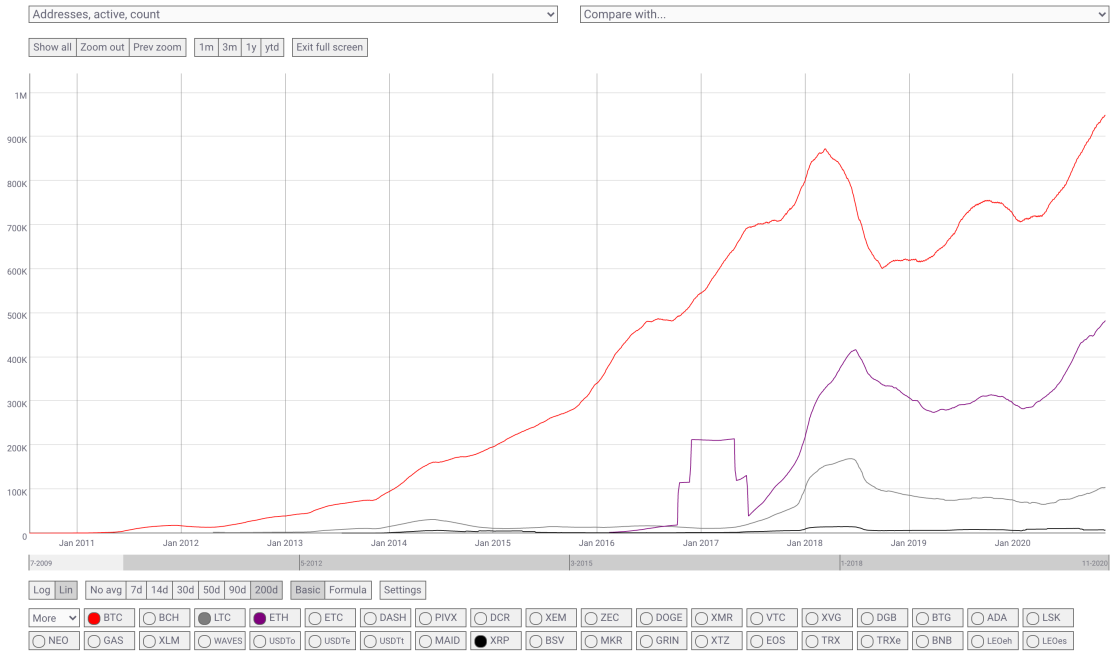

Figure 4. Number of actives user for BTC, LTC, ETH and XRP as a function of time

II. Social media

With the emergence of Internet and later the World Wide Web on top of it in the 90s, a pressing attention toward digital habits and how they shape the world has appeared. In recent times, social media has impacted many aspects of human communication: thanks to the growth of tools such as Facebook, Twitter, or Instagram people have been able to share thoughts and opinions as never before.

1. Landscape

Today various social platforms exist, each with its own features and specificity.

The most famous of all is Facebook. It is an American online social media and networking service based in Menlo Park, California. Its website was launched on 4 February 2004 by Mark Zuckerberg. Once registered, users can create a personalized profile indicating their name, occupation, schools attended, etc. Users can add other users as "friends", exchange messages, post status updates, share photos, videos and links, use various software applications - apps - and be notified of other users' activity. Facebook has over 2.7 billion active users per month as of the second quarter of 2020. [16]

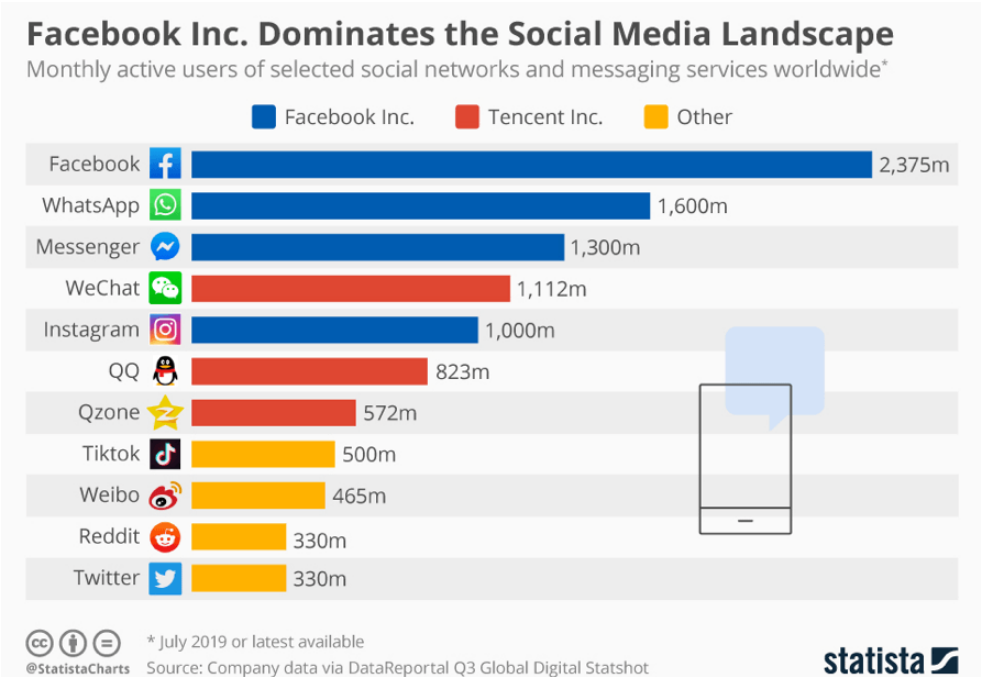

Figure 5. Social platforms ranked by number of users

Another giant of the market is Twitter, it is a micro-blogging social network managed by the company Twitter Inc. It allows a user to send short messages, called tweets, over the internet, by instant messaging or SMS for free. These messages are limited to 280 characters - previously 140 until November 2017. Twitter was created on 21 March 2006 by Jack Dorsey, Evan Williams, Biz Stone and Noah Glass. The online service quickly became popular. As of 2018, Twitter had more than 321 million monthly active users, 500 million tweets sent per day and is available in more than forty languages. [17]

Instagram is a photo and video sharing service founded and launched in October 2010 by American Kevin Systrom and Brazilian Michel Mike Krieger. Instagram claims more than one billion users worldwide, 75% of whom are outside the United States, according to official figures provided in June 2018. [20]

It's also worth noting that whilst the major platforms (Facebook, Instagram, Twitter) still dominate, there has been a rise in alternative platforms acclaimed by the new generation. It concerns YouTube for video sharing, Quora and Reddit for Q&A, Snapchat for temporary content sharing, and others. According to the Global Digital Report 2019, the number of social media user in 2109 was 3.5 billion, up to 9% from the previous year. [19]

2. Controversies

Despite the exponential growth of these kind of platforms they have been the target of many and various critiques.

The main one involves the lack of a safety net to limit the spread of fake news. Indeed, after the US 2016 elections Facebook and Twitter have been accused of having played an important role in the election of Donald Trump.
Linked to the main critique, is the one concerning the use of privacy data collected by social media platforms. They gather personal and sensitive data on each and everyone of their user: age, gender, political opinion, wealth, religion are among others items that can be directly collected or guessed by the social giants based on what each user posts, shares or likes. These highly sensitive data has been used by Cambridge Analytica in the 2016 election, to target undecided voters in the US and influence them in voting for Trump by propagating fake news or ensuring personalized social advertising [21]. Similarly, the same company and technique is susceptible to have played a major role in the Brexit referendum. [22]

As a whole it's the business model of social networks that is based on data : their revenues come mainly from social advertising. They sell to brands a privileged access to well-segmented users, so that their ads can be shown to people most likely to buy their products [23]. A study from Stanford University has shown that a computer knows more about a person's personality than their friends or flatmates from an analysis of 70 "likes", and more than their family from 150 likes. From 300 likes it can outperform one's spouse [24]. But social media advertising has other advantages over traditional advertisings. First of all, online advertising compared to traditional advertising allows the user to buy immediately via links - by clicking on banners. Another significant

advantage of online advertising is that it also allows the effect of online campaigns to be measured in real time, by monitoring click rates on a daily basis or even in real time. Finally, the speed of diffusion of online advertising is another advantage compared to other media.

In order to improve the service they sell to brands, social platforms need to make sure each user will be active and spend time on it. That's why a strategy used by social networks is to create habits: some features such as an endless scrolling, or pull-to-refresh are designed to stimulate the areas of our brain which produce dopamine, the molecule responsible for the feeling of pleasure, thus activating the reward system. [25]

Finally some concerns are expressed about consequences on user's mental health. In May 2017, a survey asking people to rate social media platforms depending on anxiety, depression, loneliness, bullying and body image, concluded that Instagram was the "worst for young mental health » [26]. While this same survey noticed its positive effects, including self-expression, self-identity, and community building, some have suggested it may contribute to digital dependence. Throughout 2019, Instagram began to test the hiding of like counts for posts made by its users.

III. Objectives of this work

It has been observed that cryptocurrencies are decentralized currencies which aim to bypass a central authority that controls transactions and imposes its own currency policy: it would be the FED for US dollars or ECB for Euro. By doing so, cryptocurrencies transfer the power to the collective force which monitors transactions. Additionally, it has been analyzed how the growing capabilities of tech giants have allowed mankind to communicate more and faster than ever without any borders.
Therefore it makes sense to study the link between social media and cryptocurrencies. If cryptocurrencies aim to be a money by the people for the people, can the social media which fosters communication for everyone have any influence on it ?

This study draws on ideas from a wide range of research and topics. Behavioral economists such as Daniel Kahneman and Amos Tversky have established that decisions, even those with financial consequences, are influenced by emotion and not just by value [27]. The ideas of these researchers open up the possibility of finding benefits using tools such as sentiment analysis, because they indicate that the demand for a good, and therefore its price, can be influenced by more than its economic fundamentals. Similarly, Paul Tetlock found that media pessimism about the stock market had an impact on trading volumes. [28]

More specifically on social media, Pieter de Jong et al. (2017) analyzed stock price and tweet data from 30 stocks in the DOW Jones Industrial Average and found that 87% of stock returns were influenced by tweets. However, they also examined whether the reverse was true, i.e. stock prices influenced tweets, and found little evidence of their influence [29].

Finally, some papers performed sentiment analysis on cryptocurrencies. The objective of sentiment analysis is to analyze a large amount of data in order to deduce the different feelings expressed in it. The extracted feelings can then be used to produce statistics on the general feelings of a community. Lamon et al. (2015) used the sentiment of headlines and tweets to predict the evolution of Bitcoin, Litecoin and Ethereum. The study showed that logistic regression was the most efficient way to classify these tweets and that they were able to correctly predict 43.9% of price increases and 61.9% of price decreases [30]. On the contrary, Jethin et al. (2018) found no evidence between Twitter sentiments and Bitcoin price. [31]

The goal of this paper will be to add its own contributions to the debate. It will analyze the relationship between social media flows and cryptocurrencies prices. The goal will be to predict a decrease or an increase of cryptocurrencies prices based on the data collected on social platforms.
The inputs will consist of the characteristics extracted from the tweets (positivity, volume, etc.) and the output will be will be a binary categorical variable : the decrease or increase of the price of Bitcoin.

Cryptocurrencies included were limited to Bitcoin considering it is the most established cryptocurrency both in age and in market share, reflecting its acceptance in the public's eye.

Similarly, Twitter data is the only one that will be focused on. It is one of the platforms with the most active users and 500 million tweets are sent each day. The result is that Twitter can be a very rich source of data on how people feel about nearly any given topic. With the ability to know when a tweet was posted it is also possible to tell how those feelings change over time. This makes Twitter an excellent resource to collect text data on a topic such as cryptocurrencies.

IV. Theoretical foundations

1. General considerations on Machine Learning

According to Arthur Samuel, an American pioneer in the field of Artificial Intelligence, Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed. It is generally ideal for problems for which existing solutions require a lot of hand-tuning, when there is a fluctuating environment, or for getting insights about complex problems and large amounts of data.

There are many different types of Machine Learning, they can be classified in various categories:
- Wether or not they are trained with human supervision - supervised, unsupervised, semisupervised and reinforcement learning. In this paper only a supervised approach will be used
- Whether or not they can learn incrementally on the fly - online versus batch learning.
- Whether they work by simply comparing new data points to known data points, or instead detect patterns in the training data and build a predictive model - instance-based versus model-based learning.

In a ML project data is gathered in a training set, that is then fed to a learning algorithm. If the algorithm is model-based it tunes some parameters to fit the model to the training set - to make good predictions on the training set itself - and then if all goes well it will be able to make good predictions on new cases as well. If the algorithm is instance-based, it just learns the examples by heart and uses a similarity measure to generalize to new instances.
The system will not perform well if the training set is too small, or if the data is not representative, noisy, or polluted with irrelevant features - garbage in, garbage out. Lastly, the model needs to be neither too simple - in which case it will underfit - nor too complex - in which case it will overfit.

When a trained a model is obtained, one doesn't want to just hope it generalizes to new cases. One wants to evaluate it. To do that the dataset is split in two sets: the training set and the test set. As these names imply, one trains a model using the training set, and tests it using the test set. The error rate on new cases is called the generalization error - or out-of-sample error - and by evaluating a model on the test set, one obtains an estimation of this error. This value tells you how well the model will perform on instances it has never seen before.

When one wants to fine tune the model one uses a 3rd set: the validation set. Multiple models are trained with various hyperparameters using the training set, the model and hyperparameters that perform best on the validation set are selected, and when one is happy with the model a single final test against the test set to get an estimate of the generalization error is run.
To avoid wasting too much training data in validation sets, a common technique is to use cross-validation: the training set is split into complementary subsets, and each model is trained against a different combination of these subsets and validated against

the remaining ones. Once the model type and hyperparameters have been selected, a final model is trained using these hyperparameters on the full training set, and the generalized error is measured on the test set.

In a famous 1996 paper, David Wolpert proved that if one makes absolutely no assumption about the data, then there is no reason to prefer one model over any other. This is called the No Free Lunch - NFL - theorem. For some datasets the best model is a linear model, while for other datasets it is a SVM . There is no model that is a priori guaranteed to work better - hence the name of the theorem. The only way to know for sure which model is best is to evaluate them all. Since this is not possible, in practice some reasonable assumptions about the data are made and only a few reasonable models are evaluated. For example, for simple tasks linear models with various levels of regularization may be evaluated, and for a complex problem various neural networks may be benchmarked.

2. Classification and metrics

In this paper classification techniques only have been used. To compare models with each other various metrics are required to evaluate the performance of each model. However evaluating a classifier is often significantly harder than evaluating a regressor, because there are many performance measures available.

The first one is accuracy, the ratio of correct predictions. It is generally not the preferred performance measure for classifiers, especially when you are dealing with skewed datasets - when some classes are much more frequent than others.

The confusion matrix is much more complete. Each row in a confusion matrix represents an actual class, while each column represents a predicted class.

$$y \begin{pmatrix} TN & FP \\ FN & TP \end{pmatrix} \hat{y}$$

$\hat{y}$ is the predictor class and y is the actual class.
TP is the number of true positives, and FP is the number of false positives.
TN is the number of true negatives, and FN is the number of false negatives.

The confusion matrix provides a lot of information, but sometimes you may prefer a more concise metric. An interesting one to look at is the accuracy of the positive predictions; this is called the precision of the classifier.

$$\text{Precision} = TP/(TP + FP)$$

A trivial way to have perfect precision is to make one single positive prediction and ensure it is correct (precision = 1/1 = 100%). This would not be very useful since the classifier would ignore all but one positive instance. So precision is typically used along with another metric named recall: it is the ratio of positive instances that are correctly detected by the classifier.

Recall = TP/(TP+FN)

It is often convenient to combine precision and recall into a single metric called the f1 score, in particular if you need a simple way to compare two classifiers. The f1 score is the harmonic mean of precision and recall. Whereas the regular mean treats all values equally, the harmonic mean gives much more weight to low values. As a result, the classifier will only get a high f1 score if both recall and precision are high.

3. Algorithms used and how they work

A. Logistic Regression

Logistic regression, despite its name, is a linear model for classification rather than regression that only allows binary classification. It is also known in the literature as maximum-entropy classification or the log-linear classifier. In this model, the probabilities describing the possible outcomes of a single trial are modeled using a logistic function - S-curved that outputs a number between 0 and 1.

$$\hat{p} = \sigma(\theta^T \cdot X)$$

where

$$\sigma(t) = \frac{1}{1 + \exp(-t)}$$

θ is the parameter of the model, X is the input and $\hat{p}$ the probability outcome.

Once the probability is estimated it can be used to make predictions using a threshold optimized for the task. For instance if $\hat{p} > 0{,}5$ then the algorithms can make the prediction $\hat{y} = 1$, $\hat{y} = 0$ otherwise.

The objective of training is to set the parameter vector θ so that the model estimates high probabilities for positive instances (y = 1) and low probabilities for negative instances (y = 0). To this end the cost function c is introduced :

$$c(\theta) = \begin{cases} -\log(\hat{p}) & \text{if } y = 1 \\ -\log(1 - \hat{p}) & \text{if } y = 0 \end{cases}$$

This cost function makes sense because $-\log(t)$ grows very large when t approaches 0, so the cost will be large if the model estimates a probability close to 0 for a positive instance, and it will also be very large if the model estimates a probability close to 1 for a negative instance. On the other hand, $-\log(t)$ is close to 0 when t is close to 1, so the cost will be close to 0 if the estimated probability is close to 0 for a negative instance or close to 1 for a positive instance, which is precisely what is wanted.

The cost function over the whole training set is simply the average cost over all training instances. It can be written in a single expression, called the log-loss:

$$J(\theta) = -\frac{1}{m}\sum_{i=0}^{n}[y^{(i)}\log(\hat{p}^{(i)}) + (1-y^{(i)})(1-\log(\hat{p}^{(i)})]$$

There is no known closed-form equation to compute the value of θ that minimizes this cost function - there is no equivalent of the normal equation in classic linear regression. But this cost function is convex, so gradient descent - or any other optimization algorithm - is guaranteed to find the global minimum if the learning rate is not too large and you wait long enough.

As a model based algorithm, logistic regression relies on various mathematical assumptions. First, it requires the observations to be independent of each other. In other words, the observations should not come from repeated measurements or matched data. Second, logistic regression requires there to be little or no multicollinearity among the independent variables. This means that the independent variables should not be too highly correlated with each other. Third, logistic regression assumes linearity of independent variables and log odds.

B. k-Nearest Neighbors

Neighbors-based classification is a type of instance-based learning or non-generalizing learning: it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the nearest neighbors of each point: a query point is assigned the data class which has the most representatives within the nearest neighbors of the point.

The k-neighbors classification in is the most commonly used technique. The optimal choice of the value k is highly data-dependent: in general a larger k suppresses the effects of noise, but makes the classification boundaries less distinct.

The basic nearest neighbors classification uses uniform weights: that is, the value assigned to a query point is computed from a simple majority vote of the nearest neighbors. Under some circumstances, it is better to weight the neighbors such that nearer neighbors contribute more to the fit.

C. Random Forest

To define what a Random Forest consists in, it is first mandatory to understand Decision Trees.

Decision Trees are versatile Machine Learning algorithms that can perform both classification and regression tasks, and even multi output tasks. They are very powerful algorithms capable of fitting complex datasets. One of the many advantages of Decision

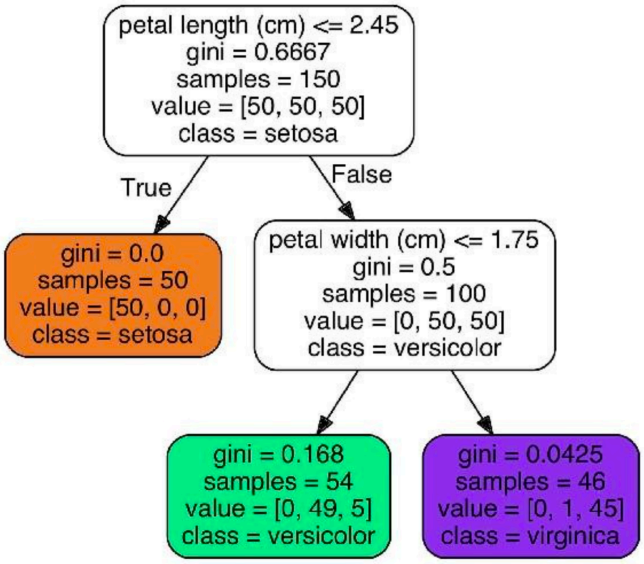

Figure 6. Decision tree from the iris dataset

Trees is that they require very little data processing. In particular, they don't require feature scaling or centering at all.

To make a prediction the algorithm start from the root node. This node asks for a specific criteria - here whether the flower's petal length is smaller than 2.45 cm. If it is so, then one moves down to the root's left child node, right child in the opposite case. Node's samples attribute counts how many training instances it applies to, value counts the different class it is trained on and class is the predict class for this node. Finally, a node's Gini attribute measures its impurity: a node is pure ($gini = 0$) if all training instances it applies to belong to the same class.

$$G_i = 1 - \sum_{k=1}^{n} p_{i,k}^2$$

where $p_{i,k}$ is the ratio of class k instances among the training instances in the i^{th} node.

A Decision Tree can also estimate the probability that an instance belongs to a particular class k: first it traverses the tree to find the leaf node for this instance, and then it returns the ratio of training instances of class k in this node.

Classification And Regression Tree (CART) algorithm is the most used to train Decision Trees. The idea is to first split the training set in two subsets using a single feature k and a threshold t_k - such as petal length ≤ 2.45 cm. To choose k and t_k it searches for the pair (k,t_k) that produces the purest subsets. The cost function that the algorithm tries to minimize is given by :

$$J(k, t_k) = \frac{m_{left}}{m} G_{left} + \frac{m_{right}}{m} G_{right}$$

where $G_{left/right}$ is the measure of the impurity of the left/right subset and $m_{left/right}$ is the number of instances in the left/right subset.

Once it has successfully split the training set in two, it splits the subsets using the same logic, then the sub subsets and so on, recursively. It stops recursing once it reaches a specific tree criteria - such as maximum depth - or if it cannot find a split that will reduce impurity.

The main issue with Decision Trees is that they are very sensitive to small variations in the training data. Random Forests can limit this instability by averaging predictions over many trees.

If one aggregates the predictions of a group of predictors he will often get better predictions than with the best individual predictor.
However, this is only true if all classifiers are, conditional on the sample, fully independent, making uncorrelated errors.
One way to get a diverse set of classifiers is to use very different training algorithms. Another approach is to use the same training algorithm for every predictor, but to train them on different random subsets of the training set. When sampling is performed with replacement, this method is called bagging. When sampling is performed without replacement, it is pasting.

For example, one can train a group of Decision Tree classifiers, each on a different random subset of the training set. To make predictions, one just obtains the predictions of all individual trees, then predict the class that gets the most votes. Such an ensemble of Decision Trees is called a Random Forest.

The Random Forest algorithm introduces extra randomness when growing trees: instead of searching for the very best feature when splitting a node, it does so for the best feature among a random subset of features. This results in a greater tree diversity, which trades a higher bias for a lower variance, generally yielding an overall better model.

Another great advantage of Random Forests is that they make it easy to measure the relative importance of each feature. It can be done by looking at how much the tree nodes that use that feature reduce impurity on average - across all trees in the forest.

V. Method

Python was chosen as the the language to perform the study. Even if it is less equipped than R on statistics, it offers a better interaction on Web-based applications as it is more general and more widespread than R. Besides, if needed, some additional libraries such as SciPy can be downloaded on Python to provide statistical analyses.

1. Data Gathering

A. Tweepy

To collect data for sentiment analysis, Twitter's API [34] was used in combination with Tweepy. It is an open source framework written in Python that facilitates tweet collection from Twitter's API [35]. It allows for filtering based on hashtags or words, and as such was considered as an efficient way of collecting relevant data. The filter keywords were chosen by selecting the most definitive Bitcoin-context words, for example "cryptocurrency" could include sentiments towards other cryptocurrencies, and so the scope must be tightened further to only include Bitcoin synonyms. These synonyms include: Bitcoin - BTC was excluded as it may be used more for commercial purposes. A CRON was created to run a Python script each day in order to harvest data from that day.

B. Binance

Many platforms exist to trade bitcoins : CoinDesk, Coinbase, Kraken, … For the purpose of this study it was chosen to gather historical price points for Bitcoin from Binance publicly available API [36]. Binance is a global cryptographic exchange platform that allows the exchange of more than 100 cryptocurrencies. Since the beginning of 2018, Binance is considered to be the world's largest cryptographic exchange platform in terms of volume. The site was launched on 14 July 2017 and is based in Hong Kong. The CEO, Zhao Changpeng, will probably become a billionaire in a few months.
Let's note that the API rules changed in August 2021. From now on each user requiring the API needs to be identified, as a consequence the API key used for this paper won't be usable after that date.

2. Sentiment Analysis

A. Preprocessing metadata and eliminating irrelevant Tweets

Once Bitcoin-related tweets are gathered, we need to preprocess the data the Twitter API gives us before feeding it to the various algorithms.
This step is actually critical in all ML projects. It is estimated that up to 80% of the ML Engineer work is spent collecting, cleaning and processing the data [37].

For this work the following pre-processing to the whole dataset is applied:
- Convert the raw data from the API from a JSON format to a relational database one
- Create a date feature more practical to use than the one given by the API using the *datetime* Python library (see VI.1)
- Create a feature specifically for the hour the tweets were posted
- Create a feature to specify which day of the week the tweet was posted
- Create a feature to change the date of creation of user account to an adequate format using the *datetime* Python library
- Create a feature calculating the average number of tweets per day for each user
- Selected all tweets in English for the sentiment analysis

When this first step is done, a more complex problem must be solved: how to handle all the automated tweets written by robots on the platform ?
It's still a pressing challenge for social media giants to handle bots. For Twitter alone, it is estimated that 15% of the users on Twitter are not actually humans [38]. Still today, platforms struggle to identify spam content posted on their platform. Often automated tweets don't convey any ideas or opinions but only have an advertising purpose to trade Bitcoins on a particular trading platform - with the URL incorporated in the tweet. In order to limit the bias of the dataset we need to get rid of them.

However, as mentioned above, the management and identification of robots is not something that can be done easily. No labels are available to categorize tweets as spam or not, so if ML is used for this task it will be needed to resort to unsupervised learning. A clustering technique - hierarchical, k-means, etc- could work but here a simpler approach was privileged.

3 different filters were tried :
- Twitter filter: Twitter proposes a categorical feature named *filter_level* on each tweet. It has 3 categories : « low », « medium » and « high ». The idea is to get rid of all low filtered tweets. However it did not work since the Twitter API categorized all the Tweets harvested as « low » - an explanation will be proposed later in the paper.
- URL: digging in the dataset, it was clear that tweets written by robots often contained a URL link - usually towards the platform of cryptocurrency trading. Hence the idea is to get rid of all tweets containing a URL.
- Get rid of the tweets were the user has an average of more than 10 tweets per day. This number is backed by the exploratory analysts of the dataset. The assumption behind this approach is that bots publish messages with a high frequency. Indeed, robots can be programmed to provide an update on Bitcoin's price each hour for example. Besides, it is probably uncommon - but not impossible - for a real person to post more than 10 times each day.

The performances of the datasets obtained from the 2 filters will be compared in section VI.

B. Clean tweets - VADER

The previous section was focused on cleaning and preprocessing metadata. Here it will be focused on the method to pre-process the content of the tweets themselves.

Usually it's not possible to directly feed text to a ML algorithm. That is because one needs to make sure the algorithm will be able to extract as much information as possible. For that purpose *preprocessor* - a library specifically for tweets - and *re* - a library to handle regular expressions - were used.
Here, thanks to the library that will be used after, the preprocessing is quite simple :

- Get rid of the first « RT » which appeared in the text when a tweet has been re-tweeted
- Get rid of the URL link, if there is one
- Get rid of mentions to others users

It was deliberately chosen to keep emojis and hashtags. Indeed they convey a sentiment by themselves and can completely change the meaning of a sentence.

Once the pre-processing is done, we need to quantify the positive or negative sentiment of each tweet. For that purpose the library *VADER* was used [39] - Valence Aware Dictionary and sEntiment Reasoner. It is a lexicon and rule-based sentiment analysis tool that is specifically adapted to sentiments expressed in social media. It is open-source under the MIT License.

The VADER sentiment lexicon is sensitive to both the polarity and the intensity of sentiments expressed in social media contexts, and is also generally applicable to sentiment analysis in other domains.

Manually creating a comprehensive sentiment lexicon is a laborious, complex and sometimes fault-prone process. It is therefore not surprising that many researchers and opinion mining practitioners rely on existing lexicons as primary resources. In the case of VADER, a list based on the review of well-established sentiment word banks -Linguistic Inquiry and Word Count, Affective Norms for English Words and General Inquirer - was generated. Many lexical features common to sentiment expression in microblogs were introduced as well.
The VADER development team empirically validated the general applicability of each candidate feature to sentiment expressions by using a wisdom of the crowd approach to obtain a reliable point estimate of each candidate feature without context.

As a result, VADER can handle complicated and ambiguous sentences including:

- negations - « not good »
- conventional use of punctuation to signal greater intensity - « Good !!! »
- conventional use of word form to signal emphasis - using ALL CAPS for words
- use of degree modifiers to alter the intensity of feeling - intensity enhancers such as « very » and intensity softeners such as « sort of »
- understanding many sentiment slang words - « sux »
- understanding many sentiment slang words as modifiers such as « friggin » or « kinda »
- understanding many sentiment emoticons such as :) and :D
- translating encoded emojis such as 💘 and 💋 and 😁

- understanding sentiment initialisms and acronyms - « lol » and « wtf »

The library returns a scoring for each string that is offered as an input. There are 4 metrics :

- The *compound* score is calculated by adding the score values of each word in the lexicon, adjusting it according to the rules, and then normalizing it so that it lies between -1 (extremely negative) and +1 (extremely positive). This is the most useful metric for a single one-dimensional measure of sentiment for a given sentence. It is also useful for researchers who wish to establish standardized thresholds for classifying sentences as positive, neutral, or negative.
- The *pos*, *neu*, and *neg* scores are ratios for the proportions of the text that fall into each category - so they should all be equal or close to 1 when using the float operation. These are the most useful measures for analyzing context and representing how the mood is conveyed in a particular sentence or embedded in the rhetoric. For example, different writing styles may contain strongly positive or negative sentiment in different proportions of neutral text, that is to say some writing styles may reflect a preference for strongly flavored rhetoric, while other ones use a large proportion of neutral text while conveying a similar compounds sentiment.

Below some examples of how the code performed on a sample are provided: [40]

VADER is smart, handsome, and funny.---------- {'pos': 0.746, 'compound': 0.8316, 'neu': 0.254, 'neg': 0.0}
VADER is smart, handsome, and funny!---------- {'pos': 0.752, 'compound': 0.8439, 'neu': 0.248, 'neg': 0.0}
VADER is very smart, handsome, and funny.---------- {'pos': 0.701, 'compound': 0.8545, 'neu': 0.299, 'neg': 0.0}
VADER is VERY SMART, handsome, and FUNNY.---------- {'pos': 0.754, 'compound': 0.9227, 'neu': 0.246, 'neg': 0.0}
VADER is VERY SMART, handsome, and FUNNY!!!---------- {'pos': 0.767, 'compound': 0.9342, 'neu': 0.233, 'neg': 0.0}
VADER is VERY SMART, uber handsome, and FRIGGIN FUNNY!!!--------- {'pos': 0.706, 'compound': 0.9469, 'neu': 0.294, 'neg': 0.0}
VADER is not smart, handsome, nor funny.---------- {'pos': 0.0, 'compound': -0.7424, 'neu': 0.354, 'neg': 0.646}
The book was good.---------- {'pos': 0.492, 'compound': 0.4404, 'neu': 0.508, 'neg': 0.0}
At least it isn't a horrible book.---------- {'pos': 0.363, 'compound': 0.431, 'neu': 0.637, 'neg': 0.0}
The book was only kind of good.---------- {'pos': 0.303, 'compound': 0.3832, 'neu': 0.697, 'neg': 0.0}
The plot was good, but the characters are uncompelling and the dialog is not great. {'pos': 0.094, 'compound': -0.7042, 'neu': 0.579, 'neg': 0.327}
Today SUX!---------- {'pos': 0.0, 'compound': -0.5461, 'neu': 0.221, 'neg': 0.779}
Today only kinda sux! But I'll get by, lol---------- {'pos': 0.317, 'compound': 0.5249, 'neu': 0.556, 'neg': 0.127}
Make sure you :) or :D today!---------- {'pos': 0.706, 'compound': 0.8633, 'neu': 0.294, 'neg': 0.0}
Catch utf-8 emoji such as 💘 and 💋 and 😁---------- {'pos': 0.279, 'compound': 0.7003, 'neu': 0.721, 'neg': 0.0}
Not bad at all---------- {'pos': 0.487, 'compound': 0.431, 'neu': 0.513, 'neg': 0.0}

A VADER analysis has been applied to each tweet and the outcome of resulting metrics in dedicated labels stored.
Additionally an another metric to categorize the tweets as generally positive (1), negative (-1) or neutral (0) was created. To this end, we use the compound metric and an arbitrary threshold found in the literature of ± 5%. [40]

C. Data aggregation

Obviously it is not possible to directly link each tweet to a tiny increase or decrease of the price of Bitcoin. It is a global point of view that could result in such a change. As a consequence, the analysis must focus on aggregated data, averaged and collected on different timeframe. A priori it is not possible to know which time window is the most adequate, so a benchmark on various time frames was conducted to determine which

one would be better fitted to study relationships between sentiments on Twitter and Bitcoin.

The time windows considered were :
- Tweets collected during 1 day
- Tweets collected during 1 hour
- Tweets collected during 5 minutes

From that point on it is possible to feed the data to the various Machine Learning algorithms (see section IV)

VI. Results

1. Data collection

A. Twitter

a. Volume

The data collected by the Twitter API follows a JSON format. Below an example is provided.

{'created_at': 'Tue Sep 01 15:12:07 +0000 2020',
'id': 1300813696306950144,
'id_str': '1300813696306950144',
'text': "Manipulators or insiders are trying to drop #Digibyte price slowly in small chunks.But I tell you guys don't get fe… https://t.co/ABrRNyBKAn",
'source': 'Twitter Web App',
'truncated': True,
'in_reply_to_status_id': None,
'in_reply_to_status_id_str': None,
'in_reply_to_user_id': None,
'in_reply_to_user_id_str': None,
'in_reply_to_screen_name': None,
'user': {'id': 954936281724841985,
'id_str': '954936281724841985',
'name': 'CryptoKing',
'screen_name': 'TheCryptoKuber',
'location': 'Alien',
'url': None,
'description': 'Out of box thinker, critical analyst',
'translator_type': 'none',
'protected': False,
'verified': False,
'followers_count': 205,
'friends_count': 344,
'listed_count': 1,
'favourites_count': 2406,
'statuses_count': 2306,
'created_at': 'Sun Jan 21 04:38:46 +0000 2018',
'utc_offset': None,
'time_zone': None,
'geo_enabled': False,
'lang': None,
'contributors_enabled': False,
'is_translator': False,

'profile_background_color': 'F5F8FA',
'profile_background_image_url': '',
'profile_background_image_url_https': '',
'profile_background_tile': False,
'profile_link_color': '1DA1F2',
'profile_sidebar_border_color': 'C0DEED',
'profile_sidebar_fill_color': 'DDEEF6',
'profile_text_color': '333333',
'profile_use_background_image': True,
'profile_image_url': 'http://pbs.twimg.com/profile_images/1272378126987857920/
EBB8nPgZ_normal.jpg',
'profile_image_url_https': 'https://pbs.twimg.com/profile_images/
1272378126987857920/EBB8nPgZ_normal.jpg',
'profile_banner_url': 'https://pbs.twimg.com/profile_banners/
954936281724841985/1592194364',
'default_profile': True,
'default_profile_image': False,
'following': None,
'follow_request_sent': None,
'notifications': None},
'geo': None,
'coordinates': None,
'place': None,
'contributors': None,
'is_quote_status': False,
'extended_tweet': {'full_text': "Manipulators or insiders are trying to drop #Digibyte
price slowly in small chunks.But I tell you guys don't get fear and sell your #Digibyte
rather buy as much you can.Not a financial advice. It is long term very profitable
investment. #Bitcoin #ethereum #litecoin #blockchain",
'display_text_range': [0, 277],
'entities': {'hashtags': [{'text': 'Digibyte', 'indices': [44, 53]},
{'text': 'Digibyte', 'indices': [132, 141]},
{'text': 'Bitcoin', 'indices': [237, 245]},
{'text': 'ethereum', 'indices': [246, 255]},
{'text': 'litecoin', 'indices': [256, 265]},
{'text': 'blockchain', 'indices': [266, 277]}],
'urls': [],
'user_mentions': [],
'symbols': []}},
'quote_count': 0,
'reply_count': 0,
'retweet_count': 0,
'favorite_count': 0,
'entities': {'hashtags': [{'text': 'Digibyte', 'indices': [44, 53]}],
'urls': [{'url': 'https://t.co/ABrRNyBKAn',
'expanded_url': 'https://twitter.com/i/web/status/1300813696306950144',
'display_url': 'twitter.com/i/web/status/1…',
'indices': [117, 140]}],
'user_mentions': [],
'symbols': []},
'favorited': False,
'retweeted': False,

'filter_level': 'low',
'lang': 'en',
'timestamp_ms': ‘1598973127806'}

The JSON format is not the most adequate for ML, it was changed it to a *pandas* data frame format. Besides, the API gives a lot of information that is not needed. Finally, it is noticeable the feature *created_at* is not directly useful and needs to be pre-processed before feeding to an algorithm, as explained in section V.B.

A total of 707,718 Bitcoin-related tweets were gathered from the Twitter API, from which a database of 13 columns was built. The dataset is complete aside from a few missing features for *user_name* and *country*.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 707718 entries, 0 to 707717
Data columns (total 13 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   text                 707718 non-null  object
 1   lang                 707718 non-null  object
 2   country              707716 non-null  object
 3   date                 707718 non-null  object
 4   user_name            707685 non-null  object
 5   user_id              707718 non-null  int64
 6   nb_of_tweets         707718 non-null  int64
 7   user_creation_date   707718 non-null  object
 8   verified             707718 non-null  int64
 9   filter_level         707718 non-null  object
 10  favorite             707718 non-null  int64
 11  retweet              707718 non-null  int64
 12  url                  707718 non-null  int64
dtypes: int64(6), object(7)
memory usage: 70.2+ MB
```

Figure 7. Raw data collected from the twitter API

Below a definition of each feature is given :

- *text* is the full content of the tweet - not limited to 140 characters
- *lang* is the language in which the tweet is written
- *country* is the country the tweet was posted from
- *date* is the date at which the tweet was posted
- *user_name* is the name of the user on Twitter
- *user_id* is a unique id used internally by Twitter
- *nb_of_tweets* if the total number of tweets posted by the user since he/she created it’s account
- *user_creation_date* is the creation date of the account
- *verified* indicates whether the account is verified by Twitter or not
- *filter_level* is the the maximum value of the parameter which may be used and still stream this tweet.
- *favorite* is the number of times this tweet has been favorited
- *retweets* is the number of times this tweet has been retweeted
- *url* indicates whether the tweet contains a URL or not

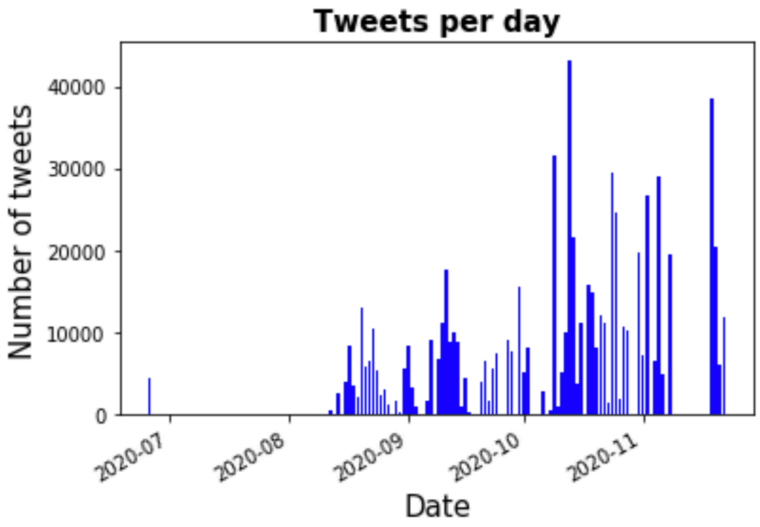

Figure 8. Tweets collected per day in the dataset

It is worth pointing out at a few facts about the data collected.

First, the 707,718 tweets collected during 2 months don't correspond to the totality of tweets posted on the platform about Bitcoin but only a sample thereof. The quality of the sample depends mainly on the Twitter API. However this API follows a freemium model and favors developers who pay to have access to the entirety of the API. The prohibitive cost of accessing the API made us stick to the free versions with all the drawbacks that it implies.

Second, the time distribution of the dataset seems a bit odd. The peak in July 2020 was a test and shouldn't be considered in the following comment. However, it appears clearly that the variance of tweets collected per day is high: it goes from 0 up to 40,000. 2 reasons may account for that fact: on the one hand the CRON may have had some difficulty to run sometimes and not access correctly the API, on the other hand a free version of Twitter API was used. It may affect the capacity at which tweets are accessed. Besides the free access to Twitter API may explain as well the « low » filter obtained on all tweets (section V.2.A)
As the access to data is not fully reliable, this may complicates the remaining part of the work.

b. Exploratory analysis

Additionally an exploratory analysis can be performed on the original dataset. Figure 9 depicts the distribution of the top languages in the dataset.

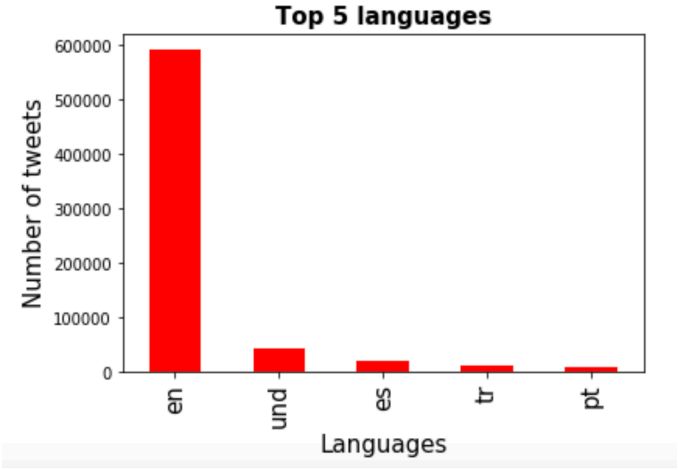

Figure 9. Most frequent languages identified from the tweets collected

English is clearly the dominant language with nearly 600,000 tweets overall. On the other hand, for many tweets it is not possible to derive a language - « undetermined ». As expected, popular languages such as Spanish and Portuguese complete the top, while Turkish occupies a fairly important place considering its number of speakers worldwide.

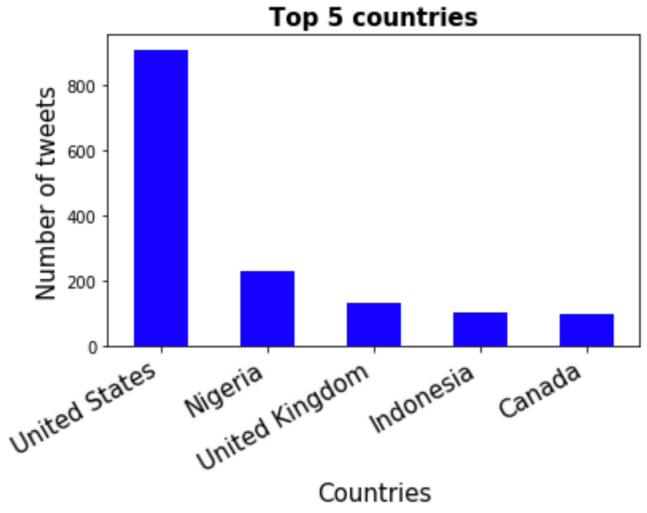

Figure 10. Most frequent countries of origin identified from the tweets collected

Analyzing the country breakdown in Figure 10, a clear bias is visible. Most of the tweets have no location - again, this may be a consequence of Twitter free AP). The countries that come out on top are rather unexpected the USA are active in the crypto-community and have a large population, but the 4 others are either relatively small or not up-to-date in relation with crypto development.

Focusing on the users, 232,528 unique users were identified in the dataset. The distribution of tweets per user seems to follow a power law with a a few users concentrating many tweets, as it is displayed on Figure 11.

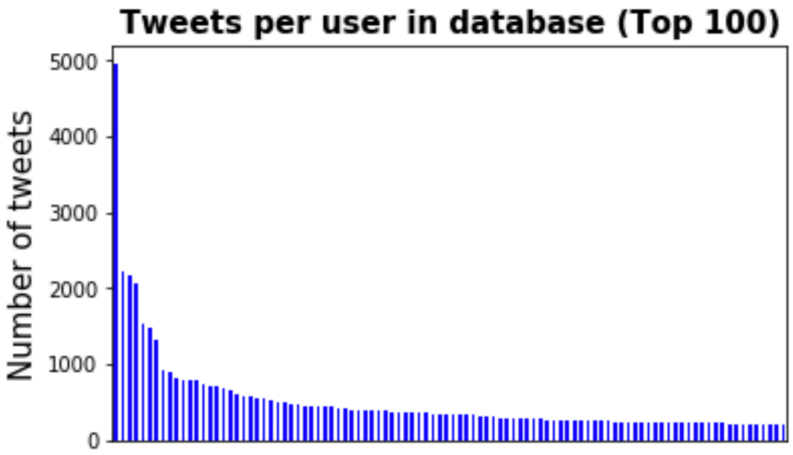

Figure 11. Number of tweets per for the 100 most active users

A study on the distribution of average tweets per day per user was also conducted. In the dataset there is an average of 36 tweets per day and a median of 5, which demonstrates a very high variability as depicted on figure 12. Some users go up to 4000 tweets per day.

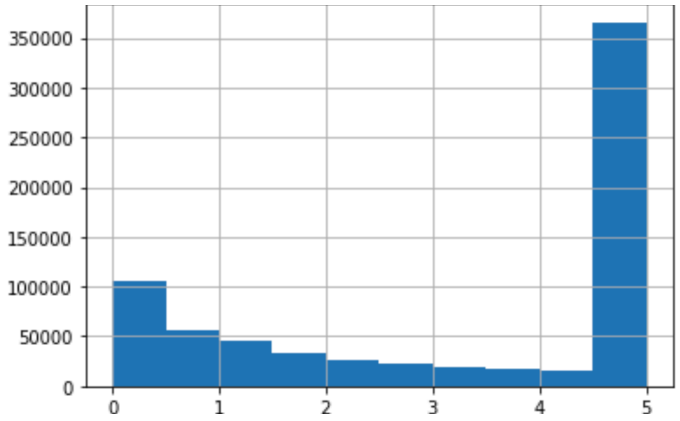

Figure 12. Distribution of the average number of tweets posted per day per user

Finally several other traits were analyzed. Nearly all of the tweets collected are from non-verified users, all of the tweets have a low filter level, many tweets contain a URL so this technique of spam filtering could be adequate, and all the tweets have 0 retweets and 0 favorites. The reason that can explain this is the live streaming of tweets: each tweet is harvested as soon as it is posted and other users don't have time to react to it.

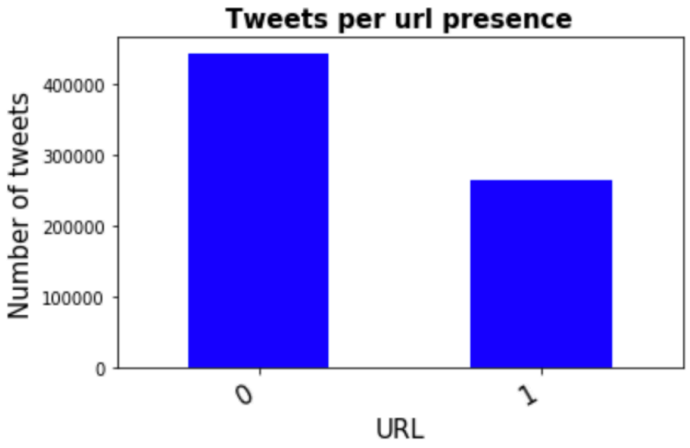

Figure 13. Presence of URL in tweets

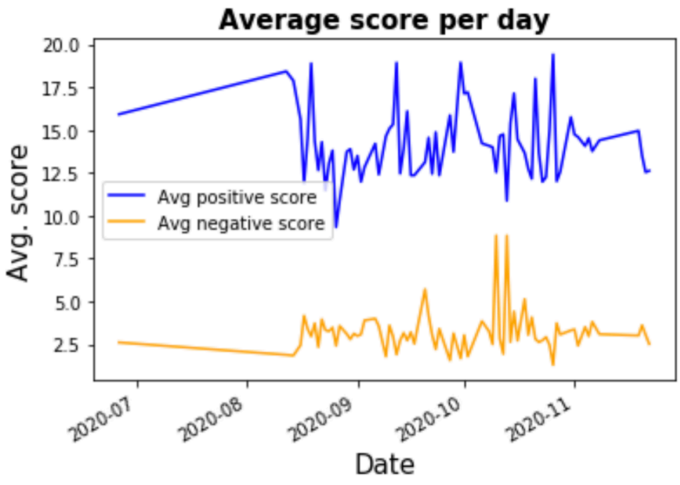

Figure 14. Positive and negative sentiment aggregated by day

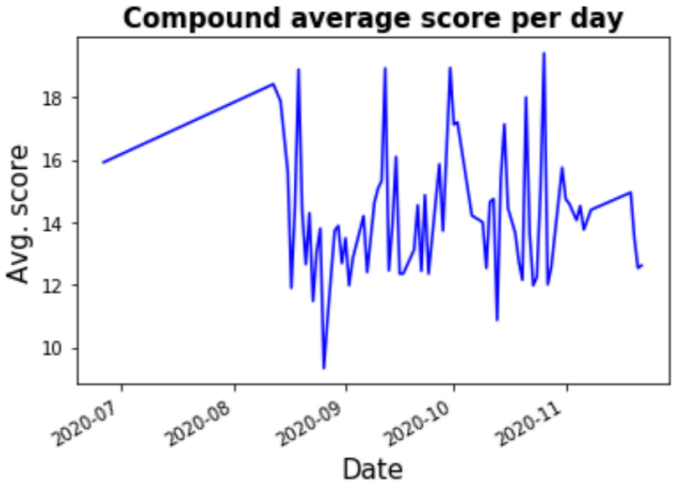

Figure 15. Compound sentiment aggregated by day

As for the sentiment analysis, as displayed on figure 14, it seems Bitcoin is more positively than negatively valued on Twitter, even if on some days negative trend gains momentum. However let's not forget that the neutral tone is dominant as the sum of positive, negative and neutral sentiment is equal to 100.
Thus, as a whole, people post about Bitcoin in a rather neutral style, even if more positively than negatively, (Figure 14) with a globally positive aspect (figure 15).

B. Binance

The Binance API can be requested directly from Python using the library *python-binance* and choosing the time frame of interest. 3 different time frames were selected - the same ones as explained in section V.3.

Here, only the results of the time window corresponding to 1 day are presented. The data returned by the API is in a JSON format and is converted into a relational database.

date_format	open	high	low	close	volume	close_time	quote_av	trades	tb_base_av	tb_quote_av	ignore	variation_cat
2020-06-13	9464.96	9494.73	9351.00	9473.34	27759.784851	1592092799999	2.616732e+08	392763	13531.007214	1.275772e+08	0	1
2020-06-14	9473.34	9480.99	9245.00	9342.10	30055.506608	1592179199999	2.821313e+08	415568	14465.416151	1.358116e+08	0	0
2020-06-15	9342.10	9495.00	8910.45	9426.02	86107.924707	1592265599999	7.935188e+08	839521	41630.747529	3.840311e+08	0	1
2020-06-16	9426.05	9589.00	9373.09	9525.59	52052.446927	1592351999999	4.944583e+08	543481	25692.839625	2.440974e+08	0	1
2020-06-17	9526.97	9565.00	9236.61	9465.14	48046.411152	1592438399999	4.532098e+08	536158	22842.831276	2.155327e+08	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
2021-01-10	40088.22	41350.00	35111.11	38150.02	118209.544503	1610323199999	4.604035e+09	2628050	55451.344673	2.160977e+09	0	0
2021-01-11	38150.02	38264.74	30420.00	35404.47	249131.539943	1610409599999	8.426880e+09	4431451	122133.406190	4.133116e+09	0	0
2021-01-12	35410.37	36628.00	32531.00	34051.24	133948.151996	1610495999999	4.651302e+09	2674145	65098.310196	2.261732e+09	0	0
2021-01-13	34049.15	37850.00	32380.00	37371.38	124477.914938	1610582399999	4.322877e+09	2514289	63981.038306	2.222873e+09	0	1
2021-01-14	37371.38	38786.10	36701.23	38004.54	41798.683654	1610668799999	1.581056e+09	934957	20973.031456	7.935999e+08	0	1

Figure 16. Data collected from Binance - aggregated by day

Below an explanation for each feature is provided:

- d*ate_format* is the day from which the following features are related to
- o*pen* is the price of Bitcoin at the opening of the train session
- *high* is the highest price reached during the session
- *low* is the lowest price reached during the session
- *close* is the price of Bitcoin at the end of the trading session
- *volume* is the number of units of Bitcoin traded in the market during the session
- *close_time* is the moment the market close. Since crypto currency trading platform works 24/7, this data points is not useful
- *quote_av* : quote asset volume is the amount of $ exchanged on the platform during the session
- *trades* is the number of trades conducted during the session
- *tb_base_av* : taker buy base asset volume, it is the volume of Bitcoin units exchanged by takers on a buy order
- *tb_quote_av* : taker buy quote asset volume, it is the volume of $ traded in the market by takers on a buy order
- *ignore* is an irrelevant data
- *variation* indicates whether the price increases or decreases during the session

The data set is composed of 217 days, 128 of which underwent a rise in the price of Bitcoin. Hence the dataset is fairly balanced but increases are much more important than

decreases in terms of absolute change, as shown in the rising price from 10,000$ to nearly 40,000$ in the dataset.

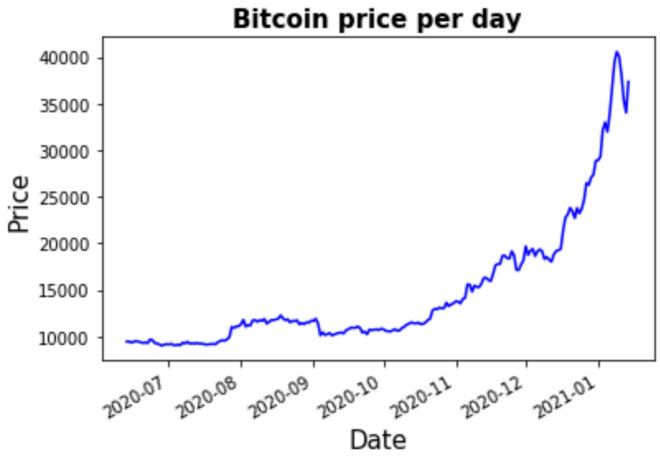

Figure 17. Bitcoin price evolution as a function of time

C. Final dataset for Machine Learning analysis

Finally both datasets from Twitter and Binance are combined. First the data from Twitter was grouped through the adequate time frame before performing the inner joining on the datetime feature.

Some features that could be of interest were added, as well so the final dataset is composed of:

- d*ate_format* is the day - or datetime - from which following features are related to
- o*pen* is the price of Bitcoin at the opening of the train session
- *close* is the price of Bitcoin at the end of the trading session
- *pos* is the average positive score of tweets during the session
- *neu* is the average neutral score of tweets during the session
- *neg* is the average negative score of tweets during the session
- *compound* is the average compound score of tweets during the session
- *variation_cat* indicates whether the price increases or decreases during the day (the output)
- *nb_of_tweets* indicates the number of tweets harvested through the session - it is not the real number of tweets on Bitcoin posted on the platform
- *variation* is the price variation along the time frame
- *variation_%* is price variation in % along the time frame
- *pos_variation%* is the variation in % of the average positive score of tweets
- *neu_variation%* is the variation in % of the average neutral score of tweets
- *neg_variation%* is the variation in % of the average negative score of tweets
- *compound_variation%* is the variation in % of the average compound score of tweets

- *nb_of_tweets_variation%* is the variation in % of the number of harvested tweets through the time frame

Let's note 2 things here :
- Such a dataset is obtained for each time frame and each way of filtering (URL or average number of tweets), that is to say 3x2 datasets.
- The variation of all scores is thought to be of interest because maybe more than the absolute feeling it may be its variation that drives changes. However, some problems when collecting the data have been encountered, so the variation computed is not the real one between 2 truly consecutive time frames, but rather the one between 2 consecutive time frames in the dataset.

A quick first approach through linear correlation does not lead to an apparent link for numeric features. It even appears that the most important linear correlation for *variation* exists with *nb_of_tweets* and *nb_of_tweets_variation%* which are not directly linked to the total number of tweets posted on Twitter and thus the « hype » of the moment.
The preprocessed data will now be applied to the 3 classification algorithms explained in IV.3 to find a potential relationship between price evolution and other variables.
Besides, for each filter, time frame and algorithm a mix of 3 difference set of features will be tested:
- One case with all features: *pos, neg, compound, nb_of_tweets, pos_variation%, neg_variation%, compound_variation%, nb_of_tweets_variation%*
- One case with only absolute features: *pos, neg, compound, nb_of_tweets*
- One case with only variation features: *pos_variation%, neg_variation%, compound_variation%, nb_of_tweets_variation%*

2. Classification results

A. Filter on average number of tweets

a. 1 day time frame

α. Logistic regression

```
Accuracy: 46.67%
Precision: 46.67%
Recall: 100.00%
f1 score: 63.64%

array([[0, 8],
       [0, 7]])
```

```
Accuracy: 46.67%
Precision: 50.00%
Recall: 75.00%
f1 score: 60.00%

array([[1, 6],
       [2, 6]])
```

```
Accuracy: 40.00%
Precision: 42.86%
Recall: 85.71%
f1 score: 57.14%

array([[0, 8],
       [1, 6]])
```

Results are displayed as followed :
- The left one is obtained using all features - absolute and the ones from variation
- The center one is obtained using only absolute features

- The right one is obtained using only variation features

The same display will be provided for all other results.

As for the various metrics results are rather similar across models. However the confusion matrices are different. For example the models trained with all features and variation features tend to only categorize the test set as an increase in price while the one with only absolute features is a bit more balanced.
In any case performances are not sufficient for one of these models to be put into production. However let's bear in mind the results obtained by the study from Lamon et al. (2015) where it was found that logistic regression was the most efficient way to classify tweets with an accuracy of 43.9% - less than tossing a coin - for price increases and 61.9% for price decreases.

β. k-Nearest Neighbors

```
Accuracy: 46.67%
Precision: 45.45%
Recall: 71.43%
f1 score: 55.56%

array([[2, 6],
       [2, 5]])
```

```
Accuracy: 53.33%
Precision: 57.14%
Recall: 50.00%
f1 score: 53.33%

array([[4, 3],
       [4, 4]])
```

```
Accuracy: 53.33%
Precision: 50.00%
Recall: 71.43%
f1 score: 58.82%

array([[3, 5],
       [2, 5]])
```

We observe results are rather similar as for the metrics. Confusion matrices are balanced in all cases. However the results are quite poor as well and couldn't be put into production. For this time frame kNN seems to perform slightly worse than Logistic Regression but that could be explained by the bias toward increase observed for Logistic Regression.

γ. Random Forest

```
Accuracy: 40.00%
Precision: 40.00%
Recall: 57.14%
f1 score: 47.06%

array([[2, 6],
       [3, 4]])
```

```
Accuracy: 40.00%
Precision: 45.45%
Recall: 62.50%
f1 score: 52.63%

array([[1, 6],
       [3, 5]])
```

```
Accuracy: 53.33%
Precision: 50.00%
Recall: 71.43%
f1 score: 58.82%

array([[3, 5],
       [2, 5]])
```

Results seems to be better with the model trained only with the variation features. Confusion matrices are rather balanced in all cases. However the results are quite poor as well and couldn't be put into production. For this time frame Random Forest seems to perform slightly worse than Logistic Regression.

δ. Conclusion on this time frame

All in all the 1-day time frame doesn't seem to be able to achieve highly accurate outputs. One reason could be 1-day is a too large time frame to explain all the micro/ macro events that could occur and have an effect at very short term on the price.

The model with the highest f1-score obtained is the Logistic Regression trained on all features but his performance stays quite poor. Besides it seems to be biased since it categorizes all new data as an increase in price.

b. 1h time frame

α. Logistic regression

```
Accuracy: 55.00%
Precision: 62.07%
Recall: 72.00%
f1 score: 66.67%

array([[ 4, 11],
       [ 7, 18]])
```

```
Accuracy: 52.50%
Precision: 68.00%
Recall: 60.71%
f1 score: 64.15%

array([[ 4,  8],
       [11, 17]])
```

```
Accuracy: 52.50%
Precision: 62.50%
Recall: 60.00%
f1 score: 61.22%

array([[ 6,  9],
       [10, 15]])
```

Results seems to be slightly better when the model is trained on all features. However the metrics tend to be similar for all 3 models. The results are better than with the 1-day time frame. However it is still not sufficient to be put into production.
What's more, the bias toward an increase is less important than in the 1 day time frame.

β. k-Nearest Neighbor

```
Accuracy: 47.50%
Precision: 59.09%
Recall: 52.00%
f1 score: 55.32%

array([[ 6,  9],
       [12, 13]])
```

```
Accuracy: 52.50%
Precision: 69.57%
Recall: 57.14%
f1 score: 62.75%

array([[ 5,  7],
       [12, 16]])
```

```
Accuracy: 65.00%
Precision: 76.19%
Recall: 64.00%
f1 score: 69.57%

array([[10,  5],
       [ 9, 16]])
```

Similar to the 1-day time frame, these models are more balanced since all models are prone to categorize new data either as an increase or a decrease in price in the same proportion. Results vary way more with different features trained on and the best one is obtained only with variation features. For this time frame kNN seems to perform a bit better than Logistic Regression.

γ. Random Forest

```
Accuracy: 40.00%
Precision: 52.63%
Recall: 40.00%
f1 score: 45.45%

array([[ 6,  9],
       [15, 10]])
```

```
Accuracy: 55.00%
Precision: 70.83%
Recall: 60.71%
f1 score: 65.38%

array([[ 5,  7],
       [11, 17]])
```

```
Accuracy: 45.00%
Precision: 57.89%
Recall: 44.00%
f1 score: 50.00%

array([[ 7,  8],
       [14, 11]])
```

Results seems to be better with the model trained only with the absolute features. Confusion matrices are well balanced in all cases. However the results are quite poor as

well and couldn't be put into production. For this time frame Random Forest seems to perform slightly worse than Logistic Regression.

δ. Conclusion on this time frame

All in all the 1h-time frame seems to be better suited than the 1-day time frame to make predictions on a future evolution of the price of Bitcoin. However the results achieved would need to be improved before being used in a trading strategy.
The best model is the one obtained for kNN trained on only variation features but its performance stays under what could be expected before being put into production.

c. 5min time frame

α. Logistic regression

```
Accuracy: 50.65%
Precision: 50.55%
Recall: 88.39%
f1 score: 64.32%

array([[ 19, 134],
       [ 18, 137]])
```

```
Accuracy: 45.45%
Precision: 47.54%
Recall: 74.36%
f1 score: 58.00%

array([[ 24, 128],
       [ 40, 116]])
```

```
Accuracy: 50.97%
Precision: 50.65%
Recall: 100.00%
f1 score: 67.25%

array([[  2, 151],
       [  0, 155]])
```

Since here we have way more data, better results could be expected given that algorithm accuracy increases with the sample size/amount of data. However that is not the case here. One explanation could be the data quality, as already explained above concerning the free Twitter API.
The bias towards an increase is really important despite the data quantity and in most cases the models predict an increase.

β. k-Nearest Neighbors

```
Accuracy: 47.40%
Precision: 47.83%
Recall: 49.68%
f1 score: 48.73%

array([[69, 84],
       [78, 77]])
```

```
Accuracy: 46.75%
Precision: 47.44%
Recall: 47.44%
f1 score: 47.44%

array([[70, 82],
       [82, 74]])
```

```
Accuracy: 48.70%
Precision: 49.09%
Recall: 52.26%
f1 score: 50.62%

array([[69, 84],
       [74, 81]])
```

Similar to the other time frames, this model is more balanced since all models are prone to categorize new data either as an increase or a decrease in price. Results are very similar and don't depend on the features the models were trained on. For this time frame kNN seems to perform worse than Logistic Regression but this is only because Logistic Regression is biased towards an increase.

γ. Random Forest

```
Accuracy: 46.75%
Precision: 48.33%
Recall: 83.87%
f1 score: 61.32%

array([[ 14, 139],
       [ 25, 130]])
```

```
Accuracy: 45.13%
Precision: 47.62%
Recall: 83.33%
f1 score: 60.61%

array([[  9, 143],
       [ 26, 130]])
```

```
Accuracy: 48.70%
Precision: 49.30%
Recall: 67.74%
f1 score: 57.07%

array([[ 45, 108],
       [ 50, 105]])
```

As for the metrics, results are rather similar, as well as the confusion matrices. In any case performances are not sufficient for a large deployment.
While it wasn't observed in previous time frames, Random Forest seems to have a bias toward increase for the 5-min one.

δ. Conclusion on this time frame

All in all the 5-min time frame seems to perform slightly worse than the 1h time frame but better than the 1-day one. As a consequence 5-min may be a bit too short to aggregate data and capture relevant events and their magnitude.
Again the results achieved would need to be improved before being used in a trading strategy.
The model with the best f1 score is the one obtained for Logistic Regression trained only on variation features but its performance stays under what could be expected before being put into production. Besides it seems to be biased since it predicts all new data as an increase .

B. Filter on URL

a. 1 day time frame

α. Logistic regression

```
Accuracy: 53.33%
Precision: 50.00%
Recall: 85.71%
f1 score: 63.16%

array([[2, 6],
       [1, 6]])
```

```
Accuracy: 46.67%
Precision: 50.00%
Recall: 75.00%
f1 score: 60.00%

array([[1, 6],
       [2, 6]])
```

```
Accuracy: 46.67%
Precision: 46.67%
Recall: 100.00%
f1 score: 63.64%

array([[0, 8],
       [0, 7]])
```

Here results are very similar to those of the other filter: there is the same bias toward an increase and similar values for all metrics.

β. k-Nearest Neighbors

```
Accuracy: 46.67%
Precision: 42.86%
Recall: 42.86%
f1 score: 42.86%

array([[4, 4],
       [4, 3]])
```

```
Accuracy: 80.00%
Precision: 77.78%
Recall: 87.50%
f1 score: 82.35%

array([[5, 2],
       [1, 7]])
```

```
Accuracy: 46.67%
Precision: 42.86%
Recall: 42.86%
f1 score: 42.86%

array([[4, 4],
       [4, 3]])
```

Here results are poorer than for the previous filter on the models trained with all features and only variation ones. However when trained only on absolute features promising results are obtained: 82% of f1 score. Considering the small training set and test set available on this time frame these results are still highly data-dependent and may not be replicated on other sets.

γ. Random Forest

```
Accuracy: 46.67%
Precision: 45.45%
Recall: 71.43%
f1 score: 55.56%

array([[2, 6],
       [2, 5]])
```

```
Accuracy: 33.33%
Precision: 41.67%
Recall: 62.50%
f1 score: 50.00%

array([[0, 7],
       [3, 5]])
```

```
Accuracy: 53.33%
Precision: 50.00%
Recall: 100.00%
f1 score: 66.67%

array([[1, 7],
       [0, 7]])
```

It seems slightly better results are obtained with this filter than with the previous one. However a bias toward increase, that wasn't noted with the other filter, is present and could explain the better outcomes.

δ. Conclusion on this time frame

In general results don't seem to differ a lot between this filter and the previous one with this time frame. Besides results are still insufficient for the models to be put into production aside from the kNN trained only on absolute features which is really promising - an additional study is required to check if this kind of results could be replicated on other datasets.

b. 1h time frame

α. Logistic regression

```
Accuracy: 50.00%
Precision: 59.26%
Recall: 64.00%
f1 score: 61.54%

array([[ 4, 11],
       [ 9, 16]])
```

```
Accuracy: 50.00%
Precision: 66.67%
Recall: 57.14%
f1 score: 61.54%

array([[ 4,  8],
       [12, 16]])
```

```
Accuracy: 52.50%
Precision: 62.50%
Recall: 60.00%
f1 score: 61.22%

array([[ 6,  9],
       [10, 15]])
```

Results are very similar to those with the other filter: same bias toward an increase and similar value for all metrics, but slightly worse.

β. k-Nearest Neighbors

```
Accuracy: 40.00%
Precision: 52.63%
Recall: 40.00%
f1 score: 45.45%

array([[ 6,  9],
       [15, 10]])
```

```
Accuracy: 40.00%
Precision: 60.00%
Recall: 42.86%
f1 score: 50.00%

array([[ 4,  8],
       [16, 12]])
```

```
Accuracy: 57.50%
Precision: 68.18%
Recall: 60.00%
f1 score: 63.83%

array([[ 8,  7],
       [10, 15]])
```

With this filter on this time frame kNN doesn't seem to work well, while with the other filter the algorithm was able to achieve better results on all 3 models.
The best model is also obtained when trained only on variation features, the same as with the previous filter.

γ. Random Forest

```
Accuracy: 47.50%
Precision: 58.33%
Recall: 56.00%
f1 score: 57.14%

array([[ 5, 10],
       [11, 14]])
```

```
Accuracy: 47.50%
Precision: 65.22%
Recall: 53.57%
f1 score: 58.82%

array([[ 4,  8],
       [13, 15]])
```

```
Accuracy: 47.50%
Precision: 59.09%
Recall: 52.00%
f1 score: 55.32%

array([[ 6,  9],
       [12, 13]])
```

Results don't vary much along all 3 models and seem to be better than with the other filter - even if the other was able to achieve better results when trained only on absolute features.

δ. Conclusion on this time frame

Broadly speaking it seems better results are achieved using the filter on the average number of tweets per day per user on the 1-hour time frame.
Besides, results are generally slightly worse on this time frame than for the 1-day time frame - as opposed to those with the other filter. An explanation could be the scarce data available on the 1-day time frame which would mean results couldn't be replicated.

c. 5min time frame

α. Logistic regression

```
Accuracy: 49.68%
Precision: 57.05%
Recall: 50.28%
f1 score: 53.45%

array([[64, 67],
       [88, 89]])
```

```
Accuracy: 45.45%
Precision: 46.64%
Recall: 78.15%
f1 score: 58.42%

array([[ 22, 135],
       [ 33, 118]])
```

```
Accuracy: 54.87%
Precision: 63.19%
Recall: 51.41%
f1 score: 56.70%

array([[78, 53],
       [86, 91]])
```

In all cases results are worse than those with the filter with the average number of tweets per day.
Besides here the best model is obtained when trained on absolute features only, while it was achieved when trained on variation features only with the previous filter.

β. k-Nearest Neighbors

```
Accuracy: 45.45%
Precision: 52.98%
Recall: 45.20%
f1 score: 48.78%

array([[60, 71],
       [97, 80]])
```

```
Accuracy: 48.38%
Precision: 47.65%
Recall: 53.64%
f1 score: 50.47%

array([[68, 89],
       [70, 81]])
```

```
Accuracy: 56.49%
Precision: 64.24%
Recall: 54.80%
f1 score: 59.15%

array([[77, 54],
       [80, 97]])
```

A large difference between the last model and the others is observed. It seems kNN performs better when absolute features are not involved at all. Results are better than with the other filter.

γ. Random Forest

```
Accuracy: 43.51%
Precision: 51.18%
Recall: 36.72%
f1 score: 42.76%

array([[ 69,  62],
       [112,  65]])
```

```
Accuracy: 45.78%
Precision: 46.80%
Recall: 77.48%
f1 score: 58.35%

array([[ 24, 133],
       [ 34, 117]])
```

```
Accuracy: 48.38%
Precision: 57.89%
Recall: 37.29%
f1 score: 45.36%

array([[ 83,  48],
       [111,  66]])
```

Results are poorer when variation features are involved, and even when they are not, better results can be explained by a bias toward a price increase.
However there is no bias toward an increase for all 3 models as there was with the other filter.

δ. Conclusion on this time frame

Broadly speaking, on this time frame it seems results are poorer using the URL filter, even if kNN performs better. The best model is obtained on kNN when trained only on variation features.

VII. General conclusion and further research

The main objective of this paper was to predict a decrease or an increase of Bitcoin prices based on the data collected on Twitter (positivity, volume, objectivity, …). Broadly speaking, it appears the sentiments are a promising indicators for such tasks. This being said, on balanced data sets, some models have reached an accuracy of 40%, i.e. less than a random classification model which should be around 50% for a binary output. For this type of model, it is possible to increase their performance by selecting the opposite output to the one predicted after training.
However we must not forget that this study is dealing with human behavior that is intrinsically random, unexpected and not always rational. As such it seems unlikely to be able to achieve results as accurate as in hard sciences based on natural laws. Nonetheless an accuracy of 75% could be achieved that would show the usefulness of Twitter sentiments. Indeed some results are encouraging, the kNN - 1day - trained on only absolute feature - filter on URL and the kNN - 1 hour - trained on only variation features - filter on number of tweets per day, but a deeper analysis is required to achieve better results and, if so, to be put into production.
What's more, in relation with previous studies, such as the one of Lamon et al. (2015) we were able to achieve better results : their study only reached an accuracy of 43.9% - less than tossing a coin - for price increases and 61.9% for price decreases, whereas in this thesis we were able to achieve a global 65% accuracy and 82% accuracy for two models.

Besides the results themselves, various interesting points from the study can be drawn.

1. In general, it seems the 1 hour time frame is the best suited to aggregate data and derive relationships from sentiments on Twitter. 1 day is too large considering all the events that could occur and impact the price, while 5 min is too short. Nonetheless, better results could be achieved on other time frames.

2. It's not possible to have an a priori assessment on the best features to use. We've observed in some cases that a mix of all features performs better, while in others it was only the absolute ones or the variation ones. We need to test all to find the best model - No Free Lunch.

3. Similarly there is no algorithm that works better overtime, although kNN seems to be the one achieving the best results globally. Again, deeper research would be needed. As a non-parametric model that can detect patterns from data without any hypothesis, it may be more equipped than Logistic Regression to achieve such a task.

4. The performances of the two filters seem to be very similar, even if the one on the average number of tweets per user may perform slightly better. A reason for that is the hypotheses being it are more robust.

5. Besides the results in terms of accuracy, precision, recall and f1-score we conducted an analysis of the most important variables with Random Forest. It appears the volume of tweets posted could be a more important input than sentiments. An hypothesis to explain it would be that sentiment analysis remains a hard task to perform, especially in social contexts where one can mean various things with one sentence and the massive amounts of bots on social media. More simply an increase in the volume of tweets

would be linked to an increase in the interest granted to Bitcoin - maybe because of other events that only Twitter would react to - such interest being translated as an increase in price.

6. The considerable difficulty of working with real life data. The data was collected by our own means without any fees engaged. Real data is often incomplete, partial, could be bias and require a lot of processing treatments before being fed to the algorithms. Bad data is harder to overcome than a bad algorithm.

In addition to all these points, some paths to follow for further researches are provided :

- Data remains the most significant obstacle. Before getting into the analysis, more tweets could be collected and more equally distributed in time.

- In addition to Twitter there are plenty of other sources, to cite a few : Reddit, Quora, Google Trends, dedicated forums that could be scraped, Telegram channels, … Maybe more than all tweets, we should consider only the tweets posted by a small group of influential people. Indeed, the frequent impacts in 2021 of some highly popular people such as Elon Musk put under limelight the tremendous power of opinion makers, maybe up to the point of market manipulation.

- Sentiment analysis can be performed using libraries others than VADER. VADER is mainly designed for social media but not for finance or Bitcoin. A specific lexicon designed for that purpose could be better suited to perform the task.

- The techniques we've used to filter tweets are basics. A more complex clustering technique can be used to perhaps achieve better results.

- In our analysis, we did not take into account the effects on the future time horizon. For example, one hypothesis could be that Twitter sentiment influences the price of Bitcoin but with a lag of 2 hours, in which case we should not relate the price movement to the sentiment of the same time period but rather consider a lag - the best remains to be determined.

- From this work some hyper-tuning can be performed to achieve better results, especially in this kind of task where a 1% increase could result in massive profits. Also, other algorithms could be tested and even a regression analysis can be performed predicting the exact value and not just the increase or decrease, as the data is already available in our datasets.

VIII. Socioeconomic Impact

Among the great digital revolutions currently at work in the world, the Blockchain is one of the most promising. Since its theorization in 2009 following the subprime crisis, many possible applications are currently being explored (smart contract, logistics, health, ...) but its primary interest still lies in finance with Bitcoin and all other crypto currencies.

Bitcoin was created as an alternative to traditional money. It achieved a significant social impact considering it is a decentralized system not regulated by any public institution. This characteristic coupled with other advantages of cryptocurrencies, such as the possibility of making transactions directly to the receiver without the need for any intermediary (peer-to-peer network), originated an an outstanding excitement for Bitcoin that resulted in a massive of its price.

As of today many users worldwide, companies and even some states are using or placing increasing importance in Bitcoin.

Therefore, it is already a fact that crypto-currencies have brought about important changes in the global financial system, challenging an established hegemony.

In addition, the environmental impact of Bitcoin and blockchain should be mentioned. The mining process, which is at the heart of the network's security, consumes a lot of energy. It is estimated that Bitcoin requires the same amount of energy as Chile's annual electricity consumption. However, some players in the ecosystem have become aware of this and are creating new types of blockchain based not on Proof-of-Work but on Proof-of-Stake, which requires less energy.

IX. Time planning and budget

1. Project Breakdown Structure

The PBS, or Project Breakdown Structure, is a tool that hierarchically decomposes the activities carried out during the course of a project. For this work, the tasks developed are shown below in the PBS in Figure 18.

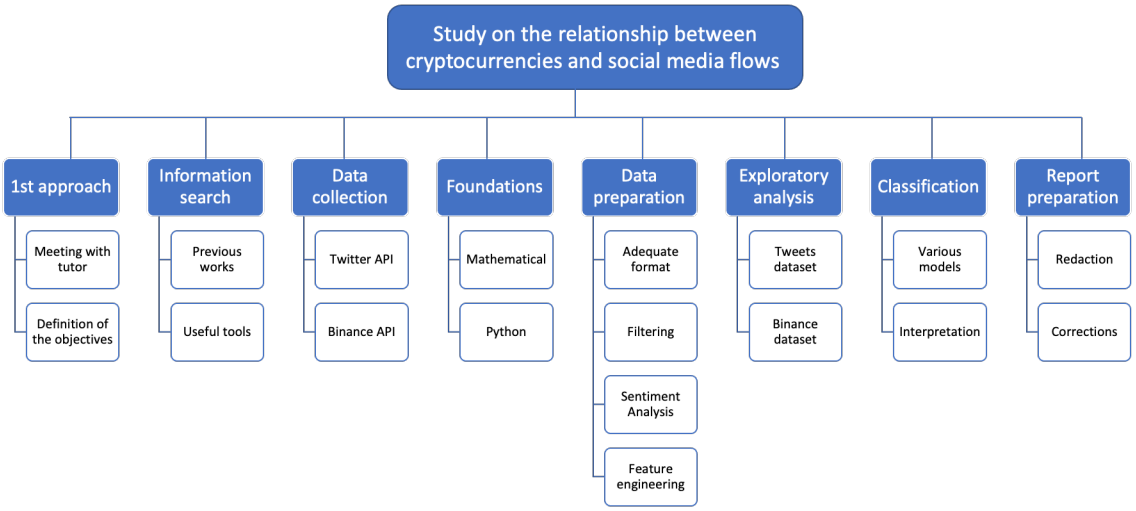

Figure 18. Project Breakdown Structure

2. Gantt chart

The Gantt Chart is a graphic tool that shows the activities of a work chronologically by means of a bar chart. The Gantt Chart of this work can be seen below in Figure 19, where the main stages are represented in dark blue and the sub-stages in light blue. These stages and sub-stages are shown in the following table as well.

Task	Start	Finish	Duration (days)
1. 1st approach	**01/04/2020**	**21/04/2020**	**21**
1.1 Meeting with tutor	01/04/2020	01/04/2020	1
1.2 Definition of the objectives	02/04/2020	21/04/2020	20
2. Information search	**22/04/2020**	**24/05/2020**	**33**
2.1 Previous works	22/04/2020	24/05/2020	33
2.2 Useful tools	22/04/2020	24/05/2020	33
3. Data collection	**25/05/2020**	**27/11/2020**	**187**
3.1 Twitter API	25/05/2020	27/11/2020	187
3.2 Binance API	25/05/2020	27/11/2020	187
4. Foundations	**28/11/2020**	**23/12/2020**	**26**
4.1 Mathematical	28/11/2020	09/12/2020	12
4.2 Python	10/12/2020	23/12/2020	14
5. Data preparation	**04/01/2021**	**20/04/2021**	**107**
5.1 Adequate format	04/01/2021	18/02/2021	46
5.2 Filtering	19/02/2021	02/03/2021	12
5.3 Sentiment analysis	03/03/2021	03/04/2021	32
5.4 Feature engineering	04/04/2021	20/04/2021	17
6. Exploratory analysis	**21/04/2021**	**14/05/2021**	**24**
6.1 Tweets dataset	21/04/2021	06/05/2021	16
6.2 Binance dataset	07/05/2021	14/05/2021	8
7. Classification	**15/05/2021**	**21/07/2021**	**68**
7.1 Various models (time frame, features, algorithms, filtering)	15/05/2021	07/07/2021	54
7.2 Interpretation	08/07/2021	21/07/2021	14
8. Report preparation	**22/07/2021**	**10/09/2021**	**51**
8.1 Redaction	22/07/2021	20/08/2021	30
8.2 Correction	21/08/2021	10/09/2021	21

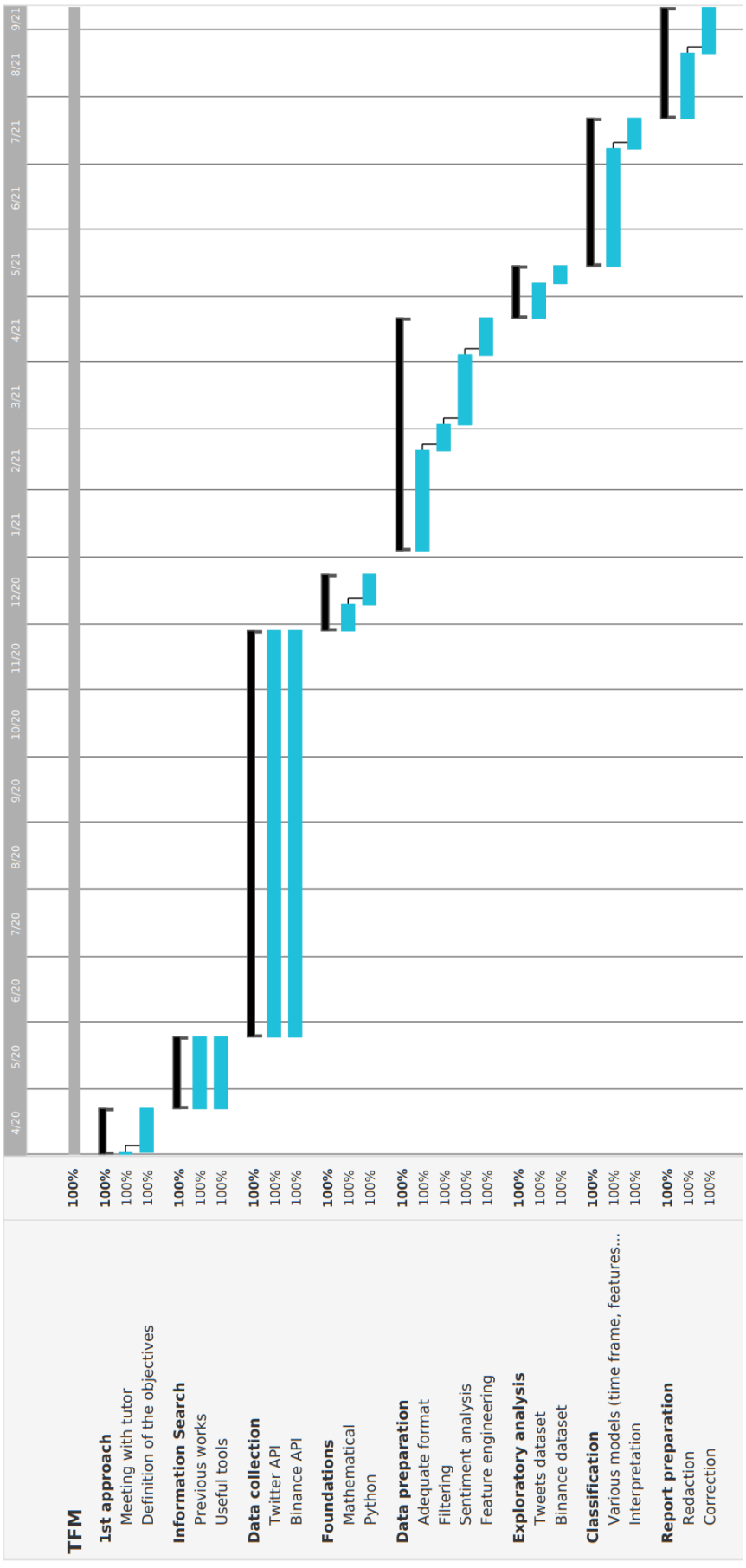

Figure 19. Gantt chart

3. Budget

This section shows the cost of the preparation of this TFM. It has been considered that the version of all the programs used does not have any license cost, as well as the Microsoft Office package, and that no material had to be purchased.
To calculate the cost of the hours dedicated to the work, it has been estimated that the salary of a graduate engineer is 20 €/hour and that of the tutor is 40 €/hour. And it has been considered that the graduate engineer has dedicated 400 hours to the work and the tutor 15 hours.

Entity	Units	Unitary cost	Total cost
Computer	1	0	0
Microsoft office	1	0	0
Jupyter	1	0	0
Twitter API	1	0	0
Binance API	1	0	0
Student salary	400 hours	20€/hour	8000 €
Tutor salary	15 hours	40€/hour	600 €
Total without taxes			8600 €
Total cost with taxes (21%)			10 406 €

Figures

[19]: Gantt chart. Source: own elaboration

References

[1]: Nebra, M. (2017). *Comprendre le bitcoin et la blockchain*, Open Classroom.
Available at https://openclassrooms.com/fr/courses/3925766-comprendre-le-bitcoin-et-la-blockchain/3925801-une-breve-histoire-des-monnaies

[2]: Nozick, R. (1974). *Anarchy, State, and Utopia*.

[3]: Harding, L. (2014). *The Snowden Files : The Inside Story of the World's Most Wanted Man*.

[4]: Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system*.

[5]: Chouli, B., Goujon, F., Leporcher, Y. (2017). *Les Blockchains*, *De la théorie à la pratique, de l'idée à l'implémentation*.

[6]: Cryptonews. *C'est quoi le minage ?* (s.f.)
Available at https://fr.cryptonews.com/guides/what-is-bitcoin-mining.htm

[9]: Digiconomist (s.f.)
Available at https://digiconomist.net/bitcoin-energy-consumption

[12]: Cryptonews. *Les avantages et les inconvénients du Bitcoin*. (s.f.)
Available at https://fr.cryptonews.com/guides/bitcoin-pros-and-cons.htm

[13]: Plus500. *What are the most traded cryptocurrencies ?* (s.f.)
Available at https://www.plus500.com/Trading/CryptoCurrencies/What-are-the-Most-Traded-Cryptocurrencies~2

[14]: Hileman, G., Rauchs, M. (2017). *Global Cryptocurrency benchmarking study*.
Available at https://www.jbs.cam.ac.uk/wp-content/uploads/2020/08/2017-04-20-global-cryptocurrency-benchmarking-study.pdf

[15]: Edosomwan, S., Prakasan, S., Kouame, D., Watson, J., Seymour, T. (2011). *The History of Social Media and its Impact on Business*.

[16]: Facebook. (s.f)
Available at https://about.facebook.com/company-info/

[17]: Twitter. (s.f)
Available at https://about.twitter.com/en/who-we-are/our-company

[18]: Smart Insights (s.f)
Available at https://www.smartinsights.com/ecommerce/social-commerce/social-commerce-trends-for-2020-you-need-to-look-out-for/

[19]: KRDSParis. (2016)
Available at https://krds.com/fr/fr/zoom-sur-le-succes-dinstagram/

[20]: Kemp, S. (2019). *Global Digital Report*.
Available at https://hootsuite.com/pages/digital-in-2019

[21]: Cadwalladr, C., Graham-Harrisson,E. (2018). *Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach*. The Guardian.
Available at https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election

[22]: (2018). *Sans Cambridge Analytica, il n'y aurait pas eu de Brexit, affirme le lanceur d'alertes Christopher Wylie*. Franceinfo.
Available at https://www.francetvinfo.fr/monde/europe/la-grande-bretagne-et-l-ue/sans-cambridge-analytica-il-n-y-aurait-pas-eu-de-brexit-affirme-le-lanceur-d-alerte-christopher-wylie_2677946.html

[23]: Joseph, S. (2018). *Why the business model of social media giants like Facebook is incompatible with human rights*. The conversation
Available at https://theconversation.com/why-the-business-model-of-social-media-giants-like-facebook-is-incompatible-with-human-rights-94016

[24]: Youyou, W., Kosinski, M. Stillwell, D. (2014). *Computer-based personality judgments are more accurate than those made by humans*.
Available at *https://www.pnas.org/content/pnas/112/4/1036.full.pdf*

[25]: Youtube, *Comment Facebook vous rend addict ? | Dopamine | ARTE* (s.f)
Available at https://www.youtube.com/watch?v=IahJWpRGbWE&ab_channel=ARTE

[26]: BBC News. (2017) *Instagram' worst for young mental health*.
Available at https://www.bbc.com/news/health-39955295

[27]: Kahneman, D., Tversky, A. (1979). *Prospect theory: An analysis of decision under risk*. Econometrica.

[28]: Tetlock, P.C. (2007). *Giving content to investor sentiment: The role of media in the stock market*. The Journal of Finance.

[29]: de Jong, P., Elfayoumy, S., Schnusenberg, O. (2017). *From returns to tweets and back: An investigation of the stocks in the dow jones industrial average*. Journal of Behavioral Finance.

[30]: Lamon, C., Nielsen, E., Redondo, E. (2015). *Cryptocurrency price prediction using news and social media sentiment*. Master's thesis, Stanford

[31]: Jethin, A., Higdon, D., Nelson, J., Ibarra, J. (2018). *Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis*. SMU Data Science Review.

[32]: Géron, A. (2017). *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*.

[33]: Pedregosa *et al*. (2011). *Scikit-learn: Machine Learning in Python*.

[34]: Twitter API (s.f)

Available at https://developer.twitter.com/en/docs/twitter-api

[35]: Tweepy. (s.f)
Available at https://www.tweepy.org/

[36]: Binance API. Git Hub. (s.f)
Available at https://github.com/binance/binance-spot-api-docs/blob/master/rest-api.md

[37]: ProjectPro (2021).
Available at https://www.dezyre.com/article/why-data-preparation-is-an-important-part-of-data-science/242

[38]: Varol *et al*. (2017). *Online Human-Bot Interactions: Detection, Estimation, and Characterization*.
Available at https://arxiv.org/abs/1703.03107

[39]: Hutto, C.J., Gilbert, E.E. (2014). VADER: *A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*.

[40]: Vader. Git Hub. (s.f)
Available at https://github.com/cjhutto/vaderSentiment

Annexes