Project Title: Investigating the connection between sentiments from bitcoin tweets and the value of bitcoin.

Group Members:
Eric Edgari - 2301902352
Jocelyn Thiojaya - 2301900454

1. Problem introduction and hypothesis

As the financial literacy of people is getting better, people are starting to think about investment to increase their financial stability. Investment can be done in numerous ways such as property, stock, gold, currency, etc. One of the trending investments right now are cryptocurrencies. People who invest like to survey the market trend or market sentiment from news or tweets. Reading one of the bad tweets doesn't justify the market sentiment regarding the cryptocurrencies being bad. But reading all of the tweets and analyzing it manually would take a long time and be very taxing. We want to solve the problem by having a machine analyze the tweets and summarize the current trend.

We also take notes from past experience on how tweets affect the price. Billionaires like Elon Musk invest in cryptocurrencies therefore making it more popular. A lot of people follow his footsteps and start to invest in cryptocurrencies. From past experience, a tweet from Elon Musk on February 2021 spiked the price for bitcoin from $40.000 to $45.000. With this past experience, we started to develop our question. The question that we will try to answer is, do tweets regarding bitcoin affect its value?

Our project was initially inspired by a kaggle topic that has been done regarding the influence of tweets on the stock market. We see that cryptocurrencies are trending right now, so we want to implement it on it.
Kaggle topic link : https://www.kaggle.com/renjithrrkj/influence-of-tweets-on-stock-martket/notebook

2. Dataset

2.1. Bitcoin Tweets Dataset

The Bitcoin Tweets Dataset contains the data of tweets using the hashtag #Bitcoin and #btc. Contents of tweets using this hashtag are related to bitcoin. The source of this dataset is Kaggle, https://www.kaggle.com/kaushiksuresh147/bitcoin-tweets.

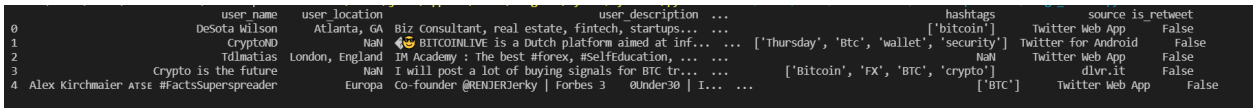
Collection of these tweets began from 10th February 2021 until 29th October 2021. There are 13 columns and 1793124 rows in this dataset. Below is a table describing each column and what it contains.

No.	Column Name	Descriptions	Data type
1	user_name	The name of the user, as they've defined it.	String
2	user_location	The user-defined location for this account's profile.	String
3	user_description	The user-defined UTF-8 string describing their account.	String

4	user_created	Time and date, when the account was created.	datetime
5	user_followers	The number of followers an account currently has.	float
6	user_friends	The number of friends an account currently has.	float
7	user_favorites	The number of favorites an account currently has.	float
8	user_verified	When true, indicates that the user has a verified account.	boolean
9	date	UTC time and date when the Tweet was created.	datetime
10	text	The actual UTF-8 text of the Tweet.	String
11	hashtags	All the other hashtags posted in the tweet along with #Bitcoin & #btc.	String array
12	source	Utility used to post the Tweet, Tweets from the Twitter website have a source value - web	String
13	is_retweet	Indicates whether this Tweet has been Retweeted by the authenticating user.	boolean

The following is a screenshot with information regarding this dataset, along with a snippet of its contents.

```
RangeIndex: 1793124 entries, 0 to 1793123
Data columns (total 13 columns):
 #   Column            Dtype
---  ------            -----
 0   user_name         object
 1   user_location     object
 2   user_description  object
 3   user_created      object
 4   user_followers    float64
 5   user_friends      object
 6   user_favourites   object
 7   user_verified     object
 8   date              string
 9   text              object
 10  hashtags          object
 11  source            object
 12  is_retweet        object
dtypes: float64(1), object(11), string(1)
memory usage: 177.8+ MB
```



	user_name	user_location	user_description	...	hashtags	source	is_retweet
0	DeSota Wilson	Atlanta, GA	Biz Consultant, real estate, fintech, startups...	...	['bitcoin']	Twitter Web App	False
1	CryptoND	NaN	🤓 BITCOINLIVE is a Dutch platform aimed at inf...	...	['Thursday', 'Btc', 'wallet', 'security']	Twitter for Android	False
2	Tdlmatias	London, England	IM Academy : The best #forex, #SelfEducation, ...	...	NaN	Twitter Web App	False
3	Crypto is the future	NaN	I will post a lot of buying signals for BTC tr...	...	['Bitcoin', 'FX', 'BTC', 'crypto']	dlvr.it	False
4	Alex Kirchmaier ATSE #FactsSuperspreader	Europa	Co-founder @RENJERJerky | Forbes 3 0Under30 | I...	...	['BTC']	Twitter Web App	False

To get a better overview of our data on tweets , we also plot the dataset on tableau. We use a line graph for the following tweets dataset. What we plot is the the date on x axis and the number of tweets on y axis. An interesting finding that we notice is that, february 8th , the tweets reached a total count of 5647 and on march 11th it dropped down to 217 tweets.

2.2 Bitcoin Price Dataset

The Bitcoin Price Dataset contains the data of bitcoin prices at every minute. The source of this dataset is Kaggle, https://www.kaggle.com/aakashverma8900/bitcoin-price-usd.

Collection of these bitcoin prices began from 1st January 2021 to 12th May 2021. There are 11 columns and 188317 rows in this dataset. Below is a table describing each column and what it contains.

No.	Column Name	Descriptions	Data type
1	Open Time	Open Time (Unix Timestamp).	integer
2	Open	Open Price of a particular minute.	float
3	High	High Price of a particular minute.	float
4	Low	Low Price of a particular minute.	float
5	Close	Close Price of a particular minute.	float
6	Volume	Total Volume of a particular minute.	float
7	Close Time	Close Time (Unix Timestamp).	integer
8	Quote asset volume	Quote asset volume.	float

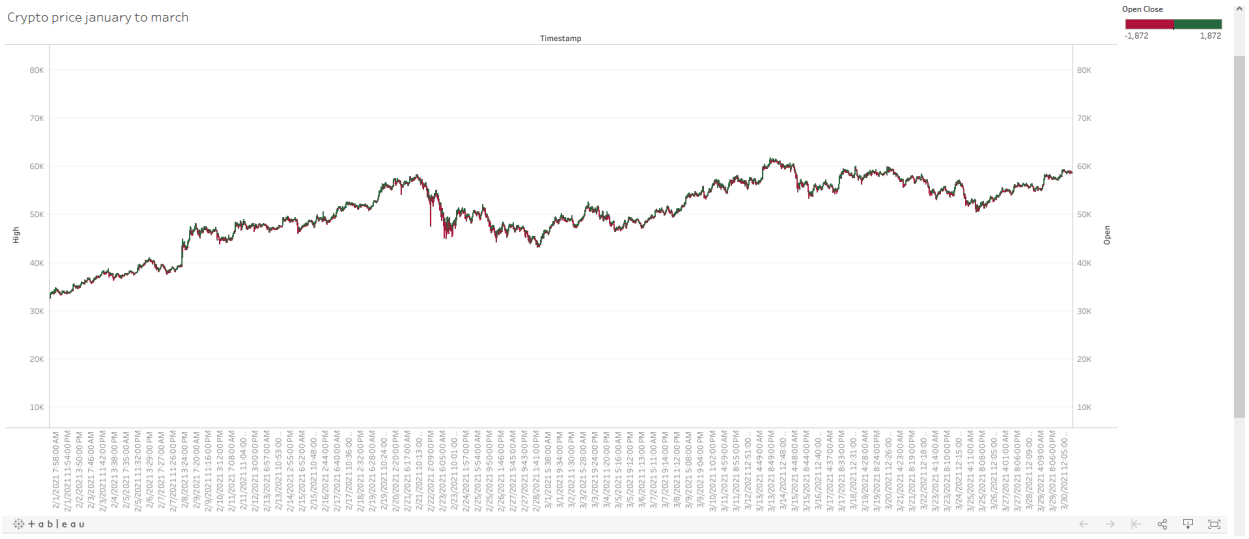
9	Number of trades	Number of trades for a particular minute.	integer
10	Taker buy base asset volume	Taker buy base asset volume.	float
11	Take buy quote asset volume	Taker buy quote asset volume.	float

The following is a screenshot with information regarding this dataset, along with a snippet of its contents.

```
RangeIndex: 188317 entries, 0 to 188316
Data columns (total 11 columns):
 #   Column                        Non-Null Count   Dtype
---  ------                        --------------   -----
 0   Open Time                     188317 non-null  int64
 1   Open                          188317 non-null  float64
 2   High                          188317 non-null  float64
 3   Low                           188317 non-null  float64
 4   Close                         188317 non-null  float64
 5   Volume                        188317 non-null  float64
 6   Close Time                    188317 non-null  int64
 7   Quote asset volume            188317 non-null  float64
 8   Number of trades              188317 non-null  int64
 9   Taker buy base asset volume   188317 non-null  float64
 10  Taker buy quote asset volume  188317 non-null  float64
dtypes: float64(8), int64(3)
memory usage: 15.8 MB
```

	Open Time	Open	High	Low	Close	Volume	Close Time	Quote asset volume	Number of trades	Taker buy base asset volume	Taker buy quote asset volume
0	1609459200000	28923.63	28961.66	28913.12	28961.66	27.457032	1609459259999	7.943820e+05	1292	16.777195	485390.826825
1	1609459260000	28961.67	29017.50	28961.01	29009.91	58.477501	1609459319999	1.695803e+06	1651	33.733818	978176.468202
2	1609459320000	29009.54	29016.71	28973.58	28989.30	42.470329	1609459379999	1.231359e+06	986	13.247444	384076.854453
3	1609459380000	28989.68	28999.85	28972.33	28982.69	30.360677	1609459439999	8.800168e+05	959	9.456028	274083.075142
4	1609459440000	28982.67	28995.93	28971.80	28975.65	24.124339	1609459499999	6.992262e+05	726	6.814644	197519.374888

To get a better overview of our data on cryptocurrencies price , we also plot the dataset on tableau. We use a standard gant bar for this following the normal visualization used on cryptocurrency price with fields such as high, open, high low, and open close price. Another interesting finding corresponds to the tweets datasets, on 8 February the dataset consisted of 5647 tweets, the highest in that month. Coincidently the bitcoin price spiked to $46.000 from $38.717 with an increase of 18% from its initial value. This corresponds with our question from the introduction whether the tweets are affection the bitcoin price or not.



3. Data preparation and processing

There are a few steps to prepare our data. The first step is to drop unwanted columns in the Bitcoin Tweets Dataset. The dropped columns are “user_name”, “user_location”, “user_description”, “user_created”, “user_friends”, “source”, “hashtags”, and “is_retweet”. The reason for dropping these columns is to reduce the number of factors that are being taken into account, especially because these data do not have a huge impact towards the overall sentiment of the tweet. Next, unwanted columns in the Bitcoin Price Dataset are dropped. The only dropped column is “Close Time” because we do not use it, we are only using the “Open Time” column to compare with the Bitcoin Tweets’ date and time.

The second step is to drop unwanted rows of our data. The Bitcoin Price Dataset starts from 1st January 2021 to 12th May 2021, so we will only be using the tweets from that date range only. All tweets before 1st January 2021 and after 12th May 2021 can be removed.

The third step is to clean and edit some of our data. The Bitcoin Tweets Dataset had 7 strange data in the “date” column, which contained text instead of a date and time. We find and remove these 7 rows with text in the “date” column.

The fourth step is to convert the “text” column of the Bitcoin Tweets Dataset with their respective sentimental value. The data type of this column becomes float. We do this because we are comparing the sentimental value of a tweet text towards the price of bitcoin.

The fifth step is to merge the Bitcoin Tweets Dataset with the Bitcoin Price Dataset. This is because we would like to study the sentiment value of the tweet with the price of bitcoin at that moment. The merge will be in the “date” column for Bitcoin Tweets with the “Open Time” column for Bitcoin Price. Since the “date” column for Bitcoin Tweets uses the datetime data type, and the “Open Time” column for Bitcoin Price uses the float data type to write Unix time, we will convert Unix time to datetime format in the Bitcoin Price dataset.

The details of the finalized dataset are listed below.

No.	Column Name	Descriptions	Data type
1	user_followers	The number of followers an account currently has.	float

2	user_favorites	The number of favorites an account currently has.	float
3	user_verified	When true, indicates that the user has a verified account.	boolean
4	date	UTC time and date when the Tweet was created.	datetime
5	text	The actual UTF-8 text of the Tweet.	String
6	Open Time	Open Time (Unix Timestamp).	integer
7	High	High Price of a particular minute.	float
8	Low	Low Price of a particular minute.	float
9	Volume	Total Volume of a particular minute.	float
10	Quote asset volume	Quote asset volume.	float
11	Number of trades	Number of trades for a particular minute.	integer
12	Taker buy base asset volume	Taker buy base asset volume.	float
13	Take buy quote asset volume	Taker buy quote asset volume.	float

4. Model and techniques

In order to conduct this research, we are going to use several different techniques such as sentimental analysis, random forest, decision tree, and deep tabular data learning. We would like to use the specified techniques because it allows us to determine and classify which factors play a significant role in the impact of bad tweets on bitcoin price. We then compared each technique's results and picked the best one.

We are going to use Python as a programming language as it has a lot of packages that allows the researchers to do data processing and visualization in a quick and easy way. Packages such as NumPy, Pandas, Matplotlib, and VaderSentiment will be used in this research. We also have an alternative sentimental analysis if Vadersentiment are not quite to our fit which is the MonkeyLearn API.

VADER Sentiment Analysis, or VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media, and works well on texts from other domains. We expect to retrieve the compound score of each tweet using VaderSentiment. The compound score is computed by summing the valence scores of each

word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). This is the most useful metric to retrieve a single unidimensional measure of sentiment for a given sentence. It is a normalized, weighted composite score. It is useful for researchers who would like to set standardized thresholds for classifying sentences as either positive, neutral, or negative. Typical threshold values (used in the literature cited on this page) are:

- positive sentiment: compound score >= 0.05
- neutral sentiment: (compound score > -0.05) and (compound score < 0.05)
- negative sentiment: compound score <= -0.05

We also pick a couple of model from our technique. We will try to train the model and compare the results for all of them. Two of our models are based on decision trees. A decision tree is a simple, decision making-diagram. It is a sequential steps designed to answer a question and provide probabilities, costs, or other consequence of making a particular program. They are simple to understand. However they're simplicity comes with disadvantages such as overfitting, bias, and variance error.

- Overfitting could happen with reason such as noise and lack of representative of instances
- Bias error happens when we place too many restrictions on a function such as a simple binary algorithm where the output is true or false, This can lead the result to become biased.
- Variance error, decision tree have high variance, which means that a slight change to our training dataset can have a large impact on our final result.

After knowing the decision flaws, we decided not to use it. We decided to use the next level version of the decision tree which is random forest and gradient boost decision tree. These are the current state of the art technology that could solve our classification problem.

List of models:

- Random Forest
  - What makes random forest differ from decision tree is that random forest generate a large number of decision tree hence it called random forest. The many decision trees will average out their solution by the algorithm in order to get an answer close to the true one. This averaging out output from each decision tree are being done in order to reduce the variance in a normal decision tree. This process is done using different samples for training, Specifying random feature subsets, and building and combining small trees. However the cons of this algorithm is that by making a lot of decision trees, we will have a slower process.
- XGBoost (based on gradient boost decision tree)
  - XGBoost is another state of the art implementation of decision trees . It is different from the random forest based on two factors which is how trees are built and the combining results for the trees. Random forests build each tree independently while gradient boosting builds one tree at a time. The combining result also follow the same mechanism,

where random forests combine results at the end of the process, but gradient boost decision tree combine results along the way.

- TabNet (Deep tabular data learning architecture)
    - Tabnet is a deep learning model that could be used for our problem which is tabular data classification. In its encoder, sequential decision steps encode features using sparse learned mask and select relevant features using the mask. By using sparsemax layers , the encoder forces the selection of a small set of features.

We decided to try the state of the art deep learning architecture to test the new technologies for the tabular data. Usually people will use decision tree based algorithm for tabular data, but since deep learning are becoming more famous, people are trying to use deep learning technologies for tabular data. In recent research comparing decision tree based machine learning and deep learning, the results are actually quite interesting. Some of the dataset performed well on decision tree-based machine learning while others performed better in deep learning. We want to compare our result and see which one performed better for our dataset.