OPEN-HIGH-LOW-CLOSE, Blockchain and Social Cryptocurrency Data Analysis

Cryptocurrency Market today counts a market capitalization of $207 Billion, with more than 3000 coins and where main dominant Cryptos, as BTC, ETH, LTC have reached a clear popularity on Social Networks such Twitter, Facebook, Reddit and GitHub. Hence, it is today an important financial reality that attracts a lot of risk lovers and digital coin users. Nevertheless, the ambiguity of Market Nature and the huge volatility makes this market complex and approach to Cryptocurrency analysis a stiff process.

It looks distant from exchange market, which appears stable and with low volatility level, and appears more similar with stock. Both in fact, present high degree of risk, but Crypto market results more fragile. All this makes price forecasting an interesting and complex game.

Looking at the actual State of Art, the most interesting trend is the application of several machine learning algorithms, such as simple and multiple Linear Regression, Support Vector machine (SVM), Multilayer Perceptron (MLP) to OHCLV financial data.

But the lack of seasonality and the continuous volatility drastically afflict models accuracy. Throughout the recent years, Sentiment Analysis has been involved into the Cryptocurrency price forecasting. It is a tool, based on Opinion Mining and Natural Processing Language that allows extracting polarity from Social Posts and Text, a good proxy of investor Sate of Confidence about Market. Most of works consider just Twitter sentiment and Google Trend with daily data sampling frequency.

Today, few papers have inferred on Blockchain quantitative features as possible Price spread explanatory variables. Blockchain is the most underlying cryptocurrency technology and it is definable as a distributed, immutable and transparent ledger that allows emitting transactions stored by blocks. This innovation paradigm is impacting on several business areas, as Financial Transactions, Supply Chain and Politic, with a hype expectation that is touching the stars.

The scope of this work, is to explore the main Cryptocurrency Sources, and evaluating which kind of data is offered, with which granularity and time horizon and in which ways (REST APIs, Web Socket APIs, csv, excel adds-on).

Under this perspective, three kinds of data are stored: the OHLCV (Open, High, Low, Close, Volume) financial data, Social Data, including Facebook likes, Reddit posts, comments, GitHub activity and Blockchian data, as Block size in Byte, the number of Transactions, the Difficulty to add a new Block, the Miners Remuneration in USD.

Once Data Crawling is reached, the work proceeds inferring on the existence of possible correlation between financial data and Social and Blockchain data.
Finally, in order to empirically evaluate the validity of the work done so far, a Multilayer Perception, a Neural Network algorithm, is rune. The forecasting performances are analyzed, computing the Mean Square Error.

1. *DATA CRAWLING*

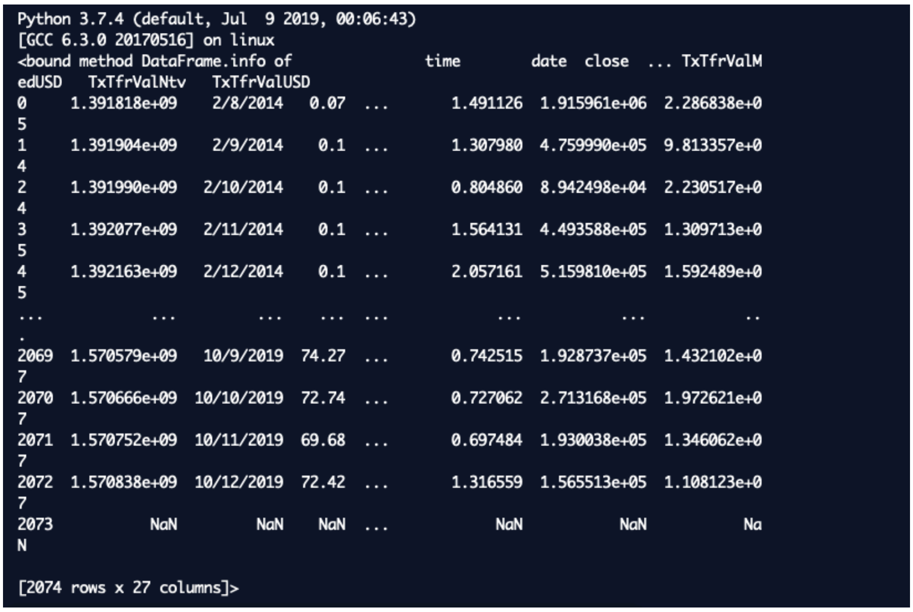
The crawling process output is the following. OHCLV data is collected both with hourly and daily granularity. The former is aligned with Social Data that have constant timeframe for ach crypto. The latter is aligned with Blockchain Data that presents different timespan dimension. For most of them it has been possible to store data back to its quasi market emission.
The crypto taken into account and respective timespan are presented in the table below.

Table 3.2: It shows the Time Horizon taken into account for each coin and for both Social and Blockchain data

Cryptocurrency (SYMBOL)	Social data (Hourly)	Blockchain Data (Daily)
Bitcoin (BTC)	1/1/19, 0.00 12/10/19, 11.00	17/07/10 12/10/19
Dash (DASH)	1/1/19 0.00 12/10/19 11.00	08/02/14 12/10/19
Ethereum (ETH)	1/1/19 0.00 12/10/19 11.00	07/08/15 12/10/19
Litecoin (LTC)	1/1/19 0.00 12/10/19 11.00	24/10/13 12/10/19
Monero (XMR)	1/1/19 0.00 12/10/19 11.00	2015-01-29 12/10/19

Calling dataframe.info () function it is possible to obtain information about the number of entries and attributes of specific dataframe.

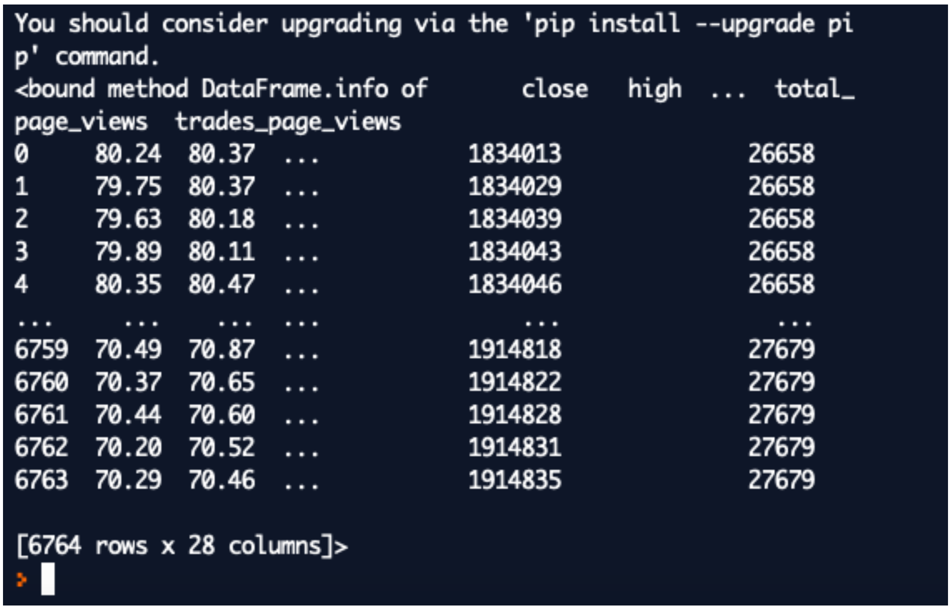
Applying it to hourly Social and daily Blockchain dataframes, for DASH coin, the result is in the Fig 3.2 and Fig 3.3



```
Python 3.7.4 (default, Jul  9 2019, 00:06:43)
[GCC 6.3.0 20170516] on linux
<bound method DataFrame.info of               time       date  close  ... TxTfrValM
edUSD   TxTfrValNtv   TxTfrValUSD
0     1.391818e+09    2/8/2014   0.07  ...      1.491126  1.915961e+06  2.286838e+0
5
1     1.391904e+09    2/9/2014    0.1  ...      1.307980  4.759990e+05  9.813357e+0
4
2     1.391990e+09   2/10/2014    0.1  ...      0.804860  8.942498e+04  2.230517e+0
4
3     1.392077e+09   2/11/2014    0.1  ...      1.564131  4.493588e+05  1.309713e+0
5
4     1.392163e+09   2/12/2014    0.1  ...      2.057161  5.159810e+05  1.592489e+0
5
...            ...         ...    ...  ...           ...           ...          ..
.
2069  1.570579e+09   10/9/2019  74.27  ...      0.742515  1.928737e+05  1.432102e+0
7
2070  1.570666e+09  10/10/2019  72.74  ...      0.727062  2.713168e+05  1.972621e+0
7
2071  1.570752e+09  10/11/2019  69.68  ...      0.697484  1.930038e+05  1.346062e+0
7
2072  1.570838e+09  10/12/2019  72.42  ...      1.316559  1.565513e+05  1.108123e+0
7
2073           NaN         NaN    NaN  ...           NaN           NaN          Na
N

[2074 rows x 27 columns]>
```

Fig 3.2: It shows the dimension of daily Blockchain and financial dataframe for DASH coin. It counts 2074 daily entries and 27 attributes. The time horizon depend on specific coin, hence the entries varies for each coin.



```
You should consider upgrading via the 'pip install --upgrade pi
p' command.
<bound method DataFrame.info of      close   high  ...  total_
page_views  trades_page_views
0     80.24  80.37  ...         1834013           26658
1     79.75  80.37  ...         1834029           26658
2     79.63  80.18  ...         1834039           26658
3     79.89  80.11  ...         1834043           26658
4     80.35  80.47  ...         1834046           26658
...     ...    ...  ...             ...             ...
6759  70.49  70.87  ...         1914818           27679
6760  70.37  70.65  ...         1914822           27679
6761  70.44  70.60  ...         1914828           27679
6762  70.20  70.52  ...         1914831           27679
6763  70.29  70.46  ...         1914835           27679

[6764 rows x 28 columns]>
> 
```

Fig 3.3: It shows the dimension of hourly Social and financial dataframe for DASH coin. It counts 6764 hourly entries and 28 attributes. The time horizon in this case, is constant for each coin.

The hourly dataframe contains OHLCV data and the following Social metrics:

- **analysis_page_views**: It counts the number of views for Analysis CryptoCompare.com page;
- **charts_page_views**: It provides statistics about charts page view from CryptoCompare.com;
- **code_repo_closed_issues**: It counts the times that a repository related to specific coin is closed on GitHub community. Generally, it comes through the syntax Close or Fix followed by the number of Issue. For example, to close the issue number 200, just needs the phrase "Closes#200" in the pull request description;
- **code_repo_forks**: It takes into account the time where a new copy of a repository for specific coin project is produced. By the way, a fork is a copy of a repository, that allows to experiment with changes without afflicting the original project;
- **code_repo_stars**: It counts the number of new repository for specific coin on GitHub;
- code_repo_subscribers: It takes into account the number of subscribers for specific coin developing activity project on GitHub;
- **fb_comments**: It insights the number of Facebook Comment under Coin Posts;
- **fb_likes**: It reports the number of Facebook likes on Coin Posts;
- **followers**: It defines the number of followers of Facebook coin page;
- **forum_page_views**: It counts the number of views for Coin forum page on CryptoCompare.com;
- **influence_page_views**: It counts the number of views for most influencing Coin news;
- **markets_page_views**: It counts the number of views for Coin market page on CryptoCompare.com;
- **overview_page_views**: It counts the number of views for Coin overview page on CryptoCompare.com;
- **forum_posts**: It counts the number of posts for Coin forum page on CryptoCompare.com;
- **reddit_active_users**: It insights the number of active users for coin subreddit on Reddit community;
- **reddit_comments_per_hour**: It insights the number of hourly comments for coin subreddit on Reddit community;
- **reddit_posts_per_hour**: It defines the number of hourly posts for coin subreddit on Reddit community;

- **reddit_subscribers**: It counts the number of hourly subscribers for coin subreddit on Reddit community;
- **total_page_views**: It returns the sum of views from all coin pages taken into account;
- **trades_page_views**: It defines the number of views of Coin trading pages on CryptoCompare.com;

The daily dataframe contains OHLCV data and the following Blockchain metrics:

- **AdrActCnt**: It shows the sum count of unique addresses that were active in the that day. Individual addresses are not double-counted if previously active;
- **BlkCnt**: It defines the sum count of blocks created that day that were included in the chain;
- **BlkSizeMeanByte:** It gives mean size in bytes of all blocks created that day.
- **DiffMean** The mean difficulty of finding a hash that meets the protocol-designated requirement (i.e., the difficulty of finding a new block) that day. This is a proxy of how difficult is to find a new block that day;
- **FeeMeanUSD**: It gives the USD value of the mean fee per transaction that day;
- **FeeTotUSD**: It defines the sum USD value of all fees paid to miners that day;
- **IssTotUSD:** The sum USD value of all new native units issued that day;
- **NVTAdj** The ratio of the network value (or market capitalization, current supply) divided by the adjusted transfer value. Also referred to as NVT;
- **SplyCur**: It defines the sum of all native units ever created and visible on the ledger of that day. For account-based protocols, only accounts with positive balances are counted.
- **TxCnt** It insights the sum count of transactions that day. Transactions represent a bundle of intended actions to alter the ledger initiated by a user (human or machine). Transactions are counted whether they execute or not and whether they result in the transfer of native units or not (a transaction can result in no, one, or many transfers);
- **TxTfrCnt** The sum count of transfers that day. Transfers represent movements of native units from one ledger entity to another distinct ledger entity. Only transfers that are the result of a transaction and that have a positive (non-zero) value are counted.
- **TxTfrValAdjNtv** The sum of native units transferred that day removing noise and certain artifacts.
- **TxTfrValAdjUSD** The USD value of the sum of native units transferred that day removing noise and certain artifacts.

- **TxTfrValMeanNt**v The mean count of native units transferred per transaction (i.e., the mean "size" of a transaction) that day.
- **TxTfrValMeanUSD:** The sum USD value of native units transferred divided by the count of transfers (i.e., the mean "size" in USD of a transfer) that day.
- **TxTfrValMedNtv:** The median count of native units transferred per transfer (i.e., the median "size" of a transfer) that day.
- **TxTfrValMedUSD**: It is the median count of USD value of native transfer that day
- **TxTfrValNt**v: The sum of native units transferred (i.e., the aggregate "size" of all transfers) that day. Hence it is a proxy of the aggregate size of all transfers that day;

2. PREPROCESSING AND DATA ANALYSIS

Once the data is stored, preprocessing step is required so that bias in Data interpretation and Modelling are avoided. In this Chapter, data is cleaned from outliers and null values, replaced with the median.

Concluded this phase, HeatMap correlation Matrix is generated for Social and Blockchain dataframes, in order to insight on the main correlated variables with the Closing Price of that coin in the next time instant (t+1). These steps are crucial for the final experiment.

Once cleaned dataset, it is important to evaluate the possible correlation across different variables. In particular, the scope of this work is to evaluate possible relationship between the Closing Price at the next time instant t+1 and the different attributes at the time instant t.

In simple terms, what it has been verified is the possibility of existing relationships between financial time series with Social and Blockchain variables.

In order to do this, HeatMap correlation matrices are generated for each coin, so that it is possible to infer on the existence of correlations. The following lines code display how HeatMap Pearson Correlation Matrices are built for Ethereum coin, through MatPlotlib Python library.

ETH Blockchain and Social HeatMaps, generated with MatPlotlib, are reported below:

Fig 4.1: Blockchain ETH MatplotLib HeatMap

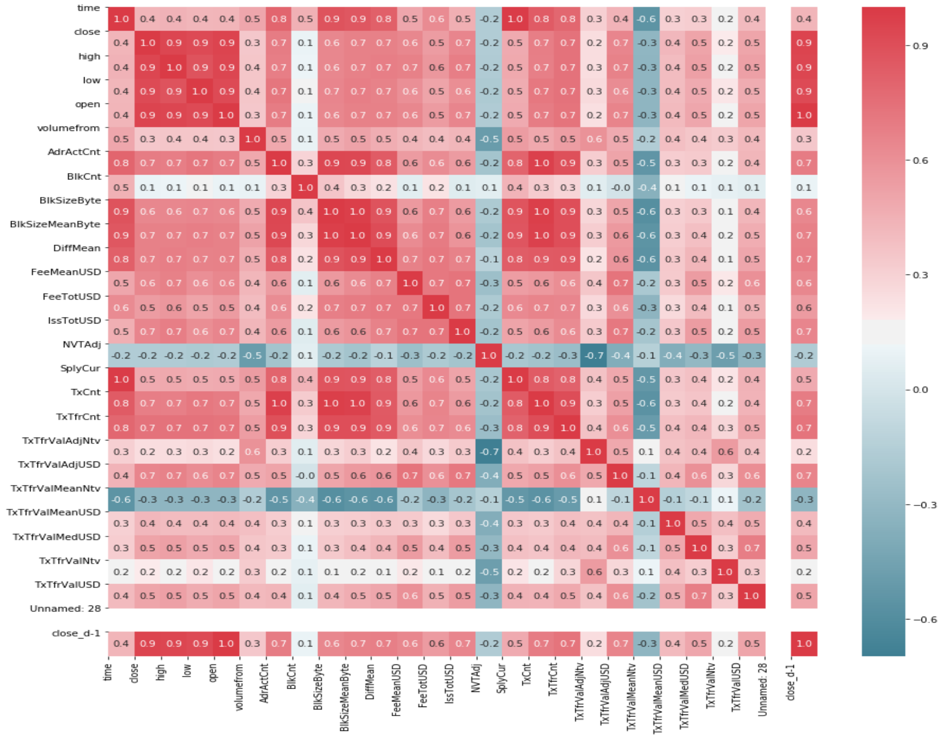
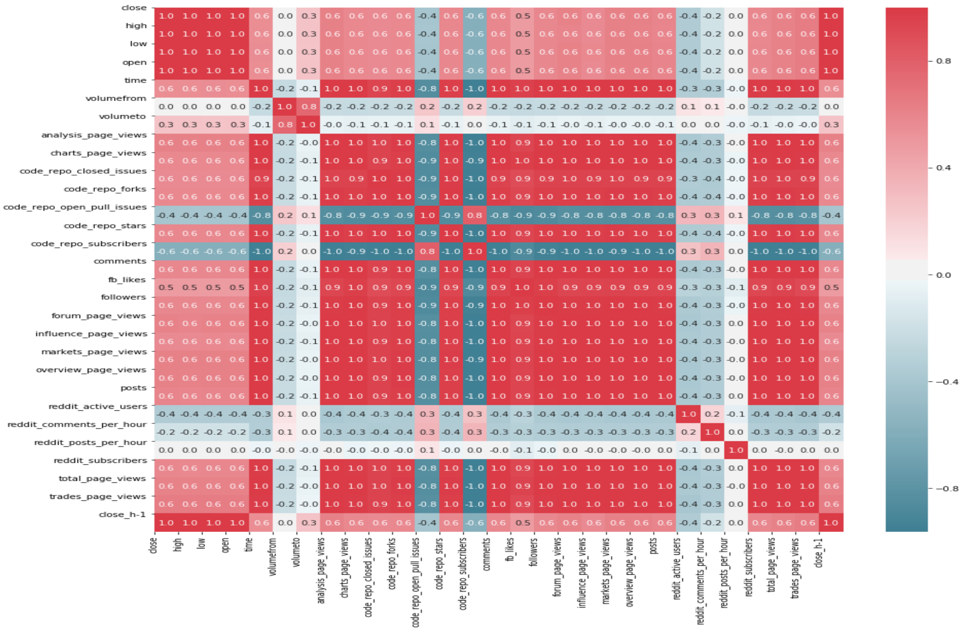


Fig 4.2: Social ETH MatplotLib HeatMap



The Pearson correlation presents value in the range of [-1 , +1], where -1 indicates the presence of maximum negative correlation, and +1 the maximum positive one. Positive means that an increase of one variable implies an increase in the correlated one. Contrary, an increase in one variable implies a decrease in the other, if negative correlated.

Zero shows the absence of correlation between the variables, while intermediate positive values between 0 and 1 shows a positive gradient of correlation, negative in the case of intermediates values between]-1,0[.

The python function HetaMap.Corr(db) uses this pictorial representation that allows identifying the most correlated variable, through Colors. The intense red insights the maximum correlation, whereas, most strength grey implies strength negative correlation.

Both Social and Blockchain HeatMaps shows correlation between Closing price at t+1 and High, Low, Close and Open at the time instant t. It is not a surprise, Financial Data Presents an obvious correlation, and what we expected is a top linkage between closing price at t+1 and closing at t, because of they are similar value. Closing at t+1 is a new column added to dataset, simply shifting of one entry the Closing price. It is important to notice that, due to the implementation of different granularity in the two dataframes, it will be used Closing d+1, when referring on Blockchian dataset and Closing h+1, when referring to Social data, reflecting the daily and hourly sampling frequency.

There is not a clear relationship with volume; in the time series with hour granularity they show a Pearson correlation of 0.3, whereas in 0.2 in the daily dataset.

An interesting result is the strong pattern defined frim close d+1 and Block metrics. The correlations with DiffMean, FeeMeanUSD and TxTfrValMedUSD are 0.7, a good level of positive correlation.

In order to better understand, what positive relationship means, it is possible to plot correlated time series. Data shows different dimension and unit measure; hence data normalization is required, assuming the normality distribution of the data, under Central Limit Theorem hypowork. Doing this, StandardScaler () function is used, once preprocessing module is imported from sklearn library.

The following MatplotLib plots (Fig 4.1 and 4.2) compare the Ethereum Closing price at t+1 with FeeMeanUSD and TxTfrValMedUsd. The former defines the value of the mean fee per transaction that day, whereas the latter the mean size of a transfer that day.

Both variables provide a good degree of correlation and how pictured in the plots below, they follow the price movements with limited lag. However, also in the timespan with high volatility, these features appear able to react and to strike a pose, comparable with the price oscillations.

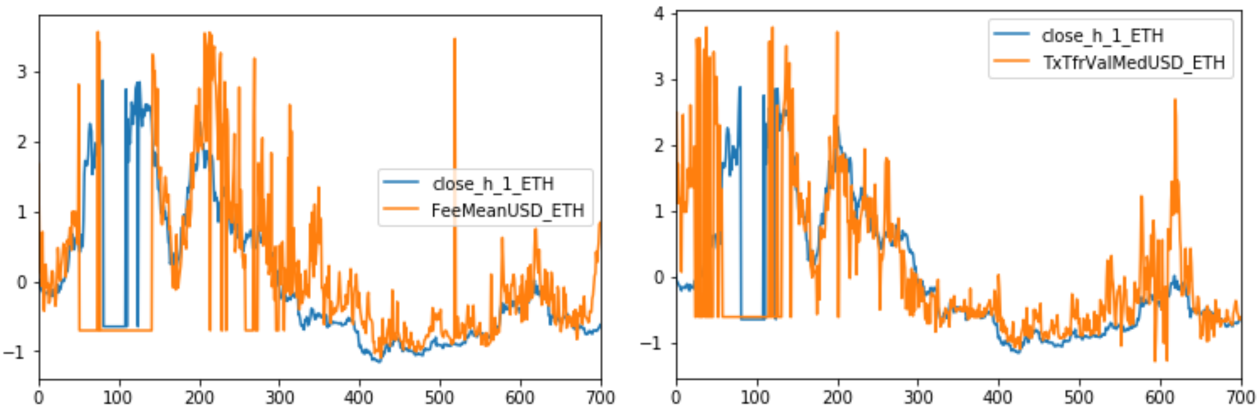


Fig 4.2 and Fig 4.3: ETH MatplotLib plots, where Block variables are compared with Closing Eth at t+1 in the last 700 days

Different is the relationship between Social Variables and Closing ETH. In the first plot, price is compared with hourly Reddit posts about Ethereum, whereas the second reports the relationship between price and Official Ethereum posts' likes.

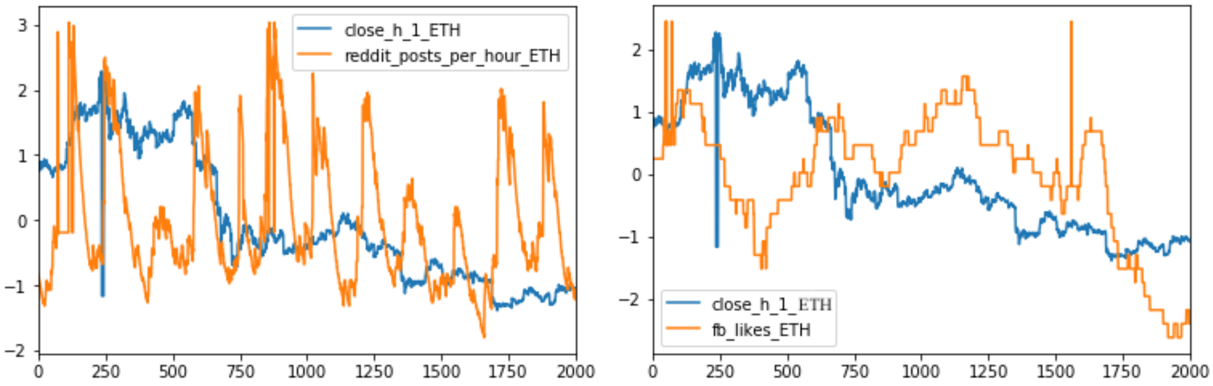


Fig 4.4 and Fig 4.5: ETH MatplotLib plots, where Social variables are compared with Closing Eth at t+1, in the last 2000 hours

The hourly Reddit posts movement follows a completely Random Walk and does not show meaningful signal about price. Facebook likes, instead, show a more similar movement, but a price oscillation is overrated. When the price goes up, the sentiment appears over expected, whereas, when it goes down, sentiment drastically falls.

BTC Close (t+1) dilspay a different behaviour, how observable in the below charts. Social data, in this case strongly afflicts BTC Financial Time Series, most of the attributes show a Person correlation value of 0.9. Code_repo_subscribers showes in the selected time horizon of Fig 4.5 a clear movments commonality, and just in few time instants there is a reaction lag.

Facebook likes (fb_likes_BTC), presents good level of positive correlation as well; but in how illustrated in Fig 4.6, Social Features do not proportionally reflect the Price peaks.

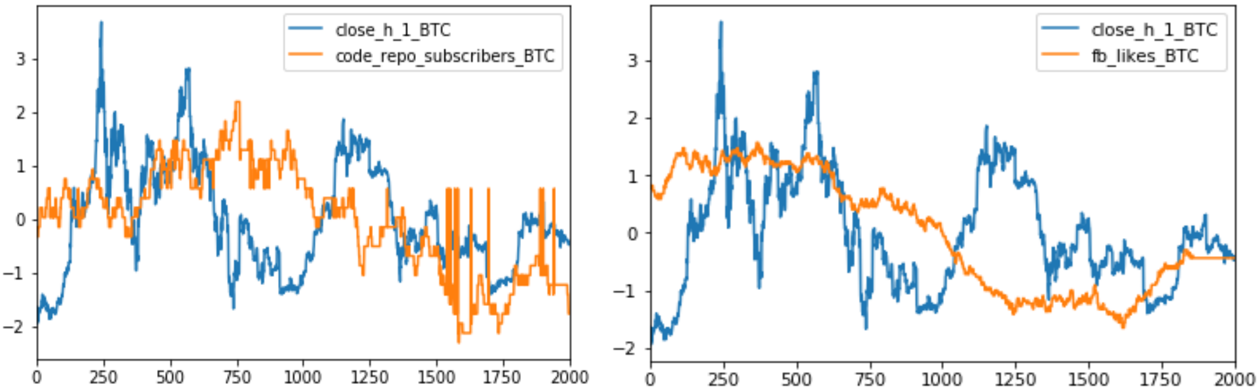


Fig 4.6 and Fig 4.7: BTC MatplotLib plots, where Social variables are compared with Closing Bth at t+1 from 1/7/19 2.00 to 12/10/19 11.00

Looking at the blockchain BTC HeatMap instead, TxTfrValMedUSD_BTC shows an interesting correlated oscillation with Close (d+1) (roughly 0.7), also in huge volatility periods.

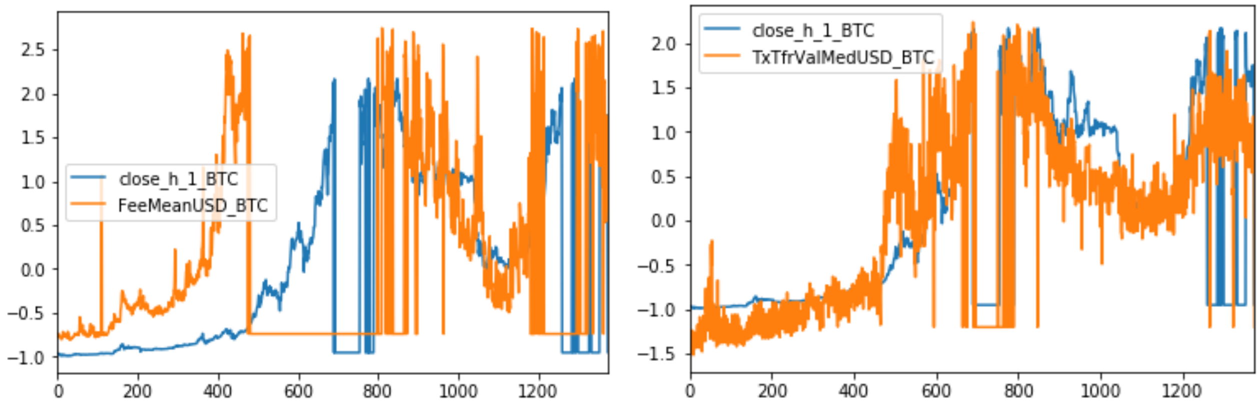


Fig 4.8 and Fig 4.9: BTC MatplotLib plots, where Block variables are compared withClosing Bth at t+1 from 18/09/19 to 12/10/19

Daily Miners Fee Mean (FeeMeanUSD_BTC), has a Pearson value of 0.4, and how represented in Fig 4.7, the plotted time series altering phase of good match and phase with clear reaction lag.
Therefore, Blockchain variables entails important correlation with daily ETH price, and with shallow intensity with daily BTC. Contrary, Social data affects drastically on financial BTC time series, whereas inconsistent relationship with hourly ETH price is emerged from the analysis.
At the same time, it is important to notice that correlation plots focus on a limited time horizon, in order to frame with more detailed the time series oscillation. Extending the plot to whole dataset,

Social Data reflects the observation exposed above, whereas Block attributes displays a strange pattern, observable on the below figures. (Fig 4.9 and 4.10)

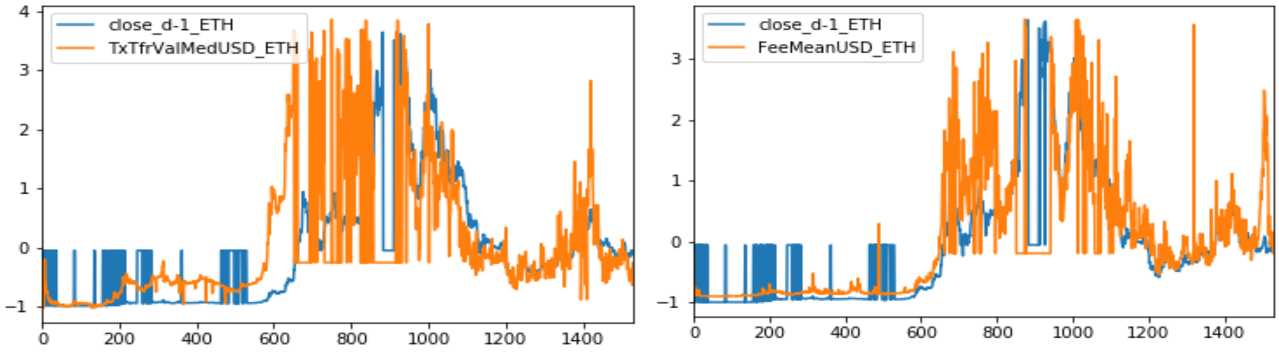


Fig 4.10 and Fig 4.11: ETH MatplotLib plots, where Block variables are compared with Closing Eth at t+1 for whole time horizon

Blockchain data show a common attitude, they are able to strike a pose similar with financial movements, but the there is a clear transitory phase close to the coin emission, where the correlation is completely out of the picture.

The HeatMap analysis is extended to XMR, DASH and LTC coins, as well.

Due to its extreme infancy, XMR HeatMap does not provides interesting relationship with the Social data. Launched on April 18th , 2014, it is the most unpopular coin on Social and Developing platforms from analyzed coin, explaining the lack of meaningful correlation. Difficult to explain, is instead the lack of deep connection with Block features. Just Trade and Market Pages Views show a timid affinity, with Pearson values of 0.4.

LTC Social date displays a good feeling with Closing trends, and appears as important candidate to explaining part of price spread. Trade Page Views, how observable in Fig 4.13, reflects at most Closing oscillations, evidencing a discrete dose of kinship through a Pearson Correlation of 06.

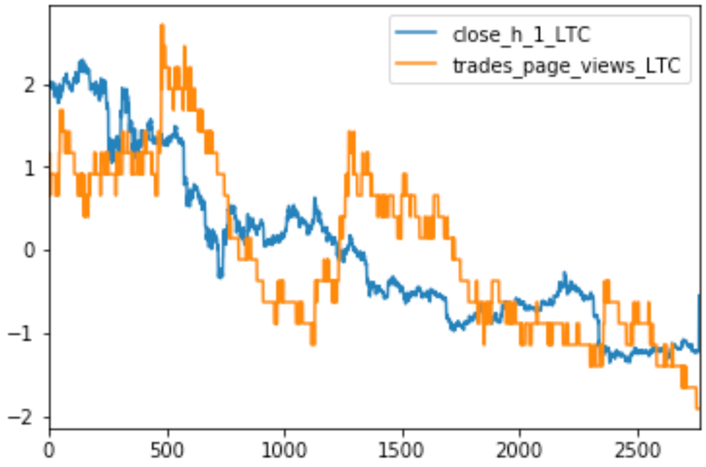


Fig 4.12: LTC MatplotLib, where Social Variables are compared with Closing LTC at t+1 from 1/7/19 2.00 to 12/10/19 11.00

Contrary Blockchain data, looks few interests to track Daily Closing Price at the next time instant t+1, and most of chain attributes appears strongly negative correlated, as illustrated in Fig 4.14

The lack of such Block attributes, instead affects the HeatMap DASH analysis. The matrix is undersized compared with the others. Nevertheless, variables as Block Size (Byte), Difficulty to Find new block and Currency Supply, display a discrete level of feeling (0.6), whereas features as FeeMeanUSD and FeeTotUSD demonstrates a negative correlation. This entails the presence of an ambiguous connection between DASH financial data and Blockchian attributes.

Social factors instead show interesting correlations, such as:

- Corr (Close_DASH (d+1), Total_page_view) = o.6
- Corr (Close_DASH (d+1), Market_page_view) = o.7
- Corr (Close_DASH (d+1), Fb_likes) = o.5
- Corr (Close_DASH (d+1), Posts) = o.6

The correlation analysis, combined with data visualization is a powerful tool that allows inferring on data insights and deciding which data can empirically implemented into Deep Learning phase.

For instance, can be interesting to design a machine algorithm experiment, trained on hourly OHLCV and Blockchain data for ETH, BTC.

3. MLP AND SVC SIMULATIONS

This part formalizes the applications of Multilayer Perceptron, a form of Neural Network algorithm, and Support Vector Machine to the data analyzed so far.

Sklearn Python library allows approaching Multilayer Perceptron and Support Vector Classifier, without coding algorithms from scratch, and considering behind optimization mathematics as a black-box. This has the great advantage of spending more time on simulations and result analysis rather than coding. Both method can be classification and regression algorithm, that in simple terms means that produce as output respectively discrete and continuous values. This work considers the former approach, so it is important to define a discrete targeting value within experimental datasets. In order to do this, a new attribute has added to dataset, called price changing computed as:

$$Price\ changing\ (t) = \frac{Closing\ Price\ (t+1\ ) - ClosingPrice(t\ )}{ClosingPrice(t\ )}$$

and then target class is qualified as:

- ***Class (t) = 'Upper', if Price Changing (t) ⩾ + 0.005***
- ***Class (t) = ' No Signal', if -0.005< Price Changing (t)< 0.005***
- ***Class (t) = ' Lower', if Price Changing ⩽ - 0.005***

Table 5.1 frames the performances achieved for each coin. The yellow side displays accuracies of Multilayer Perceptron and Support Vector Classifier trained on Datasets with daily sampling frequency. The blue side proposes coins 'accuracy of MLP and SVC, settled on hourly granularity. Granularity is not the only difference between the sides, while the former defines algorithms performances based on Blockchain and OHLCV data, the latter refers to Social and OHLCV data.

Table 5.1: Accuracy values each coins

	MLP_DAY	SVC_DAY	MLP_HOUR	SVC_HOUR
BTC	0.283972	0.39459	0.6743475	0.730869
ETH	0.447867	0.483478	0.5678897	0.687688
XMR	0.451282	0.442735	0.5656521	0.578695
LTC	0.45148	0.47708	0.54608	0.59304
DASH	0.448226	0.523445	0.5147826	0.599565

The first insight is the evident difference of performances between the two sides. The average MPL_DAY accuracy computed across the five coins is 0.4077, whereas SVC_DAY one is 0.4672. The average values for MLP_HOUR and SVC_HOUR respectively are 0.590326 and 0.6573. This is not a surprise due to the wider amount of data that hourly dataset offers compared with daily one. The first contains 6764 entries, whereas the second depends on coins emission date, but with an average entries counting of 2745. This difference affects the algorithm training process and as consequence the accuracy. This leads to an important consideration about algorithms' evaluations. While for the hourly datasets, targets forecasting yields are directly comparable, daily ones are function of the amount of data collected. For instance, it is likelihood that ancient coins, as Bitcoin and Litecoin get superior accuracy.

What is probably more interesting is the difference of performance between the two machine learning algorithms. Excluding Monero (XMR) coin, where MLP and SVC own aligned efficiency, the Support Vector Classifier emerges as the most performant.

Hourly Bitcoin Price Forecasting have produced the highest accuracy, SVC_Accuracy (BTC) = 0.73089 and MLP_Accuracy (BTC)=0.6748 are satisfactory values and confirm what explained in the Data Analysis chapter: Social Variables, combined with Financial time series result interesting Bitcoin spread explanation factors. On the other side, daily BTC price forecastings result inaccurate for both algorithms (0.2839 and 0.3945), but also this can be easily explained. The low accuracies are fruit of lack of greatest correlations between coin closing (t+1) movements and Blockchain variables and the limited data through algorithm are trained on. Support Vector Classifier has guaranteed higher performance due to its higher capability to explain output, without requiring extremely huge amount of training data.

Ethereum coin partially reflects the consideration of chapter 4. Due to the interesting relationship with Blockchain variables, daily accuracy improves (0.44768 and 0.4838), but does not justify at all the expectations that movements correlation created. In this case, it results stiff to infer on what percentage of fault is allocable to the limited training datasets.

Difficult to explain is also the optimistic result, given from hourly accuracy (0.56789 and 0.6878). Social data in fact, just report a timid affinity, making more than suspicious the presence of good randomality dose within algorithms execution.

Although the lack of deep connection between XMR closing (t+1) and Blockchain attributes, daily target forecastings present discrete values (0.451282 and 0.4427) compared with the mean values (0.4077 and 0.4672). The blue side do not provide particular insights, hourly XMR accuracies (0.56565 and 0.578695) are probably attributable at most to wider available data dump.

Strange is instead the case of DASH simulation outcomes; although the limited number of Blockchain attributes, daily SVC forecasting achieves the best result (SVC_DAY (DASH) ≃ 0.523445), whereas hourly forecasting is comparable with the mean (0.51478 and 0.59952).