VANKOR ALGORITHM

May 2023

1. **Introduction**

This document explains the process followed for the creation of the Vankor algorithm. This algorithm aims to predict market risk based on both off chain and on chain variables as well as new ones created with financial knowledge and reasoning.

Technologically speaking, the model consists of LSTM neural networks, the latest in deep learning time series prediction.

1. **Data gathering and preparation**

As mentioned in the introduction, information has been gathered both from off chain and on chain from different sources.

|  |  |  |
| --- | --- | --- |
| **Variable** | **Descripción** | **Fuente** |
|  |  |  |
| Volume | Volume traded |  |
| HL\_sprd | High and Low spread | Trading View |
| trans\_fees | Price to pay for every transaction |  |
| Value\_SP500 | The value of SP500 index | Trading View |
| vol\_current | Total amount of Bitcoin traded |  |
| Detrended Price Oscillator | Indicator in technical analysis that attempts to eliminate the long-term trends in prices by using a displaced moving average so it does not react to the most current price action | Trading View |
| unrealized-profit | Total profit (in USD) of a coin in existence whose price at realization time was lower than the current price, normalized by the market cap | GlassNode |
| Value\_USunem | Variation percentage of US unemployment | Ycharts |
| balance-100 | The number of unique addresses holding at least 100 coins. | GlassNode |
| ratio\_mvrv | ratio between market cap and realised cap. It gives an indication of when the traded price is below a "fair value" | GlassNode |
| US\_bank\_idx | Is designed to measure the performance of U.S. companies in the banks sector. | Trading View |
| block\_reward | Miner's money amount recieved for every minted block |  |
| balance-10 | The number of unique addresses holding at least 10 coins. | GlassNode |
| Value\_EFF | volume-weighted median of overnight federal funds transactions reported in the FR 2420 Report of Selected Money Market Rates | Ycharts |
| block\_size | Size of the block | GlassNode |
| tran\_num | Number of transactions |  |
| Value\_CPI | Value of Consumer Price Index | Trading View |
| Value\_M3 | Broad money (M3) includes currency, deposits with an agreed maturity of up to two years, deposits redeemable at notice of up to three months and repurchase agreements, money market fund shares/units and debt securities up to two years | Ycharts |
| RSI | Relative Strength Index |  |
| ratio\_nvt | The Network Value to Transactions (NVT) Ratio is computed by dividing the market cap by the transferred on-chain volume measured in USD | GlassNode |
| Variation\_UStotal | measure of the number of U.S. workers in the economy that excludes proprietors, private household employees, unpaid volunteers, farm employees, and the unincorporated self-employed | Ycharts |
| percent-of-supply | The percent of circulating supply that has not moved in at least 3 years | GlassNode |
| CO\_sprd | Open-Close Spread | Paper |
| Put-call Ratio | The Options Volume Put/Call Ratio shows the put volume divided by call volume traded in options contracts in the last 24 hours. | GlassNode |
| balance-10k | The number of unique addresses holding at least 10k coins. | GlassNode |
|  |  |  |
|  |  |  |
| Aggregate security spend (Thermocap) | is the aggregated amount of coins paid to miners and serves as a proxy to mining resources spent | GlassNode |
| Balance Price | is the difference between Realized Price and Transfer Price. Transfer Price is the cumulative sum of Coin Days Destroyed in USD, adjusted by circulating supply and total time since Bitcoin's inception. Balanced Price attempts to detect major cycle bottoms | GlassNode |
| Balance on Exchanges | The total amount of coins held on exchange addresses | GlassNode |
| Block Height | the total number of blocks ever created and included in the main blockchain | GlassNode |
| Circulating Supply | The total amount of all coins ever created/issued | GlassNode |
| Cumulative Value-Days Destroyed | [is the ratio of the cumulative USD value of Coin Days Destroyed and the market age (in days)](https://studio.glassnode.com/metrics?a=BTC&m=indicators.Cdd) | GlassNode |
| Delta Cap | is the difference between Realized Cap and Average Cap, where Average Cap is assumed to be the life-to-date moving average of Market Cap | GlassNode |
| Investor Capitalization | is the difference of Realized Cap and Thermocap. It discounts the capital paid to miners from the market’s general cost basis, serving as a bottom indicator in bear cycles | GlassNode |
| Median Value of created UTXOS | The median amount of coins in newly created UTXOs | GlassNode |
| Minning Difficulty | The current estimated number of hashes required to mine a block | GlassNode |
| MVRV-Z Score | is defined as the ratio between the difference of market cap and realized cap, and the standard deviation of all historical market cap data | GlassNode |
| Realized Cap | Realized Cap values different part of the supplies at different prices (instead of using the current daily close) | GlassNode |

* 1. **Off chain**

Iñakis table, explained

* 1. **On chain**

Table

1. **Exploratory Data Analysis (EDA)**

The final dataset contains 2026 rows x 66 columns with a timeframe that goes from 2016-02-29 to 2023-05-08.

Regarding the type of features, there are two categorical and the rest of them are numerical.

Something about describe??...

* 1. **Missing values**

First, a study is made of the percentage of nulls per row. The highest percentage in the dataset is approximately 0.15, so it will not be necessary to study the case of eliminating rows completely. Regarding the columns with null values, the following imputation methods have been established:

* Red/Green Arrow: The data are very unbalanced, so it is decided to impute with the mode (0).
* Value\_USreal: More than 50% missing – normal distribution with most values on 0. I'd be interesting to know the data source but imputing by mean is a good decision here.
* Columns of on chain data: from a specific date the data does not exist. We choose to delete all the last rows up to that date.

1. **Feature engineering**

The following new variables are calculated from those in the original dataset:

* Log\_price: simply the logarithm of the "close" variable.
* Returns: is the percentage change of the price ('close).
* Log\_return: the logarithm of returns.
* HL\_sprd: the logarithm (here goes better the formula)
* CO\_sprd:
* Volume: logarithm of volume

It is decided that the target variable will be the price volatility (close). For this purpose, volatility is calculated as follows:

It is calculated for periods of 7, 30, 60, 180 and 365 days. The 30-day volatility is chosen as the target variable, since the 7-day volatility is too noisy, and the 30-day volatility is too smooth.

Specifically, two new variables are created: 'vol\_current', the current volatility, and 'vol\_future', the volatility that the model will predict with an n\_lag of 7 days.

1. **Multivariate Analysis**

Once all variables are collected and the new ones calculated, we proceed to study the correlation between them and specifically with the target variable. It can be represented in a heat map that look as follows:

A picture containing pattern, colorfulness, art, creative arts

Description automatically generated

A can be seen, ther is a high correlation between features. In particular, there are columns that correlate with almost all the variables (the blackest and whitest columns), therefore, the next step is to get rid of highly correlated columns in order to decrease the size of the dataset.

The features that have been deleted are:

* Returns, log\_price: due to correlation with log\_return.
* Upper Bollinger: due to high correlation with several features.
* Ratio\_mvrv: same as upper Bollinger
* Value\_USpce:

Finally the heat map correlation looks as follows:

A picture containing pattern, colorfulness, creative arts, square

Description automatically generated

It shows now that there is no more high correlations between features. Well done!

1. **Train Validation Test splits**

As usual in a machine learning process, data must be separated in two or more parts to evaluate how model is predicting the values.

In this case, it has been decided to reserve 365 days to test the model and 30 days to predict future data. In conclusion, final splits are 2126 days to training, 365 days to validation and 30 days to test.

A picture containing text, font, plot, line

Description automatically generated

Then, X\_train and y\_train are created first one with all features but vol\_future and second one only with this feature.

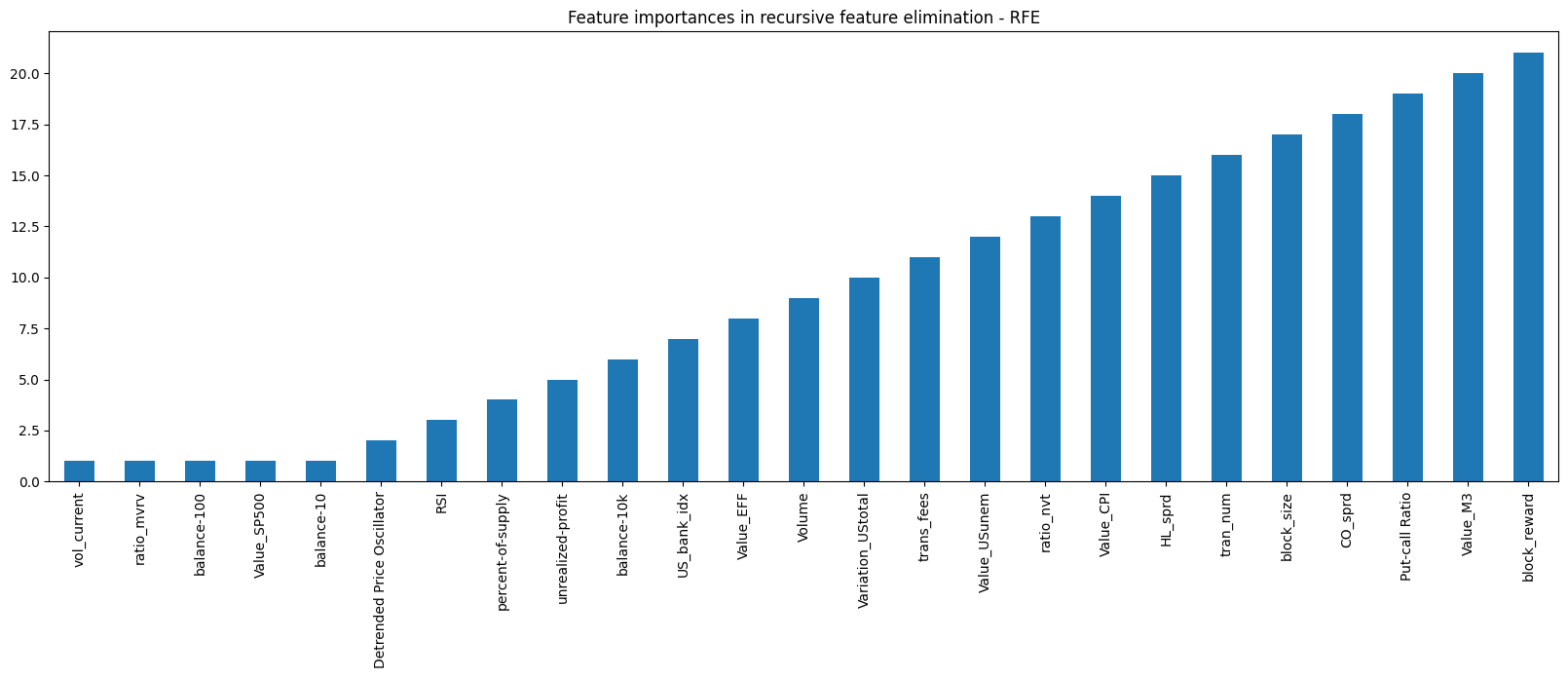
1. **Features Importance**

It has been use a lineal regression model for importance first approach.

A picture containing plot, line, diagram, screenshot

Description automatically generated

Surely there will be some gradient boosting model testing, so another recursive feature elimination (RFE) has been made with a Decision tree regressor.

****

This figure shows a different features importance. This order will be use during the testing of gradien-boosting models, selecting the following features:

selected\_features = [

'vol\_current',

'ratio\_mvrv',

'balance-100',

'Value\_SP500',

'balance-10',

'Detrended Price Oscillator',

'RSI',

'percent-of-supply',

'balance-10k',

'unrealized-profit',

'Value\_EFF',

'Volume',

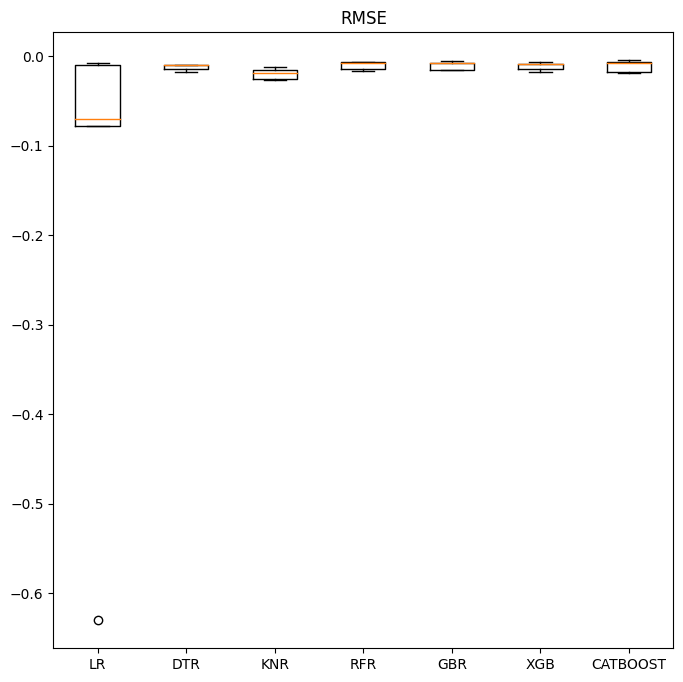
'US\_bank\_idx',

'Variation\_UStotal']

1. **Cross validation models**

A cross validation has been done to have a glance of the models that can fit best in the selected data. The different models selected are:

* Linear Regression
* Decision Tree Regressor
* KNeighbors Regressor
* Random Forest Regressor
* GradienBoosting Regressor
* XGBoost
* CatBoost Regressor



Intr

Let’s try the last two models that apparently are the best ones.