

Project Code README

Table of Contents

- Data Cleaning..... 2
- Technical indicators..... 3
- Data Normalization..... 4
- Target Creation..... 5
- Explore Feature correlation and importance 6
- Machine learning..... 7

Data Cleaning

The cleaning aspect of the project was completed in one jupyter notebook called *Dataset_cleaner.ipynb*.

The script loads the data (given the correct file structure and source files) from a csv file, to the desired structure and values in pandas.

The notebook called *Finding Price inconsistencies.ipynb* checks the effect of the cleaning process, and scans the dataset for price inconsistencies.

Technical indicators

The notebook *Visualize features.ipynb* show the default settings for the technical indicators included. This notebook was used as a workbench for the primary feature scripts with the filename format of:

Feature_{feature_name}.py , where {feature_name} can be (BB, EMA, MACD, STCO, RSI, ... , etc.)

Each of these scripts create the feature for 8 different windows of time for the formula.

Data Normalization

The notebook called *Find scalar vals for feature-standardization.ipynb* was used to create files of values for individual indicators to be scaled with. Additionally, it performs the three different transformations mentioned in the research paper.

Target Creation

Target creation was made with the notebook called *Target Creator.ipynb*. It creates the targets for the machine learning experiments which is used as a hyperparameter.

Explore Feature correlation and importance

A notebook called *Explore Feature correlation and importance.ipynb* creates a bar chart of pearsons correlation coefficient respective to all features used to the project. This was used as an exploratory data analysis method, to gain insight on the correlation of the data, which is part also helped with the conclusion of the paper.

Machine learning

data_set_builder.py, *create_lstm_train_test.py* contained functions used by *Random_search_CNN.py* and *Random_search_LSTM.py*. The two last mentioned python applications were used to random search optimal hyperparameters.

Results for each model.ipynb was used to gain intelligence on model performance, by using a self-implemented trading simulation algorithm.

