SAIDL Assgn 2021

Tanay Sheth

Q1)

There have been plenty of projects on using Classification Models on the time series dataset for stock selection. I would like to work on the idea of binarizing the prediction labels based on their direction and predicting them.

Goal: Binary Forecasting of Time Series Data [Proposed the same project for TIP]

1. A classification model able to predict the direction of Indian Stock Market.

2. Backtesting of the classification model generated trade results

Literature Review and Ingredients:

Converting the forecasting problem into a binarized directional problem makes it more accessible to analyze with classification models.

2. A set of 5 technical indicators and 23 fundamental indicators was identified to establish the possibility of generating excess returns on the stock market

https://arxiv.org/ftp/arxiv/papers/2103/2103.09106.pdf

Using these features and binarizing our outcome the forecasting classification model can generate impressive results.

3. Prior work has been done using few classification models for stock selection and forecasting but Indian Markets aren't analyzed yet.

Quantify Hypothesis:

We quantify the prediction label as uptrend or not.

For a EOD Data to be uptrend the close > Open along with an appropriate ADX Threshold.

How ADX works is

$$+\text{DI} = \left(\frac{\text{Smoothed } +\text{DM}}{\text{ATR}}\right) \times 100$$

$$\text{-DI} = \left(\frac{\text{Smoothed -DM}}{\text{ATR}}\right) \times 100$$

$$\text{DX} = \left(\frac{|\ +\text{DI} - \text{-DI}\ |}{|\ +\text{DI} + \text{-DI}\ |}\right) \times 100$$

$$\text{ADX} = \frac{(\text{Prior ADX} \times 13) + \text{Current ADX}}{14}$$

where:

$$+\text{DM (Directional Movement)} = \text{Current High} - \text{PH}$$

$$\text{PH} = \text{Previous High}$$

$$\text{-DM} = \text{Previous Low} - \text{Current Low}$$

$$\text{Smoothed } +/\text{-DM} = \sum_{t=1}^{14} \text{DM} - \left(\frac{\sum_{t=1}^{14} \text{DM}}{14}\right) + \text{CDM}$$

$$\text{CDM} = \text{Current DM}$$

$$\text{ATR} = \text{Average True Range}$$

With the help of this we can easily quantify trends.

With the help of a set of 5 technical indicators and 23 fundamental indicators as mentioned in literature review 2, we produce our feature attributes for the classifier.

Toolkit to Implement Models:

1) Start Simple with regression classifiers, random forests, XGBoost
2) Move on to neural networks starting with a basic MLP model

Model Evaluation:

Confusion Matrix along with Accuracy, Precision, Recall and F1 Score

Q2)

Tried to attempt RL Question

Main issue faced : I feel the new state variables from action aren't being integrated properly

Q3)

Central Claims of the Paper

1) Implementing pre-existing models on time series data : ARIMA, Dual Stage Attention RNN, Fully Convolutional Network, Dual Self Attention Network
2) Compare results before and after L1 trend filtering

Implementation:

Tried implementing DARNN and ARIMA (both are directly accessible models from library)

The parameters (from paper) :

To apply the L1 trend filtering feature to the input data, we set the parameter λ to 0.005. In ARIMA, we set the parameters p, d, and q to corresponding values of 1, 0, and 0. There are four parameters in the DA-RNN: $i.e.$, the number of input time steps in the window Ti , the number of output steps To , and the size of hidden states for the encoder m and decoder p. To find optimal parameters, we conducted a grid search. Finally, we find the best performance over the validation set which is used for evaluation when $Ti = 64, To = 5,$ and $m = p \in \{64, 128\}$

The dataset used indian stock market prices for past one year.

NSE (2020-2021 Aug) and 5 other constituents while implementing DARNN

The most important part of the code L1 trend filtering was implemented by following steps followed in the code provided in C by https://web.stanford.edu/~boyd/papers/l1_trend_filter.html

Evaluating the Results: Root mean squared error (RMSE), the mean absolute error (MAE),

and the mean absolute percentage error(MAPE), with the Adam optimizer were to be implemented but is incomplete. Would continue to work on it as and when time permits