PREDICTING REALIZED VOLATILITY IN STOCK PRICES

Amit Kukreja
19-October-2022

kukreja.amit76@gmail.com

WHAT IS REALIZED VOLATILITY & WHY PREDICT IT?

- Volatility captures the amount of fluctuation in bid/ask prices of a stock
- It is an important input for pricing options
- Optiver is a leading global electronic market maker firm
- It wants to further evolve its industry leading pricing algorithm

Realized Volatility computation:

$$RV = \sqrt{\sum_{t} r_{t-1,t}^2}$$

$$WAP = \frac{bid\_price * ask\_size + ask\_price * bid\_size}{ask\_size + bid\_size}$$

r is the log return of *WAP (Weighted Average Price)*

PROBLEM STATEMENT: PREDICT REALIZED VOLATILITY FOR THE FUTURE 10-MINUTE PERIOD

Context : Optiver has book and trade data for current 10-min period, using which it wants to predict realized volatility for future 10-min period. Accurately predicting future volatility is critical input for pricing stock options that Optiver trades in.

Criteria for Success : A model that can minimize RMSPE (*Root Mean Squared Percentage Error*) between predicted and true values of future volatility.

Scope : **112 stocks traded by Optiver**

Constraints:

- Non-availability of data beyond 10-minute period
- Non-availability of external factors influencing a particular stock or the market as a whole

Stakeholders to provide Key Insight :

1) Ben Bell – Springboard Mentor

Data Sources : Book & Trade data for 112 stocks for approx. 3800 ten-minute time periods.

https://www.kaggle.com/code/jiashenliu/introduction-to-financial-concepts-and-data/data?scriptVersionId=67183666

DATA INSPECTION

Prices are normalized, Time_id's have no sequential logic

Book File: Contains Top 2 bid/ask prices and sizes

	time_id	seconds_in_bucket	bid_price1	ask_price1	bid_price2	ask_price2	bid_size1	ask_size1	bid_size2	ask_size2
0	5	0	1.001422	1.002301	1.00137	1.002353	3	226	2	100
1	5	1	1.001422	1.002301	1.00137	1.002353	3	100	2	100
2	5	5	1.001422	1.002301	1.00137	1.002405	3	100	2	100
3	5	6	1.001422	1.002301	1.00137	1.002405	3	126	2	100

- Missing 'seconds_in_bucket' : means no change in bid/ask prices or sizes for those seconds.
- Each time_id **is 10-min,** has upto 600 seconds of data
- WAP is derived from bid1 / ask1 data

Trade File: Data on trade prices, sizes

	time_id	seconds_in_bucket	price	size	order_count
0	5	21	1.002301	326	12
1	5	46	1.002778	128	4
2	5	50	1.002818	55	1

- Missing 'seconds_in_bucket' : no trades for those seconds
- This file is sparse compared to Book file

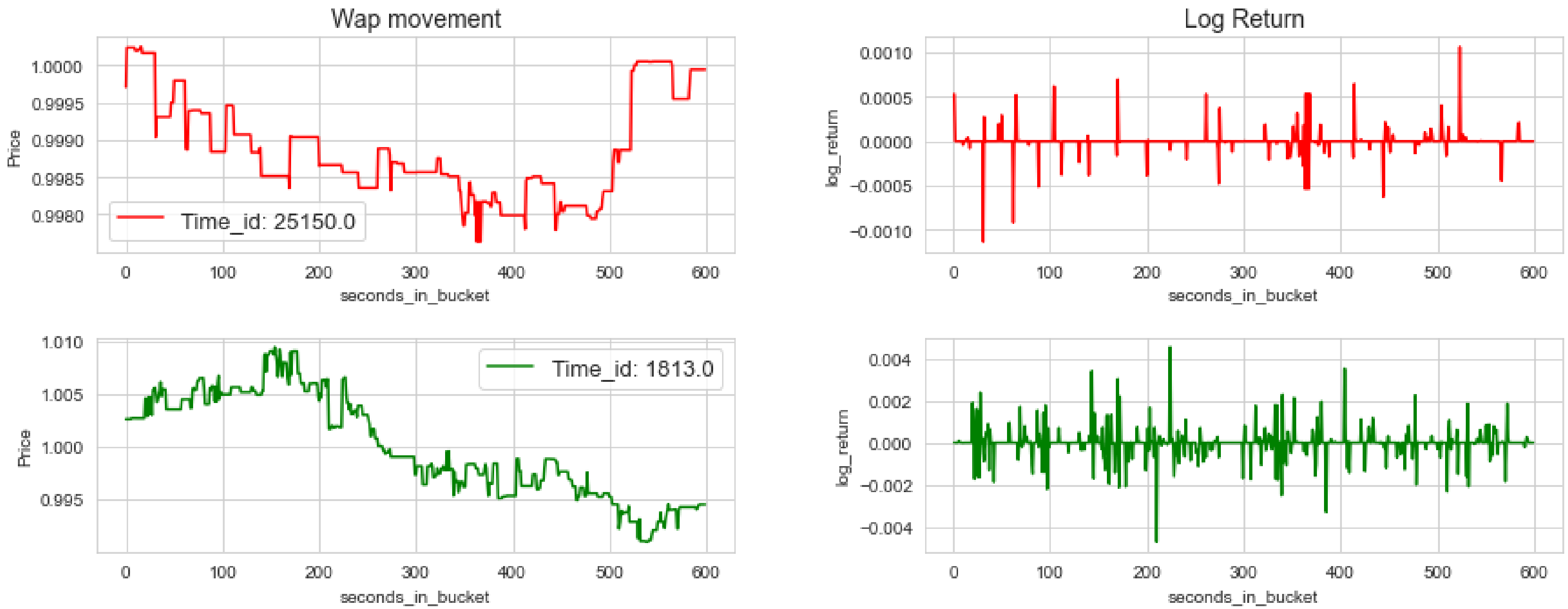
Training File: Contains target

	stock_id	time_id	target
0	0	5	0.004136
1	0	11	0.001445
2	0	16	0.002168

- 'target' : realized volatility in 10-min following time_id 5 /11/16
- To be predicted using book & trade data for respective time_id

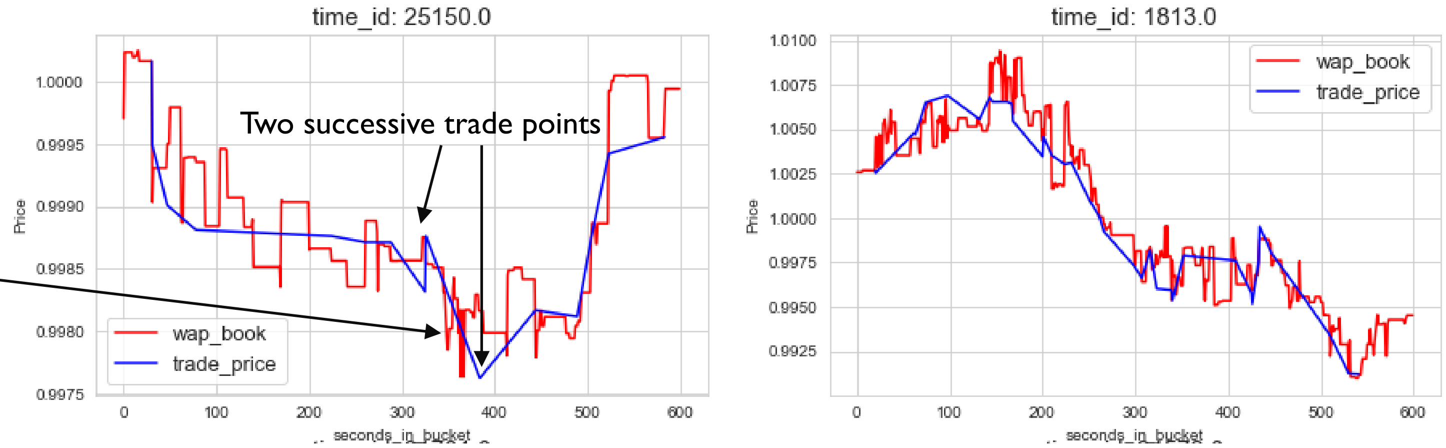
EXPLORATORY DATA ANALYSIS

WAP is a random walk, Log returns are stationary



WAP & Trade Price closely track one another

WAP moves a lot between 2 trade points

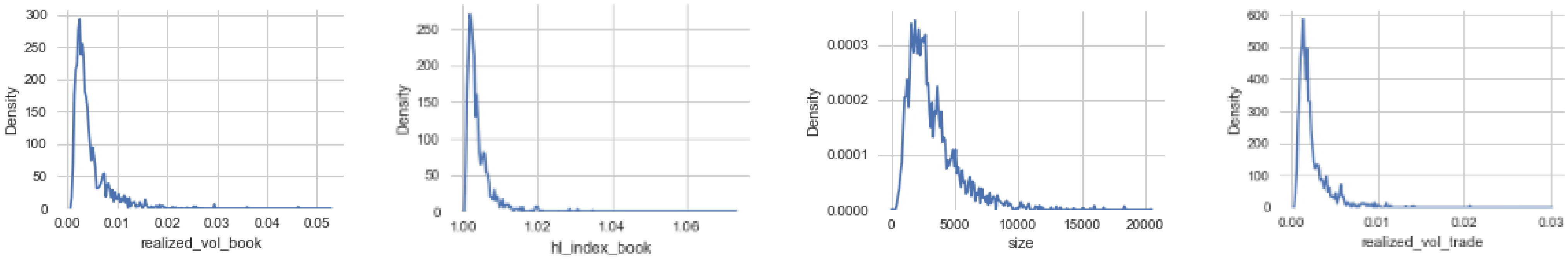


Difficult to say which one (WAP or Trade Price) drives the other.

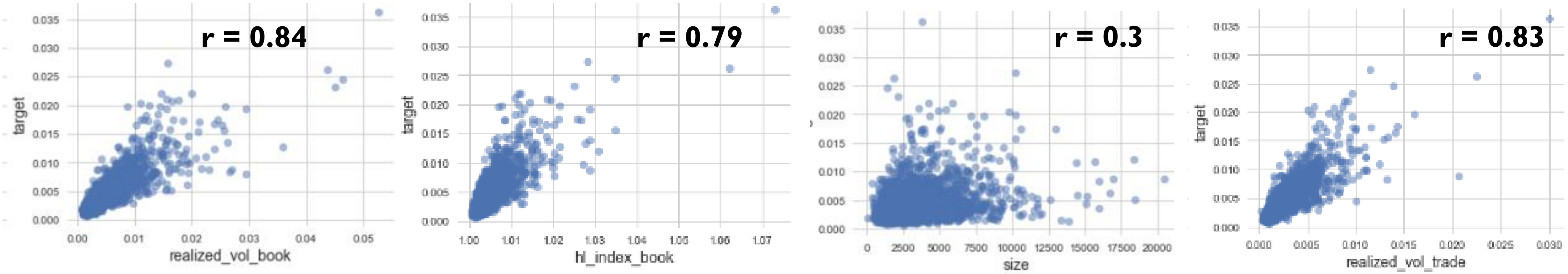
EXPLORATORY DATA ANALYSIS

Features are not normally distributed

Kolmogorov Smirnov test returns p-value of **0.00** for all features



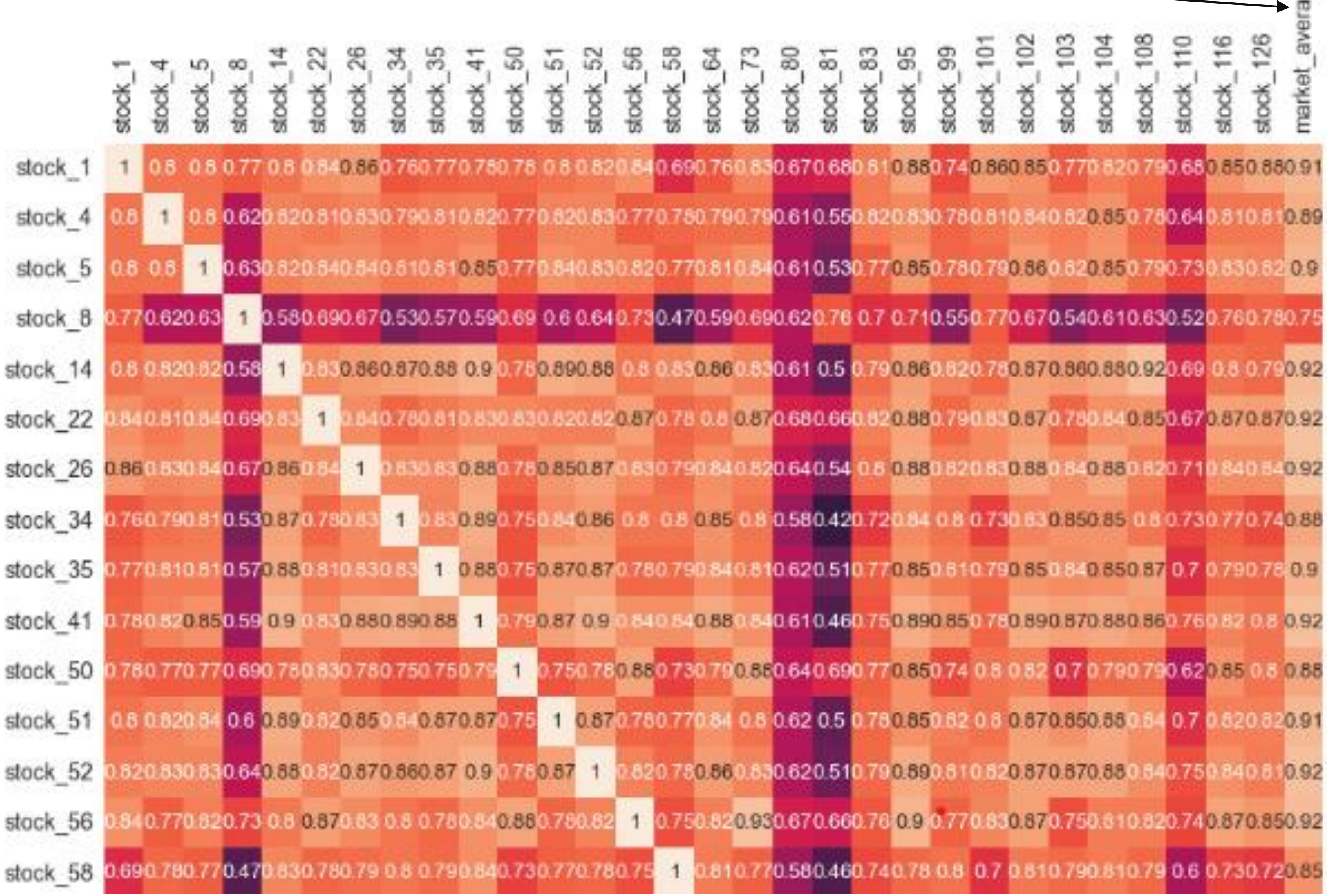
‘realized_vol_book’, ‘hl_index_book’, ‘realized_vol_trade’ have high correlation with ‘target’



Many stocks have strong correlation in realized volatility

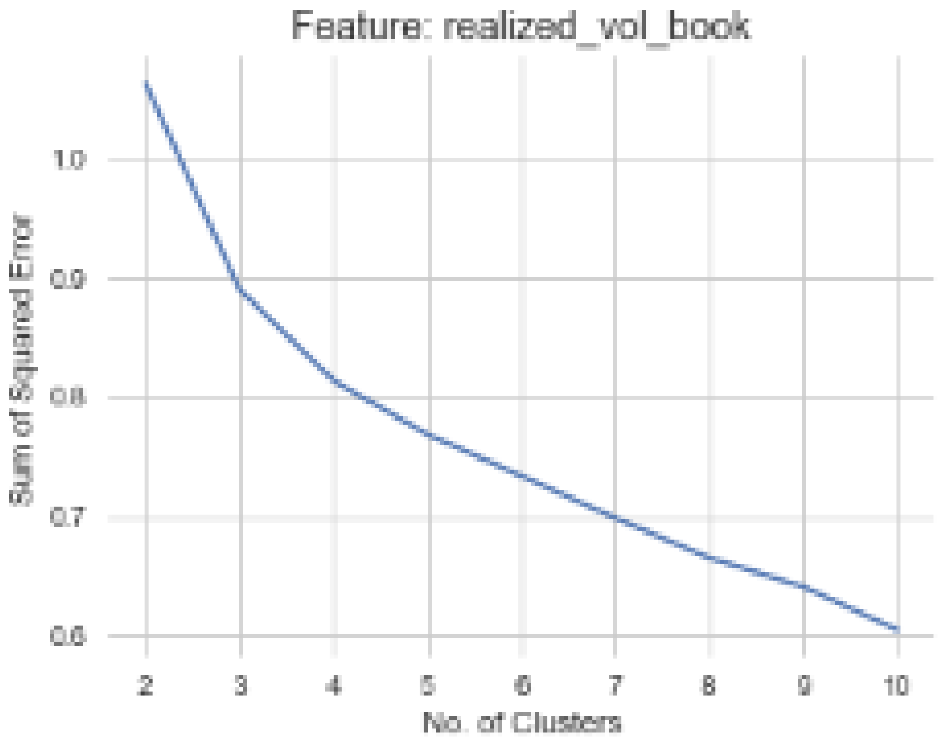
And even higher correlation to the average!

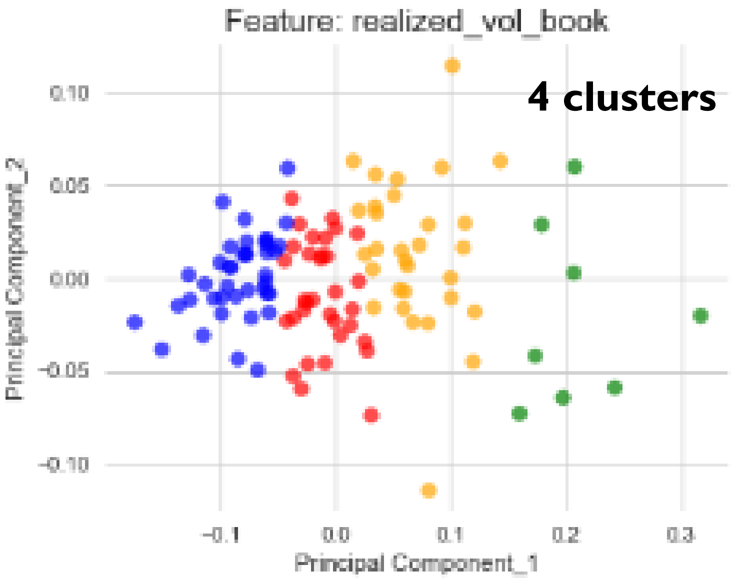
(Average of 30 stocks)

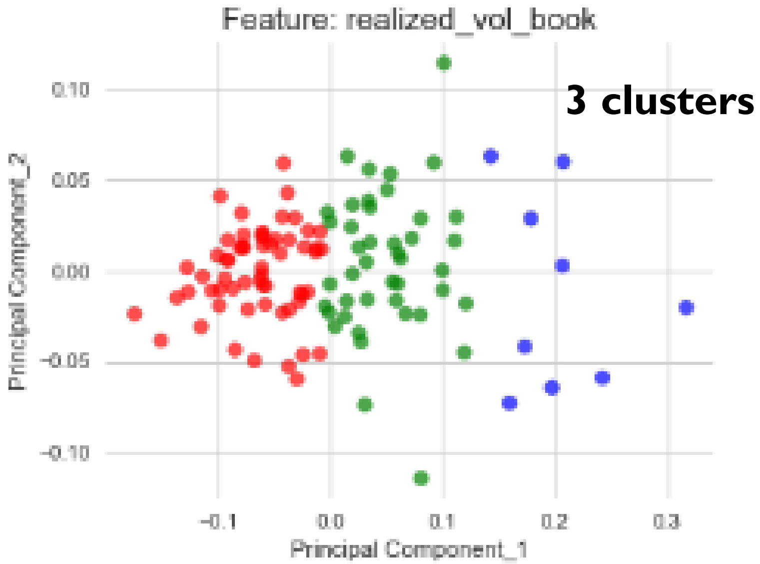


EXPLORATORY DATA ANALYSIS

The 112 stocks show 3–4 clusters







EDA: KEY LEARNINGS

- WAP is a random walk, it’s log returns are stationary
- WAP & Trade Price closely track one another, but difficult to say which one drives the other
- Features are not normally distributed
- ‘target’ is strongly correlated to
    - ‘realized_volatility_book’
    - ‘hl_index_book’
    - ‘realized_volatility_trade’
- The 112 stocks can be clustered into 3-4 groups based on realized volatility feature

FEATURE ENGINEERING

Generated 170+ features based on technical analysis, concepts of motion, stocks clusters & time-bands

1. Basic Features : simple features from book & trade files
   - wap1, wap2, spread, price_premium, realized volatility, turnover
2. Based on concepts of motion : Capture movement dynamics of price
   - speed, acceleration
3. On technical analysis : Capture magnitude & inertia of price activity
   - high-low index, momentum (size * speed) , force (size * acceleration)
4. Cluster Features :
   - Cluster (categorical feature), avg/std of realized volatility of a cluster
5. Some of above features across time-bands : Divided the 10-minutes into
   - Two 5-min halves (H1 & H2)
   - Four 2.5-min quarters (Q1 – Q4)

M O D E L L I N G

Naïve (baseline) Model: Setting the benchmark

- This model simply assumes that future 10-min volatility is the same as current 10-min volatility

- It delivers an RMSPE score of 0.341 across all 112 stocks for the training dataset

- The performance of the naive model is not amazing, but it helps to establish a benchmark

MODELLING

Linear Model: Understanding feature importance

- Removed colinear features where r > 0.95
  - Reduce approx. 90 features
- To remove multicollinearity, used variance inflation factor
  - Iteratively removed features, till important features had VIF below 10

VIF Factor	features
inf	vol_diff_from_cluster
inf	acc_sum_mean_minus_H2
inf	log_momentum_book_sum_H1
inf	realized_vol_wap1_H2
inf	log_momentum_book_sum_H2
inf	rv_mean_minus_H1
inf	acceleration_trade_price_mean
inf	rv_mean_minus_H2
inf	log_momentum_book_sum_Q1
inf	realized_vol_wap1
inf	log_momentum_book_sum_Q2

VIF Factor	features
9.512287	momentum_trade_mean
9.472200	bid_size1_mean
8.983602	ask_size1_mean
8.822655	realized_vol_wap1
8.491537	value_premium_1_mean
8.203072	price_premium_1_sum
6.794630	seconds_active
6.640426	acceleration_trade_price_mean
5.810010	value_premium_1_std
5.505904	realized_vol_wap1_cluster_mean
5.010561	force_trade_std
4.974507	order_count
4.726221	wap1_hl_index

MODELLING

Linear Model: Understanding feature importance

- Post VIF led pruning, there were 25 features in the dataset
- Top 10 features in the linear model were:

1. realized_volatility_wap1
2. realized_volatility_wap1_cluster_mean
3. realized_volatility_wap1_H2
4. price_premium_1
5. realized_volatility_trade_price
6. acceleration_trade_price_mean
7. momentum_trade_mean
8. wap1_hl_index
9. momentum_book_sum
10. value_premium_1_sum

- The top features confirmed that most of the concepts used for feature engineering were useful for the model
- Model had a RMSPE score of 0.2833 on validation set

Coef	weights
realized_vol_wap1	2.069792e-03
realized_vol_wap1_cluster_mean	5.924311e-04
price_premium_1_sum	2.559763e-04
trade_rv_mean_minus_H1	1.260712e-04
wap1_hl_index	8.742610e-05
momentum_book_sum	8.278138e-05
force_trade_std	3.483076e-05
value_premium_1_mean	1.258734e-05
ask_size1_std	9.962202e-06
wap1_mean	8.058074e-06
bid_size1_std	4.972991e-06
wap_diff_mean	7.673287e-07
const	3.997996e-19
log_momentum_book_sum	-2.415072e-06
bid_size1_mean	-2.839625e-06
seconds_active	-1.409950e-05
order_count	-1.932193e-05
ask_size1_mean	-2.384914e-05
seconds_active_book	-2.836506e-05
momentum_trade_sum	-3.737814e-05
log_force_book_std	-4.506134e-05
value_premium_1_std	-4.617142e-05
momentum_trade_mean	-9.674820e-05
acceleration_trade_price_mean	-1.226120e-04
rv_mean_minus_H2	-3.354489e-04

MODELLING

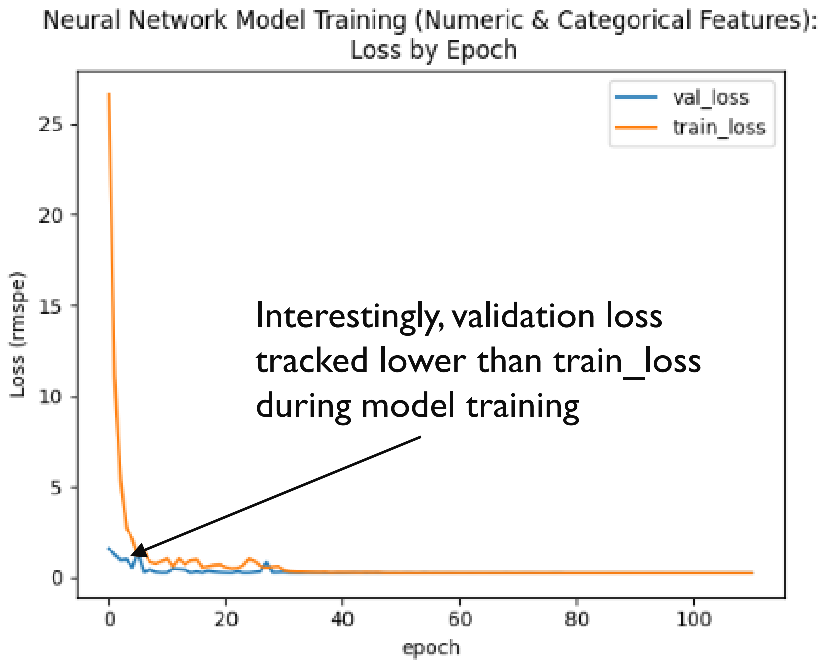
Deep Learning Model: Significant jump in performance

- Developed 2 deep learning models:
  - Model 1 : Based on only numeric features
  - Model 2 : Based on numeric features and categorical features (cluster ids)
- Tuned hyperparameters such as no. of layers, no. of neurons, activation function, learning rate & batch size
- The best model (Model 2) had the following hyperparameters
  - 3 dense hidden layers with [200, 200, 100] neurons
  - 1 embedding layer for categorical features (stock clusters)
  - Activation Function: LeakyReLU, alphas = [0.5, 0.3, 0.3]
  - Used 'ReduceLRonPlateau' callback to gradually lower LR to 1e-06
  - Batch size: 128

MODELLING

Deep Learning Model: Significant jump in performance

- RMSPE scores on validation set
  - Model 1 : 0.2125
  - Model 2 : 0.2104
- RMSPE score of Model 2 on test set was 0.215365.
- This score was among Top 30 scores present on the Kaggle contest leaderboard




```
2561/2561 [==============================] - 11s 2ms/step - loss: 0.2154
Evaluation: Neural network model based on numerical and categorical features

RMSPE loss on test set: 0.21536540985107422
\Avg. computation time per time_id for all 112 stocks: 0.018213899888818007 seconds
```

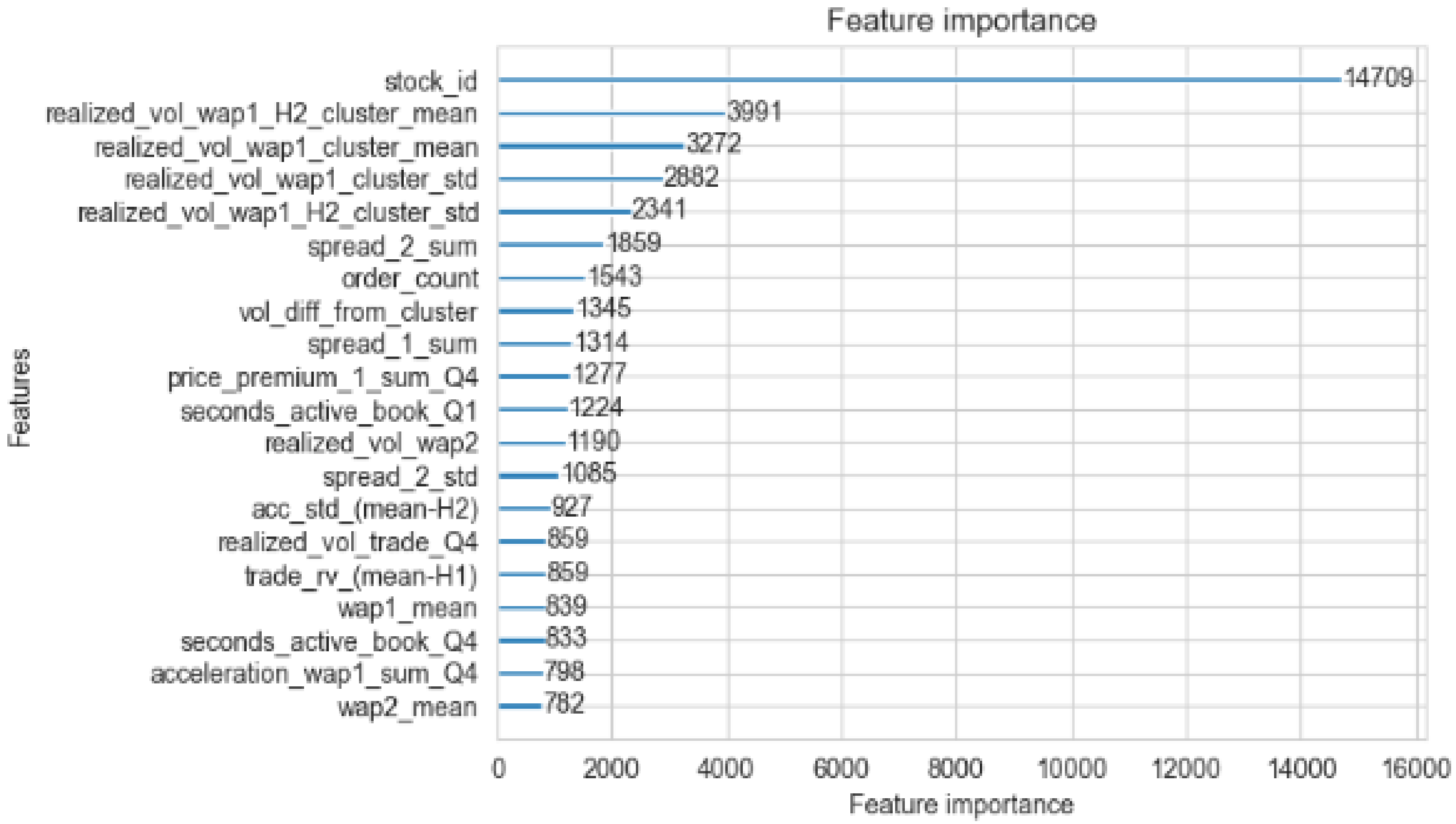
MODELLING

Light GBM Model: Understanding feature importance

- As with linear model, the Light GBM model helped understand feature importance
- Tuned hyperparameters such as learning rate, feature_fraction, max_depth, min_data_in_leaf
- Model had a RMSPE score of 0.2335 on validation set
- Top 20 features in this model were:

- '**stock_id**' had an outsized importance in this model

- realized_vol of the cluster was more important than that of the stock!



KEY FINDINGS

- Features based on technical analysis, price motion dynamics, clustering and time-bands are useful for modelling future 10-min volatility
- Features based on both book and trade data are important
- A deep learning model based on numeric and categorical features delivers a strong performance with an RMSPE score of 0.215365 on the test set
- 2 categorical features were used in addition to approx. 170 numeric features

BUSINESS APPLICATIONS

- The deep learning model has a simple architecture
  - 3 hidden layers and 1 embedding layer
  - 100-200 neurons per hidden layer
- It delivers good computational performance – takes just 0.018 seconds to make predictions for all 112 stocks
- As such, it suited for real time applications like stock options trading and can be deployed by electronic market maker firms such as Optiver

IDEAS FOR FURTHER DEVELOPMENT

- The model trained on data for ~3000 time_ids. More data could help train the model better

- Gathering more domain knowledge to further understand factors that have a bearing on short term volatility