Comparison of tree based models and SVMs on High-frequency LOB dynamics and feature importance study

IEOR 222 Final Project

Shengying Wang

Department of Statistics, UC Berkeley

December 10 2014

Outline of this presentation

- Project framework
- Strategies
- Results
- Conclusions

Project framework

- Fit a single decision tree to obtain the effect of each market feature by a tree diagram.
- Fit random forests to obtain test accuracy and time costs to compete with SVMs.
- Fit random forests to study the feature importance and compare with Lasso.
- Predict spread crossing for different time intervals by random forests and SVM to test robustness.

Strategies

- Data Attributes
- Different situations

Data Attributes (based on Kercheval and Zhang's work)

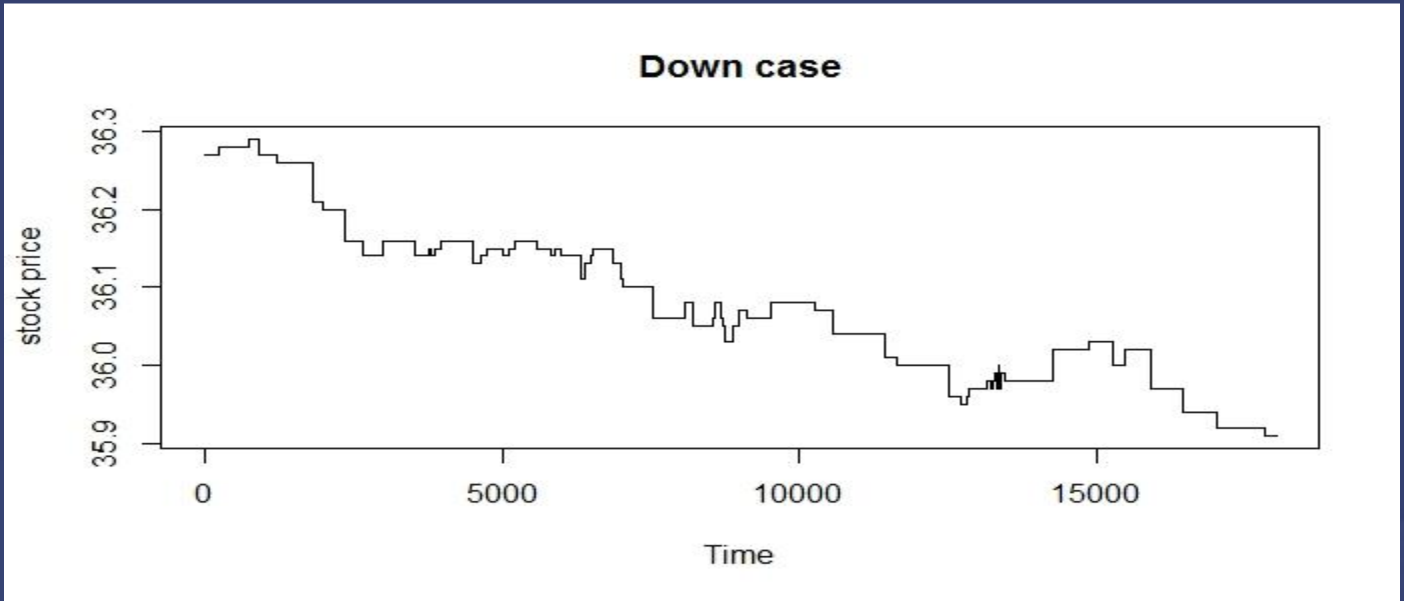
One can design his/her own feature sets.

Basic Set	Description(i = *level index*)
$v_1 = \{P_i^{ask},\ V_i^{ask},\ P_i^{bid},\ V_i^{bid}\}_{i=1}^{n},$	*price and volume(n levels, n=10)*

Time-insensitive Set	Description(i = *level index*)
$v_2 = \{(P_i^{ask} - P_i^{bid}), (P_i^{ask} + P_i^{bid})/2\}_{i=1}^{n},$	*bid-ask spreads and mid-prices*
$v_3 = \{P_n^{ask} - P_1^{ask},\ P_1^{bid} - P_n^{bid}\},$	*max-min price differences*
$v_4 = \{\lvert P_{i+1}^{ask} - P_i^{ask}\rvert,\ \lvert P_{i+1}^{bid} - P_i^{bid}\rvert\}_{i=1}^{n-1},$	*price level differences*
$v_5 = \{\frac{1}{n}\sum_{i=1}^{n} P_i^{ask},\ \frac{1}{n}\sum_{i=1}^{n} P_i^{bid},\ \frac{1}{n}\sum_{i=1}^{n} V_i^{ask},\ \frac{1}{n}\sum_{i=1}^{n} V_i^{bid}\},$	*mean prices and volumes*
$v_6 = \{\sum_{i=1}^{n}(P_i^{ask} - P_i^{bid}),\ \sum_{i=1}^{n}(V_i^{ask} - V_i^{bid})\},$	*accumulated differences*

Time-sensitive Set	Description(i = *level index*)
$v_7 = \{dP_i^{ask}/dt,\ dP_i^{bid}/dt,\ dV_i^{ask}/dt,\ dV_i^{bid}/dt\}_{i=1}^{n},$	*price and volume derivatives*
$v_8 = \{\lambda_{\Delta t}^{la},\ \lambda_{\Delta t}^{lb},\ \lambda_{\Delta t}^{ma},\ \lambda_{\Delta t}^{mb},\ \lambda_{\Delta t}^{ca},\ \lambda_{\Delta t}^{cb}\}$	*average intensity of each type*
$v_9 = \{\mathbf{1}_{\{\lambda_{\Delta t}^{la} > \lambda_{\Delta T}^{la}\}},\ \mathbf{1}_{\{\lambda_{\Delta t}^{lb} > \lambda_{\Delta T}^{lb}\}},\ \mathbf{1}_{\{\lambda_{\Delta t}^{ma} > \lambda_{\Delta T}^{ma}\}},\ \mathbf{1}_{\{\lambda_{\Delta t}^{mb} > \lambda_{\Delta T}^{mb}\}}\},$	*relative intensity indicators*
$v_{10} = \{d\lambda^{ma}/dt,\ d\lambda^{lb}/dt,\ d\lambda^{mb}/dt,\ d\lambda^{la}/dt\},$	*accelerations(market/limit)*

Different situations

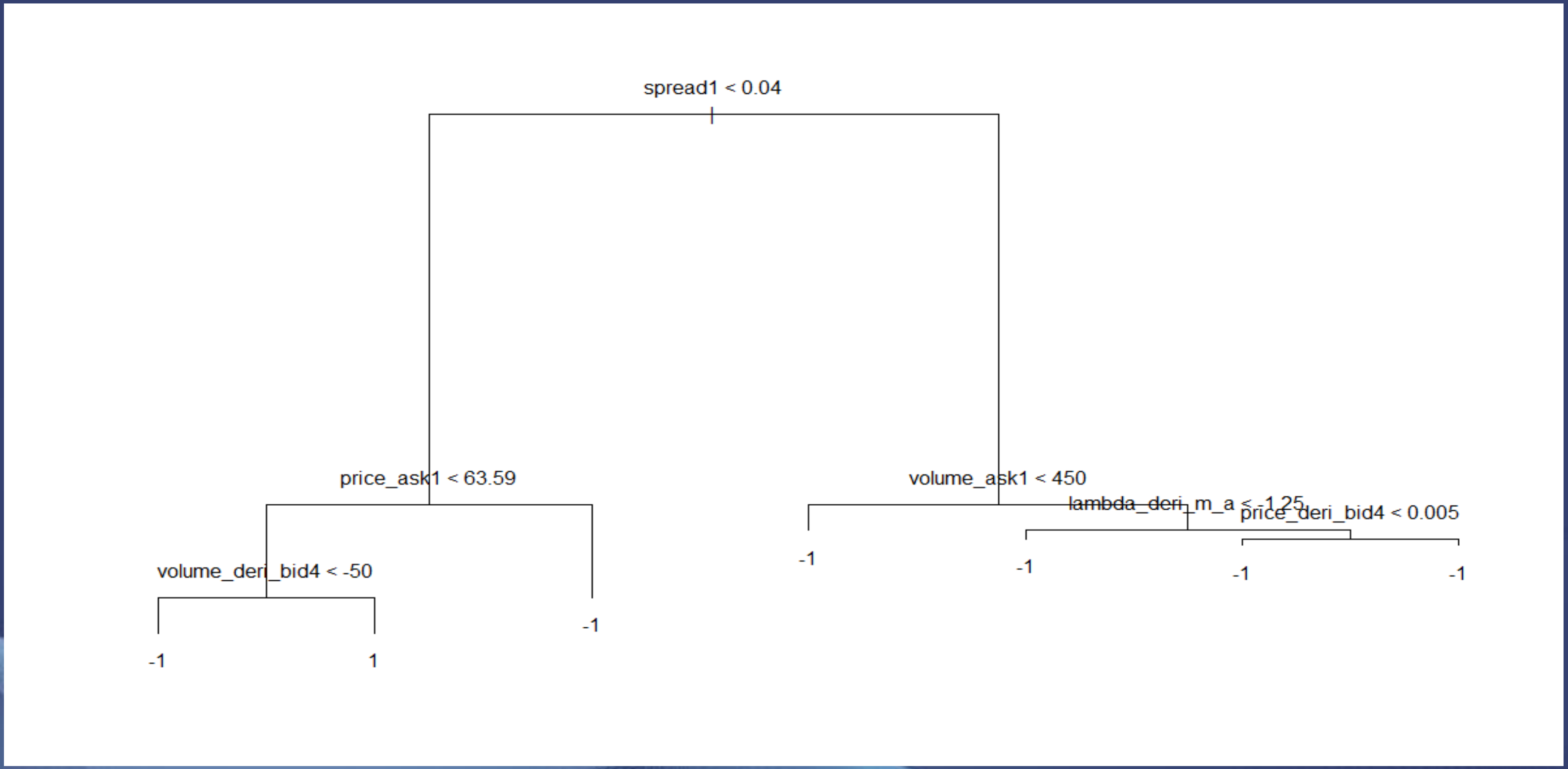


- Situation1: up and stationary
- Situation2: down and stationary
- Situation3: up, down and stationary

Sample Stock

- Tractor Supply Company (up case)
- DuPont (down case)
- Exxon Mobil (regular case)

Single Tree



Stationary: -1 Up: 1

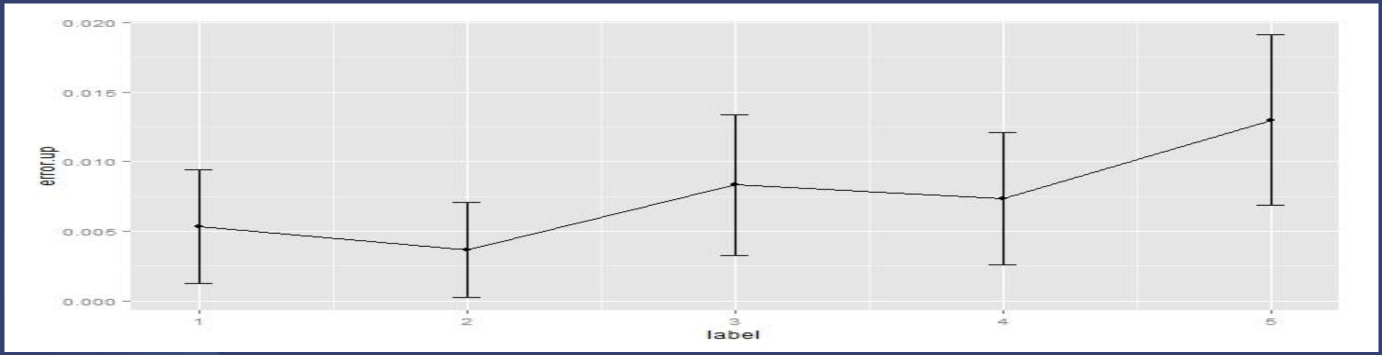
In “up case”, spread is the most significant feature.

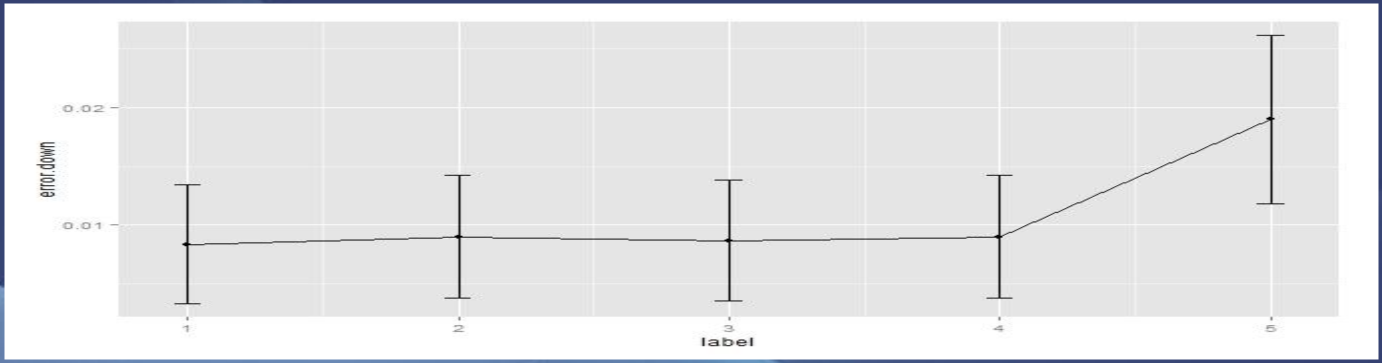
Random Forests

- Right: sample prediction table calculated from training data
- 1: up
- 0: stationary
- All the classifiers give the similar results in general.
- It is easy to make mistake classifying spread-crossings as stationary state.

predict \ truth	0	1
0	2621	27
1	1	351

Accuracy (10-fold Cross Validation)

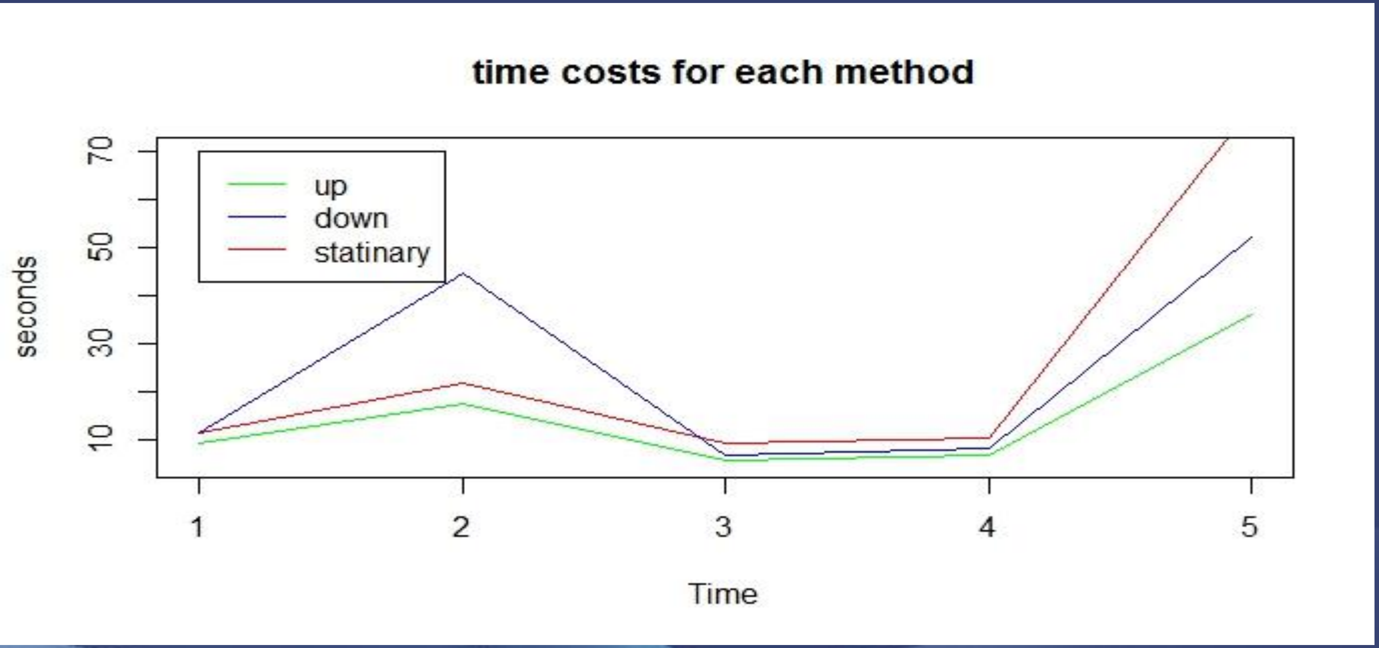




1: simple RF 2:complex RF 3:linear SVM 4:cubic SVM 5:radial SVM

- Random forests are competitive.
- Need to deal with over-fitting issues.

Time Costs

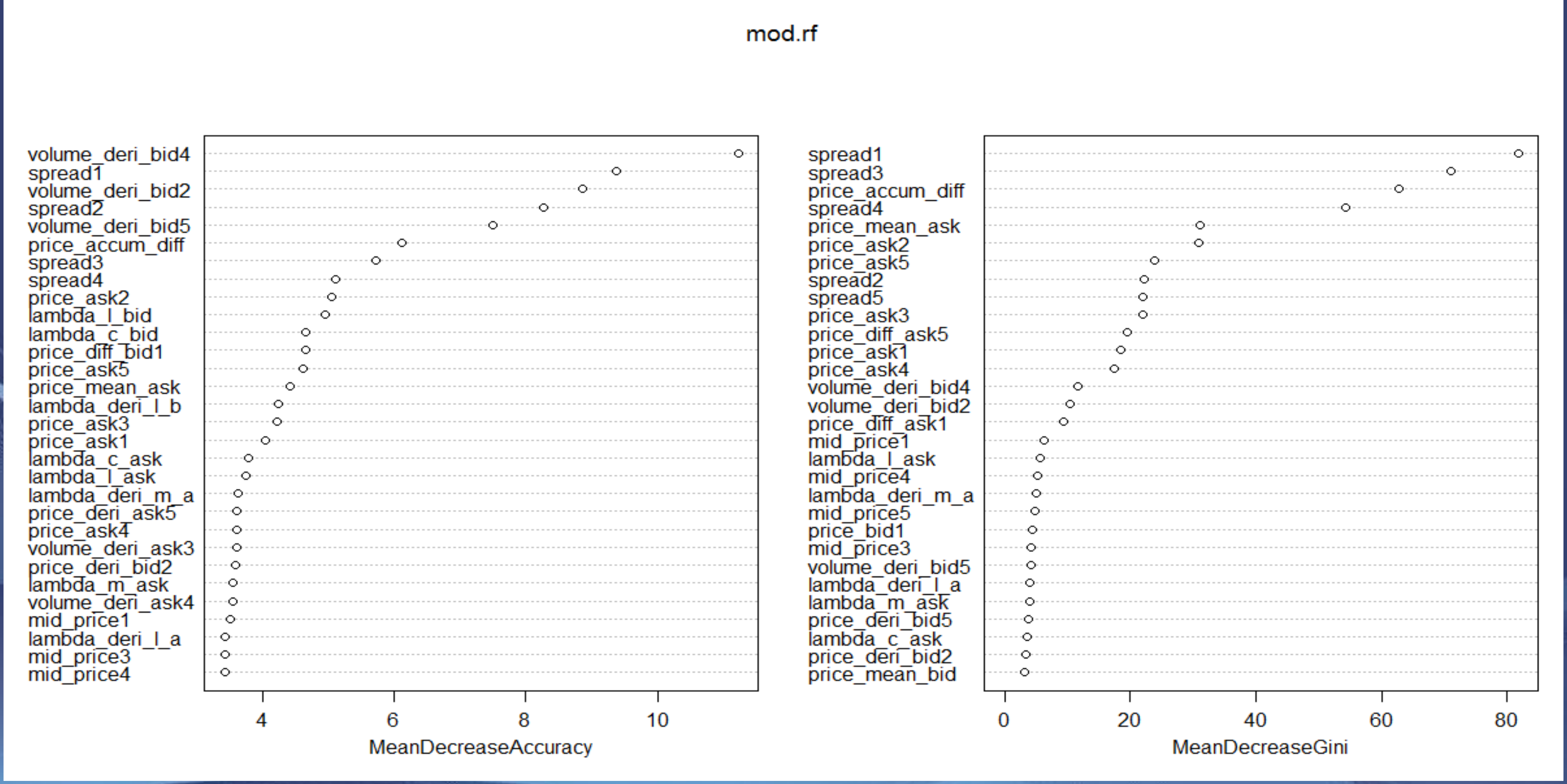


1: RF (50 trees) 2:RF (200 trees) 3:linear SVM 4:cubic SVM 5:radial SVM

- Random forests cost more time to run.
- Two-class problems save time for SVMs.

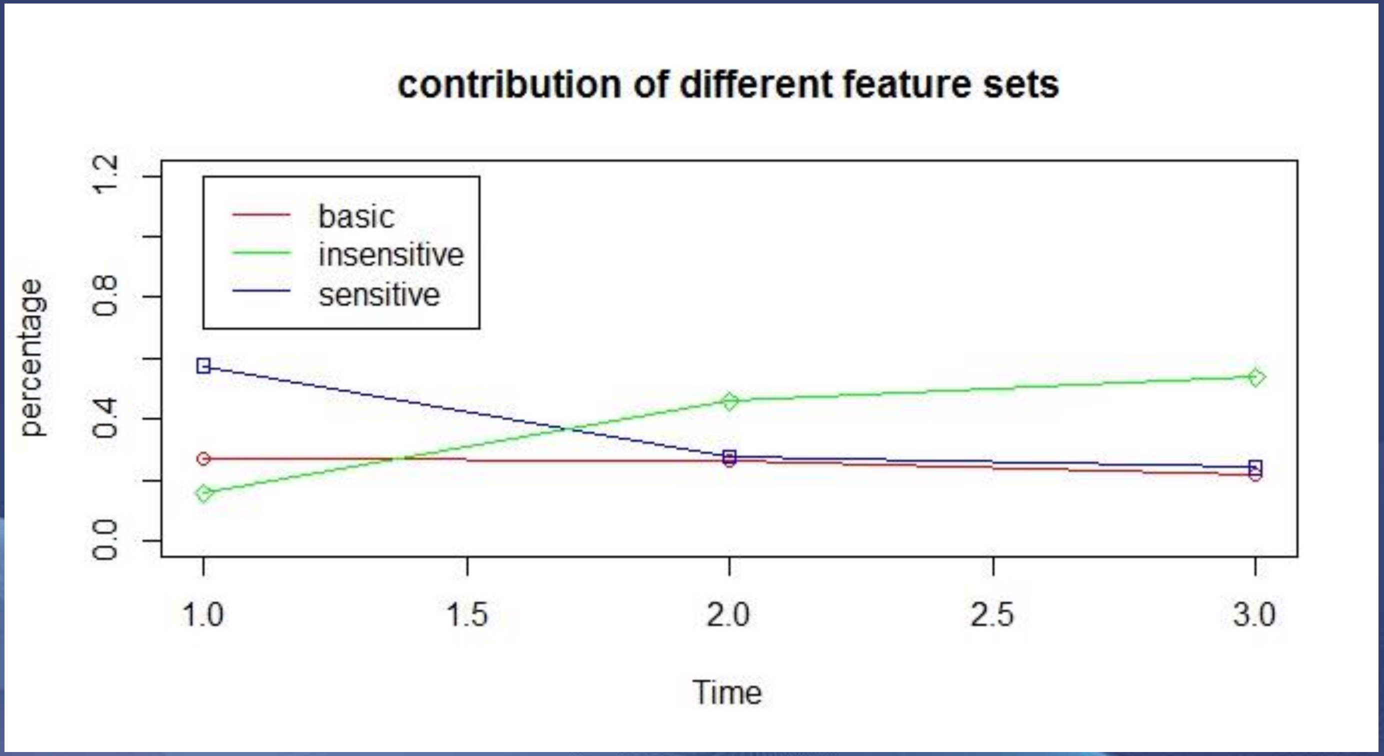
Feature Importance

Most of features don't have much contribution.



Feature Importance

All of those three feature sets make differences.



1: “Up” case 2: “Down” case 3: “Regular” case

Feature Importance

top 10 features from random forests vs top 5 features from Lasso

Sample1 (up case):

Random forests

spread1	spread3	price_accum_diff	spread4	price_ask3
5880.120	4430.281	4101.554	3531.477	2745.262
price_ask1	price_mean_ask	price_ask5	price_ask2	spread5
2741.408	2619.677	2603.736	2573.764	2524.812

Lasso

price_diff_ask5	price_diff_bid5	spread1	price_ask1	price_deri_bid1
-7.642773e+01	-3.728444e+01	-3.503420e+01	-2.052818e+01	-1.152060e+01

Sample2 (down case):

Random forests

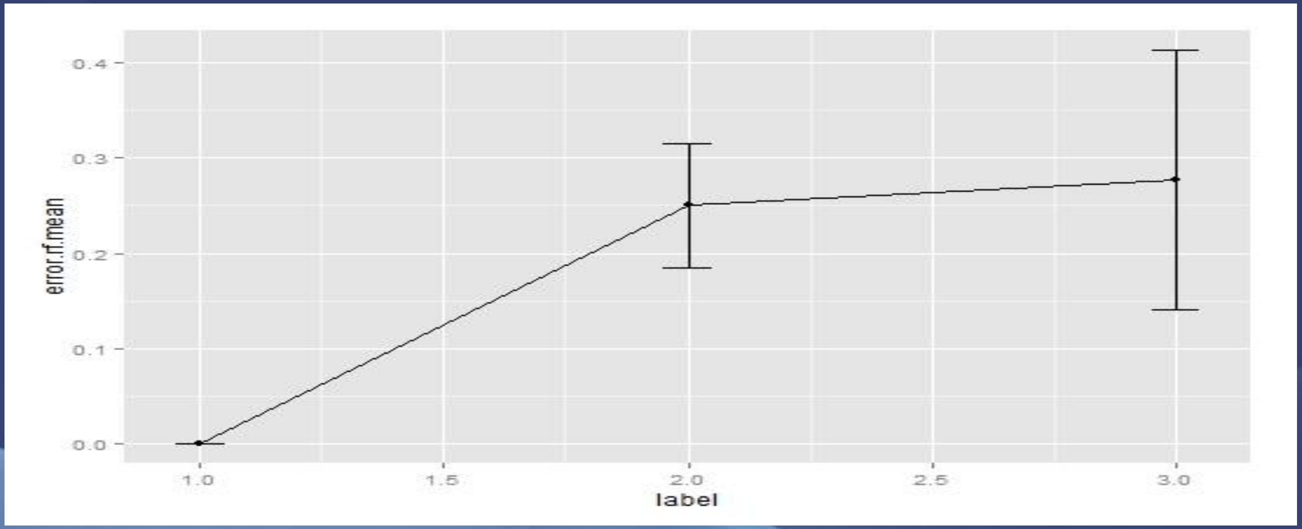
volume_deri_bid4	price_diff_bid1	volume_deri_bid3	volume_ask2	lambda_m_bid
9205.707	5399.919	5192.244	5137.731	4045.056
volume_bid1	volume_deri_bid5	lambda_l_bid	volume_ask4	spread1
3189.179	3025.360	2633.816	2374.771	2353.565

Lasso

price_diff_ask1	price_diff_ask3	price_diff_bid3	lambda_m_ask	price_bid1
2.782643e+14	1.070470e+14	1.513864e+12	1.245938e+03	1.615552e+01

Robustness Study

Build random forests models with 1500 observations and test for new observations in next 1, 5 and 10 seconds. Repeat for 10 times.



Generally, the test error is getting larger when increasing the time period for prediction. The variance is increasing as well.

Robustness Study

We divide 3000 observations within 3 minutes into 4 subsets in the order of time. Then we fit random forests model to each subset with 750 observations to get top ten features.

price_deri_ask4	price_deri_ask3	price_deri_ask1	lambda_l_ask	price_deri_ask5
24.00489	17.91012	15.65325	15.36460	15.09402
spread1	spread4	lambda_l_ask	lambda_m_ask	lambda_c_ask
26.91964	23.80486	19.04082	17.07954	15.43593

lambda_deri_m_a	price_deri_bid5	price_deri_bid4	volume_ask1	price_ask1
4.085210	2.680605	2.492422	2.388728	2.086369
price_ask5	price_ask2	mid_price5	mid_price1	price_mean_ask
1.757893	1.631791	1.513661	1.476331	1.183117

lambda_deri_m_a	price_deri_bid5	price_deri_bid4	volume_ask1	volume_mean_ask
4.29959326	2.84549321	2.65443566	2.50183664	0.90594729
lambda_l_ask	lambda_deri_l_a	volume_deri_ask1	lambda_c_ask	lambda_l_bid
0.50689173	0.50637678	0.16392072	0.03472007	0.03397282

lambda_deri_m_a	price_deri_bid4	price_deri_bid5	volume_ask1	volume_mean_ask
4.39174218	2.92531263	2.69412199	1.75392579	1.68804661
lambda_l_ask	lambda_deri_l_a	volume_deri_ask1	lambda_l_bid	lambda_c_ask
0.51965882	0.33839811	0.14381480	0.04452700	0.03642339

We do obtain some robustness since most of the top ten features are about price, volume and arrival rates. But the order is changing all the time.

Conclusions

- Trees gives clear interpretation.
- The prediction results and time costs obtained by Random Forests are competitive.
- All of those three sets of feature make contributions. Basic sets are more stable. Lasso may give different feature selection results.
- Models are not robust, with large variance in terms of prediction. But it is robust for feature importance in a short time period within a couple minutes.

Problems and Feature Work

- Get optimal number of time steps to be predicted.
- Build feature selection methods to improve models in terms of prediction accuracy and time cost.
- Deal with correlation among features.
- The response variable y (the spread crossing metric as up, down and stationary) may not be independent.
- Traditional cross validation methods no longer work.

Thank you.