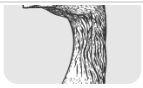
Algorithmic Trading Challenge

Develop new models to accurately predict the market response to large trades.

$10,000 · 111 teams · 6 years ago



karmic_menace
74th place

Options

posted in Algorithmic Trading Challenge 6 years ago

Congratulations to the top winners! It has been a fun competition. Question for kaggle (this is my first competition, so bear with me if it has been answered before). Would Kaggle publish the top winning algo/code ? And it would be nice if there is a way for other competitors willing to share their algo/code with other interested participants.The final swing in the results was interesting! I didn't expect that. May be you should hold another betting competition for each of these to predict the winner :)

Comments (58) Sort by Oldest

Please sign in to leave a comment.

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

0

Congratulations to Ildenfons with convincing first place and other top teams with good results.

@karmic - actually, change in teams position after final scoring was not as big as in some other competitions (it was known for a while that model by Xiaoshi Lu was overfitted to public test set).

I expect some details of top models to be known soon. I can say that simple correctly executed linear regression model could put you in the top ten. (However it was not our final model)

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

Congratulations to all top teams as well. I can vouch for Sergey's assertion that linear regression can place in the top 10, however I'm not sure what 'correctly executed' means wrt this data.

I am very interested to learn the extent to which others addressed the obvious differences in the test and training sets. In particular, it appears there was a regime shift after day 2 in the training data, with prices becoming much 'noisier' (I can quantify if interested). Scores for models from subsets of day 1 and day 2 data statistically similar to the testing set are close to the testing scores.

I am also interested if others developed distinct models for shocks near the opening (I had separate models for t<60 & t>60).

karmic_menace • (74th in this Competition) • 6 years ago • Options

0

Cole/Sergey - I am definitely interested in knowing more on the details of linear reg. model. I must have missed the forest looking for tree. I know Niel also spoke about linear reg. model that gave him pretty good results. I was also surprised this competition had less turn out than others.

Capital Markets ... • 6 years ago • Options

0

Congratulations Ildefons for winning the main prize. We have some extremely talented contestants in this competition and I would like to thank them all for their contribution and insights.

Christopher Hef... • (4th in this Competition) • 6 years ago • Options

4

This was an interesting contest! Many thanks to the organizers & other competitors, and congratulations to Ildefons. Before discussing models, I thought I'd start a discussion about the data itself & how it generally impacted peoples' modeling approaches. So here are some observations of my own, in no particular order:

Observations

1. Bids/asks from T=1...T=47 seemed to provide little predictive value. My variable-selection algorithms dropped them. In the forums, I noticed others mentioned that they also saw little value in using these prices.

2. The error contribution right at the market open (at 8AM) was extremely large. For one model, I found 12% of squared error for the entire trading DAY occured in the first MINUTE of trading. I trained a seperate model for the open (the naive benchmark worked better than a regression at the open, for example) and got about a 0.0050 improvement, best case.

3. I didn't see price "resiliency" that the organizers discussed. Some of the examples the organizers posted showed stock prices bouncing back to pre-liquidity-event levels; we did not see this on average. Looking at the trade data in aggregate (via time averages, and various PCAs), we saw that for buys, the ask price jumped up immediately due to the liquidity event, and the bid price jumped up one time period later, and then both the bids & asks rose very slowly. The opposite happened for sells.

4. For some of our models, we found that training a separate model for each stock _underperformed_ training a general model for all stocks. So a per-stock model was not necessarily a big winner, as we first suspected.

5. Prediction accuracy varied across time. Using a holdout set & one of our models, I found that the error rose as you got farther from the liquidity-event trade. The RMSE was about 0.4 at T=52, rising to over 1.6 at T=100. RMSE rose roughly with sqrt(t), which, to me, implied some random-walk behavior away from the known prices at T=51.

6. The "liquidity event" trades did not seem to impact prices very much. Roughly 99.7% of the time, the VWAP was exactly equal to the best bid or ask at T=50. If there was a huge trade that ate through multiple levels of bid or ask prices, I would expect the VWAP to be different than the inside bid/ask immediately after the trade. It might have been somewhat more interesting if the trading data had some more large, market-moving trades.

Suggesetions for Improving the Contest

There were a few things that I thought could be changed to improve this contest; others have mentioned these, but I'll reiterate them:

7. The sampling methods used to create the testing & training were different, and from my perspective, it would have been easier if they were sampled same way. The proportions of each security in testing vs training differed, of course. Also, the testing set was in random order, so why not also randomize the training set? One could correct for these testing vs training set differences by using different, per-stock weights for each row of data, or creating per-stock models. But this seemed like extra work that could have been avoided with uniform sampling. In the end, it took time away from focusing on the main goal of predicting the price behavior of the stocks.

8. The average prices for stocks in the dataset varied by a couple order of magnitudes, and when this was combined with the RMSE metric, this meant that high-price stocks (which contributed most to RMSE) dominated. For example, stock 75 -- with the highest price -- gave 36% of all squared error for one of our models. If the price data we were given was normalized (say, by dividing all prices by their VWAP), then perhaps the resulting models would be more generalizable across all stocks, regardless of price..

Everything considered, I thought this was an interesting contest in a "hot" area in finance. I look forward to reading about what others found & did to create their models!

Anil Thomas • (4th in this Competition) • 6 years ago • Options

0

Congratulations to Ildefons. Having grappled with this dataset over weeks, I can attest that an RMSE under 0.77 is a tremendous achievement.

As a side note, Kaggle allowing submissions past the deadline is a great service for the contestants. I know I will be playing with this data a bit longer.

Capital Markets ... • 6 years ago • Options

1

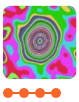
Christopher, thank you for your suggestions for improving the competition. Regarding (7) we were faced with somewhat of a conundrum. We wanted to release full tick data for the training set under the assumption that more information could lead to more comprehensive models. Initially we were going to release full tick data for testing however we then realized that this would inadvertently reveal solutions. Our end solution was somewhat of a compromise and we acknowledge there is room for improvement here.

Regarding (8) high priced stocks do have a disproportionate effect on RMSE. Again there is somewhat of a need to compromise. Suppose we normalize by dividing high stock prices by some factor. This will depress p*value. Or if we leave p*value unchanged this will distort the relationship between p_value and price. Once again we acknowledge that were we to run this again we would be able to improve implementation in this area.

Capital Markets ... • 6 years ago • Options

0

Hi Neil, passion and talent is a combination we like to see. We are glad that you wish to continue working with the data post competition. To all our top Kagglers, if you wish to explore ways to continue to build and extend your modelling efforts please contact me at dnguyen@cmcrc.com. The CMCRC has a commercialization arm in place. If you have a model with good predictive power I would very much like to discuss further opportunities if that is an avenue which you wish to pursue.



Ali Hassaïne • (8th in this Competition) • 6 years ago • Options

0

Congratulations to the winners !

My best submission is slightly better than what I picked. Did you guys select your best submission?

Kaggle public leaderboard gives you feedback to further investigate some techniques rather than others, but the public leaderboard of this competition is somewhat special, I spent too much time investigating the wrong techniques

Cole Harris • (9th in this Competition) • 6 years ago • Options

2

Additional/expanded observations

A histogram of the liquidity shock times for the initial and final testing data had a sharp peak at t<60 s, and then very flat from ~ 6 minutes through end of day.

Partially because of this, and also due to other similarities, for most of the competition I trained with the initial testing set (last 50k rows in training). It looks like most of the initial testing data, all of day 1 & 2 data, and all of the final testing data follow similar dynamics, while data from day 3 on looks different.

Towards the end of the contest I switched to training with subsets of day 1 & 2 data sampled to match testing distributions. This resulted in better predictions, but in the end I think I may have spent too little time on the (t>60) models. It has occurred to me that the ability to quickly identify this regime change could be useful. Wrt my working on this data, this may only be the beginning:)

Careful attention to t<60 models resulted in a 1.4% overall final score improvement vs naive constant, so perhaps I did something more useful here.

Again, thanks to the organizers and fellow competitors.

Cole Harris • (9th in this Competition) • 6 years ago • Options

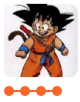
1

@Ali

I am surprised by the high correlation in public and private scores. I had done some modeling that led me to think the disparity would be much worse. Because of this, late in the competition, I tried very hard not to make too much of the public score, and to evaluate models based only on (out of sample) training data results. In the end the model I would have picked as doing best came in 2nd of my models, and the model I would have picked as 2nd came in first. But then the public leaderboard scores would have led you to the same conclusion.

It may be that the winning model was not selected?

The topic of public vs private leaderboard results and how to evaluate as a competitor is worthy of investigation.



Ildefons Magrans • (1st in this Competition) • 6 years ago • Options

0

This is an awesome way to wake up ! :-)

Congratulations to everyone that competed and thank you very much to the CMCRC team for setting up this competition and the support.

ZKC • (12th in this Competition) • 6 years ago • Options

0

Congrats Ildefons Margrans!

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

0

Now, I think, will be a good time to learn what would be the score of internal model by Capital Markets. (preferably trained on the same training data set (or subset of it))

I am wondering if anybody managed to use Neural Network successfully. In all our attempts NN did not performer better than Linear Regression. (At the end our model was combination of LR and Random Forest)

alegro • (2nd in this Competition) • 6 years ago • Options

0

Congratulations to the winner!

William Cukiers... • (4th in this Competition) • 6 years ago • Options

0

> ***Sergey Yurgenson wrote***
>
> Now, I think, will be a good time to learn what would be the score of internal model by Capital Markets. (preferably trained on the same training data set (or subset of it))
>
> I am wondering if anybody managed to use Neural Network successfully. In all our attempts NN did not performer better than Linear Regression. (At the end our model was combination of LR and Random Forest)

Agreed, and if it's really a 0.4 we want to see the code to check for signs of black magic :)

Re NN: I have yet to apply a NN with any real success in any facet of my work, etiher research or on Kaggle. I think they are just one of those methods that require a high level of expertise to set up properly. Not that they can't perform well (and they seem to be coming back into fashion in academia), but they aren't for the casual tinkerer in the same way that other methods are.

alegro • (2nd in this Competition) • 6 years ago • Options

1

> ***Christopher Hefele wrote***
>
> 2. The error contribution right at the market open (at 8AM) was extremely large. For one model, I found 12% of squared error for the entire trading DAY occured in the first MINUTE of trading. I trained a seperate model for the open (the naive benchmark worked better than a regression at the open, for example) and got about a 0.0050 improvement, best case.

Three biggest outliers (ranked by impact on simple linear regression model) from all data (train+public+private) those correspond to the out of market conditions (before market openning) unfortunately were included in the private test set. In a case of improper handling of the conditions these 3 rows may make a great impact on the private score (~0.07-0.15 depending on model).

This is fragments with row_id's 758422,759056 and 769050 (I did not checked the id's, in a case of doubts that they are right, ask here, I will check). If someone with big difference between private/public scores (>= 0.07) interested he can look on the predictions of these rows, correct them (by filling by the bid50,ask50 values for example) and repost their predictions to check the difference.

BTW. Just curious has someone confident predictions on far horizons? Probably no. If it is interesting it is possible to fill horizons 26..50 (bid76/ask76..bid100/ask100) by values of the previous prediction (bid75/ask75) and check the score difference.

Vik Paruchuri • (5th in this Competition) • 6 years ago • Options

4

Thank you, CMCRC, for creating such an engaging and interesting competition. I would also like to thank all of the competitors for creating such a competitive atmosphere. Congratulations to Ildefons; you did an excellent job.

I ended up not selecting my best score, but my private leaderboard score seems to have improved throughout the contest. I have just started learning statistics and R, and I picked up quite a lot throughout this competition. I hope to have the chance to compete against all of you in the future.

The data that we were provided with for this competition was very interesting in many respects. I would like to discuss Christopher and Cole's observations before getting into some of my own points.

1. I noticed that although most of the predictive value was concentrated after t=45, after correcting t=0 to t=45 for outliers(most notably errors right after market open, which I will get into later), there was some predictive value to be had in these values. Predictors based on data from t=45 to t=50 also had some unfortunate nonlinear tendencies, which were minimized when using longer time horizons. Observations from t=0 to t=45 were also very useful for creating predictors based on volatility.

2. Much of the daily error was concentrated right at market open. There were secondary areas of high error and volatility at 10:30, 13:30, and 15:00. There was a theory posited on these forums that these secondary areas were a result of other markets opening, which created arbitrage opportunities. The arbitrage opportunities may have unfolded in a predictable fashion, but I did not have time to solve this issue.

3. I also found that a "per stock" model underperformed a model trained on the entire data set. Models trained on specific subsets of time(because there was a definite time of day effect) also underperformed models trained on the entire set.

4. I found that this competition, because of its design, came down to two distinct predictions problems. Competitors had to predict the bid-ask spread as it recovered from t=51 to t=100, but also the trend of bid and ask prices from t=51 to t=100. As one may expect, the trend was much harder to predict than the spread. The fact that the trend could be predicted at all is interesting, but the predictive abilities of the models I tried were not extremely strong, partially because it was difficult to test my trend models(testing on out of sample training data was problematic, because the training data has similar trend characteristics throughout, whereas the test data, sampled in a different way, had drastically different trends). The spread, by contrast, had much more uniform characteristics across both the training and the testing sets.

5. There were several dozen rows in the training and testing sets that were possibly erroneous. These observations all occured at market open, and in them, the bid-ask spread was huge(up to 600 for one stock!). These large spreads affected linear models unless they were corrected. Their spreads from t=51 to t=100 did not recover in the same fashion as the typical spreads, and a significant portion of the error came from these rows. A full list of the rows that i suspect were erroneous in the testing set is attached to this post. Note that I used an automated methodology to select these rows, so not all of them may be erroneous. While there were approximately 200 of these rows in the testing set, there were only about 250 in the training set, which made them extremely hard to predict. The final outcome was doubtless influenced heavily by these rows, as alegro has pointed out. I noticed swings of up to .008 on the public and private leaderboards by varying the predictions on these rows alone. It appears that the 70% of the test set that was held out for the private leaderboard score contained significant outliers(as evinced by the higher private RMSE scores vs public, although I could be incorrect). Correcting for these outliers(or perhaps even one or two observations) was a primary goal for many competitors, I am sure.

6. I agree that the different sampling methods for the training and the testing sets affected the outcome of the competition heavily. The testing set was biased towards observations from the beginning of the day. The training set contained a large proportion of data from the beginning of the day, but also had a lot of observations from the end of the day, leading to a somewhat U-shaped time vs amount of observations graph. Attempting to correct for these variations did not aid my model, however.

7. The algorithm that was used by the CMCRC data provider to clean the data prior to it being delivered to us was also relevant to the competition. I used time series outlier filtration, which uses a process found in a few papers that involves moving averages and a distance of 3 standard deviations to filter large outlying values from tick data. I found very few observations outside of this threshold. Additionally, all the ticks were in order, and none, aside from a few at the beginning of the day, could be unequivocally called "bad." Knowing how the data was removed and filtered prior to is being delivered to us might have impacted the accuracy of our predictors.

8. Volume information would have been a huge boost to the predictive capability of my model, and I am sure those of many others. It would have aided in establishing how deep the limit book was at any given time. I would suggest that future competitions focused on predicting liquidity shocks address the outlier problem, shorten the prediction time horizon to 25, and give volume and timestamp information for every trade.

9. The data exhibited significant heteroskedasticity, but I had little luck solving the issue with weighted models, clustering, or "per-stock" models. Feature selection helped to mitigate the issue by minimizing

predictors that showed significant heteroskedasticity, but I was never able to solve the issue to my satisfaction. Did anyone manage to do so?

Again, thank you to CMCRC and all the competitors for creating such an engaging opportunity. I am very interested in finance and algorithmic trading, although my background is not necessarily in the field, and this was a good way to model tick data, which is usually very hard to obtain.

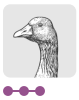
outliers.txt (1.07 KB)

Christopher Hef... • (4th in this Competition) • 6 years ago • Options

0

> ***Capital Markets CRC wrote***
>
> Regarding (8) high priced stocks do have a disproportionate effect on RMSE. Again there is somewhat of a need to compromise. Suppose we normalize by dividing high stock prices by some factor. This will depress p*value. Or if we leave p*value unchanged this will distort the relationship between p_value and price. Once again we acknowledge that were we to run this again we would be able to improve implementation in this area.

Agreed, and I acknowledge framing a competition involves a lot of difficult compromises. Perhaps another way to address this issue in future competitions might be to change the evaluation metric instead of the data -- for example, use RMSLE (root-mean-square of the difference between the logs of the prices), or the RMS of (predicted_price/actual_price) -1.



BarrenWuffet • (42nd in this Competition) • 6 years ago • Options

0

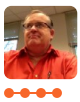
Anyone that had success with linear models want to elaborate on them? I had very little success with them (in R glm, glmnet, lm) and am curious where I went wrong.

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

> ***Christopher Hefele wrote***
>
> > ***Capital Markets CRC wrote***
> >
> > Regarding (8) high priced stocks do have a disproportionate effect on RMSE. Again there is somewhat of a need to compromise. Suppose we normalize by dividing high stock prices by some factor. This will depress p*value. Or if we leave p*value unchanged this will distort the relationship between p_value and price. Once again we acknowledge that were we to run this again we would be able to improve implementation in this area.
>
> Agreed, and I acknowledge framing a competition involves a lot of difficult compromises. Perhaps another way to address this issue in future competitions might be to change the evaluation metric instead of the data -- for example, use RMSLE (root-mean-square of the difference between the logs of the prices), or the RMS of (predicted_price/actual_price) -1.

Ideally the metric would reflect the potential monetary benefit derived from the use of the algorithm. Likely a trader would, all things being equal, trade relatively more low priced shares vs high priced shares, so a weighted metric is appropriate. Christopher's metrics accomplish this.



Bruce Cragin • (6th in this Competition) • 6 years ago • Options

0

> ***alegro wrote***
>
> Three biggest outliers (ranked by impact on simple linear regression model) from all data (train+public+private) those correspond to the out of market conditions (before market openning) unfortunately were included in the private test set. In a case of improper handling of the conditions these 3 rows may make a great impact on the private score (~0.07-0.15 depending on model).

Alegro, would you please clarify how you identified these 3 test set rows as outliers? Was it just that they had similar properties (e.g. very early trading time of day) to outliers found in the training data? Thanks!

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

6

Ok. Here is the secret recipe for Linear Regression meal:

go to your friendly neighbor datastore and choose couple fresh pieces of data (day1 and last 50k)

cut out bones and extra fat (leave only columns 5 170 206 207)

cook separately "seller initiated" transactions and "buyer initiated" transactions using your favorite linear regression function (do it separately for each askN and bidN to be predicted)

Use 200 created LRs to calculate required predictions and nicely plate them into submission file.

Serve hot, because you do not want to miss 0.77590 public score and 0.77956 private score

:)

leazar • (61st in this Competition) • 6 years ago • Options

0

Hey Sergey.....sounds quite tasty :)

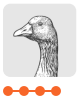
Is there a reason that you picked those columns for your regression model? Was this by trial and error?

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

> ***Sergey Yurgenson wrote***
>
> Ok. Here is the secret recipe for Linear Regression meal:
>
> go to your friendly neighbor datastore and choose couple fresh pieces of data (day1 and last 50k)
>
> cut out bones and extra fat (leave only columns 5 170 206 207)
>
> cook separately "seller initiated" transactions and "buyer initiated" transactions using your favorite linear regression function (do it separately for each askN and bidN to be predicted)
>
> Use 200 created LRs to calculate required predictions and nicely plate them into submission file.
>
> Serve hot, because you do not want to miss 0.77590 public score and 0.77956 private score
>
> :)

My best result was with a similar algorithm wrt separate buy/sell & bid/ask at each post shock time 51-100. But my model was more complicated: incorporating more predictors and designed to predict a windowed mean price rather than a price at a particular time point. Curious, how did you identify your predictors (bid41)? Your training data subset?



alegro • (2nd in this Competition) • 6 years ago • Options

0

> ***Bruce Cragin wrote***
>
> Alegro, would you please clarify how you identified these 3 test set rows as outliers? Was it just that they had similar properties (e.g. very early trading time of day) to outliers found in the training data? Thanks!

These rows are outliers in histogram of per row RMSD's of a model responses (predictions) against naive model (bid50/ask50). Manual investigation shows that they have unusual pattern (with values) of predictors that does not represented in the train data. Changing predictions for these rows to the naive model values does not change public score (private score changed from 0.85444 to 0.77965).

The model used in the experiment above is not complex linear model. Average of two runs of this model with different parameters wins the secondary milestone.

Bruce Cragin • (6th in this Competition) • 6 years ago • Options

0

Interesting approach! Thanks for sharing these details.

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

0

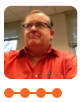
Cole Harris wrote

> *Sergey Yurgenson wrote*
>
> Ok. Here is the secret recipe for Linear Regression meal:
>
> go to your friendly neighbor datastore and choose couple fresh pieces of data (day1 and last 50k)
>
> cut out bones and extra fat (leave only columns 5 170 206 207)
>
> cook separately "seller initiated" transactions and "buyer initiated" transactions using your favorite linear regression function (do it separately for each askN and bidN to be predicted)
>
> Use 200 created LRs to calculate required predictions and nicely plate them into submission file.
>
> Serve hot, because you do not want to miss 0.77590 public score and 0.77956 private score
>
> :)

My best result was with a similar algorithm wrt separate buy/sell & bid/ask at each post shock time 51-100. But my model was more complicated: incorporating more predictors and designed to predict a windowed mean price rather than a price at a particular time point. Curious, how did you identify your predictors (bid41)? Your training data subset?

Our initial assumption was that last 50k was the best approximation to the test data sets. (At the end of competition part of our submissions was trained on "random" subsets designed to emulate test dataset). After some analysis we decided that day1 was "closer" to the last 50k and test set than other days. Thus we used day1+50k as training dataset for many submissions. For simple crossvalidation one can train models on day1 and validate them on last 50k. I do not remember testing any models without columns 206, 207, choice of other predictors was result of stepwise regression and crossvalidation.

I have to point out that we did not choose that model for our final submission.



Bruce Cragin • (6th in this Competition) • 6 years ago • Options

0

Just to add a note to what Sergey said, the random subsets he mentioned (each of which had the same stock and buyer/seller-initiated counts as the test set) were in fact each drawn from the full training file. In the end, those models were quite competitive with the Day 1+last 50k-trained ones.

Cole Harris • (9th in this Competition) • 6 years ago • Options

9

Attached are histograms of the number of total (bid and ask) price changes for various subsets of the training data.

This result led me to think (1) something happened after day 2, (2) the final testing data was sampled at a time similar to days 1 & 2, and (3) the initial testing set was sampled primarily at a time similar to days 1 & 2 as well.

Matching this statistic between training and test produced training scores much more in line with leaderboard scores.

npcplots.pdf (67.03 KB)

Neha Lahri • 5 months ago • Options

0

what is "price changes" ?
I have checked for tick to tick price change (bid[n]-bid[n-1]) , but I am not getting a plot similar to what you have shared.

Bruce Cragin • (6th in this Competition) • 6 years ago • Options

0

Cole, your two peaks in number of changes correspond closely to a quality I called easy or hard to predict (based on rmsd of a backward model prediction of the Day1 - Day6 data using the last 50000 as the training set). The plots below show, for each stock, the distribution of row_id's of a selection of the hardest (highest rmsd) and easiest to predict cases. The data points congregate in such a way that you can easily identify the various days. It's clear that for most if not all stocks Day 1 and perhaps to a slightly lesser extent Day 2 have many more of the easy to predict cases, in agreement with your result. I tried using only easy training data, using only hard training data etc, but did not find a combination that seemed to be especially effective, given that the distribution of easy vs. hard in the test set is fixed. But it's quite possible this was a missed opportunity...

plots2.zip (675.7 KB)

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

> ***Bruce Cragin wrote***
>
> I tried using only easy training data, using only hard training data etc, but did not find a combination that seemed to be especially effective, given that the distribution of easy vs. hard in the test set is fixed. But it's quite possible this was a missed opportunity...

Wrt the contest metric, I also think we missed an opportunity. On a more practical level, it seems very useful to have a means of determining a confidence level in your predictions.

Christopher Hef... • (4th in this Competition) • 6 years ago • Options

1

Given the discussion above, I wish I had paid more attention to individual rows that were outliers! I just did a quick experiement to see how much a few rows might dominate RMSE. I used the naive predictor to make some predictions on the last 50K lines of the training dataset, and then I plotted the cumulative squared error across rows.

The resulting plot is attached. It shows that errors are pretty concentrated, as we knew. But here are some numbers to back up that observation: 11% of all squared error was contributed by the worst 10 rows (out of 50K rows), 30% was contributed by the worst 100 rows, and 60% was contributed by the worst 1000. So it seems that improved predictions on just a few key rows could improve one's RMSE quite a bit.

Next, instead of using price errors (e.g. Price1-Price2), I tried using "log-errors" --- actually, log(Price1) - log(Price2) --- to see if that would be a better error metric to use with RMS. It was somwwhat better: 3% of all squared "log-error" was contributed by the worst 10 rows, 11% was contributed by the worse 100, and 35% was contributed by the worst 1000.

Comparing this 'log error' result to the 'regular' errors, one can see that about 30% of all error was caused by the 1000 rows when 'log-errors' were used, but that same ~30% percent of all error was caused by only 100 rows with 'regular' error. So 'log-errors' seem 10x less concentrated, at least in this toy example. Nevertheless, some subset of rows still dominate, regardless of what metric is used.

error_row_distribution.pdf (1.53 MB)

Anil Thomas • (4th in this Competition) • 6 years ago • Options

1

Lots of good information on this thread... still haven't digested all of it. Kudos to the clever folks who were able to pinpoint the outliers down to the row numbers. I tried to handle the outliers in a more generic fashion, which resulted in some improvement, but obviously not enough to win.

It was clear from the start that the training data does not represent the test data. I tried to make a subset of the training set by following the same steps that the organizers must have followed to make the test set. While the test set did not contain any time window overlaps, there were several overlaps in the training set. After filtering out the rows with time overlaps, there were about 150K rows left. Out of these, I picked 50K random rows to train on. The original intent of this second filtering was to speed up the training process, but the pared down 50K turned out to be as good as the whole 150K, as per prediction accuracy. In the end, I don't think eliminating the overlaps helped much. The results might have been pretty much the same

with a randomly picked subset.

I see references to day 1, day 2 etc. in the posts. How does one identify data from the same day? Just pick clumps of rows for a security that has the same number of trades on the previous day? Is the assumption that all securities were sampled on each day?

My initial impression was that the prediction accuracy would inversely correlate with volatility of the prices during the first 50 events. I still think this is true (easy to check, just haven't gotten to it), but wasn't able to capitalize on this. Categorizing the rows based on variance of the bid and ask prices and then training and predicting each category separately did not seem to help. Also tried categorizing based on other properties, such as the mean spread, jump in spread at event 50, variance of the spread, ratio of spread to price, security ID etc. - none other than categorizing based on initiator type seemed to help.

The algorithm that worked best for me was linear regression. The spreads at event 49 and 50 and the VWAP turned out to be the most useful predictors. After several tweaks to the code, I was able to extract some predictive power out of many other columns in the data set, including the prices at event 47 and 48, trade volume, count of previous day's trades, sum of previous day's trade values and number of trades vs. quotes.

The best private leaderboard scores for my models are given below. Note that the scores show generic accuracy of the model as I am not doing any per-row massaging of the results.

Linear Regression	0.7781
kNN	0.7848
SVM	0.7956
Random Forest	0.7974
k-means	0.7982
Blended	**0.7752**

Training time varied greatly from model to model - from around 5 seconds for k-means to a few hours for SVM (on a laptop with a 2GHz Intel CPU). The linear regression model with the most predictive power took under two minutes to train.

For SVM, I used an RBF kernel with the same predictors that was published by the organizers. The results were mediocre and the performance was abysmal. Hopefully, Tony will divulge more details of his model and I will know what went wrong with mine.

The linear regression model runs out of gas by the 45th event. Replacing all subsequent predictions with the prediction for event 45 returns the same score. Maybe the blended model retains its predictive power longer - I haven't checked.

I have a long list of ideas that I wanted to try, but never got to. At the same time, I don't think any of those ideas will result in a score anywhere close to Ildefons'. Maybe it's time to move on to something else. Hm... what is this CHALEARN thingy over there...?

Cole Harris • (9th in this Competition) • 6 years ago • Options

1

> ***Neil Thomas wrote***
>
> I see references to day 1, day 2 etc. in the posts. How does one identify data from the same day? Just pick clumps of rows for a security that has the same number of trades on the previous day? Is the assumption that all securities were sampled on each day?

The data prior to the initial testing set appears to be ordered by day, then by stock, then time of day. Although I didn't check, I am pretty certain that all stocks are sampled multiple times each day. There appears to be six days of data. The organizers could have done some randomization, but I would guess not as some statistics change dramatically after 'day 2'. Curious - how well did your public scores correlate?

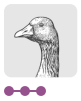
Anil Thomas • (4th in this Competition) • 6 years ago • Options

4

> ***Cole Harris wrote***
>
> Curious - how well did your public scores correlate?

Here are the corresponding public scores:

Model	Public score	Private score
Linear Regression	0.7703	0.7781
kNN	0.7790	0.7848
SVM	0.7902	0.7956
Random Forest	0.7899	0.7974
k-means	0.7811	0.7982
Blended	**0.7651**	**0.7752**



BarrenWuffet • (42nd in this Competition) • 6 years ago • Options

0

On the number of days topic. In looking at the data description it describes the p*tcount and p*value as the prior days trade count and value. I used a SQL query like this:

select distinct security*id , p*value , p*tcount*
FROM [kaggle].[dbo].[training]
*where security*id = 1

and it returns 37 lines. Am I wrong to assume that this means the data came from 37 different days? or are you just referring to the last 50k lines or the testing data?

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

1

> ***BarrenWuffet wrote***
>
> On the number of days topic. In looking at the data description it describes the p*tcount and p*value as the prior days trade count and value. I used a SQL query like this:
>
> select distinct security*id , p*value , p*tcount*
> *FROM [kaggle].[dbo].[training]*
> *where security*id = 1
>
> and it returns 37 lines. Am I wrong to assume that this means the data came from 37 different days? or are you just referring to the last 50k lines or the testing data?

Try to do the same excluding last 50k lines. Last 50k lines were initial test set that was later incorporated into training set.

Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

0

> ***Neil Thomas wrote***
>
> > ***Cole Harris wrote***
> >
> > Curious - how well did your public scores correlate?
>
> Here are the corresponding public scores:
>
> | Model | Public score | Private score |
> |---|---|---|
> | Linear Regression | 0.7703 | 0.7781 |
> | kNN | 0.7790 | 0.7848 |
> | SVM | 0.7902 | 0.7956 |
> | Random Forest | 0.7899 | 0.7974 |
> | k-means | 0.7811 | 0.7982 |
> | **Blended** | **0.7651** | **0.7752** |
>
> How did you do blending?

Vivek Sharma • (3rd in this Competition) • 6 years ago • Options

3

Thanks to all for sharing interesting details. I don't think I was able to take advantage of the idiosyncrasies of the data or the similarities between subets of training data and the test set (to the extent that others might have). My third rank was due to a random forest model that scored 0.77400 on the private and 0.76479 on the public test set. The model was trained on all of the training data (except for the last 50K which I used as the cross validation set and didn't add back to the training set). I didn't have much luck with my linear regression models - with my best model scoring 0.785 on the test set. It might have been because all my models were on log returns even though the competition metric was RMSE. Simple weighted average with the linear regression model improved my score marginally to 0.773.

Anil Thomas • (4th in this Competition) • 6 years ago • Options

0

> ***Sergey Yurgenson wrote***
>
> How did you do blending?

I had the individual models make predictions on held-out data. Linear regression was used to determine optimal weights for each model and these weights were used to blend the submissions. I believe the official term is "stacking".

Anil Thomas • (4th in this Competition) • 6 years ago • Options

0

> ***vsh wrote***
>
> My third rank was due to a random forest model that scored 0.77400 on the private and 0.76479 on the public test set.

That's a very good score for an individual model. How many trees did you use? My random forest model with 200 trees was slow to run, so I had it predict the prices at events 2, 10 and 50 and then followed up with a linear interpolation. Asking it to predict the price at event 50 was probably a tall order. It might have been better to predict up to event 20 or so and then set all subsequent prices to the price at that event. Did you check to see how far the model could predict?

Vivek Sharma • (3rd in this Competition) • 6 years ago • Options

2

Neil, I used 200 trees too. Although, with a lower sample size: 100K and larger nodesize: 100 than the defaults which reduced training time (and also increased predictability). For a single bid/ask it took 10 minutes to train on a single CPU core. I used the largest compute instance on Amazon EC2 (with 16 cores) to train the random forest models for every bid/ask - this took less than 2 hours in total. I also tried with 300 trees but it didn't make much difference.

I checked the predictability at different intervals: 55,65,75,85 and 95 and the random forests were better than my linear models at all those points. However, note that my linear regression models didn't score as well as yours. I trained using log(returns) instead of price so that different securities could be compared against each other - did you use similar transformations in your random forests? I think in general random forests should do better than linear regression almost always.

Since I was using normalized prices, I also noticed (too close to the deadline) that individual security models and models on price (as opposed to log(returns)) combined well with my random forest model. I wasn't able to take full advantage though.

- Vivek Sharma

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

Just curious, has anyone evaluated the potential profitability of application of their models?

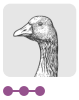
William Cukiers... • (4th in this Competition) • 6 years ago • Options

0

Cole, I have been thinking about this question and my own opinion is somewhat pessimistic. Let's put aside the (significant) practical challenges necessary to do HFT based on tick data and assume we can instantly enter/exit positions. Full disclosure: I am not a finance person and, as a grad student, have not seen finances since I found that $5 bill on the street the other week :)

We aren't the market maker, so we don't get the privilege of seeing the liqiudity shock coming, nor did this contest assess to our ability to predict when the shocks are coming. That means the soonest we can react is at the t=51 time point, after the bid and ask have already gapped. We know from the naive baseline that the steady-state do-nothing model is "on average" (a loose, some would say wrong, interpretation of the RMSE) 86 pence away from the real bid/ask reaction, while the best contest models knocked about 10 pence off that. In my (possibly incorrect) interpretation of the situation, this is sort of an arbitrage window of about 10 pence when averaged over many trades.

Is that enough? I suppose a more thorough analysis that controls for the share price is necessary to really say. However, if we add back in real-world constraints and assume that other market participants have access to the same information (e.g. there was a large market sell of X shares T milliseconds ago), you have to assume that they are at least clever enough to run the linear regression that negates 9/10 of your forecasting advantage.



BarrenWuffet • (42nd in this Competition) • 6 years ago • Options

0

A lot of the HFT stuff is based on hardware (collocation, dedicated fiber, burned chips, etc) and entity structuring (as broker/dealer in order to be market maker and collect rebates from ECNs (which from my limited knowledge is where most money is made as opposed to correctly picking direction)). That being said, there are plenty of places that would talk with you about implementation of your ideas on their hardware/communication platform such as JUMP, Tidal or any of the firms mentioned in the 'Trading' section of eFinancialCareers.com website.

Capital Markets ... • 6 years ago • Options

0

William, if we remove the shackles of specifically looking at liquidity shocks, do you believe that the techniques discussed and developed in this competition would be useful for finding market anomalies and inefficiencies?

BarrenWuffet I agree with what you have said but the game can be played on different levels. At the highest level, software is not even used. Algorithms are programmed directly onto FPGAs and colocated at various exchanges. Then there are software based HFT algorithms that do rely heavily on maker/taker rebates. Then at the 'rookie' level there are algorithmic trades that would generally rely on the less liquid end of the market where there is not enough incentive for the 'big boys' to trade.

So for a fledgling trader it would be somewhat foolhardy to jump into shark infested waters with some of the grizzled veterans of the HFT scene. However by concentrating on areas where one is likely to have an edge, maybe, just maybe it is possible to get a foothold on the ladder.

JC36 • (108th in this Competition) • 6 years ago • Options

0

On the question of profitability it would be helpful if someone would tell us what are the typical financial arrangements between an exchange and a HFT organisation. I understand the exchange pays the HFT organisation for keeping the market "ticking over". They certainly could not afford to pay the brokerage of a retail trader.

Capital Markets ... • 6 years ago • Options

0

It is very dependent on the exchange and the jurisdiction. Sometimes organisations are paid (given a rebate) to provide liquidity (post limit orders). This is known as the maker/taker model. Chi X has recently opened in Australia. The local version of maker/taker involves liquidity providers receiving a discount rather than an outright rebate.

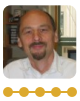
William Cukiers... • (4th in this Competition) • 6 years ago • Options

0

> ***Capital Markets CRC wrote***
>
> William, if we remove the shackles of specifically looking at liquidity shocks, do you believe that the techniques discussed and developed in this competition would be useful for finding market anomalies and inefficiencies?

I think the liquidty shock actually frames the problem nicely. You have to set your time origin somewhere, and I think picking a larget market order is one way of isolating a timeframe where you expect *something* to happen. If we were instead given unregistered, raw tick data, we would have to modify the models to handle the many times where the bid/ask are not moving (or "random walking", or "market open mayhem", etc.). This modifciation could be as simple as a feature given to a decision tree (e.g. has there been a recent market order), but I suspect it would require more deliberate intervention. I attribute any success from the models developed in this competition to the fact that the market reacts systematically to a buy vs. sell order.

An interesting offshoot of this observation: what would happen instead if you gave us t51...t100 bids/asks and asked us to classify whether the trade was buyer/seller initiatied? I suspect the accuracy would be very high, but the more interesting thing to look at would be the anomolous trades which we don't classify correctly. If they share common traits (a big "if"), then you can really get into the question of exploiting inefficiencies.



Sergey Yurgens... • (6th in this Competition) • 6 years ago • Options

0

I would look on it not as a problem of finding market anomalies and inefficiencies but as a problem of pedicuring anomalies and inefficiencies. I would expect inefficiencies to be very short-lived and thus providing good time reference point. One can select one specific type of inefficiency and create dataset containing market data before inefficiency happened (obviously, with some data records when inefficiency did not happened). Then task will be to create classification model to predict future inefficiency using past market data (close analog - Credit competition)

Cole Harris • (9th in this Competition) • 6 years ago • Options

0

@William

'The application of a model' is not trivial, even ignoring the technical issues. Turning a model prediction (derived from all liquidity events) into a trade signal may not be the best approach, which is why I asked the question. I don't think any of my contest "buy" models ever predicted a future bid beating the vwap (or initial ask), so they would break even by never trading:)

However it is much easier to directly address the question (with some assumptions) if I produced the liquidity shock, can I predict if I can get out at a profit in short order? Or, if I detect a liquidity shock, can I exploit the subsequent move in prices for some an identifiable subset? I find potential here.

Capital Markets ... • 6 years ago • Options

0

Interesting questions and observations. Thank you for your input. You are welcome to get in touch privately if you have any more questions or anything else you wish to share.

Fresh • (70th in this Competition) • 6 years ago • Options

0

Will Ildefons algorithm be revealed and explained? Thank you.

MLF26 • 4 years ago • Options

0

Varun Nagarajan • 3 years ago • Options

0

Hi Guys, I am a beginner and i am trying to learn how you guys got the solution. If anybody could please post the solution here, it would be helpful for me to understand how you guys did it. I use RStudio. Id the code you have written is in R, i would highly appreciate that. Tks

euclidriver • 2 years ago • Options

0

congrats , wish a future.