Benchmark Bond Trade Price Challenge

Benchmark Solutions

Develop models to accurately predict the trade price of a bond.

$17,500 · 264 teams · 6 years ago

Congratulations!

posted in Benchmark Bond Trade Price Challenge 6 years ago

0

Vivek Sharma

16th place

Options

Congratulations to the winners! And, thanks to the sponsors and Kaggle for a terrific contest. There was clear differentiation from the rest of the pack as well as between the winners.

I'd love to know what insights, methods worked for people, and what didn't.

Comments (21)

Sort by Hotness

Click here to enter a comment...

Sergey Yurgen... • (2nd in this Competition) • 6 years ago • Options • Reply 5

Kaggle competitions sometimes remind me cooking competitions. It very important to find that secret ingredient (and every cook has its own secret ingredient) , but then you need to find an original way of using it(slice and dice? boil? bake? use raw?) .
Our initial progress was based on RF model with minor data preprocessing. Actually, we had RF with private score of ~0.727 submitted on Feb 29. For little while we were stuck, making only small progress by using postprocessing and creating RF ensembles. Then Bruce found what we thought was the "secret" ingredient - GBM. Now we can see that it was not so secret :). We made a good run using GBMs only , however our best submission was ensemble of RFs and GBMs.

Gradient Boosting Machines

Wayne Zhang • (13th in this Competition) • 6 years ago • Options • Reply

1

@Vivek: I did have the same experience of overfitting RF to training data. That's why I turned to linear regression. I agree with Halla, so there may be some normalization.
I also used time weighted VWAP, but I found std not that helpful.

Vivek Sharma • (16th in this Competition) • 6 years ago • Options • Reply

1

Sergey, Bruce - seems like you had a very well 'cooked' RF model indeed. :-) Thanks for sharing, and congratulations again.

@desertnaut Since you asked, I had a miserable time tuning random forests. I think I got misled by the same problem that Cole and others mentioned in the other thread. After a point, the RF held-out scores diverged greatly from the test set scores, making life difficult. Would anyone have any insight into why this might have been the case? Perhaps, it was due to my silly mistake of not working with log transformed returns? GBMs were better, in that, at least the held out scores matched up with the test set scores (perhaps, they were tolerant of non-transformed prices?). Additional predictors like vwap, time weighted vwap and std dev of prices were quite useful.

I spent some time working in yield space but didn't get anywhere with that. Did anyone try anything with yields? Did anyone try anything special based on the time_to_maturity of the bonds, or apply any domain-specific tricks?

Glen • (7th in this Competition) • 6 years ago • Options • Reply

1

I ended up using a combination of locally weighted non-linear regression, random forests and gradient boosting. I only used variables up to curve*based*price_last4. I don't think that the winners used any secret trick. I'm guessing they just created an ensemble of different methods which is very effective but also very time consuming.

ivo • (14th in this Competition) • 6 years ago • Options • Reply

1

Grats to the winners!

I had a very simple modeling tech.

Generated some stats from the curve_based_prices_lastx and trade_price_lastxs (sd, average, median, linear extrapolation) and from some other features like present value of a bond with a given maturity and coupon with 10% reference rates. I only modeled the trading price of a bond, where the received_time_difference_last1>300, because the trade_price_last1 was a sufficient predictor on average for those with rtdl1<300. (That may have been a mistake.)

Tried some regression techniques: linear regression, PACE regression, Regression trees (bagged), neural nets, local linear regression. PACE was great overall, neural nets were good where bonds were callable (handled the the callable non callable bonds separately). I clustered the dataset with the training set and reached the highest accuracy by voteing together 69 different models (took the median of the distinct models, it was better than the average :))

Loved this competition! Looking forward to finance related comps!

teaserebotier • (37th in this Competition) • 6 years ago • Options • Reply

2

Vivek Sharma wrote:

@desertnaut Since you asked, I had a miserable time tuning random forests. I think I got misled by the same problem that Cole and others mentioned in the other thread. After a point, the RF held-out scores diverged greatly from the test set scores, making life difficult. Would anyone have any insight into why this might have been the case?

I had the same problem and asked in a different thread. Turns out that the test and train sets were made using different bonds, so the proper witholding loop is to withold all the trades from each of a select number of bonds. If you were witholding a random set of trades, the other trades from the same bonds would allow your model more info than it has on the test set.

Anil Thomas • (9th in this Competition) • 6 years ago • Options • Reply

2

> ***Vivek Sharma wrote***
>
> @desertnaut Since you asked, I had a miserable time tuning random forests. I think I got misled by the same problem that Cole and others mentioned in the other thread. After a point, the RF held-out scores diverged greatly from the test set scores, making life difficult. Would anyone have any insight into why this might have been the case?

Tune Hyperparameters

I ran into the same problem towards the end of the contest. At least in my case, there is a simple explanation. After getting the score on the held-out set, I went back and tweaked the parameters to make the score better. Essentially, I was overfitting to the held-out set. As the test set had completely different bonds, clearly the score on the test set had to be worse with this overfitted model.

Had I cross validated using the test set, tweaked the parameters to make the test score better and then tried the model on the held-out set, I would have gotten a worse score on the held-out set. Haven't actually tried this out, but one would expect this to be true in general.

college, PhD from HBS

Halla Yang • (11th in this Competition) • 6 years ago • Options • Reply

2

If I had to guess why RF's oob estimate was so poor, it's that the observations in the training dataset weren't really independent.

For example, suppose there is a bond that always trades at exactly 103.51, except for one instance where someone miscoded the price at 1035.10. This miscoding can show up in 10 different entries on the RHS: trade_price_last1, trade_price_last2, ... , trade_price_last10.

In this case, a "feature" defined as "any trade in the past ten trades = 1035.10" might predict a trade price of 103.51 in sample. The RF held out estimates will see the same outlier and you'll get an artifically good "oob" estimate.

In other words, it seems like outliers in the data could be correlated across pseudo-out-of-sample and pseudo-in-sample observations because the same bad data will show up in multiple rows of the training data.

Wayne Zhang • (13th in this Competition) • 6 years ago • Options • Reply

0

Congratulations!

I used linear regression, with 2nd order cross-form and log.

Halla Yang • (11th in this Competition) • 6 years ago • Options • Reply

0

Congratulations to the winners. The spread between the in-the-money and the rest of us was quite large so clearly #1, #2, #3 know something that the rest of us don't.

I used a random forest, with a severely limited set of features: thirteen predictive variables. I found most of the variables to be unhelpful, probably because there were intuitive sufficient statistics. It was clear from the data that corporate bonds are very illiquid and that they trade in bursts: weeks or months might elapse with no trade, and then you might see a flurry of matched trades in quick succession as customers and dealers pass the bonds around like hot potatoes. I found it essential to clean/filter my data, and this is one area where I wish I had spent more time.

Anil Thomas • (9th in this Competition) • 6 years ago • Options • Reply

0

I kept looking for that bottle of secret sauce, but never found it. Looking at the final scores, I am even more convinced that there is a short cut somewhere that most of us entirely missed.

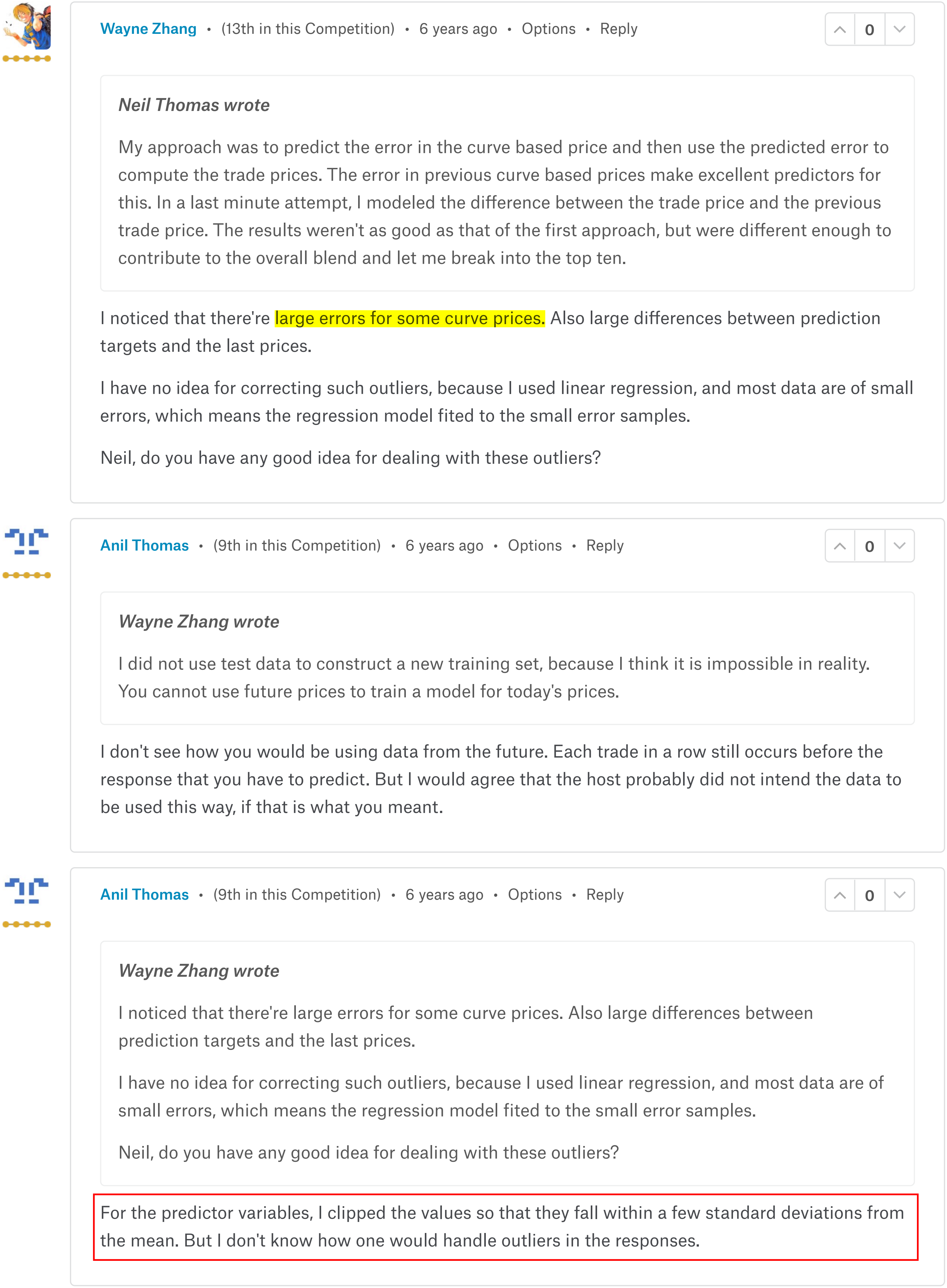
Maybe others noticed this too: it is possible to use the test data for training. Just shift the trades to the left by one, delete the last trade and bingo, you have a new training set (trade*price*last1 in the test set is now the response variable). Looks like this is fair game as the rules don't restrict the usage of test data. I tried to take advantage of this, but couldn't work it into the blend in a meaningful way. Did anyone else try this? As the test data has a different set of bonds, one would think that there is some edge to be gained there. The test set contains information that the training set just cannot provide.

My approach was to predict the error in the curve based price and then use the predicted error to compute the trade prices. The error in previous curve based prices make excellent predictors for this. In a last minute attempt, I modeled the difference between the trade price and the previous trade price. The results weren't as good as that of the first approach, but were different enough to contribute to the overall blend and let me break into the top ten.

Wayne Zhang • (13th in this Competition) • 6 years ago • Options • Reply

0

I did not use test data to construct a new training set, because I think it is impossible in reality. You cannot use future prices to train a model for today's prices.



Wayne Zhang • (13th in this Competition) • 6 years ago • Options • Reply

0

> *Neil Thomas wrote*
>
> My approach was to predict the error in the curve based price and then use the predicted error to compute the trade prices. The error in previous curve based prices make excellent predictors for this. In a last minute attempt, I modeled the difference between the trade price and the previous trade price. The results weren't as good as that of the first approach, but were different enough to contribute to the overall blend and let me break into the top ten.

I noticed that there're large errors for some curve prices. Also large differences between prediction targets and the last prices.

I have no idea for correcting such outliers, because I used linear regression, and most data are of small errors, which means the regression model fited to the small error samples.

Neil, do you have any good idea for dealing with these outliers?

Anil Thomas • (9th in this Competition) • 6 years ago • Options • Reply

0

> *Wayne Zhang wrote*
>
> I did not use test data to construct a new training set, because I think it is impossible in reality. You cannot use future prices to train a model for today's prices.

I don't see how you would be using data from the future. Each trade in a row still occurs before the response that you have to predict. But I would agree that the host probably did not intend the data to be used this way, if that is what you meant.

Anil Thomas • (9th in this Competition) • 6 years ago • Options • Reply

0

> *Wayne Zhang wrote*
>
> I noticed that there're large errors for some curve prices. Also large differences between prediction targets and the last prices.
>
> I have no idea for correcting such outliers, because I used linear regression, and most data are of small errors, which means the regression model fited to the small error samples.
>
> Neil, do you have any good idea for dealing with these outliers?

For the predictor variables, I clipped the values so that they fall within a few standard deviations from the mean. But I don't know how one would handle outliers in the responses.

The suffocated • (226th in this Competition) • 6 years ago • Options • Reply

0

> ***Neil Thomas wrote***
>
> Just shift the trades to the left by one, delete the last trade and bingo, you have a new training set (trade*price*last1 in the test set is now the response variable). ... Did anyone else try this?

I tried. I merely had time to implement a base model. When I used it to predict the most recent trade price using the previous nine trades in each row of this new training set, the errors were about 0.543±0.001 (with one third of the the test set held out at random). Unfortunately, the leaderboard score ended up with a pathetic 1.03. In hindsight, this is reasonable because the base model --- some variant of linear regression --- is not weighted, so that data weights are not taken into account in the regression process.

teaserebotier • (37th in this Competition) • 6 years ago • Options • Reply

0

Hey Neil,

I had been scalded by the other trading contest so I actually started with the shifted test--didn't get far actually but I gave it up before puttin my entire alg together. I made a few early subs with shifttest or shifttest+train, to no avail, then focused on train; in the spirit of the random forest I tried splitting the train set and optimize different models based on trade type, previous trade type for most trades, and cluster (based on variables); that meant having a lot of data to avoid bad conditioning, so I focused on the test set. Then I got mired in my own bugs and the fact that the metrics and param optimization, being done at the trade level, wasn't reflecting the real structure of the test set selection (algorithmic trading challenge, on liquidity shock recovery, where the test set was way different from the trading set), but that's another story. By the way, I still see unexplanaible differences, now between the 'public' and 'private' scores. Waiting for the silt to clear up on that one ;-)

desertnaut • (10th in this Competition) • 6 years ago • Options • Reply

0

Hi all, and congrats to the winners!

Well, it was a great experience, and our good final ranking came rather as a surprize: all three members of Toms' Friends we have started studying data mining only since last October, without any relevant prior experience whatsoever. None of us had any experience with R either (last time I had coded was back in 2000, and it was in Matlab). Despite of all these, we sat down and did some serious brainstorming, followed by extensive experimentation using R (no question we ended up with 100 submissions!).

We used random forest for a crude initial forward feature selection, and when we (thought we had) found our feature set we proceeded to modelling with gradient boosting. This gave very competitive results early on, so we proceeded with detailed parameter tuning and averaging some of our best models' outputs. Pretty much that was all - no clustering, no test set usage, no outlier detection, only some handling of the missing values. We tried different approaches to variables modelling (averages and differentials), but it proved that nothing could surpass the non-transformed variables input. We tried to remove from the training set some price value ranges that were not present in the test set, but again this gave inferior results...

We felt sorry to see Wayne (Zhang) not entering the final top ten, but hey Wayne, no worries, the future lies ahead...

Vivek, any insight from your part??

ivo,

During these last weeks of the contest, we felt that you and us we were running almost side by side... For us, rookies here, it was a hell of a race, and we would like to thank you for this (we would have sent you a hello message, if this was possible through kaggle...). Sincerely hope to see you around - and BTW, you have a HELL of a profile photo!! :-)

Many thanx to my team mates, PepFriday & tinariwen. This was our first time here, but we are just starting, and hopefully we will stick around...

des

Bruce Cragin • (2nd in this Competition) • 6 years ago • Options • Reply

0

Well said, Sergey. Congratulations to all the competitors and to Kaggle and Benchmark for an extremely well-run and well-formulated competition. Oh, and by the way, if any of you are lucky enough to get the chance to work with Sergey, take it! I've learned a lot and my scores improved considerably as a result of our collaboration.

Wayne Zhang • (13th in this Competition) • 6 years ago • Options • Reply

0

> ***Neil Thomas wrote***
>
> I don't see how you would be using data from the future. Each trade in a row still occurs before the response that you have to predict. But I would agree that the host probably did not intend the data to be used this way, if that is what you meant.

Thanks.

A piece of my thinking is: at least two drawbacks of allowing use of test data.

1) the training and test are different bonds. If use test data to train, the model may fit to test bonds.

2) backtesting usually use some period of data for training and another period for test.

In contrast, in this contest, assuming there are [t:t+9] for predicting t+10, [t+1:t+10] for predicting t+11.

Then by shifting, you have [t:t+8] for predicting t+9, [t+1:t+9] for predicting t+10. There exists overlapping between training and test.

I understand that this has been avoided since the modification of data.

But there may exist [t:t+9]->t+10 (a), [t+6:t+15]->t+16, and then [t:t+8]->t+9, [t+6:t+14]->t+15 (b) for training.

Note that (b) is future data of (a). It cannot be used for real backtesting.

Welcome any correction.

Vivek Sharma • (16th in this Competition) • 6 years ago • Options • Reply

0

Halla, interesting point. It seems that training RFs by using sampled subsets (every 12th row) would eliminate this problem. I wonder if you and others who did well with RFs, did such sampling.