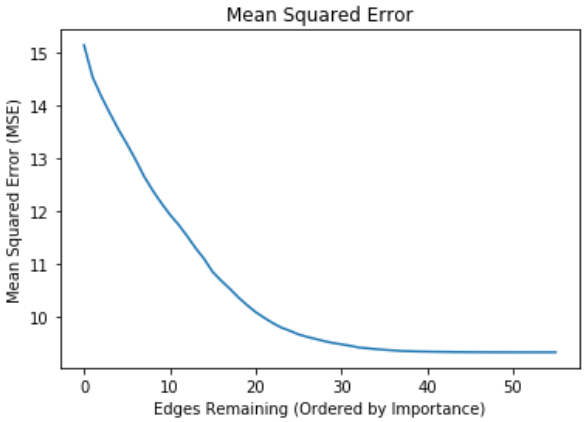
Rui Update Meeting 16

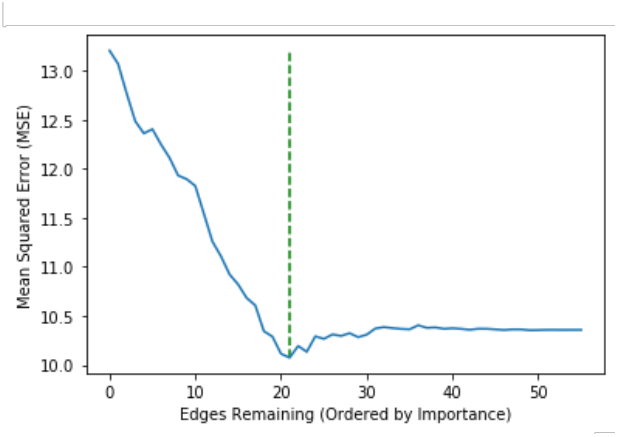
Bootstrapping to Estimate W using OMP

Implemented, and works well.

Example: We generate X and X' using the same 10-dimensional matrix W, and for 100 samples. We estimate W_OMP using X, and compare its MSE on X'. We see that bootstrapping has prevented us from overfitting on the correlated noise.



MSE of iteratively pruning W_OMP on X.



MSE of iteratively pruning W_OMP on X'.

Highlighted in green, the W_OMP^(i) that minimizes the MSE.

Performance comparisons between W_OMP and W_bootstrap on X.

```
Dense W_OMP Results on X:
True Positive Rate: 1.0.
True Negative Rate: 0.6.
Accuracy: 0.7.
R-Squared: 0.385
Mean Squared Error: 9.32
```

```
Bootstrap W_OMP Results on X:
True Positive Rate: 1.0.
True Negative Rate: 0.949.
Accuracy: 0.96.
R-Squared: 0.34
Mean Squared Error: 10.018
```

As expected, the bootstrap W_OMP yields *higher structural performance*, as there is no overfitting on the noise anymore. However, as expected, *the mean squared error and the R-Squared are lower*, but this is for the exact same reason; we do not overfit on the noise anymore.

Interestingly, the right plot always shows the same behavior, with three stages.

1. On the right side of the plot (30 – 55), we see that we remove untrue unimportant edges, but they do not reduce the MSE significantly. This is because these edges were unimportant in X, so they have low coefficients, so there was not much overfitting on these edges.
2. In the middle of the plot (20 – 30), we see that we remove untrue, but slightly important edges. On these edges, OMP has overfitted in X, but the noise on which OMP has overfitted is not in X', so we see that we have a slight improvement in MSE when we remove these overfitted edges.
3. On the left side of the plot (1-20), we have the true and therefore also important edges. We see that removing these will decrease the MSE significantly, which makes sense as these are important edges. These are the edges that we want OMP to fit on.

All in all, if we go from right to left, we see that we always first have a plateau (removing untrue not overfitted edges), followed by a slight decrease in MSE (removing untrue, but overfitted edges), followed by a steep increase in MSE (removing true edges).

Follow-Up Work

Other approaches to bootstrapping. Now, we again generate T samples, but in reality we may not have this data at hand. Also investigated different proportions of real X and bootstrapped X' (like train-test split).

Also, an approach would be leave-one-out-cross-validation, but this is more difficult in time series data, as the data is not independent. Alex and I thought about this, but things need to be written down carefully about this.

The problems are:

1. Normally, you simply leave one sample out, and train on the remaining T – 1 samples. Then, you test your model against this one left out sample. Then, we do this for all T samples. However, how do we test our model against just *one* sample? For our setting, this is not possible, as we need to check whether X_t = X_{t – 1} W, so we need at least two samples.

2. Furthermore, leaving out a sample leaves a "gap" in the training data, which is not what we want. By removing e.g. X_i, in the training data, we will compare X_{i-1}W to X_{i+1}, which is a problem. To fix this, we must not count this error. However, this might be neglible when T is large enough.

To fix 1, We take out X_i and X_{i + 1}, and I compute the MSE of the test sample, which is ||X_{i + 1} – X_i W||_2^2. I do this for all i in {1, …, T – 1}. In this way, we can estimate the MSE of each sample X_i.

To fix 2, We simply disregard the "gap" X_{i – 1}, X_{i + 1} when fitting the data.

Both fixes do stray away quite a bit from the convential LOOCV, so I am unsure whether this is the most sensible approach, but this is what I came up with for now.

Benchmarks

We compared the results of OMP and NOTEARS, but they seemed incredibly comparable, which is interesting. Both were also incredibly close to the optimal solution (when it could be computed for small n), within the range of 99% often, which perhaps indicates that it is not that difficult to solve this exactly, especially when there is no model mismatch (i.e., the data was generated by a DAG W).

Theorem Paper

I implemented basically everything from this paper and verified empirically whether the statements from the theorem were true in our setting, even though we already know that the independence assumption fails.

Interestingly, the results from the theorem still held, but they were very strict. It often required a large amount of samples to be realistic, or else we would e.g. require that all coefficients were larger than 1.5 (which is not realistic in VAR(1) settings).

All in all, there is certainly something to gain, but since our setting is not simple independent regression, there are some modifications that most likely need to be made.

Example was the mu quantity that they use throughout their paper, which penalizes the fact that we can e.g. predict X_1 well using X_3. In our setting, this is good, but in their setting, this is bad, so we need to understand this phenomenon better and find a more suitable way to estimate this mu.

Plans until next meeting

- Investigate Leave-One-Out-Cross-Validation, check whether it works better, investigate effectiveness of the bootstrapping approaches so far. What works well in which scenarios and why?
- More Benchmarks.
- Investigate the paper further, try to understand all the proofs and try to find a way to cast it into our setting; in order to manipulate these proofs, deep understanding is required.
- Catch up on some writing of different things, make things more orderly and neatly in the thesis over the holiday.