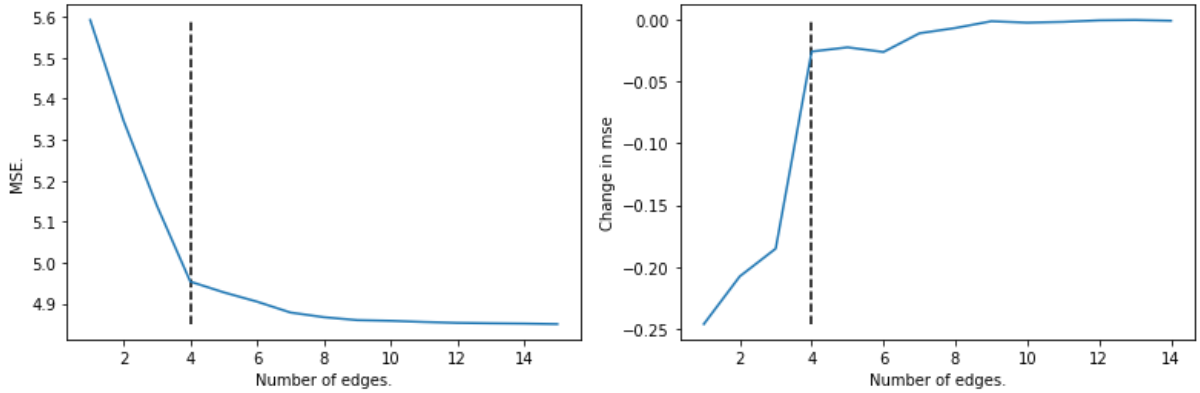
Dear Rui,

Hereby I send you a small update of what Alex and I have discussed during our meeting. Feel free to read it to update yourself, but should you be busy, please do not trouble yourself then.

1. OMP method is implemented. We greedily add coefficients (in our settings edges) using OMP. For every edge, we check whether it has created a cycle and hence, violated DAGness. If so, we fix this edge to zero and continue all the way until it is a DAG.

   Results: Were very promising, almost always very near the global optimum (which I verify using an exhaustive search. Iterating over all n! permutation, I compute the fitting OLS solution adhering to a DAG. This is by definition the W that minimizes the mean squared error / squared L2-norm.

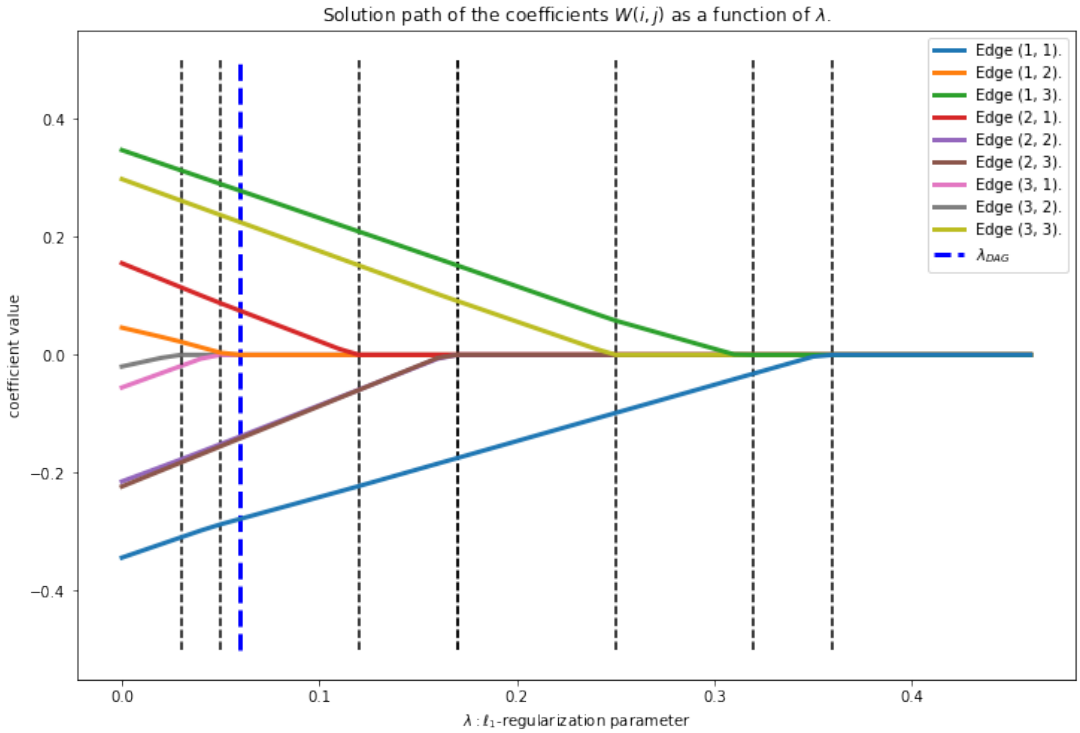
   Discussions that arose: The mean squared error decreases at every iteration (by definition of the greedy algorithm). Interestingly, the *gain* (or first derivative), is not strictly increasing. It is possible that edge #3 decreases the MSE by 0.1, and edge #4 decreases the MSE by 0.2. Bit unexpected, but nothing too strange I presume. Figure:



   Furthermore, now that we have the edges ordered by importance, how do we define the *cut-off* point? For large data sizes (large T = 1000), we see a clear “kink” or (“elbow” as Alex said) in the MSE graph, but for normal data sizes (T = 50 – 100), the transition is smoother. Firstly, do we care about this cut-off point, if our interest is minimizing the mean squared error? Again, this is the discussion of structural performance (TPR, accuracy, etc.) versus predictive performance.

   Method for finding cut-off point: Alex came up with an interesting way to fix this, which is sort of similar to cross validation. We fix a data generating model with matrix W. We first compute W using OMP using these 100 samples. Most likely, OMP will overfit slightly on these samples, and wil mistake some noise for actual coefficients. Then, we generate again 100 samples using the same procedure, but now the noise correlations are different, and we hope to use this to detect which edges were true, and which edges were produced by random noise. I can elaborate more, but the mail will get quite lengthy.

2. LINGNAM-LASSO Solution Path. Nothing too fancy and perhaps a bit out of the blue, but we had a discussion earlier of making such a solution path for the LINGNAM-LASSO and see how it behaved. Figure and matrix W are given below:




```
Estimated W at blue vertical line (LINGNAM-LASSO solution):
 [[-0.28  0.    0.28]
  [ 0.08 -0.15 -0.15]
  [ 0.    0.    0.23]].

True W:
 [[-0.25  0.    0.25]
  [ 0.    0.   -0.25]
  [ 0.    0.    0.25]].
```

3. Benchmarks created a lot of datasets and found weaknesses and strengths of the methods evaluated (Exhaustive, OMP, LINGNAM-LASSO, LINGNAM-OLS, NOTEARS). OMP was ***incredibly*** competitive with the exhaustive global optimum, and the other methods trailed behind quite a bit.

4. Simple model mismatch result: We also found an easy setting with a simple model mismatch in which both LINGNAM methods perform incredibly poorly, NOTEARS quite okay, and OMP still very close to the exhaustive. The setting is any p x p matrix, where the top right and bottom left entry are non-zero. This is a clear two-cycle and hence a violation. However, both LINGNAM methods will have to remove all entries that are smaller in absolute value than the minimum of these two violators for the matrix to be a DAG. Therefore, if we pick a matrix with these two the largest, and we have enough samples, the resulting "DAG" will simply be the zero matrix with one entry, the largest of the two. OMP still performed very well, NOTEARS not so, but less poorly than these two.

   An example:
   The true W is:

```
[[ 0.27  0.    0.6 ]
 [ 0.47  0.4   0.  ]
```

```
  [-0.6   0.33  0.34]].
```
The OLS W is:

```
[[ 0.34 -0.14  0.6 ]
 [ 0.47  0.37  0.03]
 [-0.78  0.45  0.22]]
```
And therefore, the LINGNAM-OLS (iteratively remove smallest element until DAG) is:

```
[[ 0.    0.    0.  ]
 [ 0.    0.    0.  ]
 [-0.78  0.    0.  ]].
```
With a mean squared error of 4.76. However, the global minimum with MSE = 3.60 is:

```
[[ 0.34  0.    0.  ]
 [ 0.47  0.41  0.  ]
 [-0.78  0.45  0.24]].
```
And the OMP method with MSE = 3.60 is:

```
[[ 0.33  0.    0.  ]
 [ 0.45  0.4   0.  ]
 [-0.79  0.44  0.23]].
```
Alex and I agreed that it is a nice example that shows that one simple model mismatch will immediately cause problems for both the LINGNAM approaches.

5. OMP Guarantees: We also looked into ways we can say something actually statistical about our OMP approach, and this paper that Alex found seems to be a very good first step. However, one of the assumptions is clearly violated, and that is independence of the data. However, as our data only depends on the previous time-step, some sort of Markovianity / Martingale could be used to say something perhaps less strong about our results, but that requires some digging.

That is all, apologies for the lengthy e-mail. Should you want some clarifications, feel free to ask, I'll be happy to dive into more details about things you find interesting. If you have any other questions, feel free to let me know.

Hopefully, things are a bit less chaotic now, I wish you and your household all the best!

Kind regards,

Martin.