Contents

1 Setup

This experiment tested the results of running two function prediction methods (WMV and KNN after embedding) on the DREAM3 network after performing denoising on it by adding edges using link prediction.

Two link prediction methods were used. Normalized DSD created an embedding and then pairwise distance was used to rank the edges. The GLIDE method was ran as specified by Kapil with the parameters set as follows:

```
params = {"alpha" : 1, "beta" : 1000, "delta" : 0.001, "loc" : "l3"}

def foo(a):
  pass
```

For both link prediction methods, the top 10% of edges were added to the network with a weight of 1.

Performance was evaluated using 5-fold CV with the GO label set from 2019 where all GO labels annotating less than 50 or greater than 1000 proteins are discared (I use Lily's code to do this).

2 Results

The accuracy for each method is listed below:

```
Majority Vote:
WMV on raw network: 11.804767309875142%
WMV on DSD added edges network: 11.804767309875142%
WMV on GLIDE added edges network: 11.804767309875142%

10-Nearest Neighbors using DSD embedding:
KNN on raw network: 12.553916004540294%
KNN on DSD added edges network: 12.57661748013621%
KNN on GLIDE added edges network: 12.57661748013621%
```

3 Conclusion

It is clear that when using WMV there was no difference after denoising the network by adding edges. I have a few ideas explaining this behaviour. First, we add edges with a weight of 1, while the raw network contains very high confidence interactions with weight up to 105. I believe that

these high weight interactions have the final say because of their weight and that creating fake low weight interactions makes little difference. Second, the GO label set is hierarchical, so I imagine that more popular terms are always being predicted. Terms more specific to certain proteins aren't counted for as much.

It is good to see that performance either stays the same or improves using both denoising and both FP methods. However, I am a little concerned that the local aspect of GLIDE is not being captured as it carries the **exact** same importance as normalized DSD. I think some parameter tweaking is definitely necessary, and the GLIDE code reviewed.