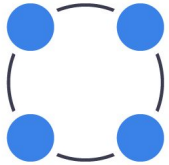


BIOINFORMATICS INSTITUTE

MultiBioNet

Link prediction in multilayered biological networks

Academic advisor:
Elena Sügis, University of Tartu

Team:
Margarita Akseshina
Evgeny Bakin
Rail Suleymanov
Mikhail Papkov

17.12.2016

Outline

1. PPI prediction in Alzheimer disease
2. Data
3. Exploratory analysis of data
4. Classifier description
5. Prediction results
6. Alternative approaches
7. Conclusions

Initial dataset

IntAct database (MI score >= 0.45 — highly confident):

1. Expert curated interactions related to Alzheimer's
2. All highly confident interactions in human
3. Automatically extracted interactions related to synaptic activity
4. Expert curated interactions related to Parkinson's disease

\+ Genes, that are co-expressed and differentially coexpressed in the Alzheimer's patients and healthy individuals

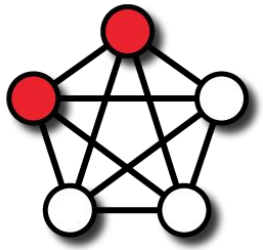
PPI prediction in Alzheimer disease

- Protein-protein interaction (PPI) - a specific physical contact between two proteins.
- Harmful effect of many diseases is a result of PPI.
- Aim of medicines for such diseases is a affecting of PPIs.
- Real experiments for a PPI detection are expensive and time-consuming.

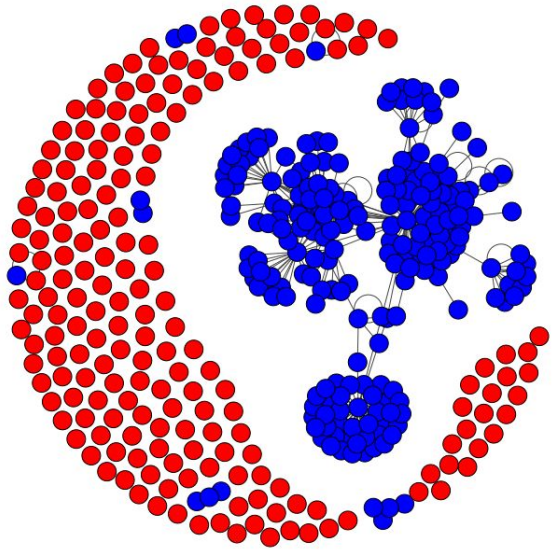
Can Machine Learning help to predict unknown PPIs by a set of known PPIs?

Data sources

- IntAct (http://www.ebi.ac.uk/intact) - datasets on intermolecular interactions
- KEGG (http://www.genome.jp/kegg) - functions and attributes of biological systems
- GWAS central (http://gwascentral.org) - data on research in polygenomic associations
- DISEASES (http://diseases.jensenlab.org) - disease-gene associations mined from literature
- STRING (http://string-db.org) - data on known protein-protein interactions



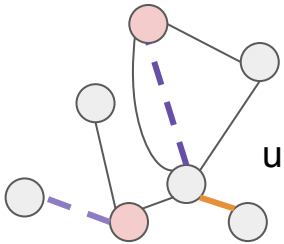
Preliminary analysis of data



We can combine all datasets in one graph:

- unifying genes and proteins with Ensembl ID
- vertices - genes
  - some vertices have specific labels (GWAS, pathway, ...)
- edges - interactions (PPI, coexpressions, ...)

We got 416 alz vertices and 392 known alz interactions in a graph with 11784 vertices and 38574 edges.



united graph

Choice of predictors

From each graph we extract two predictors:

- Inverse distance: d^{-a} $(a > 0)$
- Jaccard similarity $J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$

Graphs used: **Synapse**, **Parkinson**, **Diff. co-expression**, **Alz. co-expression**

From each list we extract two binary predictors:

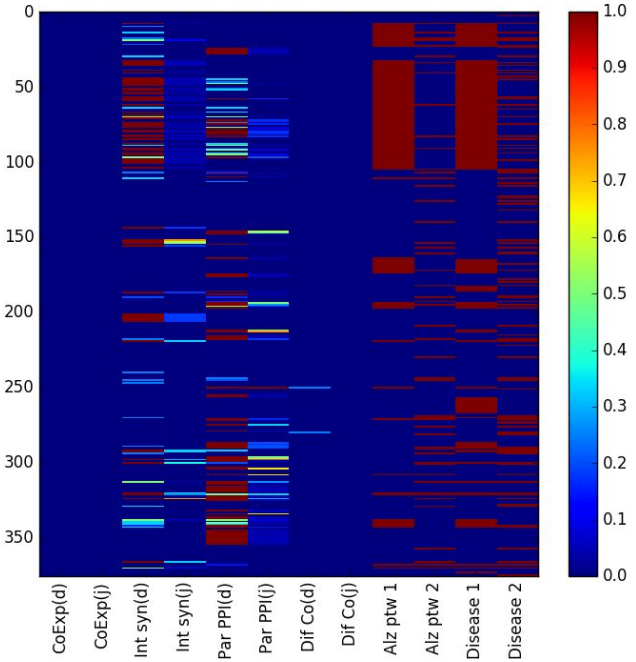
- If gene #1 is in list?
- If gene #2 is in list?

Lists used: **Alz. pathways**

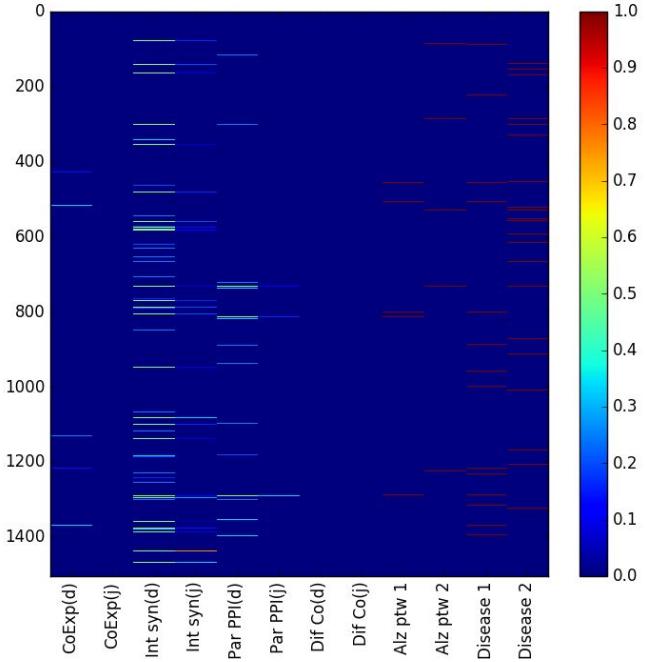
In total: 10 predictors were chosen.

Predictors heatmaps

Pair of genes with known Alzheimer PPI



Random neighbouring pair from filtered Intact dataset

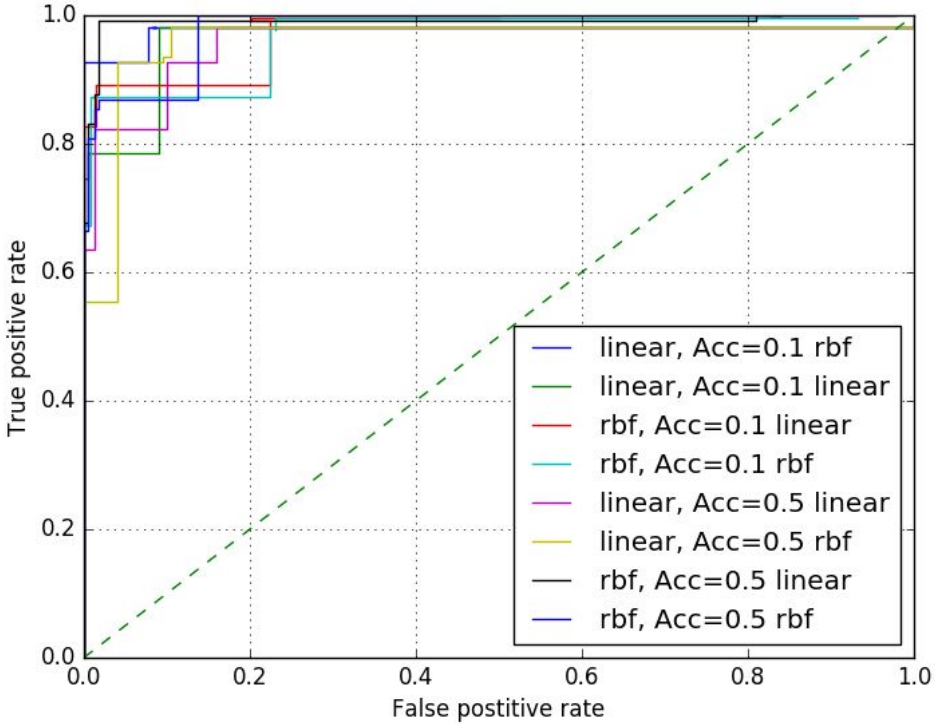


General classifier idea [1,2,3]

1. Create one-class (unary) classifier, fit on Alzheimer Intact dataset (**positives**). This was performed via one-class SVM (outlier detector).
2. Process filtered full IntAct dataset via one-class SVM. This allows to get **negatives**.
3. Conventional SVM is fit via positives and **estimated** negatives.
4. Cross-validate two-stage classifier with LOO procedure (build ROC-curve).
5. Choose appropriate classifier settings, based on FPR and TPR in ROC.
6. Use adjusted classifier for prediction of interacting genes from GWAS dataset.

[1] *Yiming Chen, Zhoujun Li, Xiaofeng Wang, Jiali Feng, Xiaohua Hu,* Predicting gene function using few positive examples and unlabeled ones
[2] *Xiao-Li Li, Bing Liu,* Learning from positive and unlabeled examples with different data distributions
[3] *Peng Yang, Xiao-Li Li, Jian-Ping Mei,Chee-Keong Kwoh and See-Kiong Ng,* Positive-unlabeled learning for disease gene identification

ROC-curves



1. Different points in ROC are obtained by means of classes weights skewness in conventional SVM (2nd stage of classifier).

2. Discrete nature of ROC curves - due to a discrete nature of features.

3. Actually this is an upper bound of ROCs (since negatives are not confident).

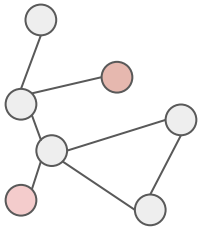
4. Chosen set of parameters:

- Linear kernel for the 1st stage (one-class SVM);
- 1st stage training accuracy: 0.1;
- RBF kernel for the 2nd stage (conventional SVM).
- 2nd stage weight: 3

Data for final prediction

We want to explore interactions with one end in the GWAS set.

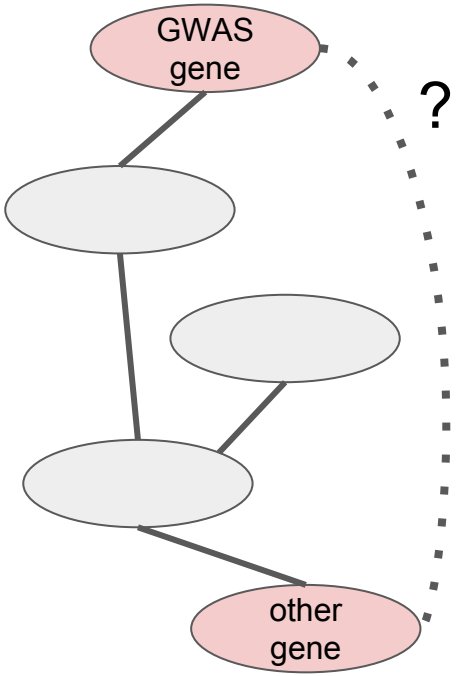
Potential neighbors should not be more than *k* edges away from each other. As a baseline we use *k* = 2. However, *k* up to 4 were checked also.



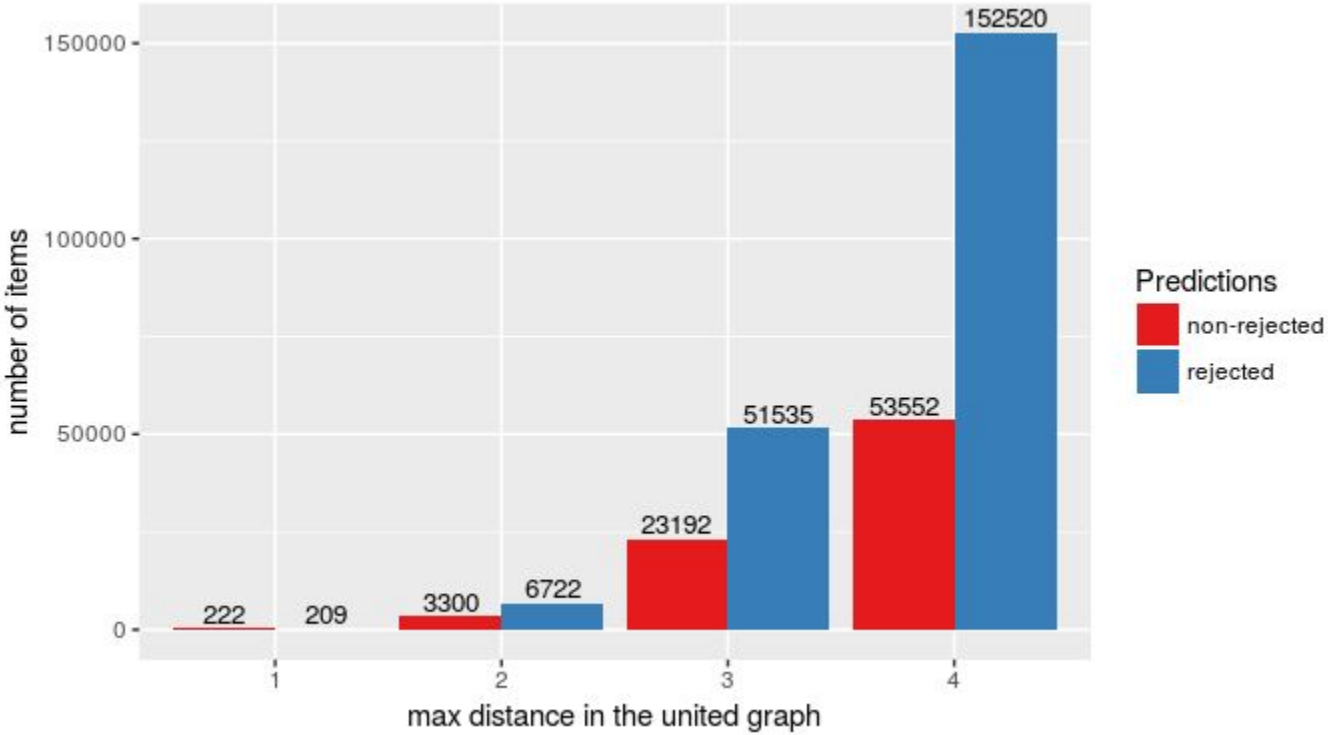
graph

find neighbors of GWAS genes

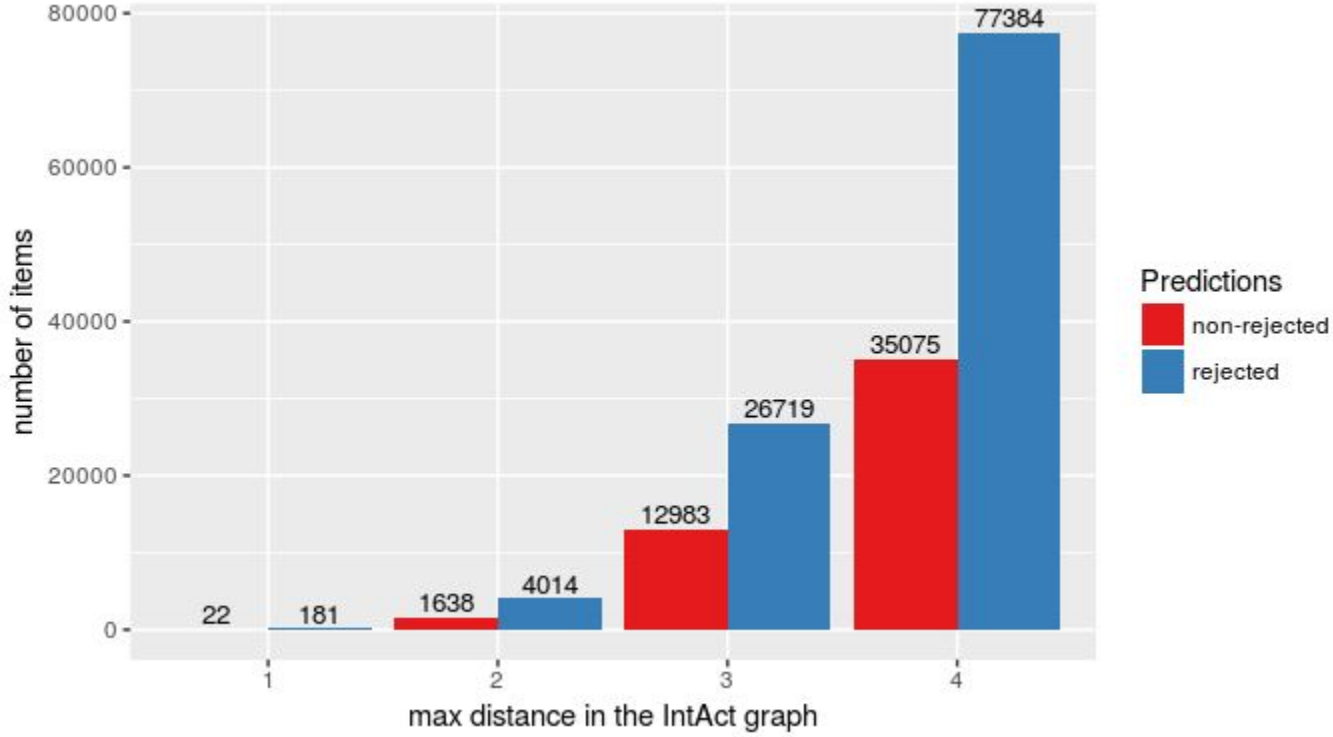
Store dict in pickle file



Prediction results (united graph)



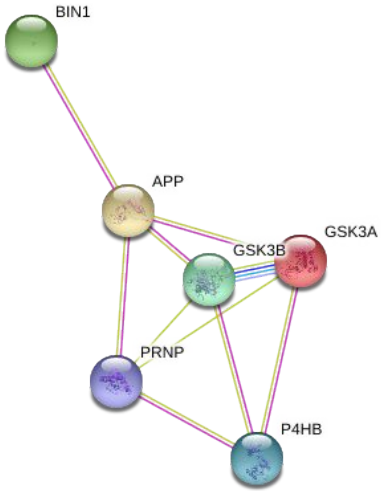
Prediction results (only IntAct)



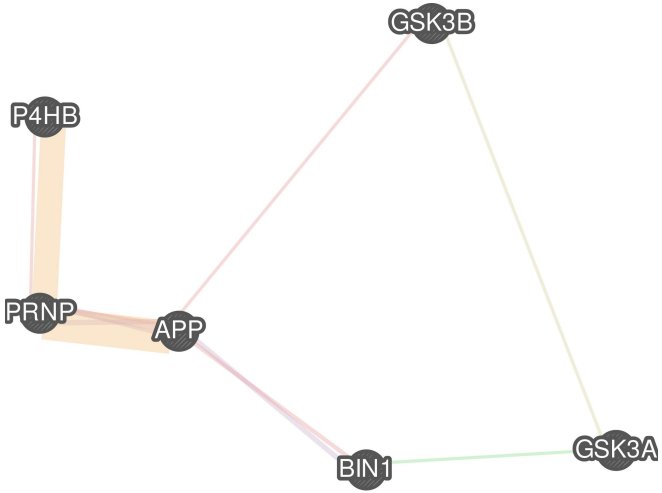
Comparison with STRING and GeneMania

Our classifier predicted 5 PPIs for **BIN1**: APP, GSK3A, GSK3B, PRNP, P4HB

STRING suspects one of them (with APP):
experimentally determined, *text mining*



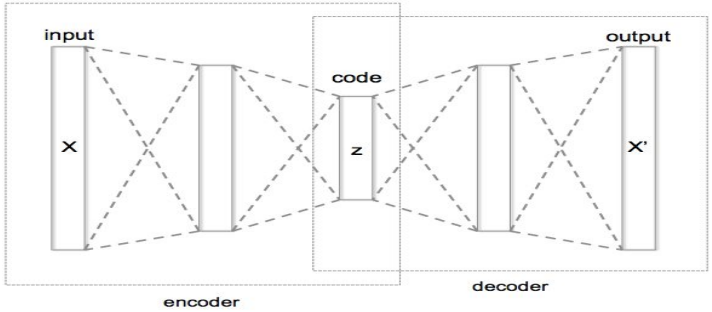
… so does GeneMania:
physical interaction, *co-expression*

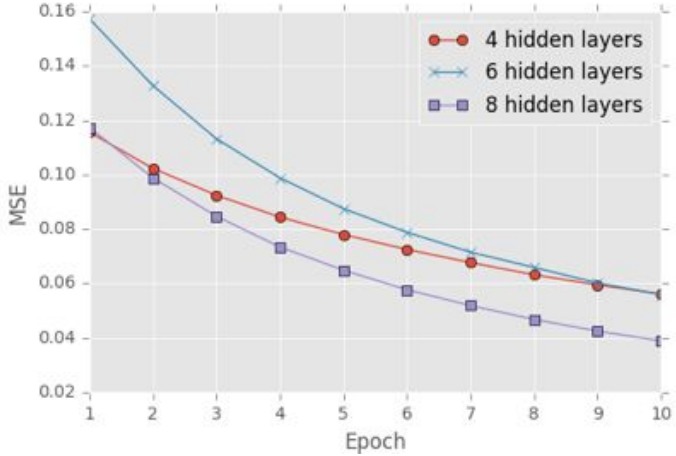


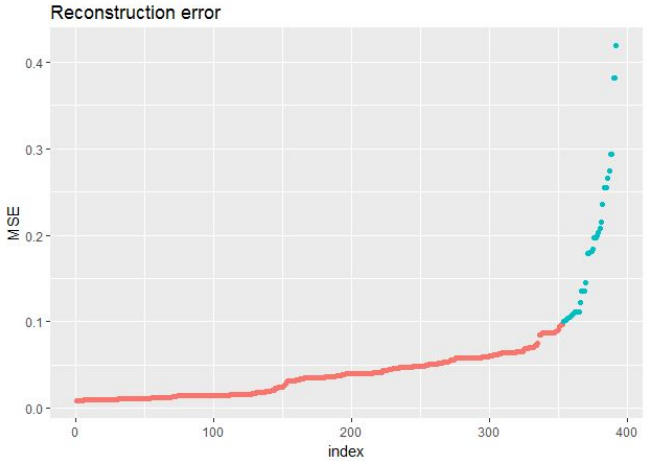
Denoising approaches (autoencoders)

Why use autoencoders?

- Denoise data
- Reduce dimensionality
- Suppress anomalies







Further work...

1. Carefully consider data intersections in training sets
2. Experiment with different sets of predictors
3. Make iterative adjustment of two-staged classifier (in EM-like fashion)
4. Search for extra datasets
5. Try classifier for prediction of PPI in other diseases
6. Try denoising approaches such as autoencoders

We are done...

Thank you!

How does one-class SVM work?