1 June 2022

SynSig Ontology

Sukhada Ghewari

Talk layout

- Introduction about Synsig
- Workflow for ontology building
  - Calculate Similarity Matrix (combined)
  - Apply hidef
  - Parameter Sweep results (table)
  - Align with Syngo
  - Visualize few good ones (cytoscape)
  - Find the best one (quantitative+heuristic)
- Future Scope

Motivation

- Create an ontology for synapse similar to the one we have for Cancer-DrugCell.
- CUL3, DDX3X, YBX1- Karen has found that these genes related to autism are found in synapse.
- Will be helpful to uncover new systems in the synapse.

Comparison between SynSig and SynGO_CC (2018)

SynSig

Number of genes: 1720

Number of common genes: 717

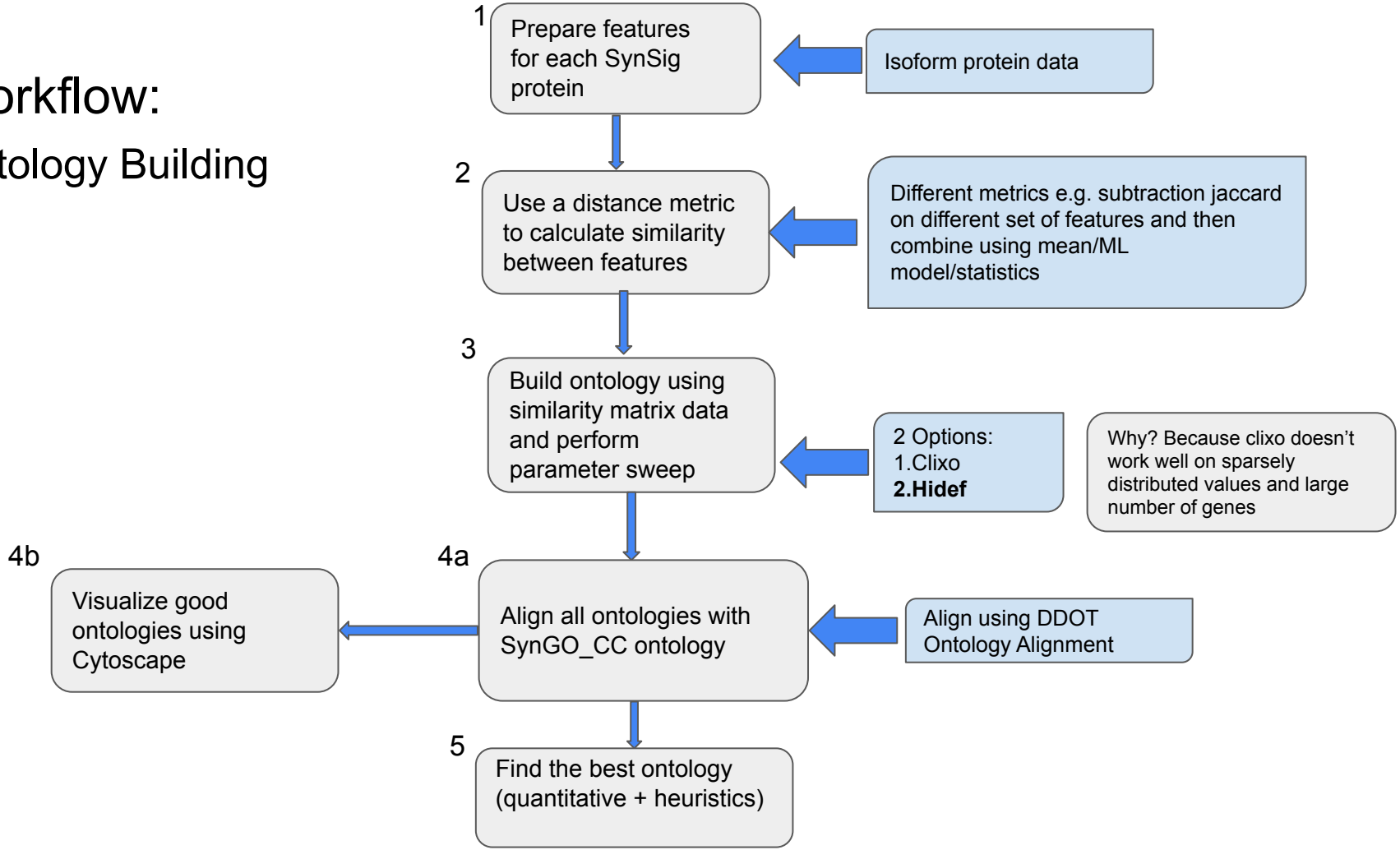
SynGO_CC (2018)

Number of genes: 1112

Number of common genes: 717

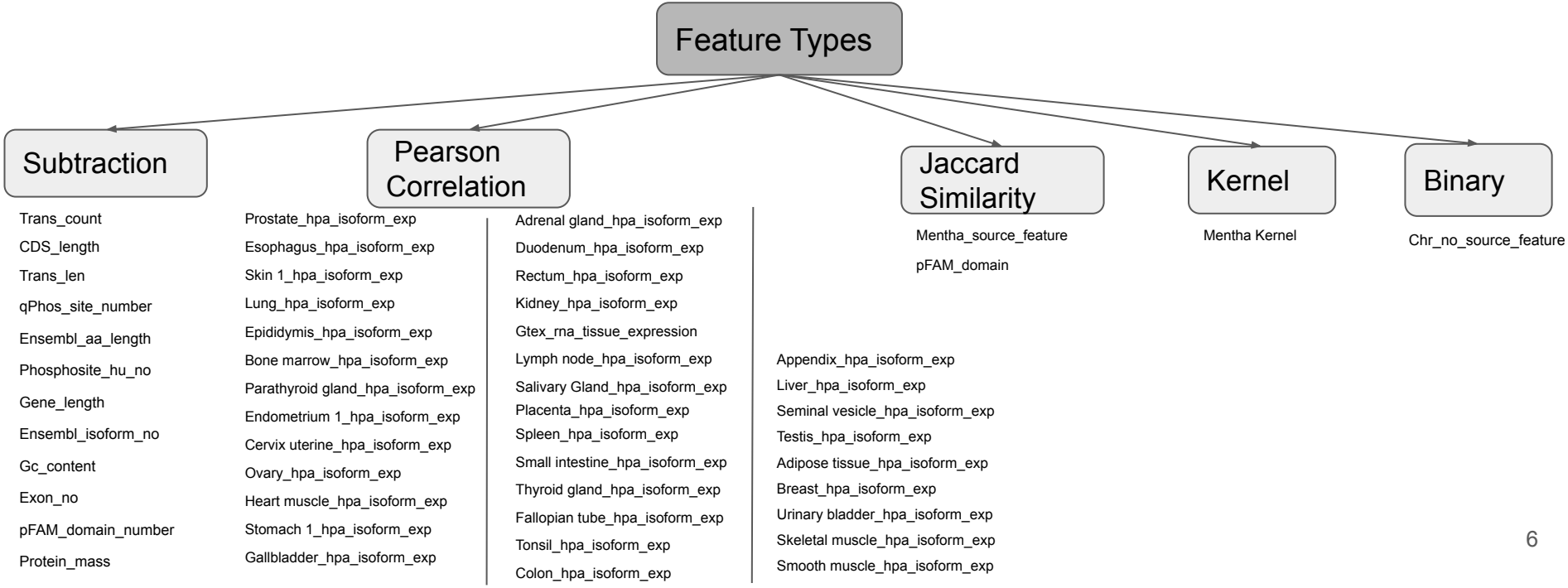
Workflow:

Ontology Building



1. Prepare Vectors for SynSig proteins

SynSig data consists of 51 different types of features that can be broadly classified by the feature type and comparison metric used.



2. Use a distance metric to calculate similarity between features

2 Choices

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Cosine Similarity

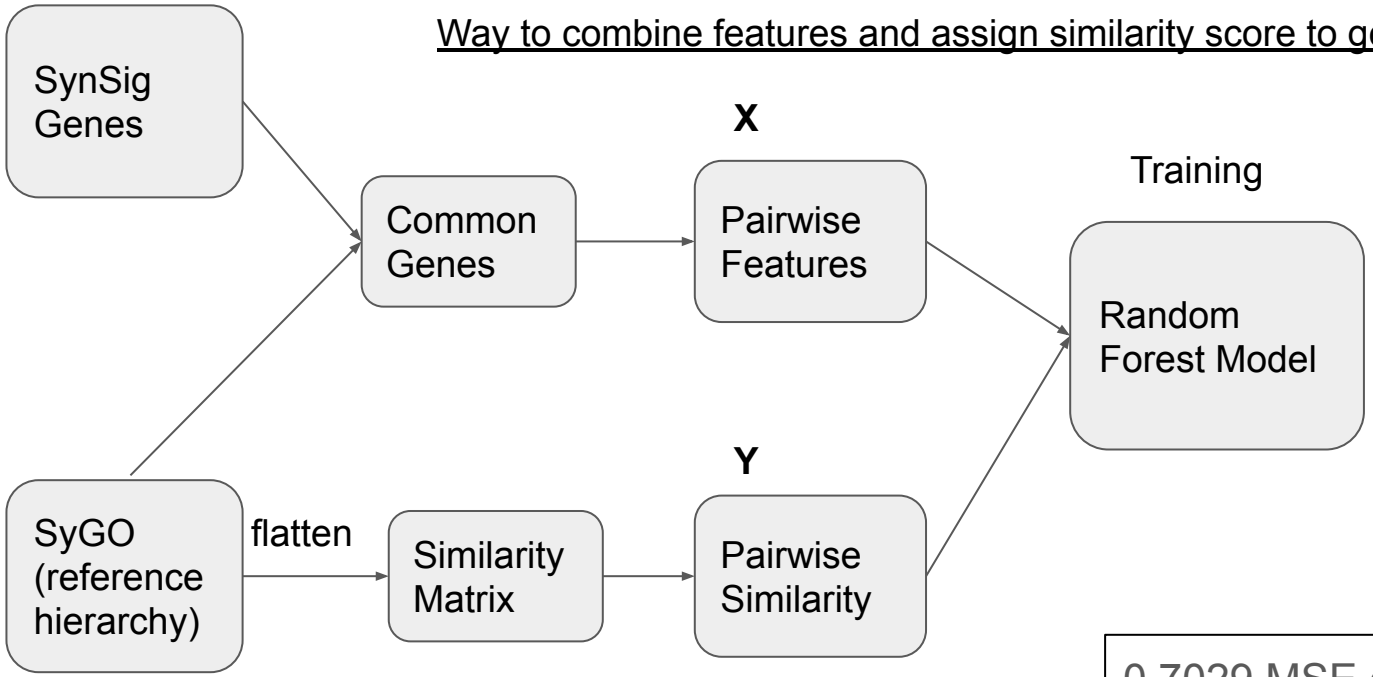
- Combine all feature values in a single giant vector
- Each protein will be represented by its own feature vector
- Calculate the similarity of each protein pair using cosine similarity
- Disadvantage: Not a single similarity metric like this is well suited for all kinds of feature types.

Integrated score using different similarity metrics

- Calculate similarity values based on feature type
- 51 different similarity values (I used subtraction, Jaccard similarity, and Pearson correlation coefficient)
- Challenge: how to integrate them and make sense of them?

2. Use a distance metric to calculate similarity between features

Training Phase: Random Forest



Way to combine features and assign similarity score to gene pairs:

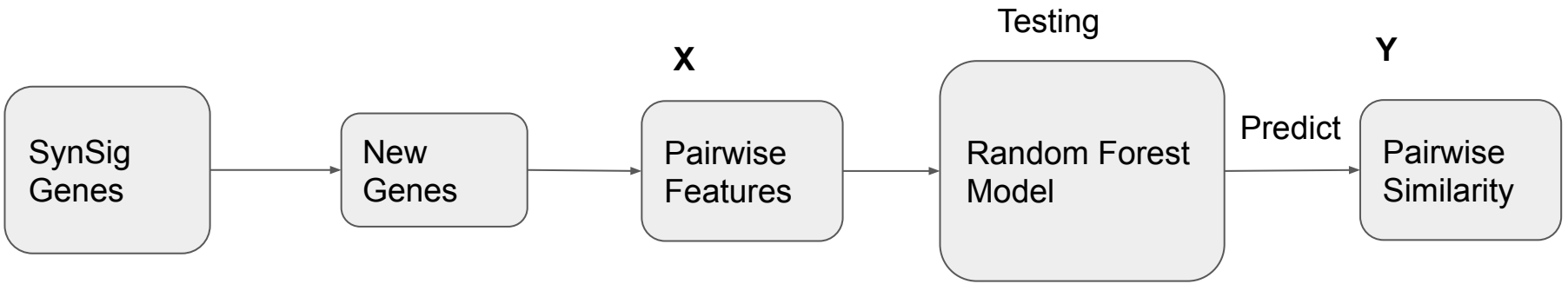
Why Random Forest?
-Ensemble learning technique
-Reduces overfitting
-Reduces Variance

0.7029 MSE on validation set.

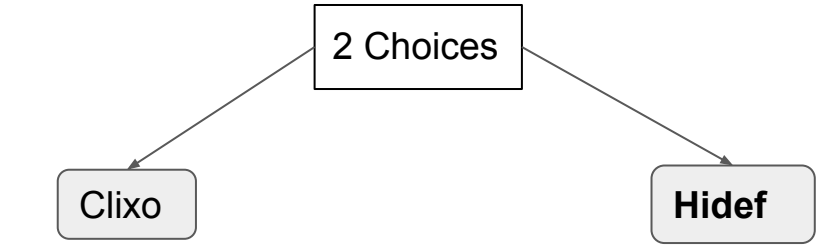
2. Use a distance metric to calculate similarity between features

Testing Phase: Random Forest

Way to combine features and assign similarity score to gene pairs:



3. Build ontology using similarity matrix data and perform parameter sweep



./clixo SIMILARITY_NETWORK_FILE ALPHA BETA

ALPHA is a noise parameter - Can roughly be thought of as representing a significant difference in the similarity values

BETA is a parameter which deals with missing edges

Try out different alpha and beta values to get different hierarchies.

python hidef_finder.py --g $graph --maxres $n --o $out

Both algorithms take similarity network file/graph in the format of (node1, node2, weight)

3. Build ontology using similarity matrix data and perform parameter sweep

3 Parameters in HIDEF swept in search of best ontology.

- **Algorithm**: - -alg (default: louvain)
- **Maxres**: - -maxres (default: 50)
  - increase to get smaller communities
- Consensus Threshold: - -ct (default: 75)
  - threshold for collapsing graph
- **Persistent Threshold**: - -k (default: 5)
  - increase to delete unstable clusters and get fewer communities

I found out that consensus threshold does not have much impact on variability of ontologies. Hence, I set it to default (75).

3. Build ontology using similarity matrix data and perform parameter sweep

The parameter search swept over maxres and persistent threshold.

For alg in [‘leiden’, ‘louvain’]:

For maxres in [10, 20, 30, 40, 50]:

For persistent threshold in [2, 3, 4, 5, 6, 7, 8]:

- Generate ontology OT
- Align ontology OT with syngo_cc
- Track information e.g. #nodes, #edges, #aligned systems, #significant systems, ratio of aligned systems, ratio of significant systems etc.

Total generated ontologies = 2*5*7 = **70**

4a. Align all ontologies with SynGO_CC ontology

Alignment looks like this:

SynSig Term	Syngo Term	Alignment Score	FDR	Number of genes in the term
Cluster1-2	presynapse	0.345703	0.000000	194
Cluster0-0	synapse	0.258070	0.800000	1720
Cluster1-0	postsynapse	0.062739	0.686667	588
Cluster2-7	postsynaptic endocytic zone	0.066672	0.640000	7
Cluster2-6	postsynaptic actin cytoskeleton	0.397524	0.000000	7
Cluster2-4	extrinsic component of postsynaptic membrane	0.124892	0.000000	11
Cluster2-3	postsynaptic Golgi apparatus	0.062381	0.217500	15
Cluster2-32	postsynaptic specialization	0.016969	1.220000	2
Cluster2-1	postsynaptic specialization, intracellular com...	0.089045	0.003333	30
Cluster1-9	synaptic cleft	0.018533	0.992500	5
Cluster1-8	extrasynaptic space	0.018605	1.296670	6
Cluster1-7	postsynaptic cytoskeleton	0.315895	0.000000	9
Cluster1-6	postsynaptic cytosol	0.419312	0.000000	12
Cluster1-5	postsynaptic membrane	0.123476	0.000000	17
Cluster1-4	postsynaptic ribosome	0.257439	0.000000	28
Cluster1-3	postsynaptic density	0.243037	0.000000	85
Cluster1-28	postsynaptic mitochondria	0.011753	3.740000	3
Cluster1-24	postsynaptic ER	0.011754	11.570000	4
Cluster1-22	postsynaptic endosome	0.011724	1.885000	3
Cluster1-1	synaptic membrane	0.024022	1.242500	422
Cluster1-16	spine apparatus	0.011736	1.700000	2
Cluster1-10	integral component of synaptic membrane	0.194715	0.220000	2

Call a system significant if its FDR <= 0.05

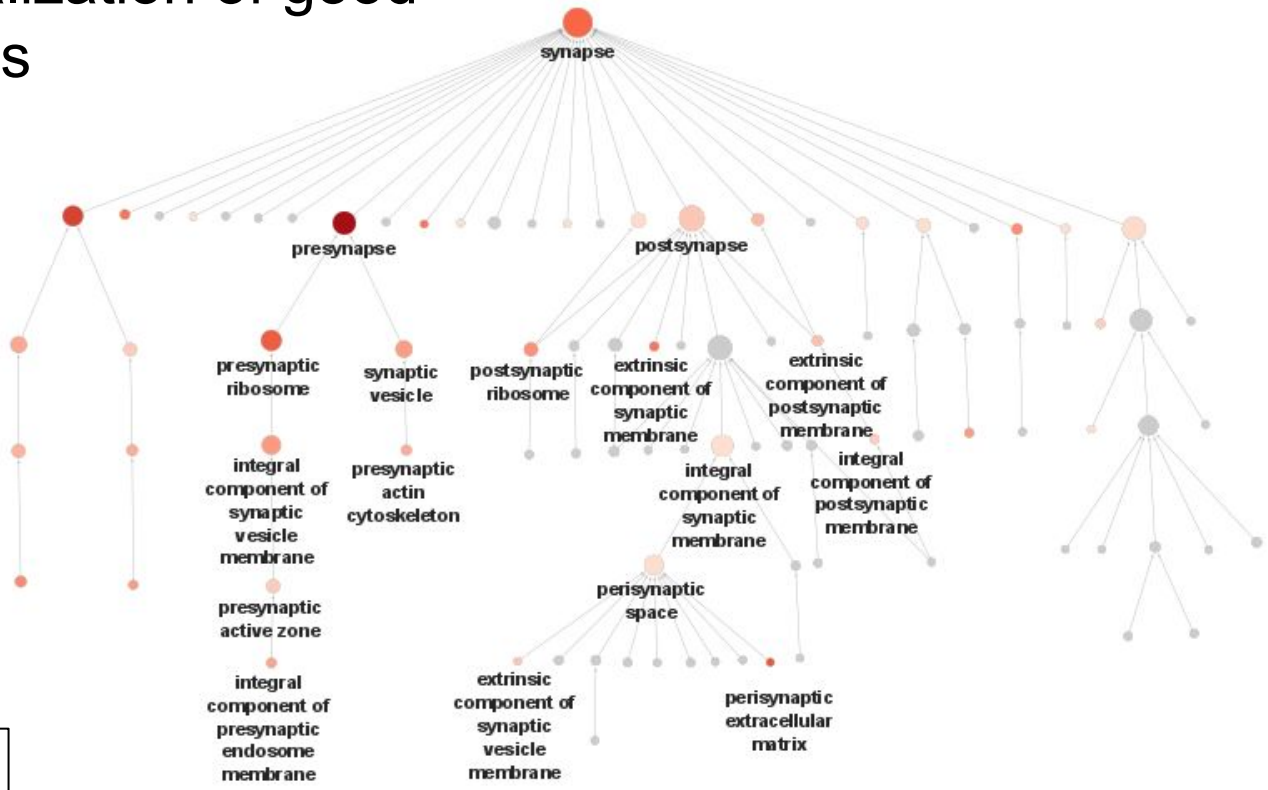
4a. Align all ontologies with SynGO_CC ontology

Algorithm	MaxRes	Consensus Threshold	Persistent Threshold	Number of Nodes	Number of Edges	MaxDepth	Number of nodes aligned to SynGO	Ratio of Aligned Systems	Number of nodes with FDR <= 0.05 (Significant Systems)	Ratio of Significant Systems
louvain	10	75	2	316	341	5	41	0.129746835	10	0.243902439
leiden	10	75	2	328	341	6	39	0.118902439	8	0.205128205
leiden	10	75	3	125	125	5	27	0.216	7	0.259259259
louvain	10	75	3	104	105	5	27	0.259615385	7	0.259259259
leiden	10	75	4	59	58	4	22	0.372881356	8	0.363636364
louvain	10	75	4	55	57	4	20	0.363636364	8	0.4
louvain	10	75	5	34	33	3	34	1	9	0.264705882
leiden	10	75	5	48	47	3	23	0.479166667	11	0.47826087
leiden	10	75	6	35	34	2	21	0.6	7	0.333333333
louvain	10	75	6	27	26	4	21	0.777777778	10	0.476190476
louvain	10	75	7	23	22	2	21	0.913043478	10	0.476190476
leiden	10	75	7	24	23	3	23	0.958333333	10	0.434782609
louvain	10	75	8	17	16	2	17	1	9	0.529411765
leiden	10	75	8	24	23	2	21	0.875	8	0.380952381
leiden	20	75	2	340	446	9	41	0.120588235	11	0.268292683
louvain	20	75	2	385	408	8	33	0.085714286	7	0.212121212
leiden	20	75	3	124	129	5	30	0.241935484	8	0.266666667
louvain	20	75	3	118	121	7	25	0.211864407	10	0.4
louvain	20	75	4	65	64	5	22	0.338461538	9	0.409090909
leiden	20	75	4	76	78	4	22	0.289473684	9	0.409090909
leiden	20	75	5	58	58	4	21	0.362068966	4	0.19047619
louvain	20	75	5	48	47	3	23	0.479166667	8	0.347826087
louvain	20	75	6	39	38	3	23	0.58974359	9	0.391304348
leiden	20	75	6	45	44	3	23	0.511111111	10	0.434782609
louvain	20	75	7	31	30	3	23	0.741935484	12	0.52173913
leiden	20	75	7	32	31	3	23	0.71875	10	0.434782609
louvain	20	75	8	28	27	2	23	0.821428571	9	0.391304348
leiden	20	75	8	26	25	3	22	0.846153846	9	0.409090909
leiden	30	75	2	345	360	6	38	0.110144928	6	0.157894737
louvain	30	75	2	319	340	6	34	0.106583072	8	0.235294118
leiden	30	75	3	125	129	6	28	0.224	9	0.321428571
louvain	30	75	3	120	129	4	29	0.241666667	7	0.24137931
louvain	30	75	4	69	69	5	22	0.31884058	5	0.227272727

4a. Align all ontologies with SynGO_CC ontology

leiden	30	75	4	72	73	5	22	0.305555556	8	0.363636364
louvain	30	75	5	51	51	5	22	0.431372549	6	0.272727273
leiden	30	75	5	52	51	3	25	0.480769231	9	0.36
louvain	30	75	6	34	33	3	23	0.676470588	12	0.52173913
leiden	30	75	6	39	39	3	22	0.564102564	7	0.318181818
louvain	30	75	7	26	25	3	24	0.923076923	12	0.5
leiden	30	75	7	29	28	3	24	0.827586207	11	0.458333333
leiden	30	75	8	29	28	2	23	0.793103448	9	0.391304348
louvain	30	75	8	25	24	2	22	0.88	9	0.409090909
leiden	40	75	2	306	323	6	36	0.117647059	10	0.277777778
louvain	40	75	2	328	341	7	33	0.100609756	6	0.181818182
louvain	40	75	3	113	117	5	26	0.230088496	7	0.269230769
leiden	40	75	3	127	130	6	31	0.244094488	13	0.419354839
louvain	40	75	4	73	73	5	21	0.287671233	6	0.285714286
leiden	40	75	4	65	64	5	23	0.353846154	8	0.347826087
leiden	40	75	5	55	55	4	24	0.436363636	6	0.25
louvain	40	75	5	51	50	3	25	0.490196078	11	0.44
leiden	40	75	6	43	42	3	23	0.534883721	10	0.434782609
louvain	40	75	6	39	38	3	24	0.615384615	10	0.416666667
louvain	40	75	7	34	33	3	24	0.705882353	12	0.5
leiden	40	75	7	31	30	3	24	0.774193548	11	0.458333333
louvain	40	75	8	27	26	2	22	0.814814815	11	0.5
leiden	40	75	8	25	24	3	22	0.88	12	0.545454545
leiden	50	75	2	332	363	7	39	0.11746988	9	0.230769231
louvain	50	75	2	339	363	6	44	0.12979351	8	0.181818182
leiden	50	75	3	133	138	7	28	0.210526316	9	0.321428571
louvain	50	75	3	130	134	5	22	0.169230769	10	0.454545455
leiden	50	75	4	70	71	5	22	0.314285714	6	0.272727273
louvain	50	75	4	69	71	5	23	0.333333333	7	0.304347826
louvain	50	75	5	50	51	4	23	0.46	6	0.260869565
leiden	50	75	5	53	52	4	25	0.471698113	7	0.28
louvain	50	75	6	42	41	3	23	0.547619048	10	0.434782609
leiden	50	75	6	43	42	3	23	0.534883721	8	0.347826087
leiden	50	75	7	34	33	2	23	0.676470588	9	0.391304348
louvain	50	75	7	28	27	3	23	0.821428571	8	0.347826087
louvain	50	75	8	27	26	4	22	0.814814815	11	0.5
leiden	50	75	8	30	29	3	22	0.733333333	11	0.5

4b. Visualization of good ontologies

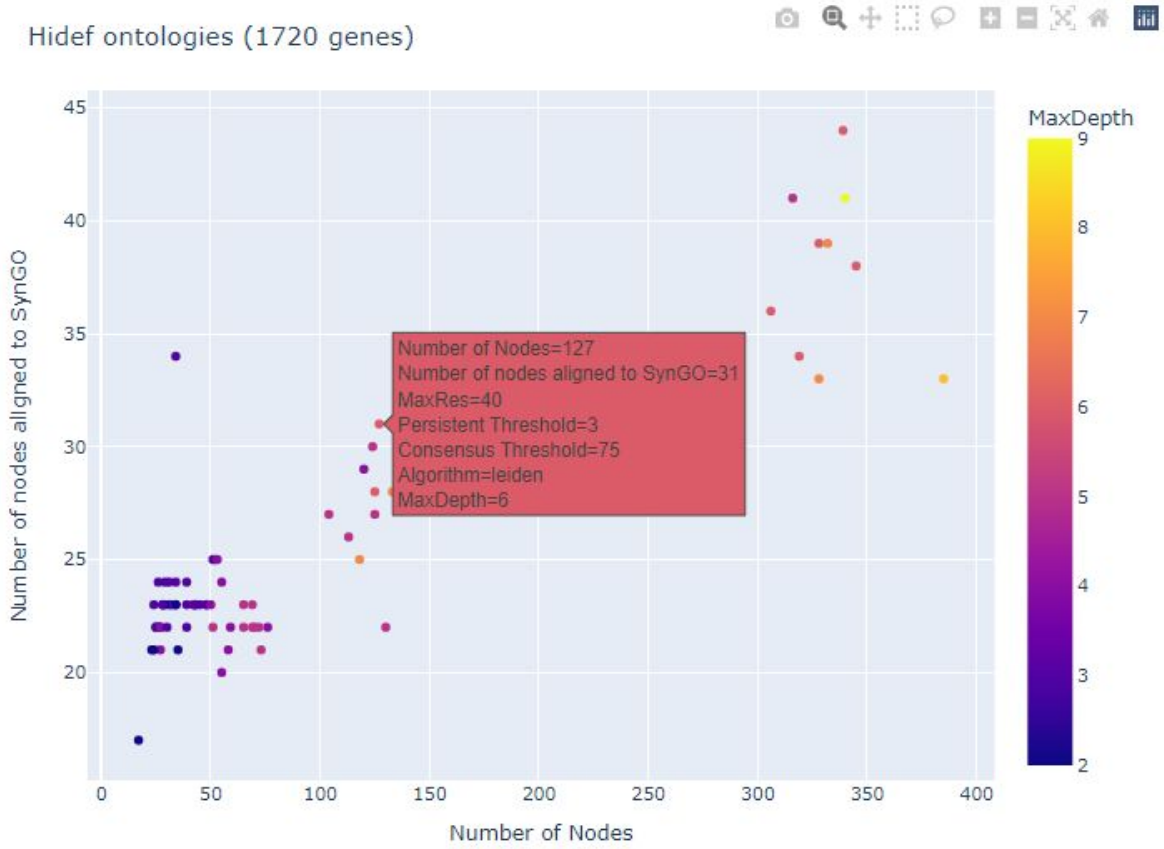


Hidef:
Algo: louvain
MaxRes: 50
Consensus Threshold: 75
Persistent Threshold: 5

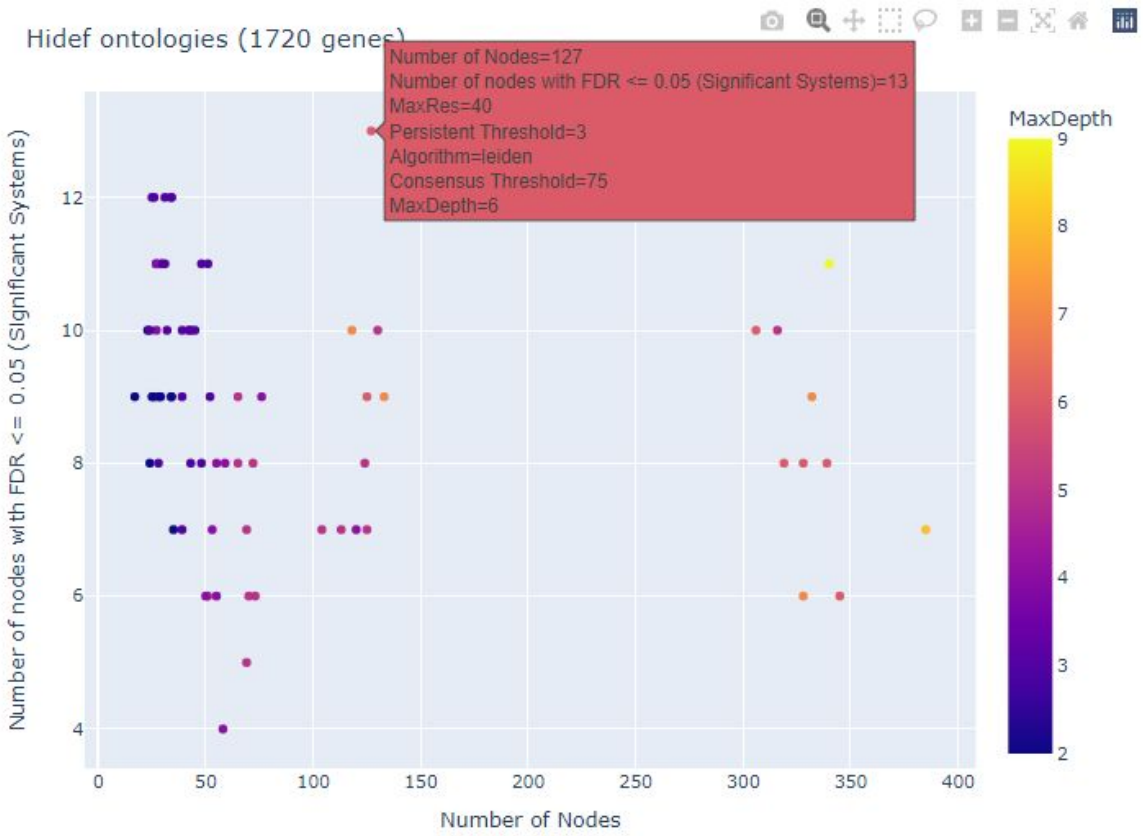
5. Find the best ontology

- There is no single quantitative measure that tells us which generated ontology is the best.
- It is a combination of quantitative measure and heuristics.
- What are we looking for?
  - Deep or shallow ontology?
  - More aligned systems or less?
  - Which ontology best describes the biology of the synapse after alignment?
- We want an ontology in which:
  - No. of nodes is good enough (not very less, not very high)
  - Good number of aligned systems but there is still a scope to find new systems
    - Exploitation vs Exploration trade-off

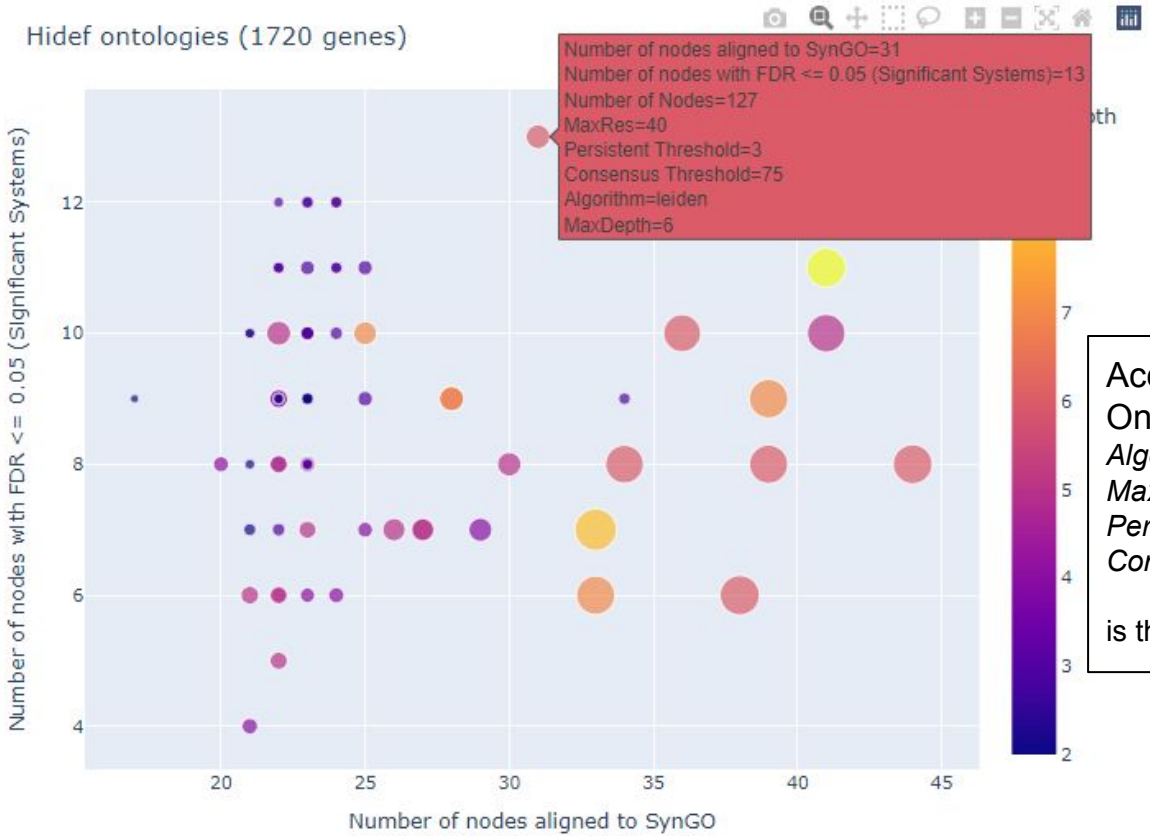
5. Find the best ontology



5. Find the best ontology



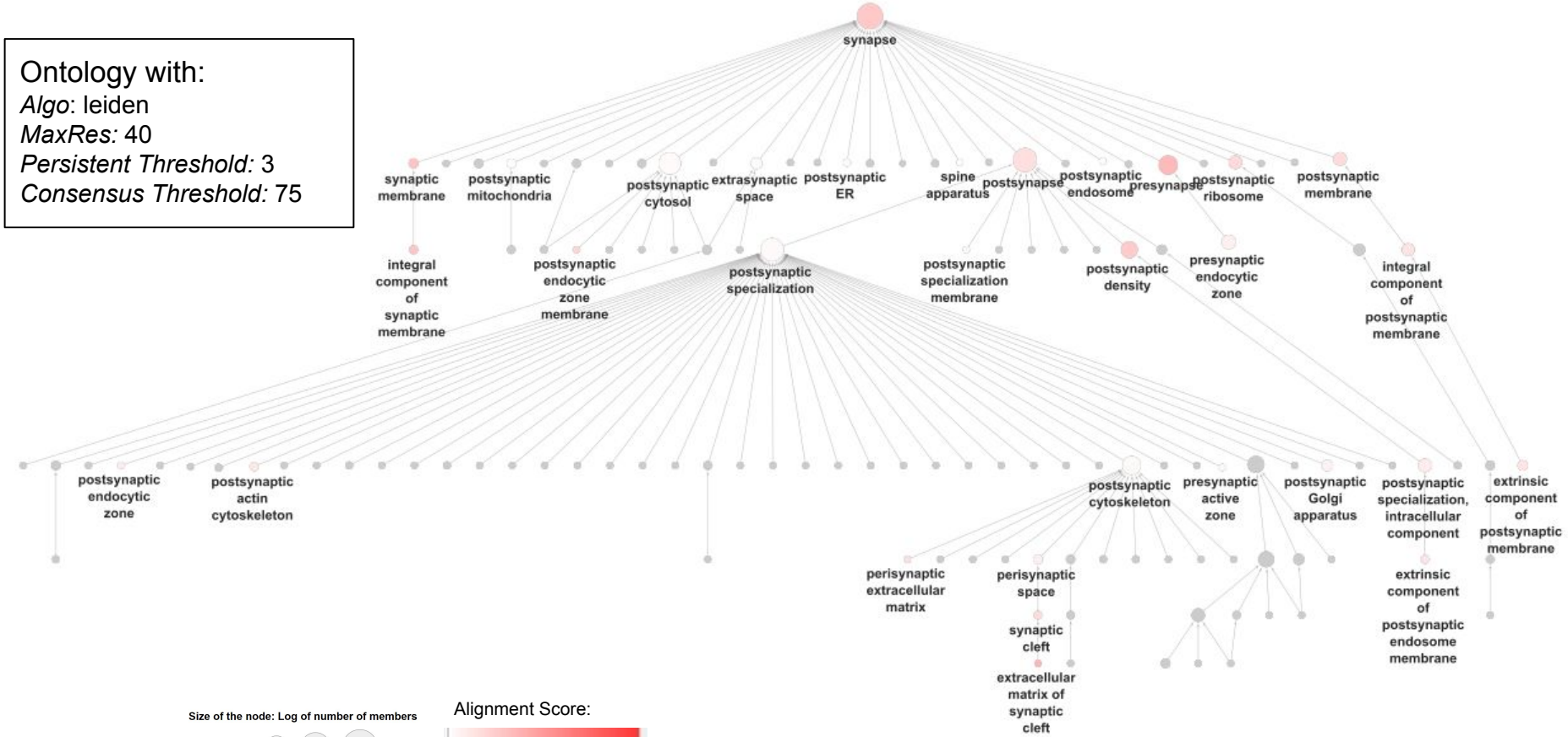
5. Find the best ontology



According to these 3 plots, Ontology with:
Algo: leiden
MaxRes: 40
Persistent Threshold: 3
Consensus Threshold: 75

is the best!

Ontology with:
Algo: leiden
MaxRes: 40
Persistent Threshold: 3
Consensus Threshold: 75



5. Find the best ontology

A possible quantitative method to rank ontologies:

- Normalize the features we are tracking e.g. number of systems, maxdepth, number of systems aligned with SynGO, number of significant systems etc. across ontologies:
  - All values will range from (0,1)
- Call a feature additive if we want to see higher value for it (e.g.number of systems aligned with SynGO)
- Call a feature subtractive if we want to see lower value for it (e.g. maxdepth)
- Call a feature mean if we want to see some middle value (e.g. number of systems)

$$\text{rank(ontology)} = \sum_{a \text{ in additive}} a + \sum_{s \text{ in subtractive}} (1-s) + \sum_{m \text{ in mean}} |\text{mean}(M) - m|$$

Future Scope

- Try different ranking methods to find good ontologies. The aim is that this process should be fully automated or semi-automated with some expert inputs.
- Build a neural network based on the finalized ontology.
- This neural network can be used for an application such as a prediction task.
    - A prediction task can be : give a person’s synaptic protein signature, predicting if that person has/will have a disease linked to synapse proteins such as ASD (autism spectrum disorder)

Acknowledgement

Thanks:

- Dr. Trey Ideker
- Dr. Karen Mei
- Chris Churas
- Sahar Alkhairy
- Charlotte Marquez
- To all the lab members who helped with my questions on Slack