Our analysis for sub-challenge 4: species specific network inference

Jie Cheng (email: jie.j.cheng@gsk.com)
Quantitative Sciences
GlaxoSmithKline

Our result

- Congratulations to the top performing teams!
- We did not do well in the sub-challenge
- Our result was just slightly better than random
  - TPR - FPR Rat : 0.029
  - TPR - FPR Human: 0.008
  - TPR - FPR rank: 5
- Goal: learn to improve the analysis

Data preprocessing

1. Within each batch, we averaged the repeats and subtracted mean control value. For value< threshold, we set value to zero.
2. For gene expression data, we removed genes that have more than 21 zeros under 26 stimulus conditions.
3. For phospho data, we combined 5 min and 25 min data by taking the value that has larger absolute value.
4. Combine data from all three platforms and select variables that are included in the reference network

- Steps 1-3 are ad hoc procedures
- Step 4 may be more problematic: probably should include more variables that are related to the variables in the reference network (did not have time; lack of knowledge in biology)

Learning Bayesian network models

- Our java tool for learning Bayesian networks / graphical Gaussian models based on our paper [Cheng et al, 2002]
  - Cheng J and Greiner R, Learning Bayesian Networks from Data: an Information-Theory Based Approach, the Artificial Intelligence Journal, Vol 137, and Pages 43-90, 2002.

- We run bootstrap 100 times and count the frequency of each edge
- Modifying the reference network
  - remove an edge if both variables are in our data and the edge has not appear in our bootstrap runs
  - Add edge if it appears more than 40% of the time in bootstrap runs

Parsimonious dependency analysis

Phases:

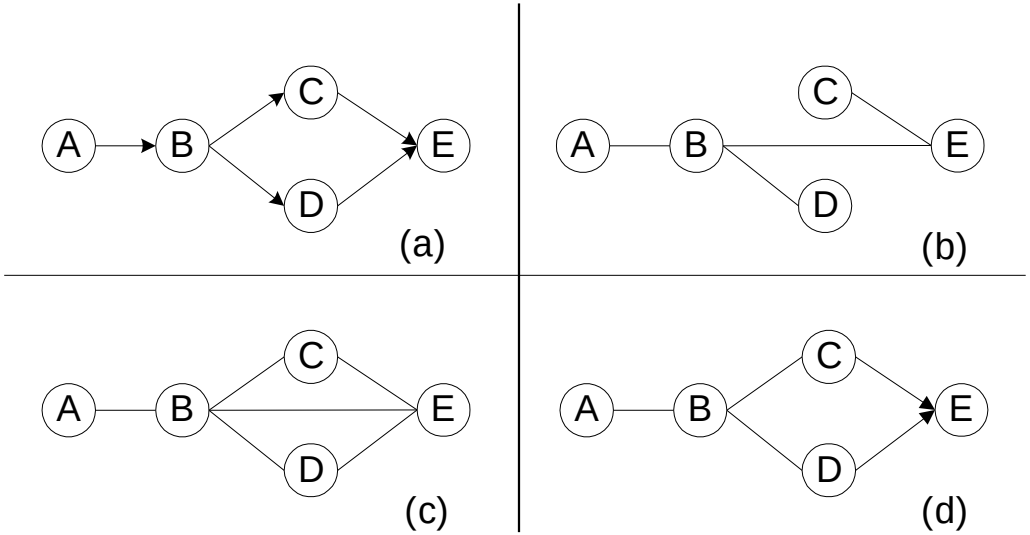
- Drafting

  All pair-wise correlations; Maximum weight spanning tree

- Thickening

- Thinning

- Edge orientation



[{*B*,*D*}, {*C*,*E*} {*B*,*E*}, {*A*,*B*}, {*B*,*C*}, {*C*,*D*}, {*D*,*E*}, {*A*,*D*}, {*A*,*E*}, {*A*,*C*}]

Simulation result of our approach

Variable	Samples	True Edges	Run	Discovered Edges	Number of True Pos.				Sum of squared error				PDA cpu time (sec)
					GeneNet	G. Lasso	space	PDA	GeneNet	G. Lasso	space	PDA	
40	40	40	1	30	21	23	23	28	9.35	9.87	6.47	2.25	0.15
40	40	40	2	38	26	21	28	31	11.59	13.69	4.46	2.19	0.15
40	40	40	3	36	22	21	27	29	10.22	11.71	6.25	2.96	0.15
40	30	40	1	30	18	15	23	24	10.99	13.86	10.27	4.66	0.15
40	30	40	2	33	21	22	24	27	11.92	11.62	7.46	4.14	0.15
40	30	40	3	27	18	15	20	24	10.78	14.23	11.37	3.32	0.15
40	20	40	1	25	16	12	13	20	10.96	11.65	10.26	7.72	0.15
40	20	40	2	25	13	17	17	18	11.51	11.05	7.68	7.50	0.15
40	20	40	3	34	19	19	22	29	12.85	15.26	11.19	4.82	0.15
200	20	200	1	150	81	71	88	110	76.12	69.69	53.00	30.97	1.5
200	20	200	2	143	79	66	80	100	65.99	62.71	50.13	37.37	1.5
200	20	200	3	208	85	78	114	147	80.75	75.98	46.38	28.56	1.5
200	100	300	1	199	159	146	165	182	52.67	38.63	25.41	11.07	3
200	100	300	2	196	159	150	172	177	51.68	34.76	22.70	12.30	3
200	100	300	3	194	165	154	176	180	51.93	38.44	26.04	10.74	3

Average true positives rate:
GeneNet 0.64, G. Lasso 0.60, space 0.71, PDA 0.82

Possible ways to improve our analysis

- Learn other types of networks?
- Include more variables in network learning
- Learn network without using the reference network?
- Get direction of edge from biological knowledge; Bayesian net learning algorithms are weak at determine the direction of edges