DSCI 573:
Model Selection and Feature Selection

Structure Learning

Winter 2018

Structure Learning: Unsupervised Feature Selection

- “News” data: presence of 100 words in 16k newsgroup posts:

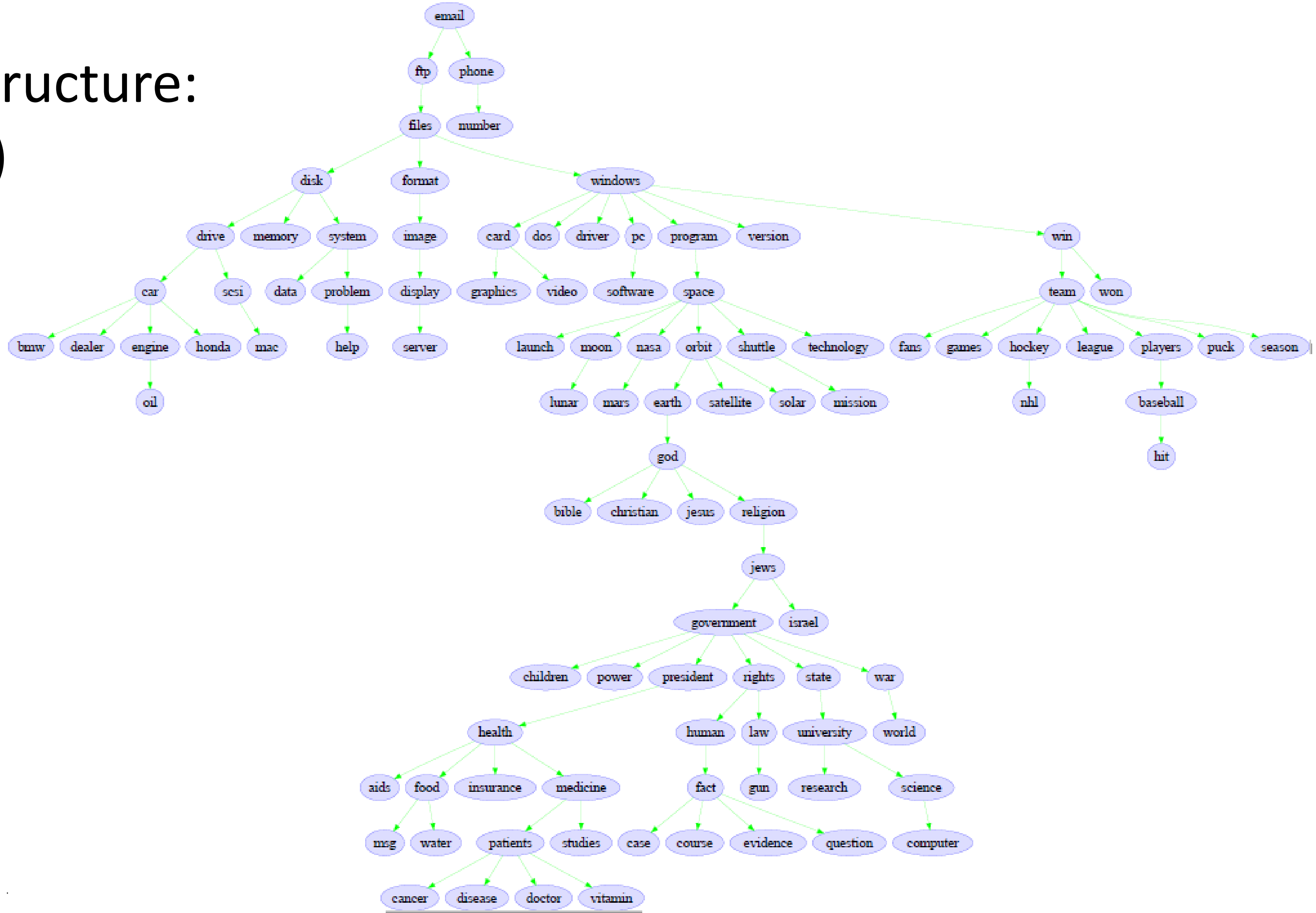
car	drive	files	hockey	mac	league	pc	win
0	0	1	0	1	0	1	0
0	0	0	1	0	1	0	1
1	1	0	0	0	0	0	0
0	1	1	0	1	0	0	0
0	0	1	0	0	0	1	1

- Which words are related to each other?

- Problem of structure learning: unsupervised feature selection.

Structure Learning: Unsupervised Feature Selection

- Optimal tree structure: (ignore arrows)



Naïve Approach: Association Networks

- A naïve approach to structure learning ("association networks"):
  - For each pair of variables, compute a measure of similarity or dependence.

- Using these n^2 similarity values either:
  - Select all pairs whose similarity is above a threshold.
  - Select the "top k" most similar features to each feature 'j'.

- Main problems:
  - Usually, most variables are dependent (too many edges).
    - "Sick" is getting connected to "Tuesdays" even if "tacos" are a variable.
  - "True" neighbours may not have the highest dependence.
    - "Sick" might get connected to "Tuesdays" before it gets connected to "milk".

- (Variation: best tree can be found as minimum spanning tree problem.)

Example: Vancouver Rain Data

- Consider modeling the "Vancouver rain" dataset.

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	…
Month 1	0	0	0	1	1	0	0	1	1	
Month 2	1	0	0	0	0	0	1	0	0	
Month 3	1	1	1	1	1	1	1	1	1	
Month 4	1	1	1	1	0	0	1	1	1	
Month 5	0	0	0	0	1	1	0	0	0	
Month 6	0	1	1	0	0	0	0	1	1	

- The strongest signal in the data is the simple relationship:
  - If it rained yesterday, it's likely to rain today (> 50% chance that $x^{t-1} = x^t$).
  - But an "association network" might connect all days (all dependent).

Dependency Networks

- A better approach is dependency networks:
  - For each variable ‘j’, make it the target in a supervised learning problem.

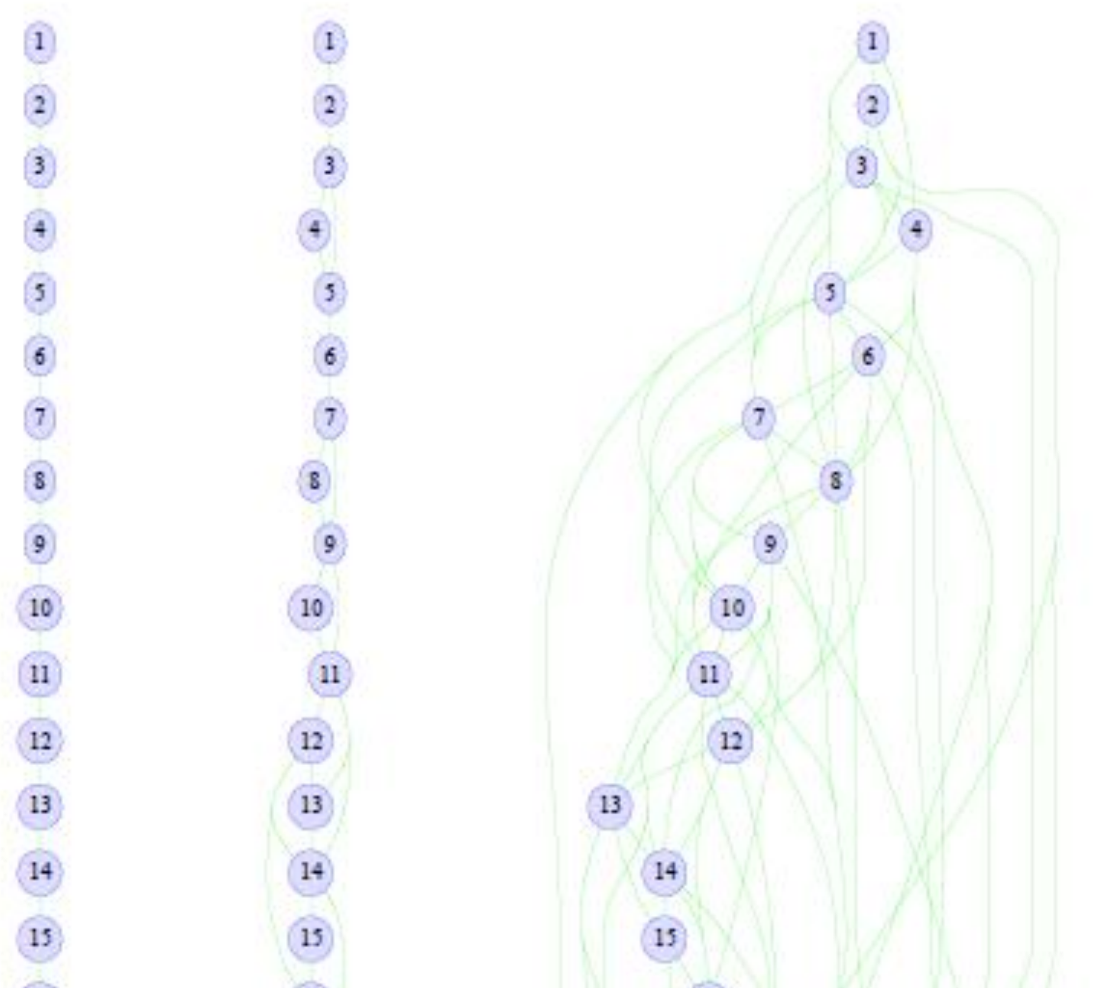
$$X = \begin{bmatrix} | & | & | & | & | \\ x^1 & x^2 & x^3 & x^4 & x^5 \\ | & | & | & | & | \end{bmatrix} \Longrightarrow \bar{X} = \begin{bmatrix} | & | & | & | \\ x^1 & x^2 & x^3 & x^5 \\ | & | & | & | \end{bmatrix} \quad y = \begin{bmatrix} | \\ x^4 \\ | \end{bmatrix}$$

  - Now we can use any feature selection method to choose j’s “neighbours”.
    - Forward selection, L1-regularization, ensemble methods, etc.

- Can capture conditional independence:
  - Might connect “sick” to “tacos”, and “tacos” to “Tuesdays” (w/o sick-tacos).

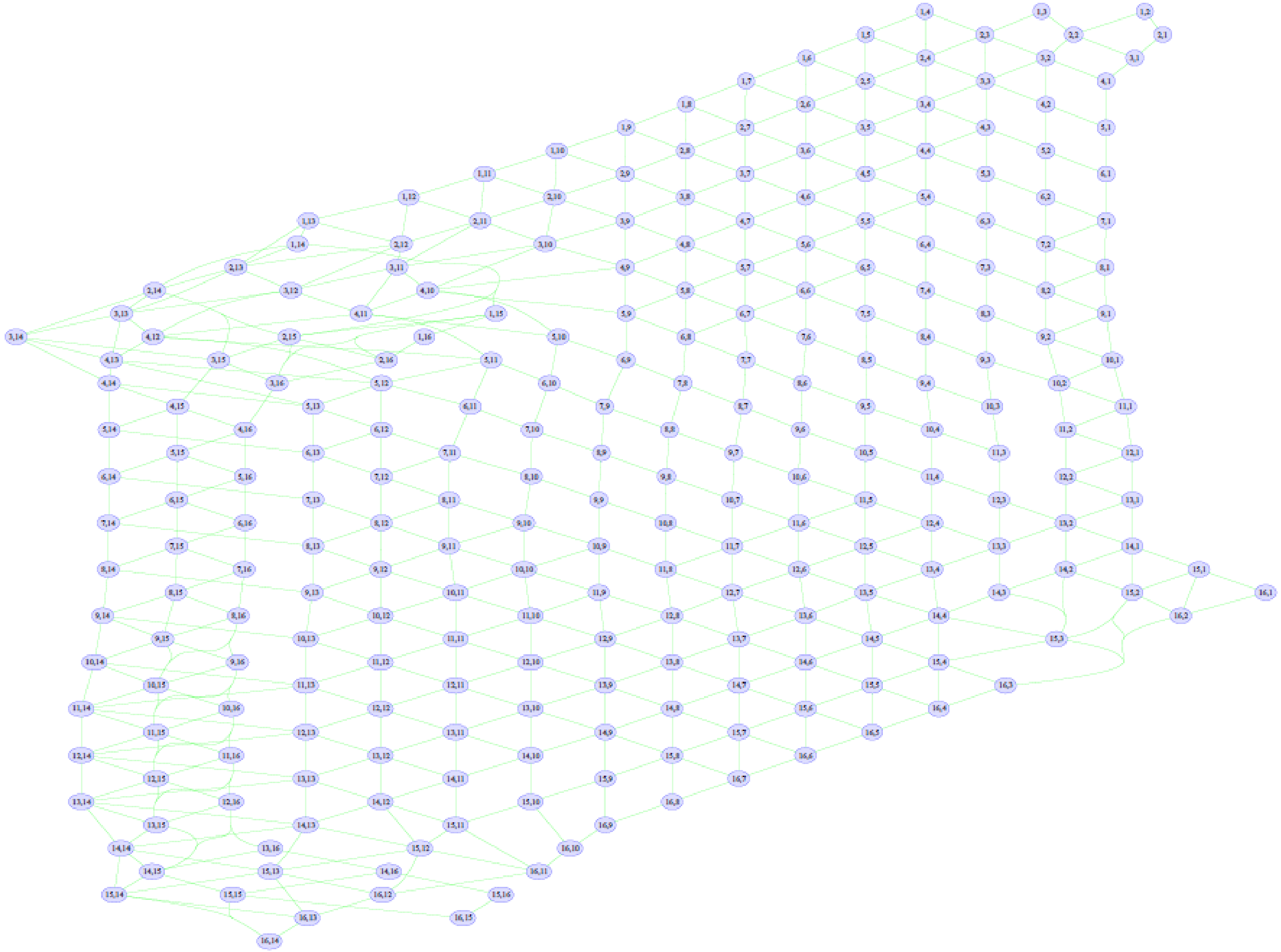
Dependency Networks

- Dependency network fit to Vancouver rain data (different λ values):



Dependency Networks

- Variation on dependency networks on digit image pixels:



Another popular structure learning method is the "PC" algorithm.

Summary

- Structure learning is “unsupervised” feature selection.
- Association networks make graph by finding similar features.
- Dependency networks use feature selection with feature ‘j’ as ‘y’.