CPSC 540: Machine Learning

Directed Acyclic Graphical Models

Mark Schmidt

University of British Columbia

Winter 2019

Last Time: Directed Acyclic Graphical (DAG) Models

- DAG models use a factorization of the joint distribution,

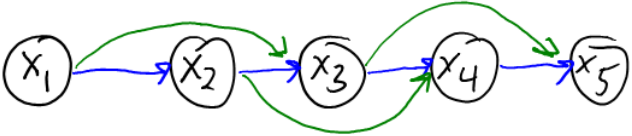
$$p(x_1, x_2, \dots, x_d) = \prod_{j=1}^{d} p(x_j | x_{\text{pa}(j)}),$$

where $\text{pa}(j)$ are the "parents" of node j.

- This assumes a Markov property (generalizing Markov property in chains),

$$p(x_j | x_{1:j-1}) = p(x_j | x_{\text{pa}(j)}),$$

- We visualize the assumptions made by the model as a graph:



DAGs and Conditional Independence

- In DAGs we make the conditional independence assumption that

$$p(x_j \mid x_{j-1}, x_{j-2}, \ldots, x_1) = p(x_j \mid x_{\text{pa}}(j)).$$

- But these conditional independence assumptions can imply other assumptions.
  - For example, in Markov chains we directly assume for all j that

$$p(x_j \mid x_{j-1}, x_{j-2}, \ldots, x_1) = p(x_j \mid x_{j-1}),$$

    but this also implies that

$$p(x_j \mid x_{j-2}, x_{j-3}, \ldots, x_1) = p(x_j \mid x_{j-2}),$$

    and it implies that

$$p(x_j \mid x_{j+1}, x_{j+2}, \ldots, x_d) = p(x_j \mid x_{j+1}).$$

- Knowing which assumptions hold can help identify which operations are efficient.
  - For example, decoding in general DAGs is NP-hard but it's easy in Markov chains.

Review of Independence

- Let A and B are random variables taking values $a \in \mathcal{A}$ and $b \in \mathcal{B}$.
- We say that A and B are independent if we have

$$p(a, b) = p(a)p(b),$$

for all a and b.

- To denote independence of x_i and x_j we use the notation

$$x_i \perp x_j.$$

Review of Independence

- For independent a and b we have

$$p(a \mid b) = \frac{p(a,b)}{p(b)} = \frac{p(a)p(b)}{p(b)} = p(a).$$

- This gives us a more intuitive definition: A and B are independent if

$$p(a \mid b) = p(a)$$

  for all a and $b \neq 0$.
  - In words: knowing b tells us nothing about a (and vice versa).

- Useful fact: $a \perp b$ iff $p(a,b) = f(a)g(b)$ for some functions f and g.

Example: Independence in Product Models

- Recall density estimation using a product of independent models:

$$p(x_1, x_2, \ldots, x_d) = p(x_1)p(x_2)\cdots p(x_d).$$

- Using marginalization rule we can show this implies pairwise independence:

$$p(x_i, x_j) = \underbrace{\sum_{x_1}\sum_{x_2}\cdots\sum_{x_d}}_{\text{excluding } x_i \text{ and } x_j} p(x_1, x_2, \ldots, x_d) = \underbrace{\sum_{x_1}\sum_{x_2}\cdots\sum_{x_d}}_{\text{excluding } x_i \text{ and } x_j} p(x_1)p(x_2)\cdots p(x_d)$$

$$= p(x_i)p(x_j) \underbrace{\sum_{x_1}\sum_{x_2}\cdots\sum_{x_d}}_{\text{excluding } x_i \text{ and } x_j} \underbrace{p(x_1)p(x_2)\cdots p(x_d)}_{\text{excluding } p(x_i) \text{ and } p(x_j)} \quad \text{(take } p(x_i) \text{ and } p(x_j) \text{ out)}$$

$$= p(x_i)p(x_j) \underbrace{\sum_{x_1} p(x_1) \sum_{x_2} p(x_2) \cdots \sum_{x_d} p(x_d)}_{\text{excluding } x_i \text{ and } x_j} = p(x_i)p(x_j) \quad \text{(sums sum to 1).}$$

Example: Independence in Product of Bernoullis Model

- In a product of Bernoullis probabilities model we have

$$p(x_1, x_2, \ldots, x_d) = p(x_1)p(x_2)\cdots p(x_d),$$

  which we showed implies

$$p(x_i, x_j) = p(x_i)p(x_j),$$

  so we have $x_i \perp x_j$ for all i and j.

- In mixture of Bernoullis x_i is not independent of x_j ($x_i \not\perp x_j$):
  - Knowing x_j tells you something about x_i.
  - But similar notation-heavy steps give the conditional independence that

$$p(x_i, x_j \mid z) = p(x_i \mid z)p(x_j \mid z),$$

    that "variables x_i and x_j are conditionally independent given the cluster z".

Conditional Independence

- We say that A is conditionally independent of B given C if

$$p(a, b \mid c) = p(a \mid c)p(b \mid c),$$

  for all a, b, and $c \neq 0$.
- Equivalently, we have

$$p(a \mid b, c) = p(a \mid c).$$

- "If you know C, then *also* knowing B would tell you nothing about A"'.
  - In mixture of Bernoullis, given cluster there is no dependence between variables.

- We often write this as

$$A \perp B \mid C.$$

- Most models have some sort of conditional independence.
  - They were used to simplify calculations in the EM notes.
  - They determine whether message passing is efficient.

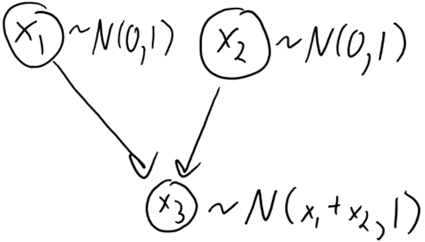
D-Separation: From Graphs to Conditional Independence

- All conditional independences implied by a DAG can be read from the graph.
- In particular: variables A and B are conditionally independent given C if:
  - "D-separation blocks all undirected paths in the graph from any variable in A to any variable in B."
- In the special case of product of independent models our graph is:

- Here there are no paths to block, which implies the variables are independent.
- Checking paths in a graph tends to be faster than tedious calculations.
  - We can start connecting properties of graphs to computational complexity.

D-Separation as Genetic Inheritance

- The rules of d-separation are intuitive in a simple model of gene inheritance:
  - Each person has single number, which we'll call a "gene".
  - If you have no parents, your gene is a random number.
  - If you have parents, your gene is a sum of your parents plus noise.
- For example, think of something like this:



- Graph corresponds to the factorization $p(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3 \mid x_1, x_2)$.
  - Are x_1 and x_2 independent here?

D-Separation as Genetic Inheritance

- Genes of people are independent if knowing one says nothing about the other.
- Your gene is dependent on your parents:
  - If I know you your parent's gene, I know something about yours.
- Your gene is independent of your (unrelated) friends:
  - If know you your friend's gene, it doesn't tell me anything about you.
- Genes of people can be conditionally independent given a third person:
  - Knowing your grandparent's gene tells you something about your gene.
  - But grandparent's gene isn't useful if you know parent's gene.

D-Separation Case 0 (No Paths and Direct Links)

Are genes in person x independent of the genes in person y?

- No path: x and y are not related (independent),

  We have $x \perp y$: there are no paths to be blocked.

- Direct link: x is the parent of y,

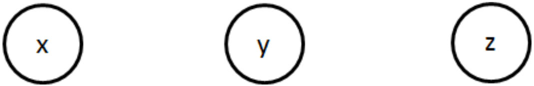
  

  We have $x \not\perp y$: knowing x tells you about y (direct paths aren't blockable).

D-Separation Case 0 (No Paths and Direct Links)

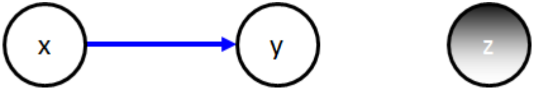
Neither case changes if we have a third independent person z:

- No path: If x and y are independent,



  We have $x \perp y$: adding z doesn't make a path.

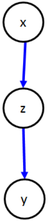
- Direct link: x is the parent of y,



  We have $x \not\perp y \mid z$: adding z doesn't block path.
  - We use **black or shaded** nodes to denote values we condition on (in this case z).

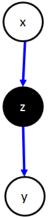
D-Separation Case 1: Chain

- Case 1: x is the grandparent of y.
  - If z is the mother we have:



We have $x \not\perp y$: knowing x would give information about y because of z

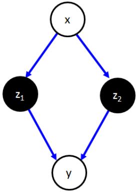
  - But if z is *observed*:



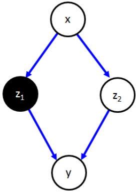
In this case $x \perp y \mid z$: knowing z "breaks" dependence between x and y.

D-Separation Case 1: Chain

- Consider weird case where parents z_1 and z_2 share parent x:
  - If z_1 and z_2 are observed we have:



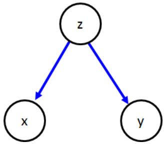
    We have $x \perp y \mid z_1, z_2$: knowing both parents breaks dependency.
  - But if only z_1 is *observed*:



    We have $x \not\perp y \mid z_1$: dependence still "flows" through z_2.

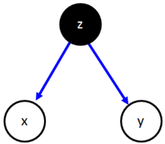
D-Separation Case 2: Common Parent

- Case 2: x and y are sibilings.
  - If z is a common unobserved parent:



We have $x \not\perp y$: knowing x would give information about y.

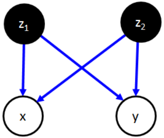
  - But if z is *observed*:



In this case $x \perp y \mid z$: knowing z "breaks" dependence between x and y.

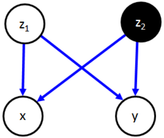
D-Separation Case 2: Common Parent

- Case 2: x and y are sibilings.
  - If z_1 and z_2 are common observed parents:



We have $x \perp y \mid z_1, z_2$: knowing z_1 and z_2 breaks dependence between x and y.

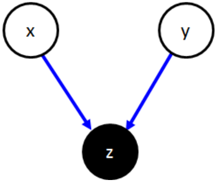
  - But if we only observe z_2:



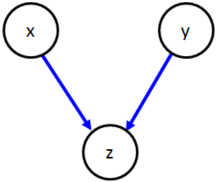
Then we have $x \not\perp y \mid z_2$: dependence still "flows" through z_1.

D-Separation Case 3: Common Child

- Case 3: x and y share a child z:
  - If we observe z then we have:



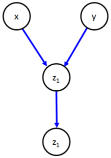
    We have $x \not\perp y \mid z$: if we know z, then knowing x gives us information about y.
  - But if z is not observed:



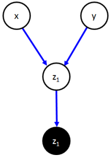
    We have $x \perp y$: if you don't observe z then x and y are independent.
- Different from Case 1 and Case 2: not observing the child blocks path.

D-Separation Case 3: Common Child

- Case 3: x and y share a child z_1:
  - If there exists an unobserved grandchild z_2:



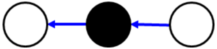
    We have $x \perp y$: the path is still blocked by not knowing z_1 or z_2.
  - But if z_2 is observed:



    We have $x \not\perp y \mid z_2$: grandchild creates dependence even with unobserved parent.
- Case 3 needs to consider descendants of child.

D-Separation Summary

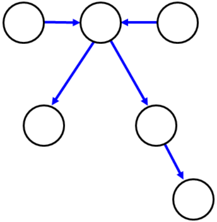
- We say that A and B are d-separated (conditionally independent) if *all paths* P from A to B are "blocked" because *at least one* of the following holds:
  1. P includes a "chain" with an observed middle node (e.g., Markov chain):

  2. P includes a "fork" with an observed parent node (e.g., mixture of Bernoulli):

  3. P includes a "v-structure" or "collider" (e.g., probabilistic PCA):

     where "child" and all its descendants are unobserved.

Alarm Example

- Case 1:
  - Earthquake $\not\perp$ Call.
  - Earthquake $\perp$ Call $\mid$ Alarm.
- Case 2:
  - Alarm $\not\perp$ Stuff Missing.
  - Alarm $\perp$ Stuff Missing $\mid$ Burglary.

Alarm Example

- Case 3:
  - Earthquake $\perp$ Burglary.
  - Earthquake $\not\perp$ Burglary $|$ Alarm.
    - "Explaining away": knowing one parent can make the other less likely.
- Multiple Cases:
  - Call $\not\perp$ Stuff Missing.
  - Earthquake $\perp$ Stuff Missing.
  - Earthquake $\not\perp$ Stuff Missing $|$ Call.

Discussion of D-Separation

- D-separation lets you say if conditional independence is implied by assumptions:

$$(A \text{ and } B \text{ are d-separated given } E) \Rightarrow A \perp B \mid E.$$

- However, there might be extra conditional independences in the distribution:
  - These would depend on specific choices of the $p(x_j \mid x_{\text{pa}(j)})$.
  - Or some *orderings* may reveal different independences.

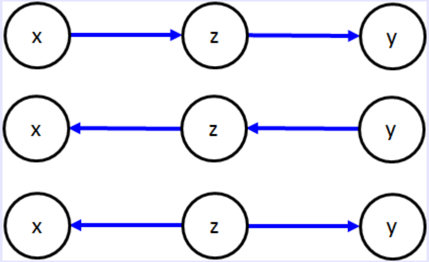
- Instead of restricting to $\{1, 2, \ldots, j-1\}$, consider general parent choices.
  - x_2 could be a parent of x_1.

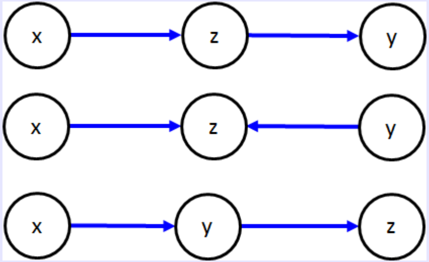
- As long the graph is acyclic, there exists a valid ordering (chain rule makes sense).

(all DAGs have a "topological order" of variables where parents are before children)

Non-Uniqueness of Graph and Equivalent Graphs

- Note that some graphs imply same conditional independences:
  - Equivalent graphs: same v-structures and other (undirected) edges are the same.
  - Examples of 3 *equivalent* graphs (left) and 3 non-equivalent graphs (right):





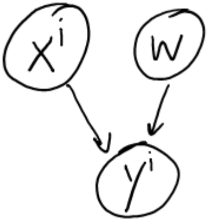
Outline

Discussion of D-Separation

- So the graph is not necessarily unique and is not the whole story.
- But, we can already do a lot with d-separation:
  - Implies every independence/conditional-independence we've used in 340/540.
- Here we start blurring distinction between data/parameters/hyper-parameters...

Tilde Notation as a DAG

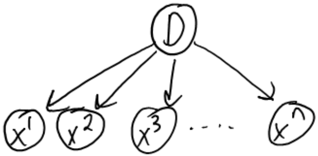
- When we write
$$y^i \sim \mathcal{N}(w^T x^i, 1),$$
this can be interpretd as a DAG model:



- "The variables on the right of $\sim$ are the parents of the variables on the left".
  - In this case, w only depends on X since we know y.
- Note that we're now including both data and parameters in the graph.
  - This allows us to see and reason about their relationships.

IID Assumption as a DAG

- On Day 2, our first independence assumption was the IID assumption:



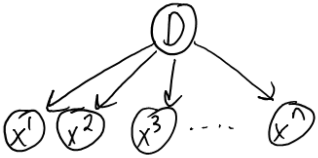
- Training/test examples come independently from data-generating process D.

- If we knew D, we wouldn't need to learn.
- But D is unobserved, so knowing about some x^i tells us about the others.

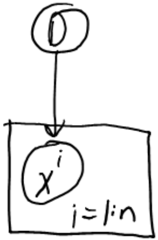
- We'll use this understanding later to relax the IID assumption.
  - Bonus: using this to ask "when does semi-supervised learning make sense?"

Plate Notation

- Graphical representation of the IID assumption:

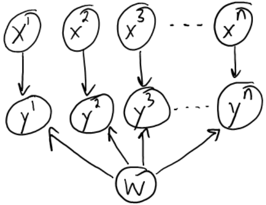


- It's common to represent repeated parts of graphs using plate notation:

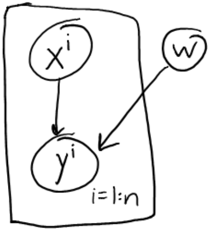


Tilde Notation as a DAG

- If the x^i are IID then we can represent regression as



or



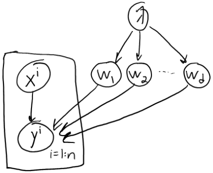
- From d-separation on this graph we have $p(y \mid X, w) = \prod_{i=1}^{n} p(y^i \mid x^i, w)$.
- We often omit the data-generating distribution D.
  - But if you want to learn then should remember that it's there.
- Note that plate reflects parameter tieing: that we use same w for all i.

Tilde Notation as a DAG

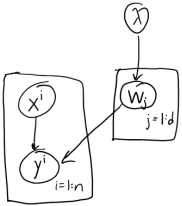
- When we do MAP estimation under the assumptions

$$y^i \sim \mathcal{N}(w^T x^i, 1), \quad w_j \sim \mathcal{N}(0, 1/\lambda),$$

we can interpret it as the DAG model:



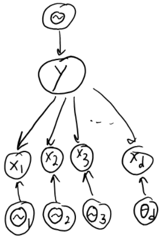
- Or introducing a second plate using:



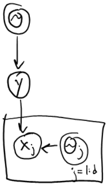
Other Models in DAG/Plate Notation

- For naive Bayes we have

$$y^i \sim \mathrm{Cat}(\theta), \quad x^i \mid y^i = c \sim Cat(\theta_c).$$



- Or in plate notation as



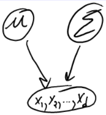
Summary

- Conditional independence of A and B given C:
  - Knowing B tells us nothing about A if we already know C.
- D-separation allows us to test conditional independences based on graph.
- Plate Notation lets us compactly draw graphs with repeated patterns.
  - There are fancier versions of plate notation called "probabilistic programming".
- Next time: trying to discover the graph structure from data.

Other Models in DAG/Plate Notation

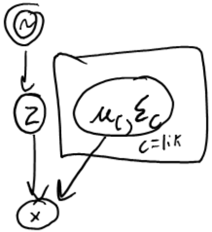
- In a full Gaussian model for a single x we have

$$x^i \sim \mathcal{N}(\mu, \Sigma).$$



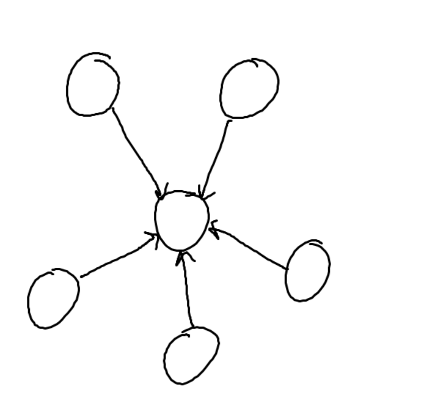
- For mixture of Gaussians we have

$$z^i \sim \text{Cat}(\theta), \quad x^i \mid z^i = c \sim \mathcal{N}(\mu_c, \Sigma_c).$$



Conditional Independence in Star Graphs

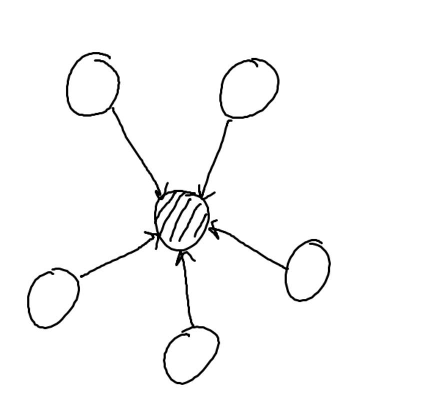
- Consider the following star graph:



- "5 aliens get together and make a baby alien".
  - Unconditionally, the 5 aliens are independent.

Conditional Independence in Star Graphs

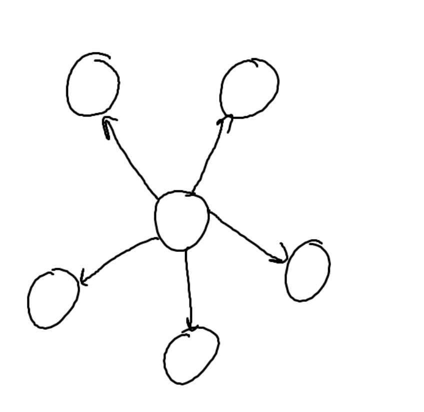
- Consider the following star graph:



- "5 aliens get together and make a baby alien".
  - Conditioned on the baby, the 5 aliens are dependent.

Conditional Independence in Star Graphs

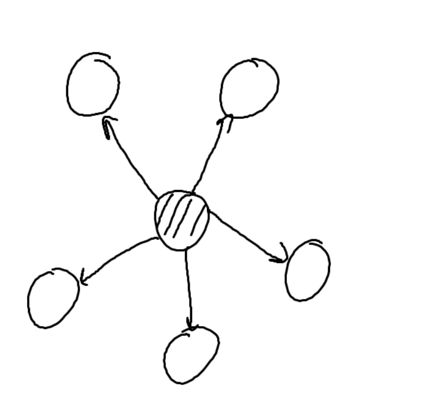
- Consider the following star graph:



- "An organism produces 5 clones".
  - Unconditionally, the 5 clones are dependent.

Conditional Independence in Star Graphs

- Consider the following star graph:



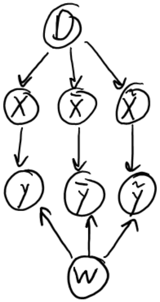
- "An organism produces 5 clones".
  - Conditioned on the original, the 5 clones are independent.

Does Semi-Supervised Learning Make Sense?

- Should unlabeled examples always help supervised learning?
  - No!
- Consider choosing unlabeled features $\bar{x}^i$ uniformly at random.
  - Unlabeled examples collected in this way will not help.
  - By construction, distribution of $\bar{x}^i$ says nothing about $\bar{y}^i$.
- Example where SSL is not possible:
  - Try to detect food allergy by trying random combinations of food:
    - The actual random process isn't important, as long as it isn't affected by labels.
    - You can sample an infinite number of $\bar{x}^i$ values, but they says nothing about labels.
- Example where SSL is possible:
  - Trying to classify images as "cat" vs. "dog.:
    - Unlabeled data would need to be images of cats or dogs (not random images).
    - Unlabeled data contains information about what images of cats and dogs look like.
    - For example, there could be clusters or manifolds in the unlabeled images.

Does Semi-Supervised Learning Make Sense?

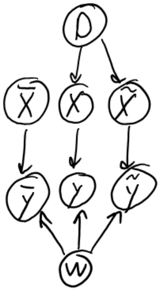
- Let's assume our semi-supervised learning model is represented by this DAG:



- Assume we observe $\{X, y, \bar{X}\}$ and are interested in test labels $\tilde{y}$:
  - There is a dependency between y and $\tilde{y}$ because of path through w.
    - Parameter w is tied between training and test distributions.
  - There is a dependency between X and $\tilde{y}$ because of path through w (given y).
    - But note that there is also a second path through D and $\tilde{X}$.
  - There is a dependency between $\bar{X}$ and $\tilde{y}$ because of path through D and $\tilde{X}$.
    - Unlabeled data helps because it tells us about data-generating distribution D.

Does Semi-Supervised Learning Make Sense?

- Now consider generating $\bar{X}$ independent of D:



- Assume we observe $\{X, y, \bar{X}\}$ and are interested in test labels $\tilde{y}$:
  - Knowing X and y are useful for the same reasons as before.
  - But knowing $\bar{X}$ is not useful:
    - Without knowing $\bar{y}$, $\bar{X}$ is d-separated from $\tilde{y}$ (no dependence).

Beware of the "Causal" DAG

- It can helpful to use the language of causality when reasoning about DAGs.
  - You'll find that they give the correct causal interpretation based on our intuition.
- However, keep in mind that the arrows are not necessarily causal.
  - "A causes B" has the same graph as "B causes A".
- There is work on causal DAGs which add semantics to deal with "interventions".
  - But these require extra assumptions: fitting a DAG to observational data doesn't imply anything about causality.