

CS 559 Machine Learning

Lecture 12: Graphical Models

Ping Wang

Department of Computer Science

Stevens Institute of Technology



Today's Lecture

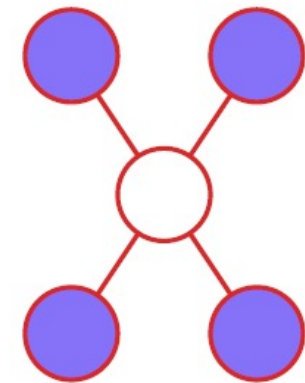
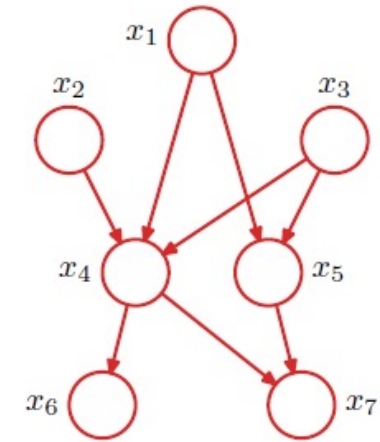
- Introduction to Graphical Models
- Naïve Bayes
- Bayesian Networks
- Other Graphical Models

Motivation of Probabilistic Graphical Models

- Probability theory are commonly represented as simple equations.
- Probabilistic graphical model:
 - A simple way to **visualize** the structure of a probabilistic model
 - Provides **insight** into the properties of the model, such as the conditional independence.
 - Complex computations can be expressed in terms of **graphical manipulations**

Definitions in Graphical Models

- A graph G contains:
 - **Nodes**, also known as vertices. Each node represents a random variable (or group of random variables).
 - **Edges**, also known as links between nodes. Express probabilistic relationships between variables.
- Edges may be directed or undirected (may also have associated weights)
 - Directed Graphs (all edges are directed)
 - Undirected Graphs (all edges are undirected)



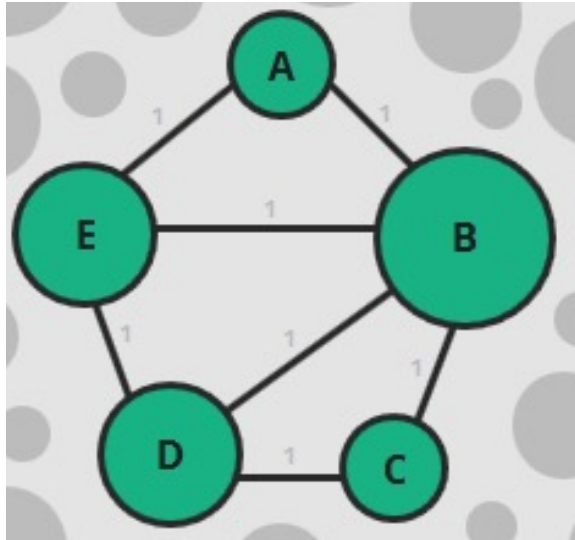
Types of Graphical Models

- **Directed** graphical models:
 - Useful for expressing causal relationships between random variables.
 - Bayesian Networks, Latent Dirichlet Allocation, etc
- **Undirected** graphical models:
 - Better suited to expressing soft constraints between random variables.
 - Markov network (Markov random field): a set of random variables having a Markov property described by an undirected graph.

More Definitions

- **Path**: The Path $A \rightarrow B$ from node A to node B is a sequence of nodes that connects A to B .
- **Cycle**: A cycle is a directed path that starts and returns to the same node.
- **Directed Acyclic Graph (DAG)**: A DAG is a graph G with directed edges (arrows on each link) between the nodes such that by following a path of nodes from one node to another along the direction of each edge, no path will revisit a node.
- **Parent and Children**: for a link going from node A to node B , then A is the parent of B .
- Graphs can be **represented** using: the edge list, the adjacency matrix.

Representation of Graphs



A	B	C	D	E
0	1	0	0	1
1	0	1	1	1
0	1	0	1	0
0	1	1	0	1
1	1	0	1	0

```
{ "Nodes": [  
  { "Name": "Alice", "Number": 1 },  
  { "Name": "Bob", "Number": 2 },  
  { "Name": "Charlie", "Number": 3 },  
  { "Name": "David", "Number": 4 },  
  { "Name": "Evan", "Number": 5 } ],  
  "Edges": [  
    { "source": "Alice", "target": "Bob", "weight": 2.0 },  
    { "source": "Alice", "target": "Evan", "weight": 1.0 },  
    { "source": "Bob", "target": "Charlie", "weight": 5.0 },  
    { "source": "Bob", "target": "David", "weight": 1.0 },  
    { "source": "Bob", "target": "Evan", "weight": 1.0 },  
    { "source": "Charlie", "target": "David", "weight": 2.0 },  
    { "source": "David", "target": "Evan", "weight": 3.0 } ] }
```

Two Important Rules

- **Chain rule or Product rule:** Let S_1, S_2, \dots, S_n be events, and $p(S_i) > 0$, then:
$$p(S_1, S_2, \dots, S_n) = p(S_1)p(S_2|S_1) \cdots p(S_n|S_{n-1}, \dots, S_1)$$

Let X, Y be two variables, then

$$p(X, Y) = p(X|Y)p(Y)$$

- **Sum rule:** let X, Y be two variables, then:

$$p(X) = \sum_Y p(X, Y)$$

Independence Between Two Variables

- Two random variables X and Y are **independent** if for any state x of X and any state y of Y , the joint probability distribution of $X = x$ and $Y = y$ is the product of marginal probability of $X = x$ and marginal probability of $Y = y$.
- This is given by:

$$p(x, y) = p(x)p(y)$$

- In this case, the conditional probability of each variable does not depend on the other variable.

$$\begin{aligned} p(x|y) &= p(x) \\ p(y|x) &= p(y) \end{aligned}$$

Graphical Models

- Graphical models are graph-based representations of various **decomposition** (or factorization into a product of factors) assumptions of the joint distributions of all random variables.
- These factorizations are typically equivalent to **independence** statement among variables in the distributions.
- Each factor depends only on **a subset of variables**.

Key Idea of Graphical Models

- Representation

- Represent the real problems as a collection of random variables with joint distribution
- Capture uncertainties in real problems
- Encode our domain knowledge/assumptions/constraints

- Learning

- Learn the distribution from the data: what model is “right” for my data?

- Inference

- Compute conditional distributions: how do I answer questions/queries according to the learnt model based on the given evidence?

Key Challenges of Graphical Models

- **Representation**: What is the joint probability distribution on multiple variables?
 - How to represent and describe the joint distribution?
 - Directed graphical models (Bayesian networks)
 - Undirected graphical models (Markov random fields)
- **Learning**: Learn the distribution from the data.
 - How many data samples do we need?
 - Maximum likelihood estimation? Any other estimation principles?
 - Where do we put domain knowledge in terms of plausible relationships between variables, and plausible values of the probabilities?
- **Inference**: If not all variables are observable, how to compute the conditional distribution of latent variables given evidence?

Two Important Operations of Probability

- Marginalization
- Conditioning

Marginalization

- Suppose X and Y are random variables with distribution $p(X, Y)$
 - X : Intelligence, $\text{Val}(X) = \{\text{"Very High"}, \text{"High"}\}$
 - Y : Grade, $\text{Val}(Y) = \{\text{"a"}, \text{"b"}\}$

- Joint distribution specified by:

	vh	h
a	0.7	0.15
b	0.1	0.05

- The marginal probability $p(Y = a) = 0.85$
- In general, suppose we have a joint distribution $p(X_1, \dots, X_n)$. Then,

$$p(X_i = x_i) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{i-1}} \sum_{x_{i+1}} \cdots \sum_{x_n} p(x_1, \dots, x_n)$$

Conditioning

- Suppose X and Y are random variables with distribution $p(X, Y)$
 - X : Intelligence, $\text{Val}(X) = \{\text{"Very High"}, \text{"High"}\}$
 - Y : Grade, $\text{Val}(Y) = \{\text{"a"}, \text{"b"}\}$
- The conditional probability can be computed as:

	vh	h
a	0.7	0.15
b	0.1	0.05

$$\begin{aligned} p(Y = a|X = vh) &= \frac{p(Y = a, X = vh)}{P(X = vh)} \\ &= \frac{p(Y=a, X=vh)}{p(Y=a, X=vh)+p(Y=b, X=vh)} \\ &= \frac{0.7}{0.7+0.1} = 0.875 \end{aligned}$$

Example: Medical Diagnosis

- Variable for each **symptom** (e.g. “fever”, “cough”, “fast breathing”, “shaking”, “nausea”, “vomiting”)
- Variable for each **disease** (e.g. “flu”, “pneumonia”, “common cold”, “bronchitis”, “tuberculosis”)
- **Diagnosis** is performed by inference in the model:

$$p(pneumonia = 1 | cough = 1, fever = 1, vomiting = 0)$$

- The famous model Quick Medical Reference for Decision-Theoretic (QMR-DT) covers 600 diseases and 4000 findings.

Representing the Distribution

- Intuitively, we could **represent** multivariate distributions with table of probabilities for each outcome.
- How many outcomes are there in QMR-DT? 2^{4600}
- **Estimation** of joint distribution would require a huge amount of data.
- **Inference** of conditional probabilities, e.g.
$$p(pneumonia = 1 | cough = 1, fever = 1, vomiting = 0)$$
would require summing over exponentially many variables' values.
- Moreover, this defeats the purpose of probabilistic modeling, which is to make predictions with previously unseen observations.

Directed Graphical Model for Joint Probability

- Based on the product rule, the joint probability of three variables a, b, c is in the following form:

$$\begin{aligned} p(a, b, c) &= p(c|a, b)p(a, b) \\ &= p(c|a, b)p(b|a)p(a) \end{aligned}$$

- A directed graphical model representing the joint probability distribution over a, b, c can be shown in the right figure.
- Different **ordering** has different factorization.

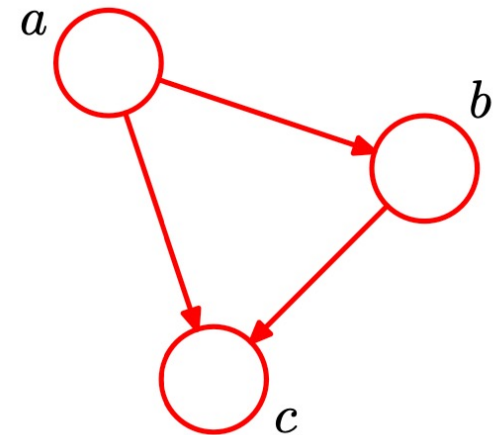


Figure: C. Bishop, PRML

Structure Through Independence

- If X_1, \dots, X_n are independent, then:

$$p(x_1, \dots, x_n) = p(x_1) \dots p(x_n)$$

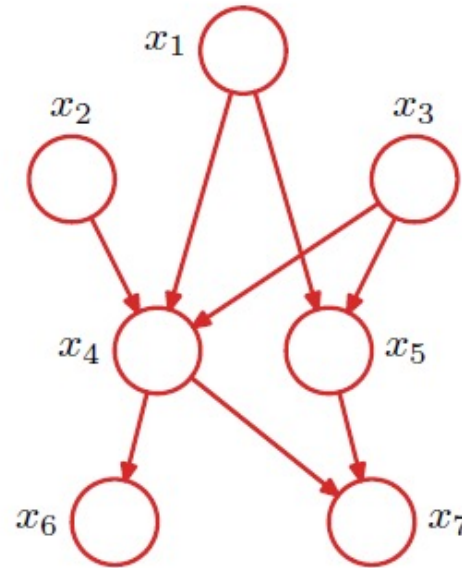
- 2^n entries can be described by just n numbers (if $|Val(X_i)| = 2$).
- However, this depends on a strong assumption that observing a variable X_i cannot influence our predictions of X_j .
- If X_1, \dots, X_n are conditionally independent given Y , denoted as $X_i \perp X_{-i} | Y$, then:

$$\begin{aligned} p(y, x_1, \dots, x_n) &= p(y)p(x_1|y) \prod_{i=2}^n p(x_i|x_1, \dots, x_{i-1}, y) \\ &= p(y)p(x_1|y) \prod_{i=2}^n p(x_i|y) \end{aligned}$$

- This model is simple, but powerful! Example: Naïve Bayes for classification

Bayesian Networks

- A Bayesian network is a directed acyclic graph (DAG) where nodes represent the random variables x_1, \dots, x_n and each x_i has a conditional probability of the node given its parents.
- The joint distribution is obtained by taking the product of the conditional probabilities:



- $$p(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)$$

Bayes Classifier

- Given the input data with different attributes and class labels.
- Consider each attribute and class label as random variables.
- Task: given a record with attributes (X_1, X_2, \dots, X_n)
 - Goal: predict the class label C
 - Specifically, we want to find the value of C that maximizes $P(C|X_1, X_2, \dots, X_n)$
- Can we estimate $P(C|X_1, X_2, \dots, X_n)$ directly from the input data?

Bayes Classifier

- Task: compute the posterior probability $P(C|X_1, X_2, \dots, X_n)$ for all values of C using the Bayes theorem

$$P(C|X_1, X_2, \dots, X_n) = \frac{P(X_1, X_2, \dots, X_n|C)P(C)}{P(X_1, X_2, \dots, X_n)}$$

- The denominator can be ignored as it is the same for all labels of the class.
- Choose the value of C that maximizes $P(C|X_1, X_2, \dots, X_n)$
- Equivalent to maximize $P(X_1, X_2, \dots, X_n|C)P(C)$
- How to estimate $P(X_1, X_2, \dots, X_n|C)$?
 - **Naïve Bayes**: assumes **independence** among attributes when the class is given.
 - **Bayesian Network**: assumes that there exist **conditional dependencies** among attributes.

Naïve Bayes

Naïve Bayes Classifier

- Naïve Bayes: assume independence among attributes when the class is given.

$$P(X_1, X_2, \dots, X_n | C_j) = \prod_{i=1}^n P(X_i | C_j)$$

- Can estimate $P(X_i | C_j)$ for all X_i and C_j
- New test data point is classified to C_j if $P(C_j) \prod_{i=1}^n P(X_i | C_j)$ is maximal.

Probability Estimation: Discrete Attributes

<i>Tid</i>	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Class: $P(C_j) = N_{C_j}/N$
- Attributes: $P(X_i = x_i | C_j) = \frac{|X_{ij}|}{N_{C_j}}$
 - N : total number of examples
 - N_{C_j} : total number of examples with label C_j
 - $|X_{ij}|$ is the number of examples having attribute $X_i = x_i$ and with label C_j
- Example:
 $P(No) = \frac{7}{10}; P(Yes) = \frac{3}{10}$
 $P(Status = Married | No) = \frac{4}{7}$
 $P(Refund = Yes | Yes) = 0$

Probability Estimation: Continuous Attributes

- **Discretize** the range into bins:
 - One ordinal attribute per bin
- **Two-way split:** $(X < v)$ or $(X > v)$
 - Choose only one of the two splits as new attribute
- **Probability density estimation:**
 - Assume an attribute follows a **normal distribution**
 - Use data to estimate parameters of distribution (e.g., mean and standard deviation)
 - Once probability distribution is known, we can use it to estimate the conditional probability $P(X_i|C_j)$.

Probability Estimation: Continuous Attributes

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Normal distribution: one for each (X_i, C_j) pair

$$P(X_i|C_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(X_i - \mu_{ij})^2}{2\sigma_{ij}^2}}$$

- Example: For (income, class=No), we can have mean=110; variance=2975

$$\begin{aligned} P(\text{Income} = 120|\text{No}) &= \frac{1}{\sqrt{2\pi \times 2975}} e^{-\frac{(120-110)^2}{2 \times 2975}} \\ &= 0.0072 \end{aligned}$$

Example of Naïve Bayes Classifier

$$P(\text{Refund}=\text{Yes} \mid \text{No})=3/7$$

$$P(\text{Refund}=\text{No} \mid \text{No})=4/7$$

$$P(\text{Refund}=\text{Yes} \mid \text{Yes})=0$$

$$P(\text{Refund}=\text{No} \mid \text{Yes})=1$$

$$P(\text{Marital Status}=\text{Single} \mid \text{No})=2/7$$

$$P(\text{Marital Status}=\text{Divorced} \mid \text{No})=1/7$$

$$P(\text{Marital Status}=\text{Married} \mid \text{No})=4/7$$

$$P(\text{Marital Status}=\text{Single} \mid \text{Yes})=2/3$$

$$P(\text{Marital Status}=\text{Divorced} \mid \text{Yes})=1/3$$

$$P(\text{Marital Status}=\text{Married} \mid \text{Yes})=0$$

For taxable income:

If class=No, mean=110, variance=2975

If class=Yes, mean=90, variance=25

- A test record:

$$X = (\text{Refund}=\text{No}, \text{Married}, \text{Income}=120\text{K})$$

$$P(X \mid \text{Class}=\text{No}) = P(\text{Refund}=\text{No} \mid \text{No}) \\ \times P(\text{Marital Status}=\text{Married} \mid \text{No}) \times P(\text{Income}=120\text{K} \mid \\ \text{Class}=\text{No})$$

$$= \frac{4}{7} \times \frac{4}{7} \times 0.0072 = 0.0024$$

$$P(X \mid \text{Class}=\text{Yes}) = P(\text{Refund}=\text{No} \mid \text{Yes}) \\ \times P(\text{Marital Status}=\text{Married} \mid \text{Yes}) \times P(\text{Income}=120\text{K} \mid \\ \text{Class}=\text{Yes})$$

$$= 1 \times 0 \times 1.2 \times 10^{-9} = 0$$

- Since $P(X \mid \text{No})P(\text{No}) > P(X \mid \text{Yes})P(\text{Yes})$,

Therefore $P(\text{No} \mid X) > P(\text{Yes} \mid X) \Rightarrow \text{Class} = \text{No}$

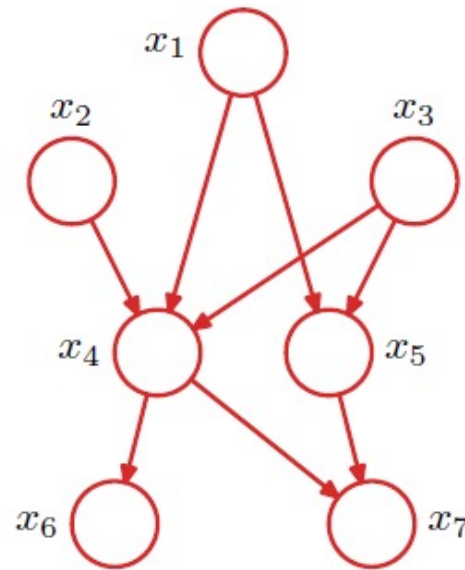
Summary of Naïve Bayes

- Robust to isolated noise points.
- Handle missing values by ignoring the instance during probability estimate calculations.
- Robust to irrelevant attributes.
- Independence assumption may not hold for some attributes.
 - Use other techniques such as Bayesian Networks (BN).

Bayesian Networks

Bayesian Networks (BN)

- A Bayesian network is a directed acyclic graph (DAG) where nodes represent the random variables x_1, \dots, x_n and each x_i has a conditional probability of the node given its parents.
- The joint distribution is obtained by taking the product of the conditional probabilities:



- $$p(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5)$$

Bayesian Networks

- Bayesian network consists of two parts:
 - A **directed network structure** in the form of a DAG, which can be represented as $G = (V, E)$.
 - A **set of local probability distributions**, one for each variable, conditional upon each value combination of its parents $p(x_i | par_G(x_k))$.
- For a graph with K nodes, the joint distribution is given by:

$$p(x) = \prod_{k=1}^K p(x_k | par_G(x_k))$$

where $par_G(x_k)$ denotes the set of parents of x_k and $x = \{x_1, \dots, x_K\}$

Alarm Example

- Sally's burglar Alarm(A) is sounding.
- Has she been Burgled (B), or was the alarm triggered by an Earthquake (E)?
- She turns the car Radio (R) on for news of earthquakes.
- Without loss of generality, we can write:

$$\begin{aligned} p(A, R, E, B) &= p(A|R, E, B)p(R, E, B) \\ &= p(A|R, E, B)p(R|E, B)p(E, B) \\ &= p(A|R, E, B)p(R|E, B)p(E|B)p(B) \end{aligned}$$

Alarm Example

$$p(A, R, E, B) = p(A|R, E, B)p(R|E, B)p(E|B)p(B)$$

Assumptions:

- The alarm is not directly influenced by any report on the radio

$$p(A|R, E, B) = p(A|E, B)$$

- The radio broadcast is not directly influenced by the burglar variable

$$p(R|E, B) = P(R|E)$$

- Burglaries don't directly “cause” earthquakes

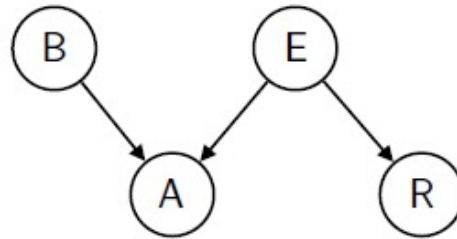
$$P(E|B) = P(E)$$

- Therefore, we have

$$p(A, R, E, B) = p(A|E, B)p(R|E)p(E)p(B)$$

Alarm Example: Probability Table

- DAG for $p(A, R, E, B) = p(A|E, B)p(R|E)p(E)p(B)$



- Probability table for $p(A|E, B)$

Alarm = 1	Burglar	Earthquake
0.9999	1	1
0.99	1	0
0.99	0	1
0.0001	0	0

- Probability table for $p(R|E)$

Radio = 1	Earthquake=1
1	1
0	0

- $p(B = 1) = 0.01$ and $p(E = 1) = 0.000001$

Alarm Example: Probability Table

- Initial evidence: the alarm is sounding

$$\begin{aligned} p(B = 1|A = 1) &= \frac{\sum_{E,R} p(B = 1, E, A = 1, R)}{\sum_{B,E,R} p(B, E, A = 1, R)} \\ &= \frac{\sum_{E,R} p(A = 1|B = 1, E)p(B=1)p(R|E)p(E)}{\sum_{B,E,R} p(A = 1|B, E)p(B)p(E)p(R|E)} \\ &\approx 0.99 \end{aligned}$$

Alarm Example: Inference

- **Additional evidence:** the radio broadcasts an earthquake warning
- **A similar calculation gives:** $p(B = 1|A = 1, R = 1) \approx 0.01$.
- Initially, because the alarm sounds, Sally thinks that she's been burgled. However, this probability drops dramatically when she hears that there has been an earthquake.
- The earthquake “explains away” to an extent the fact that the alarm is ringing.

Bayesian Networks for Classification

- Given a training dataset, the goal of Bayesian network is to correctly predict the label for C given a vector of n attributes. It can be formulated as follows:

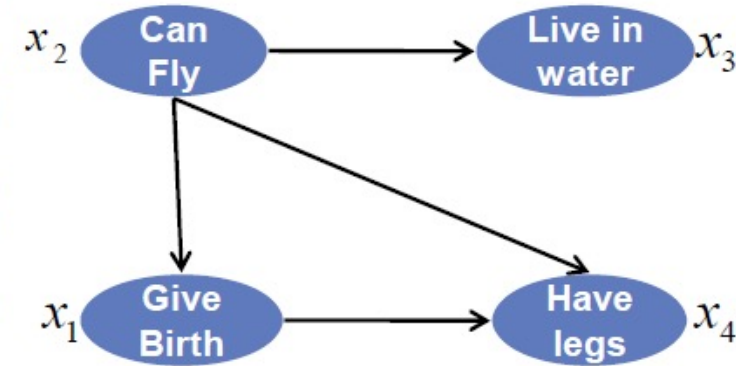
$$P(C|X_1, X_2, \dots, X_n) = \frac{P(X_1, X_2, \dots, X_n|C)P(C)}{P(X_1, X_2, \dots, X_n)}$$

- The denominator can be ignored as it is the same for all labels of the class.
- Equivalent to maximize $P(X_1, X_2, \dots, X_n|C)P(C)$
- We use Bayesian network to estimate $P(X_1, X_2, \dots, X_n|C)$.

Example of Bayesian Network

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?



$$P(x_1 = \text{yes}, x_2 = \text{no}, x_3 = \text{yes}, x_4 = \text{no} \mid M)P(M)$$

$$= \left(\frac{6}{7} \times \frac{5}{6} \times \frac{2}{6} \times \frac{2}{5}\right) \times \frac{7}{20} = \frac{3}{90}$$

$$P(x_1 = \text{yes}, x_2 = \text{no}, x_3 = \text{yes}, x_4 = \text{no} \mid N)P(N)$$

$$= \left(\frac{10}{13} \times \frac{1}{10} \times \frac{3}{10} \times \frac{1}{1}\right) \times \frac{13}{20} = \frac{3}{200}$$

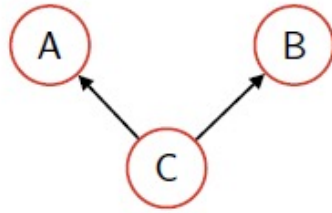
=> Mammals

$$P(C \mid x_1, x_2, x_3, x_4) = \frac{p(x_1, x_2, x_3, x_4 \mid C)P(C)}{P(x_1, x_2, x_3, x_4)}$$

$$= \frac{p(x_2 \mid C)p(x_1 \mid C, x_2)p(x_3 \mid C, x_2)p(x_4 \mid C, x_1, x_2)P(C)}{P(x_1, x_2, x_3, x_4)}$$

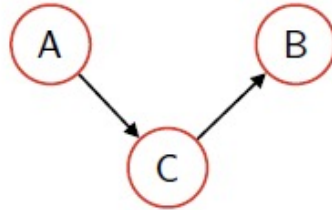
Marginal Independence in Bayesian Network

Tail-to-tail



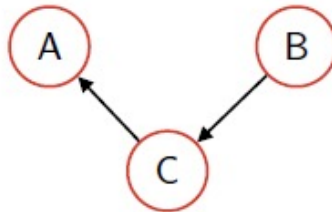
A,B are marginally dependent: $A \not\perp B$

Head-to-tail



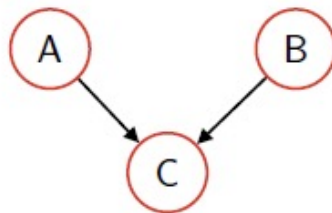
A,B are marginally dependent: $A \not\perp B$

Head-to-tail



A,B are marginally dependent: $A \not\perp B$

Head-to-head

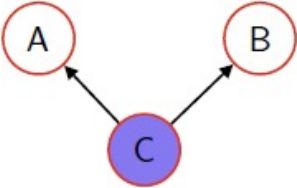
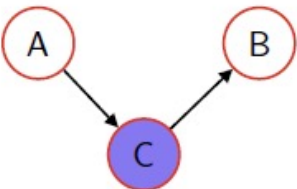
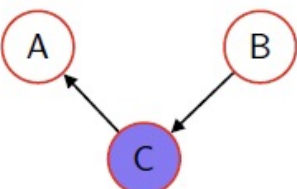
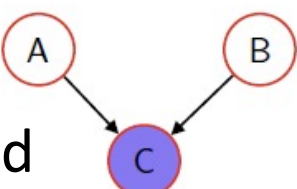


A,B are marginally independent: $A \perp B$

$$p(A, B, C) = p(A)p(B)p(C|A, B) \rightarrow p(A, B) = p(A)p(B)$$

Marginalizing both
sides over C

Conditional Independence in Bayesian Network

		$(A \perp B C)$
		A,B are conditionally independent given C
		$p(A, B C) = \frac{p(A, B, C)}{p(C)} = \frac{p(A C)p(B C)p(C)}{p(C)} = p(A C)p(B C)$
Tail-to-tail		
		A,B are conditionally independent given C
		$p(A, B C) = \frac{p(A)p(C A)p(B C)}{p(C)} = \frac{p(A, C)p(B C)}{p(C)} = p(A C)p(B C)$
Head-to-tail		
		A,B are conditionally independent given C
		$p(A, B C) = \frac{p(A)p(A C)p(C B)p(B)}{p(C)} = \frac{p(A C)p(B, C)}{p(C)} = p(A C)p(B C)$
Head-to-head		
		A,B are conditionally dependent given C
		$p(A, B C) \propto p(C A, B)p(A)p(B)$
Head-to-head		

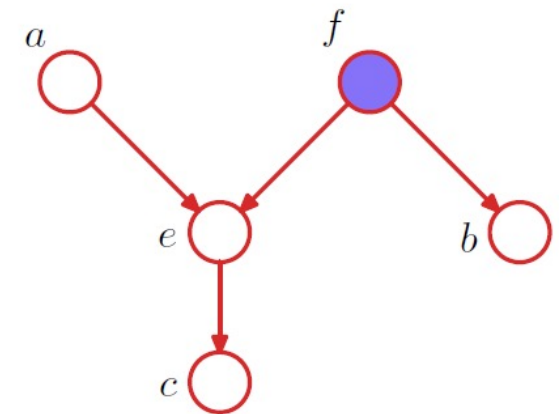
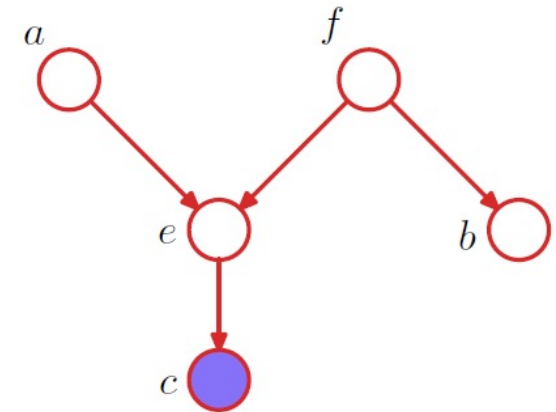
Bayesian Network structure implies conditional independencies!

D-separation (Directed Separated) in Bayesian Networks

- Algorithm to calculate whether $A \perp B | C$ by looking at graph separation, where A, B, and C are arbitrary nonintersecting sets of nodes.
- Look at all possible paths from A to B, any such path is said to be **blocked** (A and B are conditional independent given C) if it includes a node such that either
 1. The arrows on the path meet either **head-to-tail** or **tail-to-tail** at the node, and the node is in the set C.
 2. The arrows meet **head-to-head** at the node, and neither the node, nor any of its descendants, is in the set C.
- If all paths are blocked, then A is said to be d-separated from B by C.
- D-separation reduces **statistical independencies** (hard) to **connectivity in graphs** (easy).
- It allows us to quickly prune the Bayesian network, finding just the relevant variables for answering a query.

D-separation Examples

- Graph (a): path from a to b is **not blocked** by:
 - node f : tail-to-tail node in the path and is not observed.
 - node e : although it is a head-to-head node, it has a descendant c in the conditioning set.
 - Therefore, conditional independence does not hold.
- Graph (b): path from a to b is **blocked** by:
 - Node f : tail-to-tail node in the path and is observed.
 - Node e : head-to-head node and neither it nor its descendant are in the conditioning set.
 - Conditional independence is satisfied by any distribution that factorizes according to this graph.



Naïve Bayes & Bayesian Network

- Naive Bayes:
 - Encodes **incorrect independence** assumptions that, given the class label, the attributes are independent of each other.
 - But in the real-world problems, the attributes are mostly correlated and the case as in Naive Bayes **rarely happens**.
- Bayesian networks:
 - Bayesian networks capture all the **correlations** in the random variables in the form of a graph.
 - But learning such a Bayesian network is very **complex**, because there may be many random variables in a network, and each random variable may take many values.

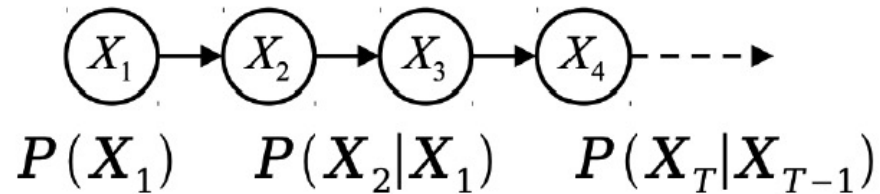
Frequently Used Graphical Models

- Markov Chain
- Hidden Markov models
- Mixture of Gaussians
- Latent Dirichlet allocation (LDA)

Markov Chain

- A Markov chain is a chain-structured BN about the probabilities of sequences of random variables (states).
 - Each node is identically distributed (stationarity)
 - Value of X at a given time is called the state
- **Strong assumption:**
 - If we want to predict the future in the sequence, all that matters is the current state.
 - The states before the current state have no impact on the future except via the current state.
- Example:
 - To predict tomorrow's weather, you could examine today's weather, but you weren't allowed to look at yesterday's weather.

Markov Chain

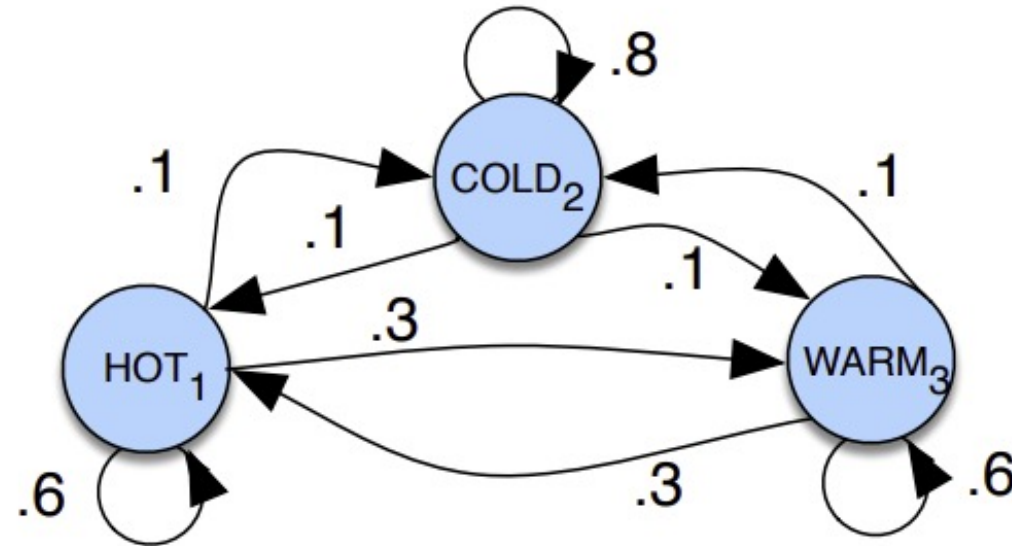


- Formally, consider a sequence of state variables, X_1, \dots, X_T .
- Markov model embodies the Markov assumption on the probabilities of the sequence. When predicting the future, the past doesn't matter, only the present.

$$p(X_t = a | X_1 \dots X_{t-1}) = p(X_t = a | X_{t-1})$$

- Parameters:
 - Transition probabilities or dynamics, specify how the state evolves over time
 - Also, initial probabilities: a start distribution
- Note that the chain is just a (growing) BN
 - We can always use generic BN to perform reasoning on it

Example of Markov Chain



- A Markov chain for weather, showing states and transitions.
- A start distribution is required.
- For example, $[0.1, 0.7, 0.2]$ indicates that a probability 0.7 of starting from state 2 (cold), and a probability 0.1 of starting in state 1 (hot).

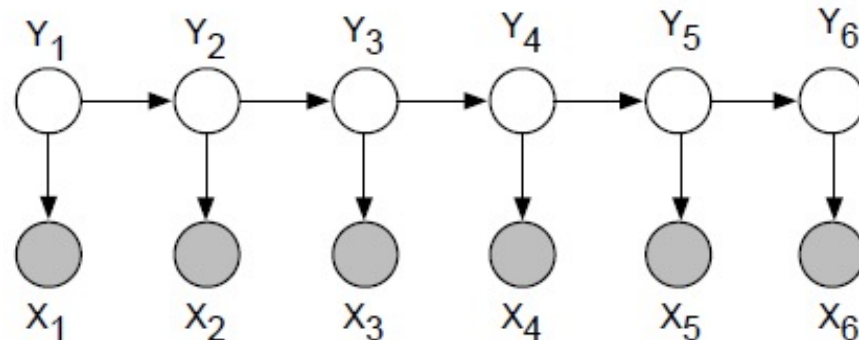
Limitations of Markov Chain

- Markov chain is useful for computing a probability for a sequence of events.
- However, for many real problems, the events are hidden and we don't observe directly.
- Need observations to update your beliefs.
- Example:
 - we don't normally observe part-of-speech tags in a text. Rather, we see words, and must infer the tags from the word sequence.
- Hidden Markov model (HMM):
 - Observed events (like words that we see in the input)
 - Hidden events (like part-of-speech tags) that we think of as causal factors in our probabilistic model

Hidden Markov Models

- Hidden Markov Models
 - Frequently used for speech recognition and part-of-speech tagging
 - Underlying Markov chain over a set of states S
 - Joint distribution factors as:

$$p(y, x) = p(y_1)p(x_1|y_1) \prod_{t=2}^T p(y_t|y_{t-1})p(x_t|y_t)$$



Hidden Markov models

- Joint distribution factors as:

$$p(y, x) = p(y_1)p(x_1|y_1) \prod_{t=2}^T p(y_t|y_{t-1})p(x_t|y_t)$$

- $p(y_1)$ is the distribution for the starting state
- $p(y_t|y_{t-1})$ is the **transition probability** between any two states
- $p(x_t|y_t)$ is the **emission probability**
- What are the conditional independencies here? e.g. $Y_1 \perp \{Y_3, \dots, Y_6\} | Y_2$
- Markov assumptions:
 - The current state is conditionally independent of all the past states given the states in the previous time step.
 - The current evidence is only dependent on the current state.
- More details about learning HMM: <https://web.stanford.edu/~jurafsky/slp3/A.pdf>

Mixture of Gaussians

- The D-dimensional multivariate normal distribution, $\mathcal{N}(\mu, \Sigma)$, has density:

$$p(x) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu)\right)$$

- Suppose we have K Gaussian densities and a distribution π indicates the mixing coefficients.
- Mixture of Gaussians distribution is given by

$$p(x) = \sum_{k=1}^K \pi_k \mathcal{N}(x | \mu_k, \Sigma_k)$$

- Each Gaussian density $\mathcal{N}(x | \mu_k, \Sigma_k)$ is called a **component of the mixture** and has its own mean μ_k and covariance Σ_k
- **Mixing coefficients**: $0 \leq \pi_k \leq 1, \sum_{k=1}^K \pi_k = 1$

Mixture of Gaussians

Introducing **latent variables** about the cluster assignments:

- The marginal distribution over z is specified in terms of the mixing coefficients:

$$p(z_k = 1) = \pi_k$$

- The conditional distribution of x given a particular value for z is a gaussian:

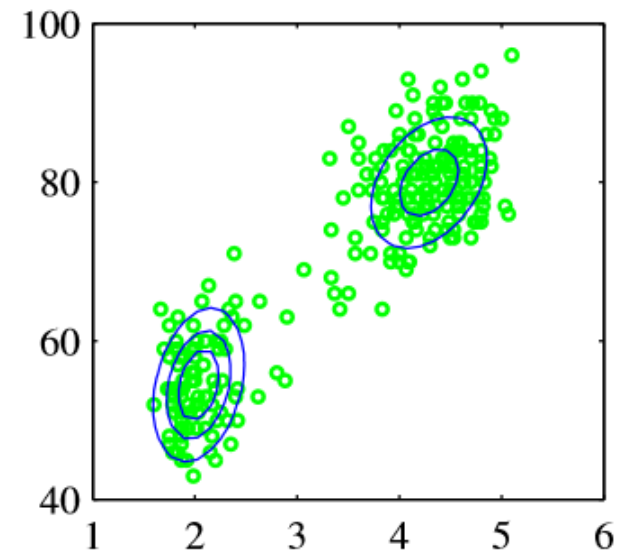
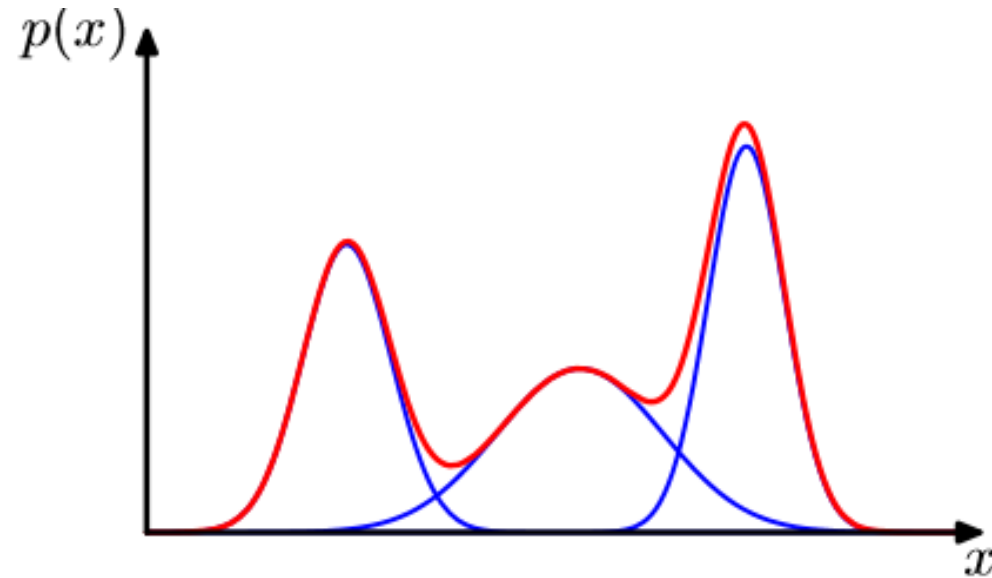
$$p(x|z_k = 1) = \mathcal{N}(x|\mu_k, \sigma_k)$$

- The marginal distribution of x is then obtained by summing the joint distribution over all possible values of z :

$$p(x) = \sum_z p(z)p(x|z) = \sum_{k=1}^K \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$$

Mixture of Gaussians

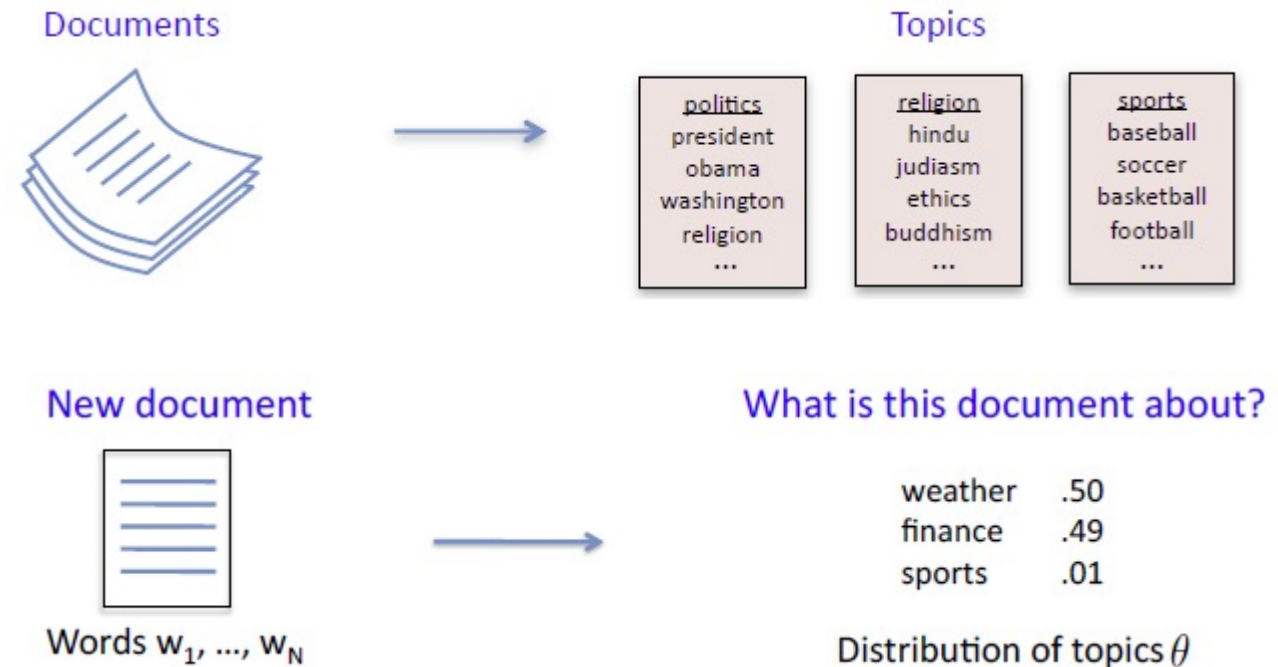
- The marginal distribution over x looks like:



Latent Dirichlet Allocation (LDA)

Paper: Latent Dirichlet Allocation

- **Topic models** are powerful tools for exploring data sets and for making inferences about the content of documents.
- Many applications in information retrieval, document summarization, and classification.
- **LDA** is one of the simplest and most widely used topic models.



Latent Dirichlet Allocation (LDA)

Assumptions:

- Each **topic** is corresponding to a **word distribution** of the words in the corresponding vocabulary.
- Each **document** is corresponding to a **topic distribution** of the topics.
- For each word in a document, we need to first generate a topic based on the topic distribution, and then generate the word based on the word distribution of the topic.

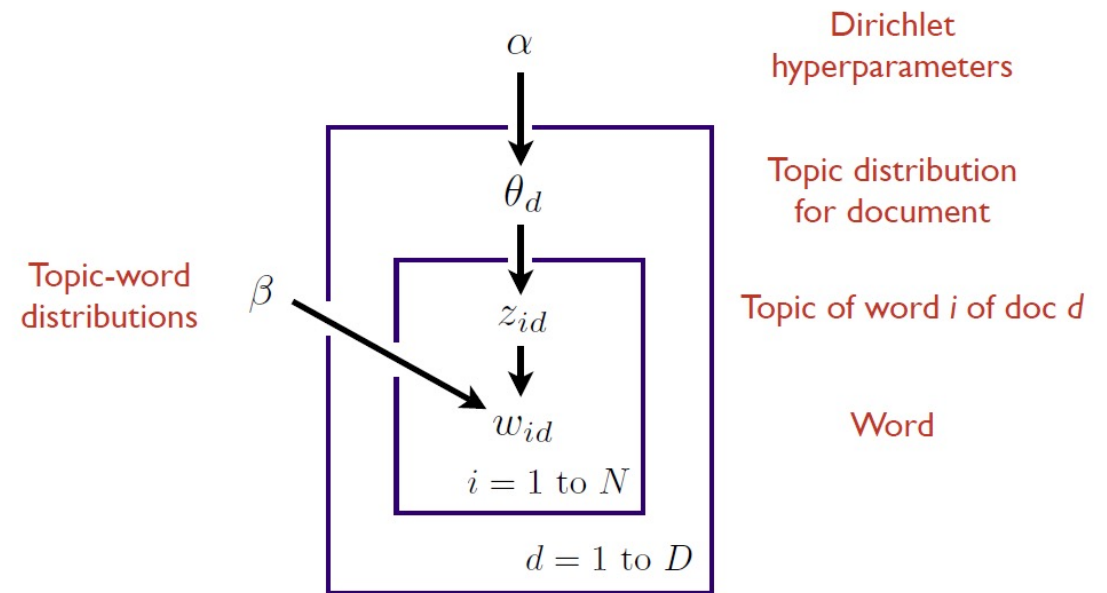


Plate notation representing the LDA model. Variables within a plate are replicated in a conditionally independent manner.

Generative Model for a Document in LDA

1. Sample the document's **topic distribution** (aka topic vector).

$$\theta \sim \text{Dirichlet}(\alpha_{1:T})$$

where $\{\alpha_t\}_{t=1}^T$ are fixed hyper-parameters. Thus, θ is a distribution over T topics

with mean $\theta_t = \frac{\alpha_t}{\sum_{t'} \alpha_{t'}}$

2. For $i = 1 \rightarrow N$, sample the **topic** z_i of the i 'th word

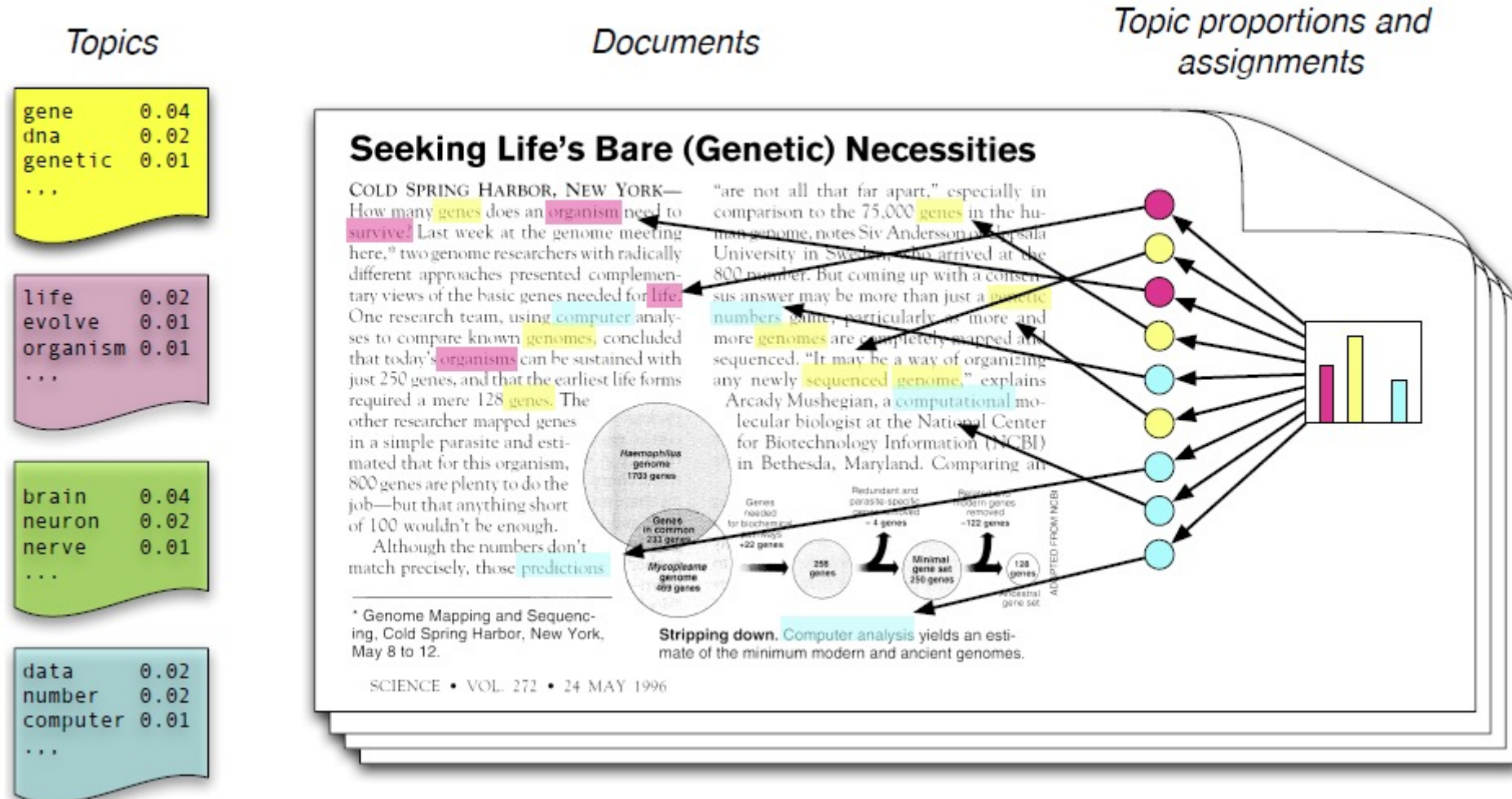
$$z_i | \theta \sim \text{Multi}(\theta)$$

3. ... and then sample the actual **word** w_i from the z_i 'th topic

$$w_i | z_i, \dots \sim \text{Multi}(\beta_{z_i})$$

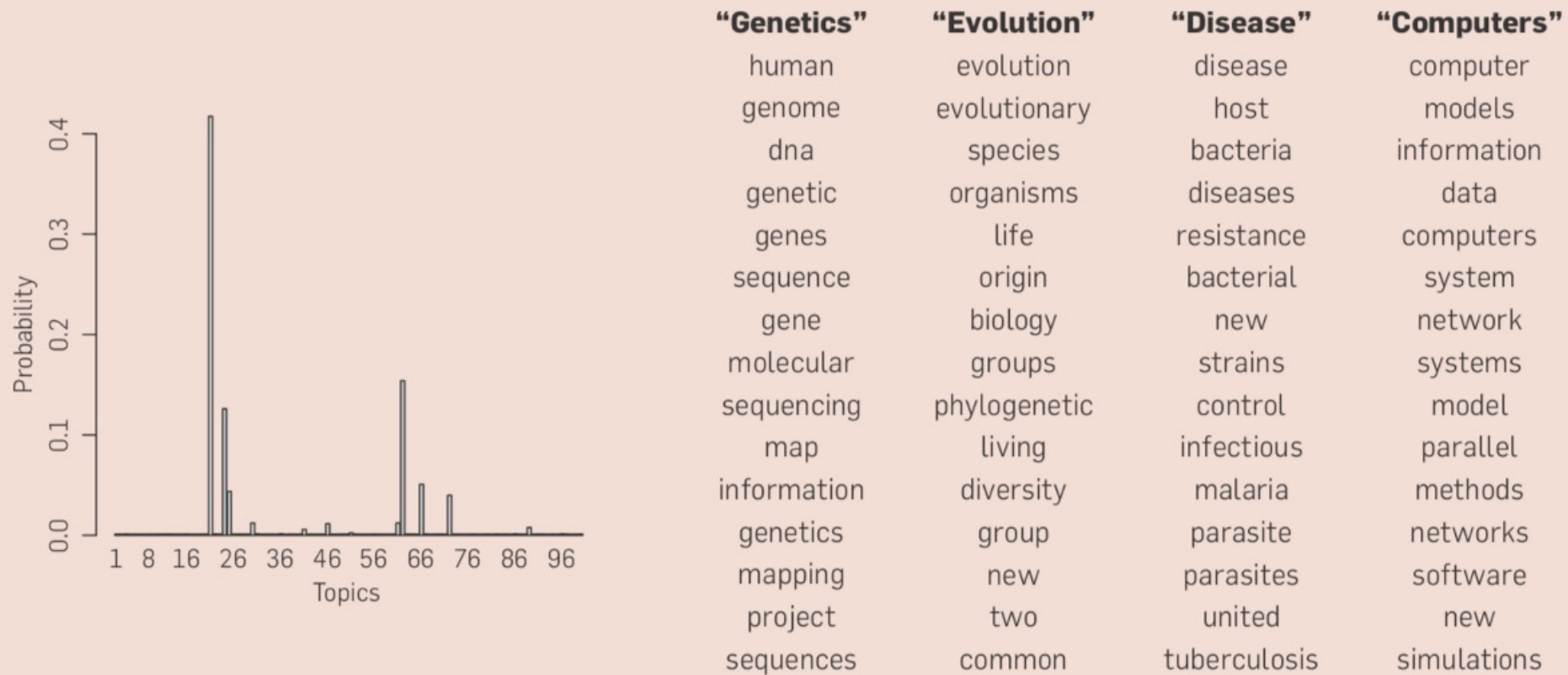
where $\{\beta_t\}_{t=1}^T$ are the topics (a fixed collection of distributions on words)

Example of Using LDA



Example of Using LDA

Figure 2. Real inference with LDA. We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.



The number of topics is fixed before generating documents.

<https://www.objectorientedsubject.net/2017/12/experiments-on-topic-modeling-lda/>

Summary of Bayesian Networks

- Bayesian networks given by (G, P) where P is specified as a set of local conditional probability distributions associated with G 's nodes
- One interpretation of a BN is as a **generative** model, where variables are sampled in topological order
- Local and global independence properties identifiable via d-separation criteria
- Computing the probability of any assignment is obtained by multiplying CPDs
 - Bayes' rule is used to compute conditional probabilities
 - Marginalization or inference is often computationally difficult
- Examples: hidden Markov models, latent Dirichlet allocation
- Limitations: not every distribution has a perfect map as a DAG.

Summary of Today's Lecture

- Introduction to Graphical Models
- Naïve Bayes
- Bayesian Networks
- Other Graphical Models