

# ECE493 Notes

June 15, 2019

## 1 Review of Probability Theory

- See [Stanford CS228 - Probability Review](#).

## 2 Representation

- **Problem:** How do we express a probability distribution  $p(x_1, x_2, \dots, x_n)$  that models some real-world phenomenon?
  - Naive Complexity:  $O(d^n)$
- **Solution:** Representation with Probabilistic Graphical Models + Verifying Independence Assumptions

### 2.1 Bayesian Networks (Directed Probabilistic Graphical Model)

### 2.2 Definition - What is a Bayesian network?

- A **Bayesian network** is a directed graph  $G$  with the following:
  - **Nodes:** A random variable  $x_i$ .
  - **Edges:** A conditional probability distribution (CPD)  $p(x_i \mid x_{A_i})$  per node, specifying the probability of  $x_i$  conditioned on its parent's values.

### 2.3 Representation - How does a Bayesian network express a probability distribution?

1. Let  $p$  be a probability distribution.
2. A naive representation of  $p$  can be derived using the chain rule:

$$p(x_1, x_2, \dots, x_n) = p(x_1)p(x_2 \mid x_1) \cdots p(x_n \mid x_{n-1}, \dots, x_2, x_1)$$

3. A Bayesian network representation of  $p$  compacts the naive representation by having each factor in the right hand side depend only on a small number of **ancestor variables**  $x_{A_i}$ :

$$p(x_i \mid x_{i-1}, \dots, x_2, x_1) = p(x_i \mid x_{A_i})$$

- e.g., Approximate  $p(x_5 \mid x_4, x_3, x_2, x_1)$  with  $p(x_5 \mid x_{A_5})$  where  $x_{A_5} = \{x_4, x_3\}$ .

## 2.4 Space Complexity - How compact is a Bayesian network?

- Consider each of the factors  $p(x_i \mid x_{A_i})$  as a **probability table**:
  - *Rows*: Values of  $x_i$
  - *Columns*: Values of  $x_{A_i}$
  - *Cells*: Values of  $p(x_i \mid x_{A_i})$
- If each discrete random variable takes  $d$  possible values and has at most  $k$  ancestors, then each probability table has at most  $O(d^{k+1})$  entries.
- **Naive Representation Space Complexity**:  $O(d^n)$
- **Bayesian Networks Representation Space Complexity**:  $O(n \cdot d^{k+1})$

$$\approx \text{Bayesian Networks Representation} \leq \text{Naive Representation}$$

## 2.5 Independence Assumptions - Why are the independence assumptions of a Bayesian network important to identify?

- A Bayesian network expresses a probability distribution  $p$  via products of smaller, local conditional probability distributions (one for each variable).
- These smaller, local conditional probability distributions introduces assumptions into the model of  $p$  that certain variables are independent.
- **Important Note**: Which independence assumptions are we exactly making by using a Bayesian network?
  - *Correctness*: Are these independence assumptions correct?
  - *Efficiency*: Do these independence assumptions efficiently compact the representation?

## 2.6 3-Variable Independencies in Directed Graphs - How do you identify independent variables in a 3-variable Bayesian network?

- Let  $x \perp y$  indicate that variables  $x$  and  $y$  are independent.
- Let  $G$  be a Bayesian network with three nodes:  $A$ ,  $B$ , and  $C$ .

### 2.6.1 Common Parent

- If  $G$  is of the form  $A \leftarrow B \rightarrow C$ ,
  - If  $B$  is observed, then  $A \perp C \mid B$
  - If  $B$  is unobserved, then  $A \not\perp C$
- **Intuition**:  $B$  contains all the information that determines the outcomes of  $A$  and  $C$ ; once it is observed, there is nothing else that affects  $A$ 's and  $C$ 's outcomes.

### 2.6.2 Cascade

- If  $G$  equals  $A \rightarrow B \rightarrow C$ ,
  - If  $B$  is observed, then  $A \perp C \mid B$
  - If  $B$  is unobserved, then  $A \not\perp C$
- **Intuition**:  $B$  contains all the information that determines the outcomes of  $C$ ; once it is observed, there is nothing else that affects  $C$ 's outcomes.

### 2.6.3 V-Structure

- If  $G$  is  $A \rightarrow C \leftarrow B$ , then knowing  $C$  couples  $A$  and  $B$ .
  - If  $C$  is unobserved, then  $A \perp B$
  - If  $C$  is observed, then  $A \not\perp B \mid C$

## 2.7 $n$ -Variable Independencies in Directed Graphs - How do you identify independent variables in a $n$ -variable Bayesian network?

- Let  $I(p)$  be the set of all independencies that hold for a probability distribution  $p$ .
- Let  $I(G) = \{(X \perp Y \mid Z) : X, Y \text{ are } d\text{-sep given } Z\}$  be a set of variables that are  $d$ -separated in  $G$ .
- If the probability distribution  $p$  factorizes over  $G$ , then  $I(G) \subseteq I(p)$  and  $G$  is an  $I$ -map (**independence map**) for  $p$ .
- **Important Note 1:** Thus, variables that are  $d$ -separated in  $G$  are independent in  $p$ .
- **Important Note 2:** However, a probability distribution  $q$  can factorize over  $G$ , yet have independencies that are not captured in  $G$ .
- **Important Caveat:** A Bayesian network cannot perfectly represent all probability distributions.

### 2.7.1 $d$ -separation (a.k.a. Directed Separation)

- $Q$  and  $W$  are  **$d$ -separated** when variables  $O$  are observed if they are **NOT CONNECTED** by an active path.

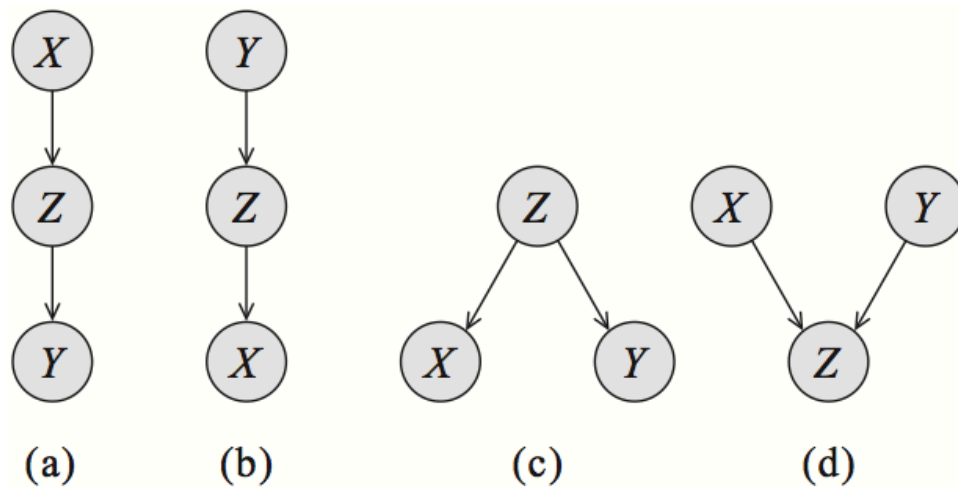
### 2.7.2 Active Path

- An undirected path in the Bayesian Network structure  $G$  is called **active** given observed variables  $O$  if for **EVERY CONSECUTIVE TRIPLE** of variables  $X, Y, Z$  on the path, one of the following holds:
  - **Evidential Trail:**  $X \leftarrow Y \leftarrow Z$ , and  $Y$  is unobserved  $Y \notin O$
  - **Causal Trail:**  $X \rightarrow Y \rightarrow Z$ , and  $Y$  is unobserved  $Y \notin O$
  - **Common Cause:**  $X \leftarrow Y \rightarrow Z$ , and  $Y$  is unobserved  $Y \notin O$
  - **Common Effect:**  $X \rightarrow Y \leftarrow Z$ , and  $Y$  or any of its descendants are observed

## 2.8 Equivalence - When are two Bayesian networks $I$ -equivalent?

- $G_1$  and  $G_2$  are  **$I$ -equivalent**...
  - If they encode the same dependencies:  $I(G_1) = I(G_2)$ .
  - If they have the same skeleton and the same v-structures.
  - If the  $d$ -separation between variables is the same.

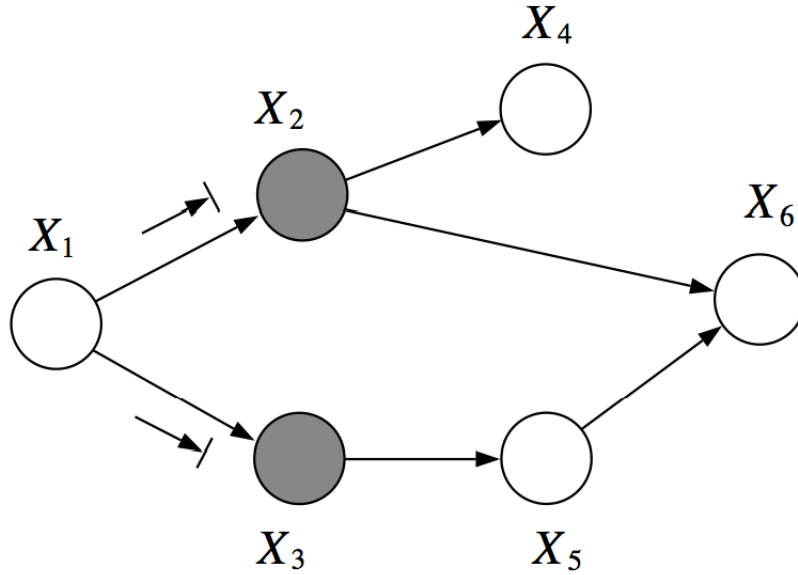
### 2.8.1 Skeleton



Skeleton

- A **skeleton** is an undirected graph obtained by dropping the directionality of the arrows.
  - (a) is Cascade
  - (b) is Cascade
  - (c) is Common Parent
  - (d) is V-Structure
  - (a), (b), (c), and (d) have the same skeleton.

## 2.9 Example Problem 1 - $d$ -separation



Problem 1 -  $d$ -separation

### 2.9.1 Question

- Are  $X_1$  and  $X_6$   $d$ -separated given  $\{X_2, X_3\}$ ?

### 2.9.2 Solution

1. **Path:**  $X_1 \rightarrow X_2 \rightarrow X_6$

1. *Consecutive Triple:*  $X_1 \rightarrow X_2 \rightarrow X_6$

- Although  $X_2$  is observed, the *common effect* does not hold.

2. As not all the consecutive triples hold, this path is not *active*.

2. **Path:**  $X_1 \rightarrow X_3 \rightarrow X_5 \rightarrow X_6$

1. *Consecutive Triple:*  $X_1 \rightarrow X_3 \rightarrow X_5$

- Although  $X_3$  is observed, the *common effect* does not hold.

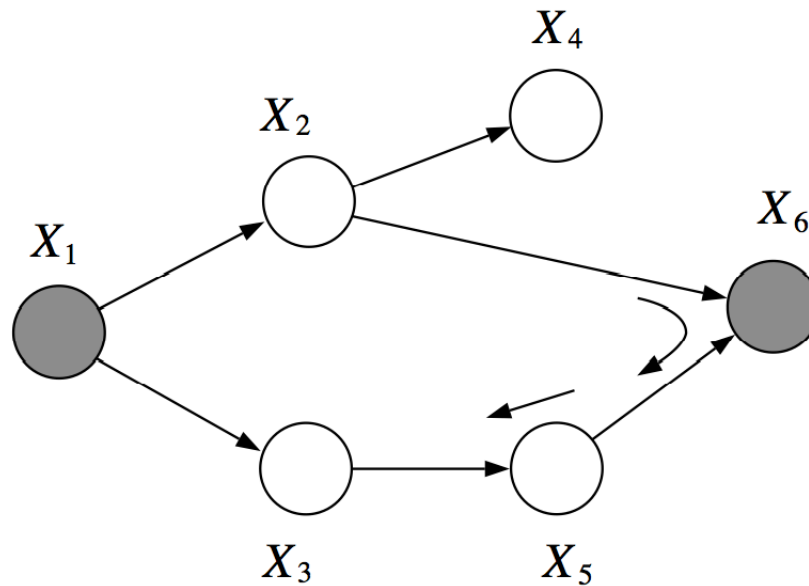
2. *Consecutive Triple:*  $X_3 \rightarrow X_5 \rightarrow X_6$

- As  $X_5$  is unobserved, the *causal trail* does hold.

3. As not all the consecutive triples hold, this path is not *active*.

3. As there are no active paths between  $X_1$  and  $X_6$ , they are  $d$ -separated given  $\{X_2, X_3\}$ .

## 2.10 Example Problem 2 - $d$ -separation



Problem 2 -  $d$ -separation

### 2.10.1 Question

- Are  $X_2$  and  $X_3$   $d$ -separated given  $\{X_1, X_6\}$ ?

### 2.10.2 Solution

1. **Path:**  $X_2 \leftarrow X_1 \rightarrow X_3$ 
  1. *Consecutive Triple:*  $X_2 \leftarrow X_1 \rightarrow X_3$ 
    - Although  $X_1$  is observed, the *common effect* does not hold.
  2. As not all the consecutive triples hold, this path is not *active*.
2. **Path:**  $X_2 \rightarrow X_6 \leftarrow X_5 \leftarrow X_3$ 
  1. *Consecutive Triple:*  $X_2 \rightarrow X_6 \leftarrow X_5$ 
    - As  $X_6$  is observed, the *common effect* does hold.
  2. *Consecutive Triple:*  $X_6 \leftarrow X_5 \leftarrow X_3$ 
    - As  $X_5$  is unobserved, the *causal trail* does hold.
  3. As all the consecutive triples hold, this path is *active*.
3. As there exists an active path between  $X_2$  and  $X_3$ , they are not  $d$ -separated given  $\{X_1, X_6\}$ .

## 2.11 Markov Random Fields (Undirected Probabilistic Graphical Model)

### 2.12 Definition - What is a Markov random field?

- A **Markov random field** is an undirected graph  $G$  with the following:

- *Nodes*: A random variable  $x_i$ .
- *Fully Connected Subgraphs*: An optional factor  $\phi_c(x_c)$  per clique, specifying the level of coupling (**potentials**) between all the dependent variables within the clique.
- **Important Note**: >...SPECIFYING THE LEVEL OF COUPLING BETWEEN ALL THE DEPENDENT VARIABLES WITHIN THE CLIQUE...

## 2.13 Representation - How does a Markov random field express a probability distribution?

1. Let  $p$  be a probability distribution.
2. A Markov random field representation of  $p$  is the following:

$$p(x_1, x_2, \dots, x_n) = \frac{1}{Z} \prod_{c \in C} \phi_c(x_c)$$

- Where  $C$  is the set of cliques of  $G$ .
- Where  $\phi_c$  is a **factor** (nonnegative function) over the variables in a clique.
- Where  $Z$  is a **normalizing constant** that ensures that  $p$  sums to one.

$$Z = \sum_{x_1, x_2, \dots, x_n} \prod_{c \in C} \phi_c(x_c)$$

## 2.14 Space Complexity - How compact is a Markov random field?

### 2.14.1 Factor Product

- Let  $A$ ,  $B$ , and  $C$  be three disjoint sets of variables.
- Let  $\phi_1(A, B)$  and  $\phi_2(B, C)$  be two factors.
- Let  $\phi_3(A, B, C)$  be the **factor product**.

$$\phi_3(A, B, C) = \phi_1(A, B) \cdot \phi_2(B, C)$$

- Where the two factors are multiplied for common values of  $B$ .

### 2.14.2 Binary Factor Tables

- Each of the optional factors  $\phi_c(x_c)$  can be expressed as a product of **binary factor tables**  $\phi(X, Y)$ :
  - *Rows*: Values of  $X$
  - *Columns*: Values of  $Y$
  - *Cells*: Values of  $\phi(X, Y)$
- If each variable takes  $d$  values, each binary factor table has at most  $O(d^2)$  entries.
- **Markov Random Fields Representation Space Complexity**:  $O(E \cdot d^2)$ 
  - Where  $E$  is the number of edges in a Markov random field.

$$\approx \text{Markov Random Field Representation} \leq \text{Naive Representation}$$

## 2.15 Markov Random Fields vs. Bayesian Networks - What are the advantages and disadvantages of Markov random fields?

### 2.15.1 Advantages

- *Applicable for Variable Dependencies Without Natural Directionality*
- *Succinctly Express Dependencies Not Easily Expressible in Bayesian Networks*

### 2.15.2 Disadvantages

- *Cannot Express Dependencies Easily Expressible in Bayesian Networks*
  - *e.g., V-Structures*
- *Computing Normalization Constant  $Z$  Is NP-Hard*
- *Generally Require Approximation Techniques*
- *Difficult to Interpret*
- *Easier to Construct Bayesian Networks*

## 2.16 Moralization - What is moralization?



Moralization

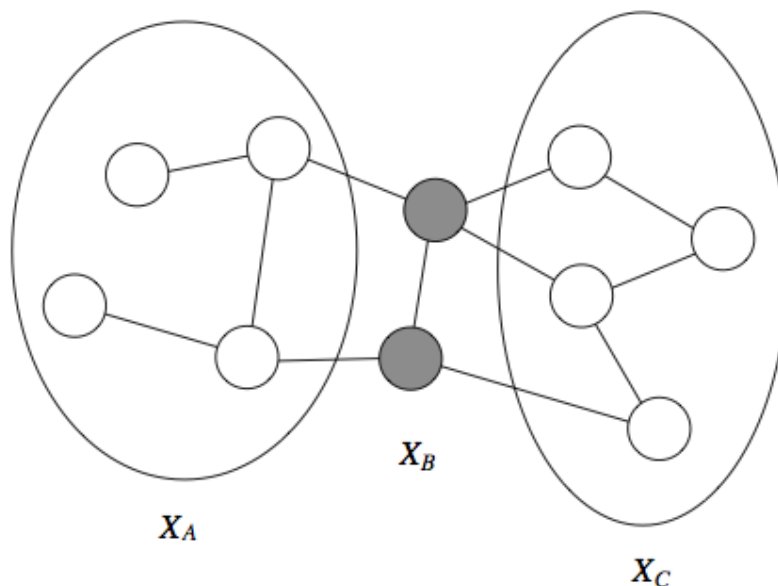
- Bayesian networks are a special case of Markov random fields with factors corresponding to conditional probability distributions and a normalizing constant of one.
- **Moralization:** Bayesian Network  $\rightarrow$  Markov Random Field
  1. Add side edges to all parents of a given node.
  2. Remove the directionality of all the edges.

## 2.17 $n$ -Variable Independencies in Undirected Graphs - How do you identify independent variables in a $n$ -variable Markov random field?

1. If variables  $X$  and  $Y$  are connected by a path of unobserved variables, then  $X$  and  $Y$  are dependent.
2. If variable  $X$ 's neighbors are all observed, then  $X$  is independent of all the other variables.
3. If a set of observed variables forms a cut-set between two halves of the graph, then variables in one half are independent from ones in the other.



### 2.17.1 Cut-Set Variable Independencies



Cut-Set Variable Independencies

### 2.17.2 Markov Blanket

- The **Markov blanket**  $U$  of a variable  $X$  is the minimal set of nodes such that  $X$  is independent from the rest of the graph if  $U$  is observed.

$$X \perp (\mathcal{X} - \{X\} - U) \mid U$$

- In an undirected graph, the Markov blanket is a node's neighborhood.

## 2.18 Conditional Random Fields - What are conditional random fields?

### 2.18.1 Definition

- A **conditional random field** is a Markov random field over variables  $\mathcal{X} \cup \mathcal{Y}$  which specifies a conditional distribution:

$$P(y \mid x) = \frac{1}{Z(x)} \prod_{c \in C} \phi_c(x_c, y_c)$$

$$Z(x) = \sum_{y \in \mathcal{Y}} \prod_{c \in C} \phi_c(x_c, y_c)$$

- Where  $x \in \mathcal{X}$  and  $y \in \mathcal{Y}$  are **VECTOR-VALUED** variables.
- Where  $Z(x)$  is the partition function.

- Important Note 1:** A conditional random field results in an instantiation of a new Markov random field for each input  $x$ .
- Important Note 2:** A conditional random field is useful for structured prediction in which the output labels are predicted considering the neighboring input samples.
  - See [Stanford CS228 - Markov Random Fields: Conditional Random Fields \(OCR Example\)](#).

### 2.18.2 Features

- Assume the factors  $\phi_c(x_c, y_c)$  are of the following form:

$$\phi_c(x_c, y_c) = \exp(w_c^T f_c(x_c, y_c))$$

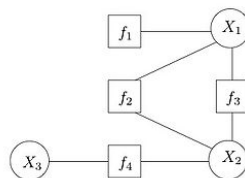
- Where  $f_c(x_c, y_c)$  can be an arbitrary set of features describing the compatibility between  $x_c$  and  $y_c$ .
- Where  $w_c^T$  is the transposed weight matrix.
- Accordingly,  $f_c(x_c, y_c)$  allows arbitrarily complex features.
  - e.g.,  $f(x, y_i)$  are features that depend on the entirety of input samples  $x$ .
  - e.g.,  $f(y_i, y_{i+1})$  are features that depend on successive pairs of output labels  $y$ .

### 2.19 Conditional Random Fields vs. Markov Random Fields - Why is a conditional random field a special case of Markov random fields?

- If we were to model  $p(x, y)$  using a Markov random field, then we need to fit two probability distributions to the data:  $p(y | x)$  and  $p(x)$ .
  - Remember Baye's Rule:  $p(x, y) = p(y | x) \cdot p(x)$
- However, if all we are interested in is predicting  $y$  given  $x$ , then modeling  $p(x)$  is expensive and unnecessary.

Prediction  $\implies$  CRF > MRF

### 2.20 Factor Graphs - What is a factor graph? Why does a factor graph exist?



Factor Graph

- A **factor graph** is a bipartite graph where one group is the variables in the distribution being modeled, and the other group is the factors defined on these variables.
  - *Edges Between Factors and Variables*
- **Side Note:** A **bipartite graph** is a graph whose vertices are divided into two disjoint and independent sets.
  - *Set 1: Variables*
  - *Set 2: Factors*
- **Important Note:** Use a factor graph to identify what variables a factor depends on when computing probability distributions.

### 3 Inference

- **Problem:** *Given a probabilistic model, how do we obtain answers to relevant questions about the world?*

- **Marginal Inference:** *What is the probability of a given variable in our model after we sum everything else out?*

$$p(y = 1) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_n} p(y = 1, x_1, x_2, \dots, x_n)$$

- \* e.g., What is the overall probability that an email is spam?
- \* *Perspective:* We desire to infer the general probability of some real-world phenomenon being observed.
  - i.e., You care more about spam as a whole than specific instances of spam.

- **Maximum A Posteriori:** *What is the most likely assignment of variables?*

$$\max_{x_1, \dots, x_n} p(y = 1, x_1, x_2, \dots, x_n)$$

- \* e.g., What is the set of words such that an email has the maximum probability of being spam?
- \* *Perspective:* We desire to infer the set of conditions that maximizes the probability of some real-world phenomenon being observed.
  - i.e., You care more about identifying indicators of spam than detecting spam.

- *Naive Complexity:* NP-Hard (**DIFFICULT PROBLEM**)

- **Solution:** *Exact Inference Algorithms & Approximate Inference Algorithms*

#### 3.1 Variable Elimination (Exact Inference Algorithm)

#### 3.2 Motivation - Why does the variable elimination algorithm exist?

- Let  $x_i$  be a discrete random variable that takes  $k$  possible values.
- **Problem:** *Marginal Inference*

$$p(y = 1) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_n} p(y = 1, x_1, x_2, \dots, x_n)$$

- **Naive Solution's Time Complexity** (*Exponential*):  $O(k^n)$ 
  - See [Rule of Product in Combinatorics](#)
- **Variable Elimination Solution's Time Complexity** (*Non-Exponential*):  $O(n \cdot k^{M+1})$ 
  - See Below.

$\therefore$  Variable Elimination Solution  $\ll$  Naive Solution

#### 3.3 Factors - How should a probabilistic graphical model express a probability distribution?

- **Assumption:** *Probabilistic Graphical Models = Product of Factors*

$$p(x_1, \dots, x_n) = \prod_{c \in C} \phi_c(x_c)$$

- **Representation:** A factor can be represented as a *multi-dimensional table* with a cell for each assignment of  $x_c$ .
- *Bayesian Networks:*  $\phi$  is Conditional Probability Distribution
- *Markov Random Fields:*  $\phi$  is Potentials

### 3.4 Factor Product - What is the product operation?

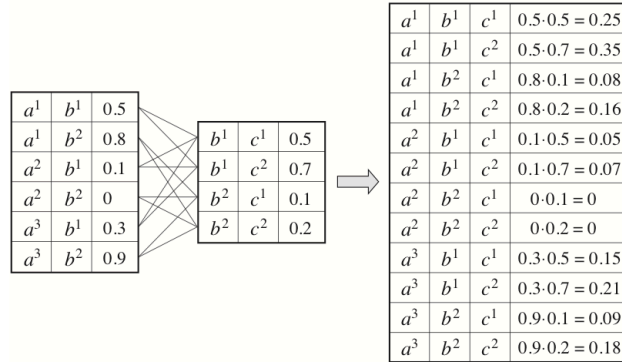


Figure 4.3 An example of factor product

#### Example of Factor Product

- Let  $A$ ,  $B$ , and  $C$  be three disjoint sets of variables.
- Let  $\phi_1(A, B)$  and  $\phi_2(B, C)$  be two factors.
- Let  $\phi_3(A, B, C)$  be the **factor product**.

$$\phi_3(A, B, C) = \phi_1(A, B) \cdot \phi_2(B, C)$$

- Where the two factors are multiplied for common values of  $B$ .

### 3.5 Factor Marginalization - What is the marginalization operation?

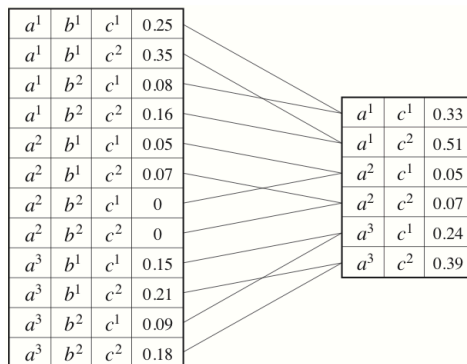


Figure 9.7 Example of factor marginalization: summing out  $B$ .

#### Example of Factor Marginalization

- Let  $A$ , and  $B$  be two disjoint sets of variables.

- Let  $\phi(A, B)$  be a factor.
- Let  $\tau(A)$  be the **factor marginalization** of  $B$  in  $\phi$ .

$$\tau(A) = \sum_B \phi(A, B)$$

- **Important Note:**  $\tau$  does not need necessarily correspond to a probability distribution.

### 3.6 Ordering - What is an ordering?

- An **ordering**  $O$  is the sequence of variables by which they will be eliminated.
- Although any ordering can be used, different orderings may dramatically alter the running time of the variable elimination algorithm.
- **Important Note:** *Finding Best Ordering = NP-Hard*

### 3.7 Algorithm - How does the variable elimination algorithm work?

- For each variable  $X_i$  (ordered according to  $O$ ),
  1. Multiply all factors  $\Phi_i$  containing  $X_i$ .
  2. Marginalize out  $X_i$  to obtain a new factor  $\tau$ .
  3. Replace the factors  $\Phi_i$  with  $\tau$ .

### 3.8 Time Complexity - What is the time complexity of variable elimination?

- **Time Complexity:**  $O(n \cdot k^{M+1})$ 
  - Where  $n$  is the number of variables.
  - Where  $M$  is the maximum number of dimensions of any factor  $\tau$  formed during the elimination process.

### 3.9 Ordering Heuristics - How should you choose an ordering for variable elimination?

- **Minimum Neighbors:** Choose a variable with the fewest dependent variables.
- **Minimum Weight:** Choose variables to minimize the product of the cardinalities of its dependent variables.
- **Minimum Fill:** Choose vertices to minimize the size of the factor that will be added to the graph.

### 3.10 Evidence - How do you perform marginal inference given some evidence using variable elimination?

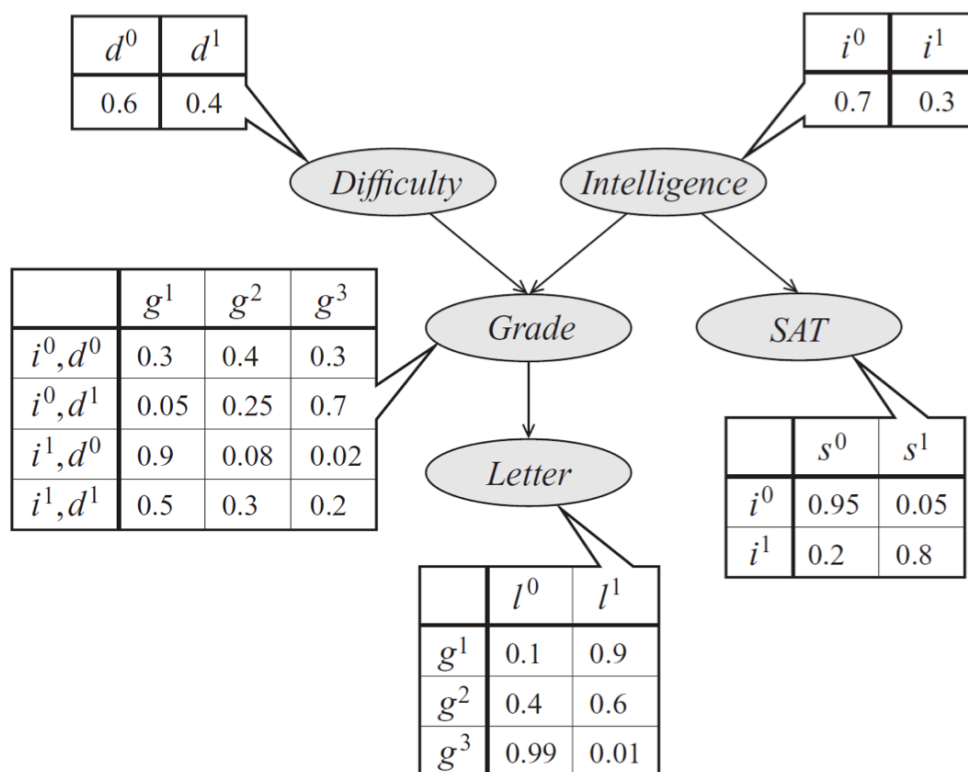
- Given a probability distribution  $P(X, Y, E)$  with unobserved variables  $X$ , query variables  $Y$ , and observed evidence variables  $E$ ,  $P(Y \mid E = e)$  can be calculated using variable elimination.

$$P(Y \mid E = e) = \frac{P(Y, E = e)}{P(E = e)}$$

### 3.10.1 Variable Elimination with Evidence

1. Set every factor  $\phi(X', Y', E')$  with values specified by  $E = e$ .
2. Compute  $P(Y, E = e)$  by performing variable elimination over  $X$ .
3. Compute  $P(E = e)$  by performing variable elimination over  $Y$ .

### 3.11 Example Problem 1 - Variable Elimination



Problem 1 - Variable Elimination

- A Bayesian network that models a student's grade on an exam:
  - $g$  is a ternary variable of the student's grade.
  - $d$  is a binary variable of the exam's difficulty.
  - $i$  is a binary variable of the student's intelligence.
  - $l$  is a binary variable of the quality of a reference letter from the professor who taught the course.
  - $s$  is a binary variable of the student's SAT score.

$$p(l, g, i, d, s) = p(l \mid g) \cdot p(s \mid i) \cdot p(i) \cdot p(g \mid i, d) \cdot p(d)$$

#### 3.11.1 Question (Marginal Inference)

- What is the probability distribution of the quality of a reference letter from the professor who taught the course?

$$p(l) = \sum_g \sum_i \sum_d \sum_s p(l, g, i, d, s)$$

### 3.11.2 Solution (Variable Elimination)

1. Order the variables according to the topological sort of the Bayesian network.

$$d, i, s, g$$

2. Eliminate  $d$  with a new factor  $\tau_1$ :

$$\begin{aligned}\tau_1(g, i) &= \sum_d p(g \mid i, d) \cdot p(d) \\ p(l, g, i, s) &= p(l \mid g) \cdot p(s \mid i) \cdot p(i) \cdot \tau_1(g, i)\end{aligned}$$

3. Eliminate  $i$  with a new factor  $\tau_2$ :

$$\begin{aligned}\tau_2(g, s) &= \sum_i p(s \mid i) \cdot p(i) \cdot \tau_1(g, i) \\ p(l, g, s) &= p(l \mid g) \cdot \tau_2(g, s)\end{aligned}$$

4. Eliminate  $s$  with a new factor  $\tau_3$ :

$$\begin{aligned}\tau_3(g) &= \sum_s \tau_2(g, s) \\ p(l, g) &= p(l \mid g) \cdot \tau_3(g)\end{aligned}$$

5. Eliminate  $g$  with a new factor  $\tau_4$ :

$$\begin{aligned}\tau_4(l) &= \sum_g p(l \mid g) \cdot \tau_3(g) \\ p(l) &= \tau_4(l)\end{aligned}$$

6. Expanding  $\tau_i$ :

$$p(l) = \sum_g p(l \mid g) \cdot \sum_s \sum_i p(s \mid i) \cdot p(i) \cdot \sum_d p(g \mid i, d) \cdot p(d)$$

### 3.11.3 Time Complexity

- **Naive Solution:**  $O(k^4)$
- **Variable Elimination Solution:**  $O(4 \cdot k^3)$ 
  - Step 2. takes  $O(k^3)$  steps as the factor product  $p(g \mid i, d) \cdot p(d)$  has a 3-dimensional table representation, and the factor marginalization of  $d$  can execute concurrently with the factor product.
  - Step 3. takes  $O(k^3)$  steps as the factor product  $p(s \mid i) \cdot p(i) \cdot \tau_1(g, i)$  has a 3-dimensional table representation, and the factor marginalization of  $i$  can execute concurrently with the factor product.
  - Step 4. takes  $O(k)$  steps for the factor marginalization of  $s$ .
  - Step 5. takes  $O(k^2)$  steps as the factor product  $p(l \mid g) \cdot \tau_3(g)$  has a 2-dimensional table representation, and the factor marginalization of  $g$  can execute concurrently with the factor product.
  - As  $O(k^3)$  is the largest step, with 4 steps, the time complexity is at most  $O(4 \cdot k^3)$ .
  - **Thus, with  $n = 4$  and  $M = 2$ , the time complexity is at most  $O(n \cdot k^{M+1}) = O(4 \cdot k^3)$ .**

### 3.12 MAP Inference

### 3.13 Overview - What is MAP inference?

- See Section ??.
- Given a probabilistic graphical model  $p(x_1, \dots, x_n) = \prod_{c \in C} \phi_c(x_c)$ , MAP inference corresponds to the following optimization problem:

$$\max_x \log p(x) = \max_x \sum_{c \in C} \theta_c(x_c) - \log Z$$

– Where  $\theta_c(x_c) = \log \phi_c(x_c)$ .

#### 3.13.1 Derivation - Why is the MAP inference optimization problem expressed the way it is?

1. All probabilistic graphical models (as BNs and CRFs are special cases of MRFs) have the following representation:

$$p(x) = \frac{1}{Z} \prod_{c \in C} \phi_c(x_c)$$

- Where  $Z$  is NP-Hard.
2. MAP inference desires to infer the set of conditions that maximizes the probability of some real-world phenomenon being observed.

$$\max_x p(x) = \max_x \frac{1}{Z} \prod_{c \in C} \phi_c(x_c)$$

3. As  $Z$  is expensive to calculate, maximize  $\log p(x)$  instead of  $p(x)$ .

$$\max_x \log p(x) = \max_x \log \left[ \frac{1}{Z} \prod_{c \in C} \phi_c(x_c) \right]$$

4. Simplify using logarithmic identities.

- $\log(x \times y) = \log(x) + \log(y)$
- $\log(x \div y) = \log(x) - \log(y)$

$$\max_x \log p(x) = \max_x \left[ \sum_c \log \phi_c(x_c) - \log Z \right]$$

5. Simplify using maximum identities.

- $\max_x (x \pm 1) = \max_x (x) \pm 1$

$$\max_x \log p(x) = \max_x \sum_c \log \phi_c(x_c) - \log Z$$

6. Let  $\theta_c(x_c) = \log \phi_c(x_c)$ .

$$\max_x \log p(x) = \max_x \sum_{c \in C} \theta_c(x_c) - \log Z$$



- As  $\log Z$  is outside the scope of the maximization, if you desire to infer the set of conditions that maximizes the probability of some real-world phenomenon being observed, then solve the following optimization problem:

$$\arg \max_x \log p(x) = \arg \max_x \sum_{c \in C} \theta_c(x_c)$$

- **Important Note 1:** Without  $Z$ , this optimization problem suggests that MAP inference is computationally cheaper than marginal inference questions.
- **Important Note 2:** As maximization and summation both distribute over products, techniques used to solve marginal inference problems can be used to solve MAP inference problems.

### 3.14 Graph Cuts - How can MAP inference problems be solved using graph cuts?

- A **graph cut** of an undirected graph  $G = (V, E)$  is a partition of  $V$  into two disjoint sets  $V_s$  and  $V_t$ .
- The **min-cut** problem is to find the partition  $V_s, V_t$  that minimize the cost of the graph cut.
  - The cost of a graph cut is the sum of the nonnegative costs of the edges that cross between the two partitions:

$$\text{cost}(V_s, V_t) = \sum_{v_1 \in V_s, v_2 \in V_t} \text{cost}(v_1, v_2)$$

- **Time Complexity 1:**  $O(|E||V| \log |V|)$
  - **Time Complexity 2:**  $O(|V|^3)$
- A MAP inference problem can be reduced into the min-cut problem in certain restricted cases of MRFs with binary variables.

### 3.15 Linear Programming - How can MAP inference problems be solved using linear programming?

- An approximate approach to computing the MAP values is to use Integer Linear Programming by introducing:
  - An indicator variable per variable in the PGM.
  - An indicator variable per edge/cliue in the PGM.
  - Constraints on consistent values in cliques.

### 3.16 Local Search - How can MAP inference problems be solved using local search?

- A heuristic solution that starts with an arbitrary assignment and performs modifications on the joint assignment that locally increase the probability.

### 3.17 Branch and Bound - How can MAP inference problems be solved using branch and bound?

- An exhaustive solution that searches over the space of assignments while pruning branches that can be provably shown not to contain a MAP assignment.

### 3.18 Simulated Annealing - How can MAP inference problems be solved using simulated annealing?

- A sampling solution that expresses a probability distribution with the following:

$$p_t(x) \propto \exp \left( \frac{1}{t} \sum_{c \in C} \theta_c(x_c) \right)$$

- Where  $t$  is temperature.
  - \*  $t \rightarrow \infty \implies p_t$  approaches a continuous uniform distribution.
  - \*  $t \rightarrow 0 \implies p_t$  approaches a continuous exponential distribution with a significant peak of  $\arg \max_x \sum_{c \in C} \theta_c(x_c)$ .
- As the peak is a MAP assignment, a sampling algorithm starting with a high temperature which gradually decreases can eventually find the peak, given a sufficiently slow cooling rate.

### 3.19 Sampling-Based Inference

#### 3.20 Motivation - Why does sampling-based inference algorithms exist?

- **Exact Inference Algorithms:** Slow / NP-Hard
- **Approximate Inference Algorithms:** Marginal Inference, MAP Inference, Expectations

##### 3.20.1 Expectations $\mathbb{E}[f(X)]$ - Why do we want to estimate expectations of random variables?

- Abstractly, approximate inference algorithms want to estimate the probability of some real-world phenomenon.
- Mathematically, estimating a probability  $p(x)$  is a **SPECIALIZATION** of estimating an expectation  $\mathbb{E}_{x \sim p}[f(x)] = \sum_x f(x)p(x)$
- If  $f(x) = \mathbb{I}_{|x|}$ , where  $\mathbb{I}_{|x|}$  is an indicator function for event  $x$ ,

$$\mathbb{E}_{x \sim p}[\mathbb{I}_{|x|}] = p(x)$$

#### 3.21 Multinomial Sampling - How do you sample a discrete CPD?

1. Let  $p$  be a multinomial probability distribution with event values  $\{x^1, \dots, x^k\}$  and event probabilities  $\{\theta_1, \dots, \theta_k\}$ .
2. Generate a sample  $s$  uniformly from the interval  $[0, 1]$ .
3. Partition the interval into  $k$  subintervals:

$$[0, \theta_1), [\theta_1, \theta_1 + \theta_2), \dots, \left[ \sum_{j=1}^{i-1} \theta_j, \sum_{j=1}^i \theta_j \right)$$

4. If  $s$  is in the  $i$ th interval, then the sampled value is  $x^i$ .

- **Time Complexity:**  $O(\log k)$  - Using Binary Search
- Remember Baye's Rule:  $p(y | x) = \frac{p(x,y)}{p(x)}$ 
  - $p(x,y)$  is a multinomial probability distribution.

### 3.22 Forward Sampling - How do you sample a discrete Bayesian network?

1. Let  $G$  be a Bayesian network representing a probability distribution  $p(x_1, \dots, x_n)$ .
2. Sample the variables in a topological order.
3. Sample the successor variables by conditioning these node's CPDs to the values sampled by their ancestors.
4. Repeat until all  $n$  variables have been sampled.

- **Time Complexity:**  $O(n)$

### 3.23 Monte Carlo Integration/Estimation - How do you take a large number of samples to estimate expectations?

- *Monte Carlo  $\approx$  Large Number of Samples*

$$\mathbb{E}_{x \sim p}[f(x)] \approx I_T = \frac{1}{T} \sum_{t=1}^T f(x^t)$$

- Where  $x^1, \dots, x^T$  are **i.i.d.** samples drawn according to  $p$ .

$$\mathbb{E}_{x^1, \dots, x^T \sim \text{i.i.d. } p}[I_T] = \mathbb{E}_{x \sim p}[f(x)]$$

$$\text{Var}_{x^1, \dots, x^T \sim \text{i.i.d. } p}[I_T] = \frac{1}{T} \text{Var}_{x \sim p}[f(x)]$$

- Where the Monte Carlo estimate  $I_T$

#### 3.23.1 Implications - What is important about Monte Carlo estimations?

1.  $I_T$  is an unbiased estimator for  $\mathbb{E}_{x \sim p}[f(x)]$ .
2. Referencing the **Weak Law of Large Numbers**, if  $T \rightarrow \infty$ , then  $I_T \rightarrow \mathbb{E}_{x \sim p}[f(x)]$ .

### 3.24 Rejection Sampling - How does rejection sampling work?

- Compute a target probability distribution  $p(x)$  by sampling a proposal probability distribution  $q(x)$ , rejecting samples inconsistent with  $p(x)$ , and applying the Monte Carlo estimation.
  - **Examples:** See [Rejection Sampling](#)
  - **Disadvantage:** *Ignores Many Samples*

### 3.25 Importance Sampling - How does importance sampling work?

- Compute a target probability distribution  $p(x)$  by sampling a proposal probability distribution  $q(x)$ , reweighing samples with  $w(x) = \frac{p(x)}{q(x)}$ , and applying the Monte Carlo estimation.

$$\begin{aligned} \mathbb{E}_{x \sim p}[f(x)] &= \sum_x f(x) p(x) \\ &= \sum_x f(x) \frac{p(x)}{q(x)} q(x) \\ &= \mathbb{E}_{x \sim q}[f(x) w(x)] \\ &\approx \frac{1}{T} \sum_{t=1}^T f(x^t) w(x^t) \end{aligned}$$

- **Examples:** See *Importance Sampling*
- **Advantage:** Uses All Samples

### 3.26 Normalized Importance Sampling - How does normalized importance sampling work?

1. Let  $p(x)$  be unknown.
2. Let  $\tilde{p}(x) = Z \cdot p(x)$  be known.
3. The weight  $w(x) = \frac{\tilde{p}(x)}{q(x)}$  is invalid for unnormalized importance sampling.
4. The **normalizing constant** of the distribution  $\tilde{p}(x)$  is the following:

$$\mathbb{E}_{x \sim q}[w(x)] = \sum_x q(x) \frac{\tilde{p}(x)}{q(x)} = \sum_x \tilde{p}(x) = Z$$

5. The **normalized importance sampling estimator** is the following:

$$\begin{aligned} \mathbb{E}_{x \sim p}[f(x)] &= \sum_x f(x) p(x) \\ &= \sum_x f(x) \frac{p(x)}{q(x)} q(x) \\ &= \frac{1}{Z} \sum_x f(x) \frac{\tilde{p}(x)}{q(x)} q(x) \\ &= \frac{1}{Z} \mathbb{E}_{x \sim q}[f(x) w(x)] \\ &= \frac{\mathbb{E}_{x \sim q}[f(x) w(x)]}{\mathbb{E}_{x \sim q}[w(x)]} \end{aligned}$$

### 3.27 Markov Chain - What is a Markov chain?

- **Markov Chain:** A sequence of random variables  $S_0, S_1, S_2, \dots$  with each random variable  $S_i \in \{1, 2, \dots, d\}$  taking one of  $d$  possible values.
  - *Initial State:*  $P(S_0)$
  - *Subsequent States:*  $P(S_i \mid S_{i-1})$
- **Markov Assumption:**  $S_i$  cannot depend directly on  $S_j$  where  $j < i - 1$ .

### 3.28 Stationary Distribution - Why is it important for a stationary distribution to exist?

- Let  $T_{ij} = P(S_{\text{new}} = i \mid S_{\text{prev}} = j)$  be a  $d \times d$  transition probability matrix.
- If the initial state  $S_0$  is drawn from a vector probabilities  $p_0$ , the probability  $p_t$  of ending in EACH STATE after  $t$  steps is the following:

$$p_t = T^t p_0$$

- Where  $T^t$  is matrix exponentiation.

- **Stationary Distribution:** If it exists, the limit  $\pi = \lim_{t \rightarrow \infty} p_t$ .
- **Important Note 1:** A Markov chain whose states are joint assignments to the variables in a probabilistic graphical model  $p$  has a stationary distribution equal to  $p$ .

### 3.28.1 Existence of Stationary Distribution

- **Irreducibility:** It is possible to get from any state  $x$  to any other state  $x'$  with probability  $> 0$  in a finite number of steps.
- **Aperiodicity:** It is possible to return to any state at any time, i.e. there exists an  $n$  such that for all  $i$  and all  $n' \geq n$ ,  $P(s_{n'} = i \mid s_0 = i) > 0$ .
- **Important Note 2:** An irreducible and aperiodic finite-state Markov chain has a stationary distribution.

### 3.29 Markov Chain Monte Carlo - How do you sample from a MCMC?

1. Let  $T$  be a **transition operator** specifying a Markov chain whose stationary distribution is  $p$ .
2. Let  $x_0$  be an initial assignment to the variables of  $p$ .
3. Run the Markov chain from  $x_0$  for  $B$  *burn-in* steps.
  - If  $B$  is sufficiently large,  $\pi \rightarrow p$ .
4. Run the Markov chain for  $N$  *sampling* steps and collect all the states that it visits.
  - The collection of states form samples from  $p$ .

#### 3.29.1 Applications of Markov Chain Monte Carlo

1. Use samples for Monte Carlo integration to estimate expectations.
2. Use samples to perform marginal inference.
3. Use the sample with the highest probability to perform MAP inference.

### 3.30 Gibbs Sampling - How do you construct a MCMC?

1. Let  $x_1, \dots, x_n$  be an ordered set of variables.
  2. Let  $x^0 = (x_1^0, \dots, x_n^0)$  be a starting configuration.
  3. Repeat until convergence for  $t = 1, 2, \dots$ ,
    1. Set  $x \leftarrow x^{t-1}$ .
    2. For each variable  $x_i$ ,
      1. Sample  $x'_i \sim p(x_i \mid x_{-i})$ .
        - Where  $x_{-i}$  is all variables in  $x$  except  $x_i$
      2. Update  $x \leftarrow (x_1, \dots, x'_i, \dots, x_n)$ .
    3. Set  $x^t \leftarrow x$
- **Important Note 1:** When  $x_i$  is updated, its new value is immediately used for sampling other variables  $x_j$ .
  - **Important Note 2:** Every iteration of  $x^t$  is a new sample from  $p$ .

## 4 Learning

- **Problem:** Given a dataset  $D$  of  $m$  i.i.d. samples from some underlying distribution  $p^*$ , how do you fit the best model, given a family of models  $M$ , to make useful predictions?
  - **Parameter Learning:** Where the graph structure is known, and we want to estimate the factors.

- **Structure Learning:** Where we want to estimate the graph, i.e. determine from data how the variables depend on each other.
- **Solution:** Best Approximation of  $p^*$ 
  - **Density Estimation:** We are interested in the full distribution.
  - **Specific Prediction Tasks:** We are using the distribution to make a prediction.
    - \* e.g. Is this email spam or not?
  - **Structure or Knowledge Discovery:** We are interested in the model itself.
    - \* e.g. How do some genes interact with each other?

#### 4.1 Maximum Likelihood Estimation

#### 4.2 Motivation - Why does maximum likelihood estimation exist?

- **Goal:** How do we approximate  $p$  as close as possible to  $p^*$ ?
- **Approach:** When the KL divergence between  $p$  and  $p^*$  is minimal,  $p$  is as close as possible to  $p^*$ .

##### 4.2.1 KL Divergence - What is KL divergence?

- **KL Divergence:** How different is one probability distribution from another probability distribution?

$$KL(p^* \parallel p) = \sum_x p^*(x) \log \frac{p^*(x)}{p(x)} = -H(p^*) - \mathbb{E}_{x \sim p^*} [\log p(x)]$$

##### 4.2.2 Minimal KL Divergence - When is KL divergence minimal?

- **General Idea:** Minimizing KL Divergence  $\iff$  Maximizing Likelihood

$$\min KL(p^* \parallel p) \iff \max \mathbb{E}_{x \sim p^*} [\log p(x)]$$

- Because  $p^*$  is unknown, approximate the log-likelihood with the empirical log-likelihood using a Monte-Carlo estimate.

$$\mathbb{E}_{x \sim p^*} [\log p(x)] \approx \frac{1}{|D|} \sum_{x \in D} \log p(x)$$

##### 4.2.3 Maximum Likelihood Learning - How do you fit the best model using maximum likelihood learning?

- Given a family of models  $M$ , to fit the best model  $p$ , compute the following.

$$\max_{p \in M} \mathbb{E}_{x \sim p^*} [\log p(x)] \approx \max_{p \in M} \frac{1}{|D|} \sum_{x \in D} \log p(x)$$

#### 4.3 Definition - What is maximum likelihood estimation?

- **Maximum Likelihood Estimation:** Given a data set  $D$ , choose parameters  $\hat{\theta}$  that satisfy the following.

$$\max_{\theta \in \Theta} L(\theta, D)$$

- i.e., Maximize the parameters  $\theta$  to best fit the data set  $D$ .

#### 4.4 Loss Function - What is a loss function?

- **Loss Function** ( $L(x, p)$ ): A measure of the loss that a model distribution  $p$  makes on a particular instance  $x$ .
  - e.g., *MLE Loss Function*:  $L(x, p) = -\log p(x)$
- **Important Note**: Assuming instances are sampled from some distribution  $p^*$ , to fit the best model, **MINIMIZE** the expected loss.

$$\mathbb{E}_{x \sim p^*}[L(x, p)] \approx \frac{1}{|D|} \sum_{x \in D} L(x, p)$$

#### 4.5 Likelihood Function - What is a likelihood function?

- **Likelihood Function** ( $L(\theta, D)$ ): The probability of observing the i.i.d. samples  $D$  for all permissible values of the parameters  $\theta$ .

##### 4.5.1 Example - Likelihood Function

1. Let  $p(x)$  be a probability distribution where  $x \in \{h, t\}$  such that  $p(x = h) = \theta$  and  $p(x = t) = 1 - \theta$ .
2. Let  $D = \{h, h, t, h, t\}$  be observed i.i.d. samples.
3. Accordingly,  $p(x)$  models the outcome of a biased coin where parameter  $\theta$  represents the probability of flipping heads and  $1 - \theta$  represents the probability of flipping tails.
4. Express the likelihood function as the following.

$$L(\theta, D) = \theta \cdot \theta \cdot (1 - \theta) \cdot \theta \cdot (1 - \theta) = \theta^3 \cdot (1 - \theta)^2$$

#### 4.6 Maximum Likelihood Learning - How does maximum likelihood learning estimate the CPDs in Bayesian networks?

1. Let  $p(x) = \prod_{i=1}^n \theta_{x_i | x_{pa(i)}}$  be a Bayesian network.
  - Where  $\theta_{x_i | x_{pa(i)}}$  are parameters (CPDs) with **UNKNOWN VALUES**.
2. Let  $D = \{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$  be i.i.d. samples.
3. Let  $L(\theta, D) = \prod_{i=1}^n \prod_{j=1}^m \theta_{x_i^j | x_{pa(i)}^j}$  be the likelihood function.
4. Log and collect like terms of the likelihood function.

$$\log L(\theta, D) = \sum_{i=1}^n \sum_{x_{pa(i)}} \sum_{x_i} \#(x_i, x_{pa(i)}) \cdot \log \theta_{x_i | x_{pa(i)}}$$

5. Maximize the (log) likelihood function by decomposing it into separate maximizations for the local conditional distributions.
- **Important Note**: The maximum-likelihood estimates of the parameters (CPDs) have closed-form solutions.

$$\theta_{x_i | x_{pa(i)}}^* = \frac{\#(x_i, x_{pa(i)})}{\#(x_{pa(i)})}$$

## 4.7 Bayesian Learning

### 4.8 Motivation - What are some problems with maximum likelihood estimation?

- A maximum likelihood estimate does not change as more data is observed because it assumes that the only source of uncertainty is explained by the parameters that are being fitted.
- **Problem 1:** *Cannot Improve Confidence*
- **Problem 2:** *Cannot Incorporate Prior Knowledge*

### 4.9 Definitions - What are a *prior* and a *posterior*?

- **Bayesian Learning:** Explicitly model uncertainty over both variables  $X$  and parameters  $\theta$  by letting parameters be random variables.
- A **prior** is the earlier probability distribution of parameter  $\theta$  **BEFORE** observing data  $D$ .
- A **posterior** is the later probability distribution of parameter  $\theta$  **AFTER** observing data  $D$ .

$$p(\theta | D) = \frac{p(D | \theta)p(\theta)}{p(D)} \propto p(D | \theta)p(\theta)$$

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

- **Important Note 1:** Bayes' rule allows prior knowledge to be incorporated into a model's parameters.
- **Important Note 2:** Using Bayes' rule, the numerator is easy to calculate, but the denominator is difficult to calculate.
- **Important Note 3:** The expected value of the posterior  $p(\theta | D)$  is the estimate of the parameter  $\theta$  that causes  $p$  to be as close as possible to  $p^*$ .

### 4.10 Conjugate Priors - What is a *conjugate prior*?

- A parametric family  $\phi$  is **conjugate** for the likelihood  $P(D | \theta)$  if:

$$P(\theta) \in \phi \implies P(\theta | D) \in \phi$$

- **Important Note:** If the normalizing constant of  $\phi$  is known, then the denominator in Bayes' rule is easy to calculate.



#### 4.11 Beta Distribution - What is the Beta distribution?

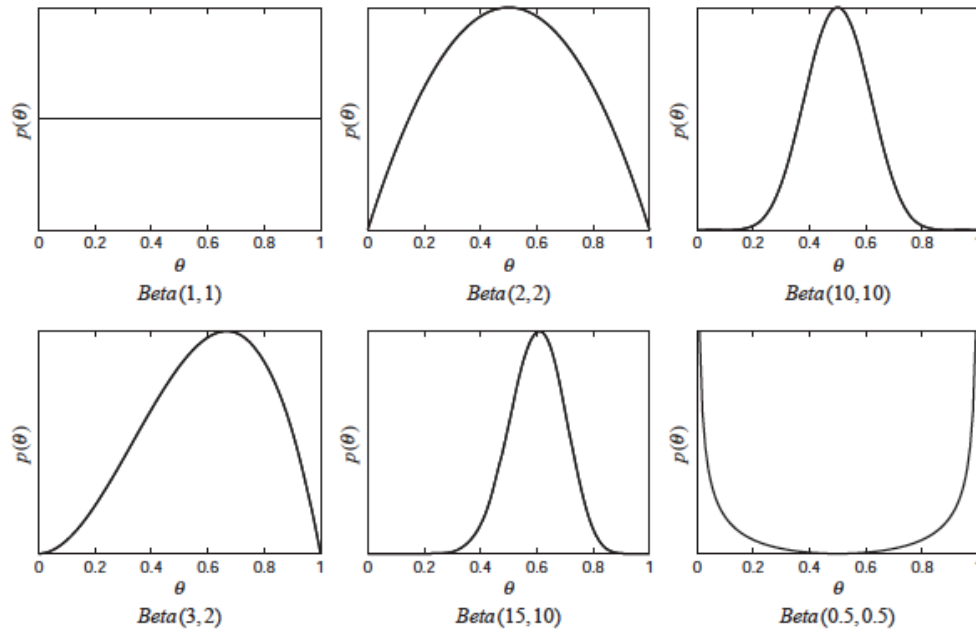


Figure 17.4 Examples of Beta distributions for different choices of hyperparameters

##### Examples of Beta Distribution

- A **Beta distribution** is parameterized by two hyperparameters  $\alpha \in \mathbb{R}$ , and  $\beta \in \mathbb{R}$  with the following continuous probability distribution.

$$\theta \sim \text{Beta}(\alpha, \beta) \implies p(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$

- Where  $\theta \in (0, 1)$ .
- Where the constant  $\alpha$  intuitively corresponds to the number of **SUCCESS** outcomes.
- Where the constant  $\beta$  intuitively corresponds to the number of **FAILURES** outcomes.
- Where the constant  $B(\alpha, \beta)$  is a normalizing constant defined by the following.

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha + \beta)}$$

- Where the Gamma function  $\Gamma(x)$  is the continuous generalization of the factorial function defined by the following.

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt$$

- **Expected Value:**

$$\mathbb{E}[X] = \frac{\alpha}{\alpha + \beta}$$

#### 4.12 Conjugate Priors and Beta Distribution - How do you calculate a posterior with data observed from a binary process?

Think Coin Toss

- The beta distribution is the conjugate prior for the following probability distributions:
  - **Bernoulli**: A discrete Bernoulli random variable,  $X$ , is the outcome from a single experiment from which this outcome is classified as either a success,  $X = 1$  with probability  $p$ , or a failure,  $X = 0$  with probability  $1 - p$ .
  - **Binomial**: A discrete binomial random variable,  $X$ , is the number of successful outcomes from a sequence of  $n$  independent experiments in which each experiment has an outcome classified as either a success with probability  $p$  or a failure with probability  $1 - p$ .
  - **Geometric**: A discrete geometric random variable,  $X$ , is the number of Bernoulli trials with probability  $p$  needed to get one success.
  - **Negative Binomial**: A discrete negative binomial random variable,  $X$ , is the number of successes in a sequence of independent and identically distributed Bernoulli trials before  $r$  failures.
- **Important Note**: To best fit a binary model with a probability distribution  $p \approx p^*$ , the beta distribution can be used by the following.
  1. Assign  $\text{Beta}(\alpha, \beta)$  as a prior to  $p$ .
  2. Observe  $n$  data points generated by a binary process with unknown, underlying  $p^*$ .
    - If  $X_i \sim \text{Bernoulli}(p^*)$ , then the posterior is  $\text{Beta}(\alpha + \sum_{i=1}^n x_i, \beta + n - \sum_{i=1}^n x_i)$ .
    - If  $X_i \sim \text{Binomial}(N, p^*)$ , then the posterior is  $\text{Beta}(\alpha + \sum_{i=1}^n x_i, \beta + \sum_{i=1}^n N_i - \sum_{i=1}^n x_i)$ .
    - If  $X_i \sim \text{Geometric}(p^*)$ , then the posterior is  $\text{Beta}(\alpha + n, \beta + \sum_{i=1}^n x_i - n)$ .
    - If  $X_i \sim \text{Negative-Binomial}(r, p^*)$ , then the posterior is  $\text{Beta}(\alpha + \sum_{i=1}^n x_i, \beta + rn)$ .

#### 4.13 Dirichlet Distribution (Multivariate Beta Distribution) - What is the Dirichlet distribution?

- A **Dirichlet distribution** is parameterized by the following hyperparameters  $\alpha = (\alpha_1, \dots, \alpha_K) \in \mathbb{R}^K$  with the following continuous probability distribution.

$$\theta \sim \text{Dirichlet}(\alpha) \implies p(\theta) = \frac{1}{B(\alpha)} \prod_{i=1}^K \theta_i^{\alpha_i - 1}$$

- Where  $\theta_i \in (0, 1)$  and  $\sum_{i=1}^K \theta_i = 1$ .
- Where the constant  $\alpha_i$  intuitively corresponds to the number of outcomes for category  $i$ .
- Where the constant  $B(\alpha)$  is a normalizing constant defined by the following.

$$B(\alpha) = \frac{\prod_{i=1}^K \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^K \alpha_i)}$$

- Where the Gamma function  $\Gamma(x)$  is the continuous generalization of the factorial function defined by the following.

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt$$

- **Expected Value:**

$$\mathbb{E}[X_i] = \frac{\alpha_i}{\sum_k \alpha_k}$$

#### 4.14 Conjugate Priors and Dirichlet Distribution - How do you calculate a posterior with data observed from a categorical process?

Think K-Sided Dice Roll

- The Dirichlet distribution is the conjugate prior for the following probability distributions:
  - **Categorical** (*Generalized Bernoulli*): A discrete categorical random variable,  $X$ , is the outcome from a single experiment from which this outcome is classified as one of  $K$  categories with probability  $p_i > 0$  and  $\sum_{i=1}^K p_i = 1$ .
  - **Multinomial** (*Generalized Binomial*): A discrete multinomial random variable,  $X$ , is the number of outcomes from a sequence of  $n$  independent experiments in which each experiment has an outcome classified as one of  $K$  categories with probability  $p_i > 0$  and  $\sum_{i=1}^K p_i = 1$ .
- **Important Note:** To best fit a categorical model with a probability distribution  $\mathbf{p} \approx \mathbf{p}^*$ , the Dirichlet distribution can be used by the following.
  1. Assign  $\text{Dirichlet}(\boldsymbol{\alpha})$  as a prior to  $\mathbf{p}$ .
  2. Observe  $n$  data points generated by a categorical process with unknown, underlying  $\mathbf{p}^*$ .
    - If  $X_i \sim \text{Categorical}(\mathbf{p}^*)$ , then the posterior is  $\text{Dirichlet}(\boldsymbol{\alpha} + (c_1, \dots, c_K))$ .
      - \* Where  $c_i$  is the number of observations in category  $i$ .
    - If  $X_i \sim \text{Multinomial}(N, \mathbf{p}^*)$ , then the posterior is  $\text{Dirichlet}(\boldsymbol{\alpha} + \sum_{i=1}^n \mathbf{x}_i)$ .

#### 4.15 Gamma Distribution - What is the Gamma distribution?

- A **Gamma distribution** is parameterized by two hyperparameters  $\alpha \in \mathbb{R}$ , and  $\beta \in \mathbb{R}$  with the following continuous probability distribution.

$$\theta \sim \text{Gamma}(\alpha, \beta) \implies p(\theta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \theta^{\alpha-1} e^{-\beta\theta}$$

- Where  $\theta \in (0, \infty)$ .
- Where the constants  $\alpha$  and  $\beta$  intuitively corresponds to  $\alpha$  total occurrences in  $\beta$  intervals.
- Where the Gamma function  $\Gamma(x)$  is the continuous generalization of the factorial function defined by the following.

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$$

- **Expected Value:**

$$\mathbb{E}[X] = \frac{\alpha}{\beta}$$

#### 4.16 Conjugate Priors and Gamma Distribution - How do you calculate a posterior with data observed from a Poisson point process?

Think Rate (Events/Second)

- The Gamma distribution is the conjugate prior for the following probability distributions:
  - **Poisson:** A discrete Poisson random variable,  $X$ , is the number of events occurring in a fixed interval of time or at a fixed rate.
  - **Exponential:** A continuous exponential random variable,  $X$ , is the time between events in a Poisson point process.
- **Important Note:** To best fit a Poisson point model with a probability distribution  $p \approx p^*$ , the Gamma distribution can be used by the following.
  1. Assign  $\text{Gamma}(\alpha, \beta)$  as a prior to  $p$ .
  2. Observe  $n$  data points generated by a Poisson point process with unknown, underlying  $p^*$ .
    - If  $X_i \sim \text{Poisson}(\lambda^*)$ , then the posterior is  $\text{Gamma}(\alpha + \sum_{i=1}^n x_i, \beta + n)$ .
    - If  $X_i \sim \text{Exponential}(N, \lambda^*)$ , then the posterior is  $\text{Gamma}(\alpha + n, \beta + \sum_{i=1}^n x_i)$ .

## 5 Decision Making Under Uncertainty

### 5.1 A/B Testing

- See *Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference*
  - *Chapter 1: Introduction to Bayesian Methods*
  - *Chapter 2: A little more on PyMC*
  - *Chapter 6: Getting our prior-ities straight*

### 5.2 Definition - What is A/B Testing?

- A randomized experiment with two variants,  $A$  and  $B$ , that determines which of the two variants is more effective with their respective cohorts at achieving successes in  $n$  trials.

### 5.3 Bayesian Learning and A/B Testing - How do you determine which of $A$ and $B$ are better after observing $n_A$ and $n_B$ trials?

1. Let  $0 < p_A < 1$  be the unknown probability that  $A$  is truly effective.
2. Let  $0 < p_B < 1$  be the unknown probability that  $B$  is truly effective.
3. Let  $\delta = A - B$  be the measure whether  $A$  is more effective than  $B$ .
4. Let  $\text{Beta}(1, 1)$  be the initial prior for  $p_A$ .
  - Use different values for  $\alpha$  and  $\beta$  to inject a subjective prior belief about  $p_A$ .
5. Let  $\text{Beta}(1, 1)$  be the initial prior for  $p_B$ .
  - Use different values for  $\alpha$  and  $\beta$  to inject a subjective prior belief about  $p_B$ .
6. Observe  $n_A$  data points generated by a Bernoulli process with unknown, underlying  $p_A$ .
7. Observe  $n_B$  data points generated by a Bernoulli process with unknown, underlying  $p_B$ .

8. Update the posterior  $p_A$  given  $n_A$  using Bayesian learning.
9. Update the posterior  $p_B$  given  $n_B$  using Bayesian learning.
10. Approximate the joint distribution of  $\delta$  using a MCMC constructed using MCMCs of posteriors  $p_A$  and  $p_B$ .
  - Count the number of samples greater than 0 for the probability that  $A$  is better than  $B$ ; i.e. the area under the curve after 0.
  - Count the number of samples less than 0 for the probability that  $A$  is worse than  $B$ ; i.e. the area under the curve before 0.

## 5.4 Multi-Armed Bandits

- See [Reinforcement Learning: An Introduction](#)
  - Chapter 2: Multi-armed Bandits

## 5.5 Definition - What is the Multi-armed bandit problem?

- A fixed limited set of resources must be allocated between competing (alternative) actions in a way that maximizes their expected reward, when each action's properties are only partially known at the time of allocation, and may become better understood as time passes or by allocating resources to the action.

## 5.6 Exploration-Exploitation Tradeoff Dilemma - What is the exploration-exploitation tradeoff dilemma?

- **Exploitation:** If we have chosen an action that yields a pretty good reward, do we keep choosing it to maintain a pretty good reward?
- **Exploration:** Otherwise, do we choose other actions in hopes of finding an even-better action?

## 5.7 $\epsilon$ -Greedy Bandits Algorithm - How do you choose actions using the $\epsilon$ -greedy bandits algorithm?

1. Let  $A_t$  be the action chosen from  $k$  possibilities at time step  $t$ .
2. Let  $R_t$  be the independent and identically distributed reward yielded by action  $A_t$ .
3. Let  $Q_t(a)$  be the expected reward from choosing action  $a$  until time step  $t$ .
4. Let  $N_t(a)$  be the number of times action  $a$  was chosen until time step  $t$ .
5. Initialize  $Q_0(a) \leftarrow 0, \forall a \in \{1, \dots, k\}$ .
6. Initialize  $N_0(a) \leftarrow 0, \forall a \in \{1, \dots, k\}$ .
7. Repeat for  $t = 0, 1, 2, \dots$ ,
  1. Set  $A_t \leftarrow \max_a Q_t(a)$  with probability  $1 - \epsilon$ .
  2. Set  $A_t \leftarrow \text{random}(a)$  with probability  $\epsilon$ .
  3. Sample  $R_t \leftarrow \text{bandit}(A_t)$ .
  4. Update  $N_{t+1}(A_t) \leftarrow N_t(A_t) + 1$ .
  5. Update  $Q_{t+1}(A_t) \leftarrow Q_t(A_t) + \frac{1}{N_t(A_t)}[R_t - Q_t(A_t)]$ .

## 5.8 Bayesian Bandits Algorithm - How do you choose actions using the Bayesian bandits algorithm?

1. Let  $A_t$  be the action chosen from  $k$  possibilities at time step  $t$ .
2. Let  $R_t$  be the independent and identically distributed reward yielded by action  $A_t$ .
3. Let  $p_t(a)$  be the initially unknown probabilities of the rewards yielded by choosing action  $a$  until time step  $t$ .
4. Repeat for  $t = 0, 1, 2, \dots$ ,
  1. Sample a random variable  $X_a$  using  $p_t(a)$  for each  $a \in \{1, \dots, k\}$ .
  2. Set  $A_t \leftarrow \max_a X_a$ .
  3. Sample  $R_t \leftarrow \text{bandit}(A_t)$ .
  4. Update  $p_{t+1}(A_t | R_t) \propto p_t(R_t | A_t)p_t(A_t)$ .