

# 6.034 Exam 4 Cheat Sheet

## Adaboost

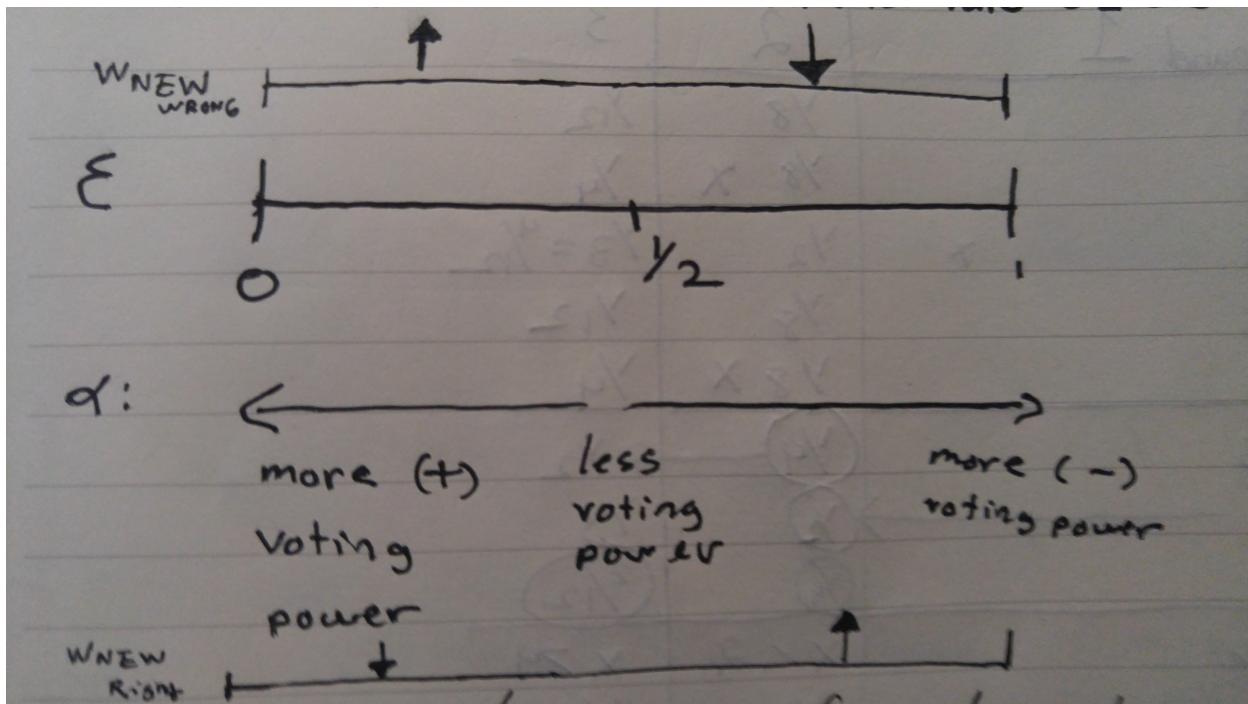
### Overview

Take imperfect (weak) classifiers  $h_i$  to create a strong overall classifier:

$$H(\vec{x}) = \text{SIGN}(\underbrace{\alpha_1}_{\text{voting power}} \cdot h_1(\vec{x}) + \alpha_2 \cdot h_2(\vec{x}) + \dots)$$

Each weak classifier  $h_i$  has an error rate,  $\epsilon$ , where  $0 \leq \epsilon \leq 1$

Error rate  $\epsilon = \sum w_i$



Note: after selecting a weak classifier  $h_i$ , the points that were incorrectly classified by  $h_i$  have their  $w_i$  increase, while the points that were correctly classified by  $h_i$  have their  $w_i$  decrease

We build  $H(\vec{x})$  in a series of rounds, each round:

- pick the best\*  $h_i$
- calculate voting power  $\alpha_i$
- append  $h_i$  to overall classifier,  $H$

\* best can mean two different things:

- smallest  $\epsilon$  value
- $\epsilon$  value furthest away from  $\frac{1}{2}$

## Steps

0. Initialize  $w_i = \frac{1}{N}$  (assign equal weight to each training points)
1. Find best\* h w/ lowest error rate
2. Calculate voting power:  $\alpha = \frac{1}{2} \ln \frac{1-\epsilon}{\epsilon}$
3. Check to see if we're finished:
  - H is good enough (perfectly classifies all of the training data)
  - no good weak classifiers left (i.e. best  $h_i$  has  $\epsilon_i = \frac{1}{2}$  - no better than a coin-flip )
  - have performed enough rounds of Adaboost
4. Update Weights:

$$w_{\text{NEW}} = \begin{cases} \frac{1}{2} \times \frac{1}{1-\epsilon} \times w_{\text{OLD}} & \text{if } w_{\text{OLD}} \text{ was correctly classified by newly selected } h_i \\ \frac{1}{2} \times \frac{1}{\epsilon} \times w_{\text{OLD}} & \text{if } w_{\text{OLD}} \text{ was incorrectly classified by newly selected } h_i \end{cases}$$

## (Somewhat) Random Helpful Facts

- If you have 3+ weak classifiers that make non-overlapping errors, you can make a perfect classifier
  - can't make a perfect classifier with less than 3
- Only after Round 1:  $\sum_{\text{right}} w_i = \sum_{\text{wrong}} w_i = \frac{1}{2}$
- At any point:  $0 < w_i \leq \frac{1}{2}$ 
  - notably:  $w_i$  can never be 0
- At any point:  $\sum w_i = 1$
- $w_i$  **must** change every round, since we aren't selecting classifiers with  $\epsilon_i = \frac{1}{2}$
- After selecting  $h_i$ , on the following round:  $\mathcal{E}(h_i) = \frac{1}{2}$
- Say we have two classifiers:  $h_1$  and  $h_2$ , and our criteria for selecting the best classifier on any given round is choosing the one with the lowest overall  $\epsilon_i$ . If  $h_1$  misclassifies points  $A, B$  and  $h_2$  misclassifies  $A, B, C$ , then  $h_2$  will **never** be selected.  $h_2$  has the superset of  $h_1$ 's errors.
  - If the “best” criteria is the furthest  $\epsilon_i$  from  $\frac{1}{2}$  instead, then  $h_2$  could be chosen.
- Adaboost tends **not** to overfit

## Bayes Nets

### Probability

**Exhaustion** If var  $A$  has 3 values:

$$P(A = a_1) + P(A = a_2) + P(A = a_3) = 1$$

$$P(A = a_1 | B = b_1) + P(A = a_2 | B = b_1) + P(A = a_3 | B = b_1) = 1$$

If  $A$  is a boolean value:

$$P(A = T) + P(A = F) = 1$$

$$P(A) + P(\bar{A}) = 1$$

### Conditionally Probability

$$P(A | B) = \frac{P(AB)}{P(B)}$$

### Independence

$$A \perp\!\!\!\perp B \iff P(A | B) = P(A)$$

$$P(A | B) = P(A | B)P(B) = P(A)P(B)$$

Conditional Independence:

$$A \perp\!\!\!\perp B | C \iff P(A | BC) = P(A | C)$$

### Chain Rule

$$P(x_1, \dots, x_n) = \prod_{i=0}^n P(x_i | x_{i+1} \dots x_n)$$

Ex:

$$P(AB) = P(A | B)P(B)$$

$$P(ABC) = P(A | BC)P(B | C)P(C)$$

### Bayes Rule

$$P(A | B) = \frac{P(A)P(B | A)}{P(B)}$$

### Number of Parameters

#### 2 Boolean Variables:

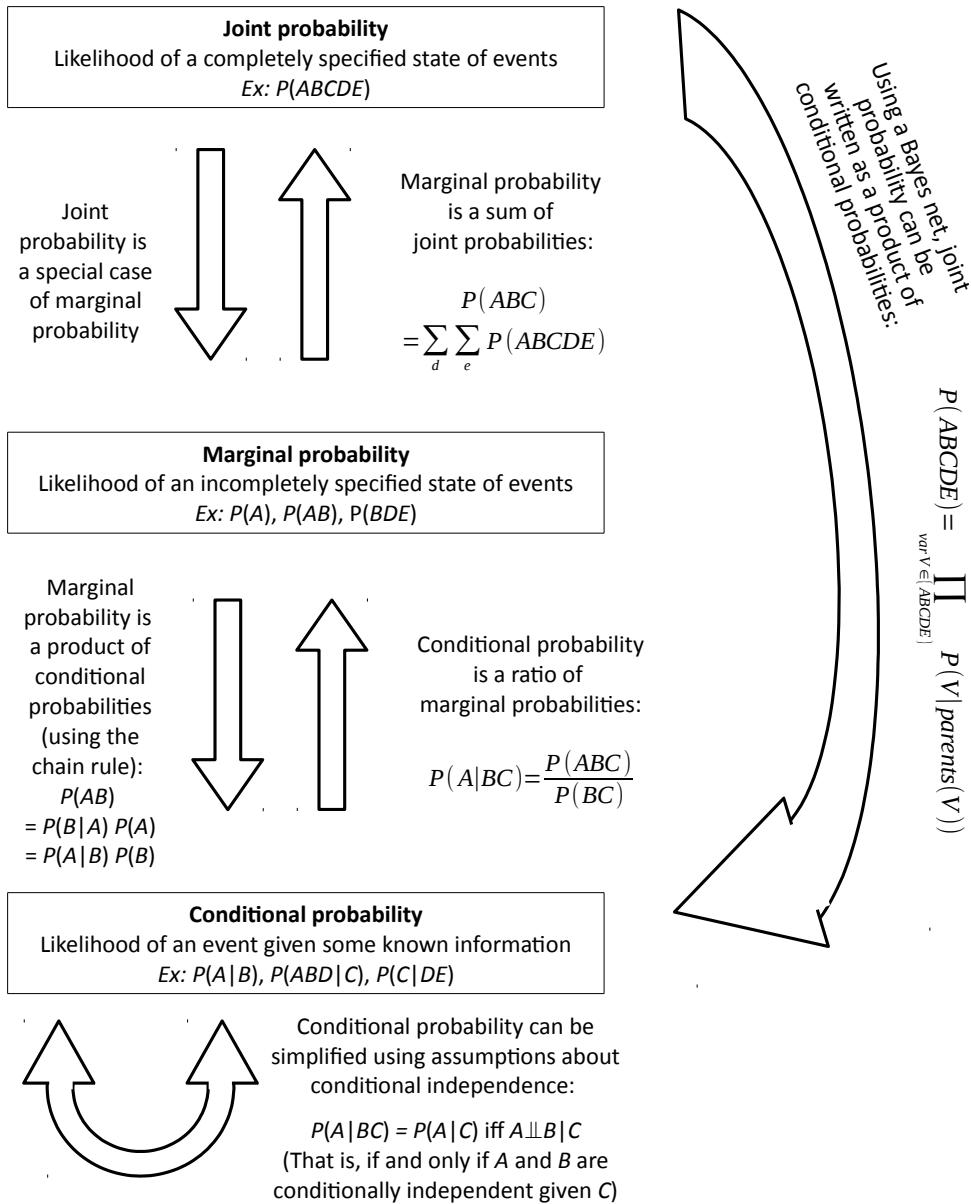
- no independence assumption:  $2^2 - 1 = 3$
- w/ independence assumption: 2, just need  $P(A), P(B)$

#### 100 Boolean Variables:

- no independence assumption:  $2^{100} - 1$
- w/ independence assumption: 100, just need  $P(A), P(B), \dots$

## Probability Flowchart

Use this flowchart to convert between different kinds of probabilities.  
In each example, the variables under consideration are  $A, B, C, D$ , and  $E$ .



(This handout is available at <http://web.mit.edu/jmn/www/6.034/probability-flowchart.pdf>)

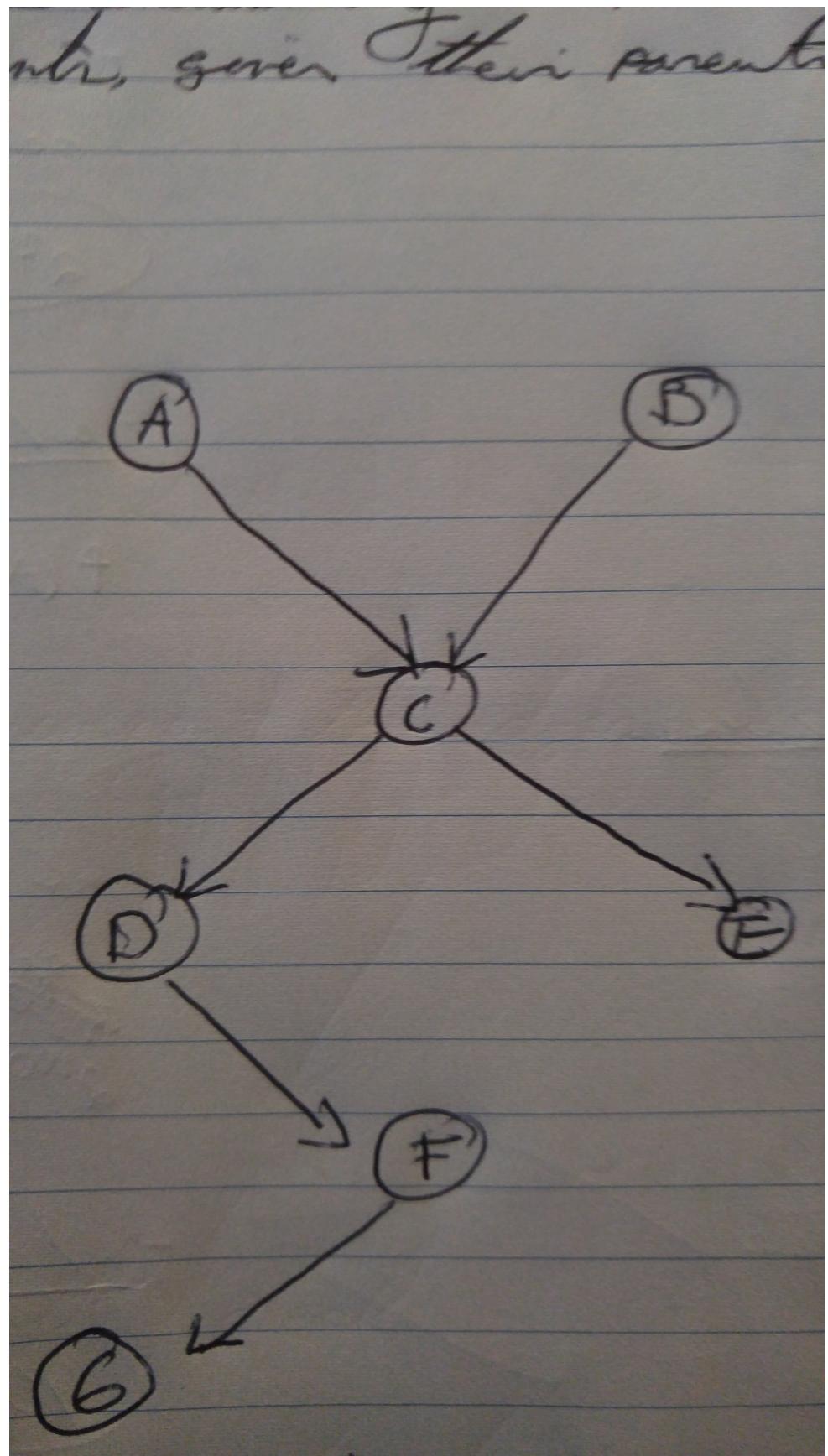
6.034 – November 2015

Page 1 of 1

Figure 1: Probability Flowchart

### **Bayes Net Assumption**

All variables are conditionally independent of their non-descendeants, given their parents



So, using the above net:

$$F \perp\!\!\!\perp \text{non-descendants}(F) \mid \text{parents}(F)$$

$$P(F \mid ABCDE)$$

$$P(F \mid A \not\perp\!\!\!\perp B \not\perp\!\!\!\perp C \not\perp\!\!\!\perp D \not\perp\!\!\!\perp E) = P(F \mid D)$$

### D-Separation

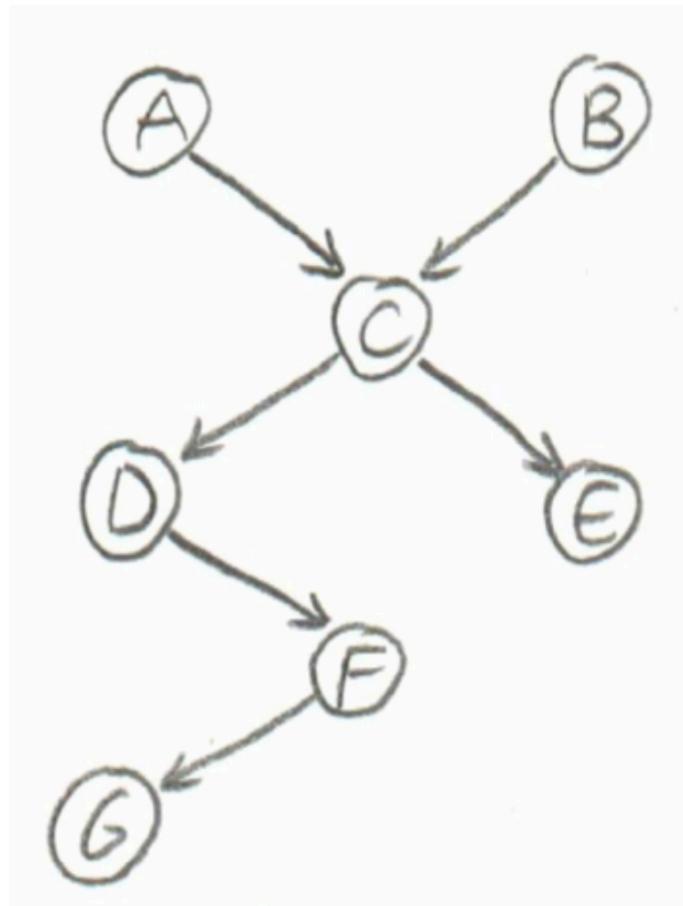


Figure 2: Example Net D-Separation

We start with an independence question in one of these forms: - “Are  $A$  and  $B$  conditionally independent, given {givens}?” - “Are  $A$  and  $B$  marginally independent?”

For instance, if we’re asked to figure out:  $P(A \mid BDF) \stackrel{?}{=} P(A \mid DF)$ , we can convert it into an independence question like this: “Are  $A$  and  $B$  independent, given  $D$  and  $F$ ?”

## Steps

1. Draw ancestral graph
  - Construct the “ancestral graph” of all variables mentioned in the probability expression. This is a reduced version of the original net, consisting only of the variables mentioned and all of their ancestors (parents, parents’ parents, etc.)
2. “Moralize” the ancestral graph by “marrying” the parents.
  - For each pair of variables with a common child, draw an undirected edge (line) between them. (If a variable has more than two parents, draw lines between every pair of parents.)
3. “Disorient” the graph by replacing the directed edges (arrows) with undirected edges (lines).
4. Delete the givens and their edges. If the independence question had any given variables, erase those variables from the graph and erase all of their connections, too. Note that “given variables” as used here refers to the question “Are  $A$  and  $B$  conditionally independent, given  $D$  and  $F$ ?”, **not** the equation:  $P(A | BDF) \stackrel{?}{=} P(A | DF)$ , and thus does not include  $B$ .

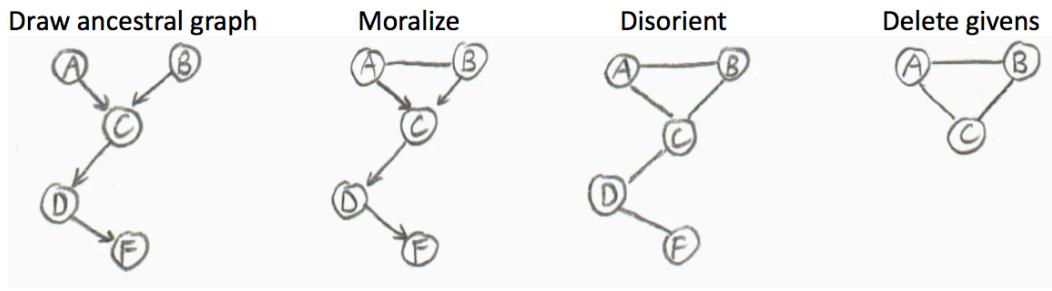


Figure 3: D-Separation Process

5. Read the answer off the graph.
  - If the variables are disconnected in this graph, they are guaranteed to be independent.
  - If the variables are connected in this graph, they are not guaranteed to be independent.\* Note that “are connected” means “have a path between them,” so if we have a path  $X-Y-Z$ ,  $X$  and  $Z$  are considered to be connected, even if there’s no edge between them.
  - If one or both of the variables are missing (because they were givens, and were therefore deleted), they are independent.

\* We can say “the variables are dependent, as far as the Bayes net is concerned” or “the Bayes net does not require the variables to be independent,” but we cannot guarantee dependency using d-separation alone, because the variables can still be numerically independent (e.g. if  $P(A | B)$  and  $P(A)$  happen to be equal for all values of  $A$  and  $B$ ).