COMP9418: Advanced Topics in Statistical Machine Learning

Bayesian Networks as Classifiers

Instructor: Gustavo Batista

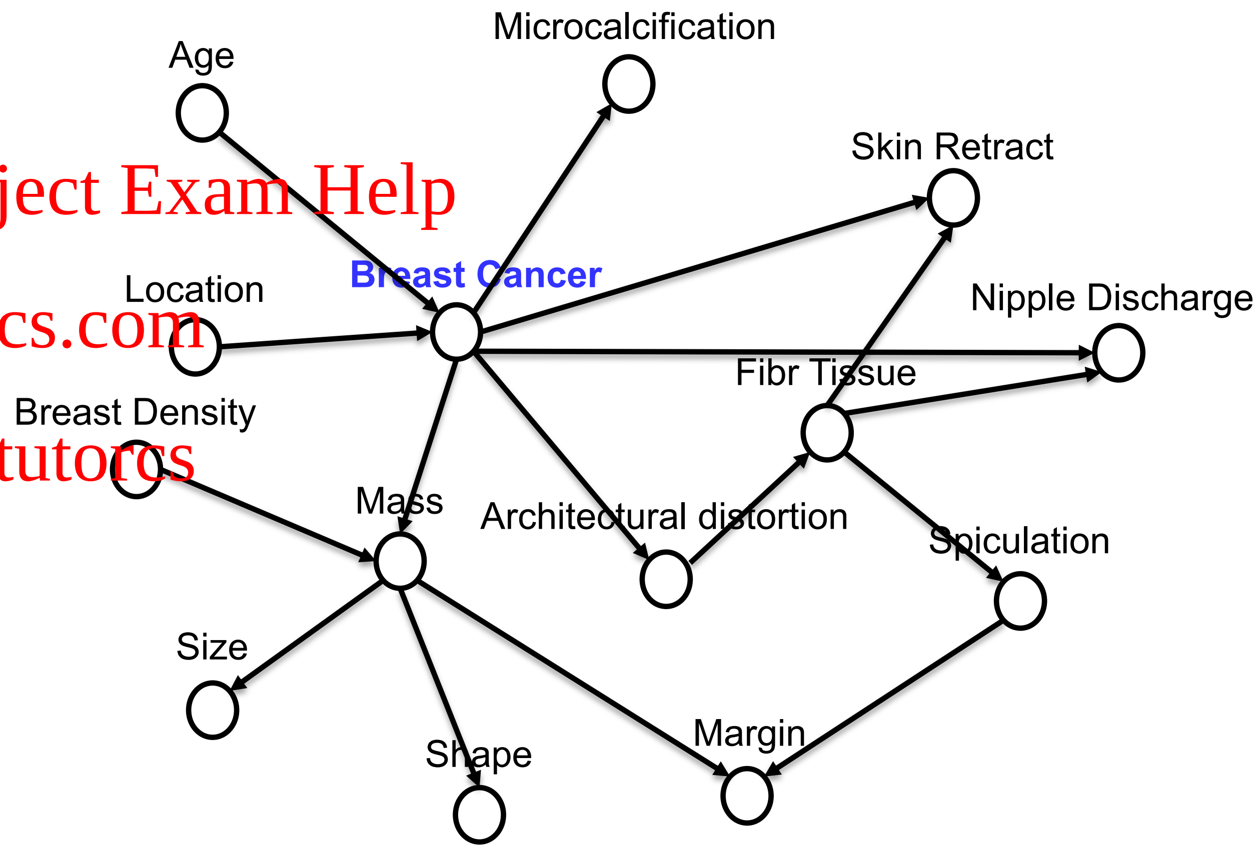
University of New South Wales

Introduction

- This lecture discusses the use of Bayesian networks as classifiers
  - Variables can be divided into attributes and class
  - We want to predict the class based on the information on variables
- Classification will help us to discuss several aspects of Bayesian networks
  - Such as independence, learning and inference
  - Some of these topics will be further discussed in forthcoming lectures
- We will review a simple Bayesian network for classification
  - The Naïve Bayes and some extensions

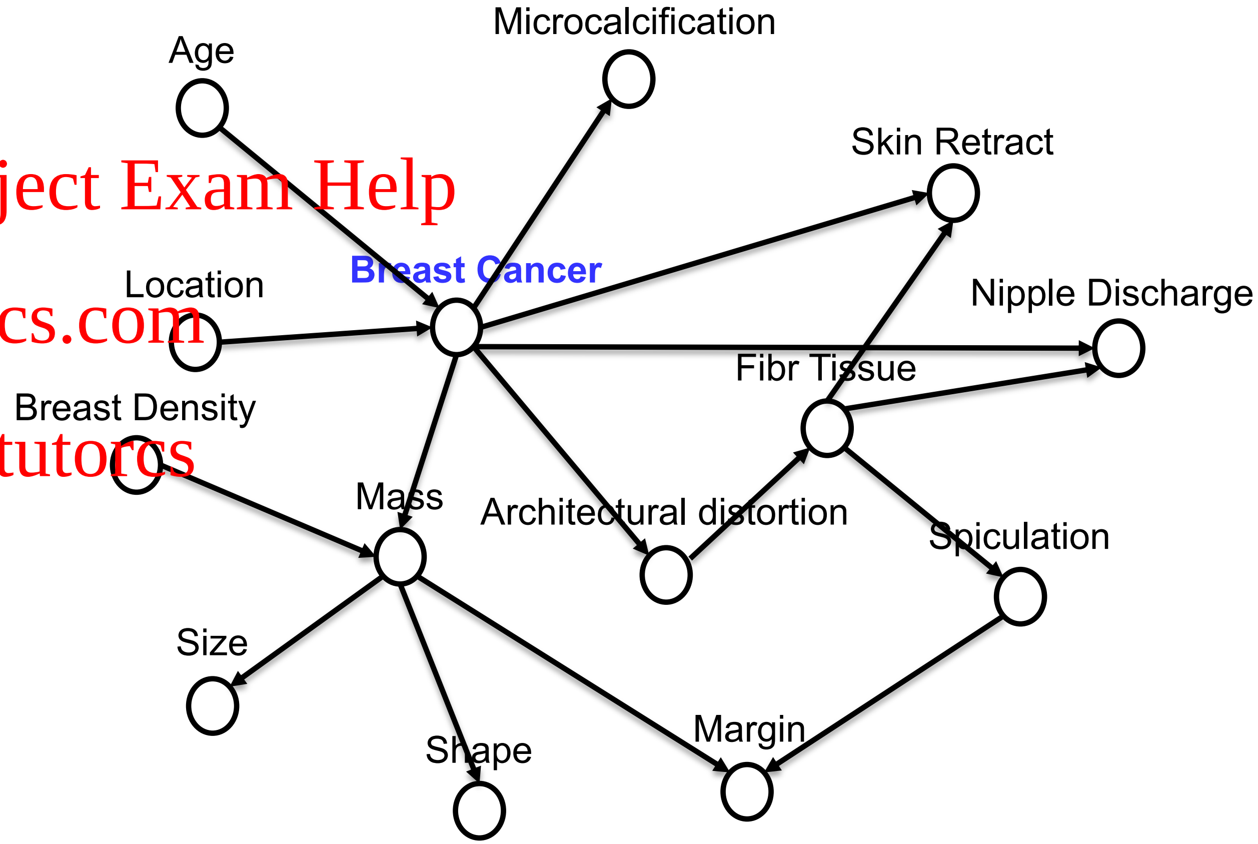
Bayesian Networks as Classifiers

- Suppose we have a Bayesian network for breast cancer diagnosis
  - Our aim is to predict whether a patient has breast cancer given a series of mammography results
- The network variables can be divided into two sets
  - Class (query variable)
  - Attributes (evidence)
- Some relevant aspects
  - Type of query
  - Independence assumptions
  - Learning structure and parameters



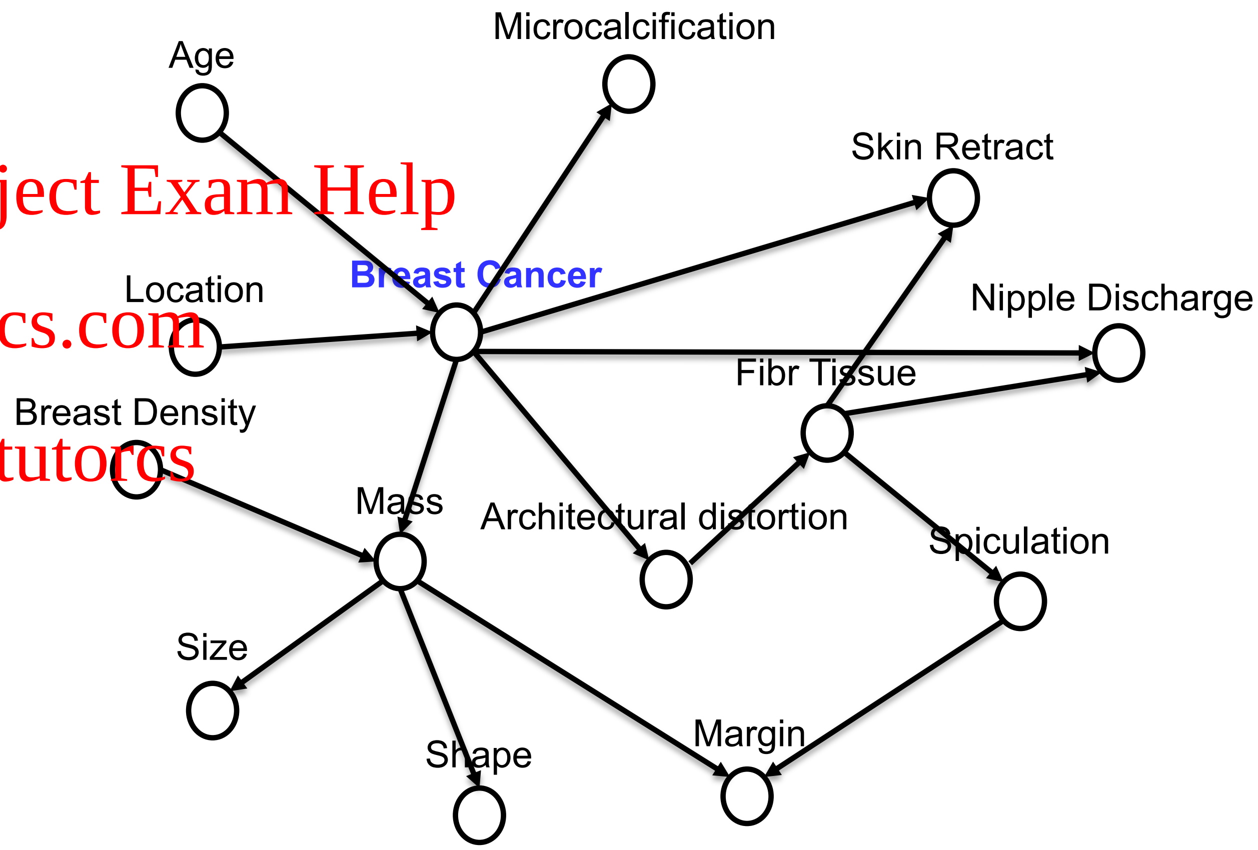
Bayesian Networks as Classifiers

- Given a set of attribute values for a patient, our objective is to correctly identify the class value
  - In this example, the values are "no", "insitu", "invasive"
  - Therefore, we have an MPE query with $Q = \{B\}$ and evidence set with the values of the remaining attributes
- Bayesian networks naturally handle missing data
  - If there is missing evidence in queries, we need to compute a MAP query
  - MAP queries are more costly than MPE since it involves eliminating unobserved variables



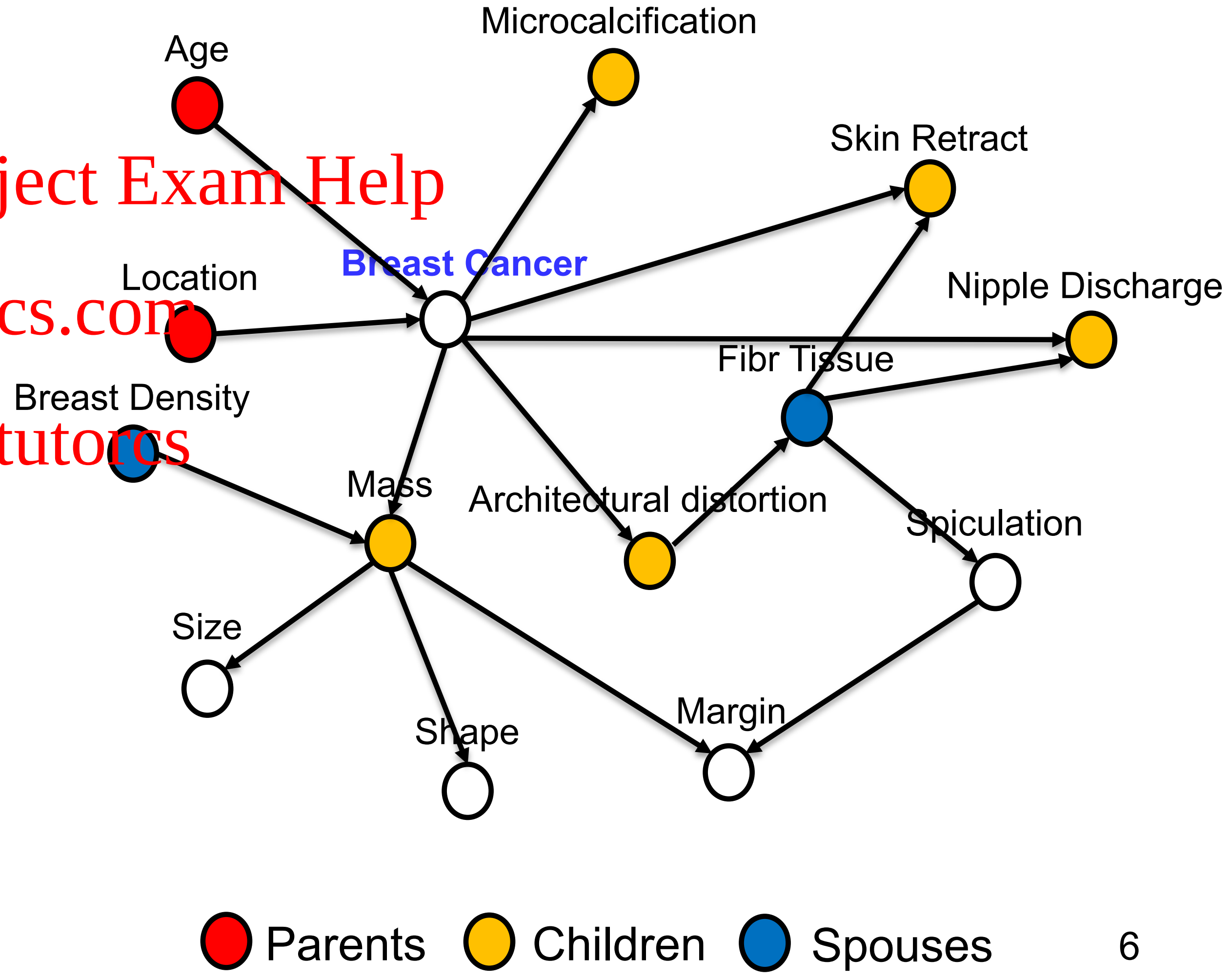
Classification of Complete Data

- A relevant question is whether all variables contribute to the classification given complete data
  - Given the network independence assumptions
  - We can use the concept of Markov blanket
- Markov blanket for X is constituted of its parents, children, and spouses



Classification of Complete Data

- A relevant question is whether all variables contribute to the classification given complete data
  - Given the network independence assumptions
  - We can use the concept of Markov blanket
- Markov blanket for X is constituted of its parents, children, and spouses
  - Not every variable contributes to the classification
  - Since some are d-separated given complete evidence
  - Let us gain more intuition looking at a simpler network

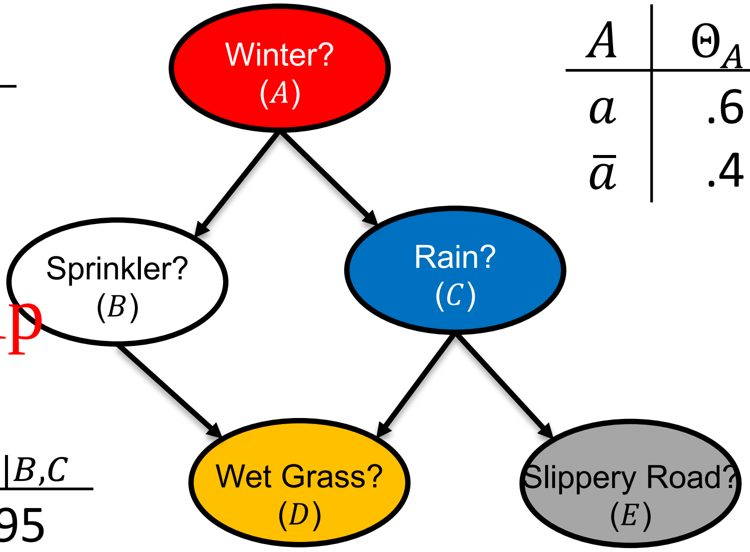


Classification of Complete Data

- Classification of complete data is a very simple case of MPE inference
  - To illustrate, let us use this simpler example with $\boldsymbol{Q} = \{B\}$
  - We select the Markov blanket of B

- Suppose we want to classify the instance
  $\boldsymbol{e}: A = true, C = false, D = true, \mathrm{E} = \mathrm{false}$
  - We can use the chain rule of Bayesian networks to compute the classifications
  - $P(B, \boldsymbol{e}) = P(a)P(B|a)P(d|B,\bar{c})P(\bar{c}|a)P(\bar{e}|\bar{c})$

A	B	$\Theta_{B\|A}$
a	b	.2
a	$\bar{b}$	.8
$\bar{a}$	b	.75
$\bar{a}$	$\bar{b}$	.25



A	Θ_A
a	.6
$\bar{a}$	.4

B	C	D	$\Theta_{D\|B,C}$
b	c	d	.95
b	c	$\bar{d}$	.05
b	$\bar{c}$	d	.9
b	$\bar{c}$	$\bar{d}$	.1
$\bar{b}$	c	d	.8
$\bar{b}$	c	$\bar{d}$	.2
$\bar{b}$	$\bar{c}$	d	.01
$\bar{b}$	$\bar{c}$	$\bar{d}$	.99

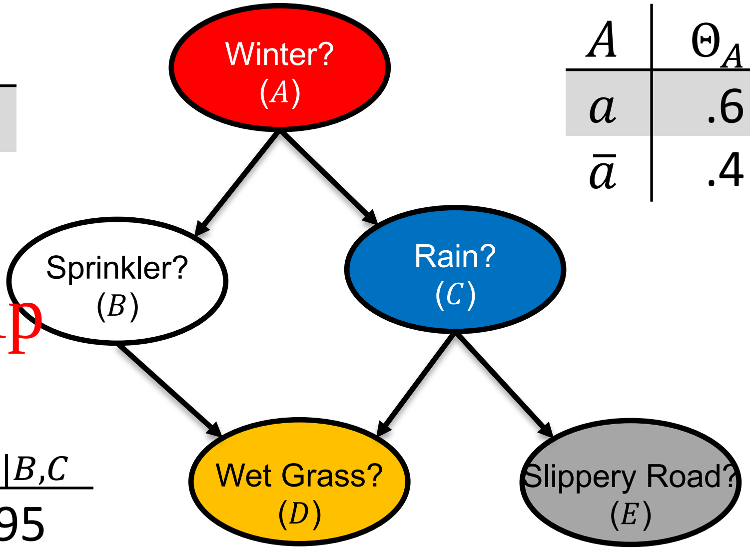
C	E	$\Theta_{E\|C}$
c	e	.7
c	$\bar{e}$	.3
$\bar{c}$	e	0
$\bar{c}$	$\bar{e}$	1

A	C	$\Theta_{C\|A}$
a	c	.8
a	$\bar{c}$	.2
$\bar{a}$	c	.1
$\bar{a}$	$\bar{c}$	.9

Classification of Complete Data

- Since variable B has only two possible values
  - $P(b, \boldsymbol{e}) =$
    $P(a)P(b|a)P(d|b,\bar{c})P(\bar{c}|a)P(\bar{e}|\bar{c}) = .0216$

A	B	$\Theta_{B|A}$
a	b	.2
a	$\bar{b}$	.8
$\bar{a}$	b	.75
$\bar{a}$	$\bar{b}$	.25



A	Θ_A
a	.6
$\bar{a}$	.4

B	C	D	$\Theta_{D|B,C}$
b	c	d	.95
b	c	$\bar{d}$	.05
b	$\bar{c}$	d	.9
b	$\bar{c}$	$\bar{d}$	.1
$\bar{b}$	c	d	.8
$\bar{b}$	c	$\bar{d}$	.2
$\bar{b}$	$\bar{c}$	d	.01
$\bar{b}$	$\bar{c}$	$\bar{d}$	.99

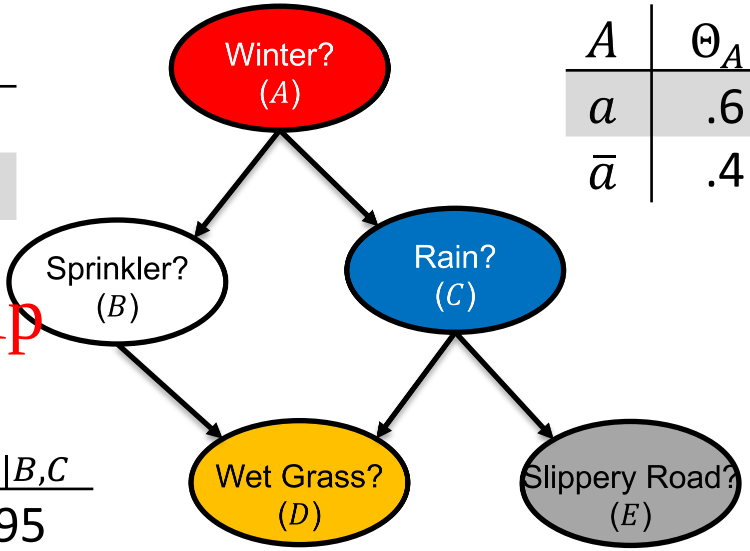
C	E	$\Theta_{E|C}$
c	e	.7
c	$\bar{e}$	.3
$\bar{c}$	e	0
$\bar{c}$	$\bar{e}$	1

A	C	$\Theta_{C|A}$
a	c	.8
a	$\bar{c}$	.2
$\bar{a}$	c	.1
$\bar{a}$	$\bar{c}$	.9

Classification of Complete Data

- Since variable B has only two possible values
  - $P(b, \boldsymbol{e}) =$
    $P(a)P(b|a)P(d|b,\bar{c})P(\bar{c}|a)P(\bar{e}|\bar{c}) = .0216$
  - $P(\bar{b}, \boldsymbol{e}) =$
    $P(a)P(\bar{b}|a)P(d|\bar{b},\bar{c})P(\bar{c}|a)P(\bar{e}|\bar{c}) = .00096$
- Although all variables in Markov(B) influence $P(B, e)$
  - Just the CPTs that include B will be determinant to the final computation
  - We can visualise this using network pruning technique

A	B	$\Theta_{B\|A}$
a	b	.2
a	$\bar{b}$	.8
$\bar{a}$	b	.75
$\bar{a}$	$\bar{b}$	.25



A	Θ_A
a	.6
$\bar{a}$	.4

B	C	D	$\Theta_{D\|B,C}$
b	c	d	.95
b	c	$\bar{d}$	.05
b	$\bar{c}$	d	.9
b	$\bar{c}$	$\bar{d}$	.1
$\bar{b}$	c	d	.8
$\bar{b}$	c	$\bar{d}$	.2
$\bar{b}$	$\bar{c}$	d	.01
$\bar{b}$	$\bar{c}$	$\bar{d}$	.99

C	E	$\Theta_{E\|C}$
c	e	.7
c	$\bar{e}$	.3
$\bar{c}$	e	0
$\bar{c}$	$\bar{e}$	1

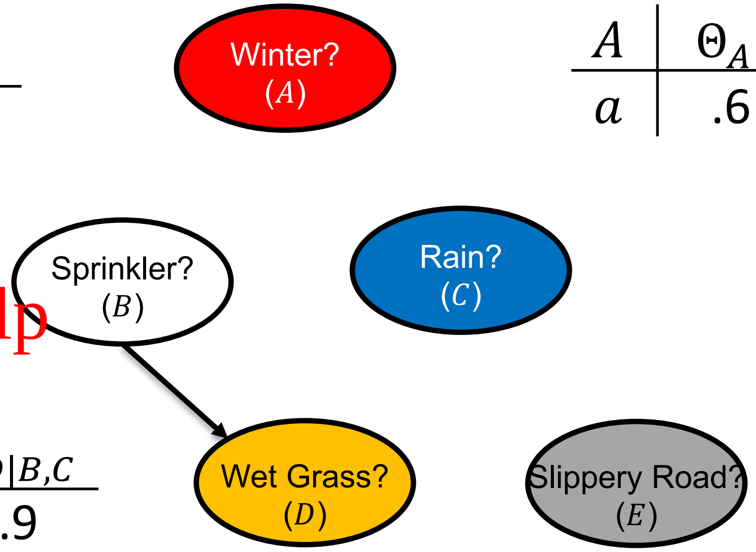
A	C	$\Theta_{C\|A}$
a	c	.8
a	$\bar{c}$	.2
$\bar{a}$	c	.1
$\bar{a}$	$\bar{c}$	.9

Classification of Complete Data

- Consider the query $P(B, \boldsymbol{e})$
  $\boldsymbol{e}: A = true, C = false, D = true, E = false$
  - The pruned network is shown on the right
  - If we remove the disconnected nodes, we end up with the compacted CPTs in nodes B and D

A	B	$\Theta_{B\|A}$
a	b	.2
a	$\bar{b}$	.8

A	Θ_A
a	.6



B	C	D	$\Theta_{D\|B,C}$
b	$\bar{c}$	d	.9
$\bar{b}$	$\bar{c}$	d	.01

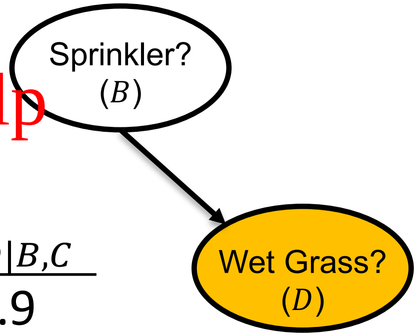
C	E	$\Theta_{E\|C}$
$\bar{c}$	$\bar{e}$	1

A	C	$\Theta_{C\|A}$
a	$\bar{c}$	.2

Classification of Complete Data

- Consider the query $P(B, \boldsymbol{e})$
  $\boldsymbol{e}: A = true, C = false, D = true, E = false$
  - The pruned network is shown on the right
  - If we remove the disconnected nodes, we end up with the compacted CPTs in nodes B and D

A	B	$\Theta_{B\|A}$
a	b	.2
a	$\bar{b}$	.8



B	C	D	$\Theta_{D\|B,C}$
b	$\bar{c}$	d	.9
$\bar{b}$	$\bar{c}$	d	.01

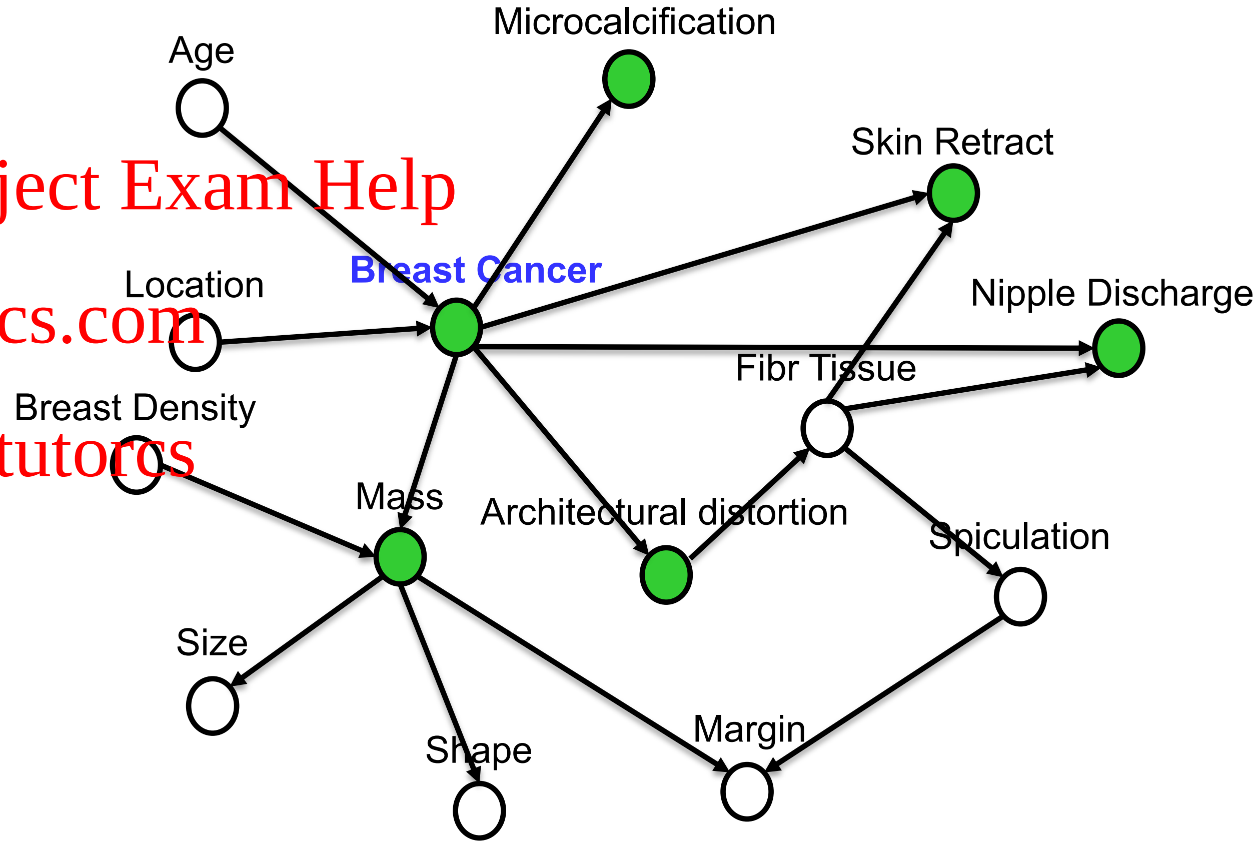
- For classification, we need to compute $argmax_b P(B|\boldsymbol{e})$
  - $argmax_b P(B|\boldsymbol{e}) = argmax_b \frac{P(B,\boldsymbol{e})}{P(\boldsymbol{e})} = argmax_b P(B, \boldsymbol{e})$

$$P(b, \boldsymbol{e}) = P(b|a)P(d|b, \bar{c}) = (.2)(.9) = .18$$
$$P(\bar{b}, \boldsymbol{e}) = P(\bar{b}|a)P(d|\bar{b}, \bar{c}) = (.8)(.01) = .008$$

$$P(b|\boldsymbol{e}) = \frac{.18}{.18 + .008} = .9574$$

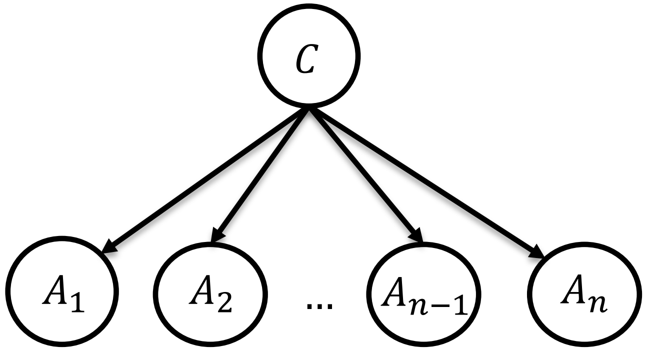
Classification of Complete Data

- Let us look back to the breast cancer network
  - Only a relatively small subset of CPTs are used for classification with complete data
  - Other methods that use all attributes may make a better use of the information
- Remember out discussion is restricted to inference with complete case
  - If some evidence is missing, then more nodes may take part of the inference
  - Our inference procedure is the same as VE_PR, but we can develop a specialised algorithm
  - VE_PR handles both complete case and missing data



Naïve Bayes (NBC)

- A different approach for classification is to use a fixed structure
  - In a Naïve Bayes classifier each attribute has the class variable as its only parent
  - As the structure is fixed, the only task involved in learning is to estimate the parameters
- NBC assumes the attributes are independent given the class
  - This is rarely the case, but NBCs are surprisingly precise
  - In classification, we are often only interested in the class of maximal probability and not in the exact probability distribution
  - They are popular models in some areas such as text classification



Spam Filter

- The task is to receive an email as input and output spam/ham

- Possible attributes
  - Words, e.g., “medicine”, “million”, “dollars”
  - Patterns, such as $?\d+ for currency
  - Non-text: sender in contact list, list of spam servers

- We need a “corpus”, i.e., a collection of emails
  - The documents must be labelled (manual task)
  - We want to be successful to label unseen emails

> Dear colleagues,
> It is with much excitement that I am writing to let you know that nominations are now open for the NSW International Student Awards 2020.

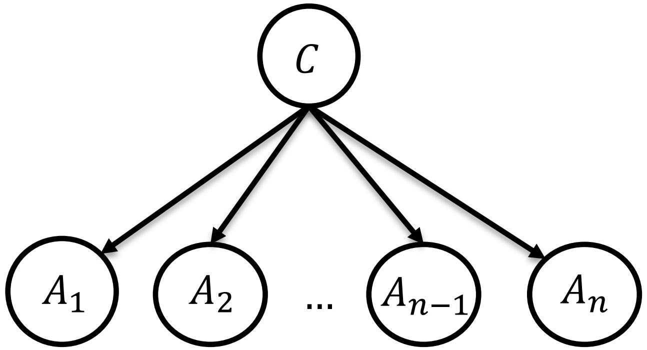
> Now, contact my secretary in Burkina Faso. Ask him to send you the total of $850,000.00 which I kept for your compensation for all the past efforts and attempts to assist me in this matter…

> My name is Mrs. Keiko Awaji, I’m from Hiroshima City in Japan. Please get back to me urgently for full details, Let discuss about charity project in your location (e.g. Less privileged people, the Orphanage home)

?

Naïve Bayes Model

- Using the chain rule for Bayesian networks, we get



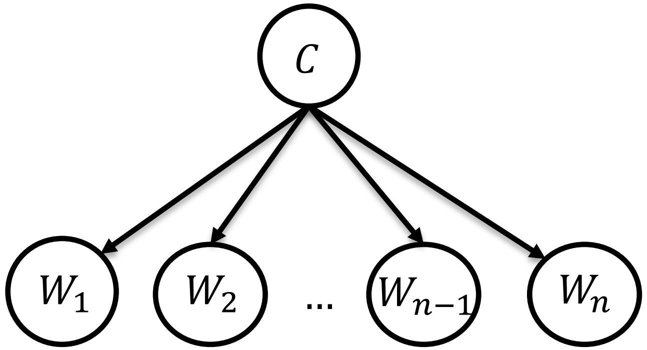
$$P(C, A_1, \dots, A_m) = P(C)P(A_1|C) \dots P(A_n|C)$$
$$= P(C) \prod_i P(A_i|C)$$

$|C||A|^n$ parameters $\quad$ $|C| + n|C||A|$ parameters

- The number of parameters is linear in n
  - If we use the rule of thumb of 10 instances per parameter
  - $n = 10$ and binary variables would require $20{,}480$ versus 400
  - $n = 50$ and binary variables would require $2{\times}10^{17}$ versus $2{,}000$

Naïve Bayes for Text

- NBC for text often use the *bag-of-words model*
  - Attribute W_i is the word at position i in the document
  - However, we assume each W_i is identically distributed, independently of i
  - This model accounts for the same word occurring multiple times
  - "Bag of words" because the model is insensitive to word order



$P(C)$

ham	0.66
spam	0.33

$P(W|\text{spam})$

the	0.0156
to	0.0153
and	0.0115
of	0.0095
you	0.0093
a	0.0086
with	0.0080
from	0.0075
...	

$P(W|\text{ham})$

the	0.0210
to	0.0133
of	0.0119
2002	0.0110
with	0.0108
from	0.0107
and	0.0105
a	0.0100
...	

Example from Berkeley AI Course

Spam Example

Word	P(w|spam)	P(w|ham)	Tot Spam	Tot Ham
(prior)	0.33333	0.66666	-1.1	-0.4
Gary	0.00002	0.00021	-11.8	-8.9
would	0.00069	0.00084	-19.1	-16.0
you	0.00881	0.00304	-23.8	-21.8
like	0.00086	0.00083	-30.9	-28.9
to	0.01517	0.01339	-35.1	-33.2
lose	0.00008	0.00002	-44.5	-44.0
weight	0.00016	0.00002	-53.3	-55.0
while	0.00027	0.00027	-61.5	-63.2
you	0.00881	0.00304	-66.2	-69.0
sleep	0.00006	0.00001	-76.0	-80.5

P(spam | w) = 98.9

Example from Berkeley AI Course

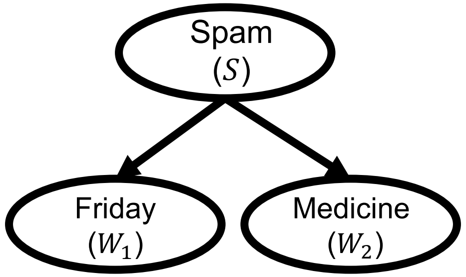
Parameter Estimation

- The parameter of a Bayesian network can be estimated
  - Through *elicitation*, which is the process of asking a human
  - Empirically using data (Machine Learning approach)

- We can define an empirical distribution $P_{\mathcal{D}}$
  - According to this distribution, the empirical probability of an instantiation is simply its frequency of occurrence
  - We can estimate parameters based on the empirical distribution

$$P_{\mathcal{D}}(w_1|s) = \frac{P_{\mathcal{D}}(w_1, s)}{P_{\mathcal{D}}(s)} = \frac{2/16}{12/16} = \frac{1}{6}$$

Case	S	W_1	W_2
1	T	F	T
2	T	F	T
3	F	T	F
4	F	F	T
5	T	F	F
6	T	F	T
7	F	F	F
8	T	F	T
9	T	F	T
10	F	F	T
11	T	F	T
12	T	T	T
13	T	F	T
14	T	T	T
15	T	F	T
16	T	F	T



S	W_1	W_2	$P_{\mathcal{D}}(.)$
T	T	T	2/16
T	T	F	0/16
T	F	T	9/16
T	F	F	1/16
F	T	T	0/16
F	T	F	1/16
F	F	T	2/16
F	T	F	1/16

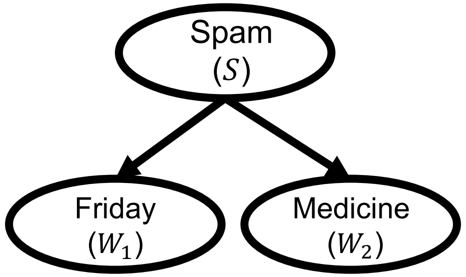
Overfitting

- Our objective is to classify unseen instances
  - We say a model "generalises" to unseen data
  - A common procedure is to split the data into training and test sets

- However, frequency parameters tend to overfit the training data
  - Some words may only appear in one of the classes in the training set. Such as "medicine" for spam and "indeed" for ham
  - Several words may not occur in the training set, but they may appear in the test set
  - In general, we should avoid assigning zero probabilities for any event, unless we are completely sure

$$P(C, W_1, \dots, W_n) = P(C) \prod_i P(W_i|C)$$

Additive Smoothing



- Also known as *Laplacian smoothing*
  - Developed by Laplace when he tried to estimate the chance the sum will rise tomorrow

$$P_L = \frac{c(X = x) + \alpha}{N + \alpha|X|}$$

  - where $c(X = x)$ is the number of occurrences of $X = x$
  - $\alpha \geq 0$ is a "pseudo count" parameter
  - N is the number of instances and
  - $|X|$ is the number of values for X

S	W_1	W_2	$P_{\mathcal{D}}(.)$
T	T	T	2/16
T	T	F	0/16
T	F	T	9/16
T	F	F	1/16
F	T	T	0/16
F	T	F	1/16
F	F	T	2/16
F	T	F	1/16

- For example, $\alpha = 1$ (weak)

$$P_L(s, w_1, w_2) = \frac{2+1}{16+8} = \frac{3}{24} \approx .125$$

$$P_L(s, w_1, \overline{w}_2) = \frac{0+1}{16+8} = \frac{1}{24} \approx .041$$

- $\alpha = 1000$ (strong)

$$P_L(s, w_1, w_2) = \frac{2+1000}{16+8000} = \frac{1002}{8016} \approx .125$$

$$P_L(s, w_1, \overline{w}_2) = \frac{0+1000}{16+8000} = \frac{1000}{8016} \approx .1248$$

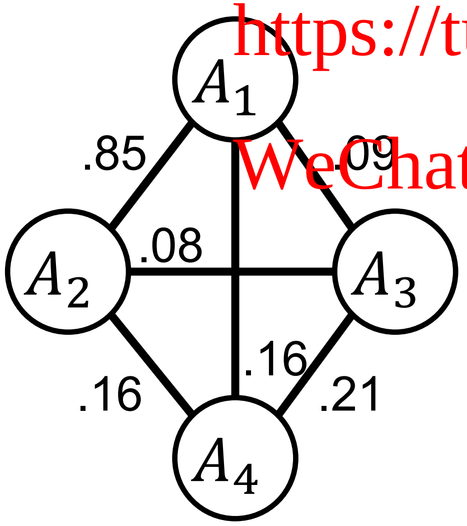
Naïve Bayes Extensions

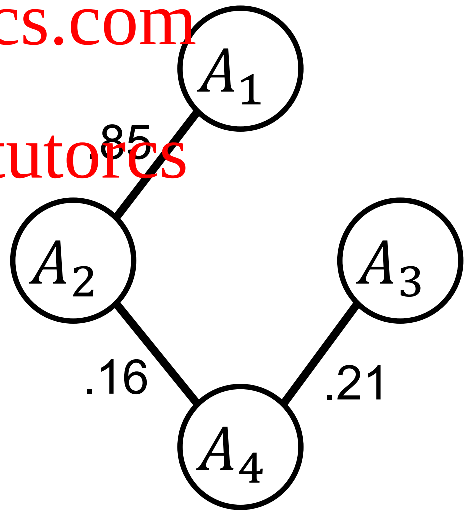
- Several NBC extensions have been proposed
  - Most of them, try to address the assumption of conditional independence of attributes given the class
- A well-known extension is the *Tree-augmented Bayes classifier* (TAN)
  - It allows a more elaborate dependency structure among variables
  - Such structure is not predefined, as in the case of NBCs
  - Tree means that each attribute variables has at most one attribute variable as parent
- The central idea is to use conditional mutual information (MI) to link attributes
  - Conditional MI can be seen as a measure of dependency of attributes (given the class)

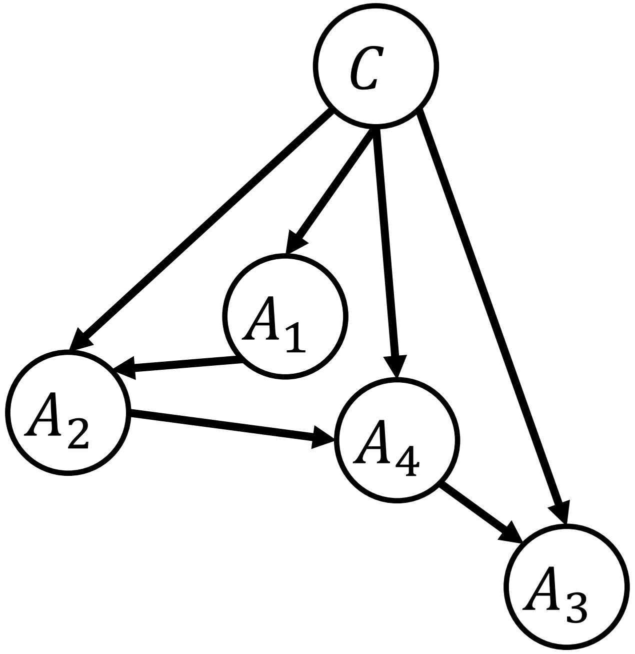
Tree-augmented Bayes Classifier

$$MI_{\mathcal{D}}(A_i, A_j|C) \stackrel{\text{def}}{=} \sum_{a_i, a_j, c} P_{\mathcal{D}}(a_i, a_j, c) \log_2 \frac{P_{\mathcal{D}}(a_i, a_j|c)}{P_{\mathcal{D}}(a_i|c) P_{\mathcal{D}}(a_j|c)}$$

$\mathcal{D}$	A_1	A_2	A_3	A_4	C
1	a_1	a_2	$\overline{a_3}$	a_4	$\bar{c}$
2	a_1	$\overline{a_2}$	a_3	a_4	c
3	a_1	$\overline{a_2}$	a_3	a_4	c
4	$\overline{a_1}$	a_2	a_3	a_4	c
5	a_1	a_2	a_3	$\overline{a_4}$	$\bar{c}$
6	a_1	$\overline{a_2}$	a_3	a_4	c
7	a_1	a_2	$\overline{a_3}$	$\overline{a_4}$	$\bar{c}$
8	$\overline{a_1}$	$\overline{a_2}$	$\overline{a_3}$	a_4	$\bar{c}$
9	$\overline{a_1}$	a_2	a_3	a_4	c
10	a_1	a_2	a_3	$\overline{a_4}$	$\bar{c}$



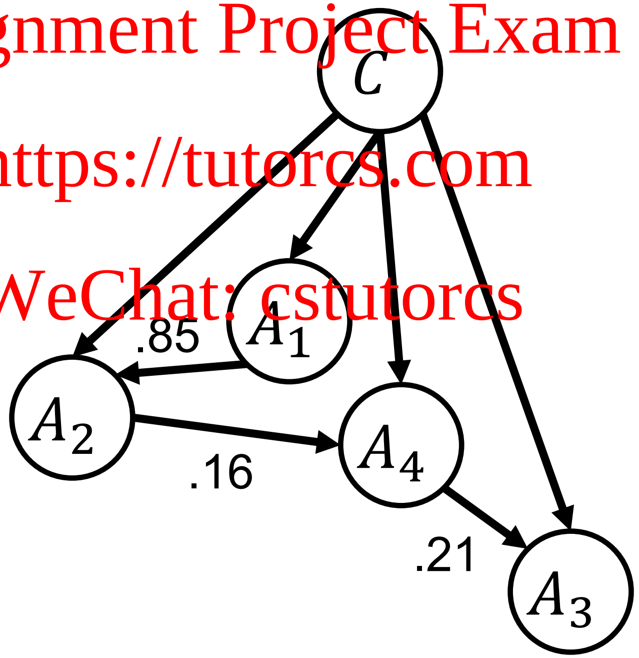


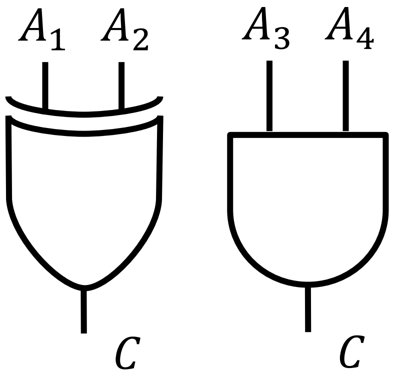


Tree-augmented Bayes Classifier

$$MI_{\mathcal{D}}(A_i, A_j|C) \stackrel{\text{def}}{=} \sum_{a_i, a_j, c} P_{\mathcal{D}}(a_i, a_j, c) \log_2 \frac{P_{\mathcal{D}}(a_i, a_j|c)}{P_{\mathcal{D}}(a_i|c) P_{\mathcal{D}}(a_j|c)}$$

$\mathcal{D}$	A_1	A_2	A_3	A_4	C
1	a_1	a_2	$\overline{a_3}$	a_4	$\bar{c}$
2	a_1	$\overline{a_2}$	a_3	a_4	c
3	a_1	$\overline{a_2}$	a_3	a_4	c
4	$\overline{a_1}$	a_2	a_3	a_4	c
5	a_1	a_2	a_3	$\overline{a_4}$	$\bar{c}$
6	a_1	$\overline{a_2}$	a_3	a_4	c
7	a_1	a_2	$\overline{a_3}$	$\overline{a_4}$	$\bar{c}$
8	$\overline{a_1}$	$\overline{a_2}$	$\overline{a_3}$	a_4	$\bar{c}$
9	$\overline{a_1}$	a_2	a_3	a_4	c
10	a_1	a_2	a_3	$\overline{a_4}$	$\bar{c}$





$$P(C, A_1, A_2, A_3, A_4) = P(C)P(A_1|C)P(A_2|A_1, C)P(A_3|A_4, C)P(A_4|A_2, C)$$

Tree-augmented Bayes Classifier

Input: Dataset D with attributes $A_1, \dots, A_n$ and class C

Output: TAN classifier

for $i = 1$ to n **do**

  for j $= 1$ to n **do**

    $m[i,j] \leftarrow MI(A_i, A_j | C)$

$G \leftarrow$ complete undirected graph over $\{A_1, \dots, A_n\}$ with weight m

$G_T \leftarrow$ maximal spanning tree for G

$G_T^D \leftarrow G_T$ directed by choosing any variable as root and setting edge directions outward from root

$G_T^D \leftarrow G_T^D$ with node C added and direct edges from C to each attribute node

Learn parameters for G_T^D

return G_T^D

Other Naïve Bayes Extensions

- Other extensions to the NBC are
  - Bayesian Network augmented Naïve Bayes (BAN)
  - General Bayesian Network (GBN)
- Both GBN and BAN are very similar to TAN
  - With the main difference they induce DAG structures from data, instead of trees
  - BANs create the network structure using only the attributes. The class is included afterwards, similarly to TAN
  - GBNs create the structure with all variables including the class. It finds the Markov blanket of the class and delete all nodes outside the blanket
- We will study the algorithms to induce DAG structures from data later on in the course

Comparison of Classification Accuracy

- This are the results of an empirical comparison of Bayesian classifiers in eight UCI dataset

Dataset	GBN	BAN	TAN	NBC	GBN (Sel. At.)
Adult	**86.11±0.27**	85.82±0.27	86.01±0.27	84.18±0.29	8/13
Nursery	89.72±0.46	**93.08±0.39**	91.71±0.42	90.32±0.45	6/8
Mushroom	99.30±0.16	**100**	99.82±0.08	95.75±0.39	5/22
Chess	**94.65±0.69**	94.18±0.72	92.50±0.81	87.34±1.02	19/36
DNA	79.09±1.18	88.28±0.93	93.59±0.71	**94.27±0.68**	43/60
Car	86.11±1.46	94.04±0.44	**94.10±0.48**	86.58±1.78	5/6
Flare	82.27±1.45	82.85±2.00	**83.49±1.29**	80.11±3.14	1-3/10
Vote	95.17±1.89	**95.63±3.85**	94.25±3.63	89.89±5.29	10-11/16

Cheng, J., & Greiner, R. Comparing Bayesian network classifiers. *UAI.*

Conclusion

- Bayesian networks are models for probabilistic reasoning
  - Classification is a task that matches MAP/MPE queries
- BNs provides an attractive approach
  - They can naturally learn (more about this later) and classify in the presence of missing data
  - For complete data, the Markov blanket provides an approach to select the most relevant features
- NBC is Bayesian classification algorithm with fixed structure
  - They are very popular in certain areas such as text mining
  - Some NBC extensions induce more complex structures, usually leading to better accuracy rates