

# Statistical machine learning

where

*Data* are evidence (instantiations of random variables)

*Hypotheses* are probabilistic theories of how the domain works

Chapters 18 + 20

## Overview: topics

- Review: Uncertain knowledge, Bayes' rule
- Bayesian learning
- Maximum a posteriori (MAP) and Maximum likelihood (ML)
- Bayesian networks
- Naïve Bayes (including a worked example)

## Bayes' rule: Review (1)

Bayes' rule: 
$$P(\beta | \alpha) = \frac{P(\alpha | \beta)P(\beta)}{P(\alpha)}$$

Multivalued variables: 
$$P(Y | X) = \frac{P(X | Y)P(Y)}{P(X)}$$

Given evidence, e: 
$$P(Y | X, e) = \frac{P(X | Y, e)P(Y | e)}{P(X | e)}$$

## Overview: aims

- Understand the application of Bayes' rule for learning from data and that all available hypotheses are considered
- Understand the Naïve Bayes classifier so well that you could implement it

## Uncertain knowledge: Review

- Prior probability
  - $P(\alpha)$  is the probability of a proposition  $\alpha$  in the absence of any other information
- Joint probability distribution
  - A full joint distribution covers the complete set of random variables
- Conditional probability
  - $P(\alpha|\beta)$  is the probability of  $\alpha$  given that all we know is  $\beta$
- Independence
  - If variables are independent, knowledge of one does not influence knowledge of another

## Bayes' rule: Review (2)

- Classification
- Diagnosis
- Finding one of the terms when you have the other three

$$P(\beta | \alpha) = \frac{P(\alpha | \beta)P(\beta)}{P(\alpha)}$$

## Bayesian learning (1)

- *Data* are evidence (instantiations of the random variables that describe the domain)
- *Hypotheses* are probabilistic theories of how the domain works

## Bayesian learning (3)

- Assume we have (or choose) a finite number of hypotheses indexed by  $i$
- Let  $D$  be a random variable representing the whole dataset and call the actual (observed) dataset  $\mathbf{d}$
- Standard statistics notation: variable in upper case, observations of it in lower case
- Then the posterior probability of any hypothesis,  $h_i$ , is given by: (using Bayes' rule)

$$P(h_i | \mathbf{d}) = \frac{P(\mathbf{d} | h_i)P(h_i)}{P(\mathbf{d})}$$

## Bayesian learning (5)

$$P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i)P(h_i)$$

- $P(h_i | \mathbf{d})$ : *likelihood* of the hypothesis  $h_i$  given the set of data  $\mathbf{d}$
- $P(\mathbf{d} | h_i)$ : *likelihood* of the data  $\mathbf{d}$  given hypothesis  $h_i$ .
  - We should be able to calculate the likelihood for almost any hypothesis
- $P(h_i)$ : *prior* probability of hypothesis  $h_i$  (without seeing any data)
  - We have to work this out from prior knowledge or experience
  - If we don't have any, we might make it the same for each hypothesis

## Bayesian learning (2)

- Calculates the probability of each available hypothesis (model or input to output function), given the data
- Predicts output using *all* available hypotheses, weighted by their probabilities – not just using a *single* hypothesis (as in e.g. ID3 decision trees)
- Learning is thus reduced to probabilistic inference
- A limited set of hypotheses are considered

## Bayesian learning (4)

$$P(h_i | \mathbf{d}) = \frac{P(\mathbf{d} | h_i)P(h_i)}{P(\mathbf{d})} \propto P(\mathbf{d} | h_i)P(h_i)$$

$$P(\mathbf{d}) = \sum_i P(\mathbf{d} | h_i)P(h_i)$$

A normalising constant which is the same for all hypotheses – it ensures that all the probabilities add up to 1

## Bayesian learning: Candy bags (1)

- Very large candy bags, either
  - $h_1$ : 100% cherry
  - $h_2$ : 75% cherry and 25% lime
  - $h_3$ : 50% cherry and 50% lime
  - $h_4$ : 25% cherry and 75% lime
  - $h_5$ : 100% lime
- These hypotheses are discrete distributions
- Prior probabilities (here we know them)
  - $P(h_1)=0.10$ ,  $P(h_2)=0.20$ ,  $P(h_3)=0.40$ ,  $P(h_4)=0.20$ ,  $P(h_5)=0.10$
- Task: What kind of bag do we have ?

## Bayesian learning: Candy bags (2)

- $P(h_1)=0.10$
- $P(h_2)=0.20$
- $P(h_3)=0.40$
- $P(h_4)=0.20$
- $P(h_5)=0.10$ 
  - If have no data, choose  $h_3$  – it's the most likely
- Data: observing the candy in the bag by opening wrappers one by one
- Task: What kind of bag do we have?
  - Choose the correct hypothesis given some data

## Bayesian learning: Candy bags (3)

- Assume the observations (opening wrappers one by one) are *independently and identically distributed (i.i.d.)* ie: the probabilities for each candy observation don't depend on the previous ones and don't change

$$P(\alpha \wedge \beta) = P(\alpha | \beta)P(\beta)$$

Assumption

$$P(\alpha | \beta) = P(\alpha)$$

$$P(\alpha \wedge \beta) = P(\alpha)P(\beta)$$

- So likelihood becomes:  $P(\mathbf{d} | h_i) = \prod_j P(d_j | h_i)$

## Candy bags: What kind of bag do we have? (1)

- $P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i) P(h_i)$
- We have  $P(h_i)$
- Calculate  $P(\mathbf{d} | h_i)$  on the basis of observed data

h1: 100% cherry	$P(h1)=0.10$
h2: 75% cherry and 25% lime	$P(h2)=0.20$
h3: 50% cherry and 50% lime	$P(h3)=0.40$
h4: 25% cherry and 75% lime	$P(h4)=0.20$
h5: 100% lime	$P(h5)=0.10$

## Candy bags: What kind of bag do we have? (2)



$$P(\mathbf{d} | h_1) = P(\text{lime} | h_1) = 0.00$$

$$P(\mathbf{d} | h_2) = P(\text{lime} | h_2) = 0.25$$

$$P(\mathbf{d} | h_3) = P(\text{lime} | h_3) = 0.50$$

$$P(\mathbf{d} | h_4) = P(\text{lime} | h_4) = 0.75$$

$$P(\mathbf{d} | h_5) = P(\text{lime} | h_5) = 1.00$$

$$P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i) P(h_i)$$

## Candy bags: What kind of bag do we have? (3)



$$P(\mathbf{d} | h_1) = P(\text{lime} | h_1)P(\text{lime} | h_1) = 0.00$$

$$P(\mathbf{d} | h_2) = P(\text{lime} | h_2)P(\text{lime} | h_2) = 0.06$$

$$P(\mathbf{d} | h_3) = P(\text{lime} | h_3)P(\text{lime} | h_3) = 0.25$$

$$P(\mathbf{d} | h_4) = P(\text{lime} | h_4)P(\text{lime} | h_4) = 0.56$$

$$P(\mathbf{d} | h_5) = P(\text{lime} | h_5)P(\text{lime} | h_5) = 1.00$$

$$P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i) P(h_i)$$

## Candy bags: What kind of bag do we have? (4)



$$P(\mathbf{d} | h_1) = P(\text{lime} | h_1)P(\text{lime} | h_1)P(\text{lime} | h_1) = 0.00$$

$$P(\mathbf{d} | h_2) = P(\text{lime} | h_2)P(\text{lime} | h_2)P(\text{lime} | h_2) = 0.02$$

$$P(\mathbf{d} | h_3) = P(\text{lime} | h_3)P(\text{lime} | h_3)P(\text{lime} | h_3) = 0.13$$

$$P(\mathbf{d} | h_4) = P(\text{lime} | h_4)P(\text{lime} | h_4)P(\text{lime} | h_4) = 0.42$$

$$P(\mathbf{d} | h_5) = P(\text{lime} | h_5)P(\text{lime} | h_5)P(\text{lime} | h_5) = 1.00$$

$$P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i) P(h_i)$$

**Candy bags:**  
**What kind of bag do we have? (5)**



$$P(\mathbf{d} | h_1) = [P(\text{lime} | h_1)]^5 = 0.00$$

$$P(\mathbf{d} | h_2) = [P(\text{lime} | h_2)]^5 = 0.00$$

$$P(\mathbf{d} | h_3) = [P(\text{lime} | h_3)]^5 = 0.03$$

$$P(\mathbf{d} | h_4) = [P(\text{lime} | h_4)]^5 = 0.24$$

$$P(\mathbf{d} | h_5) = [P(\text{lime} | h_5)]^5 = 1.00$$

$$P(h_i | \mathbf{d}) \propto P(\mathbf{d} | h_i)P(h_i)$$

**Bayesian learning:**  
**Candy bags (4)**

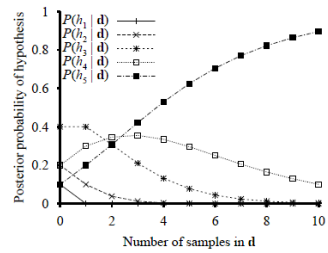
- Task 2: predict the next observation  $d'$
- Note that for Bayesian learning, all hypotheses are being used to predict

**Candy bags:**  
**Predict  $d'$  (2)**

$$P(d' | \mathbf{d}) \propto \sum_i P(d' | h_i)P(h_i) \prod_{j=1}^n P(d_j | h_i)$$

- We can calculate all these terms
- E.g. after  $d=5$  lime candies,  
 $P(d'=\text{lime}|\mathbf{d}) \propto 0(.1)^0 + .25(.2)(.25)^5 + .5(.4)(.5)^5 + .75(.2)(.75)^5 + 1(.1)^5 = 0.142$
- $P(d'=\text{cherry}|\mathbf{d}) \propto 1(.1)^0 + .75(.2)(.25)^5 + .5(.4)(.5)^5 + .25(.2)(.75)^5 + 0(.1)^5 = 0.018$
- Can ignore normalising: lime is our prediction
- If normalising:  $P(d'=\text{lime}|\mathbf{d}) = 0.142 / (0.142 + 0.018) = 0.888$
- Hence  $P(d'=\text{cherry}|\mathbf{d}) = 1 - 0.888 = 0.112$

**Candy bags:**  
**What kind of bag do we have? (6)**



- $d = 0$  to 10 lime candy observations

$$P(h_i | \mathbf{d}) = \alpha P(\mathbf{d} | h_i)P(h_i)$$

**Candy bags:**  
**Predict  $d'$  (1)**

$$P(d' | \mathbf{d}) = \sum_i P(d', h_i | \mathbf{d})$$

$$= \sum_i P(d' | h_i, \mathbf{d})P(h_i | \mathbf{d})$$

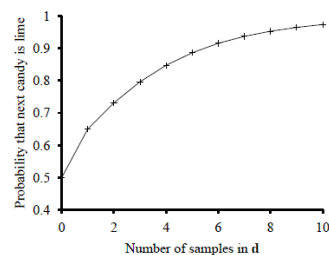
$$= \sum_i P(d' | h_i)P(h_i | \mathbf{d})$$

$$\propto \sum_i P(d' | h_i)P(h_i) \prod_{j=1}^n P(d_j | h_i)$$

- Each hypothesis  $h_i$  determines a probability distribution over possible outcomes like  $d'$
- The result is a weighted average over hypothesis-specific predictions

**Candy bags:**  
**Predict  $d'$  (3)**

$$P(d' | \mathbf{d}) \propto \sum_i P(d' | h_i)P(h_i) \prod_{j=1}^n P(d_j | h_i)$$



## Bayesian learning

- If we have the full set of possible hypotheses and the correct prior probability for each, Bayesian prediction makes the optimal decision (regardless of the number of data points)
- We can get the predicted quantity and its estimated probability, given the data, and also get the (estimated) probabilities of each hypothesis
- But the space of hypotheses is usually very large – it can be useful to use approximations
  - Maximum a posteriori (MAP)
  - Maximum likelihood (ML)

## Maximum likelihood (ML)

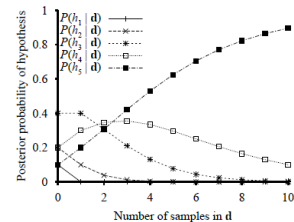
- Assume a uniform prior on hypotheses, and ignore the prior
  - In general,  $h_{MAP}$  becomes  $h_{ML}$  as more data is collected and the impact of the prior fades

## Bayesian networks (1) (Chapter 14)

- Full joint probability distribution can answer any question about the domain
- Independence and conditional independence relationships between variables reduces the probabilities that need to be specified
- Bayesian networks are a way to represent the independence and conditional independence relationships between variables

## Maximum a posteriori (MAP)

- Make predictions based on the most probable hypothesis only
  - Notes on example:  $h_{MAP}$  becomes  $h_5$  after only three observations, but e.g.  $P(d^1|d) \approx P(d^1|h_5)$  takes many more observations



## Parameter learning

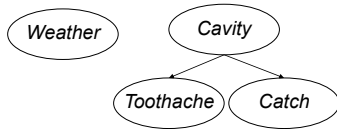
- Predict a class variable based on attribute variables
- Naïve Bayes classifier
- (Bayesian networks)

## Bayesian networks (2)

- Independence
  - If variables are independent, knowledge of one does not affect knowledge of another
- Conditional independence
  - Variables are conditionally independent if they are independent given the presence or absence of another variable

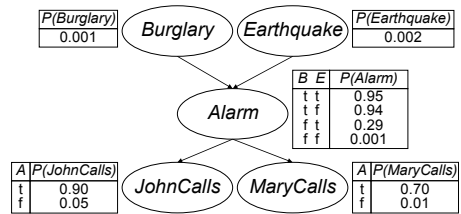
### Bayesian networks (3)

- Directed acyclic graph where each node has quantitative probability information
- Nodes are the set of random variables, and are connected by links that specify influences



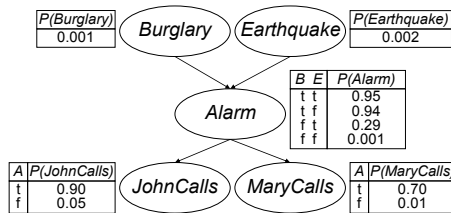
### Bayesian networks (4)

- Conditional probabilities can be defined for each



### Bayesian networks (5)

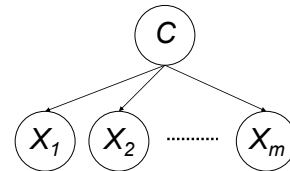
- What is the probability that a burglary has occurred but not an Earthquake, the alarm has sounded, and both John and Mary have called?



$$P(B \wedge \neg E \wedge A \wedge J \wedge M) = P(B)P(\neg E)P(A|B \wedge \neg E)P(J|A)P(M|A)$$

### Bayesian networks: Naïve Bayes

- Bayesian network model in which the *Class* variable *C* (to be predicted) is the root and the *Attribute* variables *X<sub>i</sub>* are the leaves
- Naïve: assumption that attributes are conditionally independent, given the class



### Naïve Bayes model

- We wish to predict a class variable *C* using attribute variables *X<sub>i</sub>*, *i*=1,...,*m*.
- Assume we know all the possible classes - usually only a small number e.g. 2-30.
- Assume we have some training data (index *j*=1,...,*n*) from which we can estimate, e.g. *P(X<sub>i</sub>|C)* and *P(C)*.
- The estimation of class probabilities is naïve because it assumes that all the attributes are conditionally independent of each other

$$P(C | X_1, X_2, \dots, X_m) = \alpha P(C) \prod_{i=1}^m P(X_i | C)$$

### Naïve Bayes: using training data (1)

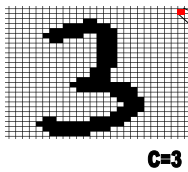
- Assume we have a training dataset *d*={*x<sub>j</sub>*, *c<sub>j</sub>*, *j*=1,...,*n*} of *n* observations, where each *x<sub>j</sub>* = {*x<sub>ij</sub>*, *i*=1,...,*m*} is a vector of *m* attribute values *x<sub>ij</sub>*, and *c<sub>j</sub>* is the correct label (class) for *x<sub>j</sub>* (e.g. given by a human expert).
- If we get a new input vector *x<sub>new</sub>* = {*x<sub>i, new</sub>*} and want to predict its class *c<sub>new</sub>*, how is this done with Naïve Bayes?

## Naïve Bayes: using training data (2)

- Calculate the posterior probabilities for each possible class and choose the most likely. All quantities on the last row can be estimated from the training data. The  $d$  part is sometimes not mentioned.

$$\begin{aligned}
 &P(c_{new} | x_{new}, d) \\
 &\propto P(x_{new} | c_{new}, d)P(c_{new} | d) \\
 &\propto P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)
 \end{aligned}$$

## Naïve Bayes example: Number recognition (2)



For this example  $j$ ,  $b_{x,y}$  is on or off, i.e. add 0 or 1 to the count for each pixel per example.

$$P(C | x_1, x_2, \dots, x_n) = \alpha P(C) \prod_i P(x_i | C)$$

- How to estimate  $P(C)$  and  $P(x_i|C)$  from training data ?

$$\begin{aligned}
 P(C = 3) &= \text{count}(C = 3) / \text{count}(\text{all}) \\
 P(x_i | C = 3) &= \text{count}(x_i \wedge C = 3) / \text{count}(C = 3)
 \end{aligned}$$

## Naïve Bayes example: Restaurant problem (1)

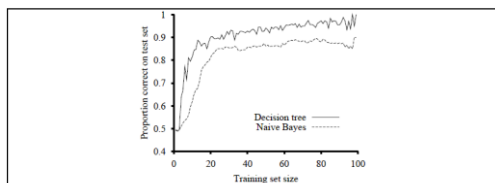
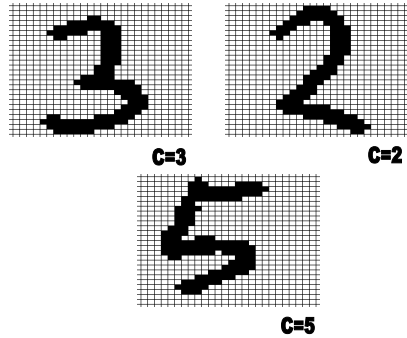
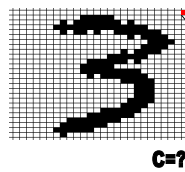


Figure 20.3 The learning curve for naive Bayes learning applied to the restaurant problem from Chapter 18; the learning curve for decision-tree learning is shown for comparison.

## Naïve Bayes example: Number recognition (1)



## Naïve Bayes example: Number recognition (3)



For this test example,  $b_{x,y}$  is on or off, i.e. 0 or 1. Work out  $P(b_{x,y}|C)$  for all  $x,y$  co-ordinates in image; can then predict  $C$  using Naive Bayes model.

- What if training sample counts are zero?
  - smooth distributions, i.e. redistribute some probability
- Conditional independence between input features?
- Problems with continuous input variables?
  - discretise or estimate using standard distributions, e.g. normal (see section 14.3 of textbook)

## Naïve Bayes example: Restaurant problem (2)

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

### Naïve Bayes example: Restaurant problem (3)

- Can construct the model and prediction as needed for each test pattern (11 and 12)
- Alternative is to build full Naïve Bayes model for every test pattern
- Can estimate probabilities from the training data (1-10)

$$P(c_{new} | x_{new}, d) \propto P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

### Naïve Bayes example: Restaurant problem (4)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $c_{new}$ : possible class output for test example
- $x_{new}$ : input variables for test example
- $d$ : dataset of training examples

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	FALSE
12	TRUE	TRUE	Full	Burger	TRUE

### Naïve Bayes example: Restaurant problem (5)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(c_{new} | x_{new}, d)$  values sum to 1 over the possible classes of  $c_{new}$
- Renormalisation of  $P(c_{new} | x_{new}, d)$  values can be performed after un-normalised values are calculated

### Naïve Bayes example: Restaurant problem (6)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{true}) = 5/10 = 0.5$
- $P(\text{WillWait}=\text{false}) = 1 - P(\text{WillWait}=\text{true}) = 0.5$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE

### Naïve Bayes example: Restaurant problem (7)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{Fri/Sat}=\text{true} | \text{WillWait}=\text{true}) = 1/5 = 0.2$
- $P(\text{Fri/Sat}=\text{false} | \text{WillWait}=\text{true}) = 4/5 = 0.8$
- $P(\text{Fri/Sat}=\text{true} | \text{WillWait}=\text{false}) = 3/5 = 0.6$
- $P(\text{Fri/Sat}=\text{false} | \text{WillWait}=\text{false}) = 2/5 = 0.4$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE

### Naïve Bayes example: Restaurant problem (8)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{true} | \text{TestExample}11) = \alpha P(\text{WillWait}=\text{true})$   
 $P(\text{Fri/Sat}=\text{false} | \text{WillWait}=\text{true})$   
 $P(\text{Hungry}=\text{false} | \text{WillWait}=\text{true})$   
 $P(\text{Patrons}=\text{none} | \text{WillWait}=\text{true})$   
 $P(\text{Type}=\text{Thai} | \text{WillWait}=\text{true})$
- $= \alpha(0.5)(4/5)(1/5)(0/5)(2/5)$
- $= \alpha \cdot 0$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	FALSE
12	TRUE	TRUE	Full	Burger	TRUE

## Naïve Bayes example: Restaurant problem (9)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{false} | \text{TestExample11}) = \alpha P(\text{WillWait}=\text{false})$   
 $P(\text{Fri/Sat}=\text{false} | \text{WillWait}=\text{false})$   
 $P(\text{Hungry}=\text{false} | \text{WillWait}=\text{false})$   
 $P(\text{Patrons}=\text{none} | \text{WillWait}=\text{false})$   
 $P(\text{Type}=\text{Thai} | \text{WillWait}=\text{false})$
- $= \alpha(0.5)(2/5)(3/5)(1/5)(1/5)$
- $= \alpha 0.0048$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

## Naïve Bayes example: Restaurant problem (10)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{true} | \text{TestExample11}) = \alpha 0.0$
- $P(\text{WillWait}=\text{false} | \text{TestExample11}) = \alpha 0.0048$
- Predicts false

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

## Naïve Bayes example: Restaurant problem (11)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{true} | \text{TestExample12}) = \alpha P(\text{WillWait}=\text{true})$   
 $P(\text{Fri/Sat}=\text{true} | \text{WillWait}=\text{true})$   
 $P(\text{Hungry}=\text{true} | \text{WillWait}=\text{true})$   
 $P(\text{Patrons}=\text{full} | \text{WillWait}=\text{true})$   
 $P(\text{Type}=\text{Burger} | \text{WillWait}=\text{true})$
- $= \alpha(0.5)(1/5)(4/5)(1/5)(1/5)$
- $= \alpha 0.0032$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

## Naïve Bayes example: Restaurant problem (12)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{false} | \text{TestExample12}) = \alpha P(\text{WillWait}=\text{false})$   
 $P(\text{Fri/Sat}=\text{true} | \text{WillWait}=\text{false})$   
 $P(\text{Hungry}=\text{true} | \text{WillWait}=\text{false})$   
 $P(\text{Patrons}=\text{full} | \text{WillWait}=\text{false})$   
 $P(\text{Type}=\text{Burger} | \text{WillWait}=\text{false})$
- $= \alpha(0.5)(3/5)(2/5)(4/5)(2/5)$
- $= \alpha 0.0384$

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

## Naïve Bayes example: Restaurant problem (13)

$$P(c_{new} | x_{new}, d) = \alpha P(c_{new} | d) \prod_{i=1}^m P(x_{i,new} | c_{new}, d)$$

- $P(\text{WillWait}=\text{true} | \text{TestExample12}) = \alpha 0.0032$
- $P(\text{WillWait}=\text{false} | \text{TestExample12}) = \alpha 0.0384$
- Predicts false

	Fri/Sat	Hungry	Patrons	Type	Will wait?
1	FALSE	TRUE	Some	French	TRUE
2	FALSE	TRUE	Full	Thai	FALSE
3	FALSE	FALSE	Some	Burger	TRUE
4	TRUE	TRUE	Full	Thai	TRUE
5	TRUE	FALSE	Full	French	FALSE
6	FALSE	TRUE	Some	Italian	TRUE
7	FALSE	FALSE	None	Burger	FALSE
8	FALSE	TRUE	Some	Thai	TRUE
9	TRUE	FALSE	Full	Burger	FALSE
10	TRUE	TRUE	Full	Italian	FALSE
11	FALSE	FALSE	None	Thai	(FALSE)
12	TRUE	TRUE	Full	Burger	(TRUE)

## Summary

- Review of uncertain knowledge and Bayes rule
- Bayesian Learning
- Maximum a posteriori (MAP) learning
- Maximum likelihood (ML) learning
- Bayesian networks
- Naïve Bayes