



# Bayesian Methods: Naïve Bayes

Rishabh Iyer  
University of Texas at Dallas

based on the slides of Vibhav Gogate and Nick Rouzzi

# Last Time

- Parameter learning Likelihood:  $P(D|\theta)$ 
  - Learning the parameter of a simple coin flipping model
    - Dice roll problem
- Prior distributions  $P(\theta)$
- Posterior distributions  $P(\theta|D) \propto p(\theta) P(D|\theta)$
- Today: more parameter learning and naïve Bayes

# Maximum Likelihood Estimation (MLE)



- **Data:** Observed set of  $\alpha_H$  heads and  $\alpha_T$  tails
- **Hypothesis:** Coin flips follow a binomial distribution
- **Learning:** Find the “best”  $\theta$
- **MLE:** Choose  $\theta$  to maximize the likelihood (probability of  $D$  given  $\theta$ )

$$\theta_{MLE} = \arg \max_{\theta} p(D|\theta)$$

Coin Flip:  $p(D|\theta) = \theta^{\alpha_H} (1-\theta)^{\alpha_T}$

# Maximum A-Posteriori Estimation

1. Data:  $\alpha_H$  Heads,  $\alpha_T$  Tails
2. Hypothesis: Coin Flip  $\rightarrow$  Binomial Distribution
3. Prior: Choose a prior Dist  $\rightarrow$  Beta Dist
4. Learning: Find best  $\theta$
5. MAP:  $\hat{\theta}^* = \underset{\theta}{\operatorname{argmax}} P(\theta | D) = \underset{\theta}{\operatorname{argmax}} p(\theta) p(D|\theta)$

Beta:

$$p(\theta) \propto \theta^{\beta_H - 1} (1-\theta)^{\beta_T - 1}$$

# MAP Estimation

- Choosing  $\theta$  to maximize the posterior distribution is called maximum a posteriori (MAP) estimation

$$\theta_{MAP} = \arg \max_{\theta} p(\theta|D)$$

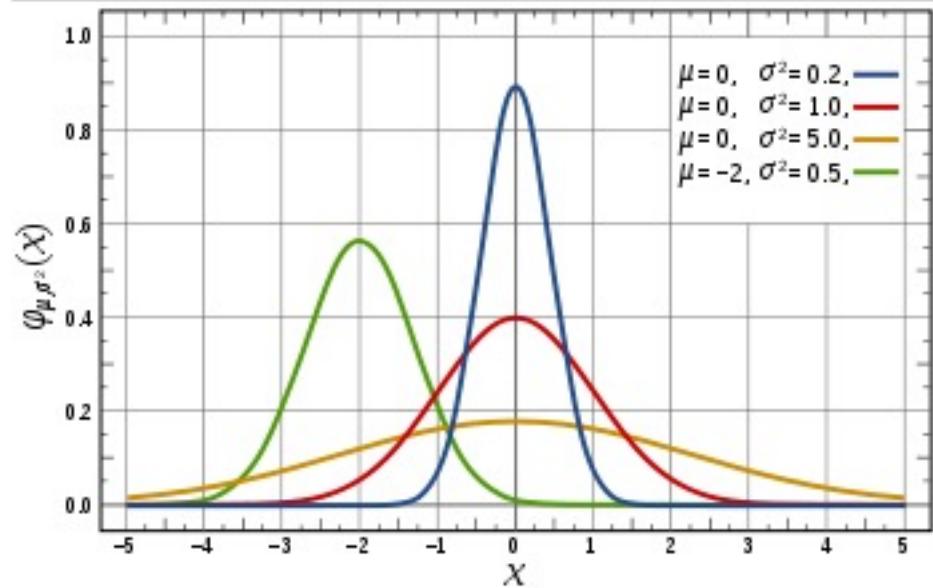
- The only difference between  $\theta_{MLE}$  and  $\theta_{MAP}$  is that one assumes a uniform prior (MLE) and the other allows an arbitrary prior

# MLE for Gaussian Distributions



- Two parameter distribution characterized by a mean and a variance

$$P(x \in [\alpha, \beta]) = \int_{\alpha}^{\beta} P(n) dn.$$



$$P(x | \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

# Some properties of Gaussians

- Affine transformation (multiplying by scalar and adding a constant) are Gaussian
  - $X \sim N(\mu, \sigma^2)$
  - $Y = aX + b \Rightarrow Y \sim N(a\mu + b, a^2\sigma^2)$
- Sum of Gaussians is Gaussian
  - $X \sim N(\mu_X, \sigma_X^2), Y \sim N(\mu_Y, \sigma_Y^2)$
  - $Z = X + Y \Rightarrow Z \sim N(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$
- Easy to differentiate, as we will see soon!

# Learning a Gaussian

- Collect data
  - Hopefully, i.i.d. samples
  - e.g., exam scores
- Learn parameters
  - Mean:  $\mu$
  - Variance:  $\sigma$

$i$	Exam Score
0	85
1	95
2	100
3	12
...	...
99	89

$$P(x | \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

# MLE for Gaussian:

- Probability of  $N$  i.i.d. samples  $D = \underbrace{x^{(1)}, \dots, x^{(N)}}_{\mathcal{D}}$

$$p(D|\mu, \sigma) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^N \prod_{i=1}^N e^{-\frac{(x^{(i)} - \mu)^2}{2\sigma^2}}$$

$$p(x^{(1)}, \dots, x^{(N)} | \mu, \sigma)$$

$$\mu_{MLE}, \sigma_{MLE} = \arg \max_{\mu, \sigma} P(\mathcal{D} | \mu, \sigma)$$

- Log-likelihood of the data

$$\ln p(D|\mu, \sigma) = -\frac{N}{2} \ln 2\pi\sigma^2 - \sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{2\sigma^2}$$

$$p(x^{(1)}, \dots, x^{(N)}) = p(x^{(1)}) p(x^{(2)}) \dots p(x^{(N)})$$

# MLE for the Mean of a Gaussian

$$\begin{aligned}\frac{\partial}{\partial \mu} \ln p(D|\mu, \sigma) &= \frac{\partial}{\partial \mu} \left[ -\frac{N}{2} \ln 2\pi\sigma^2 - \sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right] \\ &= \frac{\partial}{\partial \mu} \left[ - \sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right] \\ &= \sum_{i=1}^N \frac{(x^{(i)} - \mu)}{\sigma^2} \\ &= \frac{[N\mu - \sum_{i=1}^N x^{(i)}]}{\sigma^2} = 0\end{aligned}$$

$$\mu_{MLE} = \frac{1}{N} \sum_{i=1}^N x^{(i)}$$

# MLE for Variance

$$\begin{aligned}\frac{\partial}{\partial \sigma} \ln p(D|\mu, \sigma) &= \frac{\partial}{\partial \sigma} \left[ -\frac{N}{2} \ln 2\pi\sigma^2 - \sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right] \\ &= -\frac{N}{\sigma} + \frac{\partial}{\partial \sigma} \left[ -\sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{2\sigma^2} \right] \\ &= -\frac{N}{\sigma} + \sum_{i=1}^N \frac{(x^{(i)} - \mu)^2}{\sigma^3} = 0\end{aligned}$$

$$\sigma_{MLE}^2 = \frac{1}{N} \sum_{i=1}^N (x^{(i)} - \mu_{MLE})^2$$

# Why Log Likelihood ?

1. Log Likelihood makes it easy to optimize.

2.  $\underset{\theta}{\operatorname{argmax}} \text{LL}(\theta) \equiv \underset{\theta}{\operatorname{argmax}} \log \text{LL}(\theta)$

Likelihood - Max

$$x_1 > x_2 \Leftrightarrow \log x_1 > \log x_2$$

Log is a monotone operator (function)

# Learning Gaussian parameters

$$\mu_{MLE} = \frac{1}{N} \sum_{i=1}^N x^{(i)}$$

$$x^{(1)}, \dots, x^{(N)} \sim N(\mu, \sigma^2)$$

$$\sigma_{MLE}^2 = \frac{1}{N} \sum_{i=1}^N (x^{(i)} - \mu_{MLE})^2$$

- MLE for the variance of a Gaussian is **biased**:  $E[\sigma_{MLE}^2] \neq \sigma_{True}^2$

$$\begin{aligned} E[\mu_{MLE}] &= E\left[\frac{1}{N} \sum_{i=1}^N x^{(i)}\right] \\ &= \frac{1}{N} \sum_{i=1}^N E[x^{(i)}] = \frac{1}{N} \cdot N\mu = \mu \end{aligned}$$

Unbiased Estimator

# Learning Gaussian parameters

$$\mu_{MLE} = \frac{1}{N} \sum_{i=1}^N x^{(i)}$$

$$\sigma_{MLE}^2 = \frac{1}{N} \sum_{i=1}^N (x^{(i)} - \mu_{MLE})^2$$

- MLE for the variance of a Gaussian is **biased**:  $E[\sigma_{MLE}^2] \neq \sigma_{True}^2$ 
  - Expected result of estimation is **not** true parameter!
  - Unbiased variance estimator

$$\sigma_{unbiased}^2 = \frac{1}{N-1} \sum_{i=1}^N (x^{(i)} - \mu_{MLE})^2$$

$$E[\sigma_{unbiased}^2] = \sigma^2$$

# Bayesian Categorization/Classification



- Given features  $x = (x_1, \dots, x_m)$  predict a label  $y$
- If we had a joint distribution over  $x$  and  $y$ , given  $x$  we could find the label using MAP inference

$$\arg \max_y p(y|x_1, \dots, x_m)$$

- Can compute this in exactly the same way that we did before using Bayes rule:

$$p(y|x_1, \dots, x_m) = \frac{p(x_1, \dots, x_m|y)p(y)}{p(x_1, \dots, x_m)}$$

$$p(x_1, x_2, \dots, x_m, y) = p(x_1, \dots, x_m|y)p(y)$$

## Modeling

$$p(y | n_1, \dots, n_m) = \frac{p(n_1, n_2, \dots, n_m | y) p(y)}{p(n_1, \dots, n_m)}$$

Since denominator  $p(n_1, \dots, n_m)$  does not depend on  $y$ , we will ignore it.

① Model  $p(n_1, \dots, n_m | y)$

② Model  $p(y)$

Application

①	$y = "spam"$ or " $Ham$ "
①	$p(n_1, \dots, n_m   "spam")$
①	$p(n_1, \dots, n_m   "Ham")$
②	$p("spam")$ , $p("Ham")$

# Article Classification



- Given a collection of news articles labeled by topic goal is, given an unseen news article, to predict topic
  - One possible feature vector:
    - One feature for each word in the document, in order
    - $x_i$  corresponds to the  $i^{th}$  word
    - $x_i$  can take a different value for each word in the dictionary

# Text Classification



## Article from rec.sport.hockey

---

Path: cantaloupe.srv.cs.cmu.edu!das-news.harvard.e  
From: xxx@yyy.zzz.edu (John Doe)  
Subject: Re: This year's biggest and worst (opinic  
Date: 5 Apr 93 09:53:39 GMT

I can only comment on the Kings, but the most obvious candidate for pleasant surprise is Alex Zhitnik. He came highly touted as a defensive defenseman, but he's clearly much more than that. Great skater and hard shot (though wish he were more accurate). In fact, he pretty much allowed the Kings to trade away that huge defensive liability Paul Coffey. Kelly Hrudey is only the biggest disappointment if you thought he was any good to begin with. But, at best, he's only a mediocre goaltender. A better choice would be Tomas Sandstrom, though not through any fault of his own, but because some thugs in Toronto decided

# Text Classification

- $x_1, x_2, \dots$  is sequence of words in document
- The set of all possible features, and hence  $p(y|x)$ , is huge
  - Article at least 1000 words,  $x = (x_1, \dots, x_{1000})$
  - $x_i$  represents  $i^{th}$  word in document
    - Can be any word in the dictionary – at least 10,000 words
  - $10,000^{1000} = 10^{4000}$  possible values
  - Atoms in Universe:  $\sim 10^{80}$

# Bag of Words Model

$\{(id: value), \dots\}$

- Typically assume position in document doesn't matter

$$p(X_i = "the" | Y = y) = p(\underbrace{X_k = "the"}_{\text{in position } k} | Y = y)$$

- All positions have the same distribution
- Ignores the order of words
- Sounds like a bad assumption, but often works well!
- Features
  - Set of all possible words and their corresponding frequencies (number of times it occurs in the document)

# Bag of Words



*the world of*

**TOTAL**

The screenshot shows a corporate website for TOTAL. At the top left is the "the world of TOTAL" logo. To the right is a navigation menu titled "All About The Company" with links to Global Activities, Corporate Structure, TOTAL's Story, Upstream Strategy, Downstream Strategy, Chemicals Strategy, TOTAL Foundation, and Homepage. Below the menu is a large image of a modern office building with a red "TOTAL" sign on its side, and a small figure of a person standing next to it. The text "all about the company" is displayed in a large, stylized font. Below this, there are three main sections: one about global operations, one about strength from oil and gas reserves, and one about expanding refining and marketing operations. At the bottom, there is a section about the growing specialty chemicals sector.

**All About The Company**

- Global Activities
- Corporate Structure
- TOTAL's Story
- Upstream Strategy
- Downstream Strategy
- Chemicals Strategy
- TOTAL Foundation
- Homepage

*all about the company*

Our energy exploration, production, and distribution operations span the globe, with activities in more than 100 countries.

At TOTAL, we draw our greatest strength from our fast-growing oil and gas reserves. Our strategic emphasis on natural gas provides a strong position in a rapidly expanding market.

Our expanding refining and marketing operations in Asia and the Mediterranean Rim complement already solid positions in Europe, Africa, and the U.S.

Our growing specialty chemicals sector adds balance and profit to the core energy business.

aardvark	0
about2	
all	2
Africa	1
apple0	
anxious	0
...	
gas	1
...	
oil	1
...	
Zaire	0

# Need to Simplify Somehow

- Even with the bag of words assumption, there are too many possible outcomes
  - Too many probabilities

$$p(x_1, \dots, x_m | y)$$

- Can we assume some are the same?

$$p(x_1, x_2 | y) = p(x_1 | y) p(x_2 | y)$$

- This is a conditional independence assumption

# Conditional Independence

- X is **conditionally independent** of Y given Z, if the probability distribution for X is independent of the value of Y, given the value of Z

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

- Equivalent to

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

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

"Click here Free Gift"      Spam.

$$\begin{aligned} p(X_1=\text{click}, X_2=\text{Gift} | \text{spam}) \\ = p(X_1=\text{click} | \text{spam})p(X_2=\text{Gift} | \text{spam}) \end{aligned}$$

# Naïve Bayes

- Naïve Bayes assumption      *conditional Independence*  
    ↓

- Features are independent given class label

$$p(x_1, x_2 | y) = p(x_1 | y) p(x_2 | y)$$

$$p(x_i = 1 | y = +1)$$

- More generally

$$p(x_1, \dots, x_m | y) = \prod_{i=1}^m p(x_i | y)$$

- How many parameters now?  $\rightarrow d | Y |$

- Suppose  $x$  is composed of  $d$  binary features

Example : Spam v/s Ham

Model: Compute  $P(x_i | y)$  for all features

$$\left. \begin{array}{l} P(\text{"Free"} | \text{"spam"}) \\ P(\text{"Click"} | \text{"spam"}) \\ \vdots \\ P(\text{"Spam"}) \\ P(\text{"Ham"}) \end{array} \right\} |D| \cdot |\text{#classes}|$$

# Naïve Bayes

- Naïve Bayes assumption
  - Features are independent given class label

$$p(x_1, x_2 | y) = p(x_1 | y) p(x_2 | y)$$

- More generally

$$p(x_1, \dots, x_m | y) = \prod_{i=1}^m p(x_i | y)$$

- How many parameters now?
  - Suppose  $x$  composed of  $d$  binary features  $\Rightarrow O(d \cdot L)$  where  $L$  is the number of class labels

# The Naïve Bayes Classifier

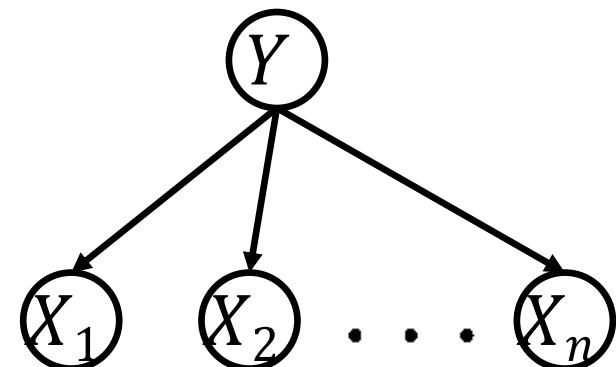
- Given → "Learning Done"

- Prior  $p(y)$
- $m$  conditionally independent features  $X$  given the class  $Y$

- For each  $X_i$ , we have likelihood  $P(X_i|Y)$

- Classify via [Inference]  $p(y|x_1, \dots, x_m)$

$$\begin{aligned}
 y^* = h_{NB}(x) &= \arg \max_y \underbrace{p(y)p(x_1, \dots, x_m|y)}_m \\
 &= \arg \max_y p(y) \prod_i p(x_i|y)
 \end{aligned}$$



$$p(y | n_1, n_2, \dots, n_m) \quad m = \# \text{ Features}$$

$$= \frac{p(n_1, n_2, \dots, n_m | y) p(y)}{p(n_1, n_2, \dots, n_m)}$$

$$y^* = \underset{y}{\operatorname{argmax}} \ p(y | n_1, n_2, \dots, n_m)$$

$$= \underset{y}{\operatorname{argmax}} \ p(y) p(n_1, n_2, \dots, n_m | y)$$

$$= \underset{y}{\operatorname{argmax}} \ p(y) \prod_{i=1}^m p(n_i | y)$$

# MLE for the Parameters of NB

- Given dataset, count occurrences for all pairs
  - $\text{Count}(X_i = x_i, Y = y)$  is the number of samples in which  $X_i = x_i$  and  $Y = y$
- MLE for discrete NB

$$p(Y = y) = \frac{\text{Count}(Y = y)}{\sum_{y'} \text{Count}(Y = y')} \rightarrow \# \text{ Instances}$$

$$p(X_i = x_i | Y = y) = \frac{\text{Count}(X_i = x_i, Y = y)}{\sum_{x'_i} \text{Count}(X_i = x'_i, Y = y)}$$

$\hookrightarrow \text{Count}(Y = y)$

See this link for more insights: <http://www.datasciencecourse.org/notes/mle/>

$$D = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(N)}, y^{(N)})\}$$

$$x^{(i)} = \{x_1^{(i)}, x_2^{(i)}, \dots, x_m^{(i)}\} \quad m = \# \text{Feat}$$

N = # Instances

$$\Theta = \text{Parameters} = \{P(y), P(x_i|y)\}$$

$$\underset{\Theta}{\operatorname{argmax}} \quad P(D|\Theta) = \underset{\Theta}{\operatorname{argmax}} \quad P((x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)}) | \Theta)$$

$$= \underset{\Theta}{\operatorname{argmax}} \quad \prod_{i=1}^N P(x^{(i)}, y^{(i)} | \Theta)$$

$$= \underset{\Theta}{\operatorname{argmax}} \quad \prod_{i=1}^N p(x^{(i)} | y^{(i)}) p(y^{(i)})$$

$$= \underset{\Theta}{\operatorname{argmax}} \quad \prod_{i=1}^N \prod_{j=1}^m p(x_j^{(i)} | y^{(i)}) p(y^{(i)})$$

$$\prod_{i=1}^n p(y^{(i)}) = p(y^{(i)} = "spam")^{\#spam} p(y^{(i)} = "Ham")^{\#Ham}$$

$$p(D|\theta) = \prod_y p(y))^{count(y)} \prod_{x_i, y} p(x_i|y)^{count(x_i|y)}$$

$$p(y) = \frac{\text{Count}(Y=y)}{\sum_{y'} \text{Count}(Y=y')}$$

$$p(x_i=x_i | Y=y) = \frac{\text{Count}(x_i=x_i, Y=y)}{\sum_{x_i'} \text{Count}(x_i=x_i', Y=y)}$$

# Naïve Bayes Calculations

Yes = 9  
No = 5



Test      Sunny      Cool      High      Strong

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

$$P(\text{"Yes"}) = 9/14, \quad P(\text{"No"}) = 5/14.$$

$$P(\text{outlook} = \text{"Sunny"} \mid \text{Play} = \text{"Yes"})$$

$$= 2/9.$$

$$P(\text{outlook} = \text{"Sunny"} \mid \text{Play} = \text{"No"}) = 3/5.$$

$$P(\text{outlook} = \text{"overcast"} \mid \text{Play} = \text{"Yes"}) = 4/9$$

$$P(\text{outlook} = \text{"overcast"} \mid \text{Play} = \text{"No"}) = 0$$

- - -

$$x^{test} = (O = "sunny", T = "cool", H = "High", W = "strong")$$

$$\operatorname{argmax}_y p(y | x^{test})$$

$$\propto p(y) p(x^{test} | y)$$

$$p(y) p(O = "sunny" | y) p(T = "cool" | y)$$

- - -

$$p(y = "yes" | x^{test})$$

$$p(y = "no" | x^{test}) = 1 - p(y = "yes" | x^{test}) \quad \times$$

# Subtleties of NB Classifier: #1

- Usually, features are not conditionally independent:

$$p(x_1, \dots, x_m | y) \neq \prod_{i=1}^m p(x_i | y)$$

- The naïve Bayes assumption is often violated, yet it performs surprisingly well in many cases
- Plausible reason: Only need the probability of the correct class to be the largest!
  - Example: binary classification; just need to figure out the correct side of 0.5 and not the actual probability (0.51 is the same as 0.99).

# Subtleties of NB Classifier

- What if you never see a training instance  $(X_1 = a, Y = b)$ 
  - Example: you did not see the word “Nigerian” in spam
    - Then  $p(X_1 = a | Y = b) = 0$
    - Thus no matter what values  $X_2, \dots, X_m$  take
  - Why?

# Subtleties of NB Classifier

- To fix this, use a prior!
  - Already saw how to do this in the coin-flipping example using the Beta distribution
  - For NB over discrete spaces, can use the Dirichlet prior
  - The Dirichlet distribution is a distribution over  $z_1, \dots, z_k \in (0,1)$  such that  $z_1 + \dots + z_k = 1$  characterized by  $k$  parameters  $\alpha_1, \dots, \alpha_k$

$$f(z_1, \dots, z_k; \alpha_1, \dots, \alpha_k) \propto \prod_{i=1}^k z_i^{\alpha_i - 1}$$

- Called **smoothing**, what are the MLE estimates under these kinds of priors?

## How does Smoothing Work?

$$p(Y=y) = \frac{\text{Count}(Y=y) + \alpha}{\sum_y \text{Count}(Y=y) + \alpha K} \quad \text{--- } K = \# \text{ classes}$$

$$p(X_i=x_i | Y=y) = \frac{\text{Count}(X_i=x_i, Y=y) + \alpha}{\sum_{x_i} [\text{Count}(X_i=x_i, Y=y) + \alpha]}$$

example: Outlook = "overcast"

$$p(\text{outlook} = \text{"overcast"} | \text{play} = \text{"yes"}) = \frac{4+1}{9+3} = \frac{5}{12} (\alpha=1)$$

$$p(\text{outlook} = \text{"overcast"} | \text{play} = \text{"no"}) = \frac{0+1}{5+3} = \frac{1}{8}$$

# Continuous Naïve Bayes

- Continuous Naïve Bayes, also known as Gaussian Naïve Bayes is where the features are continuous
- The distribution  $p(X_i = x_i | Y = y) = N(x_i, \mu_y, \sigma_y^2)$
- In other words, the conditional distribution of each feature given the class is a Gaussian distribution with mean  $\mu_y$  and variance  $\sigma_y^2$
- We can use the Naïve Bayes assumption and assume:

$$p(x_1, \dots, x_m | y) = \prod_{i=1}^m p(x_i | y)$$

- The distribution of labels is the same as the multinomial case

# Parameter Estimation of Cont. NB



- The parameter estimation can similarly be obtained using the Maximum Likelihood Estimation
- The mean and variance can be estimated as the standard Gaussian distribution except that we restrict to each label

$$\mu_y = \frac{\sum_{j=1}^m x_i^{(j)} \mathbf{1}\{y^{(j)} = y\}}{\sum_{j=1}^m \mathbf{1}\{y^{(j)} = y\}}, \rightarrow \text{Count}(y)$$

$$\sigma_y^2 = \frac{\sum_{j=1}^m (x_i^{(j)} - \mu_y)^2 \mathbf{1}\{y^{(j)} = y\}}{\sum_{j=1}^m \mathbf{1}\{y^{(j)} = y\}}$$

# Parameter Estimation of Cont. NB



- Finally, we need to estimate  $p(y)$
- This is like the discrete Naïve Bayes case:

$$p(Y = y) = \frac{\text{Count}(Y = y)}{\sum_{y'} \text{Count}(Y = y')}$$

- We can classify a test example in a similar way to discrete NB:

$$\begin{aligned} y^* &= h_{NB}(x) = \arg \max_y p(y) \prod_m p(x_m | y) \\ &= \arg \max_y p(y) \prod_i p(x_i | y) \end{aligned}$$

- Here  $p(x_i | y) = N(x_i, \mu_y, \sigma_y^2)$

# Summary of Naïve Bayes Models

- Two kinds of Naïve Bayes: Discrete and Continuous
- Learning is often very simple
  - Using counts (discrete NB) or mean/variance (cont. NB), obtain estimates for  $p(x_i | y)$
  - Using counts, obtain estimates for  $p(y)$
- At inference time, we classify based on:

$$\begin{aligned}y^* = h_{NB}(x) &= \arg \max_y p(y) p(x_1, \dots, x_m | y) \\&= \arg \max_y p(y) \prod_i^m p(x_i | y)\end{aligned}$$

We cannot use  $\sum_y p(y|x) = 1$   
↑  
Not true.