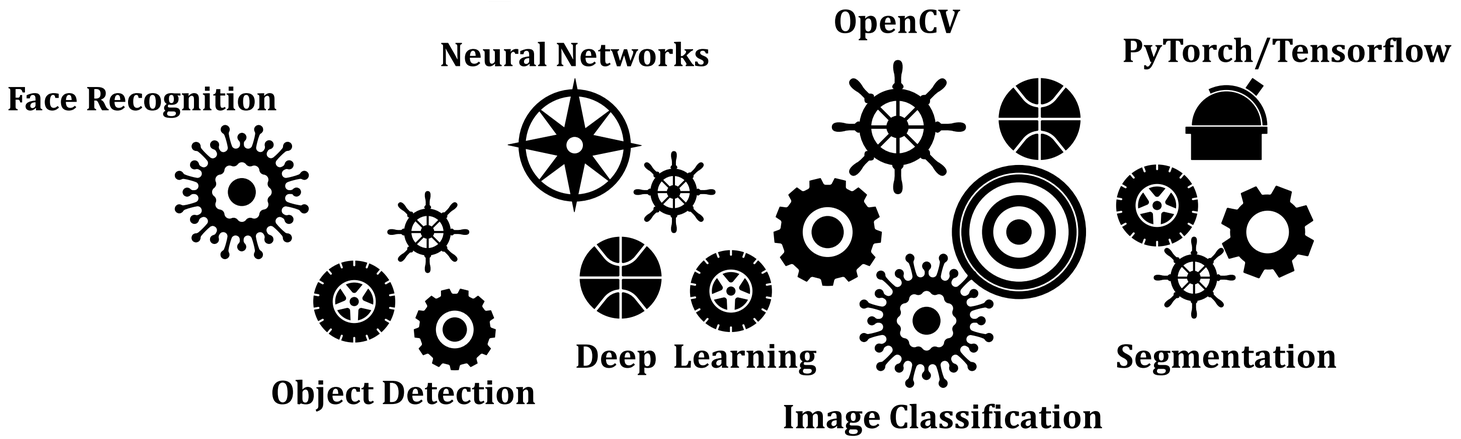
MPSSDEGB
कौशल से रोजगार, समृद्धि अपार

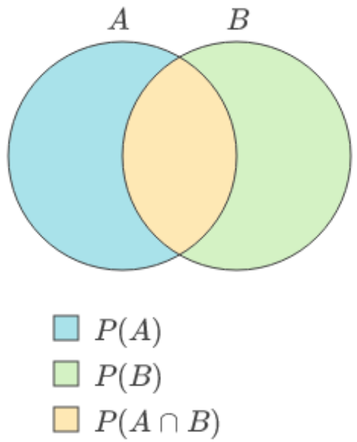
FITT
FOUNDATION FOR INNOVATION AND TECHNOLOGY TRANSFER
भारतीय प्रौद्योगिकी संस्थान दिल्ली
Indian Institute of Technology Delhi

Computer Vision



CONDITIONAL PROBABILITY

- In probability theory, conditional probability is a measure of the probability of an event given that another event has already occurred.
- If the event of interest is A and the event B is assumed to have occurred, "the conditional probability of A given B", or "the probability of A under the condition B", is usually written as P(AIB), or sometimes PB(A).



Conditional Probability Formula

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

Probability that A occurs given that B has already occured

Mirjam Nilsson 206-555-0146 mirjam@contoso.com www.contoso.com

EXAMPLES

Marbles in a Bag

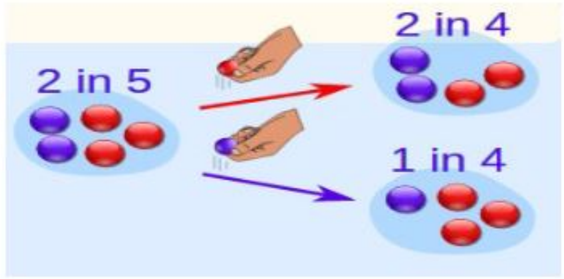
2 red and 3 blue marbles are in a bag.

What are the chances of getting a blue marble?

???

Answer: - The chance is 3 in 5

EXAMPLES



But after taking one out of these chances,

situation may change!

So the next time:

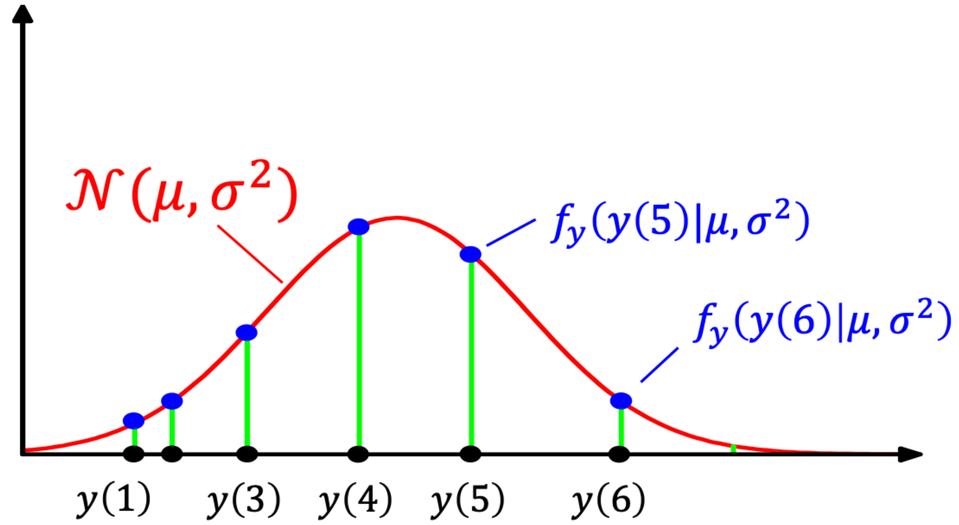
1. if we got a red marble before, then the chance of a blue marble next is 2 in 4

2. if we got a blue marble before, then the chance of a blue marble next is 1 in 4

Maximum Likelihood Estimation

The Maximum Likelihood Estimation (MLE) method is an estimation procedure that, **given a probabilistic model**, estimates its **parameters** in such a way that they are **most consistent** with the observed data

Assume to have 6 i. i. d. observations $\mathcal{D} = \{y(1), y(2), \dots, y(6)\}$, where $y(i) \sim \mathcal{N}(\mu, \sigma^2)$



The **pdf** of a single random variable is

$$f(y(i)|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{y(i) - \mu}{\sigma}\right)^2\right]$$

Maximum Likelihood Estimation

Defined the data vector $Y = [y(1), y(2), \dots, y(N)]^{\top}$. The **joint pdf** of the data vector Y is

$$f_Y(y(1), y(2), \dots, y(N)|\mu, \sigma^2) = \prod_{i=1}^{N} f_y(y(i)|\mu, \sigma^2) = \prod_{i=1}^{N} \mathcal{N}(y(i)|\mu, \sigma^2)$$

The **value assumed** by the joint pdf $f_Y(Y|\mu, \sigma^2)$, with **known** μ and σ^2, evaluated using the data $\mathcal{D}$, is the product of the **blue dots** in the previous example, where we had $N = 6$ observations

Maximizing the likelihood means maximizing this product

Maximum Likelihood Estimation

If **function of the data** Y, the joint pdf is a **multivariable distribution**. But **we know** the value of Y, since we observed those data

If we also knew μ and σ, we could compute the probability of having observed Y. But **we do not know** μ and σ! That's exactly what we want to estimate!

When $f_Y(Y|\mu, \sigma^2)$ (the **joint pdf**) is seen as **function of the parameters** μ and σ, it is called **likelihood** $\mathcal{L}(\mu, \sigma^2|Y)$

Summary

Not known variables — **KNOWN parameters**

- If $f_Y(Y|\ \mu,\ \sigma^2\ )$ is function of the data Y: **multivariable pdf**

KNOWN data — **Not known variables**

- If $f_Y(Y|\ \mu,\ \sigma^2\ )$ is function of the parameters μ e σ^2: **likelihood** $\mathcal{L}(\mu, \sigma^2|Y)$

Usually, the notation of $f_Y(Y|\mu,\sigma^2)$ changes into $\mathcal{L}(\mu,\sigma^2|Y)$, to make clearer who is supposed known («*to the right of the bar* |») and who is not known («*to the left of the bar*|»)

Maximum Likelihood Estimation

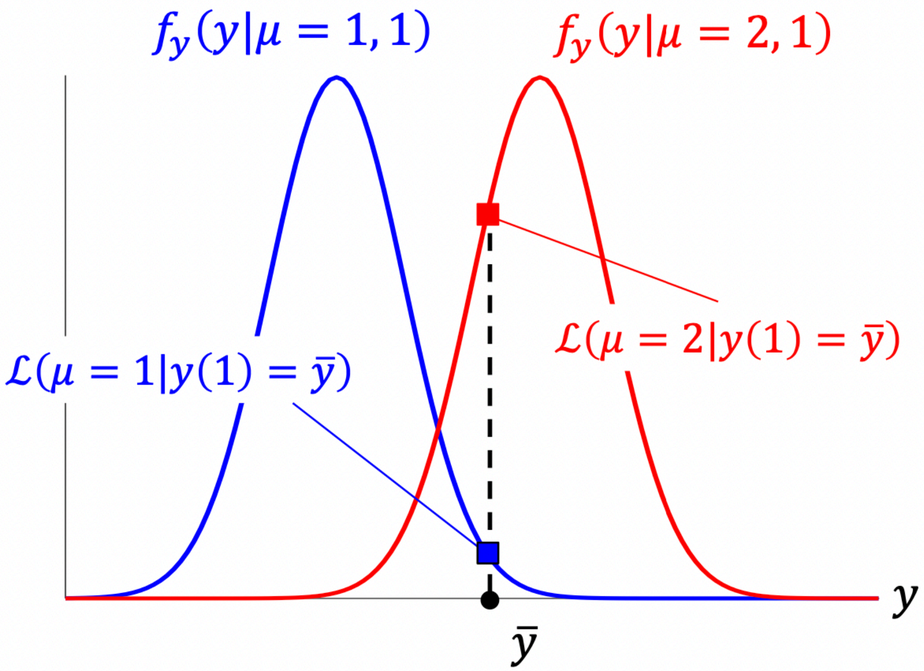
The MLE is that value of the parameters vector $\boldsymbol{\theta}$ who **maximizes the likelihood** $\mathcal{L}(\boldsymbol{\theta}|Y)$

Example: suppose to have **only one datum** $y(1) \sim \mathcal{N}(\mu, \sigma^2 = 1)$, and that its value is $y(1) = \bar{y}$. The **parameter to be estimated** is $\theta = \mu$ (the mean of the distribution)

Notice that:

$$\mathcal{L}(\mu = 2|y(1) = \bar{y}) > \mathcal{L}(\mu = 1|y(1) = \bar{y})$$

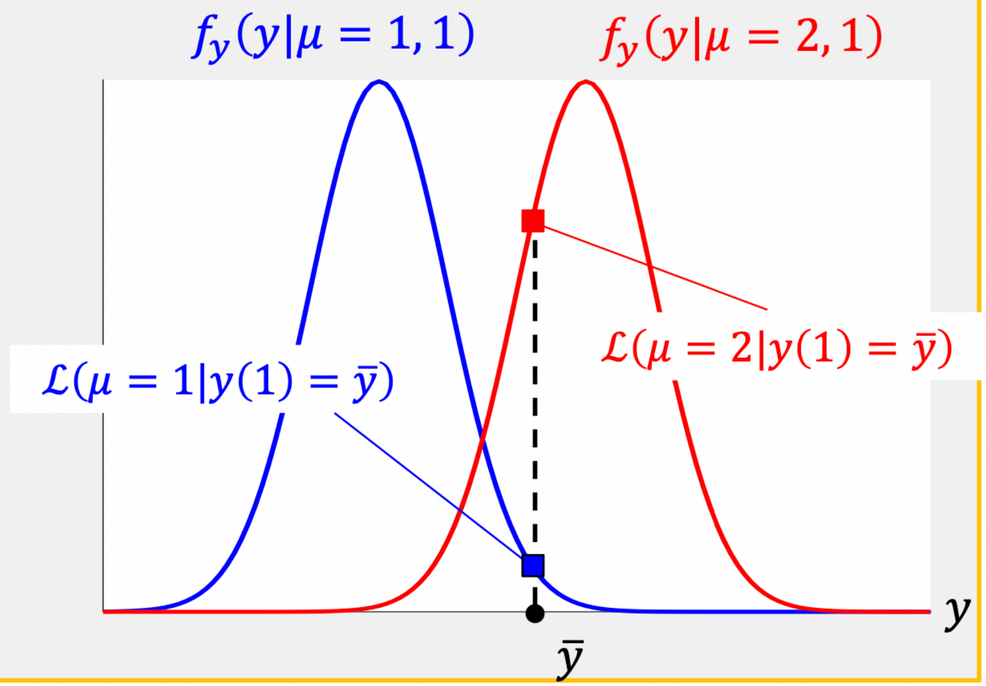
So that $\mu = 2$ is **more likely to be observed** than $\mu = 1$, given this probabilistic model and the data



QUIZ!

In this example, the **maximum likelihood estimate** is:

- ❑ $\hat{\mu} = 2\bar{y}$
- ❑ $\hat{\mu} = \bar{y}$
- ❑ $\hat{\mu} = 2$



Maximum Likelihood Estimation

The maximum likelihood estimate of the previous example can be expressed as:

$$\underset{2\times 1}{\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}} = \begin{bmatrix} \hat{\mu} \\ \hat{\sigma}^2 \end{bmatrix} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}|Y) = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{N} \mathcal{N}(y(i)|\mu, \sigma^2)$$

In general, we can attribute to the data any probability distribution $f_Y(\ )$, both continuous and discrete

$$\underset{d\times 1}{\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}|Y)$$

Maximum Likelihood Estimation

Often, instead of maximizing $\mathcal{L}(\boldsymbol{\theta}|Y)$, we maximize its **natural logarithm**

- Since the logarithm is an increasing monotone function, $\ln \mathcal{L}(\boldsymbol{\theta}|Y)$ **has the same maximum** of $\mathcal{L}(\boldsymbol{\theta}|Y)$
- Using the logarithm is efficient from an **implementation point of view**, because it avoids possible underflows given by the product of small probabilities (replacing it with the sum of the log-probabilities)

$$\underset{d \times 1}{\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}} = \arg\max_{\boldsymbol{\theta}} \ln \mathcal{L}(\boldsymbol{\theta}|Y)$$

Unless special lucky cases, the optimization is carried out with **iterative methods**

MLE of the mean of a Gaussian distribution

Let us consider the case in which we want to **estimate the mean** μ of a population of **i.i.d. Gaussian** random variables, **assuming we know the variance** of the distributions

Assume to have observed only **2 data** $y(i) \sim \mathcal{N}(\mu, \sigma^2 = 1), i = 1,2,$ i.i.d., with values $y(1) = 4, \; y(2) = 6$

The shape of the **pdf** of the single random variables $y(i)$ is:

$$f_y(y(i)|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{y(i) - \mu}{\sigma}\right)^2\right] = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(y(i) - \mu)^2\right]$$

MLE of the mean of a Gaussian distribution

The **value assumed by the pdf** in correspondence of the two observations is:

$$y(1) = 4 \qquad y(2) = 6$$

$$f_y(y(1) = 4|\mu, \sigma^2 = 1) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(4-\mu)^2\right] \qquad f_y(y(2) = 6|\mu, \sigma^2 = 1) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(6-\mu)^2\right]$$

The **joint distribution** is the product of the two single pdfs (since the data are i.i.d.)

$$p(y(1), y(2)|\mu, \sigma^2 = 1) = \left(\frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(4-\mu)^2\right]\right) \cdot \left(\frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(6-\mu)^2\right]\right)$$

MLE of the mean of a Gaussian distribution

The joint pdf is only a function of μ, since **the value of the data is known**. With this interpretation, the joint pdf is the **likelihood function**

$$\mathcal{L}(\mu|y(1)=4,y(2)=6)=\left(\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{2}(4-\mu)^2\right]\right)\cdot\left(\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{2}(6-\mu)^2\right]\right)$$

The estimate $\hat{\mu}_{\mathrm{ML}}$ is the value of μ that **maximizes** the likelihood

$$\hat{\mu}_{\mathrm{ML}}=\arg\max_{\mu}\ln\mathcal{L}(\mu|y(1)=4,y(2)=6)$$

MLE of the mean of a Gaussian distribution

It is more convenient to maximize the log of the likelihood. This new function (the **log-likelihood**) has the same maximum of the likelihood

$$\ln(\mathcal{L}) = \ln\left[\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(4-\mu)^2\right)\cdot\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(6-\mu)^2\right)\right]$$

$$= \ln\left[\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(4-\mu)^2\right)\right] + \ln\left[\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(6-\mu)^2\right)\right]$$

$$= \ln\frac{1}{\sqrt{2\pi}} + \ln\left[\exp\left(-\frac{1}{2}(4-\mu)^2\right)\right] + \ln\frac{1}{\sqrt{2\pi}} + \ln\left[\exp\left(-\frac{1}{2}(6-\mu)^2\right)\right]$$

$$= 2\cdot\ln\frac{1}{\sqrt{2\pi}} - \frac{1}{2}(4-\mu)^2 - \frac{1}{2}(6-\mu)^2$$

MLE of the mean of a Gaussian distribution

By maximizing the expression obtained with respect to μ we get:

$$\frac{\partial \ln \mathcal{L}}{\partial \mu} = 0 \Rightarrow (4-\mu) + (6-\mu) = 0 \Rightarrow \qquad \hat{\mu}_{\mathrm{ML}} = \frac{4+6}{2} = \boxed{5}$$

The **maximum likelihood estimate** of the parameter μ for the Gaussian model is equal to the estimate obtained using the **sample mean estimator!**

This result, although not generalizable, makes the maximum likelihood estimator very interpretable and intuitive

Maximum A Posteriori (MAP)

We go back to the Bayesian Rule

$$p(\Theta|\mathcal{X}) = \frac{p(\mathcal{X}|\Theta)\, p(\Theta)}{p(\mathcal{X})} \tag{1}$$

We now seek that value for Θ, called $\widehat{\Theta}_{MAP}$

It allows to maximize the posterior $p(\Theta|\mathcal{X})$

$$\begin{aligned}\widehat{\Theta}_{MAP} &= \underset{\Theta}{\text{argmax}}\, p(\Theta) \\ &= \underset{\Theta}{\text{argmax}}\, \frac{p(\mathcal{X}|\Theta)\, p(\Theta)}{P(\mathcal{X})}\end{aligned}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

We look to maximize $\widehat{\Theta}_{MAP}$

$$\begin{aligned}\widehat{\Theta}_{MAP} &= \underset{\Theta}{\text{argmax}} p(\Theta) \\ &= \underset{\Theta}{\text{argmax}} \frac{p(\mathcal{X}|\Theta)\, p(\Theta)}{P(\mathcal{X})} \\ &\approx \underset{\Theta}{\text{argmax}} p(\mathcal{X}|\Theta)\, p(\Theta) \\ &= \underset{\Theta}{\text{argmax}} \prod_{x_i \in \mathcal{X}} p(x_i|\Theta)\, p(\Theta)\end{aligned}$$

$P(\mathcal{X})$ can be removed because it has no functional relation with Θ.

Use logarithms

$$\widehat{\Theta}_{MAP} = \underset{\Theta}{\text{argmax}} \left[\sum_{x_i \in \mathcal{X}} \log p(x_i|\Theta) + \log p(\Theta) \right] \tag{2}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

BAYES THEOREM

Bayes' formula

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

Where,

1. P(A) is the probability of hypothesis A being true. This is known as the prior probability.
2. P(B) is the probability of the evidence(regardless of the hypothesis).
3. P(BIA) is the probability of the evidence given that hypothesis is true.
4. P(AIB) is the probability of the hypothesis given that the evidence is there.

Mirjam Nilsson 206-555-0146 mirjam@contoso.com www.contoso.com

Maximum A Posteriori (MAP)

Something Notable

The MAP estimate allows us to inject into the estimation calculation our prior beliefs regarding the parameters values in Θ.

For example

Let's conduct N independent trials of the following Bernoulli experiment with q parameter:

- We will ask each individual we run into in the hallway whether they will vote PRI or PAN in the next presidential election.

With probability p to vote PRI

Where the values of x_i is either PRI or PAN.

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

$$\mathcal{X} = \left\{ x_i = \begin{cases} PAN \\ PRI \end{cases} \quad i = 1, ..., N \right\} \tag{3}$$

Maximum Likelihood Estimation

The log likelihood function

$$\begin{aligned} \log P(\mathcal{X}|p) &= \sum_{i=1}^{N} \log p(x_i|q) \\ &= \sum_{i} \log p(x_i = PRI|q) + ... \\ &\quad \sum_{i} \log p(x_i = PAN|1-q) \\ &= n_{PRI} \log q + (N - n_{PRI}) \log(1-q) \end{aligned}$$

Where n_{PRI} are the numbers of individuals who are planning to vote PRI this fall

9 / 66

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

By setting

$$\mathcal{L} = \log P\left(\mathcal{X}|q\right) \tag{4}$$

We have that

$$\frac{\partial \mathcal{L}}{\partial q} = 0 \tag{5}$$

Thus

$$\frac{n_{PRI}}{q} - \frac{(N - n_{PRI})}{(1-q)} = 0 \tag{6}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

We get

$$\widehat{q}_{PRI} = \frac{n_{PRI}}{N} \tag{7}$$

Thus

If we say that $N = 20$ and if 12 are going to vote PRI, we get $\widehat{q}_{PRI} = 0.6$.

Obviously we need a prior belief distribution

We have the following constraints:

- The prior for q must be zero outside the [0,1] interval.
- Within the [0,1] interval, we are free to specify our beliefs in any way we wish.
- In most cases, we would want to choose a distribution for the prior beliefs that peaks somewhere in the [0, 1] interval.

We assume the following

The state of Colima has traditionally voted PRI in presidential elections. However, on account of the prevailing economic conditions, the voters are more likely to vote PAN in the election in question.

Maximum A Posteriori (MAP)

What prior distribution can we use?

We could use a Beta distribution being parametrized by two values α and β

$$P(q) = \frac{1}{B(\alpha, \beta)} q^{\alpha-1} (1-q)^{\beta-1}. \tag{8}$$

Where

We have $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ is the beta function where Γ is the generalization of the notion of factorial in the case of the real numbers.

Properties

When both the $\alpha, \beta > 0$ then the beta distribution has its mode (Maximum value) at

$$\frac{\alpha - 1}{\alpha + \beta - 2}. \tag{9}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

We do the following

We can choose $\alpha = \beta$ so the beta prior peaks at 0.5.

As a further expression of our belief

We make the following choice $\alpha = \beta = 5$.

Why? Look at the variance of the beta distribution

$$\frac{\alpha\beta}{(\alpha+\beta)(\alpha+\beta+1)}. \tag{10}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

We have a variance with $\alpha = \beta = 5$

$$Var\,(q) \approx 0.025$$

Thus, the standard deviation

$sd \approx 0.16$ which is a nice dispersion at the peak point!!!

We have then

$$\widehat{p}_{MAP} = \underset{\Theta}{\text{argmax}} \left[\sum_{x_i \in \mathcal{X}} \log p\,(x_i|q) + \log p\,(q) \right] \tag{11}$$

Plugging back the ML

$$\widehat{p}_{MAP} = \underset{\Theta}{\text{argmax}} \left[n_{PRI} \log q + (N - n_{PRI}) \log (1 - q) + \log p\,(q) \right] \tag{12}$$

Where

$$\log P\,(p) = \log \left(\frac{1}{B\,(\alpha, \beta)} q^{\alpha - 1} (1 - q)^{\beta - 1} \right) \tag{13}$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

We have that

$$\log P(q) = (\alpha - 1)\log q + (\beta - 1)\log(1 - q) - \log B(\alpha, \beta) \tag{14}$$

Now taking the derivative with respect to p, we get

$$\frac{n_{PRI}}{q} - \frac{(N - n_{PRI})}{(1 - q)} - \frac{\beta - 1}{1 - q} + \frac{\alpha - 1}{q} = 0 \tag{15}$$

Thus

$$\widehat{q}_{MAP} = \frac{n_{PRI} + \alpha - 1}{N + \alpha + \beta - 2} \tag{16}$$

With $N = 20$ with $n_{PRI} = 12$ and $\alpha = \beta = 5$

$$\widehat{q}_{MAP} = 0.571$$

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Maximum A Posteriori (MAP)

First

MAP estimation "pulls" the estimate toward the prior.

Second

The more focused our prior belief, the larger the pull toward the prior.

Example

If $\alpha = \beta$=equal to large value

- It will make the MAP estimate to move closer to the prior.

Third

In the expression we derived for $\widehat{q}_{MAP}$, the parameters α and β play a "smoothing" role vis-a-vis the measurement n_{PRI}.

Fourth

Since we referred to q as the parameter to be estimated, we can refer to α and β as the hyper-parameters in the estimation calculations.

https://www.slideshare.net/AndresMendezVazquez/08-machine-learning-maximum-aposteriori

Naïve Bayes Classification

- Spam Classification
  - Given an email, predict whether it is spam or not

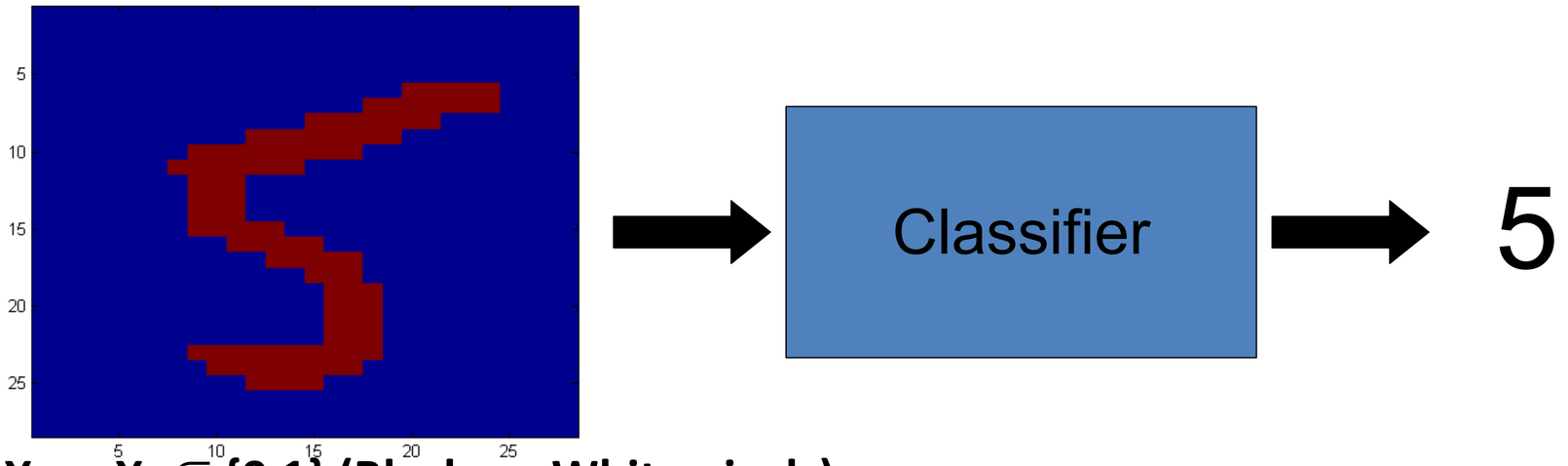
- Medical Diagnosis
  - Given a list of symptoms, predict whether a patient has disease X or not

- Weather
  - Based on temperature, humidity, etc… predict if it will rain tomorrow

Bayesian Classification

- Problem statement:
  - Given features $X_1, X_2, \ldots, X_n$
  - Predict a label Y

- **Digit Recognition**



- **$X_1,\ldots,X_n \in \{0,1\}$ (Black vs. White pixels)**
- **$Y \in \{5,6\}$ (predict whether a digit is a 5 or a 6)**

The Bayes Classifier

- A good strategy is to predict:

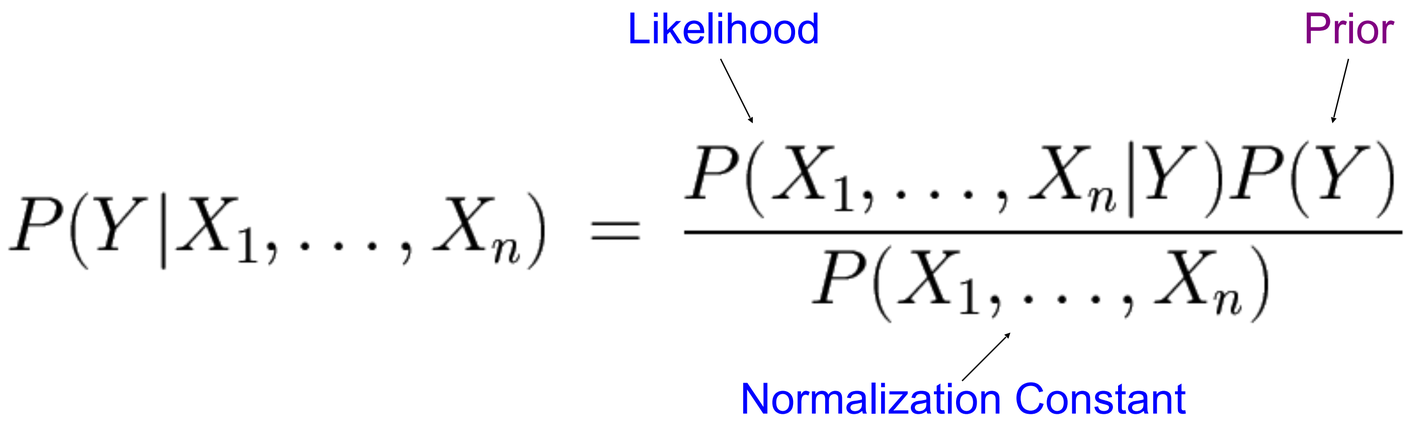
$$\arg\max_Y P(Y|X_1,\ldots,X_n)$$

  - (for example: what is the probability that the image represents a 5 given its pixels?)

- So … How do we compute that?

The Bayes Classifier

- Use Bayes Rule!



- Why did this help? Well, we think that we might be able to specify how features are "generated" by the class label

The Bayes Classifier

- Let’s expand this for our digit recognition task:

$$P(Y=5|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y=5)P(Y=5)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)}$$

$$P(Y=6|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y=6)P(Y=6)}{P(X_1,\ldots,X_n|Y=5)P(Y=5)+P(X_1,\ldots,X_n|Y=6)P(Y=6)}$$

- To classify, we’ll simply compute these two probabilities and predict based on which one is greater

Model Parameters

- For the Bayes classifier, we need to “learn” two functions, the likelihood and the prior

- How many parameters are required to specify the prior for our digit recognition example?

Model Parameters

- How many parameters are required to specify the likelihood?
  - (Supposing that each image is 30x30 pixels)

?

Model Parameters

- The problem with explicitly modeling $P(X_1,\ldots,X_n|Y)$ is that there are usually way too many parameters:
  - We'll run out of space
  - We'll run out of time
  - And we'll need tons of training data (which is usually not available)

The Naïve Bayes Model

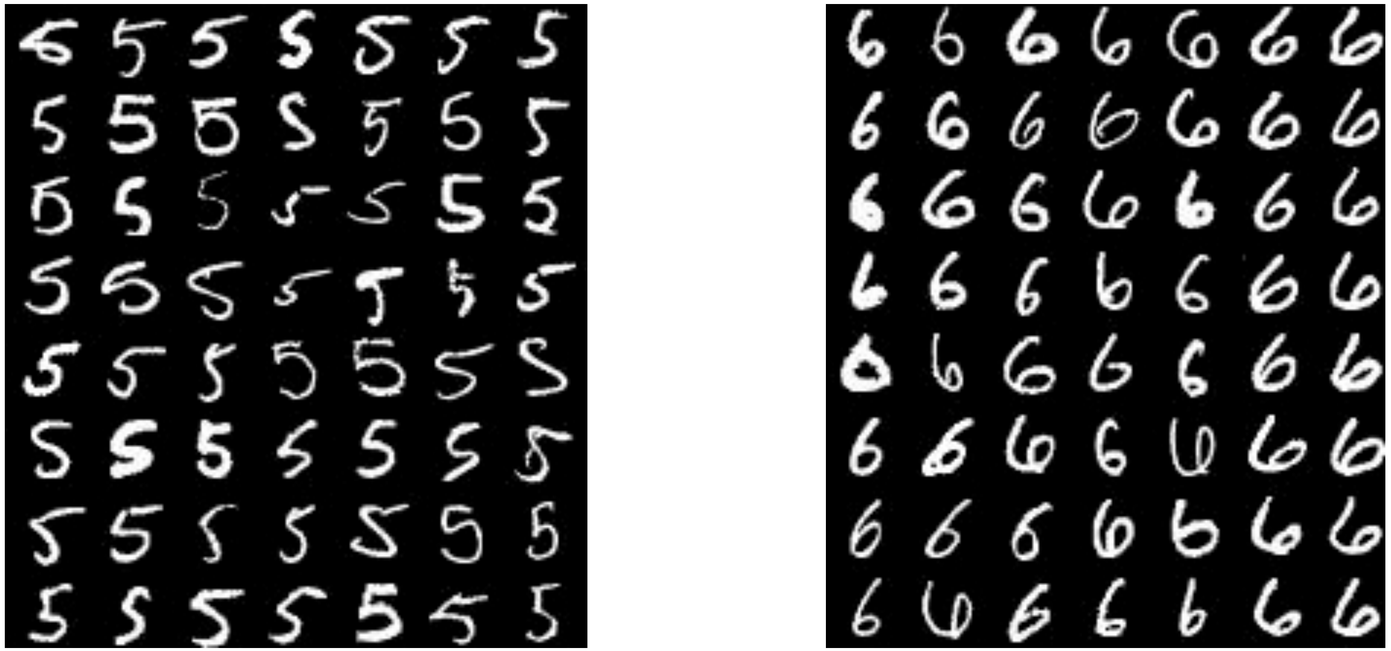
- The *Naïve Bayes Assumption*: Assume that all features are independent **given the class label Y**
- Equationally speaking:

$$P(X_1, \ldots, X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

- (We will discuss the validity of this assumption later)

Naïve Bayes Training

- Now that we've decided to use a Naïve Bayes classifier, we need to train it with some data:



MNIST Training Data

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjOpN6qlpKEAxU2SmwGHU-vArEQFnoECA8QAQ&url=https%3A%2F%2Fwww.cse.sc.edu%2F~rose%2F587%2FPPT%2FNaiveBayes&usg=AOvVaw1XMOjU7MiVZFrYFWvdPDi7&opi=89978449

Naïve Bayes Training

- Training in Naïve Bayes is **easy**:
  - Estimate P(Y=v) as the fraction of records with Y=v

$$P(Y = v) = \frac{Count(Y = v)}{\#\ records}$$

  - Estimate P(X_i=u|Y=v) as the fraction of records with Y=v for which X_i=u

$$P(X_i = u|Y = v) = \frac{Count(X_i = u \wedge Y = v)}{Count(Y = v)}$$

- (This corresponds to Maximum Likelihood estimation of model parameters)

Naïve Bayes Training

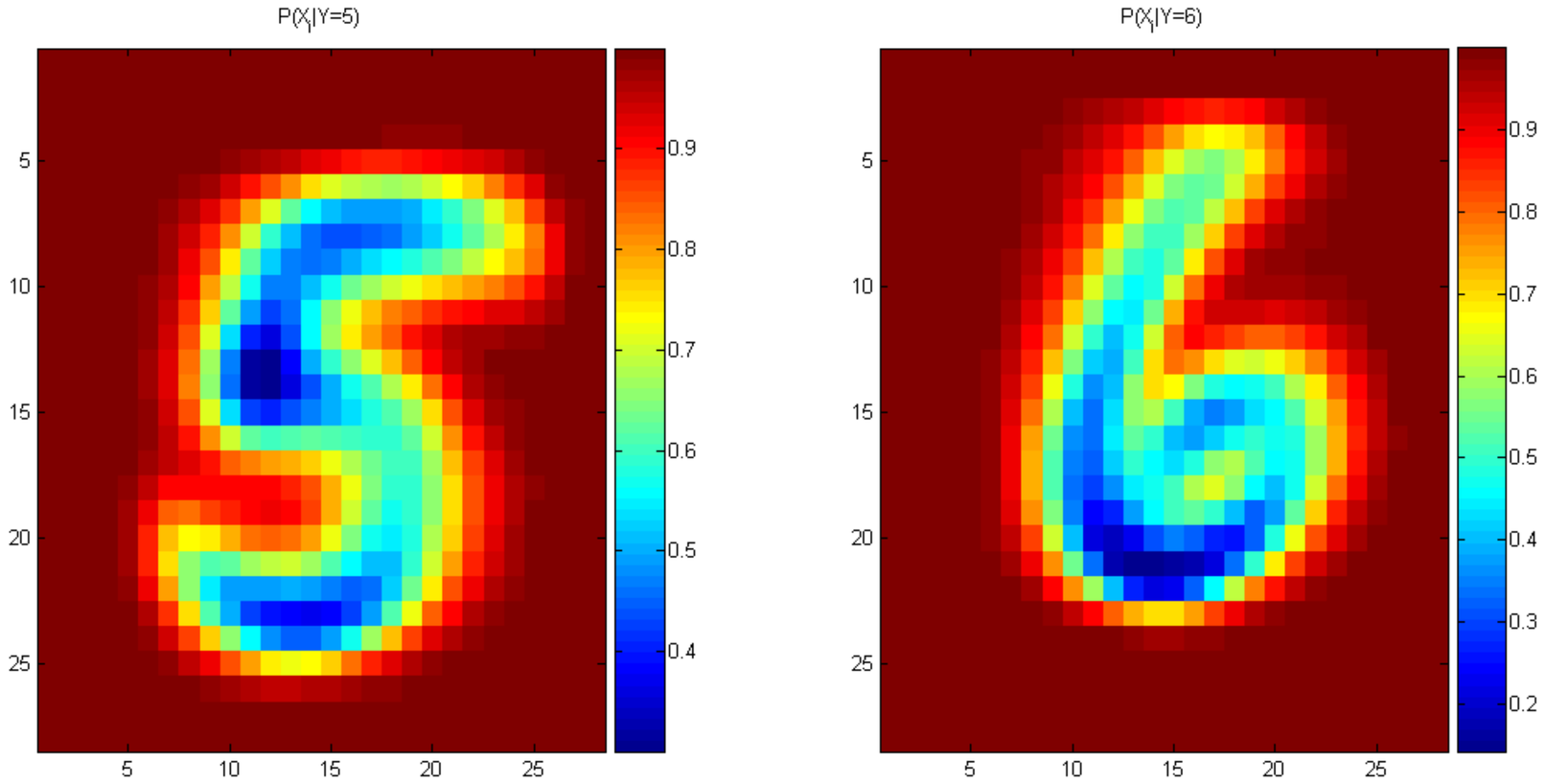
- In practice, some of these counts can be zero
- Fix this by adding “virtual” counts:

$$P(X_i = u | Y = v) = \frac{Count(X_i = u \wedge Y = v) + 1}{Count(Y = v) + 2}$$

  - (This is like putting a prior on parameters and doing MAP estimation instead of MLE)
  - This is called *Smoothing*

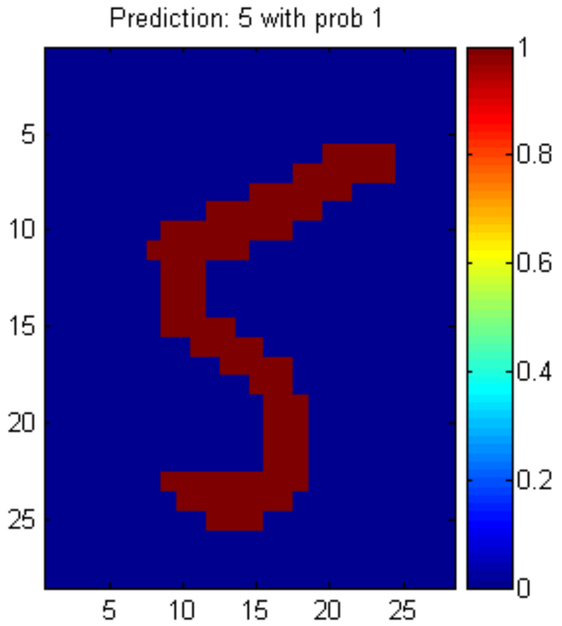
Naïve Bayes Training

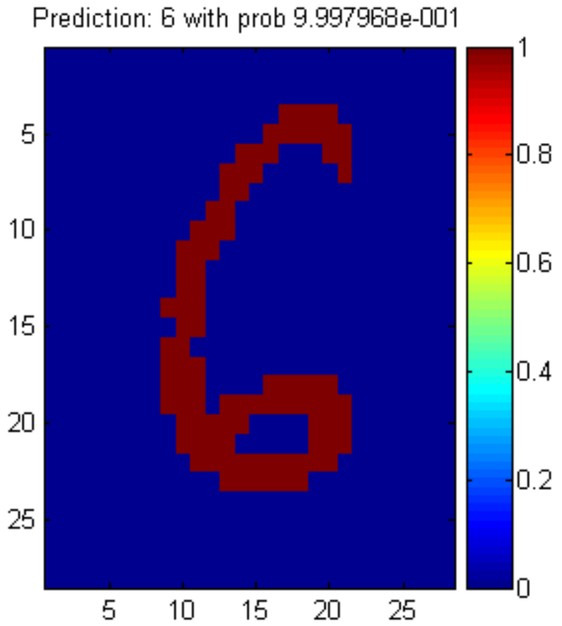
- For binary digits, training amounts to averaging all of the training fives together and all of the training sixes together.

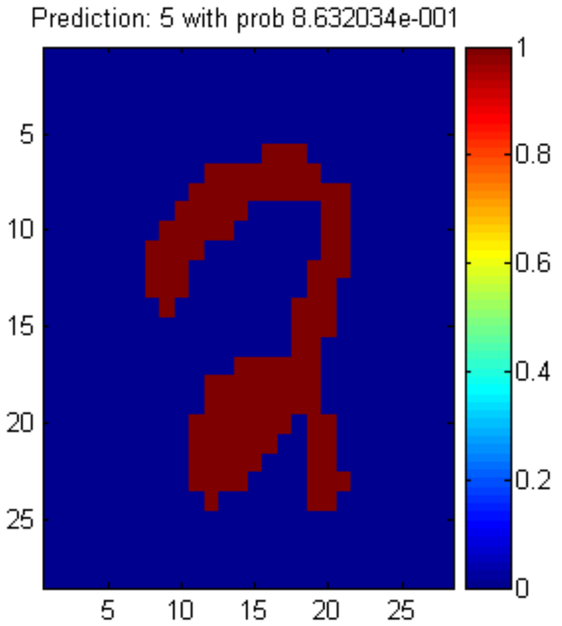


https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjOpN6qlpKEAxU2SmwGHU-vArEQFnoECA8QAQ&url=https%3A%2F%2Fwww.cse.sc.edu%2F~rose%2F587%2FPPT%2FNaiveBayes&usg=AOvVaw1XMOjU7MiVZFrYFWvdPDi7&opi=89978449

Naïve Bayes Classification







Another Example of the Naïve Bayes Classifier

The weather data, with counts and probabilities

outlook			temperature			humidity			windy		play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

A new day

outlook	temperature	humidity	windy	play
sunny	cool	high	true	?

- Likelihood of yes

$$= \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{9}{14} = 0.0053$$

- Likelihood of no

$$= \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5} \times \frac{5}{14} = 0.0206$$

- Therefore, the prediction is No

The Naive Bayes Classifier for Data Sets with Numerical Attribute Values

- One common practice to handle numerical attribute values is to assume normal distributions for numerical attributes.

The numeric weather data with summary statistics

outlook				temperature			humidity			windy		play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3		83	85		86	85	false	6	2	9	5
overcast	4	0		70	80		96	90	true	3	3		
rainy	3	2		68	65		80	70					
				64	72		65	95					
				69	71		70	91					
				75			80						
				75			70						
				72			90						
				81			75						
sunny	2/9	3/5	mean	73	74.6	mean	79.1	86.2	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	std dev	6.2	7.9	std dev	10.2	9.7	true	3/9	3/5		
rainy	3/9	2/5											

- Let x_1, x_2, …, x_n be the values of a numerical attribute in the training data set.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\sigma = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \mu)^2$$

$$f(w) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(w-\mu)^2}{\sigma^2}}$$

- For examples,

$$f(\text{temperature} = 66 \mid \text{Yes}) = \frac{1}{\sqrt{2\pi}(6.2)} e^{-\frac{(66-73)^2}{2(6.2)^2}} = 0.0340$$

- Likelihood of Yes = $\frac{2}{9} \times 0.0340 \times 0.0221 \times \frac{3}{9} \times \frac{9}{14} = 0.000036$

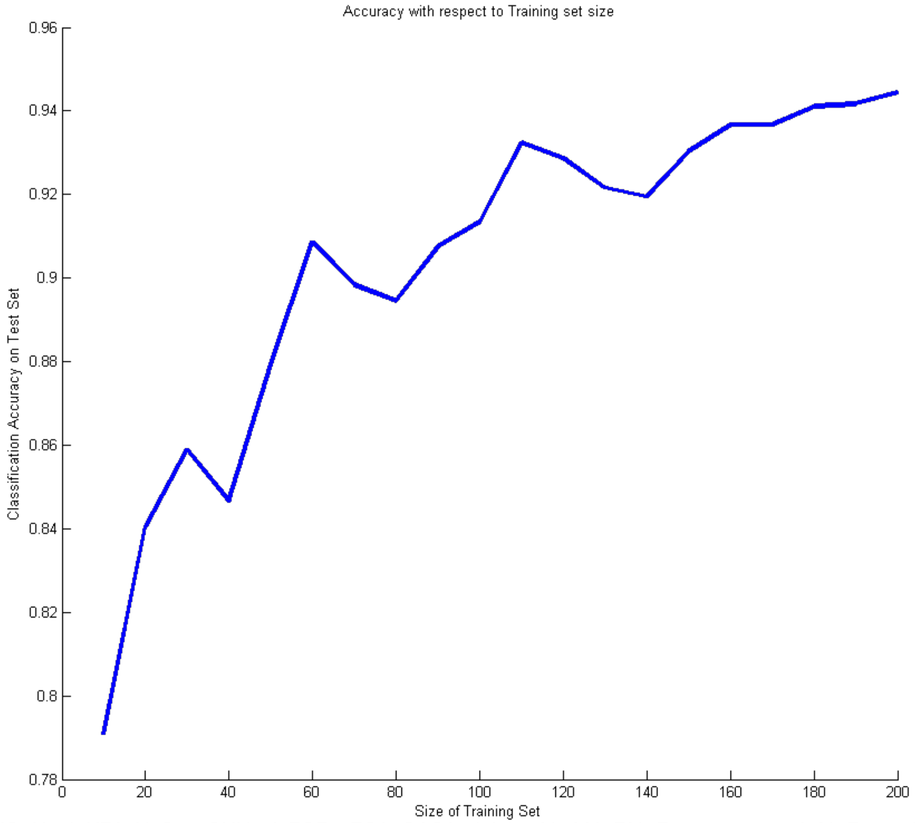
- Likelihood of No = $\frac{3}{5} \times 0.0291 \times 0.038 \times \frac{3}{5} \times \frac{5}{14} = 0.000136$

Outputting Probabilities

- What's nice about Naïve Bayes (and generative models in general) is that it returns probabilities
  - These probabilities can tell us how confident the algorithm is
  - So… don't throw away those probabilities!

Performance on a Test Set

- Naïve Bayes is often a good choice if you don’t have much training data!



https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjOpN6qlpKEAxU2SmwGHU-vArEQFnoECA8QAQ&url=https%3A%2F%2Fwww.cse.sc.edu%2F~rose%2F587%2FPPT%2FNaiveBayes&usg=AOvVaw1XMOjU7MiVZFrYFWvdPDi7&opi=89978449

Naïve Bayes Assumption

- Recall the Naïve Bayes assumption:
  - that all features are independent **given the class label Y**

- Does this hold for the digit recognition problem?

Exclusive-OR Example

- For an example where conditional independence fails:
  - Y=XOR(X_1,X_2)

X_1	X_2	P(Y=0|X_1,X_2)	P(Y=1|X_1,X_2)
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	0

The content is added from different sources

Naïve Bayes Classification

- Actually, the Naïve Bayes assumption is almost never true

- Still… Naïve Bayes often performs surprisingly well even when its assumptions do not hold

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjOpN6qlpKEAxU2SmwGHU-vArEQFnoECA8QAQ&url=https%3A%2F%2Fwww.cse.sc.edu%2F~rose%2F587%2FPPT%2FNaiveBayes&usg=AOvVaw1XMOjU7MiVZFrYFWvdPDi7&opi=89978449

Numerical Stability

- It is often the case that machine learning algorithms need to work with very small numbers
  - Imagine computing the probability of 2000 independent coin flips
  - MATLAB thinks that $(.5)^{2000}=0$

Underflow Prevention

- Multiplying lots of probabilities

➔ floating-point underflow.

- Recall: *log(xy) = log(x) + log(y),*

➔ better to sum logs of probabilities rather than multiplying probabilities.

Underflow Prevention

- Class with highest final un-normalized log probability score is still the most probable.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

Numerical Stability

- Instead of comparing $P(Y=5|X_1,\ldots,X_n)$ with $P(Y=6|X_1,\ldots,X_n)$,
  - Compare their logarithms

$$\begin{aligned}
\log\left(P(Y|X_1,\ldots,X_n)\right) &= \log\left(\frac{P(X_1,\ldots,X_n|Y)\cdot P(Y)}{P(X_1,\ldots,X_n)}\right) \\
&= \text{constant} + \log\left(\prod_{i=1}^{n} P(X_i|Y)\right) + \log P(Y) \\
&= \text{constant} + \sum_{i=1}^{n} \log P(X_i|Y) + \log P(Y)
\end{aligned}$$