

Naïve Bayes Classifier

Instructor: Alan Ritter

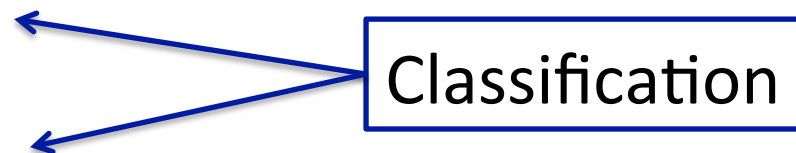
Many Slides from Luke Zettlemoyer

Supervised Learning: find f

- **Given:** Training set $\{(x_i, y_i) \mid i = 1 \dots n\}$
- **Find:** A good approximation to $f : X \rightarrow Y$

Examples: what are X and Y ?

- **Spam Detection**
 - Map email to {Spam,Ham}
- **Digit recognition**
 - Map pixels to {0,1,2,3,4,5,6,7,8,9}
- **Stock Prediction**
 - Map new, historic prices, etc. to \Re (the real numbers)



Example: Spam Filter

- **Input:** email
- **Output:** spam/ham
- **Setup:**
 - Get a large collection of example emails, each labeled “spam” or “ham”
 - Note: someone has to hand label all this data!
 - Want to learn to predict labels of new, future emails
- **Features:** The attributes used to make the ham / spam decision
 - Words: FREE!
 - Text Patterns: \$dd, CAPS
 - Non-text: SenderInContacts
 - ...



Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidential and top secret. ...



TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES
FOR ONLY \$99



Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Example: Digit Recognition

- **Input:** images / pixel grids
- **Output:** a digit 0-9
- **Setup:**
 - Get a large collection of example images, each labeled with a digit
 - Note: someone has to hand label all this data!
 - Want to learn to predict labels of new, future digit images
- **Features:** The attributes used to make the digit decision
 - Pixels: (6,8)=ON
 - Shape Patterns: NumComponents, AspectRatio, NumLoops
 - ...



0



1



2



1



??

Lets take a probabilistic approach!!!

- Can we directly estimate the data distribution $P(X,Y)$?
- How do we represent these?
How many parameters?
 - Prior, $P(Y)$:
 - Suppose Y is composed of k classes
 - Likelihood, $P(\mathbf{X}|Y)$:
 - Suppose \mathbf{X} is composed of n binary features
- Complex model ! High variance with limited data!!!

| mpg | cylinders | displacement | horsepower | weight | acceleration | modelyear | maker |
|------|-----------|--------------|------------|--------|--------------|-----------|---------|
| good | 4 | low | low | low | high | 75to78 | asia |
| bad | 6 | medium | medium | medium | medium | 70to74 | america |
| bad | 4 | medium | medium | medium | low | 75to78 | europa |
| bad | 8 | high | high | high | low | 70to74 | america |
| bad | 6 | medium | medium | medium | medium | 70to74 | america |
| bad | 4 | low | medium | low | medium | 70to74 | asia |
| bad | 4 | low | medium | low | low | 70to74 | asia |
| bad | 8 | high | high | high | low | 75to78 | america |
| : | : | : | : | : | : | : | : |
| : | : | : | : | : | : | : | : |
| : | : | : | : | : | : | : | : |
| bad | 8 | high | high | high | low | 70to74 | america |
| good | 8 | high | medium | high | high | 79to83 | america |
| bad | 8 | high | high | high | low | 75to78 | america |
| good | 4 | low | low | low | low | 79to83 | america |
| bad | 6 | medium | medium | medium | high | 75to78 | america |
| good | 4 | medium | low | low | low | 79to83 | america |
| good | 4 | low | low | medium | high | 79to83 | america |
| bad | 8 | high | high | high | low | 70to74 | america |
| good | 4 | low | medium | low | medium | 75to78 | europa |
| bad | 5 | medium | medium | medium | medium | 75to78 | europa |

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

$$(\forall i, j, k) P(X = i | Y = j, Z = k) = P(X = i | Z = k)$$

- e.g.,

$$P(\textit{Thunder} | \textit{Rain}, \textit{Lightning}) = P(\textit{Thunder} | \textit{Lightning})$$

- Equivalent to:

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

Naïve Bayes

- Naïve Bayes assumption:
 - Features are independent given class:

$$\begin{aligned}P(X_1, X_2|Y) &= P(X_1|X_2, Y)P(X_2|Y) \\ &= P(X_1|Y)P(X_2|Y)\end{aligned}$$

- More generally:

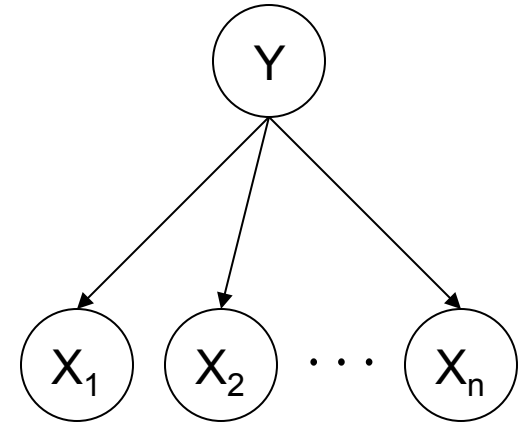
$$P(X_1 \dots X_n|Y) = \prod_i P(X_i|Y)$$

- How many parameters now?
 - Suppose \mathbf{X} is composed of n binary features

The Naïve Bayes Classifier

- Given:

- Prior $P(Y)$
- n conditionally independent features \mathbf{X} given the class Y
- For each X_i , we have likelihood $P(X_i|Y)$

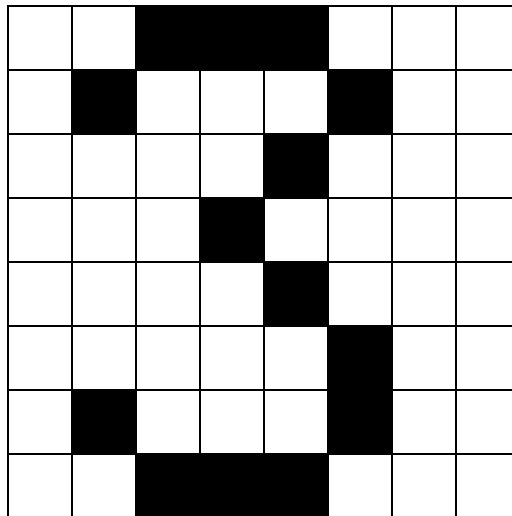


- Decision rule:

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

A Digit Recognizer

- Input: pixel grids




- Output: a digit 0-9



Naïve Bayes for Digits (Binary Inputs)

- Simple version:

- One feature F_{ij} for each grid position $\langle i,j \rangle$
- Possible feature values are on / off, based on whether intensity is more or less than 0.5 in underlying image
- Each input maps to a feature vector, e.g.

 $\rightarrow \langle F_{0,0} = 0 \ F_{0,1} = 0 \ F_{0,2} = 1 \ F_{0,3} = 1 \ F_{0,4} = 0 \ \dots F_{15,15} = 0 \rangle$

- Here: lots of features, each is binary valued

- Naïve Bayes model:

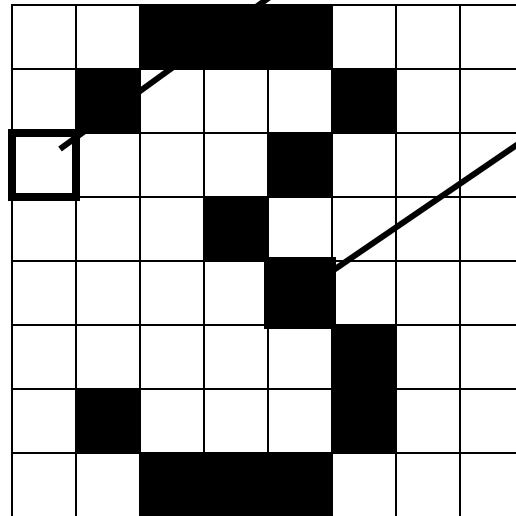
$$P(Y|F_{0,0} \dots F_{15,15}) \propto P(Y) \prod_{i,j} P(F_{i,j}|Y)$$

- Are the features independent given class?
- What do we need to learn?

Example Distributions

$P(Y)$

| | |
|---|-----|
| 1 | 0.1 |
| 2 | 0.1 |
| 3 | 0.1 |
| 4 | 0.1 |
| 5 | 0.1 |
| 6 | 0.1 |
| 7 | 0.1 |
| 8 | 0.1 |
| 9 | 0.1 |
| 0 | 0.1 |



$P(F_{3,1} = on|Y)$ $P(F_{5,5} = on|Y)$

| | |
|---|------|
| 1 | 0.01 |
| 2 | 0.05 |
| 3 | 0.05 |
| 4 | 0.30 |
| 5 | 0.80 |
| 6 | 0.90 |
| 7 | 0.05 |
| 8 | 0.60 |
| 9 | 0.50 |
| 0 | 0.80 |

| | |
|---|------|
| 1 | 0.05 |
| 2 | 0.01 |
| 3 | 0.90 |
| 4 | 0.80 |
| 5 | 0.90 |
| 6 | 0.90 |
| 7 | 0.25 |
| 8 | 0.85 |
| 9 | 0.60 |
| 0 | 0.80 |

MLE for the parameters of NB

- Given dataset
 - $\text{Count}(A=a, B=b)$: number of examples with $A=a$ and $B=b$
- MLE for discrete NB, simply:

- Prior:

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

- Likelihood:

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

Subtleties of NB classifier 1 – Violating the NB assumption

- Usually, features are not conditionally independent:

$$P(X_1 \dots X_n | Y) \neq \prod_i P(X_i | Y)$$

- Actual probabilities $P(Y | \mathbf{X})$ often biased towards 0 or 1
- Nonetheless, NB is the single most used classifier out there
 - NB often performs well, even when assumption is violated
 - [Domingos & Pazzani '96] discuss some conditions for good performance

Subtleties of NB classifier 2: Overfitting

$P(\text{features}, C = 2)$

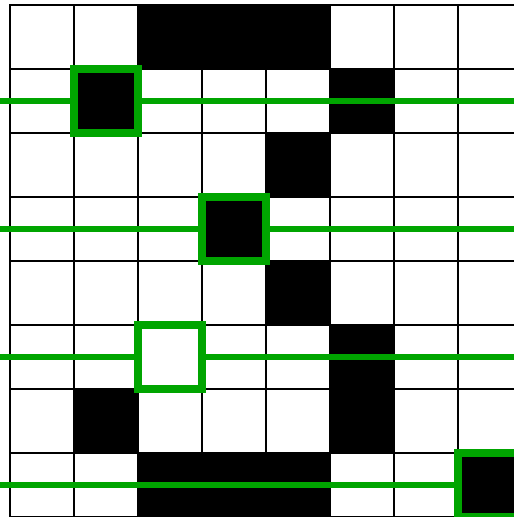
$$P(C = 2) = 0.1$$

$$P(\text{on}|C = 2) = 0.8$$

$$P(\text{on}|C = 2) = 0.1$$

$$P(\text{off}|C = 2) = 0.1$$

$$P(\text{on}|C = 2) = 0.01$$



$P(\text{features}, C = 3)$

$$P(C = 3) = 0.1$$

$$P(\text{on}|C = 3) = 0.8$$

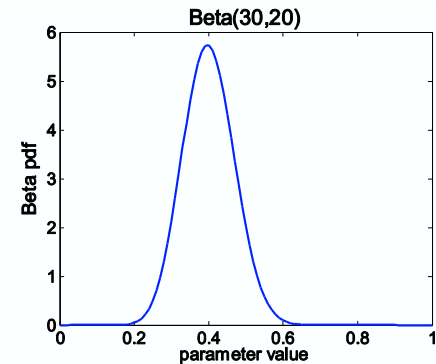
$$P(\text{on}|C = 3) = 0.9$$

$$P(\text{off}|C = 3) = 0.7$$

$$P(\text{on}|C = 3) = 0.0$$

2 wins!!

For Binary Features: We already know the answer!



$$P(\theta \mid \mathcal{D}) = \frac{\theta^{\beta_H + \alpha_H - 1} (1 - \theta)^{\beta_T + \alpha_T - 1}}{B(\beta_H + \alpha_H, \beta_T + \alpha_T)} \sim \text{Beta}(\beta_H + \alpha_H, \beta_T + \alpha_T)$$

- **MAP:** use most likely parameter

$$\hat{\theta} = \arg \max_{\theta} P(\theta \mid \mathcal{D}) = \frac{\alpha_H + \beta_H - 1}{\alpha_H + \beta_H + \alpha_T + \beta_T - 2}$$

- **Beta prior** equivalent to extra observations for each feature
- As $N \rightarrow \infty$, prior is “forgotten”
- **But, for small sample size, prior is important!**

Multinomials: Laplace Smoothing

- Laplace's estimate:

- Pretend you saw every outcome k extra times

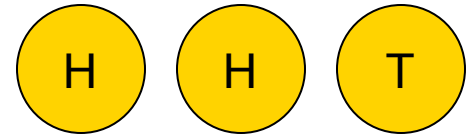
$$P_{LAP,k}(x) = \frac{c(x) + k}{N + k|X|}$$

- What's Laplace with $k = 0$?
- k is the **strength** of the prior
- Can derive this as a MAP estimate for multinomial with *Dirichlet priors*

- Laplace for conditionals:

- Smooth each condition independently:

$$P_{LAP,k}(x|y) = \frac{c(x, y) + k}{c(y) + k|X|}$$



$$P_{LAP,0}(X) = \left\langle \frac{2}{3}, \frac{1}{3} \right\rangle$$

$$P_{LAP,1}(X) = \left\langle \frac{3}{5}, \frac{2}{5} \right\rangle$$

$$P_{LAP,100}(X) = \left\langle \frac{102}{203}, \frac{101}{203} \right\rangle$$

Text classification

- Classify e-mails
 - $Y = \{\text{Spam}, \text{NotSpam}\}$
- Classify news articles
 - $Y = \{\text{what is the topic of the article?}\}$
- Classify webpages
 - $Y = \{\text{Student, professor, project, ...}\}$
- What about the features **X**?
 - The text!

Features **X** are entire document – X_i for i^{th} word in article

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 (opinion)
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 Hrucey 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

NB for Text classification

- $P(\mathbf{X}|Y)$ is huge!!!
 - Article at least 1000 words, $\mathbf{X}=\{X_1,\dots,X_{1000}\}$
 - X_i represents i^{th} word in document, i.e., the domain of X_i is entire vocabulary, e.g., Webster Dictionary (or more), 10,000 words, etc.
- NB assumption helps a lot!!!
 - $P(X_i=x_i|Y=y)$ is just the probability of observing word x_i in a document on topic y

$$h_{NB}(\mathbf{x}) = \arg \max_y P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

Bag of words model

- Typical additional assumption –
 - **Position in document doesn't matter:**
 - $P(X_i=x_i | Y=y) = P(X_k=x_i | Y=y)$ (all position have the same distribution)
 - “Bag of words” model – order of words on the page ignored
 - Sounds really silly, but often works very well!

$$P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

Bag of Words Approach

the world of

TOTAL



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.

► All About The Company

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

| | |
|----------|---|
| aardvark | 0 |
| about | 2 |
| all | 2 |
| Africa | 1 |
| apple | 0 |
| anxious | 0 |
| ... | |
| gas | 1 |
| ... | |
| oil | 1 |
| ... | |
| Zaire | 0 |

NB with Bag of Words for text classification

- Learning phase:
 - Prior $P(Y)$
 - Count how many documents from each topic (prior)
 - $P(X_i|Y)$
 - For each topic, count how many times you saw word in documents of this topic (+ prior); remember this dist'n is shared across all positions i
- Test phase:
 - For each document
 - Use naïve Bayes decision rule

$$h_{NB}(\mathbf{x}) = \arg \max_y P(y) \prod_{i=1}^{LengthDoc} P(x_i|y)$$

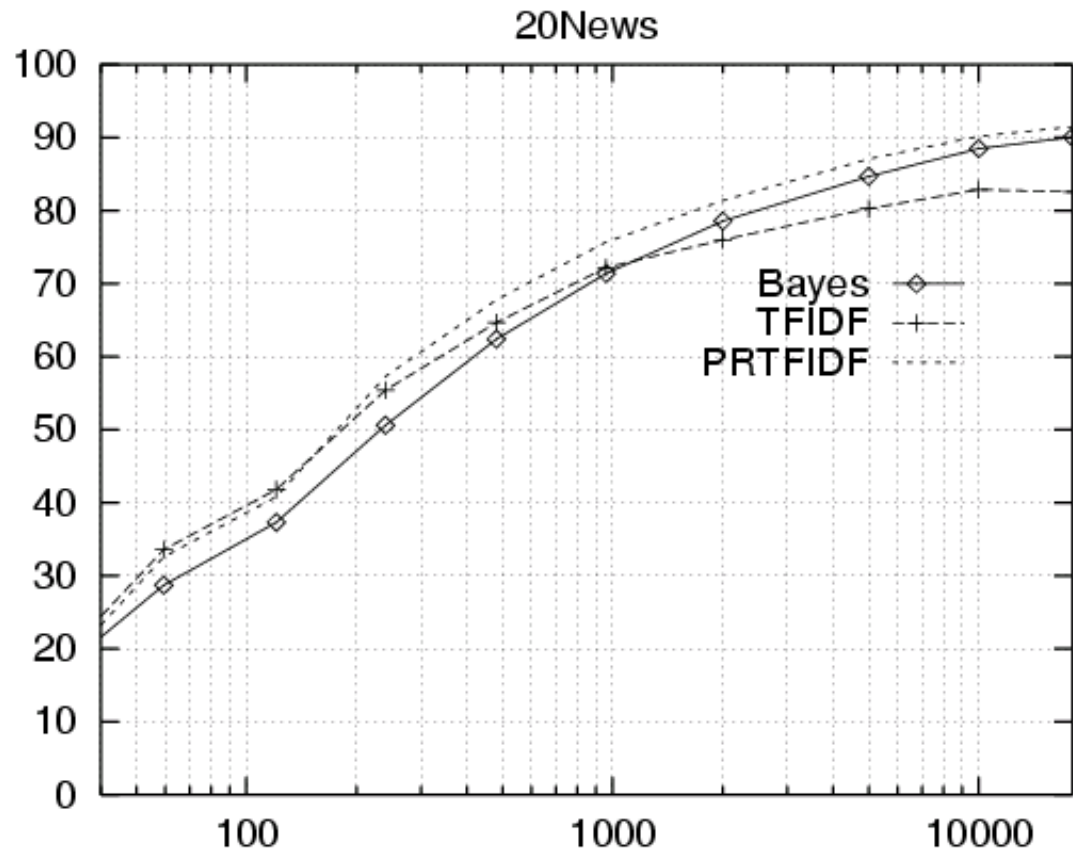
Twenty News Groups results

Given 1000 training documents from each group
Learn to classify new documents according to
which newsgroup it came from

| | |
|--------------------------|--------------------|
| comp.graphics | misc.forsale |
| comp.os.ms-windows.misc | rec.autos |
| comp.sys.ibm.pc.hardware | rec.motorcycles |
| comp.sys.mac.hardware | rec.sport.baseball |
| comp.windows.x | rec.sport.hockey |
| alt.atheism | sci.space |
| soc.religion.christian | sci.crypt |
| talk.religion.misc | sci.electronics |
| talk.politics.mideast | sci.med |
| talk.politics.misc | |
| talk.politics.guns | |

Naive Bayes: 89% classification accuracy

Learning curve for Twenty News Groups



Accuracy vs. Training set size (1/3 withheld for test)

Naïve Bayes with Log Probabilities

$$\begin{aligned}c_{MAP} &= \operatorname{argmax}_c P(c|x_1, \dots, x_n) \\&= \operatorname{argmax}_c P(c) \prod_{i=1}^n P(x_i|c) \\&= \operatorname{argmax}_c \log \left(P(c) \prod_{i=1}^n P(x_i|c) \right) \\&= \operatorname{argmax}_c \log P(c) + \sum_{i=1}^n \log P(x_i|c)\end{aligned}$$

What if we want to calculate posterior log-probabilities?

$$P(c|x_1, \dots, x_n) = \frac{P(c) \prod_{i=1}^n P(x_i|c)}{\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c')}$$

What if we want to calculate posterior log-probabilities?

$$P(c|x_1, \dots, x_n) = \frac{P(c) \prod_{i=1}^n P(x_i|c)}{\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c')}$$

$$\log P(c|x_1, \dots, x_n) = \log \frac{P(c) \prod_{i=1}^n P(x_i|c)}{\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c')}$$

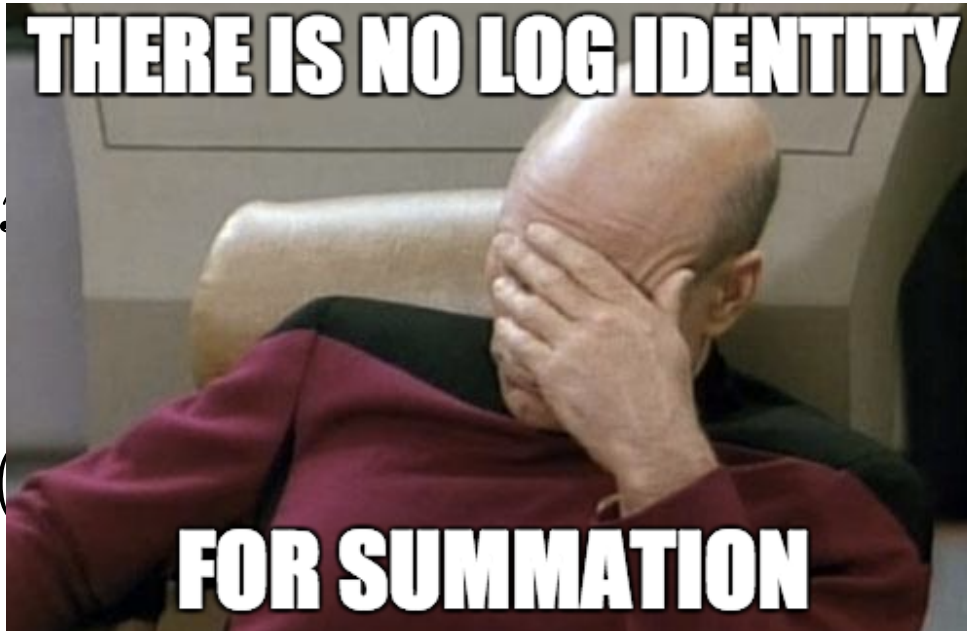
What if we want to calculate posterior log-probabilities?

$$P(c|x_1, \dots, x_n) = \frac{P(c) \prod_{i=1}^n P(x_i|c)}{\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c')}$$

$$\log P(c|x_1, \dots, x_n) = \log \frac{P(c) \prod_{i=1}^n P(x_i|c)}{\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c')}$$

$$= \log P(c) + \sum_{i=1}^n \log P(x_i|c) - \log \left[\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c') \right]$$

What if we want to calculate posterior log-probabilities?



$$P(c|s) = \frac{\prod_{i=1}^n P(x_i|c)}{\sum_{c'} \prod_{i=1}^n P(x_i|c')}$$

$$\log P(c|s) = \log P(c) + \sum_{i=1}^n \log P(x_i|c) - \log \left[\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c') \right]$$

$$= \log P(c) + \sum_{i=1}^n \log P(x_i|c) - \log \left[\sum_{c'} P(c') \prod_{i=1}^n P(x_i|c') \right]$$

Log Exp Sum Trick

- We have: a bunch of log probabilities.
 - $\log(p_1), \log(p_2), \log(p_3), \dots \log(p_n)$
- We want: $\log(p_1 + p_2 + p_3 + \dots p_n)$

Log Exp Sum Trick:

$$\log\left[\sum_i \exp(x_i)\right] = x_{max} + \log\left[\sum_i \exp(x_i - x_{max})\right]$$

What You Need to Know

- Naive Bayes Classifier
 - What assumption is made?
 - Why is this a useful assumption?
 - How to use in practice?
- Text Classification (Bag-of-Words)
- Log Probabilities
 - How to use them and why
- Log-Exp-Sum Trick