Probability Review and Naïve Bayes

Instructor: Wei Xu

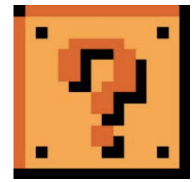
Some slides adapted from Dan Jurfasky and Brendan O'connor

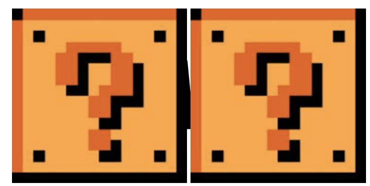
What is Probability?

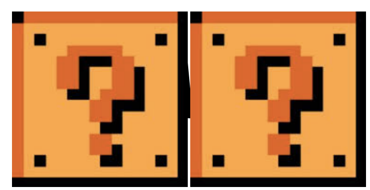
- "The probability the coin will land heads is 0.5"
  - Q: what does this mean?
- 2 Interpretations:
  - Frequentist (Repeated trials)
    - If we flip the coin many times…
  - Bayesian
    - We believe there is equal chance of heads/tails
    - Advantage: events that do not have long term

E.g. What is the probability the polar ice caps will melt by 2050?

Probability Review

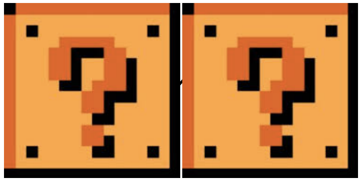
$$\sum_{x} P(X = x) =$$ 

Conditional Probability $$\frac{P(A, B)}{P(B)} =$$ 

Chain Rule $$P(A|B)P(B) =$$ 

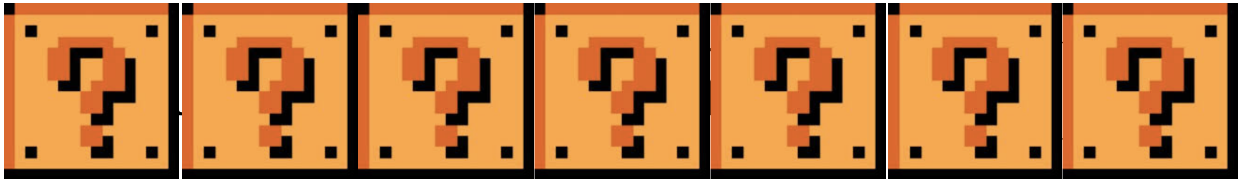
Probability Review

$$\sum_x P(X = x, Y) =$$

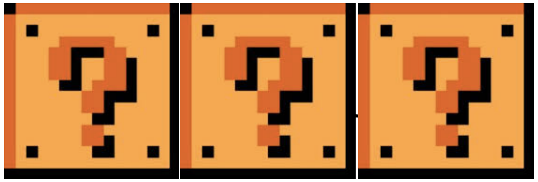


Disjunction / Union:

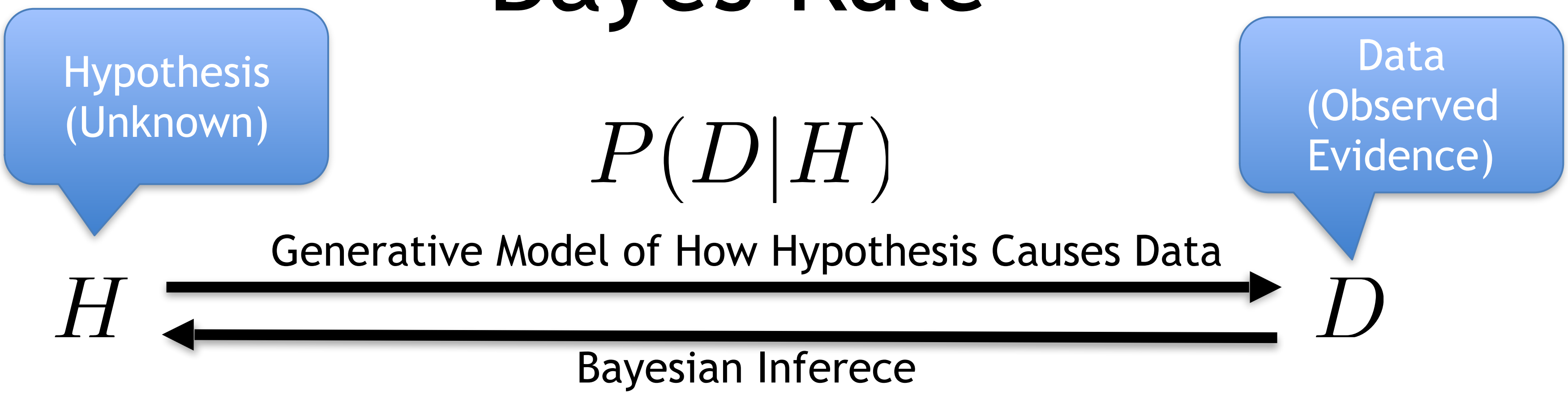
$$P(A \vee B) =$$



Negation: $P(\neg A) =$



Bayes Rule



$$P(H|D)$$

Bayes Rule tells us how to “flip” the conditional probabilities
Reason about effects to causes
Useful if you assume a generative model for your data

$$P(H|D) = \frac{P(D|H)P(H)}{P(D)}$$

Bayes Rule

Bayes Rule tells us how to "flip" the conditional probabilities
Reason about effects to causes
Useful if you assume a generative model for your data

Likelihood

Prior

$$\underbrace{P(H|D)}_{\text{Posterior}} = \frac{\overbrace{P(D|H)}^{\text{Likelihood}}\,\overbrace{P(H)}^{\text{Prior}}}{\underbrace{P(D)}_{\text{Normalizer}}}$$

Posterior

Normalizer

Bayes Rule

Bayes Rule tells us how to "flip" the conditional probabilities
Reason about effects to causes
Useful if you assume a generative model for your data

Likelihood

Prior

$$P(H|D) = \frac{P(D|H)P(H)}{\sum_h P(D|H)P(H)}$$

Posterior

Normalizer

Bayes Rule

Bayes Rule tells us how to “flip” the conditional probabilities
Reason about effects to causes
Useful if you assume a generative model for your data

Likelihood

Prior

$$P(H|D) \propto P(D|H)P(H)$$

Posterior

Proportional To
(Doesn’t sum to 1)

Bayes Rule Example

- There is a disease that affects a tiny fraction of the population (0.001%)
- Symptoms include a headache and stiff neck. 50% of patients with the disease have these symptoms
- 5% of the general population has these symptoms.

Q: Assume you have the symptom, what is your probability of having the disease?

Another Bayes Rule Example

- The well-known OJ Simpson murder trial

Another Bayes Rule Example

- The prosecution presented evidence that Simpson had been violent toward his wife, argued that a pattern of spousal abuse reflected a motive to kill.
- The defense attorney, Alan Dershowitz, argued that:
  - there was only one woman murdered for every 2500 women who were subjected to spousal abuse, and that any history of Simpson being violent toward his wife was irrelevant to the trial.
- In effect, both sides were asking the jury to consider the probability that a man murdered his ex-wife, given that he previously battered her.

What do you think?
Discuss with your neighbors

Another Bayes Rule Example

- The defense attorney, Alan Dershowitz, argued that:
  - there was only one woman murdered for every 1000 women who were subjected to spousal abuse, and that any history of Simpson being violent toward his wife was irrelevant to the trial.
- In 1994, 5000 women were murdered, 1500 by their husband. Assuming a population of 100 million women.
  - $P\ (\text{Murder} \mid \neg\text{Guilt}) = 3500/100\text{x}10^6 \approx 1/30000$

What do you have now?
Discuss with your neighbors

Text Classification

Is this Spam?

Act Fast — Junk

PUBLISHERS CLEARING HOUSE

December 22, 2015 at 11:34 PM

PC

To: Undisclosed recipients:;

Reply-To: william.kelley0@accountant.com

Act Fast

FROM THE DESK OF
DEBORAH HOLLAND
EXECUTIVE VICE PRESIDENT.
PUBLISHERS CLEARING HOUSE

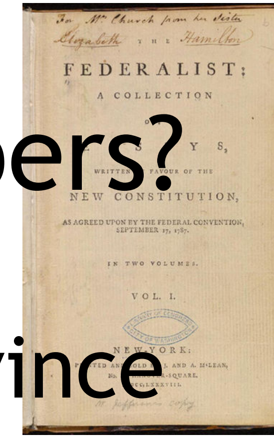
You have been declared the Publishers Clearing House mega winner of $550,000.00 in the recent on going draw held on 20th of December 2015.

For verification Contact William Kelley, Reply and fill in the infomation below ASAP .

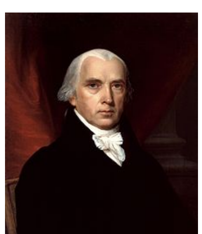
ADDRESS........
STATE.............
CITY...............
SEX................
AGE...............
CELLPHONE....

Congratulations once again.

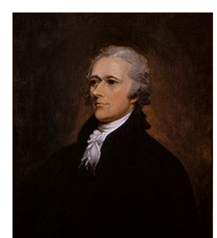
Who wrote which Federalist papers?



- 1787-8: anonymous essays try to convince New York to ratify U.S Constitution: Jay, Madison, Hamilton.
- Authorship of 12 of the letters in dispute
- 1963: solved by Mosteller and Wallace using Bayesian methods



James Madison



Alexander Hamilton

What is the subject of this article?

MEDLINE Article

Syntactic frame and verb bias in aphasia: Plausibility judgments of undergoer-subject sentences

MeSH Subject Category Hierarchy

- Antogonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- …

Positive or negative movie review?

- unbelievably disappointing

- Full of zany characters and richly applied satire, and some great plot twists

- this is the greatest screwball comedy ever filmed

- It was pathetic. The worst part about it was the boxing scenes.

Text Classification: definition

- *Input*:
  - a document d
  - a fixed set of classes $C = \{c_1, c_2, \ldots, c_J\}$

- *Output*: a predicted class $c \in C$

Classification Methods: Hand-coded rules

- Rules based on combinations of words or other features
  - spam: black-list-address OR (“dollars” AND “have been selected”)
- Accuracy can be high
  - If rules carefully refined by expert
- Running time is usually very good and fast
- But, building and maintaining these rules is expensive

Classification Methods: Supervised Machine Learning

- *Input:*
  - a document d
  - a fixed set of classes $C = \{c_1, c_2,\ldots, c_J\}$
  - A training set of m hand-labeled documents $(d_1,c_1),\ldots,(d_m,c_m)$
- *Output:*
  - a learned classifier $y{:}d \to c$

Classification Methods: Supervised Machine Learning

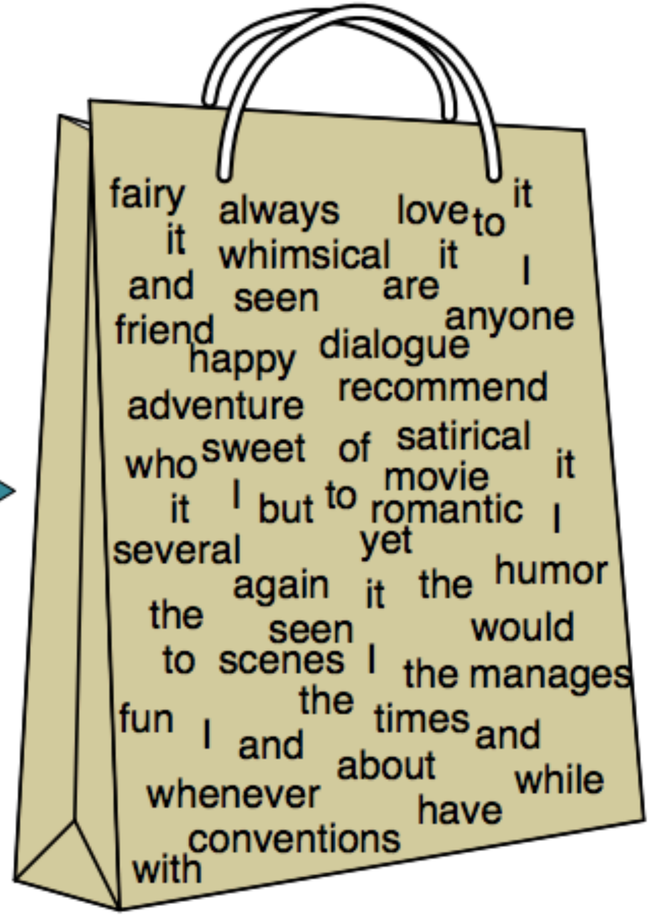
- Any kind of classifier
  - Naïve Bayes
  - Logistic regression
  - Support-vector machines
  - k-Nearest Neighbors

  - …

Naïve Bayes Intuition

- Simple ("naïve") classification method based on Bayes rule
- Relies on very simple representation of document:
  - Bag of words

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
…	…

Bag of words for document classification

Test document

parser
language
label
translation
…

?

Machine Learning	NLP	Garbage Collection	Planning	GUI
learning	parser	garbage	planning	...
training	tag	collection	temporal	
algorithm	training	memory	reasoning	
shrinkage	translation	optimization	plan	
network...	language...	region...	language...	

Bayes' Rule Applied to Documents and Classes

- For a document d and a class c

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)}$$

Naïve Bayes Classifier (I)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}}\, P(c \mid d)$$

MAP is "maximum a posteriori" = most likely class

$$= \underset{c \in C}{\operatorname{argmax}} \frac{P(d \mid c)P(c)}{P(d)}$$

Bayes Rule

$$= \underset{c \in C}{\operatorname{argmax}}\, P(d \mid c)P(c)$$

Dropping the denominator

Naïve Bayes Classifier (II)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(d \mid c) P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

Naïve Bayes Classifier (IV)

$$c_{MAP} = \operatorname*{argmax}_{c \in C} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

$O(|X|^n \bullet |C|)$ parameters

How often does this class occur?

Could only be estimated if a very, very large number of training examples was available.

We can just count the relative frequencies in a corpus

Multinomial Naïve Bayes Independence Assumptions

$$P(x_1, x_2, \ldots, x_n \mid c)$$

- **Bag of Words assumption**: Assume position doesn't matter
- **Conditional Independence**: Assume the feature probabilities $P(x_i|c_j)$ are independent given the class c.

Multinomial Naïve Bayes Classifier

$$c_{MAP} = \operatorname*{argmax}_{c \in C} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

$$c_{NB} = \operatorname*{argmax}_{c \in C} P(c_j) \prod_{x \in X} P(x \mid c)$$

Applying Multinomial Naive Bayes to Text Classification

positions $\leftarrow$ all word positions in test document

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

Multinomial Naïve Bayes: Learning

- First attempt: maximum likelihood estimates
  - simply use the frequencies in the data

$$\hat{P}(c_j) = \frac{doccount(C = c_j)}{N_{doc}}$$

$$\hat{P}(w_i \mid c_j) = \frac{count(w_i, c_j)}{\sum_{w \in V} count(w, c_j)}$$

Parameter Estimation

$$\hat{P}(w_i \mid c_j) = \frac{count(w_i, c_j)}{\sum_{w \in V} count(w, c_j)}$$

fraction of times word w_i appears among all words in documents of topic c_j

- Create mega-document for topic *j* by concatenating all docs in this topic
  - Use frequency of *w* in mega-document

Problem with Maximum Likelihood

- What if we have seen no training documents with the word ***fantastic*** and classified in the topic **positive** (***thumbs-up)***?

$$\hat{P}(\text{"fantastic"} \mid \text{positive}) = \frac{count(\text{"fantastic"}, \text{positive})}{\sum_{w \in V} count(w, \text{positive})} = 0$$

- Zero probabilities cannot be conditioned away, no matter the other evidence!

$$c_{MAP} = \text{argmax}_c \hat{P}(c) \prod_i \hat{P}(x_i \mid c)$$

Laplace (add-1) smoothing for Naïve Bayes

$$\hat{P}(w_i \mid c) = \frac{count(w_i, c)}{\sum_{w \in V} \left(count(w, c)\right)}$$

$$= \frac{count(w_i, c) + 1}{\left(\sum_{w \in V} count(w, c)\right) + |V|}$$

Multinomial Naïve Bayes: Learning

- Calculate $P(c_j)$ terms
  - For each c_j in C do

$$docs_j \leftarrow \text{all docs with class} = c_j$$

$$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$

Multinomial Naïve Bayes: Learning

- From training corpus, extract *Vocabulary*
- Calculate $P(w_k \mid c_j)$ terms
  - $Text_j \leftarrow$ single doc containing all $docs_j$
  - For each word w_k in *Vocabulary*

    $n_k \leftarrow$ # of occurrences of w_k in $Text_j$

$$P(w_k \mid c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha \,|Vocabulary|}$$

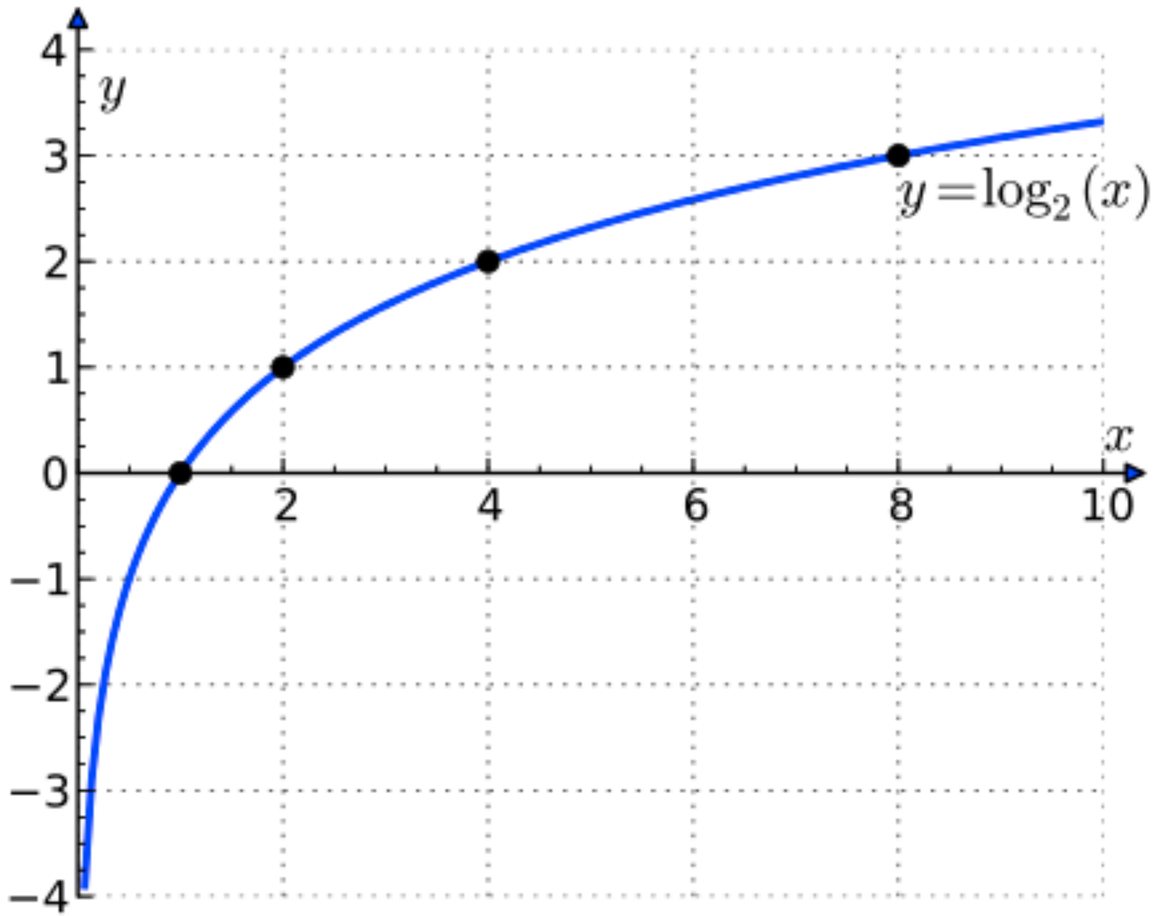
Naïve Bayes: Practical Issues

$$
\begin{aligned}
c_{MAP} &= \text{argmax}_c P(c|x_1, \ldots, x_n) \\
&= \text{argmax}_c P(x_1, \ldots, x_n|c) P(c) \\
&= \text{argmax}_c P(c) \prod_{i=1}^{n} P(x_i|c)
\end{aligned}
$$

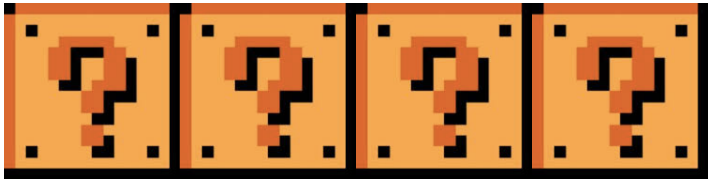
- Multiplying together lots of probabilities
- Probabilities are numbers between 0 and 1

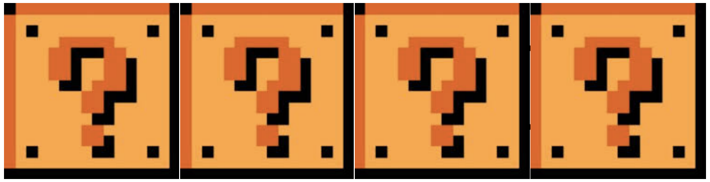
Q: What could go wrong here?

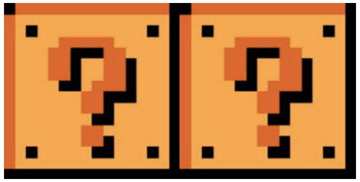
Working with probabilities in log space



Log Identities (review)

$$\log(a \times b) =$$ 

$$\log(\frac{a}{b}) =$$ 

$$\log(a^n) =$$ 

Naïve Bayes with Log Probabilities

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

Naïve Bayes with Log Probabilities

$$c_{MAP} = \operatorname{argmax}_c \log P(c) + \sum_{i=1}^{n} \log P(x_i|c)$$

- Q: Why don't we have to worry about floating point underflow anymore?

Working with probabilities in log space

x	log(x)
0.0000001	-16.118095651
0.000001	-13.815511
0.00001	-11.512925
0.0001	-9.210340
0.001	-6.907755
0.01	-4.605170
0.1	-2.302585

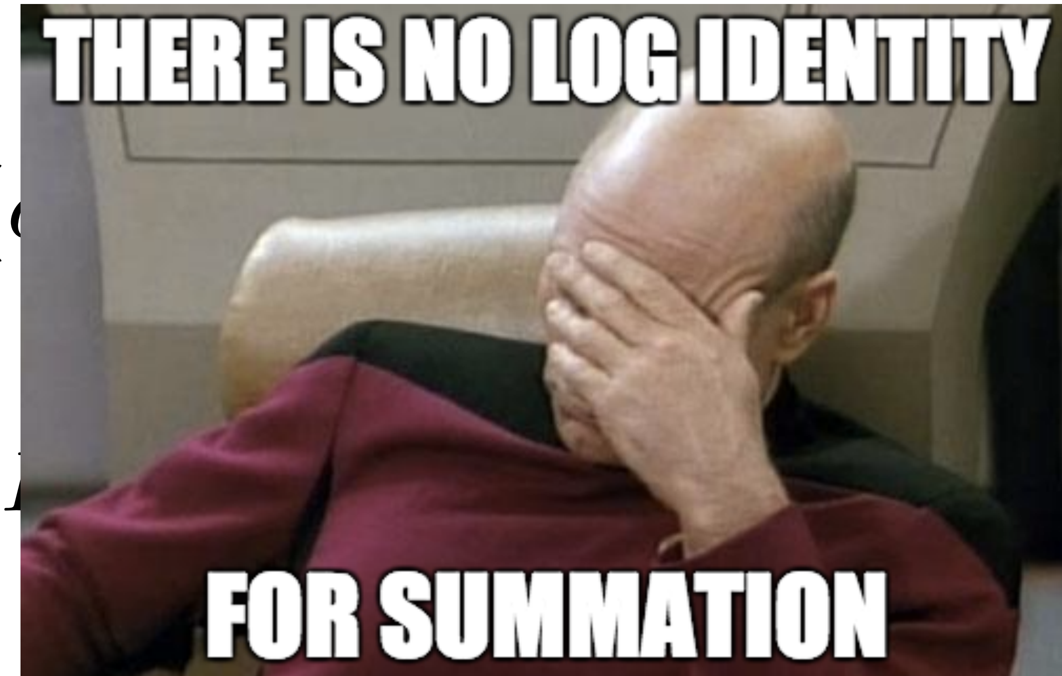
What if we want to calculate posterior log-probabilities?

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

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

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

What if we want to calculate posterior log-probabilities?



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

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

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

Log Exp Sum Trick: motivation

- We have: a bunch of log probabilities.
  - log(p1), log(p2), log(p3), … log(pn)
- We want: log(p1 + p2 + p3 + … pn)
- We could convert back from log space, sum then take the log.
  - If the probabilities are very small, this will result in floating point underflow

Log Exp Sum Trick:

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

Another issue: Smoothing

$$\hat{P}(w_i|c) = \frac{\mathrm{count}(w,c)+1}{\sum_{w' \in V} \mathrm{count(w',c)} + |V|}$$

Another issue: Smoothing

Alpha doesn't necessarily need to be 1 (hyperparmeter)

$$\hat{P}(w_i|c) = \frac{\text{count}(w,c) + \alpha}{\sum_{w' \in V} \text{count}(w',c) + \alpha|V|}$$

Another issue: Smoothing

Can think of alpha as a "pseudocount".
Imaginary number of times this word has been seen.

$$\hat{P}(w_i|c) = \frac{\text{count}(w,c) + \alpha}{\sum_{w' \in V} \text{count}(w',c) + \alpha|V|}$$

Another issue: Smoothing

$$\hat{P}(w_i|c) = \frac{\text{count}(w,c) + \alpha}{\sum_{w' \in V} \text{count}(w',c) + \alpha|V|}$$

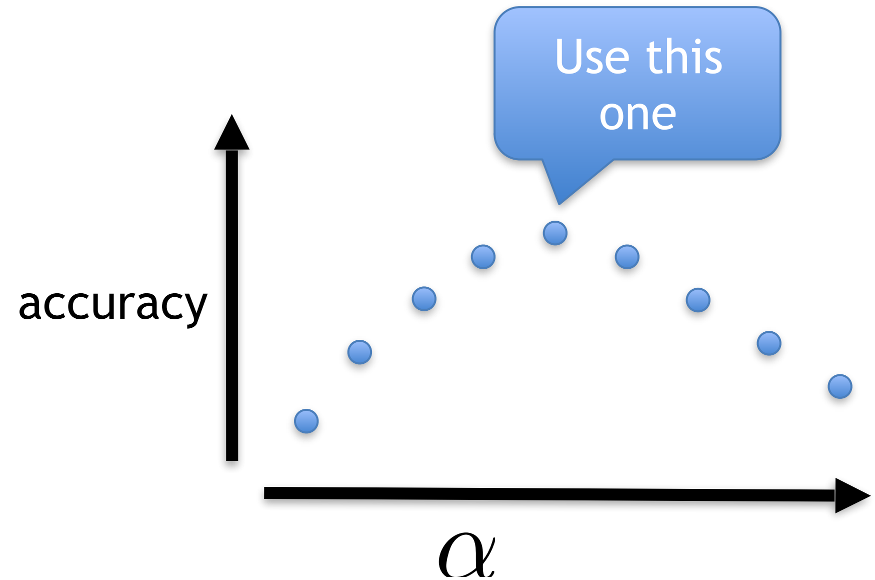
- Q: What if alpha = 0?
- Q: what if alpha = 0.000001?
- Q: what happens as alpha gets very large?

Overfitting

- Model cares too much about the training data
- How to check for overfitting?
  - Training vs. test accuracy
- Pseudocount parameter combats overfitting

Q: how to pick Alpha?

- Split train vs. Test
- Try a bunch of different values
- Pick the value of alpha that performs best
- What values to try? Grid search
  - (10^-2,10^-1,...,10^2)



Data Splitting

- Train vs. Test

- Better:
  - Train (used for fitting model **parameters**)
  - Dev (used for tuning **hyperparameters**)
  - Test (reserve for final evaluation)

- Cross-validation

Feature Engineering

- What is your word / feature representation
  - Tokenization rules: splitting on whitespace?
  - Uppercase is the same as lowercase?
  - Numbers?
  - Punctuation?
  - Stemming?