

MODIFIED BAYES' THEOREM:

$$P(H|X) = P(H) * \left(1 + P(C) * \left(\frac{P(X|H)}{P(X)} - 1 \right) \right)$$

H: HYPOTHESIS

X: OBSERVATION

P(H): PRIOR PROBABILITY THAT H IS TRUE

P(X): PRIOR PROBABILITY OF OBSERVING X

P(C): PROBABILITY THAT YOU'RE USING
BAYESIAN STATISTICS CORRECTLY

xkcd

INFO 251: Applied Machine Learning

Naïve Bayes

Announcements

- You're halfway done!
- Short quiz next Tuesday
- PS4 available

Key Concepts (last lecture)

- Logistic regression
- Simplified sigmoid cost function
- Odds ratios
- Overfitting revisited
- Support vector machines
- Hard vs. soft margins
- Kernel functions

Course Outline

- Causal Inference and Research Design
 - Experimental methods
 - Non-experiment methods
- Machine Learning
 - Design of Machine Learning Experiments
 - Linear Models and Gradient Descent
 - **Non-linear models**
 - Neural models
 - Unsupervised Learning
 - Practicalities, Fairness, Bias
- Special topics

Outline

- Rules and models
- Naïve Bayes Classifier
- Spam Example
- Smoothing
- Summary
- Mid-Semester Evaluation

This Lecture: Key Concepts

- Bayes' theorem
- Prior probability
- Conditional probability
- Posterior probability
- Log-Likelihood
- Spam classification
- Laplace smoothing

Rules vs. statistical models

■ Rules

- The first spam filters were manually curated blacklists
- Hard to maintain; caught only 25% of spam (circa 2000)

■ Models

- Instead, treat words in email as evidence
- Models allow us to combine prior knowledge with data
- Paul Graham catches 99.5% of spam in 2002 with a simple statistical ("Bayesian") model
- Our goal: understand how this model worked

Thomas Bayes

- Thomas Bayes
 - English mathematician and minister
 - 1701 - 1761
 - Wrote two books:
 - One tried to prove Newton's calculus correct
 - The other tried to prove that God is benevolent
 - But best known for his unpublished notes on probability



Is this really Bayes?

Bayes' Theorem

- Conditional probability: $P(A|B)$
 - Probability of A given that B is true
 - Recall: $P(A|B) = \frac{P(A \& B)}{P(B)}$
 - Why? Because $P(A \& B) = P(B)P(A|B)$

- Bayes Theorem:

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

Bayes' Theorem: Proof

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

1. $P(A|B) = \frac{P(A \& B)}{P(B)}$ (cond. probability)

2. $P(B|A) = \frac{P(B \& A)}{P(A)} = \frac{P(A \& B)}{P(A)}$ (cond. probability)

3. $P(A \& B) = P(B|A) * P(A)$ (rearrange 2)

4. $P(A|B) = \frac{P(B|A) P(A)}{P(B)}$ (combine 1 & 3)

Bayes' Theorem

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

- In practice we are typically dealing with hypothesis h and data D
 - h = "I have a cold"
 - D = "runny nose," "watery eyes," "coughing"
- $P(h|D) = \frac{P(D|h) P(h)}{P(D)}$
 - => Bayes' theorem is often thought of as "diagnostic"
 - Helps us understand the probability of a hypothesis, given data

Bayes' Theorem

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

- $P(h)$: **Prior probability** of h
 - What we know (or think we know) about h with no other evidence
 - Initial best guess that the hypothesis is true, based on no real data
- $P(D|h)$: **conditional probability** of D given that h happened
 - Also called the “likelihood” of D
 - Probability that cause produces effect
 - E.g. probability that nose is runny if you *are* sick
- $P(D)$: Probability of observing data, irrespective of any hypothesis,
 - Also called the “normalizing probability”
 - E.g. probability of a runny nose (for anyone, sick or healthy)
- $P(h|D)$ is the **posterior probability** of h given D
 - This is what we want
 - Note that the posterior is influenced by the prior, so Bayes' has a GIGO liability

Bayes' Theorem

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

- Who cares?
 - Idea: Update our belief about h given evidence D
- Why is this helpful?
 - Sometimes we can come up with estimates of $P(h)$ and $P(D|h)$ in situations where it's very hard to estimate $P(h|D)$

Bayes Rule: Diagnosis

- What is the probability that you have meningitis, given that your neck is stiff?
 - $P(\text{meningitis}|\text{stiff neck}) = ?$
- What we know
 - $P(\text{meningitis}) = 1/50000 \leq$ Prior probability, i.e., the “base rate”
 - $P(\text{stiff neck} | \text{meningitis}) = 1/2 \leq$ Conditional probability (effect | cause)
 - $P(\text{stiff neck}) = 1/20 \leq$ Normalizing probability
- What we can infer
 - $$\begin{aligned} P(m|s) &= P(m)P(s|m)/P(s) \\ &= (1/50000)(1/2)/(1/20) \\ &= (1/5000) \end{aligned}$$
- Got a stiff neck?
 - Chances of having meningitis increase by 10x. But still not so likely

Bayes Rule: Example Quiz Question

- Calculate the probability that an email is spam given that it contains the word "viagra"
- $P(\text{spam} | \text{"viagra"}) = ?$
 - $P(\text{spam}) = 0.4$
 - $P(\text{"viagra"}) = 0.05$
 - $P(\text{"viagra"} | \text{spam}) = 0.06$
- Answer: $(.4)(.06)/(.05) = .48$

Outline

- Mid-semester evaluations
- Rules and models
- **Naïve Bayes Classifier**
- Spam Example
- Practicalities
- Summary

Spam classification

- Input (X): an email corpus (collection)
 - $X: x_1, \dots, x_n$ (the corpus x is comprised of n emails x_i)
 - $x_i: t_1, \dots, t_k$ (each email x_i is comprised of k features t_j)
- Output (Y): [spam, ham]
- Goal: predict labels of new email
 - $P(Y=\text{spam}|\text{email}) = ?$
- In practice, the email is a vector of features
 - words
 - urls
 - sender

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.

Naïve Bayes Classifier

- We are trying to learn a function $f(x)$ that maps input data x_i , consisting of terms t_1, \dots, t_k , to predicted class \hat{Y}_i

- Most likely class is:

- $\hat{Y}_i = \operatorname{argmax}_{c_j \in C} P(c_j | t_1, \dots, t_k)$

Why can't we calculate this directly?

$$= \operatorname{argmax}_{c_j \in C} \frac{P(t_1, \dots, t_k | c_j) P(c_j)}{P(t_1, \dots, t_k)}$$

(this is Bayes' rule in action)

$$= \operatorname{argmax}_{c_j \in C} P(t_1, \dots, t_k | c_j) P(c_j)$$

(we can ignore $P(t_1, \dots, t_k)$ - same for all c_j)

This is still a problem

Naïve Bayes Classifier

- Naïve Bayes (independence) Assumption:

$$P(t_1, \dots, t_k | c_j) = \prod_{1 \leq i \leq k} P(t_i | c_j)$$

- This allows us to complete the NB Classifier:

$$\hat{Y}_i = \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j | t_1, \dots, t_k) P(c_j)$$

...

$$= \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j) \prod_{1 \leq i \leq k} P(t_i | c_j)$$

- Is this an appropriate generative model?

Classification

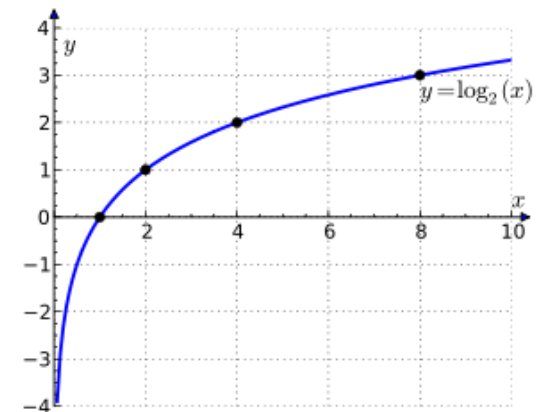
- For classification, we want the most likely class given the data. This is the “Maximum A-Posteriori (MAP) estimate”

$$\operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j) \prod_{1 \leq i \leq k} P(t_i | c_j)$$

- In practice, it's easier to work with logs:

$$\operatorname{argmax}_{c_j \in \mathcal{C}} \left[\log P(c_j) + \sum_{1 \leq i \leq k} \log P(t_i | c_j) \right]$$

- Why is it valid to work with logs?
- What do we gain by working with logs?



Naïve Bayes Classifier: Recap

- Naïve Bayes Classifier:

$$\hat{Y}_i = \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j | t_1, \dots, t_k)$$

$$\begin{aligned} &\dots \\ &= \operatorname{argmax}_{c_j \in \mathcal{C}} P(t_1, \dots, t_k | c_j) P(c_j) \end{aligned}$$

$$\begin{aligned} &\dots \\ &= \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j) \prod_{1 \leq i \leq k} P(t_i | c_j) \end{aligned}$$

$$\begin{aligned} &\dots \\ &= \operatorname{argmax}_{c_j \in \mathcal{C}} \left[\log P(c_j) + \sum_{1 \leq i \leq k} \log P(t_i | c_j) \right] \end{aligned}$$

Outline

- Mid-semester evaluations
- Rules and models
- Naïve Bayes Classifier
- **Spam Example**
- Practicalities
- Summary

Spam classification parameters

- Example parameters:

$P(c_j)$	

ham	0.6
spam	0.4

$P(t_i c_j=\text{spam})$	

the	0.0156
to	0.0153
and	0.0115
of	0.0095
you	0.0093
a	0.0086
with	0.0080
from	0.0075
...	

$P(t_i c_j=\text{ham})$	

the	0.0210
to	0.0133
of	0.0119
2002	0.0110
with	0.0108
from	0.0107
and	0.0105
a	0.0100
...	

- Example test case:

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. ...

Spam classification example

- Prediction: $\operatorname{argmax}_{c_j} \log P(c_j) + \sum_i \log P(t_i | c_j)$

Feature	$P(d_i \text{spam})$	$P(d_i \text{ham})$	Tot Spam	Tot Ham
(prior)	0.4	0.6	-0.92	-0.51
dear	0.0013	0.0009	-7.56	-7.52
sir	0.0023	0.0004	-13.64	-15.35
,	0.0220	0.0241	-17.45	-19.07
first	0.0018	0.0023	-23.77	-25.15
I	0.0062	0.0119	-28.86	-29.57
must	0.0034	0.0028	-34.54	-35.45
solicit	0.0007	0.0002	-41.08	-43.97

Naive Bayes parameter estimation

- Naive Bayes is appealing because the parameters are so easy to estimate
 - All we need are counts
 - Easy to parallelize

Naïve Bayes Algorithm

Naive_bayes_train(*examples*):

For each target c_j :

$\hat{P}(c_j) \leftarrow \text{estimate } P(c_j)$

For each attribute t_i :

$\hat{P}(t_i|c_i) \leftarrow \text{estimate } P(t_i|c_i)$

Classify_new_instance(x):

$$c_x = \operatorname{argmax}_{c_j \in C} \left[\log \hat{P}(c_j) + \sum_{1 \leq i \leq k} \log \hat{P}(t_i|c_j) \right]$$

Outline

- Mid-semester evaluations
- Rules and models
- Naïve Bayes Classifier
- Spam Example
- **Practicalities**
- Summary

Some nuances

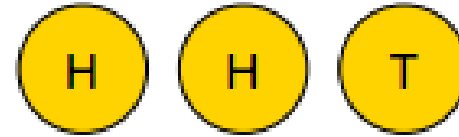
- Probabilities may not add up to 1. Why?
 - Not normalizing by likelihood of data. In practice, doesn't matter
 - What matters is relative likelihood.
- Conditional independence often violated
 - So why does NB perform so well?
 - Model is bad, classifier is good

Smoothing

- Lots of words don't occur in training set (because of power law behavior), even though those terms may occur in class documents outside the training set
 - Who cares?
- Just one of these drives the posterior estimate for that document/class to zero (or undefined), when in fact it may be a good choice
- You can avoid zero probability estimates by **smoothing**

Laplace smoothing

- Idea: Pretend we saw every outcome 1 more time than we actually did (“add-one smoothing”)
- If we observed [hht]:
 - MLE $P(\text{heads}) = 2/3$
 - LAP₁ $P(\text{heads}) = 3/5$
 - LAP₁₀₀ $P(\text{heads}) = 102/203$ (larger k = stronger smoothing)



$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V} (T_{ct'} + 1)} = \frac{T_{ct} + 1}{(\sum_{t' \in V} T_{ct'}) + B}$$

Additive Smoothing

- Additive smoothing generalizes this. In practice, it's worth trying smaller values of the additive smoothing parameter
 - Typically a positive value $\alpha < 1$ is used

$$\hat{P}(t|c) = \frac{T_{ct} + \alpha}{\sum_{t' \in V} (T_{ct'} + \alpha)} = \frac{T_{ct} + \alpha}{(\sum_{t' \in V} T_{ct'}) + B\alpha}$$

- How do we choose α ?
 - Cross-validation!

Outline

- Mid-semester evaluations
- Rules and models
- Naïve Bayes Classifier
- Spam Example
- Practicalities
- **Summary**

Naïve Bayes Classifier: Recap

- Naïve Bayes Classifier:

$$\hat{Y}_i = \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j | t_1, \dots, t_k)$$

...

$$= \operatorname{argmax}_{c_j \in \mathcal{C}} P(t_1, \dots, t_k | c_j) P(c_j)$$

...

$$= \operatorname{argmax}_{c_j \in \mathcal{C}} P(c_j) \prod_{1 \leq i \leq k} P(t_i | c_j)$$

...

$$= \operatorname{argmax}_{c_j \in \mathcal{C}} [\log P(c_j) + \sum_{1 \leq i \leq k} \log P(t_i | c_j)]$$

Naive Bayes summary

- Use Bayes Rule to get the probability of the label given the input
- The Naive Bayes assumption says that the features are independent conditional on the label
- Smoothing is important in real systems
- Use held-out data to tune features and hyper-parameters

Naïve Bayes: Pros and Cons

■ Pros

- Simple and fast. Depends only on term frequency data for the classes.
- Very well-behaved numerically. Term weights depend only on term frequency
- Easy to parallelize
- Can work very well with sparse data, where combinations of dependent terms are rare

■ Cons

- Can't model more complex patterns in the data
- Generative model not appropriate in many settings

Spam: state of the art

- Gmail and Yahoo Mail are better than 99.9% accurate
 - huge amount of training data -> good estimates for a huge vocabulary
 - use OCR to convert images to text
 - some personalization
- Remains an active area of research
 - multilingual models
 - other varieties of spam require more creative features
 - review spam
 - youtube comments, twitter, ...

For Next Class:

- Read:
 - Chapter 8 of Intro to Statistical Learning
 - Chapter 13 of Daume