CSE 446: Naïve Bayes
Winter 2012

Dan Weld

Some slides from Carlos Guestrin, Luke Zettlemoyer & Dan Klein

Today

- Gaussians
- Naïve Bayes
- Text Classification

Long Ago

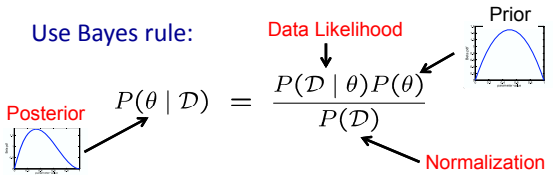
- Random variables, distributions
- Marginal, joint & conditional probabilities
- Sum rule, product rule, Bayes rule
- Independence, conditional independence

Last Time

	Prior	Hypothesis
Maximum Likelihood Estimate	Uniform	The most likely
Maximum A Posteriori Estimate	Any	The most likely
Bayesian Estimate	Any	Weighted combination

Bayesian Learning

Use Bayes rule:

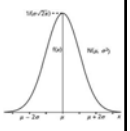


Or equivalently: $P(\theta \mid \mathcal{D}) \propto P(\mathcal{D} \mid \theta)P(\theta)$

Conjugate Priors?

Those Pesky Distributions

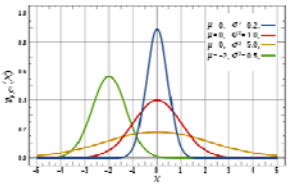
	Discrete		Continuous
	Binary {0, 1}	M Values	
Single Event	Bernouilli		
Sequence (N trials) $N= \alpha_H+\alpha_T$	Binomial $P(\mathcal{D} \mid \theta) = \theta^{\alpha_H}(1-\theta)^{\alpha_T}$	Multinomial	
Conjugate Prior	Beta	Dirichlet	



7

What about continuous variables?

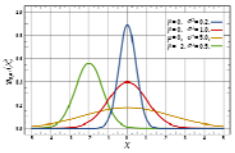
- Billionaire says: If I am measuring a continuous variable, what can you do for me?
- **You say: Let me tell you about Gaussians…**



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

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
  - $X \sim N(\mu_X,\sigma^2_X)$
  - $Y \sim N(\mu_Y,\sigma^2_Y)$
  - $Z = X+Y \rightarrow Z \sim N(\mu_X+\mu_Y, \sigma^2_X+\sigma^2_Y)$
- Can make easy to differentiate, as we will see soon!



Learning a Gaussian

X_i $i =$	Exam Score
0	85
1	95
2	100
3	12
…	…
99	89

- Collect a bunch of data
  - Hopefully, i.i.d. samples
  - e.g., exam scores
- Learn parameters
  - Mean: μ
  - Variance: σ

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

MLE for Gaussian: $P(x \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$

- Prob. of i.i.d. samples $D=\{x_1,\ldots,x_N\}$:

$$P(\mathcal{D} \mid \mu, \sigma) = \left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}$$

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

- Log-likelihood of data:

$$\ln P(\mathcal{D} \mid \mu, \sigma) = \ln\left[\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}\right]$$
$$= -N \ln \sigma\sqrt{2\pi} - \sum_{i=1}^{N} \frac{(x_i-\mu)^2}{2\sigma^2}$$

Your second learning algorithm: MLE for mean of a Gaussian

- What's MLE for mean?

$$\frac{d}{d\mu} \ln P(\mathcal{D} \mid \mu, \sigma) = \frac{d}{d\mu}\left[-N \ln \sigma\sqrt{2\pi} - \sum_{i=1}^{N} \frac{(x_i-\mu)^2}{2\sigma^2}\right]$$
$$= \frac{d}{d\mu}\left[-N \ln \sigma\sqrt{2\pi}\right] - \sum_{i=1}^{N} \frac{d}{d\mu}\left[\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$
$$= -\sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} = 0$$
$$= -\sum_{i=1}^{N} x_i + N\mu = 0$$

$$\hat{\mu}_{MLE} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

MLE for variance

- Again, set derivative to zero:

$$\frac{d}{d\sigma}\ln P(\mathcal{D}\mid\mu,\sigma) = \frac{d}{d\sigma}\left[-N\ln\sigma\sqrt{2\pi}-\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$
$$= \frac{d}{d\sigma}\left[-N\ln\sigma\sqrt{2\pi}\right]-\sum_{i=1}^{N}\frac{d}{d\sigma}\left[\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$
$$= -\frac{N}{\sigma}+\sum_{i=1}^{N}\frac{(x_i-\mu)^2}{\sigma^3} = 0$$

$$\hat{\sigma}^2_{MLE} = \frac{1}{N}\sum_{i=1}^{N}(x_i-\hat{\mu})^2$$

Learning Gaussian parameters

- MLE:

$$\hat{\mu}_{MLE} = \frac{1}{N}\sum_{i=1}^{N}x_i$$

$$\hat{\sigma}^2_{MLE} = \frac{1}{N}\sum_{i=1}^{N}(x_i-\hat{\mu})^2$$

Bayesian learning of Gaussian parameters

- Conjugate priors
  - Mean: Gaussian prior
  - Variance: Wishart Distribution

Supervised Learning of Classifiers

Find f

- Given: Training set $\{(x_i, y_i) \mid i = 1 \ldots n\}$
- 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)

Classification

Bayesian Categorization

- Let set of categories be $\{c_1, c_2, \ldots c_n\}$
- Let E be description of an instance.
- Determine category of E by determining for each c_i

$$P(c_i \mid E) = \frac{P(c_i)P(E \mid c_i)}{P(E)}$$

- P(E) can be ignored since is factor $\forall$ categories

$$P(c_i \mid E) \sim P(c_i)P(E \mid c_i)$$

20

Text classification

- Classify e-mails
  - Y = {Spam,NotSpam}
- Classify news articles
  - Y = {what is the topic of the article?}
- Classify webpages
  - Y = {Student, professor, project, …}
- What to use for features, **X**?

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
  - For email specifically, Semantic features: SenderInContacts
  - …

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial 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, Iknow 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.

Features **X** are word sequence in 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 (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
```

Features for Text Classification

- **X** is sequence of words in document
- **X** (and hence P(**X**|Y)) is huge!!!
  - Article at least 1000 words, $\mathbf{X}=\{X_1,\ldots,X_{1000}\}$
  - 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.
- $10{,}000^{1000} = 10^{4000}$
- Atoms in Universe: 10^{80}
  - We may have a problem…

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 positions have the same distribution)

– Ignore the order of words

When the lecture is over, remember to wake up the person sitting next to you in the lecture room.

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)

– Ignore the order of words

– Sounds really silly, but often works very well!

in is lecture lecture next over person remember room sitting the the the to to up wake when you

– From now on:

- X_i = Boolean: "$word_i$ is in document"
- $\mathbf{X} = X_1 \wedge \ldots \wedge X_n$

Bag of Words Approach

aardvark	0
about	2
all	2
Africa	1
apple	0
anxious	0
…	
gas	1
…	
oil	1
…	
Zaire	0

Bayesian Categorization

$P(y_1 \mid \mathbf{X}) \sim P(y_i)P(\mathbf{X}|y_i)$

- Need to know:
  - Priors: $P(y_i)$
  - Conditionals: $P(\mathbf{X} \mid y_i)$
- $P(y_i)$ are easily estimated from data.
  - If n_i of the examples in D are in y_i, then $P(y_i) = n_i / |D|$
- Conditionals:
  - $\mathbf{X} = X_1 \wedge \ldots \wedge X_n$
  - Estimate $P(X_1 \wedge \ldots \wedge X_n \mid y_i)$
- Too many possible instances to estimate!
  - *(exponential in n)*
  - Even ***with*** bag of words assumption!

Problem!

Need to Simplify Somehow

- Too many probabilities
  - $P(x_1 \wedge x_2 \wedge x_3 \mid y_i)$

$P(x_1 \wedge x_2 \wedge x_3 \mid \text{spam})$
$P(x_1 \wedge x_2 \wedge \neg x_3 \mid \text{spam})$
$P(x_1 \wedge \neg x_2 \wedge x_3 \mid \text{spam})$
....
$P(\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \mid \neg \text{spam})$

- Can we assume some are the same?
  - $P(x_1 \wedge x_2) \stackrel{?}{=} P(x_1)P(x_2)$

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.,

$$P(Thunder|Rain, Lightning) = P(Thunder|Lightning)$$

- Equivalent to:

$$P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$$

Naïve Bayes

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

$$P(X_1, X_2|Y) = P(X_1|X_2, Y)P(X_2|Y)$$
$$= P(X_1|Y)P(X_2|Y)$$

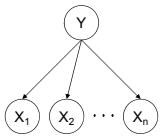
  - More generally:

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

- How many parameters now?
  - Suppose **X** is composed of *n* binary features

The Naïve Bayes Classifier

- **Given:**
  - Prior P(Y)
  - *n* conditionally independent features **X** given the class Y
  - For each X_i, we have likelihood $P(X_i|Y)$



- **Decision rule:**

$$y^* = h_{NB}(\mathbf{x}) = \arg\max_y P(y)P(x_1, \ldots, x_n \mid y)$$
$$= \arg\max_y P(y) \prod_i P(x_i|y)$$

MLE for the parameters of NB

- Given dataset, count occurrences
- MLE for discrete NB, simply:
  - Prior:

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

  - Likelihood:

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

Subtleties of NB Classifier 1
Violating the NB Assumption

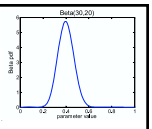
- Usually, features are not conditionally independent:

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

- Actual probabilities P(Y|**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

For Binary Features: We already know the answer!



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

- 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 $N \rightarrow 1$, prior is "forgotten"
- **But, for small sample size, prior is important!**

That's Great for Bionomial

- Works for Spam / Ham
- What about multiple classes
  - Eg, given a wikipedia page, predicting type

38

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:

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

H H T

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

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

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

Naïve Bayes for Text

- Modeled as generating a bag of words for a document in a given category by repeatedly sampling with replacement from a vocabulary $V = \{w_1, w_2, \ldots w_m\}$ based on the probabilities $P(w_j \mid c_i)$.
- Smooth probability estimates with Laplace *m*-estimates assuming a uniform distribution over all words ($p = 1/|V|$) and $m = |V|$
  - Equivalent to a virtual sample of seeing each word in each category exactly once.

40

Easy to Implement

- But…
- If you do… it probably won't work…

41