CS 760: Machine Learning
Naïve Bayes

Fred Sala

University of Wisconsin-Madison

Oct. 4, 2021

Logistics

- **Announcements**:
  - HW 3 is out, due on Oct. 12th
  - Project proposals due Oct. 14th

- Class roadmap:

Tuesday, Oct. 5	Naive Bayes
Thursday, Oct. 7	Neural Networks I
Tuesday, Oct. 12	Neural Networks II
Thursday, Oct. 14	Neural Networks III
Tuesday, Oct. 19	Neural Networks IV

Supervised Learning

Outline

- **Generative and Discriminative Models**
  - Comparison, MAP vs MLE
- **Naïve Bayes**
  - Motivation, Training, Inference, Smoothing
- **Naïve Bayes Examples**
  - Bernoulli, Multiclass, Gaussian

Outline

- **Generative and Discriminative Models**
  - Comparison, MAP vs MLE
- **Naïve Bayes**
  - Motivation, Training, Inference, Smoothing
- **Naïve Bayes Examples**
  - Bernoulli, Multiclass, Gaussian

Supervised Learning: Review

Problem setting

- Set of possible instances $\mathcal{X}$
- Unknown *target function* $f : \mathcal{X} \to \mathcal{Y}$
- Set of *models* (a.k.a. *hypotheses*) $\mathcal{H} = \{h | h : \mathcal{X} \to \mathcal{Y}\}$

Get

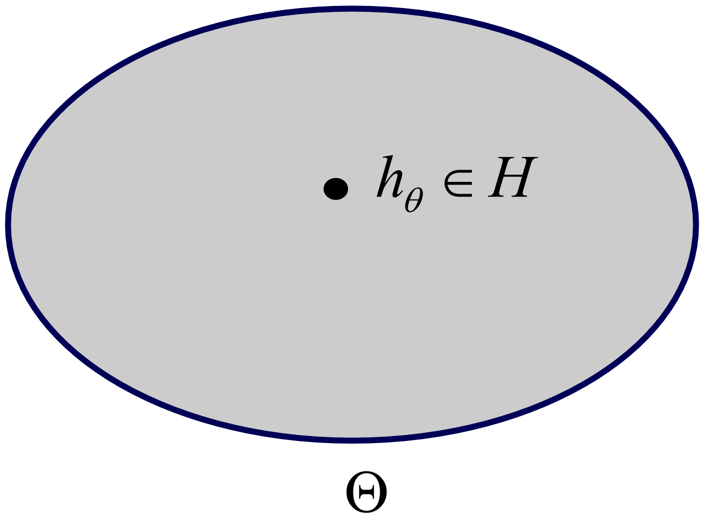
- Training set of instances for unknown target function *f*,

$$(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})$$

Goal: model *h* that best approximates *f*

Parametric Learning

- A way to categorize learning techniques
  - Parametric: hypotheses indexed by a **parameter**
  - Learning: find parameter yielding model that best approximates the target
  - **Ex**: linear models, neural networks



- Nonparametric methods:
  - Instance-based methods (KNN)
  - Decision trees

Discriminative Models

- **Idea**: hypothesis h directly predicts the label (given features)
  - y = h(x) or p(y|x) = h(x)

- We saw this already in linear regression & logistic regression
  - Linear regression:

$$h_\theta(x) = \sum_{i=0}^{d} \theta_i x_i$$

  - Logistic regression:

$$P_\theta(y = 1|x) = \sigma(\theta^T x) = \frac{1}{1 + \exp(-\theta^T x)}$$

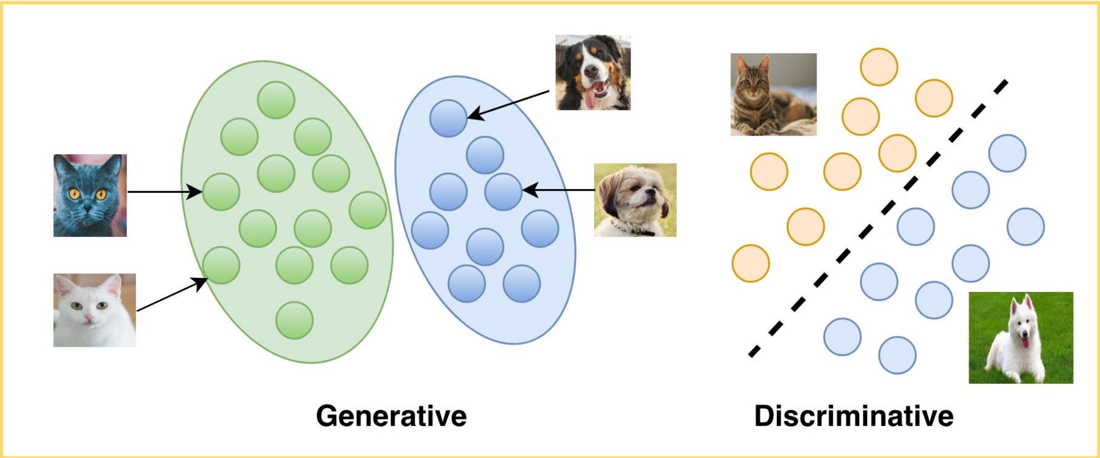
Generative Models

- Hypothesis h specifies a **generative story** for how the data was created
  - $h(x,y) = p(x,y)$ or $h(x) = p(x)$ ← **Note: supervised or unsupervised**

- Select a hypothesis via ML (or MAP)
  - Ex: roll a die. Weights for each side define data generation
  - Observe training data to learn hypothesis

Discriminative vs **Generative**

- Can define both for supervised/unsupervised learning
  - k-means (discriminative-like) vs mixture-of-Gaussians (generative)
- When should we use one over the other?
  - Discussed next



LearnOpenCV

- Typical examples:
  - Discriminative: linear regression, logistic regression, SVM, many neural networks (not all!)
  - Generative: Naïve Bayes, Bayesian Networks, …

Review: **Maximum Likelihood**

- For some set of data, find the parameters that maximize the likelihood / log-likelihood

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta; X)$$

- Example: suppose we have n samples from a Bernoulli distribution

$$P_\theta(X = x) = \begin{cases} \theta & x = 1 \\ 1 - \theta & x = 0 \end{cases}$$

Then,

$$\mathcal{L}(\theta; X) = \prod_{i=1}^{n} P(X = x_i) = \theta^k (1 - \theta)^{n-k}$$

Review: **Maximum Likelihood**

- For some set of data, find the parameters that maximize the likelihood / log-likelihood
- Example: exponential distribution

  - pdf of Exponential(λ): $f(x) = \lambda e^{-\lambda x}$
  - Suppose $X_i \sim$ Exponential(λ) for $1 \leq i \leq N$.
  - Find MLE for data $\mathcal{D} = \{x^{(i)}\}_{i=1}^{N}$
  - First write down log-likelihood of sample.
  - Compute first derivative, set to zero, solve for λ.
  - Compute second derivative and check that it is concave down at λ^{MLE}.

Review: **Maximum Likelihood**

- Example: exponential distribution
  - First write down log-likelihood of sample.

$$\ell(\lambda) = \sum_{i=1}^{N} \log f(x^{(i)}) \quad (1)$$

$$= \sum_{i=1}^{N} \log(\lambda \exp(-\lambda x^{(i)})) \quad (2)$$

$$= \sum_{i=1}^{N} \log(\lambda) + -\lambda x^{(i)} \quad (3)$$

$$= N \log(\lambda) - \lambda \sum_{i=1}^{N} x^{(i)} \quad (4)$$

Review: **Maximum Likelihood**

- Example: exponential distribution

  - Compute first derivative, set to zero, solve for λ.

$$\frac{d\ell(\lambda)}{d\lambda} = \frac{d}{d\lambda} N \log(\lambda) - \lambda \sum_{i=1}^{N} x^{(i)} \quad (1)$$

$$= \frac{N}{\lambda} - \sum_{i=1}^{N} x^{(i)} = 0 \quad (2)$$

$$\boxed{\Rightarrow \lambda^{\text{MLE}} = \frac{N}{\sum_{i=1}^{N} x^{(i)}}} \quad (3)$$

Another Approach: **Bayesian Inference**

- Let's consider a different approach
- Need a little bit of terminology

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

- *H* is the hypothesis
- *E* is the evidence

Bayesian Inference Definitions

- Terminology:

$$P(H|E) = \frac{P(E|H){\color{red}P(H)}}{P(E)}$$

Prior

- Prior: estimate of the probability **without** evidence

Bayesian Inference Definitions

- Terminology:

$$P(H|E) = \frac{\overbrace{P(E|H)}^{\text{Likelihood}}P(H)}{P(E)}$$

- Likelihood: probability of evidence **given a hypothesis**.
  - Compare to the way we defined the likelihood earlier

Bayesian Inference Definitions

- Terminology:

$$\underset{\textbf{Posterior}}{\underset{\uparrow}{P(H|E)}} = \frac{P(E|H)P(H)}{P(E)}$$

- Posterior: probability of hypothesis **given evidence.**

MAP Definition

- Suppose we think of the parameters as random variables
  - There is a prior

- Then, can do learning as Bayesian inference
  - "Evidence" is the data

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{P(X)}$$

  - **Maximum a posteriori probability** (**MAP**) estimation

$$\theta^{\mathrm{MAP}} = \arg\max_{\theta} \prod_{i=1}^{n} p(x^{(i)}|\theta)p(\theta)$$

MAP vs ML

- What's the difference between ML and MAP?

$$\theta^{\mathrm{MLE}} = \arg\max_{\theta} \prod_{i=1}^{n} p(x^{(i)}|\theta)$$

$$\theta^{\mathrm{MAP}} = \arg\max_{\theta} \prod_{i=1}^{n} p(x^{(i)}|\theta)p(\theta)$$

  - Prior!

Break & Quiz

Outline

- **Generative and Discriminative Models**
  - Comparison, MAP vs MLE
- **Naïve Bayes**
  - Motivation, Training, Inference, Smoothing
- **Naïve Bayes Examples**
  - Bernoulli, Multiclass, Gaussian

Application: Parody Detection

- The Economist
- The Onion

La paralización

Spain may be heading for its third election in a year

All latest updates

Stubborn Socialists are blocking Mariano Rajoy from forming a centre-right government

Sep 5th 2016 | MADRID | Europe

Timekeeper · Like 80 · Tweet

BACK in June, after Spain's second indecisive election in six months, the general expectation was that Mariano Rajoy, the prime minister, would swiftly form a new government. Although his conservative People's Party (PP) did not win back the absolute majority it had lost in December, it remained easily the largest party, with 137 of the 350

★ ELECTION 2016 ★

MORE ELECTION COVERAGE ▸

Tim Kaine Found Riding Conveyor Belt During Factory Campaign Stop

NEWS IN BRIEF

August 23, 2016

VOL 52 ISSUE 33

Politics · Politicians · Election 2016 · Tim Kaine

AIKEN, SC—Noting that he disappeared for over an hour during a campaign stop meet-and-greet with workers at a Bridgestone tire manufacturing plant, sources confirmed Tuesday that Democratic vice presidential candidate Tim Kaine was finally discovered riding on one of the factory's conveyor belts. "Shortly after we arrived, Tim managed to get out of our sight, but after an extensive search of the facilities, one of our interns found him moving down the assembly line between several radial tires," said senior campaign advisor Mike Henry, adding that Kaine could be seen smiling and laughing as

Model 0: Not-Naïve Model

Generative story:

1. Flip a weighted coin (Y)
2. If heads, sample a document ID (X) from the Spam distribution
3. If tails, sample a document ID (X) from the Not-Spam distribution

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

Model 0: Not-Naïve Model

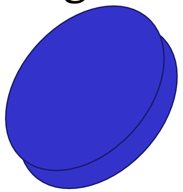
Generative story:

1. Flip a weighted coin (Y)
2. If heads, roll the **yellow** many sided die to sample a document vector ($\boldsymbol{X}$) from the Spam distribution
3. If tails, roll the **blue** many sided die to sample a document vector ($\boldsymbol{X}$) from the Not-Spam distribution

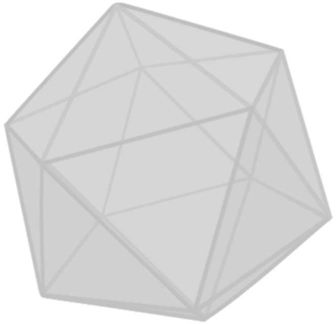
$$P(X_1, \ldots, X_K, Y) = P(X_1, \ldots, X_K | Y) P(Y)$$

Model 0: Not-Naïve Model

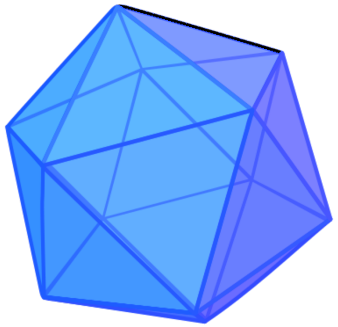
Flip weighted coin



If HEADS, roll yellow die



If TAILS, roll blue die



y	x_1	x_2	x_3	…	x_K
0	1	0	1	…	1
1	0	1	0	…	1
1	1	1	1	…	1
0	0	0	1	…	1
0	1	0	1	…	0
1	1	0	1	…	0

Each side of the die is labeled with a document vector (e.g. [1,0,1,…,1])

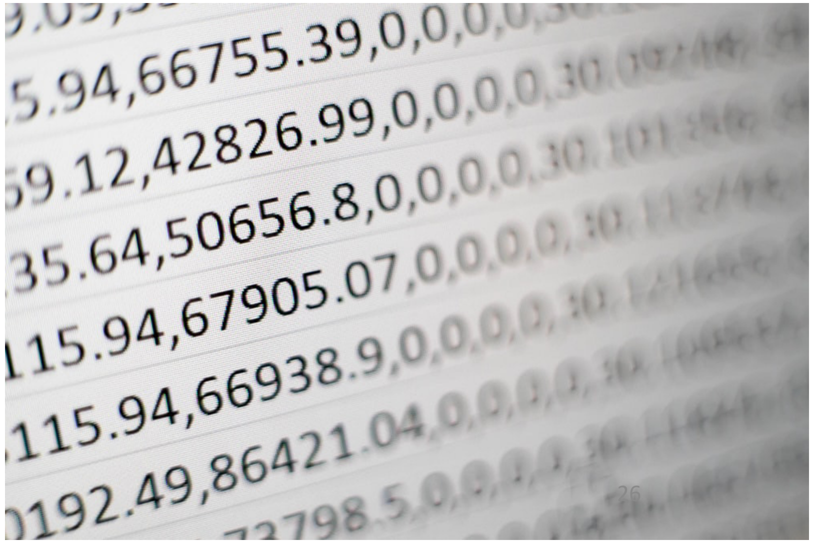
Model 0: Main Problem

How many terms are we modeling?

- Say features are binary: $X_i \in \{0, 1\}$

$$P(X_1, \ldots, X_K | Y)$$

- 2^k choices of feature vector, each gets its own probability…
  - Exponentially big table (in feature vector size)



Naïve Bayes: Core Assumption

How do we fix this problem?

- Conditional **independence** of features:

$$P(X_1, \ldots, X_K, Y) = P(X_1, \ldots, X_K | Y) P(Y)$$
$$= \left(\prod_{k=1}^{K} P(X_k | Y) \right) P(Y)$$

  - What do we gain? With binary features, get 2 entries per feature
  - So, number of probabilities

$$2^k \to 2k$$

Naïve Bayes: Overall Model

Support: Depends on the choice of **event model,** $P(X_k|Y)$

Model: Product of **prior** and the event model

$$P(\mathbf{X}, Y) = P(Y) \prod_{k=1}^{K} P(X_k|Y)$$

Training: Find the **class-conditional** MLE parameters

For $P(Y)$, we find the MLE using the data. For each $P(X_k|Y)$ we condition on the data with the corresponding class.

Prediction: Find the class that maximizes the posterior

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\mathbf{x})$$

Naïve Bayes: Training

- **Training**: empirically estimate the probabilities
  - Store: conditional probability tables (CPTs)
  - Suppose $A \perp B | C$

Independence

  - Need to estimate:

C	P(C)
0	0.33
1	0.67

B	C	P(B|C)
0	0	0.1
0	1	0.9
1	0	0.9
1	1	0.1

A	C	P(A|C)
0	0	0.2
0	1	0.5
1	0	0.8
1	1	0.5

Naïve Bayes: Smoothing

- **Training**: empirically estimate the probabilities
  - We're just obtaining counts to estimate P(B|C)
  - Suppose b has k possible values, and our counts are $b_1,\ldots,b_k$
  - **What if $b_i = 0$?**
    - Predictions will end up being zero… not ideal

  - Solution: smooth!

Smoothing parameter

$$\hat{P}(B|C) = \frac{b_i + \alpha}{N + \alpha k}$$

Points with class C

Naïve Bayes: Predicting

- With conditional probabilities, how to predict?

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\mathbf{x}) \quad \text{(posterior)}$$

$$= \underset{y}{\operatorname{argmax}}\, \frac{p(\mathbf{x}|y)p(y)}{p(x)} \quad \text{(by Bayes' rule)}$$

$$= \underset{y}{\operatorname{argmax}}\, p(\mathbf{x}|y)p(y)$$

Break & Quiz

Outline

- **Generative and Discriminative Models**
  - Comparison, MAP vs MLE
- **Naïve Bayes**
  - Motivation, Training, Inference, Smoothing
- **Naïve Bayes Examples**
  - Bernoulli, Multiclass, Gaussian

Naïve Bayes Example 1: Bernoulli

Support: Binary vectors of length K

$$\mathbf{x} \in \{0,1\}^K$$

Generative Story:

$$Y \sim \text{Bernoulli}(\phi)$$

$$X_k \sim \text{Bernoulli}(\theta_{k,Y}) \ \forall k \in \{1, \ldots, K\}$$

Model:

$$\begin{aligned} p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x}, y) &= p_{\phi,\boldsymbol{\theta}}(x_1, \ldots, x_K, y) \\ &= p_\phi(y) \prod_{k=1}^{K} p_{\boldsymbol{\theta}_k}(x_k | y) \\ &= (\phi)^y (1-\phi)^{(1-y)} \prod_{k=1}^{K} (\theta_{k,y})^{x_k} (1-\theta_{k,y})^{(1-x_k)} \end{aligned}$$

Naïve Bayes Example 1: Bernoulli

Support: Binary vectors of length K

$$\mathbf{x} \in \{0,1\}^K$$

Generative Story:

$$Y \sim \text{Bernoulli}(\phi)$$

$$X_k \sim \text{Bernoulli}(\theta_{k,Y}) \ \forall k \in \{1, \ldots, K\}$$

Same as Generic
Naïve Bayes

Model: $p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x}, y) = (\phi)^y (1-\phi)^{(1-y)} \prod_{=1}$

Classification: Find the class that maximizes the posterior

$$\hat{y} = \underset{y}{\operatorname{argmax}}\, p(y|\mathbf{x})$$

Training Bernoulli Naïve Bayes

- Recall: train (by MLE) is to find **class-conditional** parameters

- To find P(Y): use all the data
  - For $P(X_i|Y=y)$: use the data for that class

$$\phi = \frac{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)}{N}$$

$$\theta_{k,0} = \frac{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0 \wedge x_k^{(i)} = 1)}{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 0)}$$

$$\theta_{k,1} = \frac{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1 \wedge x_k^{(i)} = 1)}{\sum_{i=1}^{N} \mathbb{I}(y^{(i)} = 1)}$$

$$\forall k \in \{1, \ldots, K\}$$

Naïve Bayes Example 2: Multinomial

Integer vector (word IDs)

$\mathbf{x} = [x_1, x_2, \ldots, x_M]$ where $x_m \in \{1, \ldots, K\}$ a word id.

Generative Story:

for $i \in \{1, \ldots, N\}$**:**

$y^{(i)} \sim$ Bernoulli(ϕ)

for $j \in \{1, \ldots, M_i\}$**:**

$x_j^{(i)} \sim$ Multinomial$(\boldsymbol{\theta}_{y^{(i)}}, 1)$

Model:

$$p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x}, y) = p_\phi(y) \prod_{k=1}^{K} p_{\boldsymbol{\theta}_k}(x_k|y)$$

$$= (\phi)^y (1-\phi)^{(1-y)} \prod_{j=1}^{M_i} \theta_{y,x_j}$$

Naïve Bayes Example 3: Gaussian

Support: $\mathbf{x} \in \mathbb{R}^K$

Model: Product of **prior** and the event model

$$p(\boldsymbol{x}, y) = p(x_1, \ldots, x_K, y)$$
$$= p(y) \prod_{k=1}^{K} p(x_k | y)$$

Gaussian Naive Bayes assumes that $p(x_k|y)$ is given by a Normal distribution.

Thanks Everyone!

Some of the slides in these lectures have been adapted/borrowed from materials developed by Mark Craven, David Page, Jude Shavlik, Tom Mitchell, Nina Balcan, Elad Hazan, Tom Dietterich, Pedro Domingos, Jerry Zhu, Yingyu Liang, Volodymyr Kuleshov