CS 760: Machine Learning
Naïve Bayes

Josiah Hanna

University of Wisconsin-Madison

October 3, 2023

1

Announcements

- HW3 due next Tuesday (Oct 10)

- **Midterm**
  - Will be in-person
  - Wednesday 18th October at 5:45 pm.
  - 132 Noland Hall
  - Alternatives only for last minute emergencies.

Learning Outcomes

At the end of today's lecture, you will be able to:

1. Contrast discriminative and generative models.
2. Contrast maximum likelihood and maximum a poster learning.
3. Explain the modeling assumptions for a Naïve Bayes Classifier.

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
- Unknown *target function*
- Set of *models* (a.k.a. *hypotheses*)

$$\mathcal{X}$$
$$f : \mathcal{X} \to \mathcal{Y}$$
$$\mathcal{H} = \{h | h : \mathcal{X} \to \mathcal{Y}\}$$

Get

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

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

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

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

- (Typically) probabilistic model which states how the data was **generated.**
  h(x,y) = p(x,y) or h(x) = p(x) ⟵———————— **Note: supervised or unsupervised**

- Select a hypothesis via MLE, MAP etc.
- Use Bayes' rule to determine predictions

- In *discriminative probabilistic models*, we model how the labels y were generated conditioned on the features x. We do not usually model p(x).

Probabilistic models: Generative vs Discriminative

Generative models	Discriminative models
Can be used for both supervised and unsupervised learning.	Typically used only for supervised learning.
Specifies a probabilistic model for how data was generated. $p(X)$ for unsupervised learning, $p(X, Y)$ for supervised learning.	Specifies a probabilistic model for how the labels were generated conditioned on the features for supervised learning, i.e $p(Y|X)$.
In supervised learning, we may model *(i)* $p(X, Y)$ jointly, *(ii)* $p(Y)$ first and then $p(X|Y)$, or *(iii)* $p(X)$ first and then $p(Y|X)$.	In supervised learning, we always model $p(Y|X)$.
Use MLE, MAP etc to estimate model parameters.	

Review: **Maximum Likelihood Estimation (MLE)**

- 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, if k of the n samples are 1

$$\mathcal{L}(\theta; X) = \prod_{i=1}^{n} P(X = x_i) = \theta^k (1 - \theta)^{n-k} \implies \hat{\theta}_{\text{MLE}} = \frac{k}{n}$$

MLE Example (do at home)

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

- Example: exponential distribution

Pdf of Exponential: $f(x; \lambda) = \lambda e^{-x\lambda}$

Data: $X_i \sim f(\,\cdot\,; \lambda)$ for $1 \leq i \leq N$.

Find MLE for data $\mathscr{D} = \{x_i\}_{i=1}^{N}$

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 solution.

    - Ensures you have a maximum.

MLE Example (do at home)

- Example: exponential distribution

  - First write down log-likelihood of sample.

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

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

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

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

MLE Example (do at home)

- Example: exponential distribution

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

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

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

$$\Rightarrow \lambda^{\mathsf{MLE}} = \frac{N}{\sum_{i=1}^{N} x^{(i)}} \qquad (3)$$

Another Approach: **Bayesian Inference**

- Let us 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:

$$P(H|E) = \frac{P(E|H)\textcolor{red}{P(H)}}{P(E)} \longleftarrow \textbf{Prior}$$

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

Bayesian Inference Definitions

- Terminology:

Likelihood

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

- Likelihood: probability of evidence **given a hypothesis**.

Bayesian Inference Definitions

- Terminology:

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

↑

Posterior

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

Maximum a Posteriori (MAP) Estimation

- We treat the parameters of a model as random variables with a *prior* probability distribution.

- Then, treat 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 is 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

Q: Are these statements true or false?

(A) Generative methods model joint probability distribution while discriminative methods model conditional probabilities of Y given X.

(B) We always train a discriminative model by maximizing the conditional probability of the true labels in supervised tasks.

1. True, True
2. True, False
3. False, True
4. False, False

Q: Are these statements true or false?

(A) Generative methods model joint probability distribution while discriminative methods model conditional probabilities of Y given X.

(B) We always train a discriminative model by maximizing the conditional probability of the true labels in supervised tasks.

1. True, True
2. True, False ⬅
3. False, True
4. False, False

(At home practice)

Show that ridge regression gives the MAP estimate for θ under the model $y \sim \mathcal{N}(\theta^\top x, \sigma^2)$ and $\theta \sim \mathcal{N}(0, \tau^2 I)$ when $\lambda = \sigma^2/\tau^2$.

Bonus practice: Now show that Lasso regression gives the MAP estimate under a Laplace prior for θ.

(At home practice)

Show that ridge regression gives the MAP estimate for θ under the model $y \sim \mathcal{N}(\theta^\top x, \sigma^2)$ and $\theta \sim \mathcal{N}(0, \tau^2 I)$ when $\lambda = \sigma^2 / \tau^2$.

Data: $\mathcal{D} = \{(x_i, y_y)\}_{i=1}^n$

Write down the log-likelihood posterior: $\mathcal{L}(\mathcal{D}, \theta) = \log p(\theta) + \sum_{i=1}^n \log p(y \,|\, x; \theta)$

$$\arg\max_\theta \mathcal{L}(\mathcal{D}, \theta) = \arg\max_\theta \log C + \frac{-1}{2}(\theta^\top \Sigma \theta) + \sum_{i=1}^n \frac{-1}{2\sigma^2}(y_i - \theta^\top x_i)^2$$

$$= \arg\min_\theta \frac{1}{2\tau^2}||\theta||_2^2 + \sum_{i=1}^n \frac{1}{2\sigma^2}(y_i - \theta^\top x_i)^2$$

$$= \arg\min_\theta \frac{\sigma^2}{\tau^2}||\theta||_2^2 + \sum_{i=1}^n (y_i - \theta^\top x_i)^2 = \arg\min_\theta \lambda ||\theta||_2^2 + \sum_{i=1}^n (y_i - \theta^\top x_i)^2$$

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

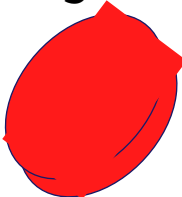# Model 0: Not-Naïve Model

Generative model:

1. Flip a weighted coin (Y)

2. If heads, sample a document (X) from the Parody distribution

3. If tails, sample a document (X) from the Not-Parody distribution

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

4. We first sample Y, then sample X given Y.

Model 0: Not-Naïve Model using Bag of words representation

Flip weighted coin

If HEADS, roll
gray die
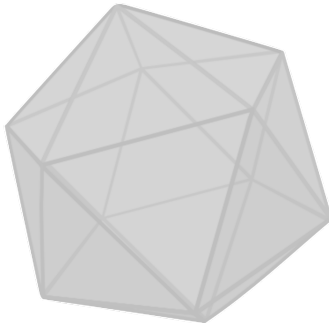
If TAILS, roll
blue die

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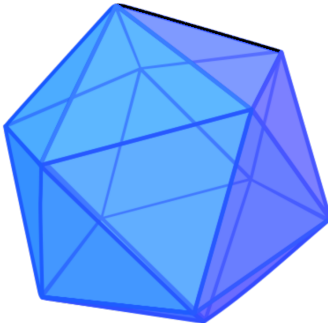
Each side of the
die is labeled with
a document vector
(e.g. [1,0,1,...,1])

Model 0: Not-Naïve Model using Bag of words representation

Generative model:

1. Flip a weighted coin (Y)

2. If heads, roll the **gray** many sided die to sample a bag of words document vector (X) from the Parody distribution

3. If tails, roll the **blue** many sided die to sample a bag of words document vector (X) from the Not-Parody distribution

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

Model 0: Main Problem

How many terms are we modeling?

- Say K is the number of words in a dictionary and the features are binary (a word exists or not): $X_i \in \{0, 1\}$

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

- 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?

- **Assume:** 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 $\quad 2^k \to 2k$

Break & Quiz

Q2-1: Are these statements true or false?

(A) Naïve Bayes assumes conditional independence of features to decompose the joint probability into the conditional probabilities.

(B) We can use Naïve Bayes' to reduce model complexity which helps with over-fitting

1. True, True
2. True, False
3. False, True
4. False, False

Q2-1: Are these statements true or false?

(A) Naïve Bayes assumes conditional independence of features to decompose the joint probability into the conditional probabilities.

(B) We can use Naïve Bayes' to reduce model complexity which helps with over-fitting

1. True, True ⬅
2. True, False
3. False, True
4. False, False

(A) Just as we learnt in the lecture.
(B) True, since the fully-fledged joint model subsumes the conditionally independent model.

Outline

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

Naïve Bayes: Overall Model

Support: Depends on the problem, $P(X_k|Y)$

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: Predicting

- With conditional probabilities, how to predict?

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

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

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

Naïve Bayes Example 1: Bernoulli

Support: Binary vectors of length K

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

Generative Model:

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

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

Joint probability :

$$p_{\phi,\boldsymbol{\theta}}(\boldsymbol{x},y) = p_{\phi,\boldsymbol{\theta}}(x_1,\ldots,x_K,y)$$

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

$$= (\phi)^y (1-\phi)^{(1-y)} \prod_{k=1}^{K} (\theta_{k,y})^{x_k} (1-\theta_{k,y})^{(1-x_k)}$$

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

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

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

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

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

Training Bernoulli Naïve Bayes

- **Training**: empirically estimate the probabilities
  - Store: conditional probability tables (CPTs)

Y	P(Y)
0	0.33
1	0.67

X_1	Y	$P(X_1|Y)$
0	0	0.1
0	1	0.9
1	0	0.9
1	1	0.1

X_2	Y	$P(X_2|C)$
0	0	0.2
0	1	0.5
1	0	0.8
1	1	0.5

Naïve Bayes Example 2: Multinomial

Support: multinomial vectors of length d

Generative model: (for each data point)

- Generate label:

$$y \sim \text{Mult}(\phi), \quad \text{where} \quad \sum_{k=1}^{K} \phi_k = 1$$

- For each feature,

$$x_i \sim \text{Mult}(\theta_{i,y}), \quad \text{where} \quad \sum_{k=1}^{K_i} \theta_{i,y,k} = 1$$

Joint probability:

$$p_{\phi,\theta}(x, y) = p_\phi(y) \prod_{i=1}^{d} p_{\theta_{i,y}}(x_i|y) = \phi_y \prod_{i=1}^{d} \theta_{i,y,x_i}$$

Training & Smoothing

- **Training**: empirically estimate the probabilities
  - We are just obtaining counts to estimate $P(X_i|Y)$
  - Suppose X_i has K possible values, and our counts for Y=y are $b_1,\ldots,b_K$
  - **What if b_i = 0?**
    - Predictions will end up being zero. We want to prevent this (why?).

  - Solution: smooth!

$$\widehat{P}(X_i = j | Y = y) = \frac{b_j + \alpha}{\sum_k b_k + \alpha K}$$

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.

Learning Outcomes

At the end of today's lecture, you will be able to:

1. Contrast discriminative and generative models.
2. Contrast maximum likelihood and maximum a poster learning.
3. Explain the modeling assumptions for a Naïve Bayes Classifier.

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, and Fred Sala