

# Machine Learning (CS 181): 7. Probabilistic Classification

David Parkes and Sasha Rush

# Contents

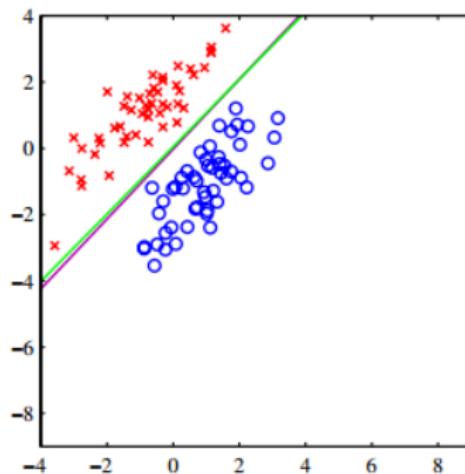
- 1 Generative Probabilistic View
- 2 Discrete Features
- 3 Multinomial Naive Bayes
- 4 Discriminative Probabilistic View
- 5 Logistic Regression
- 6 Multiclass Classification

# Contents

- 1 Generative Probabilistic View
- 2 Discrete Features
- 3 Multinomial Naive Bayes
- 4 Discriminative Probabilistic View
- 5 Logistic Regression
- 6 Multiclass Classification

# Last Class: Binary Classification

- Output space  $\mathcal{Y}$  is a fixed set of classes.
- Simplest case  $\mathcal{Y} = \{-1, 1\}$  (red/blue)
- Discriminant function:



Binary Classification

# Generative Classification View

Model the joint probability of class  $y$  and data  $\mathbf{x}$ ,

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

Note: Switch to different binary class representation,

$$\mathcal{Y} = \{0, 1\}$$

Process:

- Class is generated with probability  $p(y)$
- Input  $\mathbf{x}$  is generated conditional on class  $p(\mathbf{x}|y)$

# Class Probability

Choice of prior can depend on problem format,

- In binary case, we will use Bernoulli distribution,

$$p(y = 1; \theta) = \theta \quad \text{and} \quad p(y = 0; \theta) = 1 - \theta$$

or more compactly (motivates notation change),

$$p(y; \theta) = \theta^y (1 - \theta)^{1-y}$$

# Class-Conditional Probability

- Choice depends on modeling assumptions
- Select parameteric model for data given class

$$p(\mathbf{x}|y, \mathbf{w}_0, \mathbf{w}_1)$$

- Use continuous data, use multivariate Gaussian (HW)

$$p(\mathbf{x}|y=0; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$$

$$p(\mathbf{x}|y=1; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$$

- Today's class, discrete inputs,

$$p(\mathbf{x}|y; \pi_0, \pi_1)$$

# Class-Conditional Probability

- Choice depends on modeling assumptions
- Select parameteric model for data given class

$$p(\mathbf{x}|y, \mathbf{w}_0, \mathbf{w}_1)$$

- Use continuous data, use multivariate Gaussian (HW)

$$p(\mathbf{x}|y=0; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$$

$$p(\mathbf{x}|y=1; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$$

- Today's class, discrete inputs,

$$p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1)$$

# Maximum Likelihood

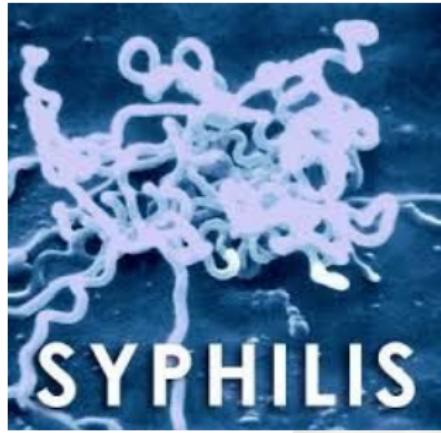
Same probabilistic approach as before,

1. Decide on generating process
2. Fix the parameterization of the model
3. Minimize negative log-likelihood of the data.

$$\min_{\pi_0, \pi_1, \theta} \mathcal{L}(\pi_0, \pi_1, \theta) = \min_{\pi_0, \pi_1, \theta} - \sum_{i=1}^n (\ln p(y_i; \theta) + \ln p(\mathbf{x}_i | y_i; \pi_0, \pi_1))$$

- Again, benefits will be having explicit probabilities for classes.
- Can also combine with Bayesian approaches from last class.

# Probabilistic Classification



# Probabilistic Classification

## Frequently Bought Together



- This item: Roots, Shoots, Buckets & Boots: Gardening Together with Children
- Toad Cottages and Shooting St
- Trowel and Error: Over 700 Tips

Google

Search

poliburo

About 3,100,000 results (0.49 seconds)

## Customers Who Bought This



Toad Cottages and  
Shooting Stars:  
Grandma's... by Sharon  
Lovejoy



Ga  
(Br)  
Ga  
Har

Everything

Images

Maps

Videos

News

Shopping

Mars

Show search tools

### Poliburo - Wikipedia, the free encyclopedia

en.wikipedia.org/w/index.php?title=Poliburo

**Poliburo** (Russian: Политбюро, literally "Political Bureau" [of the Central Committee]) is the executive committee for a number of communist political parties.

#### Poliburo of the Communist...

The Central Politburo of the Communist Party of China or ...

#### Poliburo of the Central...

The Politburo (Russian: Политбюро, literally "Political Bureau" [of the Central Committee]) is the executive committee for a number of communist political parties.

#### Poliburo of the Zimbabwe...

Poliburo of the Zimbabwe African National Union – Patriotic Front ...

#### Category:Poliburos

Category:Politburos. From Wikipedia, the free ...

More results from wikipedia.org »

### Poliburo - Definition and More from the Free Merriam-Webster...

www.merriam-webster.com/dictionary/poliburo

Definition of **poliburo** from the Merriam-Webster Online Dictionary with audio pronunciation, usage examples, etymology, Word of the Day, and word games.

### Poliburo (political body) - Britannica Online Encyclopedia

www.britannica.com/EBchecked/topic/430000/Poliburo

In Russian and Soviet history, the supreme political body of the Communist Party of the Soviet Union. The Poliburo until July 1956 exercised supreme ...

### Poliburo - Define Poliburo at Dictionary.com

dictionary.reference.com/browse/poliburo

Poliburo definition at Dictionary.com, a free online dictionary with pronunciation, synonyms and translation. Look it up now!

### Poliburo - Spartacus Educational

www.spartacus.schoolnet.co.uk/RUS/poliburo.htm

By John Simkin. Mose by John Simkin. It was therefore replaced by a more modern Poliburo (increased to nine in 1955 and ten in 1956). Its first members were Vladimir Lenin, Leon Trotsky, Joseph Stalin, ...

### Wiktionary

1. The governing council of the Communist Party of the Soviet Union and other Leninist political systems
2. A senior policymaking body in a political organization, especially one controlled by a party, which is dominated by the party in control of the organization or who attain membership through their personal political affiliations.

### Wikipedia

#### Poliburo

Poliburo (Russian: Политбюро, literally "Political Bureau" [of the Central Committee]) is the executive committee for a number of communist political parties.

[1] Contents 1 History 2 Marxist-Leninist states 3 Trotskyist parties 4 See also 5 References [edit]

### View talk

### Flickr



### YouTube

# Contents

1 Generative Probabilistic View

2 Discrete Features

3 Multinomial Naive Bayes

4 Discriminative Probabilistic View

5 Logistic Regression

6 Multiclass Classification

# Discrete Features / Basis

- Features so far, mostly continuous:
  - revenue, distance, crime rate, etc.
- For many domains, discrete features are more natural.
- Features represent (sparse) indicators or counts
  - visits, responses, properties, etc

$$\mathbf{x} = [0; 0; 1; \dots; 0; 10; 0]$$

- Want to model with discrete distributions as opposed to Gaussians.

# Discrete Features / Basis

- Features so far, mostly continuous:
  - revenue, distance, crime rate, etc.
- For many domains, discrete features are more natural.
- Features represent (sparse) indicators or counts
  - visits, responses, properties, etc

$$\mathbf{x} = [0; 0; 1; \dots; 0; 10; 0]$$

- Want to model with discrete distributions as opposed to Gaussians.

Earn a Degree based on your Life Experience

Obtain a Bachelors, Masters, MBA, or PhD based on your present knowledge and life experience.

No required tests, classes, or books. Confidentiality assured.

Join our fully recognized Degree Program.

Are you a truly qualified professional in your field but lack the appropriate, recognized documentation to achieve your goals?

Or are you venturing into a new field and need a boost to get your foot in the door so you can prove your capabilities?

Call us for information that can change your life and help you to achieve your goals!!!

**CALL NOW TO RECEIVE YOUR DIPLOMA WITHIN 30  
DAYS**

## Good Sentences

- A thoughtful, provocative, insistently humanizing film.
- Occasionally melodramatic, it's also extremely effective.
- Guaranteed to move anyone who ever shook, rattled, or rolled.

## Bad Sentences

- A sentimental mess that never rings true.
- This 100-minute movie only has about 25 minutes of decent material.
- Here, common sense flies out the window, along with the hail of bullets, none of which ever seem to hit Sascha.

## Features 1: Sparse Bag-of-Words Features

Example: Movie review input,

A sentimental mess

$$\phi(\mathbf{x}) = [\phi_{\text{word:A}}(\mathbf{x}); \dots; \phi_{\text{word:sentimental}}(\mathbf{x}); \phi_{\text{word:mess}}(\mathbf{x}); \dots]$$

$$\phi(\mathbf{x}) = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{array}{l} \text{word:A} \\ \vdots \\ \text{word:mess} \\ \text{word:sentimental} \end{array}$$

## Features 2: Sparse Word Properties

Example: Spam Email

Your diploma puts a UUNIVERSITY JOB PLACEMENT COUNSELOR  
at your disposal.

$$\phi(\mathbf{x}) = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{array}{l} \text{misspelling} \\ \vdots \\ \text{capital} \\ \text{word:diploma} \end{array}$$

# Contents

1 Generative Probabilistic View

2 Discrete Features

3 Multinomial Naive Bayes

4 Discriminative Probabilistic View

5 Logistic Regression

6 Multiclass Classification

# Multinomial Distribution

Probability of histogram  $\mathbf{x}$ , assuming  $\sum_{j=1}^m x_j$  trials, (exp trick)

$$p(\mathbf{x}; \boldsymbol{\pi}) = \frac{(\sum x_j)!}{\prod x_j!} \prod_{j=1}^m \pi_j^{x_j} \propto \prod_{j=1}^m \pi_j^{x_j}$$

Maximum likelihood estimate over  $\mathbf{x}_1 \dots \mathbf{x}_n$ , (log trick)

$$\arg \max_{\boldsymbol{\pi} \geq 0} \prod_{i=1}^n p(\mathbf{x}_i; \boldsymbol{\pi}) =$$

$$\arg \max_{\boldsymbol{\pi} \geq 0} \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j$$

$$\text{s.t. } \sum_{j=1}^m \pi_j = 1$$

# Multinomial Distribution

Probability of histogram  $\mathbf{x}$ , assuming  $\sum_{j=1}^m x_j$  trials, (exp trick)

$$p(\mathbf{x}; \boldsymbol{\pi}) = \frac{(\sum x_j)!}{\prod x_j!} \prod_{j=1}^m \pi_j^{x_j} \propto \prod_{j=1}^m \pi_j^{x_j}$$

Maximum likelihood estimate over  $\mathbf{x}_1 \dots \mathbf{x}_n$ , (log trick)

$$\arg \max_{\boldsymbol{\pi} \geq 0} \prod_{i=1}^n p(\mathbf{x}_i; \boldsymbol{\pi}) =$$

$$\arg \max_{\boldsymbol{\pi} \geq 0} \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j$$

$$\text{s.t. } \sum_{j=1}^m \pi_j = 1$$

# Maximum Likelihood (1)

Lagrange multipliers:

$$L(\boldsymbol{\pi}, \lambda) = \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j + \lambda \left( \sum_{j=1}^m \pi_j - 1 \right)$$

Solve  $\boldsymbol{\pi}$  for all  $j$ ,

$$\begin{aligned}\frac{\partial}{\partial \pi_j} L(\boldsymbol{\pi}, \lambda) &= \sum_{i=1}^n x_{ij}/\pi_j + \lambda = 0 \\ \pi_j &= \sum_{i=1}^n x_{ij}/\lambda\end{aligned}$$

# Maximum Likelihood (1)

Lagrange multipliers:

$$L(\boldsymbol{\pi}, \lambda) = \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j + \lambda \left( \sum_{j=1}^m \pi_j - 1 \right)$$

Solve  $\boldsymbol{\pi}$  for all  $j$ ,

$$\begin{aligned}\frac{\partial}{\partial \pi_j} L(\boldsymbol{\pi}, \lambda) &= \sum_{i=1}^n x_{ij}/\pi_j + \lambda = 0 \\ \pi_j &= \sum_{i=1}^n x_{ij}/\lambda\end{aligned}$$

## Maximum Likelihood (2)

Lagrange multipliers:

$$L(\boldsymbol{\pi}, \lambda) = \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j + \lambda \left( \sum_{j=1}^m \pi_j - 1 \right)$$

$$\pi_j = \sum_{i=1}^n x_{ij} / \lambda$$

Solve  $\lambda$ ,

$$\frac{\partial}{\partial \lambda} L(\boldsymbol{\pi}, \lambda) = \sum_{j=1}^m \pi_j = 1$$

$$\sum_{j=1}^m \sum_{i=1}^n x_{ij} / \lambda = 1$$

$$\lambda = \sum_{i=1}^n \sum_{j=1}^m x_{ij}$$

## Maximum Likelihood (2)

Lagrange multipliers:

$$L(\boldsymbol{\pi}, \lambda) = \sum_{i=1}^n \sum_{j=1}^m x_{ij} \ln \pi_j + \lambda \left( \sum_{j=1}^m \pi_j - 1 \right)$$

$$\pi_j = \sum_{i=1}^n x_{ij} / \lambda$$

Solve  $\lambda$ ,

$$\frac{\partial}{\partial \lambda} L(\boldsymbol{\pi}, \lambda) = \sum_{j=1}^m \pi_j = 1$$

$$\sum_{j=1}^m \sum_{i=1}^n x_{ij} / \lambda = 1$$

$$\lambda = \sum_{i=1}^n \sum_{j=1}^m x_{ij}$$

## Multinomial MLE Matrix Form

The maximum likelihood parameters become the counts over the data:

$$\hat{\boldsymbol{\pi}} = \frac{\sum_{i=1}^n \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m x_{ij}} = \frac{\mathbf{X}^\top \mathbf{1}}{\mathbf{1}^\top \mathbf{X} \mathbf{1}}$$

# Multinomial Binary-Class Naive Bayes

Assume each discrete features  $\mathbf{x}$ , model joint probability as

$$p(\mathbf{x}, y) = p(y; \theta)p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1)$$

- Class probability, Bernoulli  $p(y; \theta)$
- Features conditional Multinomial:

$$p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1) \propto \prod_{j=1}^m \pi_{yj}^{x_j}$$

- Multinomial: each feature is generated independently (Naive)
- Conditional classification using Bayes rule (Bayes)

$$p(y|\mathbf{x}) \propto p(y; \theta)p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1)$$

## Multinomial Binary-Class Naive Bayes

Assume each discrete features  $\mathbf{x}$ , model joint probability as

$$p(\mathbf{x}, y) = p(y; \theta)p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1)$$

- Class probability, Bernoulli  $p(y; \theta)$
- Features conditional Multinomial:

$$p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1) \propto \prod_{j=1}^m \pi_{yj}^{x_j}$$

- Multinomial: each feature is generated independently ([Naive](#))
- Conditional classification using Bayes rule ([Bayes](#))

$$p(y|\mathbf{x}) \propto p(y; \theta)p(\mathbf{x}|y; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1)$$

# Loss for Multinomial Naive Bayes

$$\max_{\theta, \boldsymbol{\pi}_0, \boldsymbol{\pi}_1} \sum_{i=1}^n \ln p(\mathbf{x}_i, y_i) = \max_{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2} \sum_{i=1}^n \ln p(\mathbf{x}_i | y_i; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1) + \max_{\theta} \sum_{i=1}^n \ln p(y_i; \theta)$$

## ■ Class probability

$$\hat{\theta} = \frac{1}{n} \mathbf{y}$$

## ■ Class-Conditional probability

$$\hat{\boldsymbol{\pi}}_0 = \frac{\sum_{i=1}^n (1 - y_i) \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m (1 - y_i) x_{ij}}$$

$$\hat{\boldsymbol{\pi}}_1 = \frac{\sum_{i=1}^n y_i \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m y_i x_{ij}}$$

# Loss for Multinomial Naive Bayes

$$\max_{\theta, \boldsymbol{\pi}_0, \boldsymbol{\pi}_1} \sum_{i=1}^n \ln p(\mathbf{x}_i, y_i) = \max_{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2} \sum_{i=1}^n \ln p(\mathbf{x}_i | y_i; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1) + \max_{\theta} \sum_{i=1}^n \ln p(y_i; \theta)$$

## ■ Class probability

$$\hat{\theta} = \frac{\mathbf{1}\mathbf{y}}{n}$$

## ■ Class-Conditional probability

$$\hat{\boldsymbol{\pi}}_0 = \frac{\sum_{i=1}^n (1 - y_i) \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m (1 - y_i) x_{ij}}$$

$$\hat{\boldsymbol{\pi}}_1 = \frac{\sum_{i=1}^n y_i \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m y_i x_{ij}}$$

# Loss for Multinomial Naive Bayes

$$\max_{\theta, \boldsymbol{\pi}_0, \boldsymbol{\pi}_1} \sum_{i=1}^n \ln p(\mathbf{x}_i, y_i) = \max_{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2} \sum_{i=1}^n \ln p(\mathbf{x}_i | y_i; \boldsymbol{\pi}_0, \boldsymbol{\pi}_1) + \max_{\theta} \sum_{i=1}^n \ln p(y_i; \theta)$$

- Class probability

$$\hat{\theta} = \frac{\mathbf{1y}}{n}$$

- Class-Conditional probability

$$\hat{\boldsymbol{\pi}}_0 = \frac{\sum_{i=1}^n (1 - y_i) \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m (1 - y_i) x_{ij}}$$

$$\hat{\boldsymbol{\pi}}_1 = \frac{\sum_{i=1}^n y_i \mathbf{x}_i}{\sum_{i=1}^n \sum_{j=1}^m y_i x_{ij}}$$

# Multinomial Naive Bayes (In Practice)

$$\ln p(y|\mathbf{x}) \propto \ln p(\mathbf{x}|y) + \ln p(y)$$

How do you decide what class? (Discriminant function)

$$\begin{aligned} h(\mathbf{x}) &= (\ln p(\mathbf{x}|y=1) + \ln p(y=1)) - (\ln p(\mathbf{x}|y=0) + \ln p(y=0)) \\ &= [\ln \prod_{j=1}^m \pi_{1j}^{x_j} - \ln \prod_{j=1}^m \pi_{0j}^{x_j}] + [\ln \theta - \ln(1-\theta)] \\ &= \sum_{j=1}^m x_j \ln \frac{\pi_{1j}}{\pi_{0j}} + \ln \frac{\theta}{1-\theta} = \mathbf{x}^\top (\ln \frac{\boldsymbol{\pi}_1}{\boldsymbol{\pi}_0}) + \ln \frac{\theta}{1-\theta} \end{aligned}$$

But this formulation is a linear model.

$$h(\mathbf{x}; \mathbf{w}) = \mathbf{x}^\top \mathbf{w} + w_0$$

# Multinomial Naive Bayes (In Practice)

$$\ln p(y|\mathbf{x}) \propto \ln p(\mathbf{x}|y) + \ln p(y)$$

How do you decide what class? (Discriminant function)

$$\begin{aligned} h(\mathbf{x}) &= (\ln p(\mathbf{x}|y=1) + \ln p(y=1)) - (\ln p(\mathbf{x}|y=0) + \ln p(y=0)) \\ &= [\ln \prod_{j=1}^m \pi_{1j}^{x_j} - \ln \prod_{j=1}^m \pi_{0j}^{x_j}] + [\ln \theta - \ln(1-\theta)] \\ &= \sum_{j=1}^m x_j \ln \frac{\pi_{1j}}{\pi_{0j}} + \ln \frac{\theta}{1-\theta} = \mathbf{x}^\top (\ln \frac{\boldsymbol{\pi}_1}{\boldsymbol{\pi}_0}) + \ln \frac{\theta}{1-\theta} \end{aligned}$$

But this formulation is a linear model.

$$h(\mathbf{x}; \mathbf{w}) = \mathbf{x}^\top \mathbf{w} + w_0$$

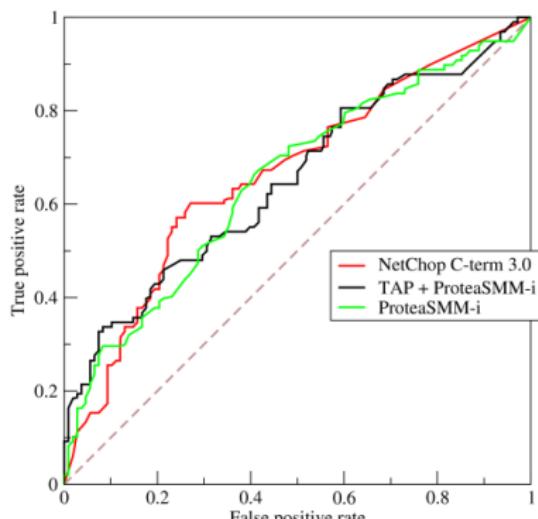
# Role of Class Probabilities

$$\mathbf{x}^\top \left( \ln \frac{\pi_1}{\pi_0} \right) + \ln \frac{\theta}{1 - \theta}$$

Class probabilities  $p(y; \theta)$  determine the threshold.

$$h(\mathbf{x}; \mathbf{w}) = \mathbf{x}^\top \mathbf{w} + w_0$$

Brings us back again to ROC curves.



# Naive Bayes In Practice

- Super fast to train.
- Relatively interpretable.
- Performs quite well on small datasets.

Method	RT-s	MPQA	CR	Subj.
MNB-uni	77.9	85.3	79.8	<b>92.6</b>
MNB-bi	<b>79.0</b>	<b>86.3</b>	80.0	<u>93.6</u>
SVM-uni	76.2	86.1	79.0	90.8
SVM-bi	77.7	<u>86.7</u>	80.8	91.7
NBSVM-uni	<b>78.1</b>	85.3	80.5	92.4
NBSVM-bi	<b>79.4</b>	<b>86.3</b>	<b>81.8</b>	<b>93.2</b>
RAE	76.8	85.7	—	—
RAE-pretrain	77.7	<b>86.4</b>	—	—
Voting-w/Rev.	63.1	81.7	74.2	—

(RT-S [movie review], CR [customer reports], MPQA [opinion polarity], SUBJ [subjectivity])

# Naive Bayes Example

# Contents

- 1 Generative Probabilistic View
- 2 Discrete Features
- 3 Multinomial Naive Bayes
- 4 Discriminative Probabilistic View
- 5 Logistic Regression
- 6 Multiclass Classification

# Generative versus Discriminative Model

- Generative models: Parameterize Joint Distribution

$$\arg \max_{\mathbf{w}} \prod_i p(\mathbf{x}_i, y_i; \mathbf{w})$$

- Discriminative model: Parameterize Conditional Distribution

$$\arg \max_{\mathbf{w}} \prod_i p(y_i | \mathbf{x}_i; \mathbf{w})$$

- Why does this matter?

# Linear Discriminative Model

Set log prob to be proportional to some linear model, shorthand  $h$

$$\ln p(y=1|\mathbf{x}; \mathbf{w}) \propto \mathbf{w}^\top \mathbf{x} + w_0 = h$$

As before threshold at  $h > 0$ ,

$$\ln p(y=0|\mathbf{x}; \mathbf{w}) \propto 0$$

Now remove log and normalize,

$$p(y=1|\mathbf{x}; \mathbf{w}) = \frac{\exp h}{\exp h + \exp 0} = (1 + \exp -h)^{-1}$$

$$p(y=0|\mathbf{x}; \mathbf{w}) = \frac{\exp 0}{\exp h + \exp 0} = (1 + \exp h)^{-1}$$

Call this function the logistic sigmoid activation.

$$\sigma(h) = (1 + \exp -h)^{-1}$$

# Linear Discriminative Model

Set log prob to be proportional to some linear model, shorthand  $h$

$$\ln p(y=1|\mathbf{x}; \mathbf{w}) \propto \mathbf{w}^\top \mathbf{x} + w_0 = h$$

As before threshold at  $h > 0$ ,

$$\ln p(y=0|\mathbf{x}; \mathbf{w}) \propto 0$$

Now remove log and normalize,

$$\begin{aligned} p(y=1|\mathbf{x}; \mathbf{w}) &= \frac{\exp h}{\exp h + \exp 0} = (1 + \exp -h)^{-1} \\ p(y=0|\mathbf{x}; \mathbf{w}) &= \frac{\exp 0}{\exp h + \exp 0} = (1 + \exp h)^{-1} \end{aligned}$$

Call this function the logistic sigmoid activation.

$$\sigma(h) = (1 + \exp -h)^{-1}$$

# Linear Discriminative Model

Set log prob to be proportional to some linear model, shorthand  $h$

$$\ln p(y=1|\mathbf{x}; \mathbf{w}) \propto \mathbf{w}^\top \mathbf{x} + w_0 = h$$

As before threshold at  $h > 0$ ,

$$\ln p(y=0|\mathbf{x}; \mathbf{w}) \propto 0$$

Now remove log and normalize,

$$p(y=1|\mathbf{x}; \mathbf{w}) = \frac{\exp h}{\exp h + \exp 0} = (1 + \exp -h)^{-1}$$

$$p(y=0|\mathbf{x}; \mathbf{w}) = \frac{\exp 0}{\exp h + \exp 0} = (1 + \exp h)^{-1}$$

Call this function the **logistic sigmoid** activation.

$$\sigma(h) = (1 + \exp -h)^{-1}$$

# Contents

1 Generative Probabilistic View

2 Discrete Features

3 Multinomial Naive Bayes

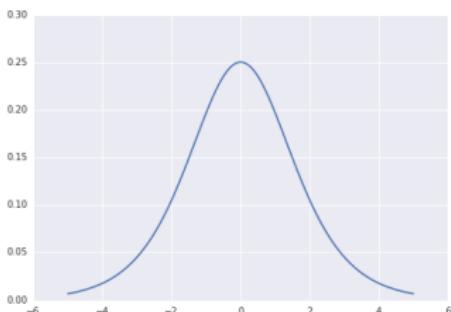
4 Discriminative Probabilistic View

5 Logistic Regression

6 Multiclass Classification

# (Logistic) Sigmoid Activation

$$\sigma(h) = (1 + \exp(-h))^{-1}$$



## Sigmoid Function and Derivative

- “Squashes”  $\mathbb{R}$  to a probabilities.

# Logistic Regression

Linear model converted to probability estimated by sigmoid

$$p(y = 1 | \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^\top \mathbf{x} + w_0) = (1 + \exp(-h))^{-1}$$

$$p(y = 0 | \mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^\top \mathbf{x} + w_0) = (1 + \exp h)^{-1}$$

- Linear “Regression” transformed to probability estimate.
- Name is confusing, mostly used for *classification*.

# Fitting Model

Reminder:

$$p(y = 1 | \mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^\top \mathbf{x} + w_0) = (1 + \exp(-h))^{-1}$$

$$p(y = 0 | \mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^\top \mathbf{x} + w_0) = (1 + \exp h)^{-1}$$

As this is now a probabilistic model, can fit with MLE.

$$\begin{aligned}\mathcal{L}(\mathbf{w}) &= -\sum_{i=1}^n \ln p(y_i | \mathbf{x}_i; \mathbf{w}) = -\sum_{i=1}^n \ln \sigma(h)^{y_i} (1 - \sigma(h))^{1-y_i} \\ &= \sum_{i=1}^n y_i \ln(1 + \exp(-h)) + (1 - y_i) \ln(1 + \exp h)\end{aligned}$$

# Likelihood and Estimation

Reminder:

$$\begin{aligned} h &= \mathbf{w}^\top \mathbf{x}_i + w_0 \\ \mathcal{L}(\mathbf{w}) &= \sum_{i=1}^n y_i \ln(1 + \exp(-h)) + (1 - y_i) \ln(1 + \exp h) \end{aligned}$$

Take gradients:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{w}} \ln(1 + \exp(-h)) &= -\mathbf{x}_i \frac{\exp -h}{1 + \exp(-h)} = -\mathbf{x}_i p(y_i = 0 | \mathbf{x}) \\ \frac{\partial}{\partial \mathbf{w}} \ln(1 + \exp h) &= \mathbf{x}_i \frac{\exp h}{1 + \exp h} = \mathbf{x}_i p(y_i = 1 | \mathbf{x}) \end{aligned}$$

$$\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}) = \sum_{i=1}^n -y_i \mathbf{x}_i p(y_i = 0 | \mathbf{x}_i) + (1 - y_i) \mathbf{x}_i p(y_i = 1 | \mathbf{x}_i)$$

# Likelihood and Estimation

Reminder:

$$\begin{aligned} h &= \mathbf{w}^\top \mathbf{x}_i + w_0 \\ \mathcal{L}(\mathbf{w}) &= \sum_{i=1}^n y_i \ln(1 + \exp(-h)) + (1 - y_i) \ln(1 + \exp h) \end{aligned}$$

Take gradients:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{w}} \ln(1 + \exp(-h)) &= -\mathbf{x}_i \frac{\exp -h}{1 + \exp(-h)} = -\mathbf{x}_i p(y_i = 0 | \mathbf{x}) \\ \frac{\partial}{\partial \mathbf{w}} \ln(1 + \exp h) &= \mathbf{x}_i \frac{\exp h}{1 + \exp h} = \mathbf{x}_i p(y_i = 1 | \mathbf{x}) \end{aligned}$$

$$\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}) = \sum_{i=1}^n -y_i \mathbf{x}_i p(y_i = 0 | \mathbf{x}_i) + (1 - y_i) \mathbf{x}_i p(y_i = 1 | \mathbf{x}_i)$$

## Recall: Perceptron Algorithm

1. Iterate over the data:
  - If correct ( $y_i = \hat{y}_i$ ), do nothing.
  - If incorrect, add/subtract  $\eta \times \mathbf{x}_i$  to weights
2. If errors, repeat process.
3. Otherwise separator is found.

# SGD on Logistic Regression

1. Iterate over the data:

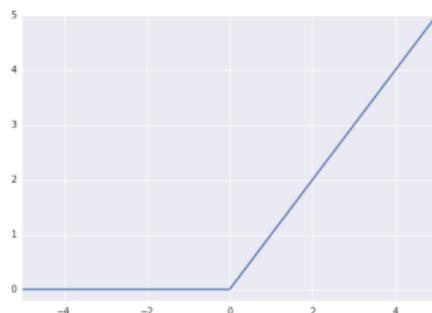
- Compute  $p(y_i = 1 | \mathbf{x}_i)$ .
- If ( $y_i = 1$ ), add  $\eta \times \mathbf{x}_i p(y_i = 0 | \mathbf{x}_i)$  to  $\mathbf{w}$
- If ( $y_i = 0$ ), add  $\eta \times -\mathbf{x}_i p(y_i = 1 | \mathbf{x}_i)$  to  $\mathbf{w}$

2. Repeat until convergence.

**Guaranteed** to maximize conditional likelihood of data.

# Algorithms Comes from Activations

- What is the difference between Perceptron and Logistic Regression?



Sigmoid Function versus Hinge

# Contents

1 Generative Probabilistic View

2 Discrete Features

3 Multinomial Naive Bayes

4 Discriminative Probabilistic View

5 Logistic Regression

6 Multiclass Classification

# Multiclass

- ★★★★  
I visited The Abbey on several occasions on a visit to Cambridge and found it to be a solid, reliable and friendly place for a meal.
- ★★  
However, the food leaves something to be desired. A very obvious menu and average execution
- ★★★★★  
Fun, friendly neighborhood bar. Good drinks, good food, not too pricey. Great atmosphere!

# Multiclass Naive Bayes

Multiclass outputs:  $\mathcal{Y} = \{C_1, \dots, C_c\}$

One-hot vectors

$$C_2 = [0; 1; 0; \dots; 0]$$

- Class distribution uses categorical for each  $k \in \{1 \dots c\}$ ,

$$p(\mathbf{y} = C_k; \boldsymbol{\pi}) = \pi_k$$

- Class-Conditional uses separate parameters for each class

$$\{\mathbf{w}_\ell\}_{\ell=1}^c$$

$$p(\mathbf{x}|\mathbf{y}; \{\mathbf{w}_\ell\}_{\ell=1}^c)$$

In homework, these each parameterize a multivariate Gaussian.

# Multiclass Logistic Regression

- Multiclass logistic regression uses a  $\mathbf{w}_\ell$  for each class.
- Generalization of sigmoid is softmax function.

$$p(\mathbf{y} = C_k | \mathbf{x}; \{\mathbf{w}_\ell\}_{\ell=1}^c) = \frac{\exp(\mathbf{x}^\top \mathbf{w}_k)}{\sum_\ell \exp(\mathbf{x}^\top \mathbf{w}_\ell)}$$

(Derivation and exploration on homework.)

# Neural Network Preview: Softmax In Action

$$p_{\theta/\rho}(a|s)$$

