# CS247: ADVANCED DATA MINING

## 02: Basics: Probabilistic Classifiers

---

**Instructor: Yizhou Sun**

yzsun@cs.ucla.edu

January 9, 2024

# Content

- Probabilistic Models
- Naïve Bayes
- Logistic Regression
- Generative Models and Discriminative Models
- Summary

# Recap: Classification

- Given a training dataset $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ with categorical labels
  - Training stage: Construct a model
  - Test stage: apply the model to an unseen data point

# Example: Tabular data classification

- Heart disease diagnosis

| Age | Sex | ChestPain | RestBP | Chol | Fbs | RestECG | MaxHR | ExAng | Oldpeak | Slope | Ca | Thal | AHD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 63 | 1 | typical | 145 | 233 | 1 | 2 | 150 | 0 | 2.3 | 3 | 0.0 | fixed | No |
| 67 | 1 | asymptomatic | 160 | 286 | 0 | 2 | 108 | 1 | 1.5 | 2 | 3.0 | normal | Yes |
| 67 | 1 | asymptomatic | 120 | 229 | 0 | 2 | 129 | 1 | 2.6 | 2 | 2.0 | reversable | Yes |
| 37 | 1 | nonanginal | 130 | 250 | 0 | 0 | 187 | 0 | 3.5 | 3 | 0.0 | normal | No |
| 41 | 0 | nontypical | 130 | 204 | 0 | 2 | 172 | 0 | 1.4 | 1 | 0.0 | normal | No |

# Example: Text Classification

- Spam detection

  From: airak@medicana.com.tr
  Subject: Loan Offer
  Do you need a personal or business loan urgent that can be process within 2 to 3 working days? Have you been frustrated so many times by your banks and other loan firm and you don't know what to do? Here comes the Good news Deutsche Bank Financial Business and Home Loan is here to offer you any kind of loan you need at an affordable interest rate of 3% If you are interested let us know.

- Sentiment analysis


The Lion King, complete with jaunty songs by Elton John and Tim Rice, is undeniably and fully worthy of its glorious Disney heritage. It is a gorgeous triumph -- one lion in which the studio can take justified pride.

Between traumas, the movie serves up soothingly banal musical numbers (composed by Elton John and Tim Rice) and silly, rambunctious comedy.

July 31, 2013 | Full Review…

slido

# Join at slido.com
# #146811

ⓘ Start presenting to display the joining instructions on this slide.

slido

# Write down the classification algorithms that you've learned

ⓘ Start presenting to display the poll results on this slide.

# Recap: Probability and Bayes' Theorem

- Marginal probability: $p(X = x) \coloneqq p(x)$
- Joint probability: $p(X = x, Y = y) \coloneqq p(x, y)$
- Conditional probability: $p(X = x|Y = y) \coloneqq p(x|y)$

- Bayes' theorem
  - $p(y|x) = \frac{p(x|y)p(y)}{p(x)}$

# Probabilistic Models for Classification

- Data: $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$
  - A data point $(\boldsymbol{x}_i, y_i)$ contains a feature vector and a discrete label
  - n: number of data points
- Model: $p(D|\theta)$
  - E.g., under I.I.D. assumption
    - $p(D|\theta) = \prod_i p(\boldsymbol{x}_i, y_i|\theta)$ (if modeling joint distribution)
    - $p(D|\theta) = \prod_i p(y_i|\boldsymbol{x}_i, \theta)$ (if modeling conditional distribution, conditional i.i.d.)
- Inference: query the model
  - $e.g.,\ classification,\ p(y_i|\boldsymbol{x}_i, \theta) = ?$
- Learning: $\hat{\theta} = ?$

# Content

- Probabilistic Models
- Naïve Bayes 
- Logistic Regression
- Generative Models and Discriminative Models
- Summary

# Take text classification as an example

- Spam detection

From: airak@medicana.com.tr
Subject: Loan Offer
Do you need a personal or business loan urgent that can be process within 2 to 3 working days? Have you been frustrated so many times by your banks and other loan firm and you don't know what to do? Here comes the Good news Deutsche Bank Financial Business and Home Loan is here to offer you any kind of loan you need at an affordable interest rate of 3% If you are interested let us know.

# Represent A Document

**Ex: "Do you need a personal or business loan urgent?"**

- A document d is represented by a sequence of words selected from a vocabulary with $N$ words/tokens
  - $\boldsymbol{w}_d = (w_{d1}, w_{d2}, \ldots, w_{dN_d})$, where $w_{di}$ is the id of i-th word in document $d$ and $N_d$ is the length of document $d$
- A bag-of-words representation
  - $\boldsymbol{x}_d = (x_{d1}, x_{d2}, \ldots, x_{dN})$, where $x_{dn}$ is the number of words for n-th word in the vocabulary and $N$ is the total number of words in the vocabulary
  - $x_{dn} = \sum_i 1(w_{di} == n)$

# Example of Bag-of-Words Representation

c1: *Human* machine *interface* for Lab ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user*-perceived *response time* to error measurement

m1: The generation of random, binary, unordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

| | c1 | c2 | c3 | c4 | c5 | m1 | m2 | m3 | m4 |
|---|---|---|---|---|---|---|---|---|---|
| *human* | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| *interface* | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| *computer* | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| *user* | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| *system* | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 |
| *response* | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| *time* | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| *EPS* | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| *survey* | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| *trees* | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| *graph* | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| *minors* | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

$\boldsymbol{x}_d$

# Problem formalization

- Data: $D = \{(\boldsymbol{x}_d, y_d)\}_{d=1}^{|D|}$
  - A data point $(\boldsymbol{x}_d, y_d)$ contains a document vector and its label
  - $|\mathrm{D}|$: number of documents
- Model: $p(D|\theta)$
  - $p(D|\theta) = \prod_d p(\boldsymbol{x}_d, y_d|\theta)$ (modeling joint distribution)
- Classification Inference: query the model
  - $p(y_d|\boldsymbol{x}_d, \theta) = ?$
- Learning: $\hat{\theta} = ?$

# The Key Problem

- How to model $p(\boldsymbol{x}_d, y_d|\theta)$?

- $p(\boldsymbol{x}_d, y_d|\theta) = p(\boldsymbol{x}_d|y_d, \theta)p(y_d|\theta)$
  - $p(\boldsymbol{x}_d|y_d, \theta)$: how to generate a document given its class label
    - E.g., $p($"**Do you need a personal or business loan urgent**"$|spam, \theta)$
  - $p(y_d|\theta)$: what is the distribution for class labels?
    - E.g., $p(spam|\theta)$

# Recap: Bernoulli and Categorical Distribution

- Bernoulli distribution
  - Discrete distribution that takes two values $\{0,1\}$
    - $P(X = 1) = p$ and $P(X = 0) = 1 - p$
    - E.g., toss a coin with head and tail

- Categorical distribution
  - Discrete distribution that takes more than two values, i.e., $x \in \{1, \dots, K\}$
    - Also called generalized Bernoulli distribution, multinoulli distribution
    - $P(X = k) = p_k \; and \; \sum_k p_k = 1$
    - E.g., get 1-6 from a dice with 1/6

slido

**Please write down words that you are going to use for a healthcare article. (write down one word each time)**

ⓘ Start presenting to display the poll results on this slide.

slido

**Please write down words that you are going to use for a sports article. (write down one word each time)**

ⓘ Start presenting to display the poll results on this slide.

# Conditional Independence Assumption

- $p(\textbf{Do you need a personal or business loan urgent}|spam, \theta) = p(\textbf{Do}|spam, \theta) \times p(\textbf{you}|spam, \theta) \times \cdots \times p(\textbf{urgent}|spam, \theta)$
- Each token is sampled from a categorical distribution
  - $p(\textbf{urgent}|spam, \theta) = \theta_{spam,\textbf{urgent}}$

# Now Come to Modeling

- Model $p(\boldsymbol{x}_d, y_d) = p(\boldsymbol{x}_d|y_d)p(y_d)$
  - Model $p(\boldsymbol{w}_d|y_d = j)$ or $p(\boldsymbol{x}_d|y_d = j)$for class $j$
    - Each word in the sequence $w_{di}$ is sampled from categorical distribution with parameter vector $\boldsymbol{\beta}_j = (\beta_{j1}, \beta_{j2}, \dots, \beta_{jN})$ independently
      - $p(w_{di}|y_d = j) = \beta_{jw_{di}}$ and $p(\boldsymbol{w}_d|y_d = j) = \prod_i \beta_{jw_{di}} = \prod_n \beta_{jn}^{x_{dn}}$
      - Where $x_{dn}$ is the number of words for nth word in the vocabulary
  - Model $p(y_d = j)$
    - Follow categorical distribution with parameter vector $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_m)$, i.e.,
      - $p(y_d = j) = \pi_j$

# Classification Process Assuming Parameters are Given: Inference

- Find $y_d = j$ that maximizes $p(y_d|\boldsymbol{x}_d)$, which is equivalently to maximize

$$y_d^* = \underset{j}{argmax}\ p(\boldsymbol{x}_d, y_d = j)$$

$$= argmax_j\ p(\boldsymbol{x}_d|y_d = j)p(y_d = j)$$

$$= argmax_j \prod_n \beta_{jn}^{x_{dn}} \times \pi_j$$

$$= argmax_j \sum_n x_{dn} log\beta_{jn} + log\pi_j$$

# Announcement

- Three sessions have been combined in BruinLearn
  - Everyone could access Echo360 for recordings
- Slides will be updated in course website
- Please join Piazza if not yet
- About PTE
  - I will enroll students in waitlist
  - Can at most give out 5 more PTEs if no one is dropping
  - Will keep monitoring the number of enrollment

# Parameter Estimation via MLE: Learning

- Given a corpus and labels for each document
  - $D = \{(\boldsymbol{x}_d, y_d)\}$
  - Find the MLE estimators for $\Theta = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m, \boldsymbol{\pi})$
- The log likelihood function for the training dataset

$$logL(\Theta) = log \prod_d p(\boldsymbol{x}_d, y_d|\Theta) = \sum_d log\, p(\boldsymbol{x}_d, y_d|\Theta)$$

$$= \sum_d log\, [p(\boldsymbol{x}_d|y_d)p(y_d)] = \sum_d (\sum_n x_{dn} log \beta_{y_d n} + log \pi_{y_d})$$

- The optimization problem

$$\max_{\Theta} \log L\,(\Theta)$$

$$s.t.$$

$$\pi_j \geq 0\ and \sum_j \pi_j = 1$$

$$\beta_{jn} \geq 0\ and \sum_n \beta_{jn} = 1\ for\ all\ j$$

# Question from Baotao from last lecture

- How the length of the document affect the parameter learning?
  - Hint: repeating a document 10 times => $\boldsymbol{x}_d$ becomes ?

# Recap: Lagrangian

- Objective with equality constraints

$$\min_{w} f(w)$$

$$s.t.$$

$$h_i(w) = 0, for\ i = 1,2, \dots, l$$

- Lagrangian:
  - $L(w, \boldsymbol{\alpha}) = f(w) + \sum_i \alpha_i h_i(w)$
    - $\alpha_i$: Lagrangian multipliers
- Solution: setting the derivatives of Lagrangian to be 0
  - $\frac{\partial L}{\partial w} = 0\ and \frac{\partial L}{\partial \alpha_i} = 0$ for every i

# Solve the Optimization Problem

- Use the Lagrange multiplier method

$$\max_{\Theta} \sum_{d} \left( \sum_{n} x_{dn} log \beta_{y_d n} + log \pi_{y_d} \right)$$

$$s.t.$$

$$\pi_j \geq 0 \; and \sum_{j} \pi_j = 1$$

$$\beta_{jn} \geq 0 \; and \sum_{n} \beta_{jn} = 1 \; for \; all \; j$$

# Solve the Optimization Problem

- Solution
  - $\hat{\beta}_{jn} = \frac{\sum_{d:y_d==j} x_{dn}}{\sum_{d:y_d==j} \sum_{n'} x_{dn'}}$
    - $\sum_{d:y_d=j} x_{dn}$: total count of word n in class j
    - $\sum_{d:y_d=j} \sum_{n'} x_{dn'}$: total count of words in class j
  - $\hat{\pi}_j = \frac{\sum_d 1(y_d==j)}{|D|}$
    - $1(y_d = j)$ is the indicator function, which equals to 1 if $y_d = j$ holds
    - |D|: total number of documents

# Smoothing

- What if some word n does not appear in some class j in training dataset?
  - $\hat{\beta}_{jn} = \frac{\sum_{d:y_d=j} x_{dn}}{\sum_{d:y_d=j} \sum_{n'} x_{dn'}} = 0$
  - $\Rightarrow p(\boldsymbol{x}_d | y_d = j) = \prod_n \beta_{jn}^{x_{dn}} = 0$
  - But other words may have a strong indication the document belongs to class j
- Solution: add-1 smoothing or Laplace smoothing
  - $\hat{\beta}_{jn} = \frac{\sum_{d:y_d=j} x_{dn}+1}{\sum_{d:y_d=j} \sum_{n'} x_{dn'}+N}$
  - $N: total\ number\ of\ words\ in\ the\ vocabulary$
  - Check: $\sum_n \hat{\beta}_{jn} = 1$?

# Example

- Data:

| | Doc | Words | Class |
|---|---|---|---|
| Training | 1 | Chinese Beijing Chinese | c |
| | 2 | Chinese Chinese Shanghai | c |
| | 3 | Chinese Macao | c |
| | 4 | Tokyo Japan Chinese | j |
| Test | 5 | Chinese Chinese Chinese Tokyo Japan | ? |

- Vocabulary:

| Index | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Word | Chinese | Beijing | Shanghai | Macao | Tokyo | Japan |

- Learned parameters (with smoothing):

$$\hat{\beta}_{c1} = \frac{5+1}{8+6} = \frac{3}{7} \qquad \hat{\beta}_{j1} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\beta}_{c2} = \frac{1+1}{8+6} = \frac{1}{7} \qquad \hat{\beta}_{j2} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{c3} = \frac{1+1}{8+6} = \frac{1}{7} \qquad \hat{\beta}_{j3} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{c4} = \frac{1+1}{8+6} = \frac{1}{7} \qquad \hat{\beta}_{j4} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{c5} = \frac{0+1}{8+6} = \frac{1}{14} \qquad \hat{\beta}_{j5} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\beta}_{c6} = \frac{0+1}{8+6} = \frac{1}{14} \qquad \hat{\beta}_{j6} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\pi}_c = \frac{3}{4}$$

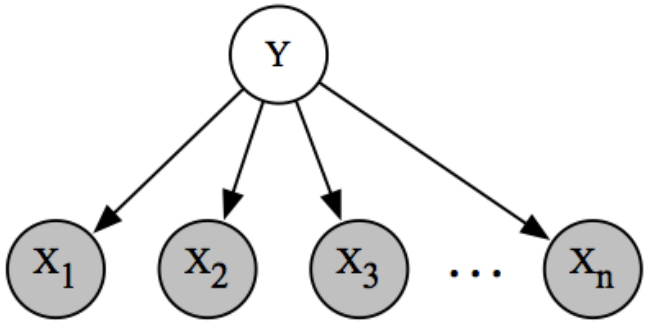
$$\hat{\pi}_j = \frac{1}{4}$$

# Example (Continued)

- Classification stage
  - For the test document d=5, compute
  - $p(y_5 = c|\boldsymbol{x}_5) \propto p(y_5 = c) \times \prod_n \beta_{cn}^{x_{5n}} = \frac{3}{4} \times \left(\frac{3}{7}\right)^3 \times \left(\frac{1}{14}\right) \times \left(\frac{1}{14}\right) \approx 0.0003$
  - $p(y_5 = j|\boldsymbol{x}_5) \propto p(y_5 = j) \times \prod_n \beta_{jn}^{x_{5n}} = \frac{1}{4} \times \left(\frac{2}{9}\right)^3 \times \left(\frac{2}{9}\right) \times \left(\frac{2}{9}\right) \approx 0.0001$
  - Conclusion: which class $\boldsymbol{x}_5$ should be classified into?
    - c class

# A More General Naïve Bayes Framework

- Let D be a training set of tuples and their class labels, and each tuple is represented by an p-D attribute vector **x** = ($x_1$, $x_2$, …, $x_p$)
- Suppose there are *m* classes y∈{1, 2, …, m}
- Goal: Find $\mathrm{y} = \arg\max_y p(y|\boldsymbol{x}) = p(y,\boldsymbol{x})/p(\boldsymbol{x}) \propto p(\boldsymbol{x}|y)p(y)$
- A simplified assumption: attributes are conditionally independent given the class (class conditional independency):
  - $p(\boldsymbol{x}|y) = \prod_k p(x_k|y)$
  - $p(x_k|y)$ can follow any distribution,
    - e.g., Gaussian, Bernoulli, categorical, …



# Content

- Probabilistic Models for I.I.D. Data
- Naïve Bayes
- Logistic Regression 
- Generative Models and Discriminative Models
- Summary

# Q&A

# Problem Formalization

- Data: $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$
  - A data point $(\boldsymbol{x}_i, y_i)$ contains a feature vector and a discrete label
  - n: number of data points
- Model: $p(D|\theta)$
  - $p(D|\theta) = \prod_i p(y_i|\boldsymbol{x}_i, \theta)$ (modeling conditional distribution, conditional i.i.d.)
- Inference: query the model
  - $p(y_i|\boldsymbol{x}_i, \theta) = ?$
- Learning: $\hat{\theta} = ?$

# The Key Problem

- How to model $p(y_i \mid \boldsymbol{x}_i, \theta)$?
  - $p(y_i \mid \boldsymbol{x}_i, \theta)$: what is the distribution for class labels given the current feature vector?

# Model: Linear Regression VS. Logistic Regression

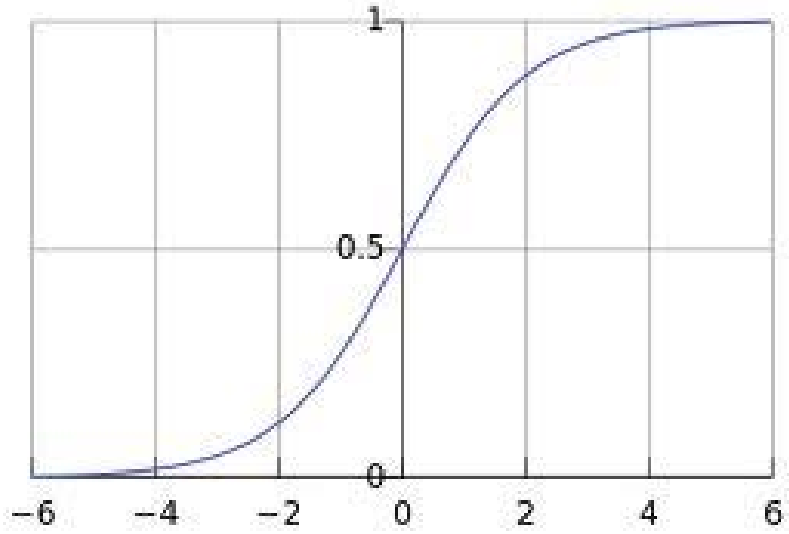
- Linear Regression (prediction)
  - Y: $continuous\ value\ (-\infty, +\infty)$
    - $y = \boldsymbol{x}^T\boldsymbol{\beta} + \varepsilon = \beta_0 + x_1\beta_1 + x_2\beta_2 + \cdots + x_p\beta_p + \varepsilon$
      - $\varepsilon \sim N(0, \sigma^2)$
    - $y|\boldsymbol{x}, \beta \sim N(\boldsymbol{x}^T\beta, \sigma^2)$

Note: add additional constant feature 1 to original $x$ to accommodate bias term

- Logistic Regression (classification)
  - Y: $discrete\ value\ from\ m\ classes$
    - $P(Y = j|\boldsymbol{x}, \boldsymbol{\beta}) \in [0,1]\ and\ \sum_j P(Y = j|\boldsymbol{x}, \boldsymbol{\beta}) = 1$

# Logistic Function

- Logistic Function / sigmoid function: $\sigma(x) = \frac{1}{1+e^{-x}}$




$$Note: \sigma'(x) = \sigma(x)(1 - \sigma(x))$$

# Modeling Probabilities of Two Classes

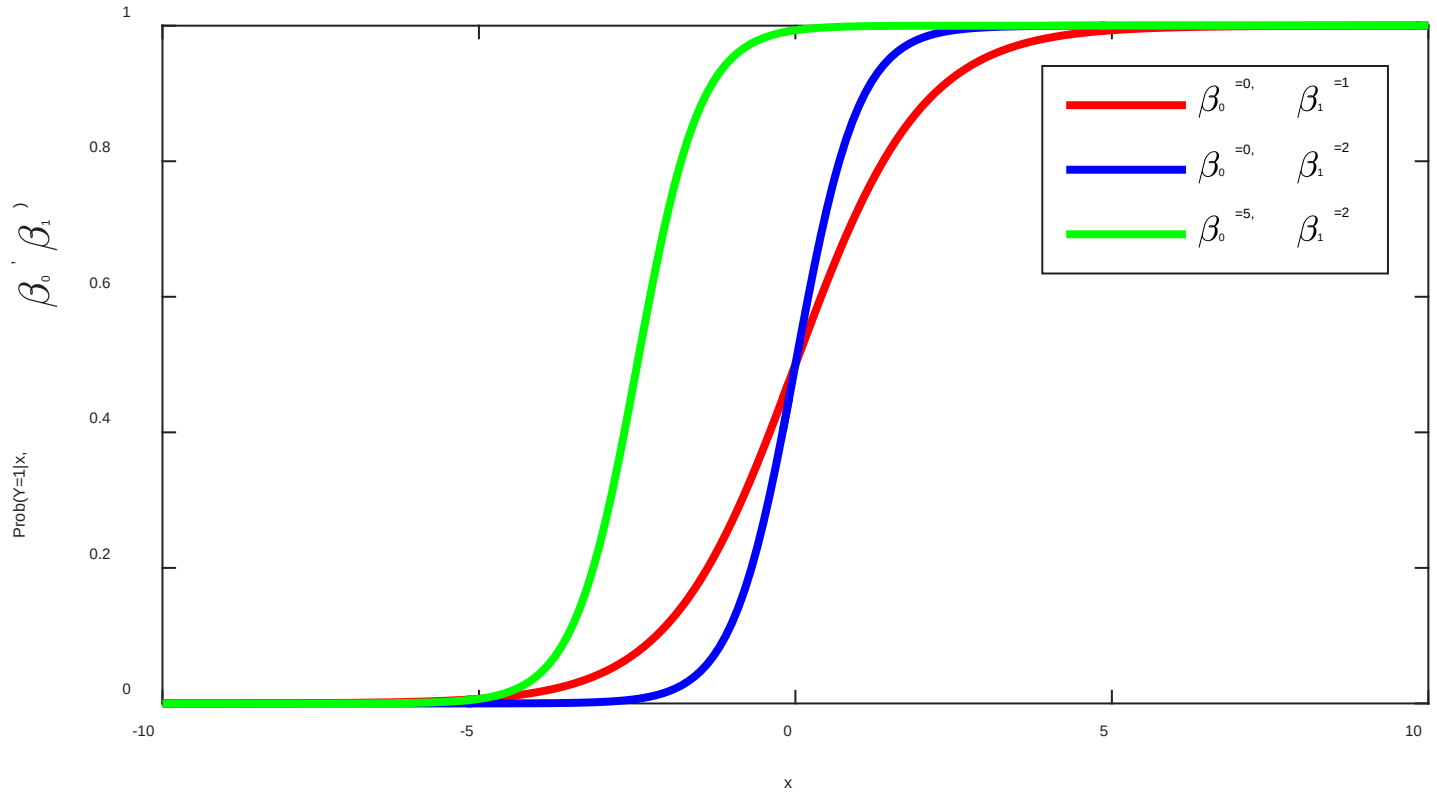
- $P(Y = 1|\boldsymbol{x}, \beta) = \sigma(\boldsymbol{x}^T\beta) = \frac{1}{1+\exp\{-\boldsymbol{x}^T\beta\}} = \frac{\exp\{\boldsymbol{x}^T\beta\}}{1+\exp\{\boldsymbol{x}^T\beta\}}$
- $P(Y = 0|\boldsymbol{x}, \beta) = 1 - \sigma(\boldsymbol{x}^T\beta) = \frac{\exp\{-\boldsymbol{x}^T\beta\}}{1+\exp\{-\boldsymbol{x}^T\beta\}} = \frac{1}{1+\exp\{\boldsymbol{x}^T\beta\}}$

$$\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$$

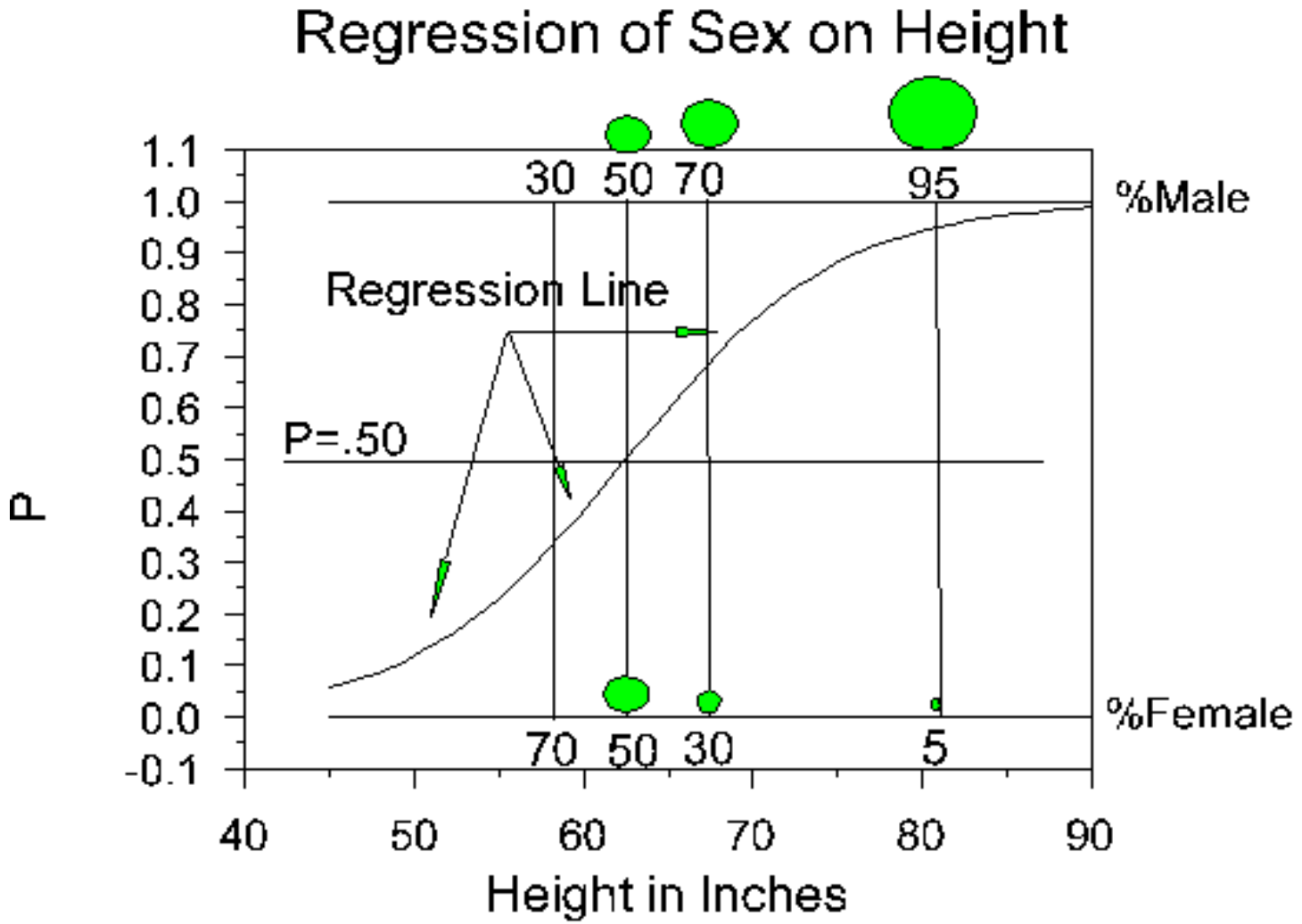
- In other words
  - $y|\boldsymbol{x}, \beta \sim Bernoulli(\sigma(\boldsymbol{x}^T\beta))$

# The 1-d Situation

- $P(Y = 1|x, \beta_0, \beta_1) = \sigma(\beta_1 x + \beta_0)$



# Example




Q: Is $\beta_0$ positive or negative here?

# Classification Assuming Parameters are Given: Inference

- $If\ P(Y = 1|\boldsymbol{x}, \beta) = \sigma(\boldsymbol{x}^T\beta) > 0.5$
  - Class 1
- Otherwise
  - Class 0

- Question:
  - What is the decision boundary? (Hint: Decision boundary is the set of points where its probability going to 1 or 0 is equal.)

# Parameter Estimation: Learning

- MLE estimation
  - Given a dataset $D, with\ N\ data\ points$
  - For a single data object with attributes $\boldsymbol{x}_i$, class label $y_i$
    - Let $p_i = p(y_i = 1|\boldsymbol{x}_i, \beta), the\ prob.of\ \boldsymbol{x}_i\ in\ class\ 1$
    - The probability of observing $y_i$ would be
      - If $y_i = 1, then\ \ p_i$
      - If $y_i = 0, then\ 1 - p_i$
      - Combing the two cases: $p_i^{y_i}(1 - p_i)^{1-y_i}$

$$L(\beta; D) = \prod_i p_i^{y_i}(1 - p_i)^{1-y_i} = \prod_i \left(\frac{\exp\{\boldsymbol{x}^T\beta\}}{1+\exp\{\boldsymbol{x}^T\beta\}}\right)^{y_i} \left(\frac{1}{1+\exp\{\boldsymbol{x}^T\beta\}}\right)^{1-y_i}$$

# Optimization

- Equivalent to maximize log likelihood
  - $logL = \sum_i y_i \boldsymbol{x}_i^T \beta - \sum_i \log\left(1 + \exp\{\boldsymbol{x}_i^T \beta\}\right)$
  - Show proof:
    - $L(\beta; D) = \prod_i p_i^{y_i}(1-p_i)^{1-y_i} = \prod_i \left(\frac{\exp\{\boldsymbol{x}^T\beta\}}{1+\exp\{\boldsymbol{x}^T\beta\}}\right)^{y_i} \left(\frac{1}{1+\exp\{\boldsymbol{x}^T\beta\}}\right)^{1-y_i}$

# Optimization

- Equivalent to maximize log likelihood
  - $logL = \sum_i y_i \boldsymbol{x}_i^T \beta - \sum_i \log\left(1 + \exp\{\boldsymbol{x}_i^T \beta\}\right)$
- Gradient ascent update:
  - $$\beta^{new} = \beta^{old} + \eta \frac{\partial logL(\beta)}{\partial \beta}$$

Step size

- Newton-Raphson update
  - $$\beta^{new} = \beta^{old} - \left(\frac{\partial^2 logL(\beta)}{\partial \beta \partial \beta^T}\right)^{-1} \frac{\partial logL(\beta)}{\partial \beta}$$
  - where derivatives are evaluated at $\beta^{\text{old}}$

# Show calculation

- $\frac{\partial logL(\beta)}{\partial \beta}$, where $logL = \sum_i y_i \boldsymbol{x}_i^T \beta - \sum_i \log\left(1 + \exp\{\boldsymbol{x}_i^T \beta\}\right)$

# First Derivative

- It is a (p+1) vector, with jth element as

$$\frac{\partial logL(\beta)}{\partial \beta_j} = \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} \frac{x_{ij} e^{\beta^T \mathbf{x}_i}}{1 + e^{\beta^T \mathbf{x}_i}}$$

$$p_i(\beta) = \sigma(\boldsymbol{\beta}^T \boldsymbol{x}_i)$$

$$= \sum_{i=1}^{N} y_i x_{ij} - \sum_{i=1}^{N} p_i(\beta) x_{ij}$$

$$= \sum_{i=1}^{N} x_{ij}(y_i - p_i(\beta))$$

For j = 0, 1, …, p

**Matrix form:**

$$\frac{\partial logL(\beta)}{\partial \beta} = \sum_i (y_i - p_i(\beta)) \boldsymbol{x}_i^T = \big(\boldsymbol{y} - p(\beta)\big)^T X$$

$note\ X \in R^{N \times (p+1)}$

# Second Derivative

- It is a (p+1) by (p+1) matrix, **Hessian Matrix**, with jth row and nth column as

$$\begin{aligned}\frac{\partial^2 logL(\beta)}{\partial\beta_j\partial\beta_n} &= -\sum_{i=1}^{N}\frac{(1+e^{\beta^T\mathbf{x}_i})e^{\beta^T\mathbf{x}_i}x_{ij}x_{in}-(e^{\beta^T\mathbf{x}_i})^2x_{ij}x_{in}}{(1+e^{\beta^T\mathbf{x}_i})^2}\\ &= -\sum_{i=1}^{N}x_{ij}x_{in}p_i(\beta)-\sum_{i=1}^{N}x_{ij}x_{in}(p_i(\beta))^2\\ &= -\sum_{i=1}^{N}x_{ij}x_{in}p_i(\beta)(1-p_i(\beta))\end{aligned}$$

For j,n = 0, 1, …, p

**Matrix form:**

$$\begin{aligned}\frac{\partial^2 logL(\beta)}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T} &= -\sum_{i=1}^{N}\mathbf{x}_i p_i(\boldsymbol{\beta})(1-p_i(\boldsymbol{\beta})\mathbf{x}_i^T\\ &= -\mathbf{X}^T\begin{bmatrix}p_1(\boldsymbol{\beta})(1-p_1(\boldsymbol{\beta}) & & \\ & \ddots & \\ & & p_N(\boldsymbol{\beta})(1-p_N(\boldsymbol{\beta})\end{bmatrix}\mathbf{X}\end{aligned}$$

where $\mathbf{X}$ is the $N\times(p+1)$ feature matrix. Note $\mathbf{X}^T=[\mathbf{x}_1\ \mathbf{x}_2\ldots\mathbf{x}_N]$.

# Tips

- Regularization is usually needed in logistic regression
  - L1 or l2 regularization
- Think about a case where the two classes are linearly separable
  - Will scaling $\beta$ into c$\beta$ (c>1) affect the decision boundary?
  - Will scaling $\beta$ into c$\beta$ (c>1) affect the likelihood function?

# What about Multiclass Classification?

- It is easy to handle under logistic regression, say M classes, using softmax function
  - $P(Y = j|x) = \frac{\exp\{x^T \beta_j\}}{1+\sum_{m=1}^{M-1} \exp\{x^T \beta_m\}}$, for j $= 1, \ldots, M-1$
  - $P(Y = M|x) = \frac{1}{1+\sum_{m=1}^{M-1} \exp\{x^T \beta_m\}}$
- What's the likelihood?

# Q&A

# Recall Linear Regression and Logistic Regression

- Linear Regression
  - $y|\boldsymbol{x}, \beta \sim N(\boldsymbol{x}^T\beta, \sigma^2)$
- Logistic Regression
  - $y|\boldsymbol{x}, \beta \sim Bernoulli(\sigma(\boldsymbol{x}^T\beta))$

- How about other distributions?
  - Yes, generalized linear models for exponential family

# *Exponential Family

- Canonical Form
  - $p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$
  - $\eta$: natural parameter
  - $T(y)$: sufficient statistic
  - $a(\eta)$: log partition function for normalization
  - $b(y)$: function that only dependent on $y$

# *Examples of Exponential Family

- Many:

$$p(y;\eta) = b(y)\exp\left(\eta^T T(y) - a(\eta)\right)$$

  - Gaussian, Bernoulli, Poisson, beta, Dirichlet, categorical, …

- For Gaussian (not interested in $\sigma$)

$$\begin{aligned} p(y;\mu) &= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(y-\mu)^2\right) \\ &= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}y^2\right)\cdot\exp\left(\mu y - \frac{1}{2}\mu^2\right) \end{aligned}$$

$$\begin{aligned} \eta &= \mu \\ T(y) &= y \\ a(\eta) &= \mu^2/2 \\ &= \eta^2/2 \\ b(y) &= (1/\sqrt{2\pi})\exp(-y^2/2) \end{aligned}$$

- For Bernoulli

$$\begin{aligned} p(y;\phi) &= \phi^y(1-\phi)^{1-y} \\ &= \exp(y\log\phi + (1-y)\log(1-\phi)) \\ &= \exp\left(\left(\underbrace{\log\left(\frac{\phi}{1-\phi}\right)}_{\eta}\right)y + \log(1-\phi)\right) \end{aligned}$$

$$\begin{aligned} T(y) &= y \\ a(\eta) &= -\log(1-\phi) \\ &= \log(1+e^\eta) \\ b(y) &= 1 \end{aligned}$$

# *Recipe of GLMs

- Determines a distribution for y
  - E.g., Gaussian, Bernoulli, Poisson
- Form the linear predictor for $\eta$
  - $\eta = \boldsymbol{x}^T \beta$
- Determines a link function: $\mu = g^{-1}(\eta)$
  - Connects the linear predictor to the mean of the distribution
  - E.g., $\mu = \eta$ for Gaussian, $\mu = \sigma(\eta)$ for Bernoulli, $\mu = exp(\eta)$ for Poisson

# Content

- Probabilistic Models
- Naïve Bayes
- Logistic Regression
- Generative Models and Discriminative Models 
- Summary

# Generative Models vs. Discriminative Models

- Generative model
  - $model\ joint\ probability\ p(\boldsymbol{x}, y)$
  - E.g., naïve Bayes
- Discriminative model
  - $model\ conditional\ probability\ p(y|\boldsymbol{x})$
  - E.g., logistic regression

# Which One is Better?

- Consider $p(\boldsymbol{x}, y) = p(y|\boldsymbol{x}) \times p(\boldsymbol{x})$
  - Generative models require additional model of marginal distribution $p(\boldsymbol{x})$
    - Need more data to learn $p(\boldsymbol{x})$
    - Distribution assumption of $p(\boldsymbol{x})$ might be incorrect
  - In practice, discriminative models work very well

https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf

# Content

- Probabilistic Models for I.I.D. Data
- Naïve Bayes
- Logistic Regression
- Generative Models and Discriminative Models
- Summary 

# Summary

- Probabilistic Models
  - I.I.D. assumption enables joint distribution of data as a product of probability of single data points
- Naïve Bayes
  - Assuming independence among features
- Logistic Regression
  - Assuming conditional distribution follows Bernoulli distribution
- Generative Models and Discriminative Models
  - Modeling joint distribution vs. conditional distribution

# Q&A

# References


- http://pages.cs.wisc.edu/~jerryzhu/cs769/nb.pdf
- http://cs229.stanford.edu/notes/cs229-notes1.pdf
- https://ai.stanford.edu/~ang/papers/nips01-discriminativegenerative.pdf

# More about Lagrangian

- Objective with equality constraints

$$\min_{w} f(w)$$

$$s.t.$$

$$h_i(w) = 0, for\ i = 1,2, \dots, l$$

- Lagrangian:
  - $L(w, \boldsymbol{\alpha}) = f(w) + \sum_i \alpha_i h_i(w)$
    - $\alpha_i$: Lagrangian multipliers
- Solution: setting the derivatives of Lagrangian to be 0
  - $\frac{\partial L}{\partial w} = 0\ and \frac{\partial L}{\partial \alpha_i} = 0$ for every $i$

# Generalized Lagrangian

- Objective with both equality and inequality constraints

$$\min_{w} f(w)$$

$$s.t.$$

$$h_i(w) = 0, for\ i = 1,2, \dots, l$$

$$g_j(w) \leq 0, for\ j = 1,2, \dots, k$$

- Lagrangian
  - $L(w, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(w) + \sum_i \alpha_i h_i(w) + \sum_j \beta_j g_j(w)$
    - $\alpha_i$: Lagrangian multipliers
    - $\beta_j \geq 0$: Lagrangian multipliers

# Why It Works

- Consider function

$$\theta_p(w) = \max_{\alpha,\beta:\beta_j \geq 0} L(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

- $\theta_p(w) = \begin{cases} f(w), & if\ w\ satisfies\ all\ constraints \\ \infty, & if\ w\ doesn't\ satisfy\ constraints \end{cases}$
- Therefore, minimize $f(w)$ with constraints is equivalent to minimize $\theta_p(w)$

# Lagrange Duality

- The primal problem

$$p^* = \min_{w} \max_{\alpha,\beta:\beta_j \geq 0} L(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

- The dual problem

$$d^* = \max_{\alpha,\beta:\beta_j \geq 0} \min_{w} L(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$$

- According to max-min inequality

$$p^* \geq d^*$$

  - When does equation hold?

# Primal = Dual

- $p^* = d^*$, under some proper condition (Slater conditions)
  - $f, g_j$ convex, $h_i$ affine
  - Exists $w$, such that all $g_j(\mathrm{w}) < 0$
- $(w^*, \alpha^*, \beta^*)$ need to satisfy KKT conditions
  - $\frac{\partial L}{\partial w} = 0$
  - $\beta_j g_j(w) = 0$
  - $h_i(w) = 0, g_j(w) \leq 0, \beta_j \geq 0$

https://cs.stanford.edu/people/davidknowles/lagrangian_duality.pdf