Lecture 06. Logistic regression

Xin Chen

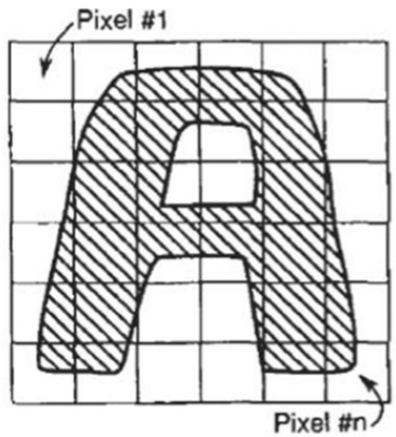
These slides are based on slides from Mahdi Roozbahani

Outline

- Generative classification and discriminative classification
- The logistic regression model
- Understanding the objective model
- Gradient descent for parameter learning
- Multiclass logistic regression

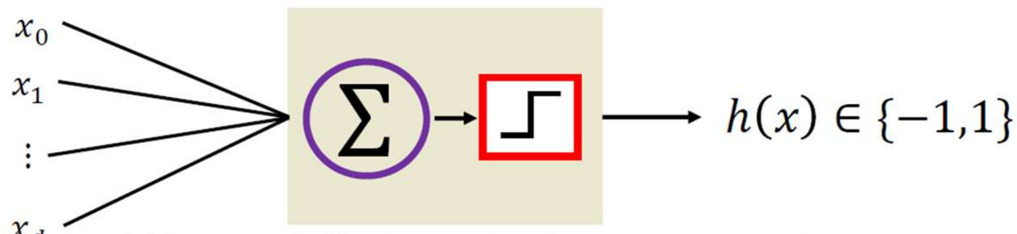
Classification

- Represent the data



$$X = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{bmatrix}$$

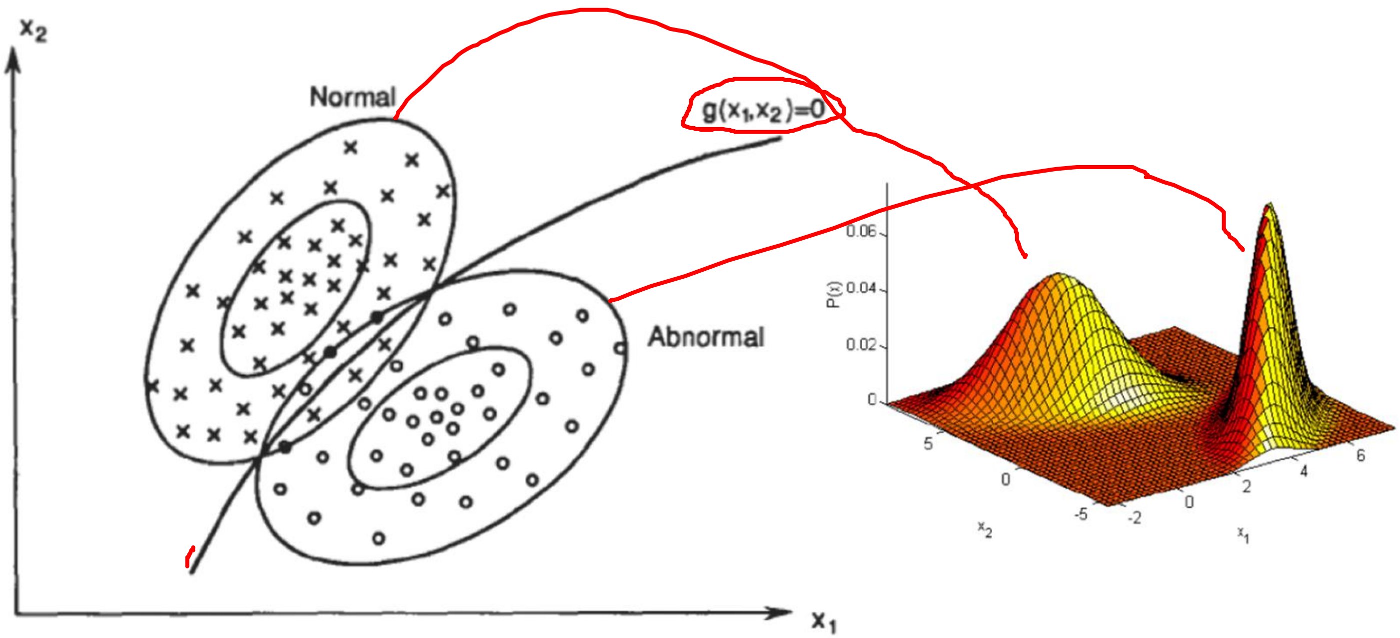
- A label is provided for each data point, $y \in \{-1, +1\}$
- Classifier:



$h(x) = sign(x\theta) \rightarrow$ linear classification (Perceptron)

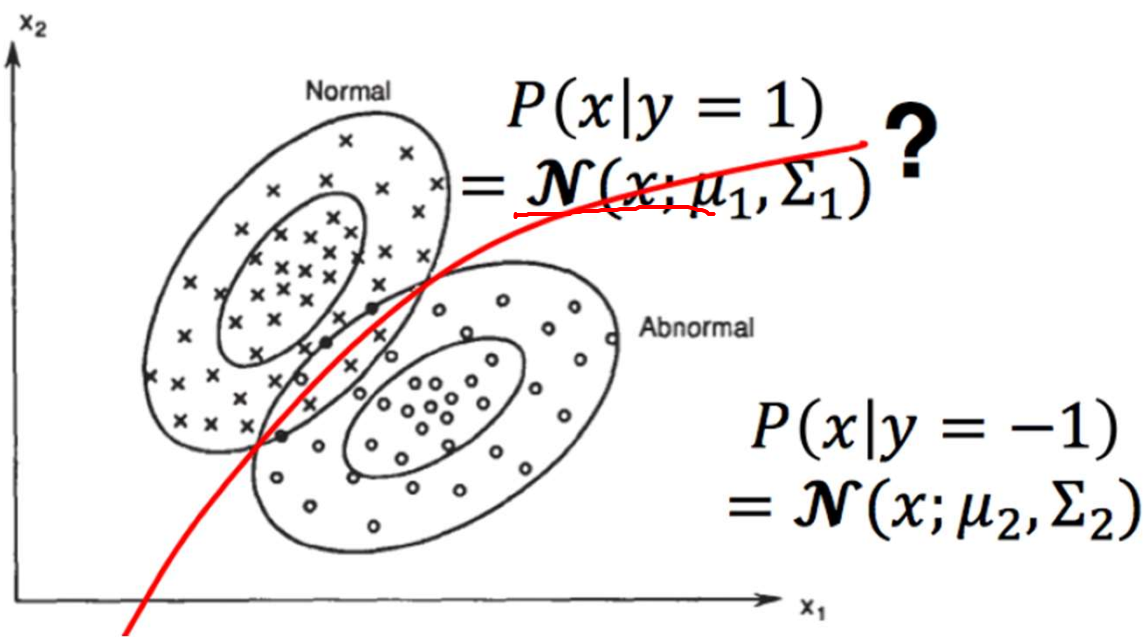
Decision making: dividing the feature space

- Distribution of sample from normal (positive class) and abnormal (negative class) issues.



How to determine the decision boundary?

- Given class conditional distribution: $P(x|y=1), P(x|y=-1)$ and class prior: $P(y=-1), P(y=1)$



Bayes Decision Rule

$$\underbrace{P(y|x)}_{\text{posterior}} = \frac{\overbrace{P(x|y)}^{\text{likelihood}}\,\overbrace{P(y)}^{\text{Prior}}}{\underbrace{P(x)}_{\text{normalization constant}}} = \frac{P(x,y)}{\sum_z P(x,y)}$$

- Prior: $P(y)$
- Class conditional distribution: $P(x|y) = \boldsymbol{\mathcal{N}}(x|u_y, \sum y)$
- Posterior: $P(y|x) = \frac{\boldsymbol{\mathcal{N}}(x|u_y,\sum y)}{\sum p(y)\boldsymbol{\mathcal{N}}(x|u_y,\sum y)}$

Bayes Decision Rule

- Learning: (1) Prior: $P(y)$(2)Condition distribution: $P(x|y)$
- The poster probability of a test point $q_i(x) \coloneqq P(y = i|x) = \frac{P(x|y)P(y)}{P(x)}$
- Bayes decision rule:
  - If $q_i(x) > q_j(x)$, then $y = i$, otherwise $y = j$
- Alternatively
  - If ratio $\mathrm{l}(x) = \frac{P(x|y=i)}{P(x|y=j)} > \frac{P(y=j)}{P(y=i)}$, then $y = i$, otherwise $y = j$
  - Or look at the log-likelihood ratio $h(x) = -\ln(x)\frac{q_i(x)}{q_j(x)}$

What do people do in practice

- Generative model
  - Model prior and likelihood explicitly
  - “Generative” means able to generate synthetic data points
  - Examples: Naive Bayes, Hidden Markov models

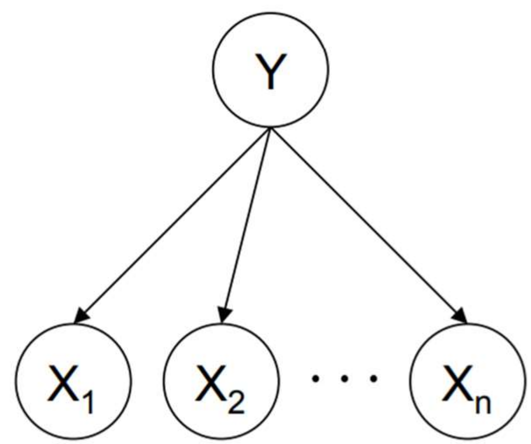
- Discriminative models
  - Directly estimate the posterior probabilities
  - No need to model underlying prior distributions
  - Examples: Logistic regression, SVM, Neural network

Generative Model: Naive Bayes

- Use Bayes decision rule for classification
- Assume $p(x|y=1)$ is fully factorized: dimensions are independent.
- Or the variables corresponding to each dimension of the data are independent given the label

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

$$P(x|y=1) = \prod_{i=1}^{d} p(x_i|y=1)$$



$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

Join probability model

$$P(x, y_{label=1}) = \mathrm{P}(\mathrm{x}_1, \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) = \mathrm{P}(\mathrm{x}_1 | \mathrm{x}_2, , \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) P(\mathrm{x}_2, \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}})$$

$$= \mathrm{P}(\mathrm{x}_1 | \mathrm{x}_2, , \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) P(\mathrm{x}_2 | \mathrm{x}_3 \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) P(\mathrm{x}_3, \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}})$$

$$= \cdots$$

$$= \mathrm{P}(\mathrm{x}_1 | \mathrm{x}_2, , \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) P(\mathrm{x}_2 | \mathrm{x}_3 \dots, \mathrm{x}_\mathrm{d}, \mathrm{y}_{\mathrm{label=1}}) \dots P(\mathrm{x}_{\mathrm{d}-1} | x_d, y_{label=1}) P(x_d | y_{label=1}) \mathrm{P}(y_{label=1})$$

Naive Bayes assumption:

$$P(x, y_{label}) = P(x_1 | y_{label=1}) P(x_2 | y_{label}) \dots P(x_n | y_{label=1}) P(y_{label}) = P(y_{label=1}) \prod_{i=1}^{d} P(x_i | y_{label=1})$$

Discriminative models

- Directly estimate decision boundary: the posterior distribution $p(y|x)$ or $h(x) = -\ln(x)\frac{q_i(x)}{q_j(x)}$
  - Logistic regression, Neural networks
  - Do not estimate $p(x|y)$ and $p(y)$

- Why discriminative classifier?
  - Avoid difficult density estimation problem
  - Empirically achieve better classification results

Outline

- Generative classification and discriminative classification
- The logistic regression model
- Understanding the objective model
- Gradient descent for parameter learning
- Multiclass logistic regression

Gaussian Naive Bayes

$$P(y=1|x) = \frac{P(x|y=1)P(y=1)}{P(x)} = \frac{P(y=1)P(x|y=1)}{P(y=1)P(x|y=1)+P(y=-1)P(x|y=-1)}$$

$$= \frac{1}{1+\frac{P(y=-1)P(x|y=-1)}{P(y=1)P(x|y=1)}} \Rightarrow S$$

$$\boxed{P(x_i|y) \sim \mathcal{N}(u_{ki}, \sigma_i)}$$

Class independent variance

$$P(x|y) = \prod_{i=1}^{d} p(x_i|y) = \prod_{i=1}^{d} \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2\sigma_i{}^2}(x_i - u_i)^2\right)$$

Prior: $P(y=1) = \pi_1$

$$S = \frac{P(y=-1)P(x|y=-1)}{P(y=1)P(x|y=1)} = \frac{(1-\pi_1)\prod_{i=1}^{d}\frac{1}{\sqrt{2\pi}\sigma_i}\exp(-\frac{1}{2\sigma_i{}^2}(x_i-u_{0i})^2)}{\pi_1\prod_{i=1}^{d}\frac{1}{\sqrt{2\pi}\sigma_i}\exp(-\frac{1}{2\sigma_i{}^2}(x_i-u_{1i})^2)}$$

$$\ln(S) = ln\frac{1-\pi_1}{\pi_1} + \sum_{i=1}^{d}\ln[\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{1}{2\sigma_i{}^2}(x_i-u_{0i})^2\right)] - \sum_{i=1}^{d}\ln[\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{1}{2\sigma_i{}^2}(x_i-u_{1i})^2\right)]$$

$$= \sum_{i=1}^{d}(\frac{u_{0i}-u_{1i}}{\sigma_i{}^2}x_i + \frac{u_{1i}{}^2-u_{0i}{}^2}{2\sigma_i{}^2}) + ln\frac{1-\pi_1}{\pi_1}$$

$$P(y=1|x) = \frac{1}{1+\exp[\ln(s)]}$$

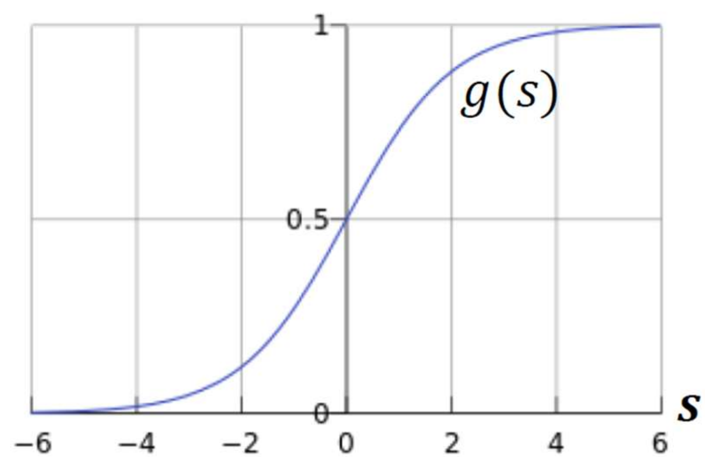
$$P(y=1|x) = \frac{1}{1+\exp[\sum_{i=1}^{d}(\frac{u_{0i}-u_{1i}}{\sigma_i^{\,2}}x_i + \frac{u_{1i}^{\,2}-u_{0i}^{\,2}}{2\sigma_i^{\,2}}) + ln\frac{1-\pi_1}{\pi_1}\ ]}$$

Let: $w_i = \frac{u_{0i}-u_{1i}}{\sigma_i^{\,2}},\ w_0 = ln\frac{1-\pi_1}{\pi_1} + \sum_{i=1}^{n}\frac{u_{1i}^{\,2}-u_{0i}^{\,2}}{2\sigma_i^{\,2}}$

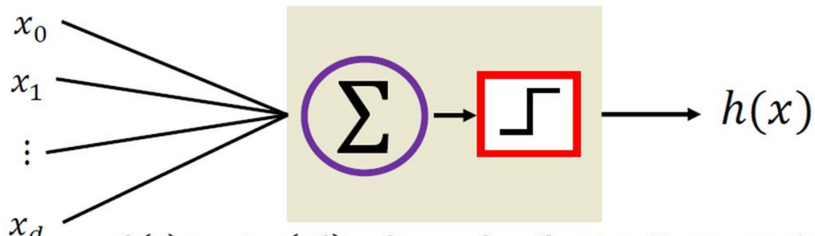
$$P(y=1|x) = \frac{1}{1+\exp(w_0 + \sum_{i=1}^{n} w_i x_i)}$$

Logistic function for posterior probability

- Let's use the following function:
  $g(s) = \frac{e^s}{1+e^s} = \frac{1}{1+e^{-s}}$ where $s = x\theta$
- This is also called sigmoid function
- It's easier to use this function for optimization
- Logistic regression assumption: the form of $P(y=0|x,\theta) = \frac{1}{1+\exp(-\sum \theta x_i)}$

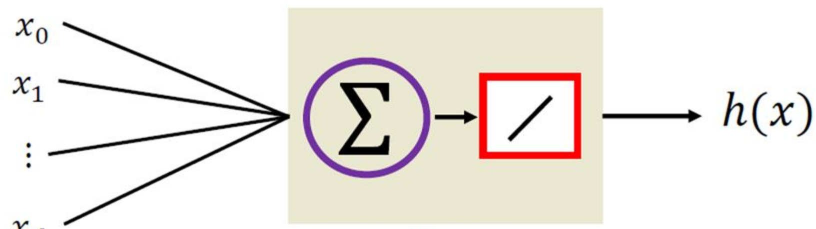


$$s = x\theta = \sum_{i=0}^{d} x_i \theta_i$$

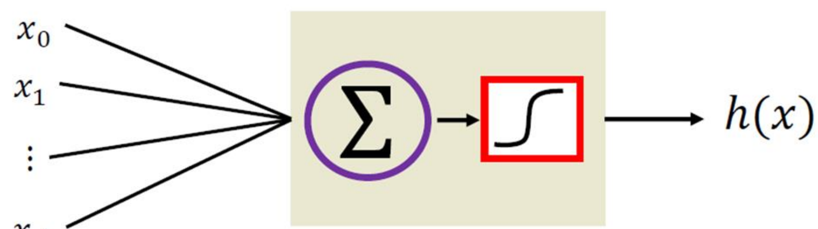


Hard classification

$h(x) = \ sign(x\theta) \rightarrow$ linear classification (Perceptron)



$h(x) \rightarrow$ linear regression



Soft classification
Posterior probability

$h(x) = u(x\theta) \rightarrow$ logistic regression

An example

- An example of predicting heart attacks
- Inputs: cholesterol level, age, weight, foot size, etc.
  - $g(s)$ is the probability of heart attack within a certain time
  - $s = x\theta$, is called risk score.

$$h_\theta(x) = p(y|x) = \begin{cases} g(s), & y = 0 \\ 1 - g(s), & y = 1 \end{cases}$$

Using posterior probability directly

$$h_\theta(x) = p(y|x) = \begin{cases} \dfrac{1}{1+\exp(-x\theta)}, & y = 0 \\ \dfrac{\exp(-x\theta)}{1+\exp(-x\theta)}, & y = 1 \end{cases}$$

We need to find parameters θ, let's set up log-likelihood for n data points:

$$l(\theta) = \log \prod_{i=1}^{n} p(y_i|x_i,\theta) \qquad l(\theta) = log \prod_{i=1}^{n} g(x_i)^{y_i}(1-g(x_i))^{(y_i-1)}$$

$$l(\theta) = \sum_{i=1}^{n} [\theta^T {x_i}^T (y_i - 1) - \log(1+\exp(-x_i\theta)]$$

Outline

- Generative classification and discriminative classification
- The logistic regression model
- Understanding the objective model
- Gradient descent for parameter learning
- Multiclass logistic regression

Calculate gradient of $l(\theta)$

$$l(\theta) = \sum_{i=1}^{n} [\theta^T x_i{}^T (y_i - 1) - \log(1 + \exp(-x_i \theta))]$$

- Maximum conditional likelihood on data by calculate its gradient

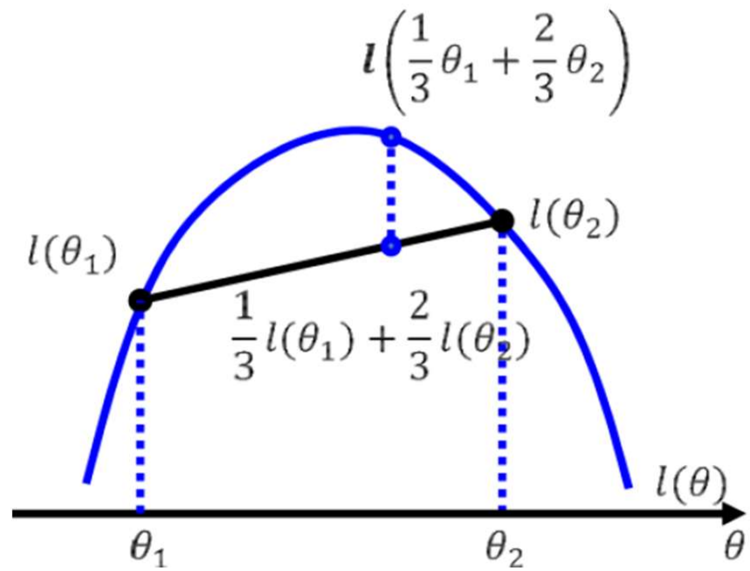
$$\frac{\partial l(\theta)}{\partial \theta} = \sum_i x_i^T (y_i - 1) + x_i^T \frac{\exp(-x_i \theta)}{1 + \exp(-x_i \theta)}$$

Logistic regression only models $P(y|x)$, so we only maximize $P(y|x)$, ignoring $P(x)$

The objective function

- Find θ such that the conditional likelihood of the labels is maximized.

max $l(\theta) = \log \prod_{i=1}^{n} p(y_i|x_i,\theta)$



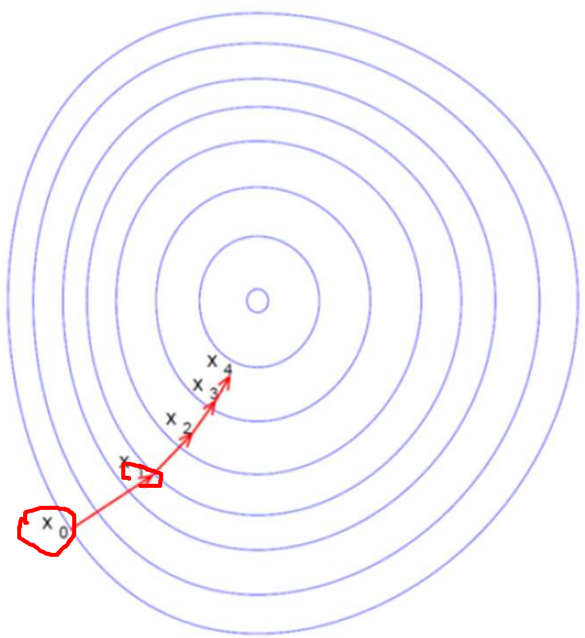
- Good news: $l(\theta)$ is concave function of θ, and there is a single global optimum.
- Bad news: no closed form solution (resort to numerical method)

Outline

- Generative classification and discriminative classification
- The logistic regression model
- Understanding the objective model
- Gradient descent for parameter learning
- Multiclass logistic regression

Gradient descent

- One way to solve an unconstrained optimization problem is gradient descent.
- Given an initial guess, we iteratively refine the guess by taking the direction of the negative gradient.
- Think about going down a hill by taking the steepest direction at each step.
- Update rule:
  - $x_{k+1} = x_k - \gamma_k \nabla f(x_k)$
  - γ_k is called the step size or learning rate.



Gradient descent algorithm

- Initialize parameter θ_0
- Do

$$\theta^{t+1} \leftarrow \theta^t + \eta \sum_i x_i^T (y_i - 1) + x_i^T \frac{\exp(-x_i\theta)}{1 + \exp(-x_i\theta)}$$

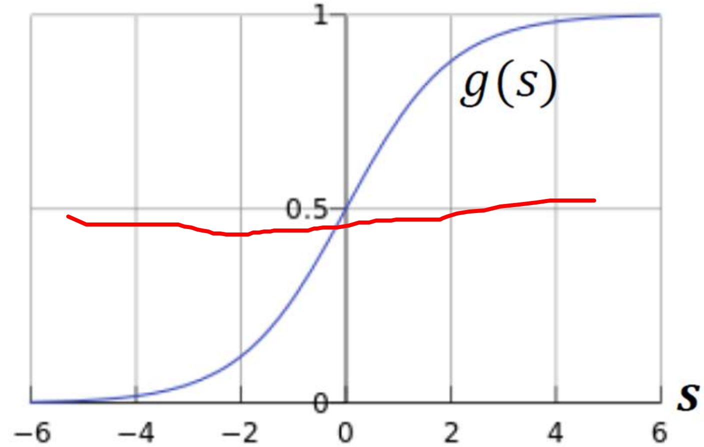
- While the $||\theta^{t+1} - \theta^t|| > \epsilon$

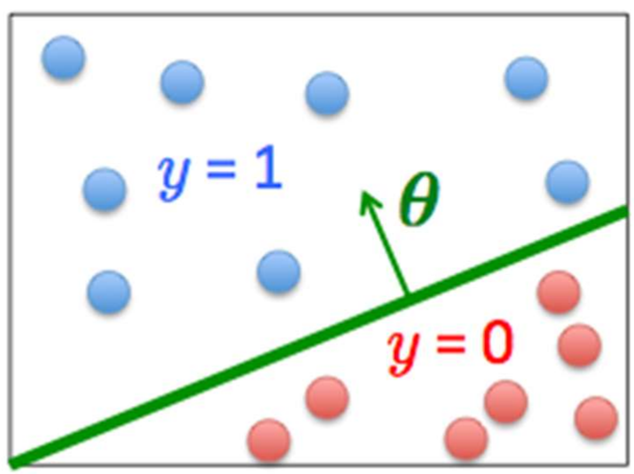
Outline

- Generative classification and discriminative classification
- The logistic regression model
- Understanding the objective model
- Gradient descent for parameter learning
- Multiclass logistic regression

Logistic regression

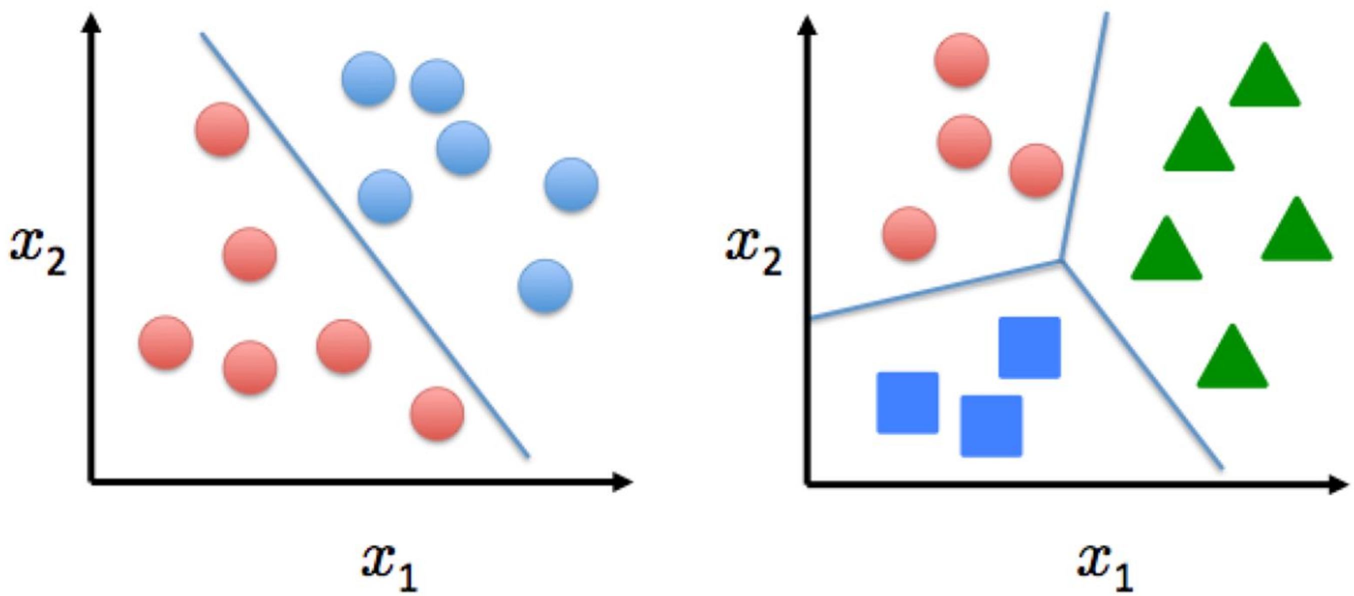
$$g(s) = \frac{e^s}{1+e^s} = \frac{1}{1+e^{-s}} \text{ where } s = x\theta$$





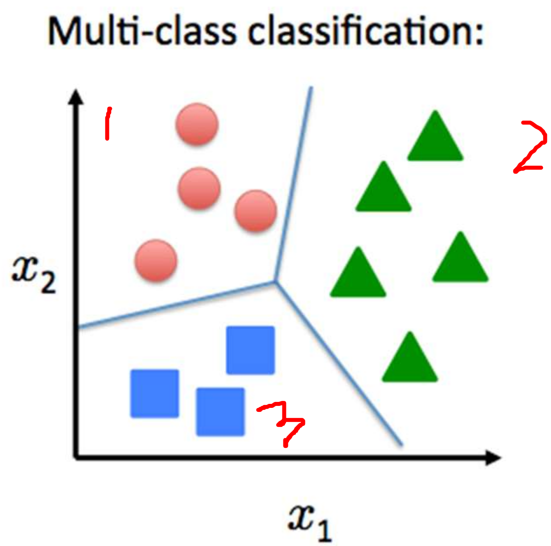
- Assume a threshold
  - Predict $y = 1$ if $g(s) > 0.5$
  - Predict $y = 0$ if $g(s) \leq 0.5$

Multiclass Logistic regression

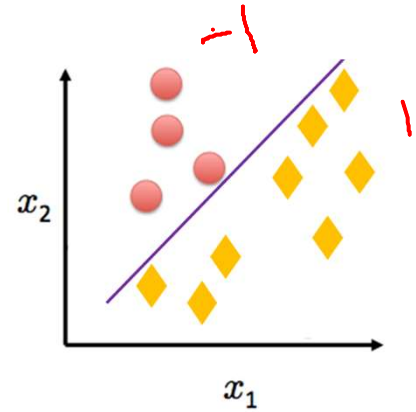


- Disease diagnosis: healthy / cold / flu / pneumonia
- Object classification: desk / chair / monitor / bookcase

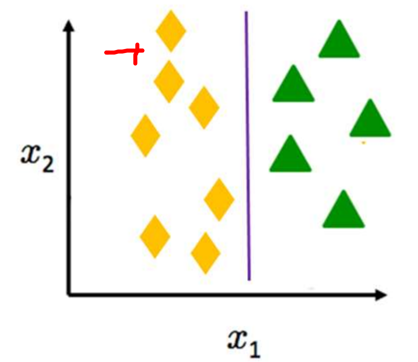
One-vs-All (One-vs-Rest)



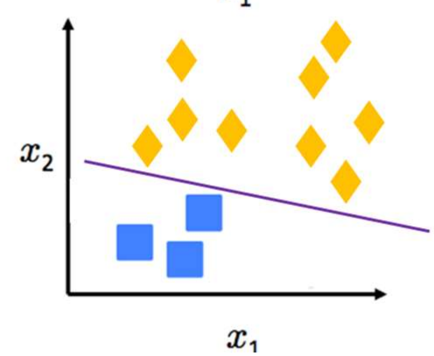
$$h_\theta^{(i)}(x) = p(y = 1|x, \theta)\ (i = 1,2,3)$$



$h_\theta^1(x)$



$h_\theta^2(x)$



$h_\theta^3(x)$

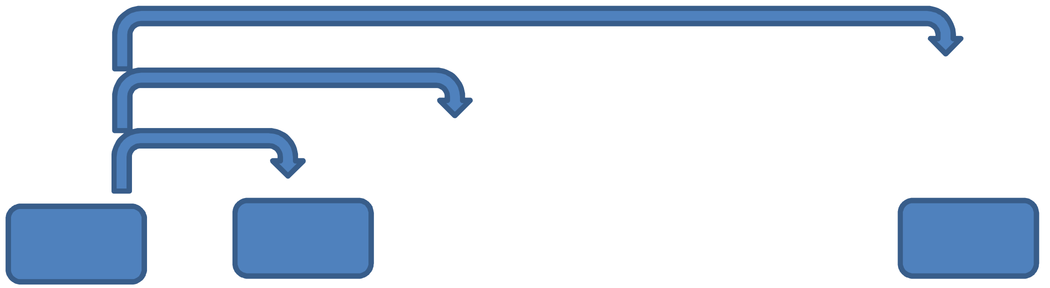
One-vs-All (One-vs-Rest)

Train a logistic regression $h_\theta^{(i)}(x)$ for each class $\boldsymbol{i}$

To predict the label of a new input x, pick class i that maximizes:

$$\max_i h_\theta^{(i)}(x)$$

One-vs.-One



K classes

In total it has $\frac{K*(K-1)}{2}$ combinations

Train logistic regression $h_\theta^{(i)}(x)$, $\frac{K*(K-1)}{2}$ binary classifiers

To predict the label of a new input x, pick class i that maximizes: $\max_i h_\theta^{(i)}(x)$

Vote with a combined classifier

Generative and discriminative classifier

- Generative classifiers
  - Modeling the joint distribution $P(x, y)$ [illegible]
  - Usually via $P(x, y) = P(y)P(x|y)$
  - Example: Gaussian naive Bayes

- Discriminative classifiers
  - Modeling $P(y|x)$ or simply $f: x \rightarrow y$
  - Do not care about $P(x)$
  - Examples: logistic regression, support vector machine

Gaussian Naive Bayes vs Logistic regression

- How can we compare Gaussian naive Bayes with a logistic regression?
  - $P(x, y) = P(y)P(x|y)$ vs. $P(y|x)$

$$P(y = 1|x) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^{n} w_i x_i)}$$

where: $w_i = \frac{u_{0i} - u_{1i}}{\sigma_i{}^2}$, $w_0 = ln \frac{1-\pi_1}{\pi_1} + \sum_{i=1}^{n} \frac{u_{1i}{}^2 - u_{0i}{}^2}{2\sigma_i{}^2}$

$\boxed{P(x_i|y) \sim \mathcal{N}(u_{ki}, \sigma_i)}$

Class independent variance

Gaussian Naive Bayes vs Logistic regression

- $P(y|x)$ of GNB is a subset of $P(y|x)$ of LR, with the assumption that GNB has independent variance.

- Given infinite training data:
  - We claim: LR >= GNB

- For a general Gaussian Naive Bayes, none of them can encompass the other

Take-Home Messages

- Generative and Discriminative Classification
- The Logistic Regression Model
- Understanding the Objective Function
- Gradient Descent for Parameter Learning
- Multiclass Logistic Regression