Happy Wednesday!

- Quiz 8, Friday, Oct 16th 6am until Oct 17th 11:59am (noon)
  - Regularization and Naïve Bayes
- Assignment 3 Early bird special → 1 complete programming question by Mon, Oct 19th 11:59pm (midnight)

CS4641B Machine Learning

Lecture 16: Logistic regression

Rodrigo Borela ▸ rborelav@gatech.edu

These slides are adopted based on slides from Le Song, Eric Eaton, and Chao Zhang and Mahdi Roozbahani

Outline

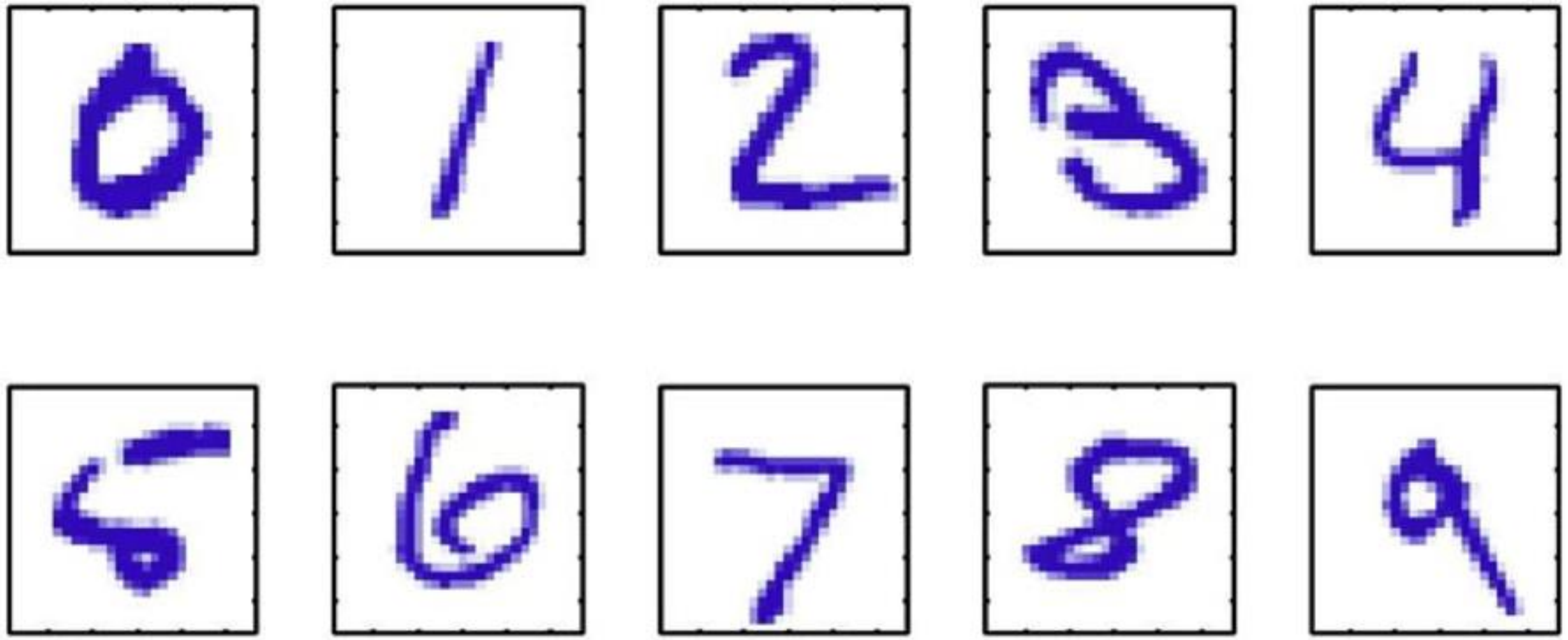
- Generative and Discriminative Classification
- The Logistic Regression Model
- Understanding the Objective Function
- Gradient Descent for Parameter Learning
- Multiclass Logistic Regression

- *Complementary reading: Bishop PRML – Chapter 1, Section 1.5; Chapter 4, Section 4.1 through 4.3.*

Outline

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

Classification

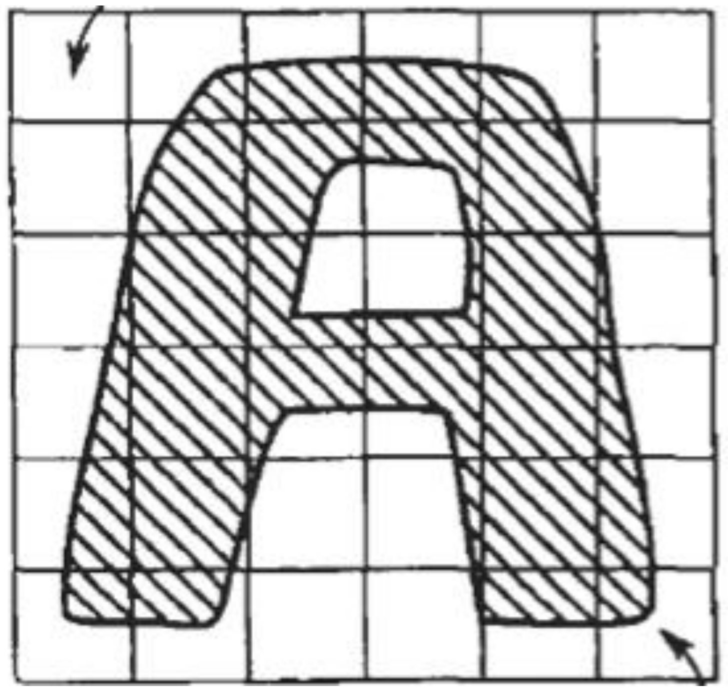


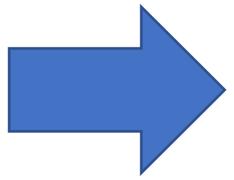
- Images are 28×28 pixels
- Represent input image as a vector $\mathbf{x} \in \mathbb{R}^{784}$
- Learn a classifier $f(\mathbf{x})$ such that,

$$f: \mathbf{x} \rightarrow \{0,1,2,3,4,5,6,7,8,9\}$$

Classification

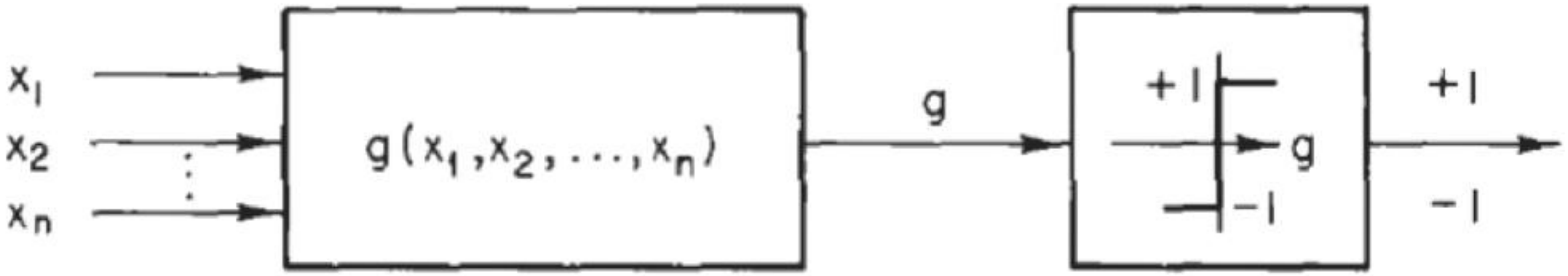
- Represent the data



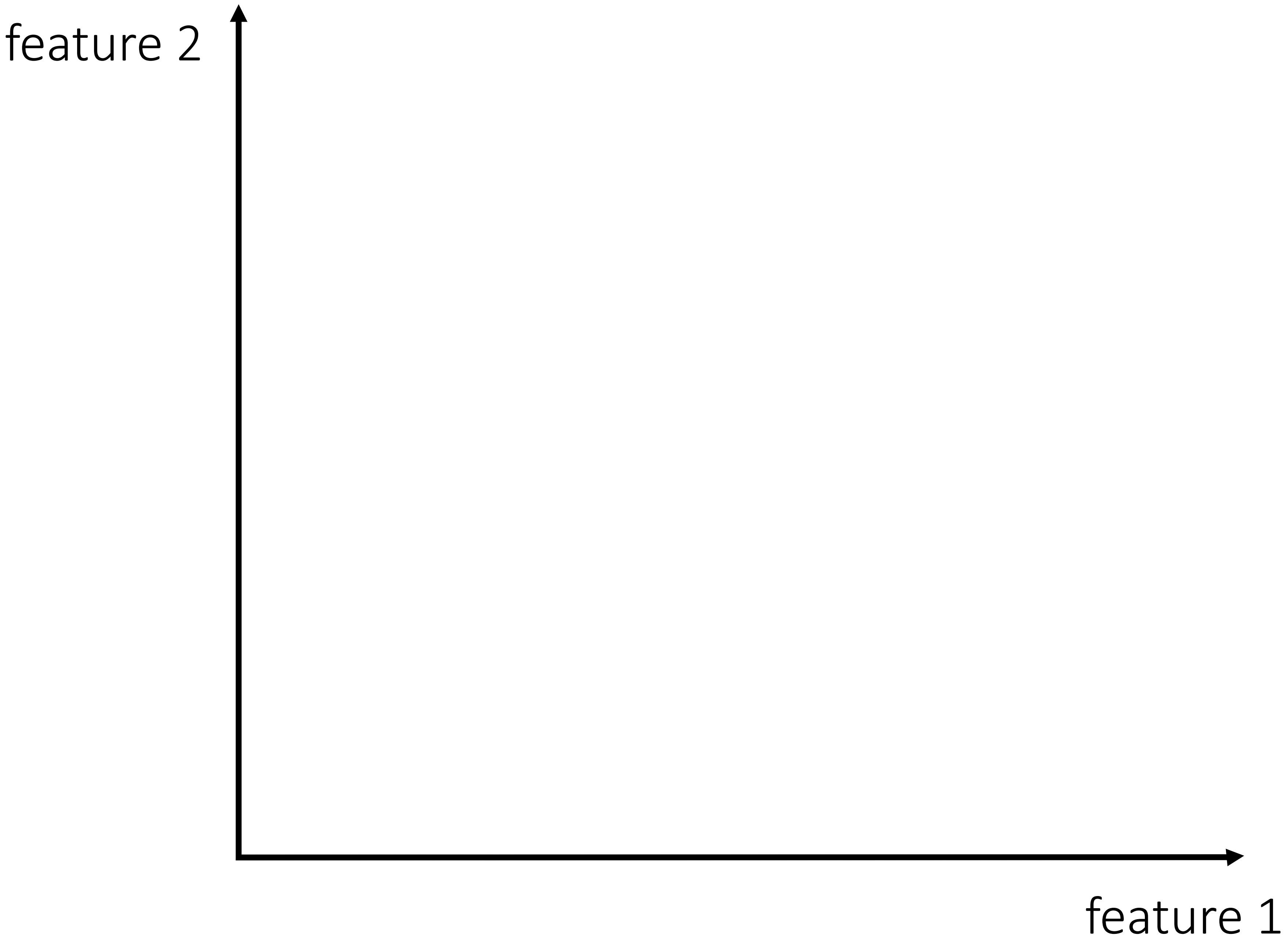


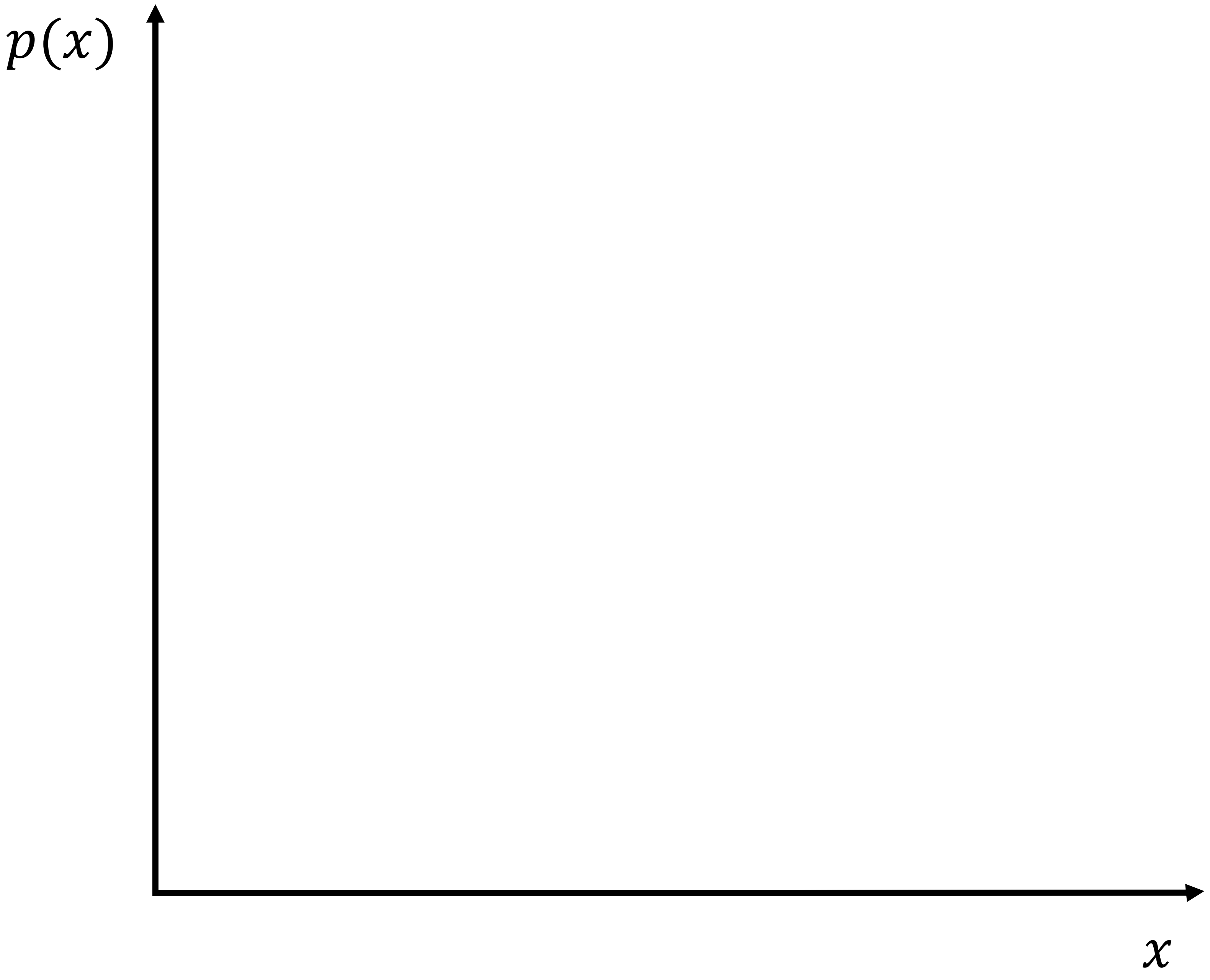
$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix}$$

- A label (target) is provided for each data point, e.g. $t_n \in \{-1, +1\}$
- Train a classifier:



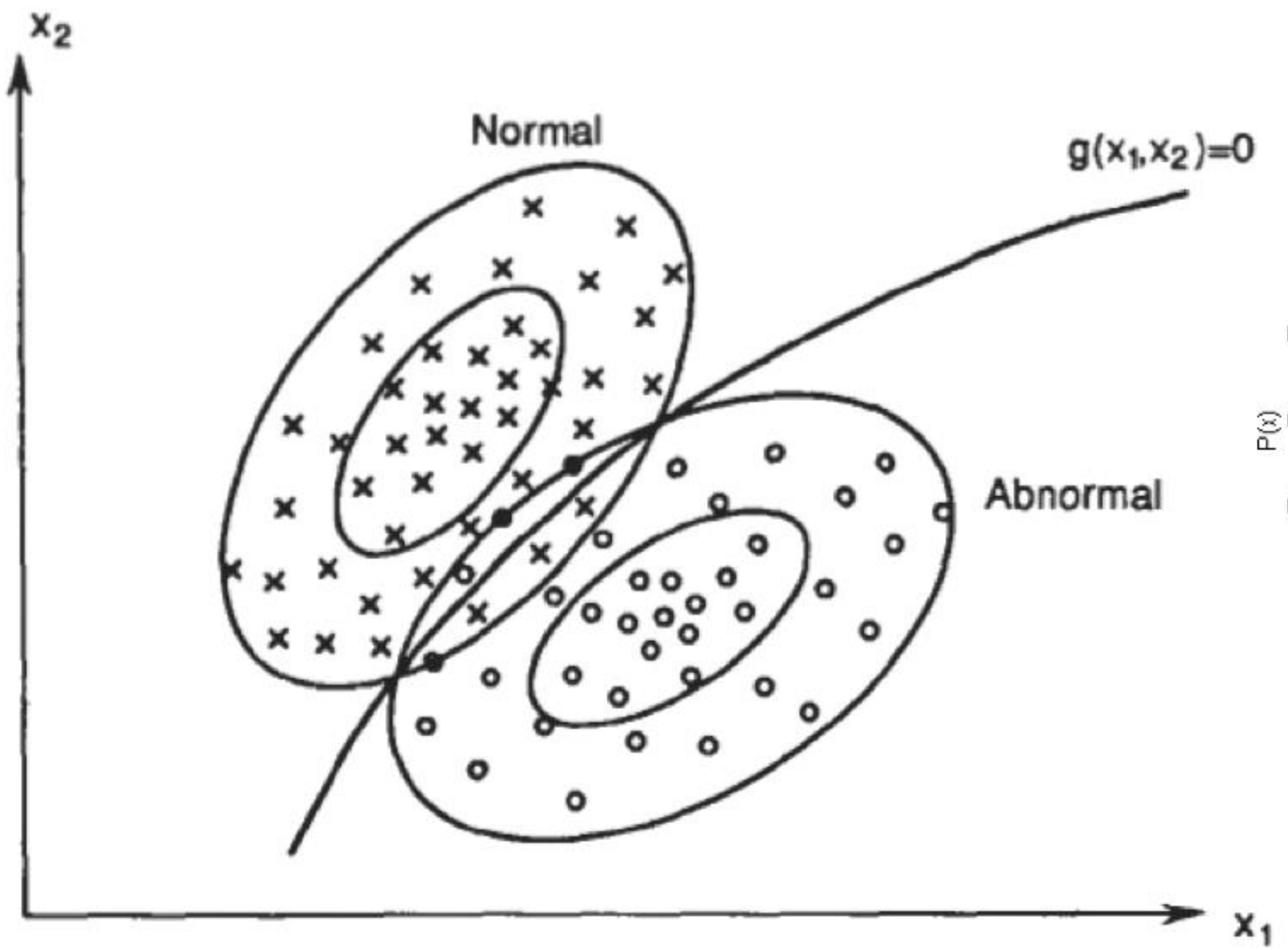
Decision making: intuition

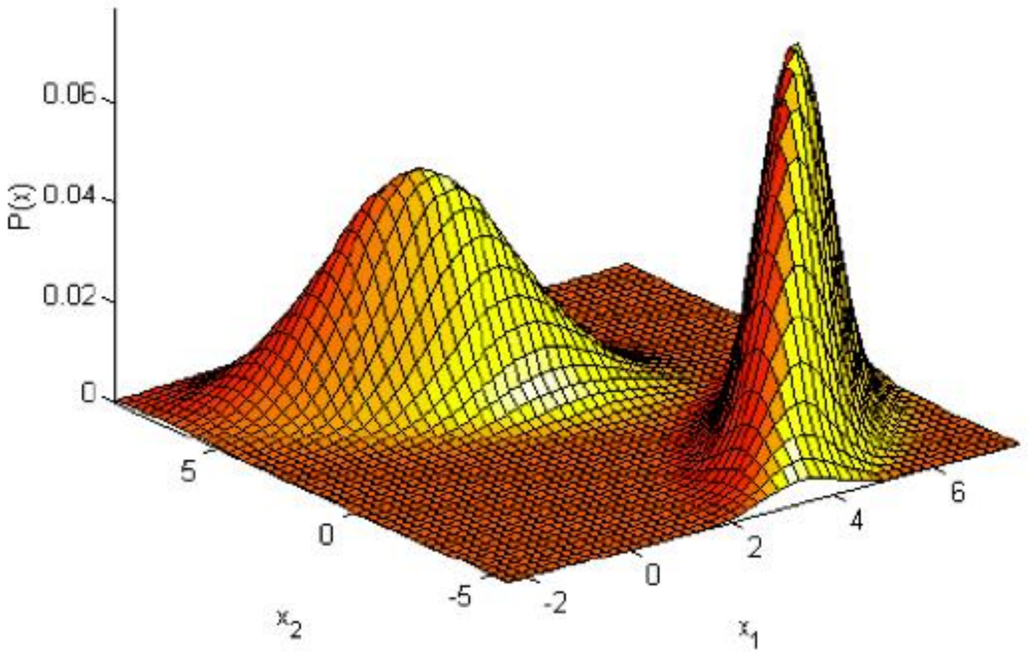




Decision making: dividing the feature space

- Distributions of sample from normal (positive class) and abnormal (negative class) tissues

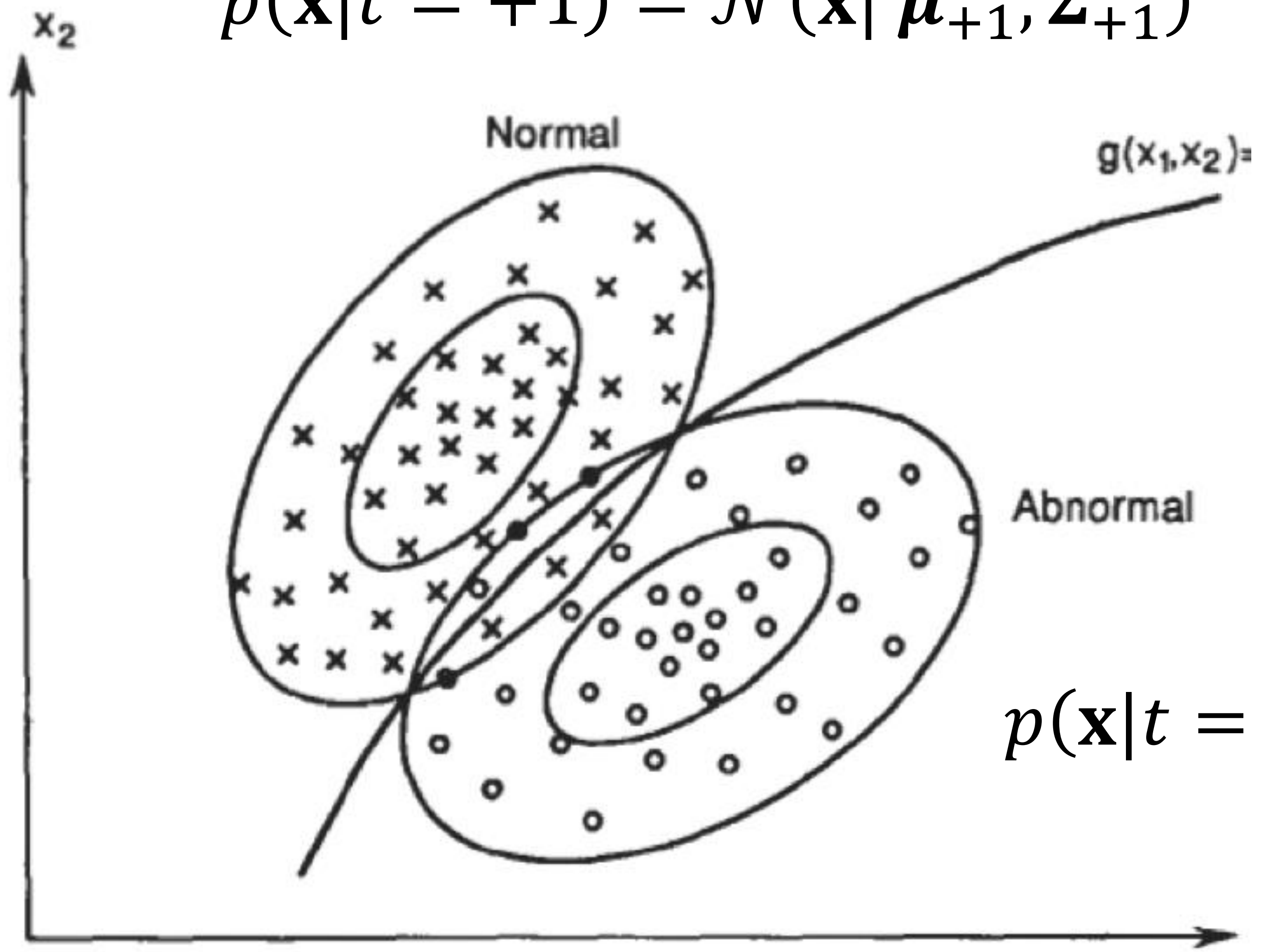




How to determine the decision boundary?

- Given class conditional distribution: $p(\mathbf{x}|t = +1)$, $p(\mathbf{x}|t = -1)$ and class prior: $p(t = +1)$ and $p(t = -1)$

$$p(\mathbf{x}|t = +1) = \mathcal{N}(\mathbf{x}|\, \boldsymbol{\mu}_{+1}, \boldsymbol{\Sigma}_{+1})$$



$$p(\mathbf{x}|t = -1) = \mathcal{N}(\mathbf{x}|\, \boldsymbol{\mu}_{-1}, \boldsymbol{\Sigma}_{-1})$$

Bayes decision rule

- Let's refresh our memories on two important rules in probability and statistics:

- Product rule: $p(x, y) = p(x|y)p(y)$
- Sum rule: $p(x) = \sum_y p(x, y)$
- **Bayes theorem:**

$$p(t|\mathbf{x}) = \frac{p(\mathbf{x}|t)p(t)}{p(\mathbf{x})} = \frac{p(\mathbf{x}, t)}{\sum_t p(\mathbf{x}, t)}$$

- Prior: $p(t)$
- Likelihood (class conditional distribution): $p(\mathbf{x}|t) = \mathcal{N}(x|\mu_t, \Sigma_t)$
- Posterior: $p(t|x) = \frac{p(t)\mathcal{N}(x|\mu_t,\Sigma_t)}{\sum_k p(k)\mathcal{N}(x|\mu_k,\Sigma_k)}$

Bayes decision rule

- For a binary classification problem:

$$p(t=+1|\mathbf{x}) = \frac{p(\mathbf{x}|t=+1)p(t=+1)}{p(\mathbf{x}|t=+1)p(t=+1) + p(\mathbf{x}|t=-1)p(t=-1)}$$

Bayes decision rule

- **Learning:** prior $p(t)$, class conditional distribution $p(\mathbf{x}|t)$
- **Inference:** calculating the posterior probability of a test point

$$p(t = i|\mathbf{x}) = \frac{p(\mathbf{x}|t = i)p(t = i)}{p(\mathbf{x})}$$

- Bayes decision rule:
  - If $p(t = i|\mathbf{x}) > p(t = j|\mathbf{x})$, then $t = i$, otherwise $t = j$
  - Alternatively, if the likelihood ratio:

$$\frac{p(\mathbf{x}|t = i)}{p(\mathbf{x}|t = j)} > \frac{p(t = i)}{p(t = j)}$$

Then $t = i$, otherwise, $t = j$

Generative model: Naïve Bayes

- Use Bayes decision rule for classification

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

- Joint probability model:

$$p(\mathbf{x}|t) = p(x_1, x_2, \dots, x_D|t) = p(x_1|x_2, \dots, x_D, t)p(x_2|x_3, \dots, x_D, t) \dots p(x_{D-1}|x_D, t)p(x_D|t)$$

- But assume $p(\mathbf{x}|t)$ is fully factorized:

$$p(\mathbf{x}|t) = p(x_1, x_2, \dots, x_D|t) = p(x_1|t)p(x_2|t) \dots p(x_D|t)$$

$$p(\mathbf{x}|t) = \prod_{d=1}^{D} p(x_d|t)$$

- Or the variables corresponding to each dimensions of the data are independent given the label

Gaussian Naïve Bayes

- Use Bayes decision rule for classification

$$p(t = 1|\mathbf{x}) = \frac{p(\mathbf{x}|t)p(t)}{p(\mathbf{x})} = \frac{\pi_1 \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)}{\sum_k \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

Because of the independence assumption

$$p(t = 1|\mathbf{x}) = \frac{\pi_1 \prod_{d=1}^{D} \mathcal{N}\left(x_d|\mu_{1d}, \sigma_{1d}^2\right)}{\sum_k \pi_k \prod_{d=1}^{D} \mathcal{N}(x_d|\mu_{kd}, \sigma_{kd}^2)}$$

$$p(t = 1|\mathbf{x}) = \frac{\pi_1 \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_{1d}} \exp\left\{-\frac{1}{2\sigma_{1d}^2}(x_d - \mu_{1d})^2\right\}}{\sum_k \pi_k \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_{kd}} \exp\left\{-\frac{1}{2\sigma_{kd}^2}(x_d - \mu_{kd})^2\right\}}$$

Gaussian Naïve Bayes

$$p(t=1|\mathbf{x}) = \frac{\pi_1 \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_{1d}} \exp\left\{-\frac{1}{2\sigma_{1d}^2}(x_d - \mu_{1d})^2\right\}}{\sum_k \pi_k \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_{kd}} \exp\left\{-\frac{1}{2\sigma_{kd}^2}(x_d - \mu_{kd})^2\right\}}$$

get $\exp(\ln(u))$ of numerator and denominator

$$p(t=1|\mathbf{x}) = \frac{\exp\left\{-\sum_{d=1}^{D}\left(\frac{1}{2\sigma_{1d}^2}(x_d - \mu_{1d})^2 + \ln\sigma_{1d} + C\right) + \ln\pi_1\right\}}{\sum_k \exp\left\{-\sum_{d=1}^{D}\left(\frac{1}{2\sigma_{kd}^2}(x_d - \mu_{kd})^2 + \ln\sigma_{kd} + C\right) + \ln\pi_k\right\}}$$

Gaussian Naïve Bayes

$$p(t=1|\mathbf{x}) = \frac{1}{1+\exp\left\{-\underbrace{\sum_{d=1}^{D}\left(x_d\frac{1}{\sigma_d}(\mu_{1d}-\mu_{2d})\right.}_{\sum_{d=1}^{D} w_d x_d}\underbrace{\left.+\frac{1}{\sigma_d^2}(\mu_{1d}^2-\mu_{2d}^2)\right)+\ln\frac{\pi_2}{\pi_1}}_{w_0}\right\}}$$

Gaussian Naïve Bayes

$$p(t=1|\mathbf{x}) = \frac{1}{1+\exp\left\{-\sum_{d=1}^{D}\left(x_d\frac{1}{\sigma_d}(\mu_{1d}-\mu_{2d})+\frac{1}{\sigma_d^2}(\mu_{1d}^2-\mu_{2d}^2)\right)+\ln\frac{\pi_2}{\pi_1}\right\}}$$

- Number of parameters: $2D+1$ (D mean, D variance, and 1 for prior)

$$p(t=1|\mathbf{x}) = \frac{1}{1+\exp\{-(w_0+\sum_{d=1}^{D}w_d x_d)\}}$$

- Number of parameters = $D+1 \rightarrow w_0, w_1, w_2, \dots, w_D$

- Why not directly learning $\mathrm{p}(t=1|\mathbf{x})$ or $\mathbf{w}$ parameters?

Gaussian Naïve Bayes is a subset of logistic regression

Outline

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

Classification approaches

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

- Discriminative models
  - Directly estimate the posterior probabilities
  - No need to model underlying prior and likelihood distributions
  - Examples: Logistic regression, SVM, neural networks

Discriminative Models

- Directly estimate decision boundary $h(x) = -\ln \frac{q_i(\mathbf{x})}{q_j(\mathbf{x})}$ or posterior distribution $p(t|x)$

- Logistic regression, neural networks
  - Do not estimate $p(x|t)$ and $p(t)$

- Why discriminative classifier?
  - Avoid difficult density estimation problem coming from generative models
  - Empirically achieve better classification results

Logistic function for posterior probability

- Let's use the following function:

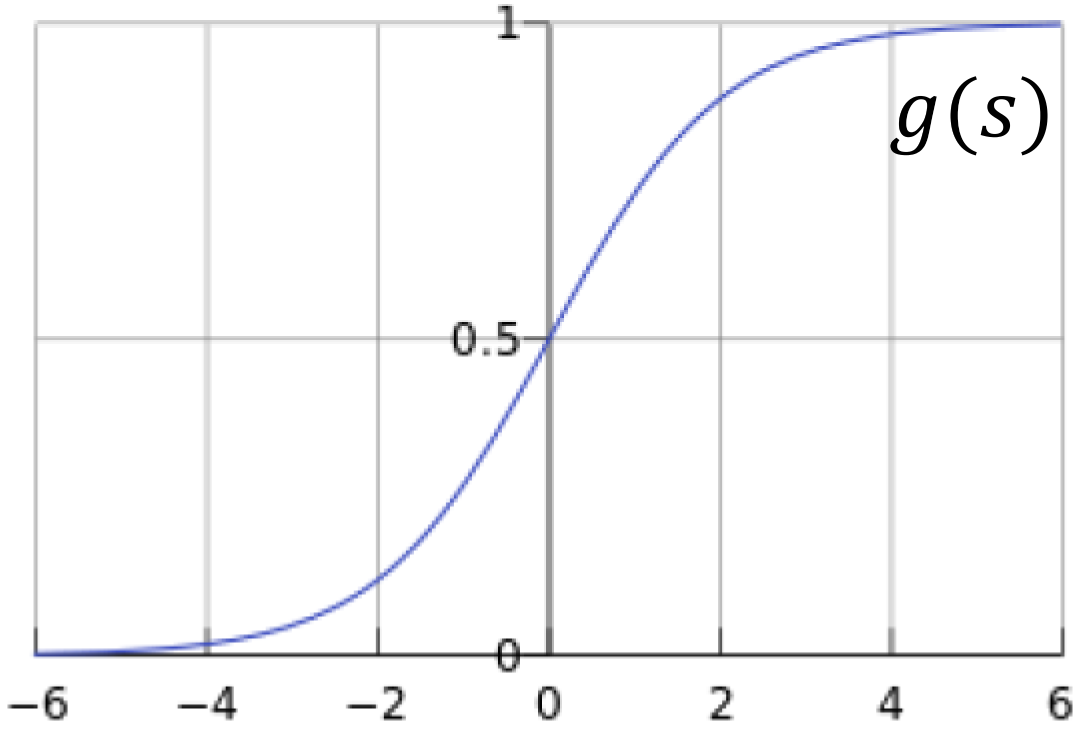
$$s = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$$

$$g(s) = \frac{e^s}{1 + e^s} = \frac{1}{1 + e^{-s}}$$

- This formula is called a sigmoid function

- It is easier to use this function for optimization

Many equations can give us this shape

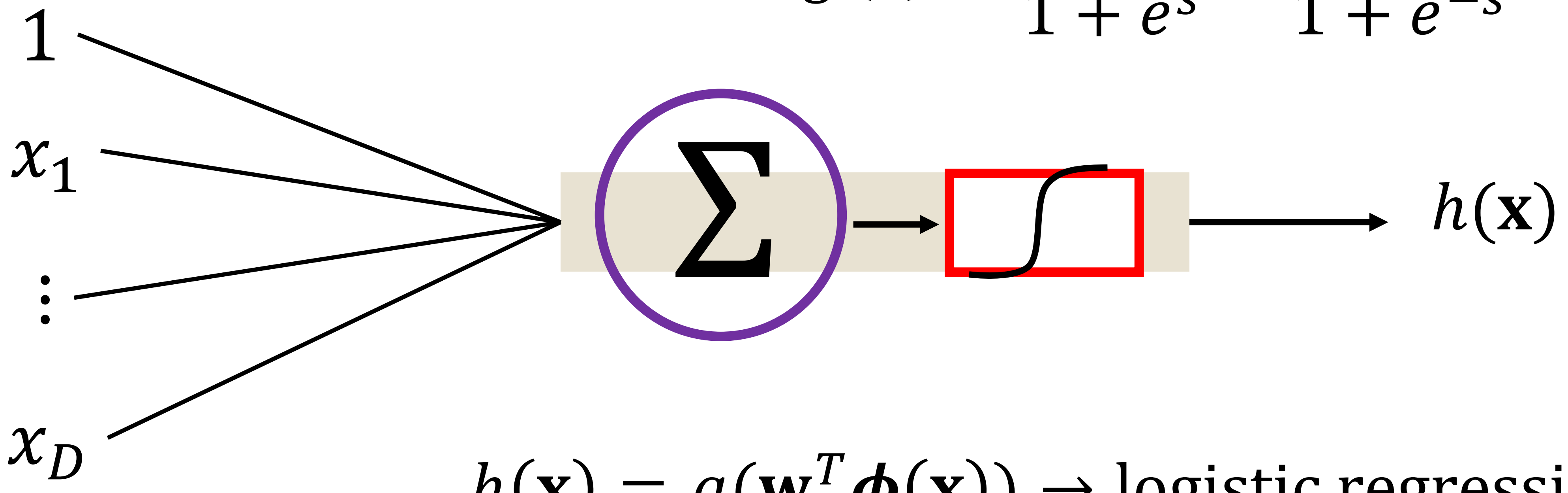


Sigmoid Function

- We enforce $\phi_0(\mathbf{x}) = 1$, so for a simple mapping function of a vector $\mathbf{x}$ with D dimensions, we have the following: obtain:

$$s = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) = \sum_{m=0}^{M-1} w_m \phi_m(\mathbf{x}) = w_0 + w_1 x_1 + \cdots + w_D x_D$$

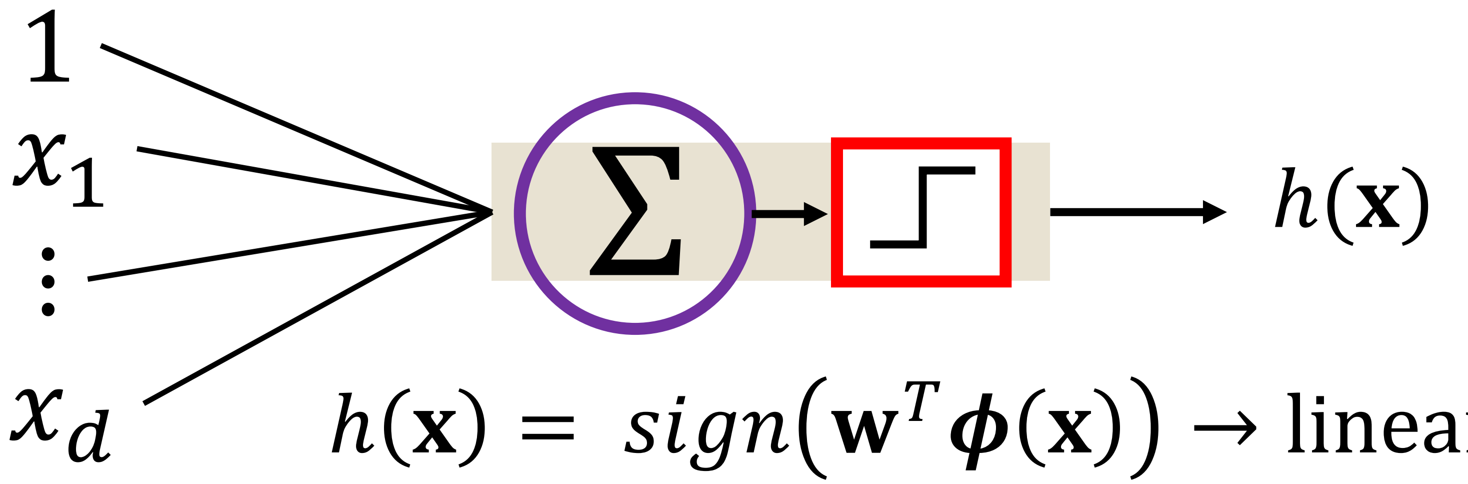
$$g(s) = \frac{e^s}{1 + e^s} = \frac{1}{1 + e^{-s}}$$



Soft classification
Posterior probability

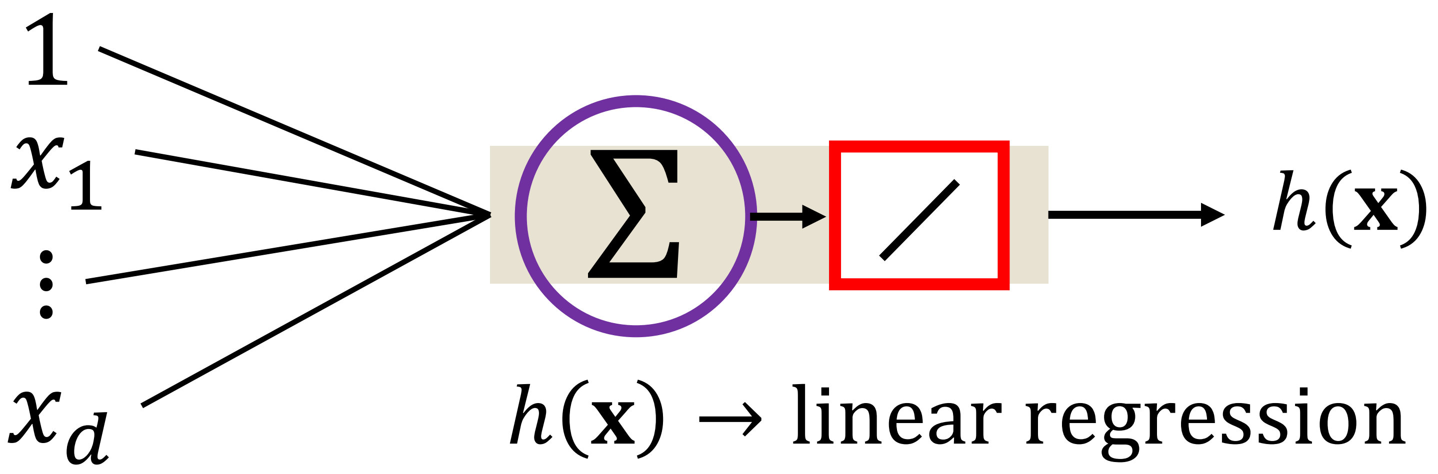
$$h(\mathbf{x}) = g(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})) \rightarrow \text{logistic regression}$$

Three linear models

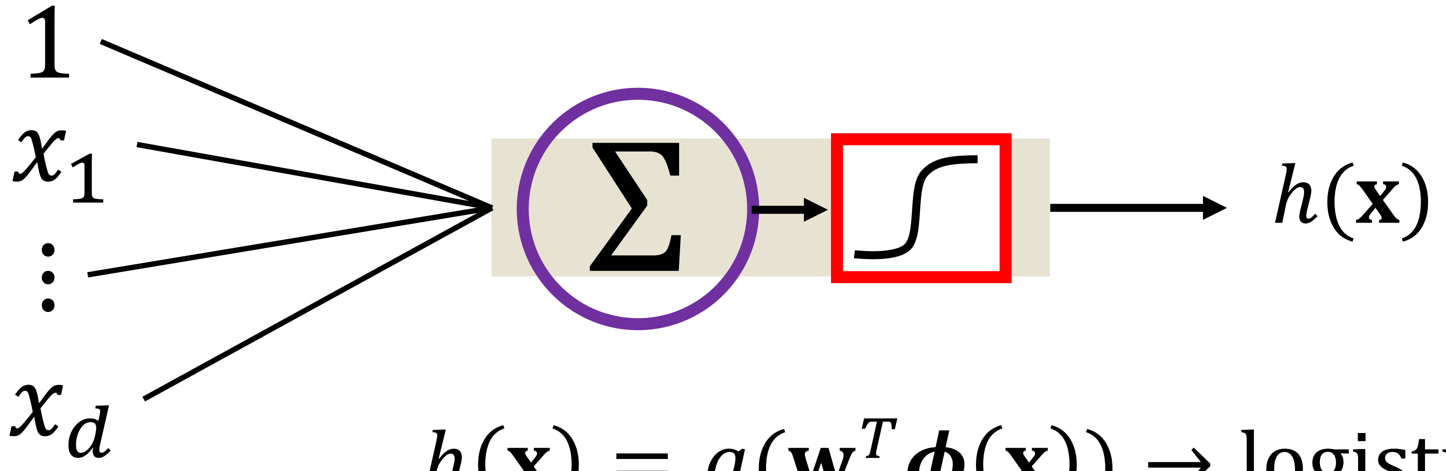


Hard classification

$h(\mathbf{x}) = sign\left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right) \rightarrow$ linear classification (perceptron)



$h(\mathbf{x}) \rightarrow$ linear regression



Soft classification
Posterior probability

$h(\mathbf{x}) = g(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})) \rightarrow$ logistic regression

Outline

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

Logistic function for posterior probability

- $g(s)$ is interpreted as probability

- **Example:** Prediction of heart attacks
  - Input $\mathbf{x}$: cholesterol level, age, weight, finger size, etc.
  - $g(s)$: probability of heart attack within a certain time
  - Let's call this risk score $s = \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$
  - We can't have a hard prediction here
  - $h(x) = p(t|x) = \begin{cases} g(s), & t = 1 \\ 1 - g(s), & t = 0 \end{cases}$
  - Using posterior probability directly

Logistic regression model

- $p(t|x) = \begin{cases} \frac{1}{1+\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right)} & t = 1 \\ 1 - \frac{1}{1+\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right)} = \frac{\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right)}{1+\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right)} & t = 0 \end{cases}$

- We need to find $\mathbf{w}$ parameters, let's set up log-likelihood for N datapoints

$$ll(\mathbf{w}) = \log \prod_{n=1}^{N} p(t_n, |\mathbf{x}_n, \mathbf{w})$$

$$ll(\mathbf{w}) = \sum_{n=1}^{N} (\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n))(t_n - 1) - \log\left(1 + \exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)\right)$$

- <u>This form is concave, negative of this form is convex</u>

The gradient of $ll(\mathbf{w})$

$$ll(\mathbf{w}) = \log \prod_{n=1}^{N} p(t_n, | \mathbf{x}_n, \mathbf{w})$$

$$ll(\mathbf{w}) = \sum_{n=1}^{N} (\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n))(t_n - 1) - \log\left(1 + \exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)\right)$$

- Gradient

$$\frac{\partial ll(\mathbf{w})}{\partial \mathbf{w}} = \sum_{n=1}^{N} \left\{ \boldsymbol{\phi}(\mathbf{x}_n)(t_n - 1) + \frac{\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)}{1 + \exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)} \boldsymbol{\phi}(\mathbf{x}_n) \right\}$$

- <u>Setting it to 0 does not lead to closed form solution</u>

The objective function

- Find $\mathbf{w}$, such that the conditional likelihood of the labels is maximized

$$\max_{\mathbf{w}} ll(\mathbf{w}) = \log \prod_{n=1}^{N} p(t_n, |\mathbf{x}_n, \mathbf{w})$$

- **Good news:** $ll(\mathbf{w})$ is a concave function of $\mathbf{w}$, and there is a single global optimum

- **Bad news:** no closed form solution (resort to numerical method)

Outline

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

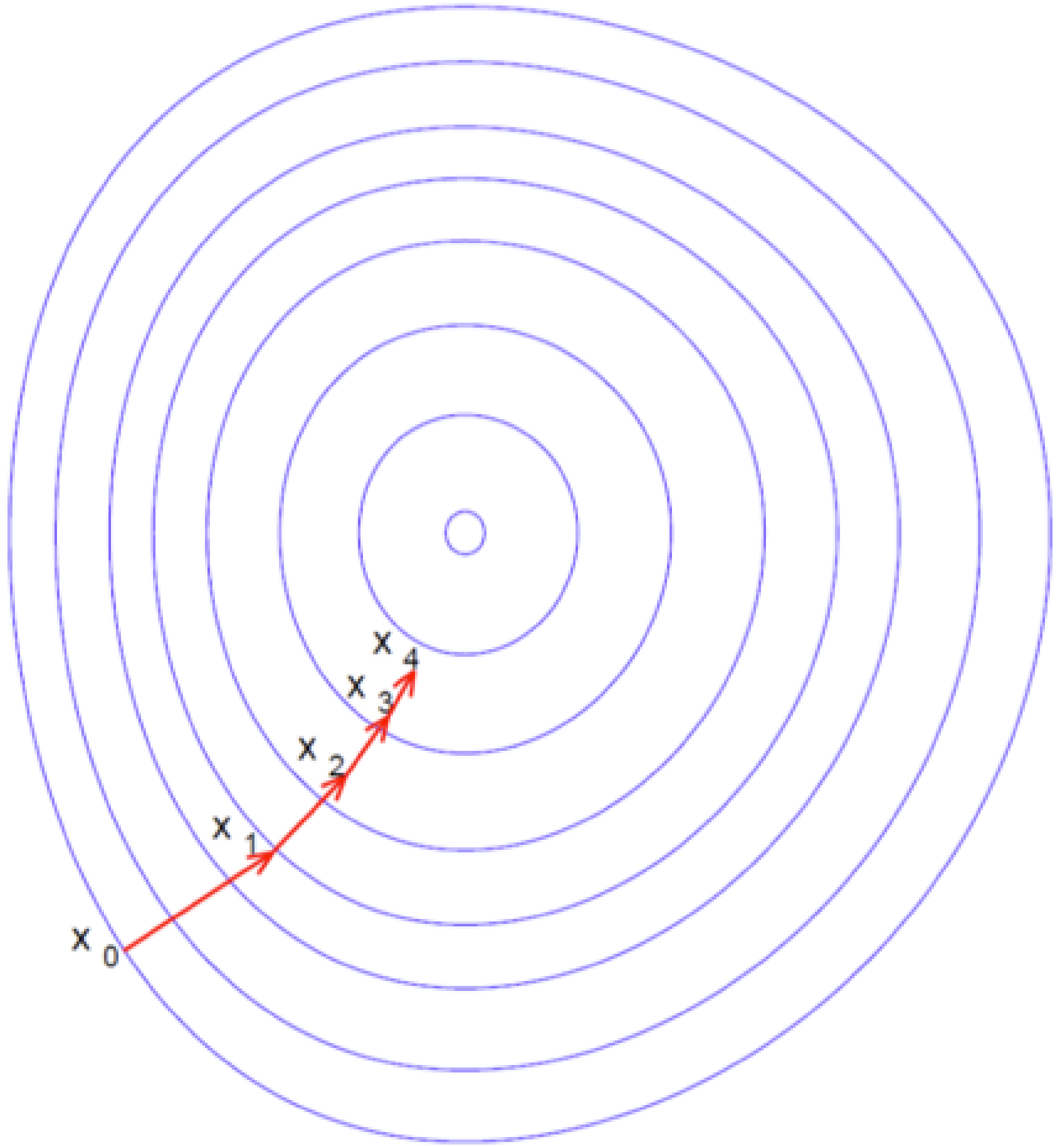
Gradient descent: intuition

Gradient descent

- One way to solve an unconstrained optimization problem is gradient descent
- Given an initial guess, we are iteratively refining 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

$$\mathbf{z}_{(\tau+1)} = \mathbf{z}_{\tau} - \gamma_{\tau} \nabla f(\mathbf{z}_{\tau})$$

γ_{τ} is the learning rate



Gradient ascent (concave) / descent (convex)

- Initialize parameter $\mathbf{w}_{\tau=0}$
- Do

$$\mathbf{w}_{(\tau+1)} = \mathbf{w}_{\tau} - \eta_{\tau} \frac{\partial ll(\mathbf{w})}{\partial \mathbf{w}} = \mathbf{w}_{\tau} - \eta_{\tau} \sum_{n=1}^{N} \left\{ \boldsymbol{\phi}(\mathbf{x}_n)(t_n - 1) + \frac{\exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)}{1 + \exp\left(-\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_n)\right)} \boldsymbol{\phi}(\mathbf{x}_n) \right\}$$

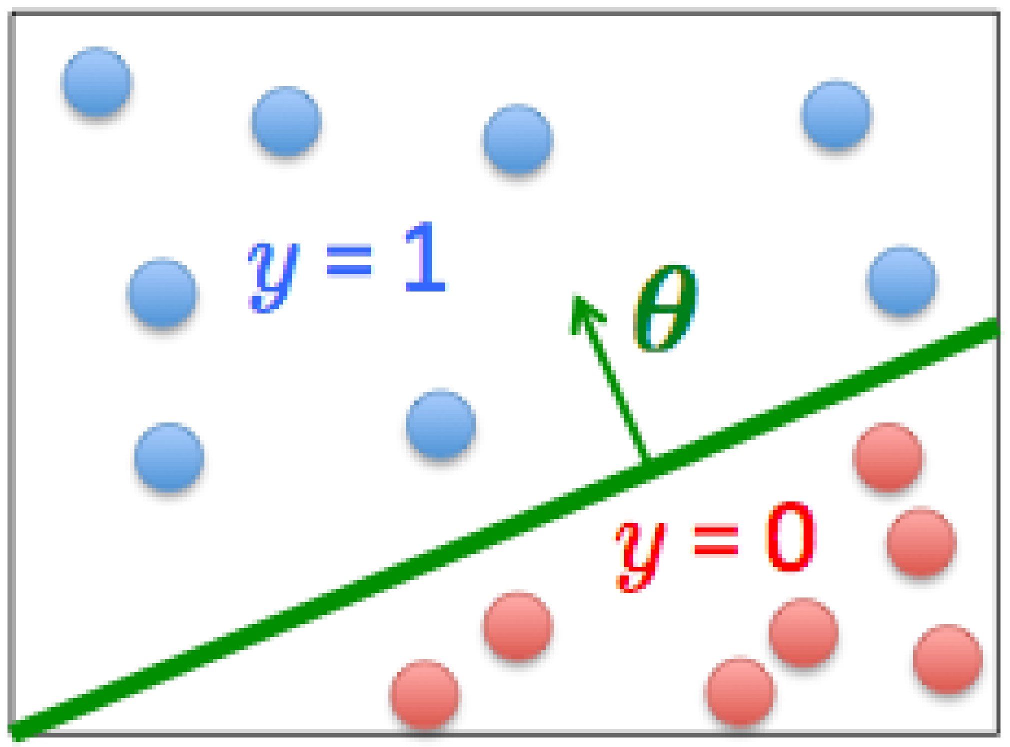
- While the $\left\| \mathbf{w}_{(\tau+1)} - \mathbf{w}_{\tau} \right\| > \epsilon$

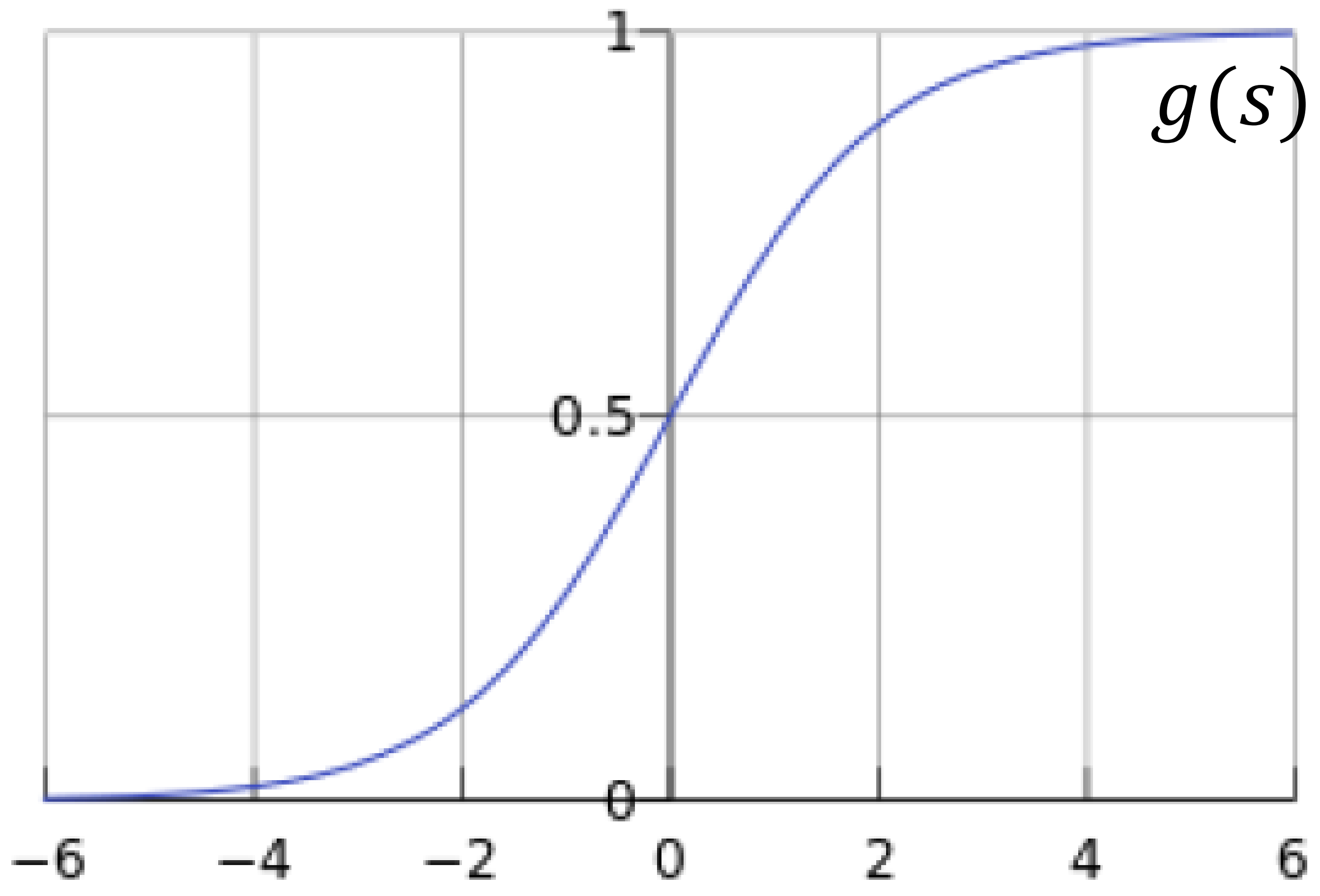
Logistic regression

$$h_{\mathbf{w}}(x) = g\left(\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})\right)$$

$$g(s) = \frac{1}{1 + e^{-s}}$$

- Assume a threshold and...
  - Predict $t = 1$ if $h_w(\mathbf{x}) \geq 0.5$
  - Predict $t = 0$ if $h_w(\mathbf{x}) \geq 0.5$





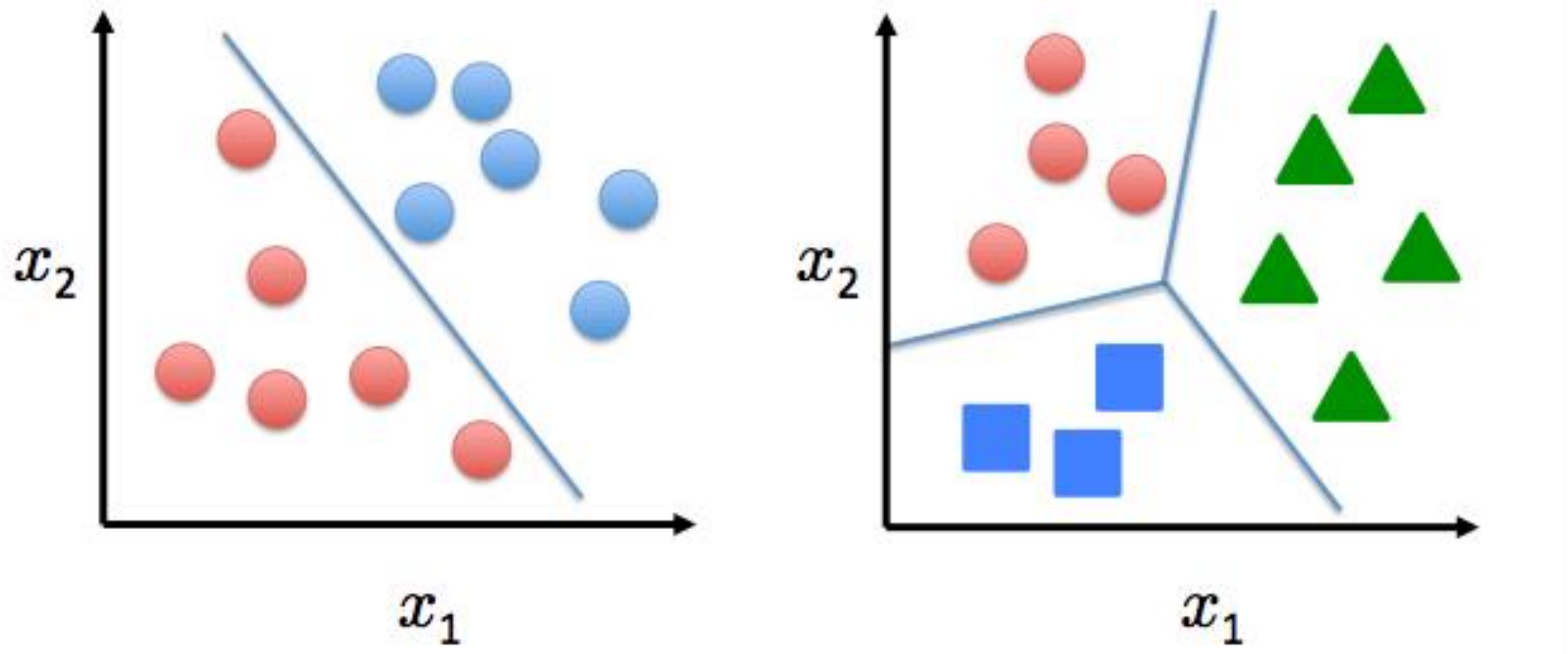
$\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$ should be large negative values for negative instances	$\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x})$ should be large positive values for positive instances

Outline

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

Multiclass logistic regression

Binary classification | Multi-class classification

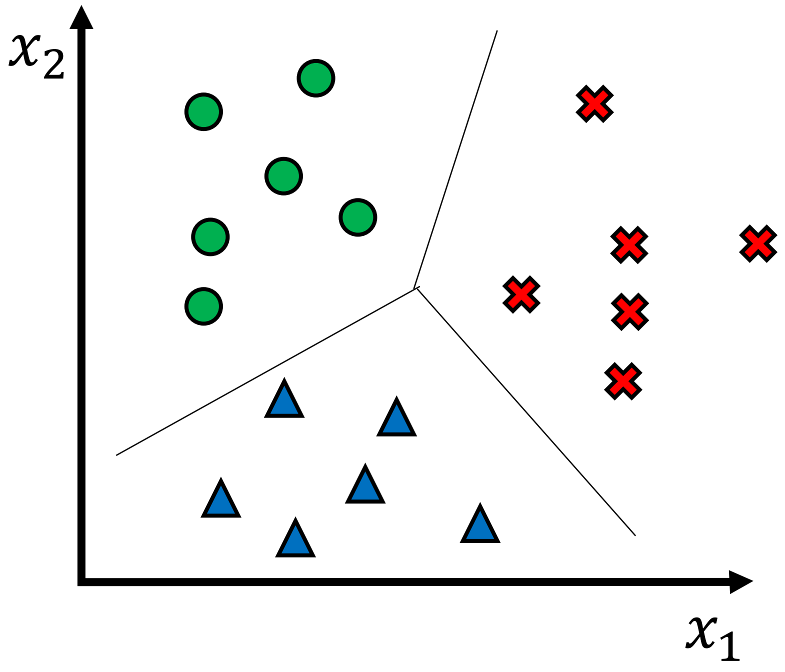


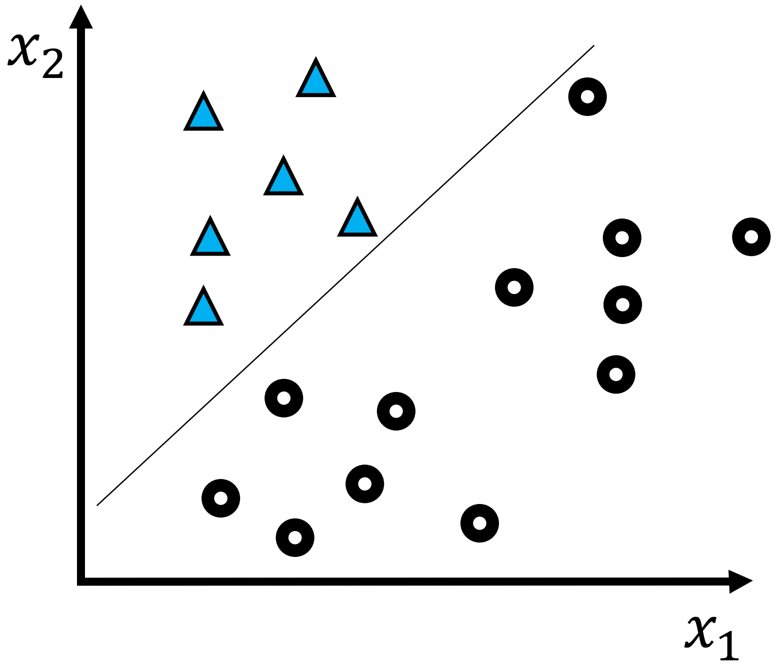
- Disease diagnosis: healthy / cold / flu / pneumonia
- Object classification: desk / chair/ monitor / bookcase

One-vs-all (one-vs-rest)

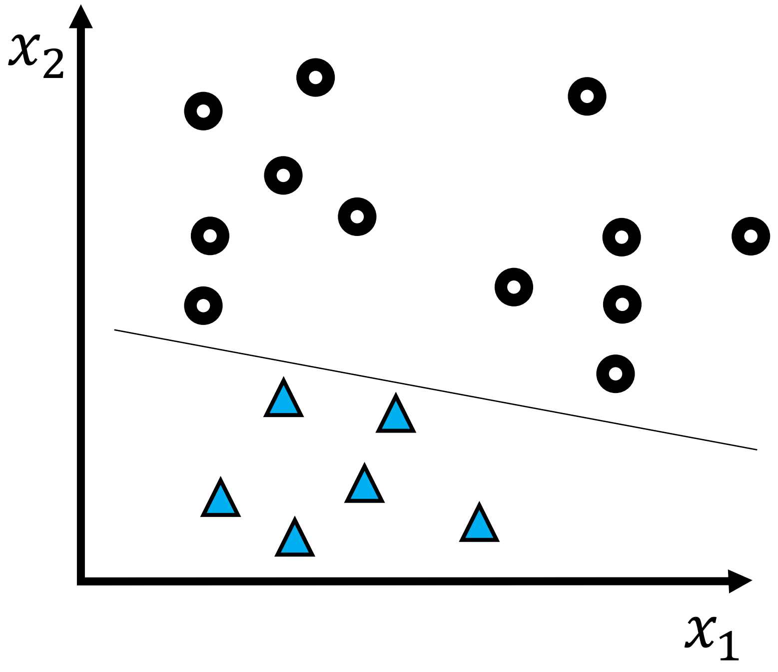
- Train a logistic regression $h_{\mathbf{w}}^{(k)}(\mathbf{x})$ for each class k
- To predict the label of a new input x, pick class i that maximizes:

$$\max_k h_{\mathbf{w}}^{(k)}(\mathbf{x})$$

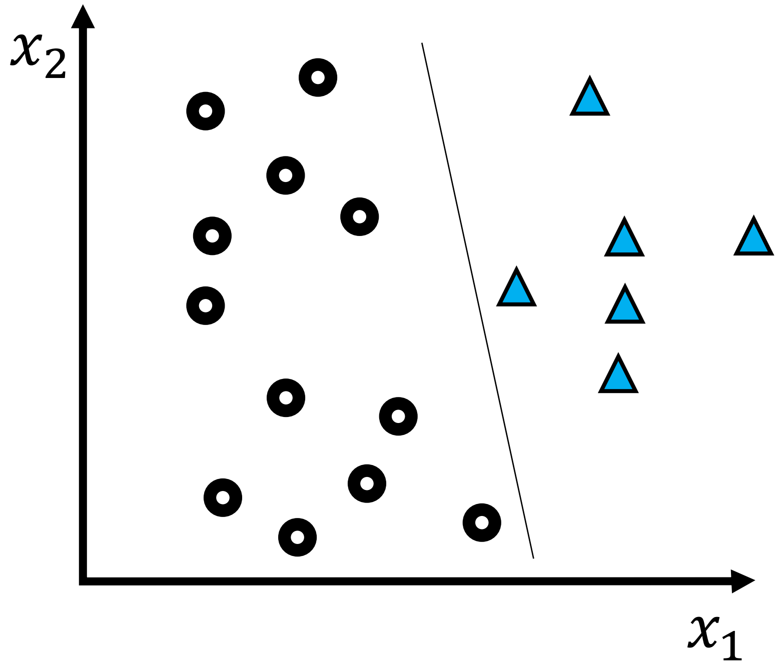




$h_{\mathbf{w}}^{1}(\mathbf{x})$



$h_{\mathbf{w}}^{2}(\mathbf{x})$



$h_{\mathbf{w}}^{3}(\mathbf{x})$