Machine Learning

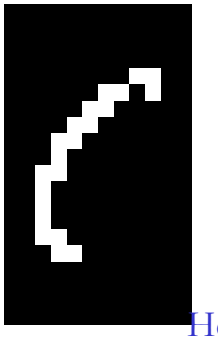
CSE 6363 (Fall 2016)

Lecture 6 Linear and Quadratic Discriminant Analysis

Heng Huang, Ph.D.

Department of Computer Science and Engineering

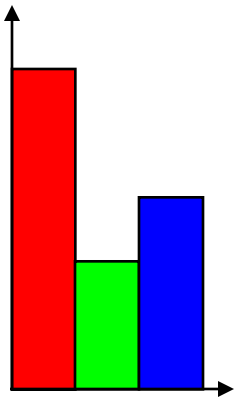
Basic Classification in ML

	Input $\mathbf{x} \in \mathcal{X}$	Output $y \in \mathcal{Y}$
Spam filtering		**Binary** 
Character recognition		**Multi-Class** **C**

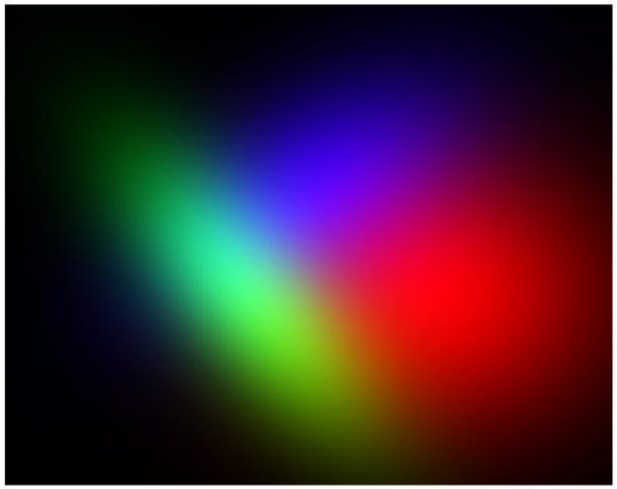
Generative Models

As in the binary case:

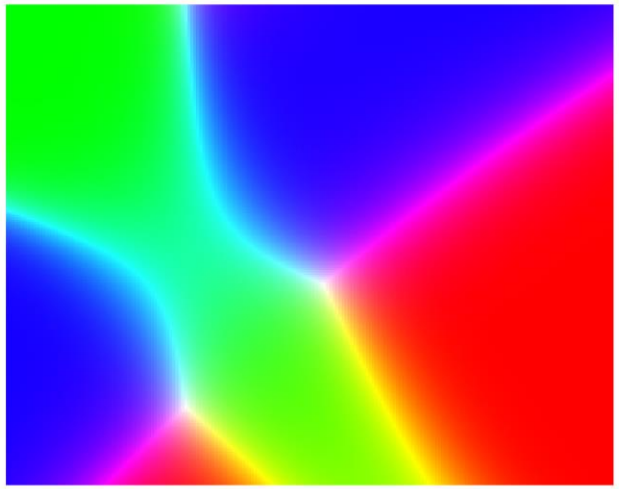
1. Learn p(y) and p(y|x)
2. Use Bayes rule: $p(y=k|x) = \frac{p(x|y=k)p(y=k)}{p(x)}$
3. Classify as $\hat{y}(x) = \text{argmax}_y \, p(y|x)$



p(y)



p(x|y)



p(y|x)

Linear Regression of An Indicator Matrix

If $\mathcal{G}$ has K classes, there will be K class indicators Y_k, $k = 1, ..., K$.

g	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_4$
3	0	0	1	0
1	1	0	0	0
2	0	1	0	0
4	0	0	0	1
1	1	0	0	0

Fit a linear regression model for each Y_k, $k = 1, 2, ..., K$, using X:

$$\hat{\mathbf{y}}_k = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}_k \ .$$

Define $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_k)$:

$$\hat{\mathbf{Y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \ .$$

Classification Procedure

Define $\hat{\mathbf{B}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$.

For a new observation with input x, compute the fitted output

$$\begin{aligned}\hat{f}(x) &= [(1,x)\hat{B}]^T \\ &= [(1,x_1,x_2,...,x_p)\hat{B}]^T \\ &= \begin{pmatrix} \hat{f}_1(x) \\ \hat{f}_2(x) \\ ... \\ \hat{f}_K(x) \end{pmatrix}\end{aligned}$$

Identify the largest component of $\hat{f}(x)$ and classify accordingly:

$$\hat{G}(x) = \arg\max_{k\in\mathcal{G}} \hat{f}_k(x) \ .$$

Classification - Summary

- **Want to Learn**: h:**X** $\mapsto$ Y
  - **X** – features
  - Y – target classes

- **Generative classifier**, e.g., Naïve Bayes:
  - Assume some **functional form for P(X|Y), P(Y)**
  - Estimate parameters of P(X|Y), P(Y) directly from training data
  - Use Bayes rule to calculate P(Y|X= x)
  - This is a '***generative***' **model**
    - **Indirect** computation of P(Y|X) through Bayes rule
    - But, **can generate a sample of the data**, P(X) = $\sum_y$ P(y) P(X|y)
- **Discriminative classifiers**, e.g., Logistic Regression:
  - Assume some **functional form for P(Y|X)**
  - Estimate parameters of P(Y|X) directly from training data
  - This is the '***discriminative***' **model**
    - Directly learn P(Y|X)
    - But **cannot obtain a sample of the data**, because P(X) is not available

Ref: Carlos Guestrin

Linear Discriminant Analysis

Essentially minimum error Bayes' classifier

Assumes that the conditional class densities are (multivariate) Gaussian

Assumes equal covariance for every class

Posterior probability $\Pr(G = k \mid X = x) = \frac{f_k(x)\pi_k}{\sum_{l=1}^{K} f_l(x)\pi_l}$ ← Application of Bayes rule

π_k is the prior probability for class k

$f_k(x)$ is class conditional density or likelihood density

$$f_k(x) = \frac{1}{(2\pi)^{p/2} |\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)\right)$$

Linear Discriminant Analysis

$$
\begin{aligned}
\hat{G}(x) &= \arg\max_k Pr(G = k \mid X = x) \\
&= \arg\max_k f_k(x)\pi_k \\
&= \arg\max_k \log(f_k(x)\pi_k) \\
&= \arg\max_k \left[-\log((2\pi)^{p/2}|\Sigma|^{1/2}) \right. \\
&\quad \left. -\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) + \log(\pi_k) \right] \\
&= \arg\max_k \left[-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) + \log(\pi_k) \right] \\
&\quad -\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k) \\
&= x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k - \frac{1}{2}x^T\Sigma^{-1}x
\end{aligned}
$$

Linear Discriminant Analysis

- To sum up

$$\hat{G}(x) = \arg\max_k \left[x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log(\pi_k) \right]$$

- Define the **linear discriminant function**

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log(\pi_k)$$

Then

$$\hat{G}(x) = \arg\max_k \delta_k(x)$$

- The decision boundary between class *k* and *l* is:

$$\{x : \delta_k(x) = \delta_l(x)\}$$

- Or equivalently the following holds

$$\log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1} (\mu_k - \mu_l) + x^T \Sigma^{-1} (\mu_k - \mu_l) = 0$$

Linear Discriminant Analysis

Consider the classification through decision boundary

$$\log \frac{\Pr(G=k \mid X=x)}{\Pr(G=l \mid X=x)} = \log \frac{\pi_k}{\pi_l} + \log \frac{f_k}{f_l}$$

$$= \underbrace{(\log \pi_k + x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k)}_{\delta_k(x)} - \underbrace{(\log \pi_l + x^T \Sigma^{-1} \mu_l - \frac{1}{2} \mu_l^T \Sigma^{-1} \mu_l)}_{\delta_l(x)}$$

Classification rule: $\hat{G}(x) = \arg\max_k \delta_k(x)$

is equivalent to: $\hat{G}(x) = \arg\max_k \Pr(G=k \mid X=x)$

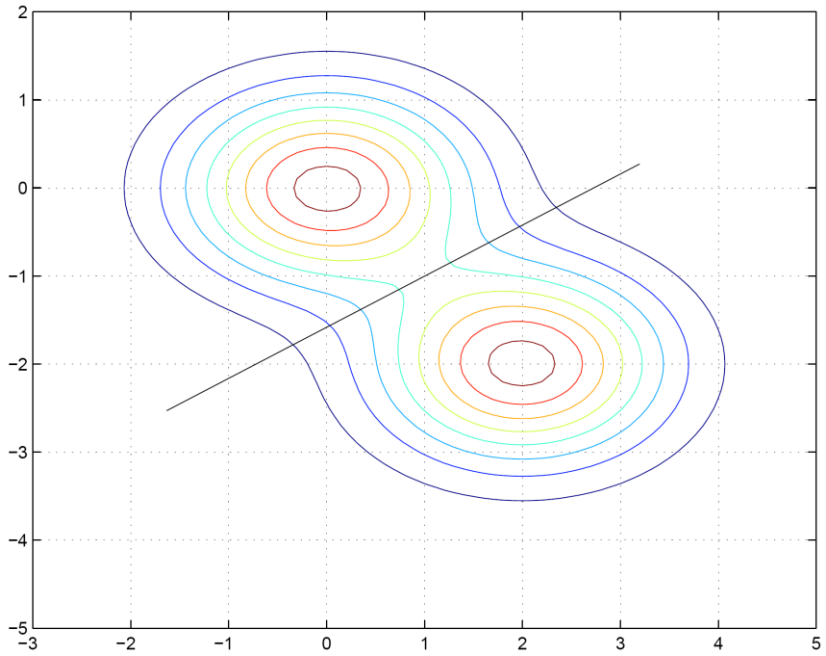
The good old Bayes classifier!

Binary Classification Example

Binary classification ($k = 1, l = 2$):

- Define $a_0 = \log \frac{\pi_1}{\pi_2} - \frac{1}{2}(\mu_1 + \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2)$.
- Define $(a_1, a_2, ..., a_p)^T = \Sigma^{-1}(\mu_1 - \mu_2)$.
- Classify to class 1 if $a_0 + \sum_{j=1}^{p} a_j x_j > 0$; to class 2 otherwise.
- An example:
  - $\pi_1 = \pi_2 = 0.5$.
  - $\mu_1 = (0, 0)^T$, $\mu_2 = (2, -2)^T$.
  - $\Sigma = \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 0.5625 \end{pmatrix}$.
  - Decision boundary:

$$5.56 - 2.00x_1 + 3.56x_2 = 0.0 \ .$$



Estimate Gaussian Distributions

In practice, we need to estimate the Gaussian distribution.

Total N input-output pairs
N_k number of pairs in class k
Total number of classes: K
Training data utilized to estimate

$$(g_i, x_i),\ i = 1:N$$

Prior probabilities: $\hat{\pi}_k = N_k / N$

Means: $\hat{\mu}_k = \sum_{g_i = k} x_i / N_k$

Covariance matrix: $\hat{\Sigma} = \sum_{k=1}^{K} \sum_{g_i} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T / (N - K)$

Diabetes Data Example

- **Diabetes data**

  The diabetes data set is taken from the UCI machine learning database repository at:
  http://www.ics.uci.edu/~mlearn/Machine-Learning.html .

  The original source of the data is the National Institute of Diabetes and Digestive and Kidney Diseases. There are 768 cases in the data set, of which 268 show signs of diabetes according to World Health organization criteria. Each case contains 8 quantitative variables, including diastolic blood pressure, triceps skin fold thickness, a body mass index, etc.

  – Two classes: with or without signs of diabetes.
  – Denote the 8 original variables by $\tilde{X}_1, \tilde{X}_2, ..., \tilde{X}_8$
  – Remove the mean of $\tilde{X}_j$ and normalize it to unit variance.

Diabetes Data Set

Two input variables computed from the principal components of the original 8 variables.

$$X_1 = 0.1284\tilde{X}_1 + 0.3931\tilde{X}_2 + 0.3600\tilde{X}_3 + 0.4398\tilde{X}_4 + 0.4350\tilde{X}_5 + 0.4519\tilde{X}_6 + 0.2706\tilde{X}_7 + 0.1980\tilde{X}_8$$

$$X_2 = 0.5938\tilde{X}_1 + 0.1740\tilde{X}_2 + 0.1839\tilde{X}_3 - 0.3320\tilde{X}_4 - 0.2508\tilde{X}_5 - 0.1010\tilde{X}_6 - 0.1221\tilde{X}_7 + 0.6206\tilde{X}_8$$

Prior probabilities: $\hat{\pi}_1 = 0.651$, $\hat{\pi}_2 = 0.349$.

$\hat{\mu}_1 = (-0.4035, -0.1935)^T$, $\hat{\mu}_2 = (0.7528, 0.3611)^T$.

$$\hat{\Sigma} = \begin{pmatrix} 1.7925 & -0.1461 \\ -0.1461 & 1.6634 \end{pmatrix}$$

Classification rule:

$$\hat{G}(x) = \begin{cases} 1 & 0.7748 - 0.6771x_1 - 0.3929x_2 \geq 0 \\ 2 & \text{otherwise} \end{cases}$$

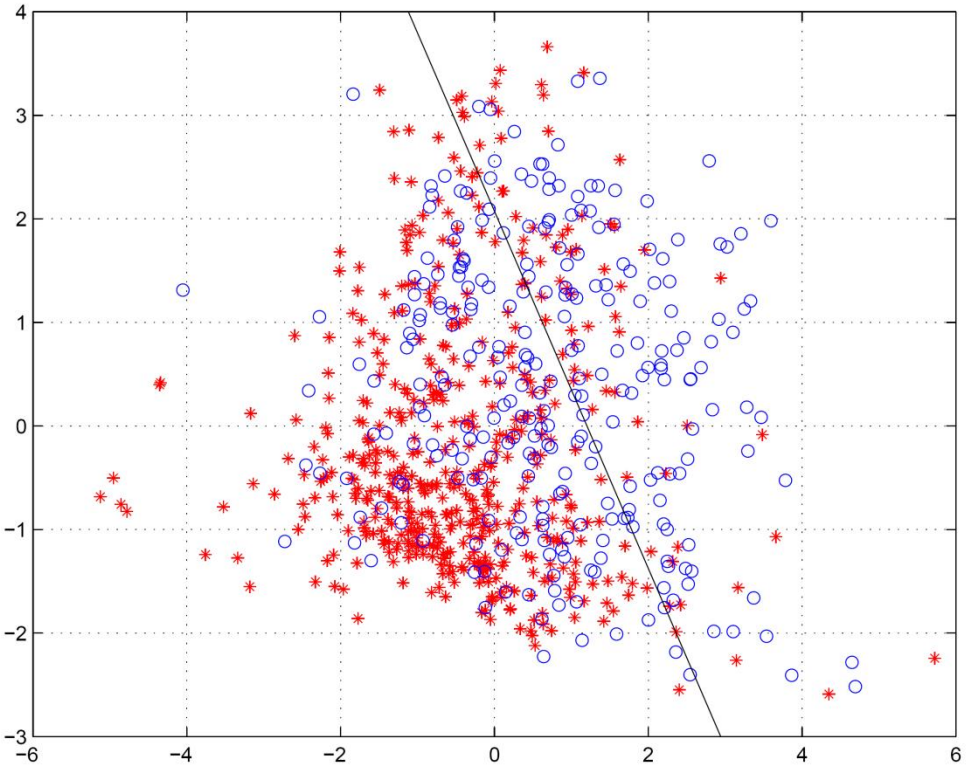
$$= \begin{cases} 1 & 1.1443 - x_1 - 0.5802x_2 \geq 0 \\ 2 & \text{otherwise} \end{cases}$$

Result of Linear Regression Based Classification

- Classification error rate: 28.52%.

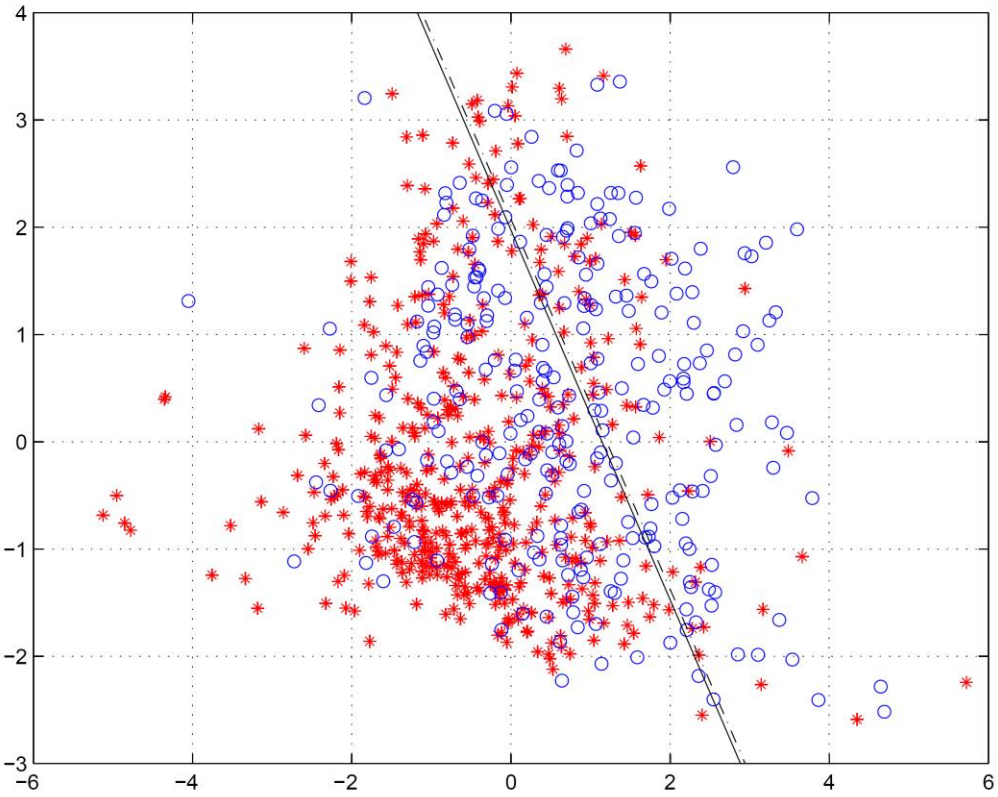
$$\hat{G}(x) = \begin{cases} 1 & \hat{Y}_1 \geq \hat{Y}_2 \\ 2 & \hat{Y}_1 < \hat{Y}_2 \end{cases}$$

$$= \begin{cases} 1 & 0.151 - 0.1256X_1 - 0.0729X_2 \geq 0 \\ 2 & otherwise \end{cases}$$



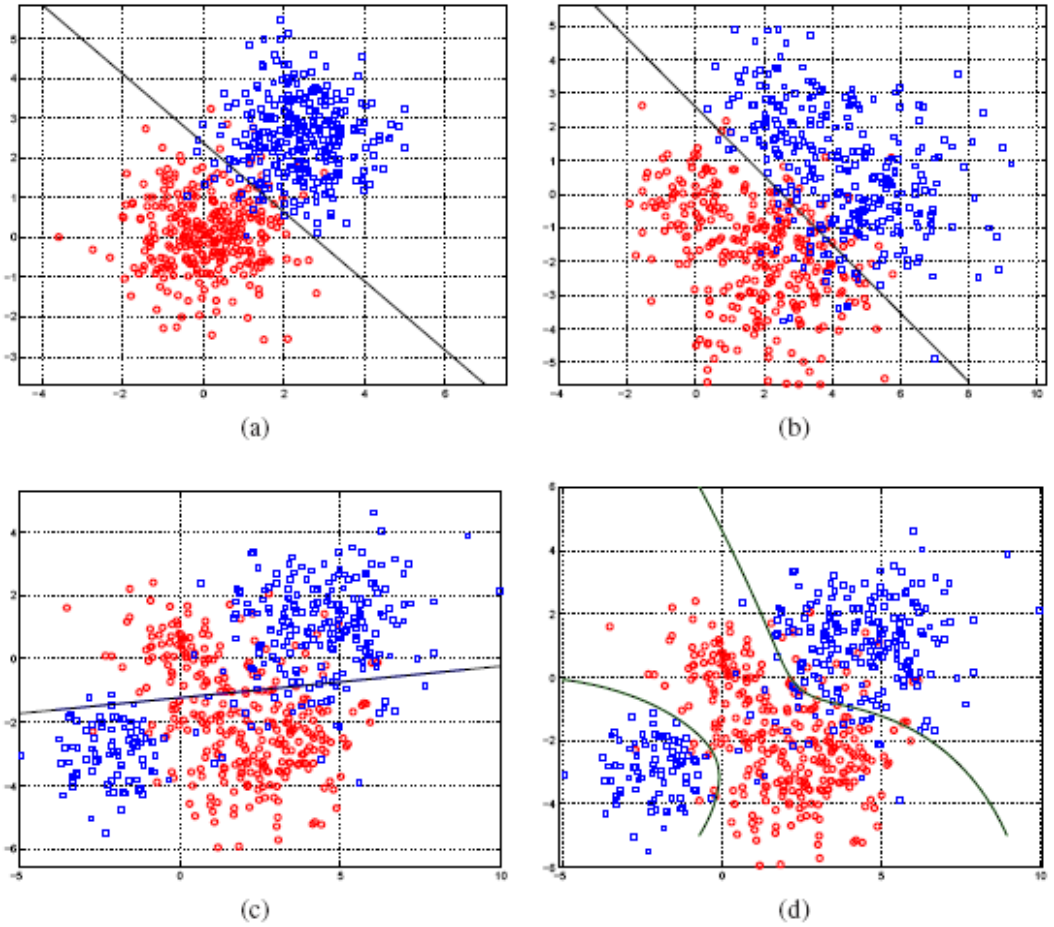
LDA Classification Result

- The scatter plot follows. Without diabetes: stars (class 1), with diabetes: circles (class 2). Solid line: classification boundary obtained by LDA. Dash dot line: boundary obtained by linear regression of indicator matrix.
- Within training data classification error rate: 28.26%.



Simulated Examples

- LDA applied to three simulated data sets in (a)-(c). (a): The true within class densities are Gaussian with identical covariance matrices across classes. (b) and (c): The true within class densities are mixtures of two Gaussians.
- (d): The data set is the same as that in (c). Decision boundaries are obtained by modeling each class by a mixture of two Gaussians.



Quadratic Discriminant Analysis

- Relaxes the same covariance assumption – class conditional probability densities (still multivariate Gaussians) are allowed to have different covariant matrices
- The class decision boundaries are not linear rather quadratic

$$\log \frac{\Pr(G=k \mid X=x)}{\Pr(G=l \mid X=x)} = \log \frac{\pi_k}{\pi_l} + \log \frac{f_k}{f_l} =$$

$$\underbrace{(\log \pi_k - \frac{1}{2}(x-\mu_k)^T \boldsymbol{\Sigma}_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\boldsymbol{\Sigma}_k|)}_{\delta_k(x)} - \underbrace{(\log \pi_l - \frac{1}{2}(x-\mu_l)^T \boldsymbol{\Sigma}_l^{-1}(x-\mu_l) - \frac{1}{2}\log|\boldsymbol{\Sigma}_l|)}_{\delta_l(x)}$$

Classification rule: $\hat{G}(x) = \arg\max_k \delta_k(x)$

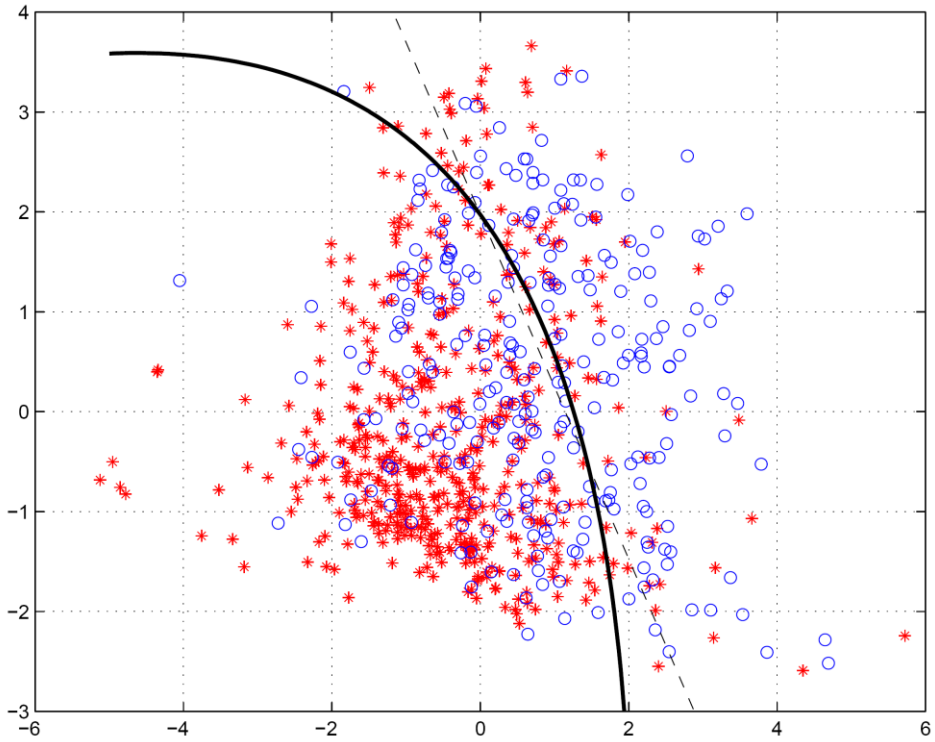
Diabetes Data Set

Prior probabilities: $\hat{\pi}_1 = 0.651$, $\hat{\pi}_2 = 0.349$.

$$\hat{\mu}_1 = (-0.4035, -0.1935)^T, \hat{\mu}_2 = (0.7528, 0.3611)^T$$

$$\hat{\Sigma}_1 = \begin{pmatrix} 1.6769 & -0.0461 \\ -0.0461 & 1.5964 \end{pmatrix}$$

$$\hat{\Sigma}_2 = \begin{pmatrix} 2.0087 & -0.3330 \\ -0.3330 & 1.7887 \end{pmatrix}$$



Within training data
classification error rate: 29.04%.

LDA on Expanded Basis

Expand input space to include X_1X_2, X_1^2, and X_2^2.
Input is five dimensional: $X = (X_1, X_2, X_1X_2, X_1^2, X_2^2)$.

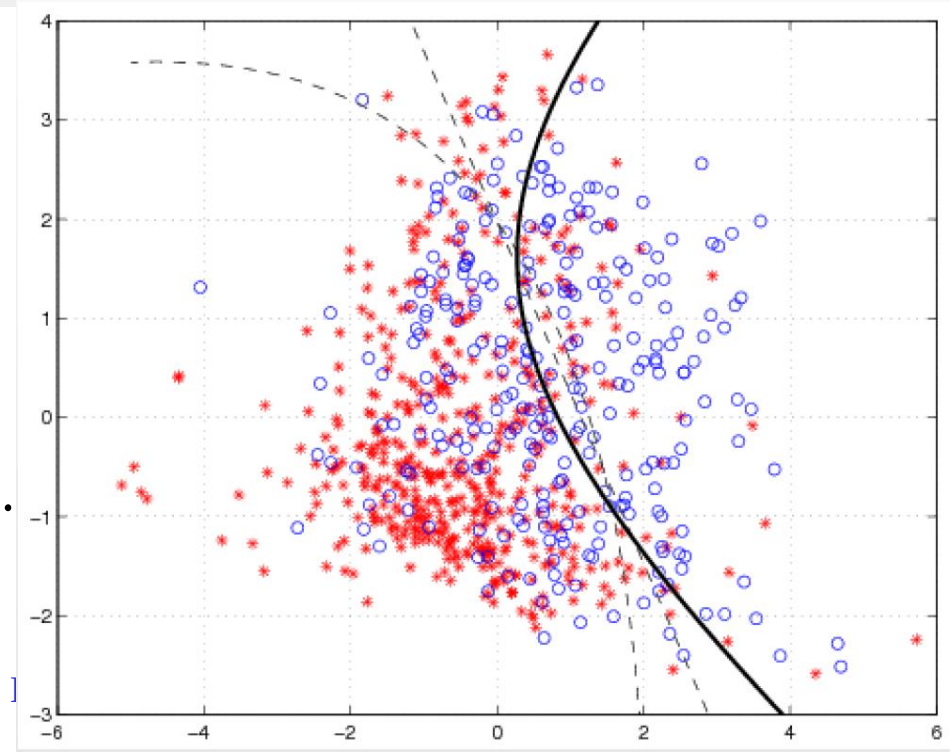
$$\hat{\mu}_1 = \begin{pmatrix} -0.4035 \\ -0.1935 \\ 0.0321 \\ 1.8363 \\ 1.6306 \end{pmatrix} \qquad \hat{\mu}_2 = \begin{pmatrix} 0.7528 \\ 0.3611 \\ -0.0599 \\ 2.5680 \\ 1.9124 \end{pmatrix}$$

$$\hat{\Sigma} = \begin{pmatrix} 1.7925 & -0.1461 & -0.6254 & 0.3548 & 0.5215 \\ -0.1461 & 1.6634 & 0.6073 & -0.7421 & 1.2193 \\ -0.6254 & 0.6073 & 3.5751 & -1.1118 & -0.5044 \\ 0.3548 & -0.7421 & -1.1118 & 12.3355 & -0.0957 \\ 0.5215 & 1.2193 & -0.5044 & -0.0957 & 4.4650 \end{pmatrix}$$

Classification boundary:

$$0.651 - 0.728x_1 - 0.552x_2 - 0.006x_1x_2 - 0.071x_1^2 + 0.170x_2^2 = 0$$

If the linear function on the right hand side is non-negative, classify as 1; otherwise 2.



Within training data
classification error rate: 26.82%.

Linear Classification

- All we require here is the class boundaries $\{x: \delta_k(x) = \delta_j(x)\}$ be linear for every (k, j) pair
- One can achieve this if $\delta_k(x)$ themselves are linear or any monotone transform of $\delta_k(x)$ is linear
  - An example:

$$P(G=1 \mid X=x) = \frac{\exp(\beta_0 + \beta^T x)}{1+\exp(\beta_0 + \beta^T x)}$$

$$P(G=2 \mid X=x) = \frac{1}{1+\exp(\beta_0 + \beta^T x)}$$

So that $$\log\left[\frac{P(G=1 \mid X=x)}{P(G=2 \mid X=x)}\right] = \beta_0 + \beta^T x$$

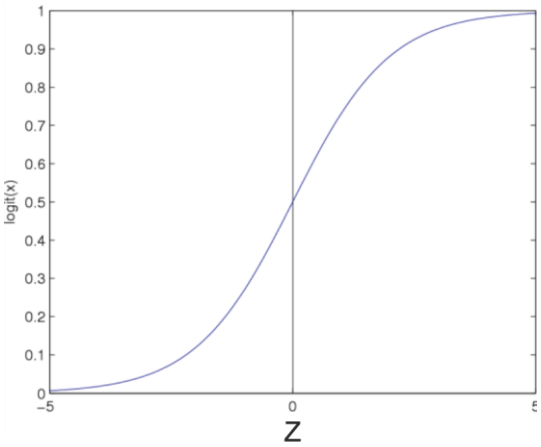
Linear

Logistic Regression

- Learn P(Y | **X**) directly!
  - Assume a particular functional form
  - Sigmoid applied to a linear function of the data:

$$P(Y = 1|X) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^{n} w_i X_i)}$$

Logistic function (or Sigmoid): $\frac{1}{1 + exp(-z)}$

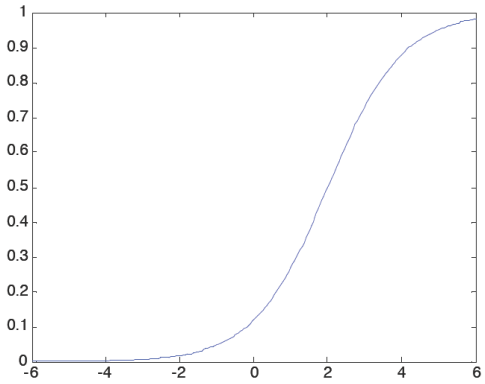


Ref: Carlos Guestrin

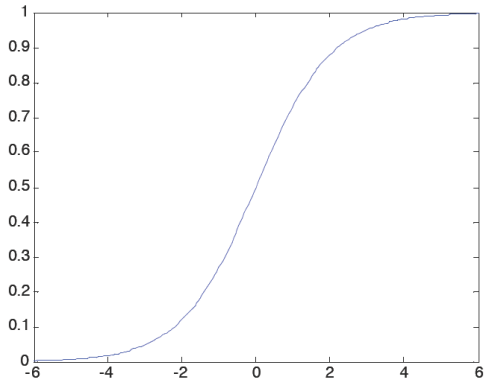
Understanding The Sigmoid

$$g(w_0 + \sum_i w_i x_i) = \frac{1}{1 + e^{w_0 + \sum_i w_i x_i}}$$

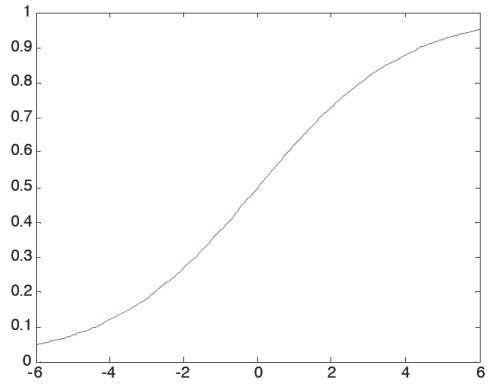
w_0=-2, w_1=-1



w_0=0, w_1=-1



w_0=0, w_1=-0.5



Ref: Carlos Guestrin

Logistic Regression – A Linear Classifier

$$P(Y=1|X=<X_1,...X_n>)=\frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

implies

$$P(Y=0|X=<X_1,...X_n>)=\frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

implies

$$\frac{P(Y=0|X)}{P(Y=1|X)}=exp(w_0+\sum_i w_i X_i)$$

implies

$$\ln\frac{P(Y=0|X)}{P(Y=1|X)}=w_0+\sum_i w_i X_i$$

linear classification rule!

Ref: Carlos Guestrin

Solving Binomial Logistic Regression

$$p = \frac{1}{1 + e^{-(b + \vec{x} \cdot \vec{w})}} \qquad L(\vec{w}, b) = \prod_{i=1}^{n} p(\vec{x}_i)^{y_i} (1 - p(\vec{x}_i)^{1 - y_i}$$

$$\begin{aligned}
\ell(\vec{w}, b) &= \sum_{i=1}^{n} y_i \log p(\vec{x}_i) + (1 - y_i) \log 1 - p(\vec{x}_i) \\
&= \sum_{i=1}^{n} \log 1 - p(\vec{x}_i) + \sum_{i=1}^{n} y_i \log \frac{p(\vec{x}_i)}{1 - p(\vec{x}_i)} \\
&= \sum_{i=1}^{n} \log 1 - p(\vec{x}_i) + \sum_{i=1}^{n} y_i (b + \vec{x}_i \cdot \vec{w}) \\
&= \sum_{i=1}^{n} - \log 1 + e^{b + \vec{x}_i \cdot \vec{w}} + \sum_{i=1}^{n} y_i (b + \vec{x}_i \cdot \vec{w}) \\
\frac{\partial \ell}{\partial w_j} &= - \sum_{i=1}^{n} \frac{1}{1 + e^{b + \vec{x}_i \cdot \vec{w}}} e^{b + \vec{x}_i \cdot \vec{w}} x_{ij} + \sum_{i=1}^{n} y_i x_{ij} \\
&= \sum_{i=1}^{n} (y_i - p(\vec{x}_i; b, \vec{w})) x_{ij}
\end{aligned}$$