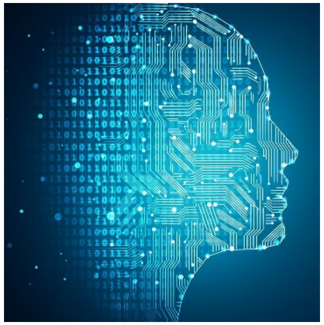
Principles of Machine Learning

Bayes Classifier

Kevin Moon (kevin.moon@usu.edu)

STAT/CS 5810/6655



Outline

1. Multivariate Gaussian distribution
2. Probabilistic setting for classification
3. Bayes classifier
4. Plug-in Methods
    1. Linear Discriminant Analysis
    2. Logistic Regression
    3. Naïve Bayes

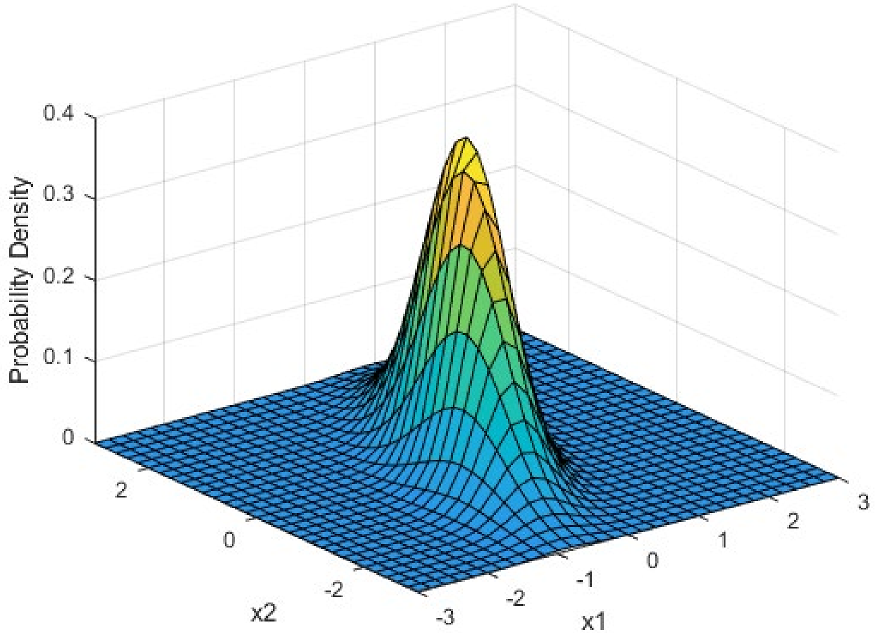
Multivariate Gaussian Distribution

- We say $\boldsymbol{X} \in \mathbb{R}^d$ has a *(multivariate) Gaussian distribution* if its joint pdf is

$$\phi(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) := (2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})\right)$$

  where $\boldsymbol{\mu} \in \mathbb{R}^d$ and Σ is symmetric and positive definite.

- Notation: $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$
- $E[\boldsymbol{X}] = \boldsymbol{\mu}$
- $E[(\boldsymbol{X}-\boldsymbol{\mu})(\boldsymbol{X}-\boldsymbol{\mu})^T] = \Sigma$
- Level sets of a MVG are
- Many uses in machine learning



Classification: Probabilistic Setting

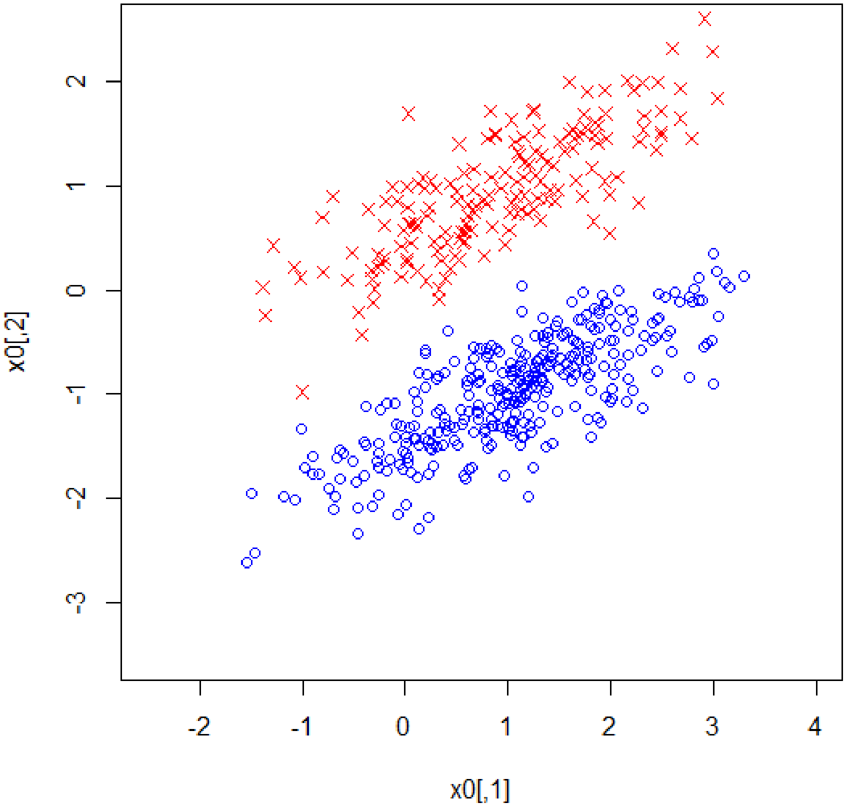
- We are interested in classification (part of supervised learning)
- Feature vector $\boldsymbol{X} \in \mathbb{R}^d$
- Label $Y \in \{1, \dots, M\}$
- Assume $(\boldsymbol{X}, Y)$ are jointly distributed ($d+1$ dimensional)
- Two ways to think about the joint distribution:
    1. $P_{\boldsymbol{X}Y} \leftrightarrow \left(P_{\boldsymbol{X}|Y}, P_Y\right)$
    2. $P_{\boldsymbol{X}Y} \leftrightarrow \left(P_{\boldsymbol{X}}, P_{Y|\boldsymbol{X}}\right)$

Classification: Probabilistic Setting

- Two ways to think about the joint distribution
- Binary classification: $Y \in \{0,1\}$
- Notation:
  - Prior class distribution
    - $\pi := \Pr(Y = 1)$
  - Class-conditional distributions
    - $p_0(\boldsymbol{x}) := p_{\boldsymbol{X}|Y=0}(\boldsymbol{x}|0)$
    - $p_1(\boldsymbol{x}) := p_{\boldsymbol{X}|Y=1}(\boldsymbol{x}|1)$
  - Marginal distribution of $\boldsymbol{X}$
    - $p(\boldsymbol{x}) := P_{\boldsymbol{X}}(\boldsymbol{x})$
  - Posterior class distribution
    - $\eta(\boldsymbol{x}) := P_{Y|\boldsymbol{X}=\boldsymbol{x}}(1|\boldsymbol{x})$
- First way: $P_{\boldsymbol{X}Y} \leftrightarrow \left(\pi, p_0(\boldsymbol{x}), p_1(\boldsymbol{x})\right)$
- Second way: $P_{\boldsymbol{X}Y} \leftrightarrow \left(p(\boldsymbol{x}), \eta(\boldsymbol{x})\right)$
- The two representations are equivalent, but one may be more useful than the other depending on the context

Classification: Probabilistic Setting

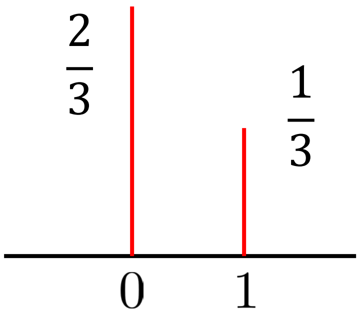
- **Example**: $\pi = 1/3$, p_y are bivariate Gaussians
- $P_{XY} \leftrightarrow \left(\pi, p_0(\boldsymbol{x}), p_1(\boldsymbol{x})\right)$

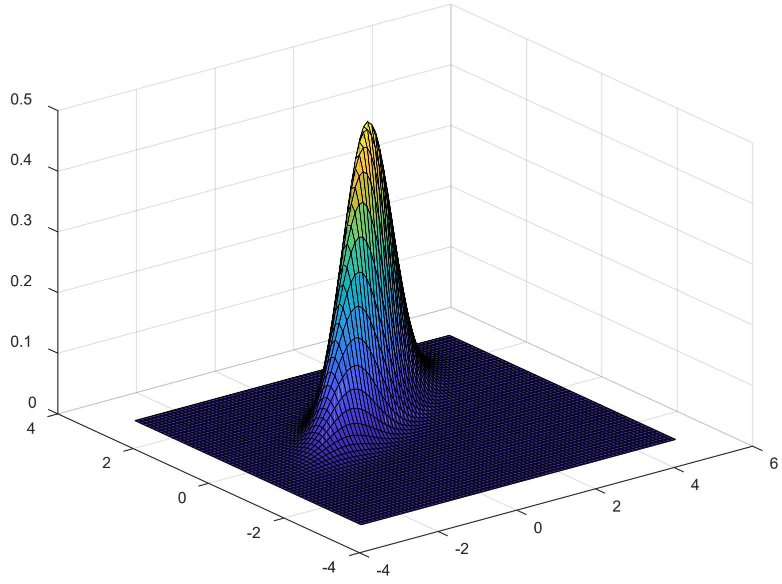


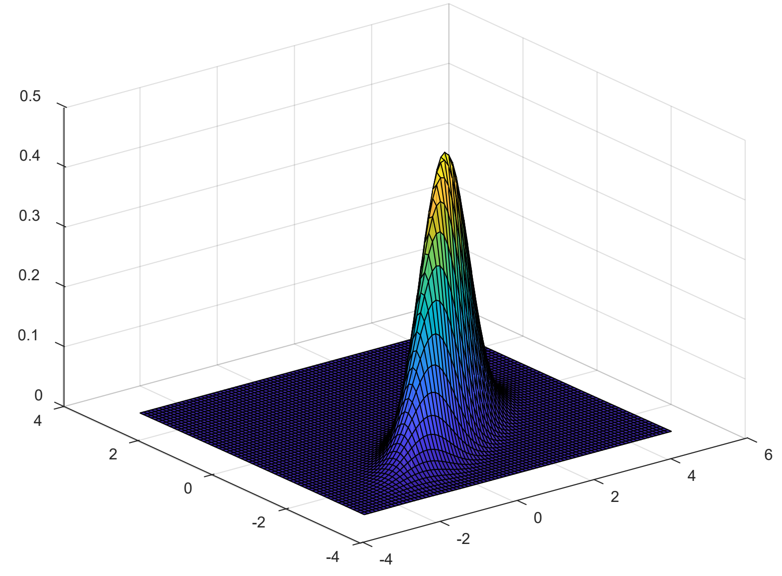

```
N = 500
p = 1/3
y = rbinom(N,1,p)
mu0 = c(1,-1)
mu1 = c(1,1)
Sigma = matrix(c(.9,.4,.4,.3),2,2)
N1 = sum(y)
N0 = N-N1
x0 = mvrnorm(N0,mu0,Sigma)
x1 = mvrnorm(N1,mu1,Sigma)
plot(x0,col='blue',xlim=c(-2.5,4),ylim=c(-3.5,2.5))
points(x1,col='red',pch=4)
```

Classification: Probabilistic Setting

- **Example**: $\pi = 1/3$, p_y are bivariate Gaussians
- $P_{\boldsymbol{X}Y} \leftrightarrow \left(\pi, p_0(\boldsymbol{x}), p_1(\boldsymbol{x})\right)$







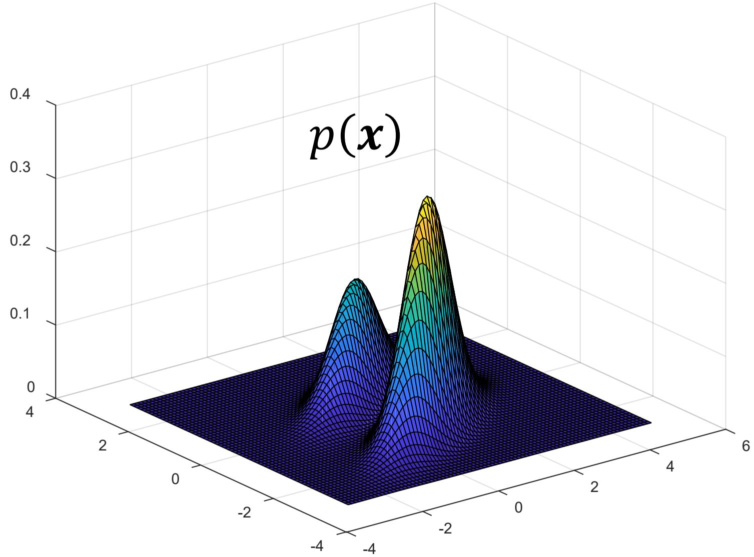
Classification: Probabilistic Setting

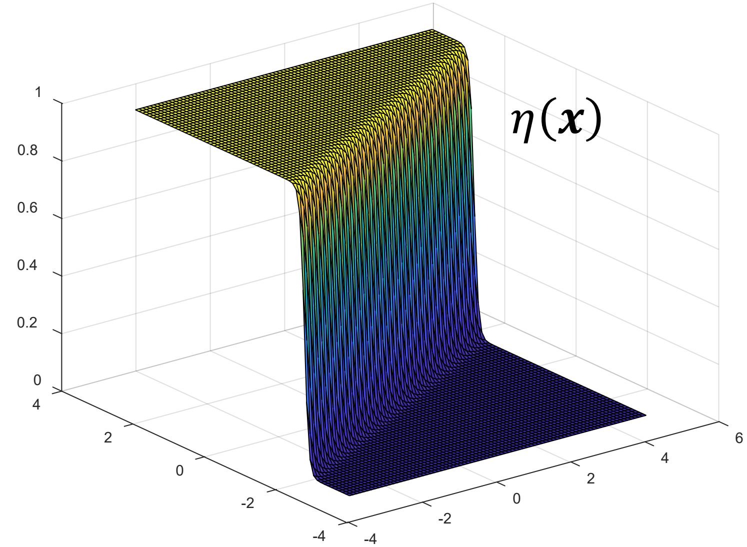
- **Example**: $\pi = 1/3$, p_y are bivariate Gaussians
- $P_{XY} \leftrightarrow (p(\boldsymbol{x}), \eta(\boldsymbol{x}))$
- Law of total probability:

$$p(\boldsymbol{x}) = \pi p_1(\boldsymbol{x}) + (1-\pi) p_0(\boldsymbol{x})$$

- Bayes rule:

$$\eta(\boldsymbol{x}) = \frac{\pi p_1(\boldsymbol{x})}{\pi p_1(\boldsymbol{x}) + (1-\pi) p_0(\boldsymbol{x})}$$





Multiclass Classification

- Feature vector $\boldsymbol{X} \in \mathbb{R}^d$
- Label $Y \in \{1, \dots, M\}$
- Assume $(\boldsymbol{X}, Y)$ are jointly distributed ($d + 1$ dimensional)
- Notation:
  - $\pi_k \coloneqq \Pr(Y = k)$
  - $p_k(\boldsymbol{x}) \coloneqq p_{\boldsymbol{X}|Y=k}(\boldsymbol{x}|k)$
  - $p(\boldsymbol{x}) = \sum_{k=1}^{M} \pi_k p_k(\boldsymbol{x})$
  - $\eta_k(\boldsymbol{x}) \coloneqq P_{Y|\boldsymbol{X}=\boldsymbol{x}}(k|\boldsymbol{x})$
- Equivalent representations
  - $P_{\boldsymbol{X}Y} \leftrightarrow \left(\pi_1, \dots, \pi_M, p_1(\boldsymbol{x}), \dots, p_M(\boldsymbol{x})\right)$
  - $P_{\boldsymbol{X}Y} \leftrightarrow \left(p(\boldsymbol{x}), \eta_1(\boldsymbol{x}), \dots, \eta_M(\boldsymbol{x})\right)$

Bayes Classifier

- A *classifier* is a function $f: \mathbb{R}^d \rightarrow \{1, \ldots, M\}$
- Given a joint distribution $P_{\boldsymbol{X}Y}$ of $(\boldsymbol{X}, Y)$, what is the best possible classifier?
  - Depends on how you measure performance
- Most common classification performance measure is the probability of error, or *risk*
$$R(f) := P_{\boldsymbol{X}Y}(f(\boldsymbol{X}) \neq Y)$$
  - I.e., the probability of the event
$$\{(\boldsymbol{x}, y) \in \mathbb{R}^d \times \{1, \ldots, M\} \big| f(\boldsymbol{x}) \neq y\}$$
- The *Bayes risk* is the smallest risk of any classifier, and is denoted R^*
- If $R(f) = R^*$, f is called a *Bayes classifier*

Bayes Classifier

- **Theorem:** The classifier

$$f^*(\boldsymbol{x}) = \arg\max_{k=1,\ldots,M} \eta_k(\boldsymbol{x})$$
$$= \arg\max_{k=1,\ldots,M} \pi_k p_k(\boldsymbol{x})$$

is a Bayes classifier.

Bayes Classifier: Proof

- **Theorem:** The classifier

$$f^*(\boldsymbol{x}) = \arg\max_{k=1,\ldots,M} \eta_k(\boldsymbol{x}) \\ = \arg\max_{k=1,\ldots,M} \pi_k p_k(\boldsymbol{x})$$

is a Bayes classifier.

Bayes Classifier: Proof

For convenience, assume $\boldsymbol{X} \mid Y = k$ has a continuous distribution for each k. Let f denote an arbitrary classifier. Denote the decision regions

$$\Gamma_k(f) = \{\boldsymbol{x} \mid f(\boldsymbol{x}) = k\}$$

Then

$$\begin{aligned}
1 - R(f) &= P_{\boldsymbol{X}Y}(f(\boldsymbol{X}) = Y) \\
&= \sum_{k=1}^{M} P_Y(Y = k) \cdot P_{\boldsymbol{X}|Y=k}(f(\boldsymbol{X}) = k) \\
&= \sum_{k=1}^{M} \pi_k \cdot \int_{\Gamma_k(f)} p_k(\boldsymbol{x}) d\boldsymbol{x} \\
&= \int_{\mathbb{R}^d} (\sum_{k=1}^{M} \pi_k p_k(\boldsymbol{x}) \mathbf{1}_{\{\boldsymbol{x} \in \Gamma_k(f)\}}) d\boldsymbol{x}
\end{aligned}$$

where $\mathbf{1}_A$ denotes the indicator function on event A.

Bayes Classifier: Proof (cont.)

Notice that $\Gamma_1(f), \ldots \Gamma_K(f)$ from a partition of $\mathbb{R}^d$, i.e., every $\boldsymbol{x} \in \mathbb{R}^d$ belongs to one and only one $\Gamma_k(f)$. Thus, to maximize $1 - R(f)$, we should choose $\Gamma_k(f)$ such that

$$\boldsymbol{x} \in \Gamma_k(f) \iff \pi_k p_k(\boldsymbol{x}) \text{ is maximal.}$$

So a Bayes classifier is

$$f^*(\boldsymbol{x}) = \arg\max_k \pi_k p_k(\boldsymbol{x}).$$

Now note that $\sum_{l=1}^{M} \pi_l p_l(\boldsymbol{x})$ is independent of k. The proof is completed by observing

$$\eta_k(\boldsymbol{x}) = \frac{\pi_k p_k(\boldsymbol{x})}{\sum_{l=1}^{M} \pi_l p_l(\boldsymbol{x})}$$

which follows by Bayes' rule.

The Bayes Risk

- Binary case: a corollary of the theorem is that

$$R^* = \int \min\big(\eta(\boldsymbol{x}), 1 - \eta(\boldsymbol{x})\big) p(\boldsymbol{x}) d\boldsymbol{x}$$

$$= \int \min\big(\pi p_1(\boldsymbol{x}), (1 - \pi) p_0(\boldsymbol{x})\big) d\boldsymbol{x}$$

- Multi-class case

$$R^* = 1 - \int \max_k \big(\eta_k(\boldsymbol{x})\big) p(\boldsymbol{x}) d\boldsymbol{x}$$

$$= 1 - \int \max_k \big(\pi_k p_k(\boldsymbol{x})\big) d\boldsymbol{x}$$

Plug-in Classifiers

LDA, naïve Bayes, logistic regression

Plug-in classifiers

- In most machine learning problems, P_{XY} is unknown
  - Therefore, so is the Bayes classifier
- One approach: estimate the quantities from training data and plug the estimates in the formula to get a classifier
- Linear discriminant analysis (LDA) and naïve Bayes have the form

$$f(\boldsymbol{x}) = \arg\max_k \hat{\pi}_k \, \hat{p}_k(\boldsymbol{x})$$

- Logistic regression has the form

$$f(\boldsymbol{x}) = \arg\max_k \hat{\eta}_k(\boldsymbol{x})$$

Linear Discriminant Analysis

- Training data

$$(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \overset{iid}{\sim} P_{\boldsymbol{X}Y}.$$

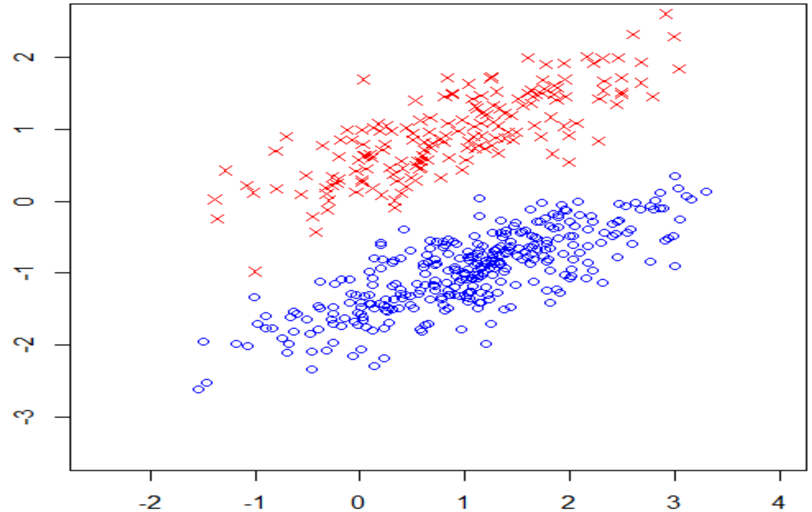
- *LDA assumption*:

$$\boldsymbol{X} \mid Y = k \sim \mathcal{N}(\boldsymbol{\mu}_k, \Sigma), \quad k = 1, \ldots, M$$

for some unknown $\mu_1, \ldots, \mu_M$ and Σ. Equivalently

$$p_k(\boldsymbol{x}) = (2\pi)^{-d/2}|\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_k)^T \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_k)\right)$$

- LDA is the plug-in rule based on this model. We use training data to estimate $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_M$ and Σ.



LDA Estimates

- LDA is the classifier obtained by plugging the following into the Bayes classifier formula:
  - $\hat{\pi}_k = \frac{n_k}{n}$, $n_k = |\{i : y_i = k\}|$
  - $\hat{\boldsymbol{\mu}}_k = \frac{1}{n_k} \sum_{i: y_i = k} \boldsymbol{x}_i$
  - $\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} \left(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_{y_i}\right)\left(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_{y_i}\right)^T$
- $\hat{\boldsymbol{\mu}}_k$ is the *sample mean* for each class
- $\hat{\Sigma}$ is the *pooled sample covariance*
- These estimates are all *maximum likelihood estimates*

LDA is a linear classifier

- Binary setting, $Y \in \{0, 1\}$. A classifier f is called *linear* if it has the form

$$f(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{w}^T\boldsymbol{x} + b \geq 0 \\ 0 & \text{if } \boldsymbol{w}^T\boldsymbol{x} + b < 0 \end{cases}$$

  for some $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$. (The case $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ can be labeled arbitrarily.)

LDA is a linear classifier

- Binary setting, $Y \in \{0, 1\}$. A classifier f is called *linear* if it has the form

$$f(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{w}^T\boldsymbol{x} + b \geq 0 \\ 0 & \text{if } \boldsymbol{w}^T\boldsymbol{x} + b < 0 \end{cases}$$

for some $\boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$. (The case $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ can be labeled arbitrarily.)

$$f(x) = \begin{cases} 1 & \hat{\pi}_1 \hat{p}_1(x) \geq \hat{\pi}_0 \hat{p}_0(x) \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} 1 & \log \hat{\pi}_1 + \log \hat{p}_1(x) \geq \log \hat{\pi}_0 + \log \hat{p}_0(x) \\ 0 & \text{ow} \end{cases}$$

Need to show

$$\log \hat{\pi}_1 + \log \hat{p}_1(x) - \log \hat{\pi}_0 - \log \hat{p}_0(x) = w^T x + b$$

LDA is a linear classifier

$$\log \hat{p}_1(x) - \log \hat{p}_0(x) = \log\left(\cancel{(2\pi)^{-d/2}} |\hat{\Sigma}|^{-1/2} \right)$$

$$- \frac{1}{2}(x - \hat{\mu}_1)^T \hat{\Sigma}^{-1} (x - \hat{\mu}_1)$$

$$- \log\left(\cancel{(2\pi)^{-d/2}} |\hat{\Sigma}|^{-1/2} \right) + \frac{1}{2}(x - \hat{\mu}_0)^T \hat{\Sigma}^{-1} (x - \hat{\mu}_0)$$

$$= -\frac{1}{2}\left[\cancel{x^T \hat{\Sigma}^{-1} x} - 2x^T \hat{\Sigma}^{-1} \hat{\mu}_1 + \hat{\mu}_1^T \hat{\Sigma}^{-1} \hat{\mu}_1 \right]$$

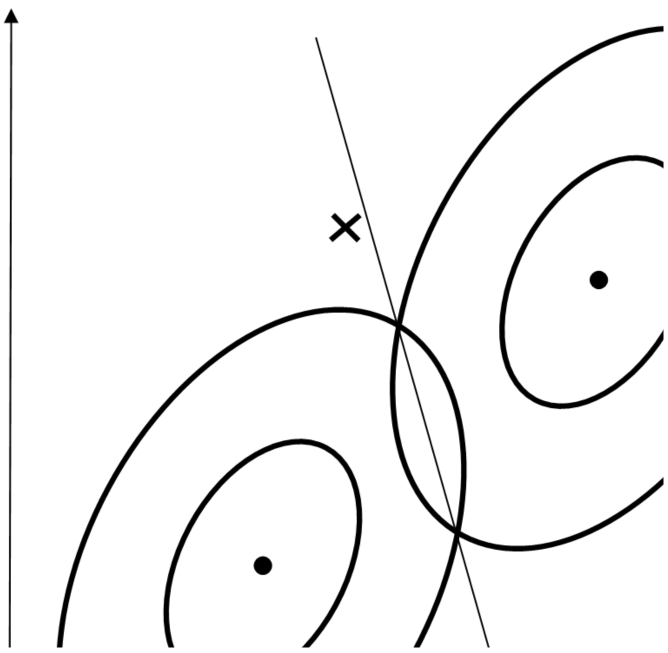
$$+ \frac{1}{2}\left[\cancel{x^T \hat{\Sigma}^{-1} x} - 2x^T \hat{\Sigma}^{-1} \hat{\mu}_0 + \hat{\mu}_0 \hat{\Sigma}^{-1} \hat{\mu}_0 \right]$$

$$= w^T x + b$$

Mahalanobis Distance

- Which mean is closer to the test point?
  - Figure assumes $\pi_0 = \pi_1$
- The LDA classifier assigns $\boldsymbol{x}$ to the class with the nearest “centroid” $\widehat{\boldsymbol{\mu}}_0$ or $\widehat{\boldsymbol{\mu}}_1$ where distance is the Mahalanobis distance:

$$d_M(\boldsymbol{x}, \widehat{\boldsymbol{\mu}}) := \sqrt{(\boldsymbol{x} - \widehat{\boldsymbol{\mu}})^T \widehat{\Sigma}^{-1} (\boldsymbol{x} - \widehat{\boldsymbol{\mu}})}$$



Group Exercise

1. Is LDA generative or discriminative?
2. Is LDA parametric or nonparametric?
3. What do the decision regions look like for the multiclass case?
4. Interpret LDA in the case where Σ is assumed to be a multiple of the identity $\sigma^2 I$
5. Describe the decision boundary in the two-class case when the covariance matrices are not assumed to be the same but are estimated separately
6. What are some drawbacks of LDA?

Naïve Bayes

Review: Plug-in classifiers

- In most machine learning problems, $P_{\boldsymbol{X}Y}$ is unknown
  - Therefore, so is the Bayes classifier
- One approach: estimate the quantities from training data and plug the estimates in the formula to get a classifier
- Linear discriminant analysis (LDA) and naïve Bayes have the form

$$f(\boldsymbol{x}) = \arg\max_k \hat{\pi}_k \, \hat{p}_k(\boldsymbol{x})$$

- Logistic regression has the form

$$f(\boldsymbol{x}) = \arg\max_k \hat{\eta}_k(\boldsymbol{x})$$

Naïve Bayes Assumption

- Training data
$$(\boldsymbol{X}_1, Y_1), \dots, (\boldsymbol{X}_n, Y_n) \sim P_{\boldsymbol{X}Y}.$$
- Notation:
$$\boldsymbol{X} = \begin{bmatrix} X^{(1)} \\ \vdots \\ X^{(d)} \end{bmatrix}$$
- Naïve Bayes assumption: given Y, the components $X^{(1)}, \dots, X^{(d)}$ are independent
- Naïve Bayes is a plug-in method. It could be generative or discriminative and parametric or nonparametric depending on how the distribution of $X^{(j)}|Y = k$ is modeled.

Naïve Bayes

- Main use: Features with finite range
- Assume the possible outcomes of $X^{(j)}$ are $z_1, \dots, z_L$.
- **Example:** Document Classification
- Suppose we wish to classify documents into categories like "business," "politics," "sports," etc. A simple yet popular feature representation is the bag-of-words representation. A document is represented as a vector

$$\boldsymbol{X} = \begin{bmatrix} X^{(1)} \\ \vdots \\ X^{(d)} \end{bmatrix}$$

where d is the number of words in the vocabulary, and

$$X^{(j)} = \begin{cases} 1 & \text{if } j^{th} \text{ word occurs in document} \\ 0 & \text{otherwise.} \end{cases}$$

- In this example, $L = 2$

Naïve Bayes Classifier

- Let $p_k(\boldsymbol{x})$ be the pmf of $\boldsymbol{X}|Y = k$. By the Naïve Bayes assumption

$$p_k(\boldsymbol{x}) = \prod_{j=1}^{d} p_k^{(j)}\left(x^{(j)}\right)$$

where $p_k^{(j)}\left(x^{(j)}\right)$ is the marginal pmf of $X^{(j)}|Y = k$.

- Let $(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n)$ be the training data and let

$$\hat{\pi}_k = \frac{n_k}{n}, \qquad n_k = |\{i: y_i = k\}|$$

$$\hat{p}_k^{(j)} = \text{estimate of } p_k^{(j)}$$

- Then the Naïve Bayes classifier is

$$\hat{f}(x) = \arg\max_{\mathrm{k}} \hat{\pi}_k \prod_{j=1}^{d} \hat{p}_k^{(j)}\left(x^{(j)}\right)$$

Naïve Bayes Classifier

- How should we estimate $p_k^{(j)}$?
- Denote

$$n_k = |\{i: y_i = k\}|$$

$$n_{kl}^{(j)} = \left|\left\{i: y_i = k \; AND \; x_i^{(j)} = z_l\right\}\right|$$

- Then the natural (and maximum likelihood) estimate of

$$p_k^{(j)}(z_l) = \Pr\left\{X^{(j)} = z_l | Y = k\right\}$$

is

$$\hat{p}_k^{(j)}(z_l) = \frac{n_{kl}^{(j)}}{n_k}$$

Logistic Regression

Review: Plug-in classifiers

- In most machine learning problems, $P_{\boldsymbol{X}Y}$ is unknown
  - Therefore, so is the Bayes classifier
- One approach: estimate the quantities from training data and plug the estimates in the formula to get a classifier
- Linear discriminant analysis (LDA) and naïve Bayes have the form

$$f(\boldsymbol{x}) = \arg\max_k \hat{\pi}_k \, \hat{p}_k(\boldsymbol{x})$$

- Logistic regression has the form

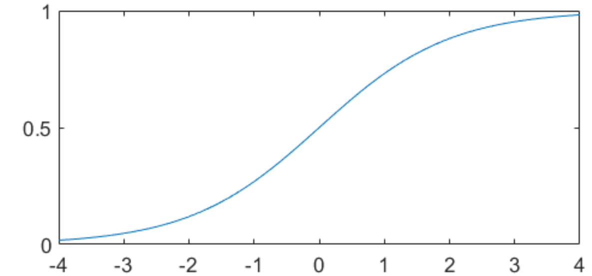
$$f(\boldsymbol{x}) = \arg\max_k \hat{\eta}_k(\boldsymbol{x})$$

Logistic Regression

- For binary classification with labels $Y \in \{0,1\}$, the Bayes classifier can be written as

$$f^*(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \eta(\boldsymbol{x}) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

  - Recall $\eta(\boldsymbol{x}) \coloneqq \Pr(Y = 1 | \boldsymbol{X} = \boldsymbol{x})$



- Logistic Regression is a plug-in method
  1. Assume
  $$\eta(\boldsymbol{x}) = \frac{1}{1 + \exp\left(-(\boldsymbol{w}^T\boldsymbol{x} + b)\right)}, \qquad \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}$$
  2. Determine an estimate
  $$\hat{\theta} = \begin{bmatrix} \hat{b} \\ \widehat{\boldsymbol{w}} \end{bmatrix} \text{ of } \theta = \begin{bmatrix} b \\ \boldsymbol{w} \end{bmatrix} \in \mathbb{R}^{d+1}$$
  3. Plug the below estimate into the formula for the Bayes classifier
  $$\hat{\eta}(\boldsymbol{x}) = \frac{1}{1 + \exp\left(-\left(\widehat{\boldsymbol{w}}^T\boldsymbol{x} + \hat{b}\right)\right)}$$

Logistic Regression is a linear classifier

$$\hat{f}(x) = 1 \quad \Leftrightarrow \quad \hat{\eta}(x) \geq \frac{1}{2}$$

$$\Leftrightarrow \quad \frac{1}{1 + \exp\left(-(\widehat{\boldsymbol{w}}^T \boldsymbol{x} + \hat{b})\right)} \geq \frac{1}{2}$$

$$\Leftrightarrow \quad 1 \geq \exp\left(-(\widehat{\boldsymbol{w}}^T \boldsymbol{x} + \hat{b})\right)$$

$$\Leftrightarrow \quad \widehat{\boldsymbol{w}}^T \boldsymbol{x} + \hat{b} \geq 0$$

Why Logistic Regression?

- More than a classifier—it predicts the probability of each class
  - Gives a little bit of interpretability
- Slightly more flexible than LDA
- Widely used in health sciences and other applications

Visualizing the Posterior

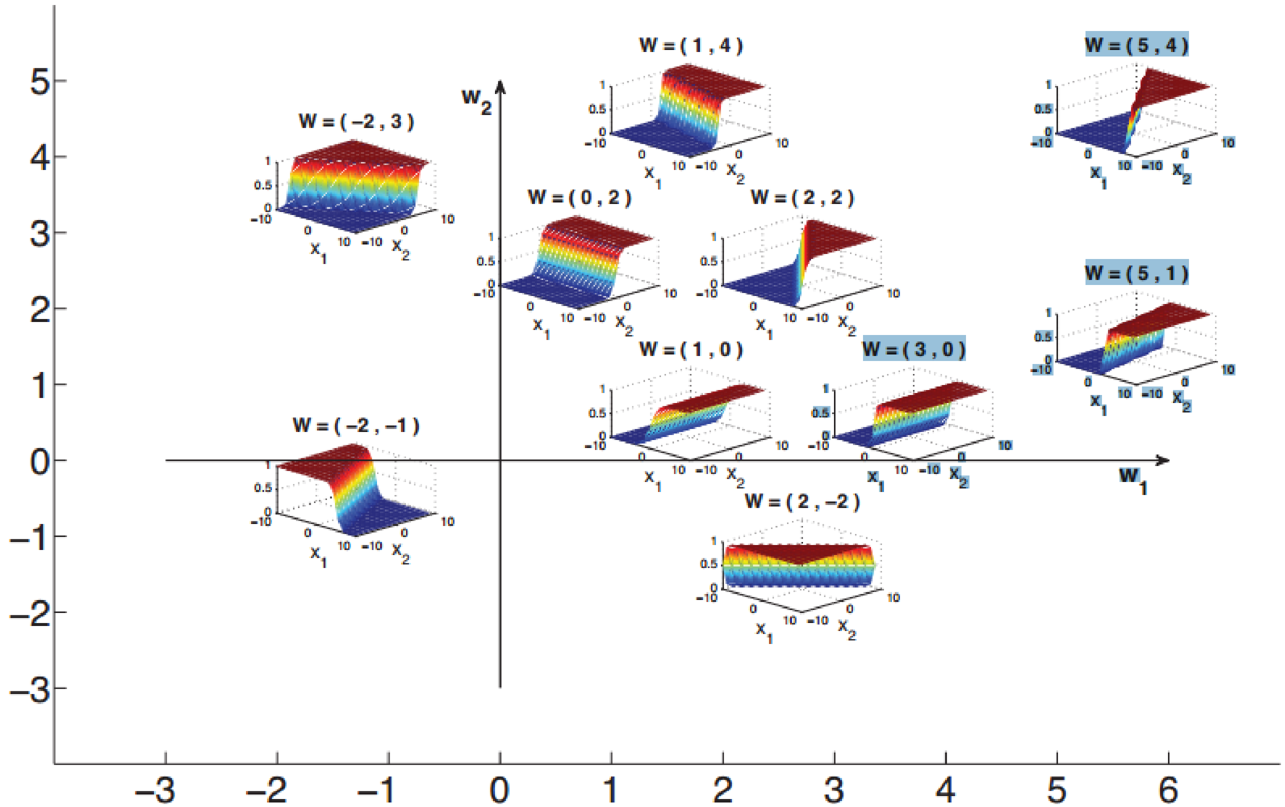
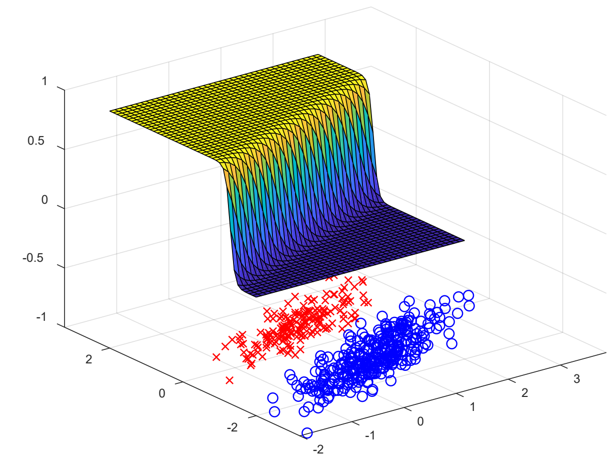


Figure 8.1 Plots of $\mathrm{sigm}(w_1 x_1 + w_2 x_2)$. Here $\mathbf{w} = (w_1, w_2)$ defines the normal to the decision boundary. Points to the right of this have $\mathrm{sigm}(\mathbf{w}^T\mathbf{x}) > 0.5$, and points to the left have $\mathrm{sigm}(\mathbf{w}^T\mathbf{x}) < 0.5$. Based on Figure 39.3 of (MacKay 2003). Figure generated by `sigmoidplot2D`.

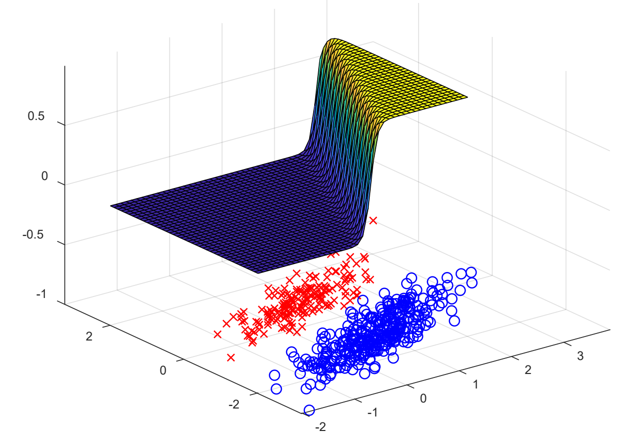
Figure from Murphy, p. 246

Implementing Logistic Regression

- How do you estimate $\theta = \begin{bmatrix} b \\ \boldsymbol{w} \end{bmatrix}$?
- $\min_{\boldsymbol{\theta}} \sum_i \left(y_i - \eta(\boldsymbol{x}_i; \boldsymbol{\theta})\right)^2$?
  - Not convex
- $\min_{\boldsymbol{\theta}}$ training error?
  - Not convex nor differentiable
- Maximum likelihood



Good



Bad

Maximum likelihood estimation (MLE)

- Let $p(y|\boldsymbol{x};\boldsymbol{\theta})$ denote the conditional pmf of y given $\boldsymbol{x}$
  - It is also a function of $\boldsymbol{\theta}$
- Observe

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \begin{cases} 1-\eta(\boldsymbol{x};\boldsymbol{\theta}) & y=0 \\ \eta(\boldsymbol{x};\boldsymbol{\theta}) & y=1 \end{cases}$$

$$= \eta(\boldsymbol{x};\boldsymbol{\theta})^y\left(1-\eta(\boldsymbol{x};\boldsymbol{\theta})\right)^{1-y}$$

- The *likelihood* of $\boldsymbol{\theta}$ is defined to be

$$L(\boldsymbol{\theta}) := \prod_{i=1}^n p(y_i|\boldsymbol{x}_i;\boldsymbol{\theta})$$

$$= \prod_{i=1}^n \eta(\boldsymbol{x}_i;\boldsymbol{\theta})^{y_i}\left(1-\eta(\boldsymbol{x}_i;\boldsymbol{\theta})\right)^{1-y_i}$$

- Choose $\boldsymbol{\theta}$ that maximizes $L(\boldsymbol{\theta})$

Log likelihood

- Notation

$$\tilde{\boldsymbol{x}}_i = \left[1, x_i^{(1)}, \dots, x_i^{(d)}\right]^T$$
$$\boldsymbol{\theta} = \left[b, w^{(1)}, \dots, w^{(d)}\right]^T$$

- The *log-likelihood* of $\boldsymbol{\theta}$ is

$$\begin{aligned}\ell(\boldsymbol{\theta}) &:= \log L(\boldsymbol{\theta}) \\ &= \sum_{i=1}^{n}\left[y_i \log\left(\frac{1}{1+e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}\right) + (1-y_i)\log\left(\frac{e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}{1+e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}\right)\right]\end{aligned}$$

- Take the derivative wrt $\boldsymbol{\theta}$ and set to zero
  - No closed form solution
  - Need other tools to solve this (optimization theory)

Further reading

- Murphy, Machine Learning: A Probabilistic Perspective
- ISL Sections 2.2 and 4.4
- ESL Sections 2.4 and 4.3