Statistical Learning– MATH 6333
Set 4 (Linear Methods for Classification)

Tamer Oraby

UTRGV

tamer.oraby@utrgv.edu

* Last updated October 18, 2021

Classification Methods

Classification Methods

1. Multiple linear regression and K-NN
2. Logistic (binomial) regression, and multinomial regression
3. Log-linear (Poisson) regression, and negative binomial regression
4. Linear discriminant analysis (LDA)
5. Quadratic discriminant analysis (QDA)
6. Naïve Bayes

for inference and prediction. Here, we use 0-1 loss function.

Classification Methods

1. Multiple linear regression and K-NN
2. Logistic (binomial) regression, and multinomial regression
3. Log-linear (Poisson) regression, and negative binomial regression
4. Linear discriminant analysis (LDA)
5. Quadratic discriminant analysis (QDA)
6. Naïve Bayes

for inference and prediction. Here, we use 0-1 loss function.

Classification Methods

1. Multiple linear regression and K-NN
2. Logistic (binomial) regression, and multinomial regression
3. Log-linear (Poisson) regression, and negative binomial regression
4. Linear discriminant analysis (LDA)
5. Quadratic discriminant analysis (QDA)
6. Naïve Bayes

for inference and prediction. Here, we use 0-1 loss function.

Classification Methods

Special case for illustration

The linear classification model (classifier)

$$f_k(X) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = X\beta_k$$

for the training data

$$\mathcal{T} = \{(x_{i1}, x_{i2}, \ldots, x_{ip}, y_i \text{ or } G_i) : i = 1, 2, \ldots, N\}$$

where

$$G \in \mathcal{G} = \{\text{class}_1, \ldots, \text{class}_K\}$$

The idea is to classify items using linear decision boundaries (affine set/hyperplane) between class_k and class_ℓ

$$\mathcal{B}_{k,\ell} = \{x : x^T\hat{\beta}_k = x^T\hat{\beta}_\ell\}$$

for $k \neq \ell$ and $k, \ell = 1, \ldots, K$.

Classification Methods

Special case for illustration

The linear classification model (classifier)

$$f_k(X) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = X\beta_k$$

for the training data

$$\mathcal{T} = \{(x_{i1}, x_{i2}, \ldots, x_{ip}, y_i \text{ or } G_i) : i = 1, 2, \ldots, N\}$$

where

$$G \in \mathcal{G} = \{\text{class}_1, \ldots, \text{class}_K\}$$

The idea is to classify items using linear decision boundaries (affine set/hyperplane) between class_k and class_ℓ

$$\mathcal{B}_{k,\ell} = \{x : x^T\hat{\beta}_k = x^T\hat{\beta}_\ell\}$$

for $k \neq \ell$ and $k, \ell = 1, \ldots, K$.

Classification Methods

Special case for illustration

The linear classification model (classifier)

$$f_k(X) = \beta_{k,0} + \beta_{k,1} x_1 + \cdots + \beta_{k,p} x_p = X\beta_k$$

for the training data

$$\mathcal{T} = \{(x_{i1}, x_{i2}, \ldots, x_{ip}, y_i \text{ or } G_i) : i = 1, 2, \ldots, N\}$$

where

$$G \in \mathcal{G} = \{\text{class}_1, \ldots, \text{class}_K\}$$

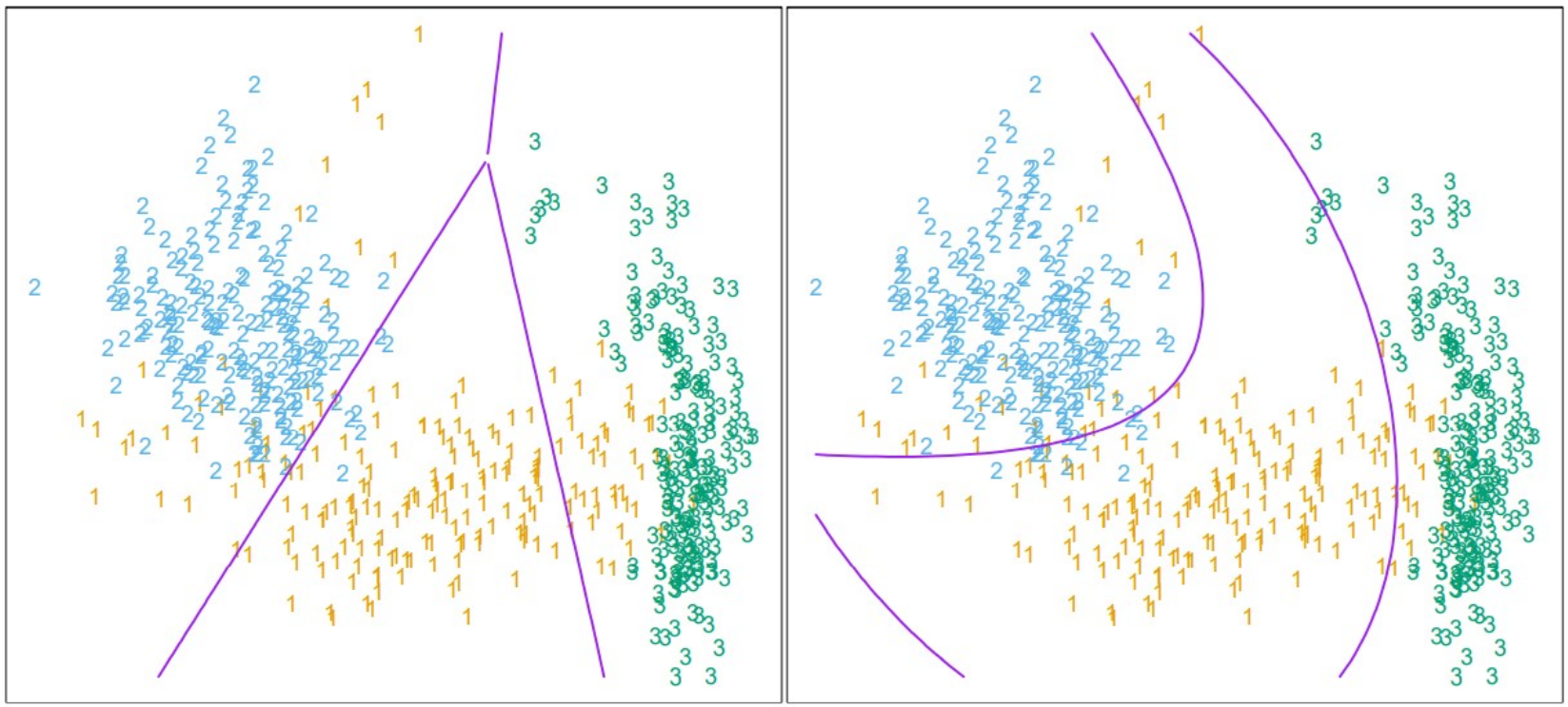
The idea is to classify items using linear decision boundaries (affine set/hyperplane) between class_k and class_ℓ

$$\mathcal{B}_{k,\ell} = \{x : x^T \hat{\beta}_k = x^T \hat{\beta}_\ell\}$$

for $k \neq \ell$ and $k, \ell = 1, \ldots, K$.

Classification Methods

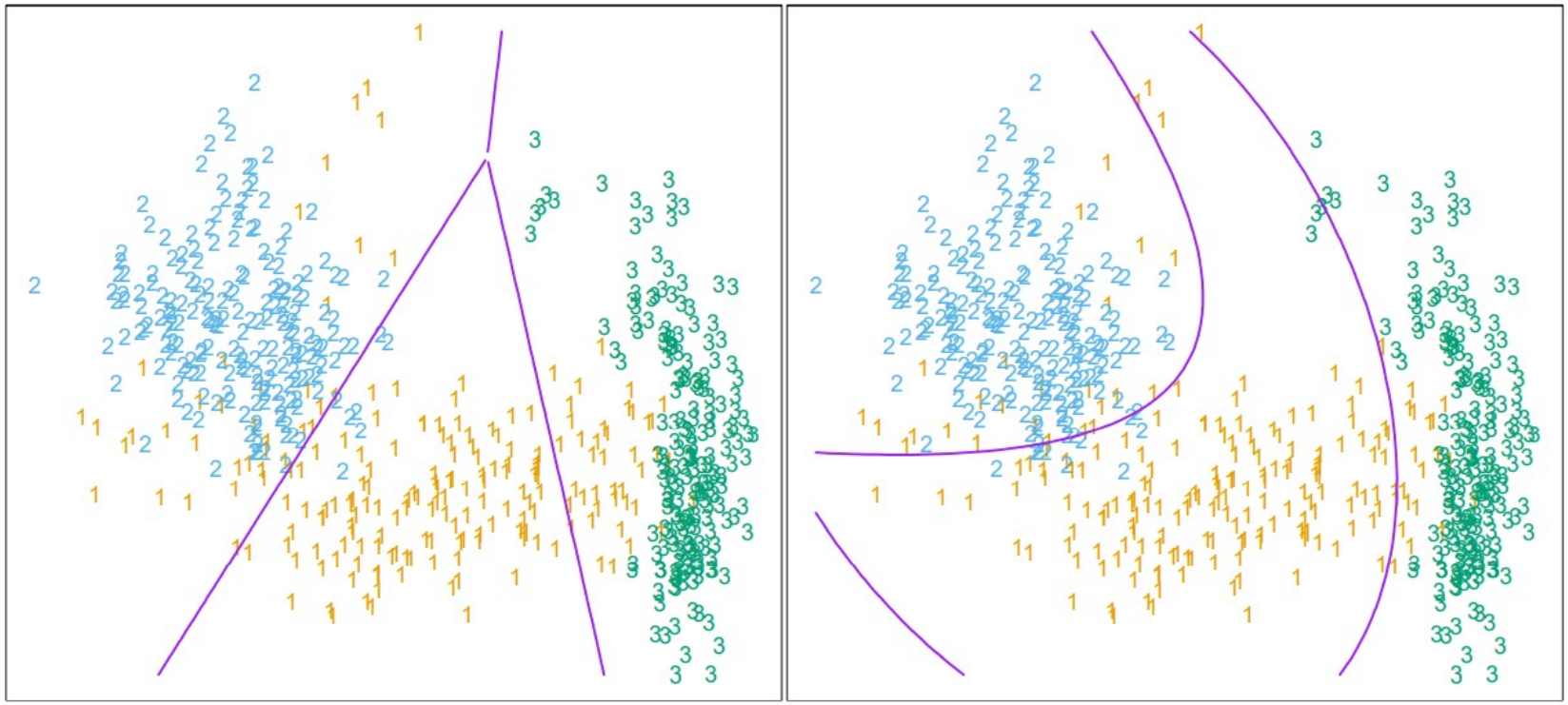
The left panel: decision boundaries are linear in X_1 and X_2.



The right panel: decision boundaries are linear in X_1, X_2, X_1X_2, X_1^2 and X_2^2.

Classification Methods

The left panel: decision boundaries are linear in X_1 and X_2.



The right panel: decision boundaries are linear in X_1, X_2, X_1X_2, X_1^2 and X_2^2.

Classification Methods

Example (South African Heart Disease Data)

A retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa.

Variables		
Name	Description	Type
sbp	systolic blood pressure	continuous
tobacco	cumulative tobacco (kg)	continuous
ldl	low density lipoprotein cholesterol	continuous
adiposity	waist circumference	continuous
famhist	family history of heart disease (Present, Absent)	dichotomous
typea	type-A behavior	continuous
obesity	body mass index (BMI)	continuous
alcohol	current alcohol consumption	continuous
age	age at onset	continuous
chd	response, coronary heart disease (1=yes,0=no)	dichotomous

Classification Methods

Example (Classification of iris flowers)

Setosa

Versicolor

Virginica

Goal: To classify an iris flower based on the inputs: sepal length in cm, sepal width in cm, petal length in cm, and petal width in cm. Iris data and its description are available at `https://archive.ics.uci.edu/ml/datasets/iris`

Classification Methods

Example (Vowel Recognition (Speech) Data)

Recognition of the eleven steady state vowels of British English using the following list of words that are uttered once

Number	Vowel	Word	Number	Vowel	Word
1	i	heed	7	O	hod
2	I	hid	8	C:	hoard
3	E	head	9	U	hood
4	A	had	10	u:	who'd
5	a:	hard	11	3:	heard
6	Y	hud			

The response y is the vowel number and the inputs $x.1, \ldots, x.10$ are log areas calculated using speech signals.

Classification Methods

All textbook data, including SA heart disease data and vowel recognition data, and their description are available at `https://web.stanford.edu/~hastie/ElemStatLearn/data.html`

Classification - Partitioning Methods

Classification - Partitioning Methods

Partitioning methods split the space of inputs into a number of disjoint sub-spaces.

Using a discriminant functions $\delta_k(x)$ for $k = 1, \ldots, K$, a classification prediction about an input x_* could be made using

$$k_* = \mathit{argmax}\, \delta_k(x_*)$$

and the decision boundary between class_k and class_ℓ

$$\mathcal{B}_{k,\ell} = \{x : \delta_k(x) = \delta_\ell(x)\}$$

for $k \neq \ell$ and $k, \ell = 1, \ldots, K$.

Classification - Partitioning Methods

Partitioning methods split the space of inputs into a number of disjoint sub-spaces.

Using a discriminant functions $\delta_k(x)$ for $k = 1, \ldots, K$, a classification prediction about an input x_* could be made using

$$k_* = \textit{argmax}\, \delta_k(x_*)$$

and the decision boundary between class$_k$ and class$_\ell$

$$\mathcal{B}_{k,\ell} = \{x : \delta_k(x) = \delta_\ell(x)\}$$

for $k \neq \ell$ and $k, \ell = 1, \ldots, K$.

Classification - Partitioning Methods

Example (Illustrative example - not real)

For $\mathcal{G} = \{1, 2, 3, 4\}$, let the discriminant function be

$$\delta(x) = c \times (.1 + .2x,\ .2 + .15x,\ .3 - .1x,\ .1 + .05x)$$

for $0 < x < 1$ and some constant $c > 0$.

For $x_* = .25$,

$$\delta(x_*) = c \times (.15, .2375, .275, .1125),$$

thus $k_* = 3$.

Also, the decision boundary between class_1 and class_3

$$\mathcal{B}_{1,3} = \{x : x = \frac{2}{3}\}.$$

Classification - Partitioning Methods

Example (Illustrative example - not real)

For $\mathcal{G} = \{1, 2, 3, 4\}$, let the discriminant function be

$$\delta(x) = c \times (.1 + .2x,\ .2 + .15x,\ .3 - .1x,\ .1 + .05x)$$

for $0 < x < 1$ and some constant $c > 0$.

For $x_* = .25$,

$$\delta(x_*) = c \times (.15, .2375, .275, .1125),$$

thus $k_* = 3$.

Also, the decision boundary between class_1 and class_3

$$\mathcal{B}_{1,3} = \{x : x = \frac{2}{3}\}.$$

Classification - Partitioning Methods

Example (Illustrative example - not real)

For $\mathcal{G} = \{1, 2, 3, 4\}$, let the discriminant function be

$$\delta(x) = c \times (.1 + .2x,\ .2 + .15x,\ .3 - .1x,\ .1 + .05x)$$

for $0 < x < 1$ and some constant $c > 0$.

For $x_* = .25$,

$$\delta(x_*) = c \times (.15, .2375, .275, .1125),$$

thus $k_* = 3$.

Also, the decision boundary between class_1 and class_3

$$\mathcal{B}_{1,3} = \{x : x = \frac{2}{3}\}.$$

Classification - Partitioning Methods

Example (Illustrative example - not real)

For $\mathcal{G} = \{1, 2, 3, 4\}$, let the discriminant function be

$$\delta(x) = c \times (.1 + .2x,\ .2 + .15x,\ .3 - .1x,\ .1 + .05x)$$

for $0 < x < 1$ and some constant $c > 0$.

For $x_* = .25$,

$$\delta(x_*) = c \times (.15, .2375, .275, .1125),$$

thus $k_* = 3$.

Also, the decision boundary between class_1 and class_3

$$\mathcal{B}_{1,3} = \{x : x = \frac{2}{3}\}.$$

Classification - Partitioning Methods

Two general types:

- Probability-based Classification Methods; e.g., multiple linear regression and logistic regression
- Bayes-based Classification Methods; e.g., linear and quadratic discriminant analyses, and naïve Bayes

Classification Methods

Example (Vowel Recognition (Speech) Data)

Technique	Error Rates	
	Training	Test
Linear regression	0.48	0.67
Linear discriminant analysis	0.32	0.56
Quadratic discriminant analysis	0.01	0.53
Logistic regression	0.22	0.51

Probability-based Classification Methods

(Multi-response or K-response) Multiple Linear Regression

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Response Y_i of item i,

$$\begin{aligned} Y_i &= (I(G_i = 1), \ldots, I(G_i = k), \ldots, I(G_i = K)) \\ &= (0, \ldots, \underbrace{1}_{k^{th} \text{ element}}, \ldots, 0) \end{aligned}$$

is an indicator that item i belongs to class k. It is a vector of indicator variables $I(G = k)$, $k = 1, \ldots, K$.

The $N \times K$ indicator response matrix $Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix}$

Let us call column k of the matrix Y by

$$y_k := (I(G_1 = k), \ldots, I(G_N = k))^T$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Response Y_i of item i,

$$Y_i = (I(G_i = 1), \ldots, I(G_i = k), \ldots, I(G_i = K))$$
$$= (0, \ldots, \underbrace{1}_{k^{th} \text{ element}}, \ldots, 0)$$

is an indicator that item i belongs to class k. It is a vector of indicator variables $I(G = k)$, $k = 1, \ldots, K$.

The $N \times K$ indicator response matrix $Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix}$

Let us call column k of the matrix Y by

$$y_k := (I(G_1 = k), \ldots, I(G_N = k))^T$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Response Y_i of item i,

$$\begin{aligned} Y_i &= (I(G_i = 1), \ldots, I(G_i = k), \ldots, I(G_i = K)) \\ &= (0, \ldots, \underbrace{1}_{k^{th} \text{ element}}, \ldots, 0) \end{aligned}$$

is an indicator that item i belongs to class k. It is a vector of indicator variables $I(G = k)$, $k = 1, \ldots, K$.

The $N \times K$ indicator response matrix $Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix}$

Let us call column k of the matrix Y by

$$y_k := (I(G_1 = k), \ldots, I(G_N = k))^T.$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

The model is

$$Y = XB + \mathcal{E}$$

where

- B is a $(p+1) \times K$ coefficient matrix
- X is the $N \times (p+1)$ model matrix with ones in the first column
- $\mathcal{E}$ is a $N \times K$ matrix noise

It is a K simultaneous regression problems of the columns

$$y_k = (I(G_1 = k), \ldots, I(G_N = k))^T$$

over the inputs X to estimate the column vector β_k.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Predictions of the training data X are

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

For a new input x_* (with 1 in the first entry), the prediction $\hat{f}(x_*) = (\hat{f}_1(x_*), \ldots, \hat{f}_K(x_*))$ is the discriminant function $\delta(x_*) = (\delta_1(x_*), \ldots, \delta_K(x_*))$ and is given by

$$\hat{f}(x_*) = x_*^T \hat{B}$$

where $\hat{f}_k(x) \in \mathbb{R}$ and $\sum_{k=1}^{K} \hat{f}_k(x) = 1$ for any x.

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k \in \mathcal{G}} \hat{f}_k(x_*)$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Predictions of the training data X are

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

For a new input x_* (with 1 in the first entry), the prediction $\hat{f}(x_*) = (\hat{f}_1(x_*), \ldots, \hat{f}_K(x_*))$ is the discriminant function $\delta(x_*) = (\delta_1(x_*), \ldots, \delta_K(x_*))$ and is given by

$$\hat{f}(x_*) = x_*^T \hat{B}$$

where $\hat{f}_k(x) \in \mathbb{R}$ and $\sum_{k=1}^{K} \hat{f}_k(x) = 1$ for any x.

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k \in \mathcal{G}} \hat{f}_k(x_*)$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Predictions of the training data X are

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

For a new input x_* (with 1 in the first entry), the prediction $\hat{f}(x_*) = (\hat{f}_1(x_*), \ldots, \hat{f}_K(x_*))$ is the discriminant function $\delta(x_*) = (\delta_1(x_*), \ldots, \delta_K(x_*))$ and is given by

$$\hat{f}(x_*) = x_*^T \hat{B}$$

where $\hat{f}_k(x) \in \mathbb{R}$ and $\sum_{k=1}^{K} \hat{f}_k(x) = 1$ for any x.

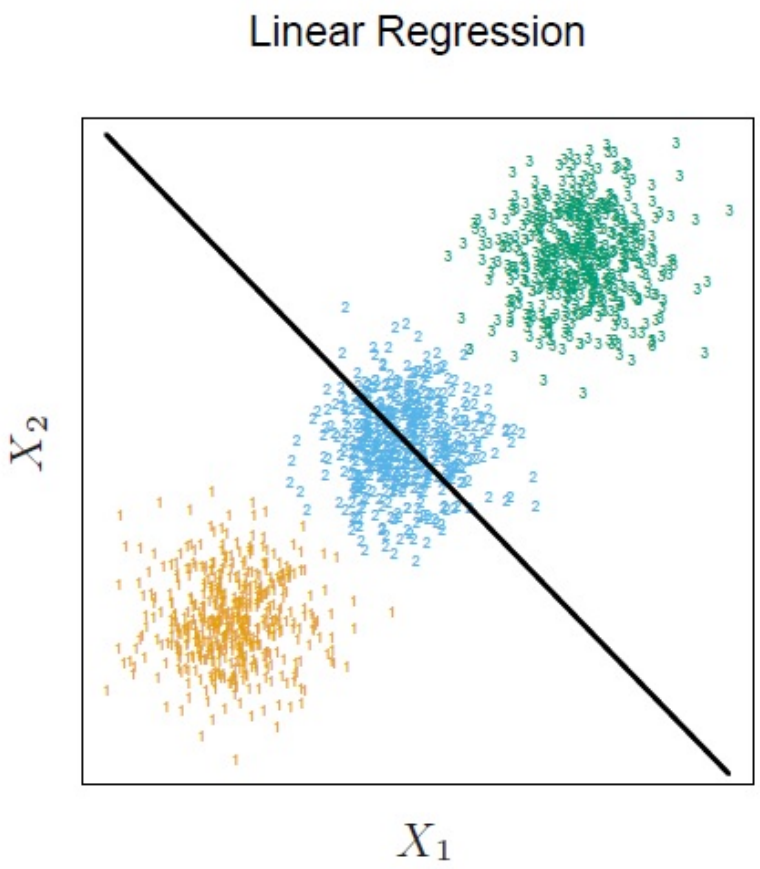
The optimal class k_* is

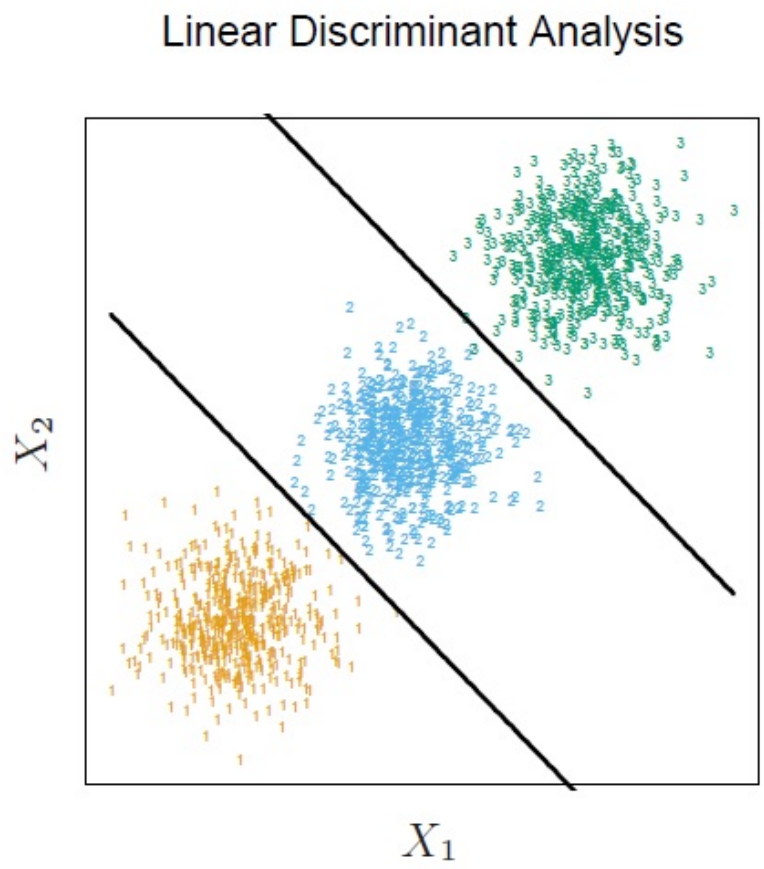
$$\hat{G}(x_*) = argmax_{k \in \mathcal{G}} \hat{f}_k(x_*)$$

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

But that method has problems when $K \geq 3$. For example,





Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Still MLR is a valid method for classification, but where is the probability in it?

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\,\hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\,||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\,\hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\,||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\,\hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\,||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\,\hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\,||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\, \hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\, ||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

Probability-based Classification Methods - Partitioning Methods

Multiple Linear Regression

Note that the regression function

$$E(I(G = k)|X = x) = P(G = k|X = x)$$

is the discriminant function $\delta_k(x)$ that is modeled by

$$f_k(x) = \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p = x^T\beta_k$$

and is estimated by $\hat{f}_k(x)$, while it could be outside the interval $[0, 1]$. (A drawback.)

The optimal class k_* is

$$\hat{G}(x_*) = argmax_{k\in\mathcal{G}}\, \hat{f}_k(x_*) = argmin_{k\in\mathcal{G}}\, ||\hat{f}(x_*) - t_k^T||$$

where the target vector t_k is a vector of 1 in the k^{th} location and zeros otherwise.

A Preamble to Generalized Linear Regression

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

Generalized Linear Regression, takes the form

$$g(E(Y|X = x)) = x^T\beta$$

- $g(\mu)$ is the **link** function in the mean $\mu = E(Y|X = x)$ of an exponential family distribution of $[Y|X = x]$.
- If $[Y|X = x] \sim N(\mu, \sigma^2)$, which is an exponential family with mean μ.
- Then g is the **identity** link function $g(\mu) = \mu$. So the generalized linear regression is nothing but multiple linear regression:

$$g(E(Y|X = x)) = E(Y|X = x) = x^T\beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

Generalized Linear Regression, takes the form

$$g(E(Y|X = x)) = x^T\beta$$

- $g(\mu)$ is the **link** function in the mean $\mu = E(Y|X = x)$ of an exponential family distribution of $[Y|X = x]$.
- If $[Y|X = x] \sim N(\mu, \sigma^2)$, which is an exponential family with mean μ.
- Then g is the **identity** link function $g(\mu) = \mu$. So the generalized linear regression is nothing but multiple linear regression:

$$g(E(Y|X = x)) = E(Y|X = x) = x^T\beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

Generalized Linear Regression, takes the form

$$g(E(Y|X = x)) = x^T\beta$$

- $g(\mu)$ is the **link** function in the mean $\mu = E(Y|X = x)$ of an exponential family distribution of $[Y|X = x]$.
- If $[Y|X = x] \sim N(\mu, \sigma^2)$, which is an exponential family with mean μ.
- Then g is the **identity** link function $g(\mu) = \mu$. So the generalized linear regression is nothing but multiple linear regression:

$$g(E(Y|X = x)) = E(Y|X = x) = x^T\beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Poisson}(\lambda)$, which is an exponential family with mean $\mu = \lambda$.
- Then g is the **log**arithmic link function $g(\lambda) = \log(\lambda)$. So the generalized linear regression is the log-linear (Poisson) regression:

$$g(E(Y|X = x)) = \log(E(Y|X = x)) = x^T\beta$$

or

$$E(Y|X = x) = e^{x^T\beta}$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Poisson}(\lambda)$, which is an exponential family with mean $\mu = \lambda$.
- Then g is the **log**arithmic link function $g(\lambda) = \log(\lambda)$. So the generalized linear regression is the log-linear (Poisson) regression:

$$g(E(Y|X = x)) = \log(E(Y|X = x)) = x^T\beta$$

or

$$E(Y|X = x) = e^{x^T\beta}$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Poisson}(\lambda)$, which is an exponential family with mean $\mu = \lambda$.
- Then g is the **log**arithmic link function $g(\lambda) = \log(\lambda)$. So the generalized linear regression is the log-linear (Poisson) regression:

$$g(E(Y|X = x)) = \log(E(Y|X = x)) = x^T\beta$$

or

$$E(Y|X = x) = e^{x^T\beta}$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Poisson}(\lambda)$, which is an exponential family with mean $\mu = \lambda$.
- Then g is the **log**arithmic link function $g(\lambda) = \log(\lambda)$. So the generalized linear regression is the log-linear (Poisson) regression:

$$g(E(Y|X = x)) = \log(E(Y|X = x)) = x^T\beta$$

or

$$E(Y|X = x) = e^{x^T\beta}$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Bernoulli}(p) \equiv \text{Binomial}(1, p)$, which is an exponential family with mean $\mu = p$.
- That is $E(Y|X = x) = P(Y = 1|X = x) = 1 - P(Y = 0|X = x)$.
- Then g is the **logit** link function $g(p) = \text{logit}(p) := \log(\frac{p}{1-p})$. So the generalized linear regression is the logistic (Binomial) regression:

$$\text{logit}(E(Y|X = x)) = x^T \beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Bernoulli}(p) \equiv \text{Binomial}(1, p)$, which is an exponential family with mean $\mu = p$.
- That is
  $E(Y|X = x) = P(Y = 1|X = x) = 1 - P(Y = 0|X = x)$.
- Then g is the **logit** link function $g(p) = \text{logit}(p) := \log(\frac{p}{1-p})$. So the generalized linear regression is the logistic (Binomial) regression:

$$\text{logit}(E(Y|X = x)) = x^T \beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

- If $[Y|X = x] \sim \text{Bernoulli}(p) \equiv \text{Binomial}(1, p)$, which is an exponential family with mean $\mu = p$.
- That is $E(Y|X = x) = P(Y = 1|X = x) = 1 - P(Y = 0|X = x)$.
- Then g is the **logit** link function $g(p) = \text{logit}(p) := \log(\frac{p}{1-p})$. So the generalized linear regression is the logistic (Binomial) regression:

$$\text{logit}(E(Y|X = x)) = x^T \beta$$

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

which is

$$\text{logit}(P(Y = 1|X = x)) = x^T\beta$$

and is

$$\log\left(\frac{P(Y = 1|X = x)}{P(Y = 0|X = x)}\right) = x^T\beta$$

or equivalently

$$P(Y = 1|X = x) = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}}$$

and

$$P(Y = 0|X = x) = \frac{1}{1 + e^{x^T\beta}}$$

In terms of groups, $G = 1$ when $Y = 1$ and $G = 2$ when $Y = 0$.

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

which is

$$\text{logit}(P(Y = 1|X = x)) = x^T\beta$$

and is

$$\log\left(\frac{P(Y = 1|X = x)}{P(Y = 0|X = x)}\right) = x^T\beta$$

or equivalently

$$P(Y = 1|X = x) = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}}$$

and

$$P(Y = 0|X = x) = \frac{1}{1 + e^{x^T\beta}}$$

In terms of groups, $G = 1$ when $Y = 1$ and $G = 2$ when $Y = 0$.

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

which is

$$\text{logit}(P(Y = 1|X = x)) = x^T\beta$$

and is

$$\log\left(\frac{P(Y = 1|X = x)}{P(Y = 0|X = x)}\right) = x^T\beta$$

or equivalently

$$P(Y = 1|X = x) = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}}$$

and

$$P(Y = 0|X = x) = \frac{1}{1 + e^{x^T\beta}}$$

In terms of groups, $G = 1$ when $Y = 1$ and $G = 2$ when $Y = 0$.

Probability-based Classification Methods - Partitioning Methods

A Preamble to Generalized Linear Regression

which is

$$\text{logit}(P(Y = 1|X = x)) = x^T\beta$$

and is

$$\log\left(\frac{P(Y = 1|X = x)}{P(Y = 0|X = x)}\right) = x^T\beta$$

or equivalently

$$P(Y = 1|X = x) = \frac{e^{x^T\beta}}{1 + e^{x^T\beta}}$$

and

$$P(Y = 0|X = x) = \frac{1}{1 + e^{x^T\beta}}$$

In terms of groups, $G = 1$ when $Y = 1$ and $G = 2$ when $Y = 0$.

Multinomial ($K-$ response or softmax) Logistic Regression

Probability-based Classification Methods - Partitioning Methods

Multinomial distribution

- Recall: $(Y_1, Y_2, \ldots, Y_K) \sim$multinomial$(n, p_1, p_2, \ldots, p_K)$ has a probability function given by

$$P(Y_1 = y_1, Y_2 = y_2, \ldots, Y_K = y_K) = \frac{n!}{y_1!\, y_2! \cdots y_K!}\, p_1^{y_1} p_2^{y_2} \cdots p_K^{y_K}$$

for $y_k = 0, 1, \ldots, n$, and $k = 1, 2, \ldots, K$; such that $\sum_{k=1}^{K} p_k = 1$ and $\sum_{k=1}^{K} y_k = n$.

- The mean of Y_k is $E(Y_k) = np_k$,
the variance is $Var(Y_k) = np_k(1 - p_k)$,
and the covariance is $COV(Y_k, Y_\ell) = -np_k p_\ell$ for $k \neq \ell$.

- If $n = 1$, then all of the (indicators) y_k's are equal to zero except one and only one of them that must be equal to 1.

Probability-based Classification Methods - Partitioning Methods

Multinomial distribution

- Recall: $(Y_1, Y_2, \ldots, Y_K) \sim$multinomial$(n, p_1, p_2, \ldots, p_K)$ has a probability function given by

$$P(Y_1 = y_1, Y_2 = y_2, \ldots, Y_K = y_K) = \frac{n!}{y_1!\, y_2! \cdots y_K!}\, p_1^{y_1} p_2^{y_2} \cdots p_K^{y_K}$$

  for $y_k = 0, 1, \ldots, n$, and $k = 1, 2, \ldots, K$; such that $\sum_{k=1}^{K} p_k = 1$ and $\sum_{k=1}^{K} y_k = n$.

- The mean of Y_k is $E(Y_k) = np_k$,
  the variance is $Var(Y_k) = np_k(1 - p_k)$,
  and the covariance is $COV(Y_k, Y_\ell) = -np_k p_\ell$ for $k \neq \ell$.

- If $n = 1$, then all of the (indicators) y_k's are equal to zero except one and only one of them that must be equal to 1.

Probability-based Classification Methods - Partitioning Methods

Multinomial distribution

- Recall: $(Y_1, Y_2, \ldots, Y_K) \sim$multinomial$(n, p_1, p_2, \ldots, p_K)$ has a probability function given by

$$P(Y_1 = y_1, Y_2 = y_2, \ldots, Y_K = y_K) = \frac{n!}{y_1!\, y_2! \cdots y_K!}\, p_1^{y_1} p_2^{y_2} \cdots p_K^{y_K}$$

  for $y_k = 0, 1, \ldots, n$, and $k = 1, 2, \ldots, K$; such that $\sum_{k=1}^{K} p_k = 1$ and $\sum_{k=1}^{K} y_k = n$.

- The mean of Y_k is $E(Y_k) = np_k$,
  the variance is $Var(Y_k) = np_k(1 - p_k)$,
  and the covariance is $COV(Y_k, Y_\ell) = -np_k p_\ell$ for $k \neq \ell$.

- If $n = 1$, then all of the (indicators) y_k's are equal to zero except one and only one of them that must be equal to 1.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The model of the (softmax) logistic regression:

$$\log\left(\frac{P(G_i = \ell | X = x_i)}{P(G_i = K | X = x_i)}\right) = x_i^T \beta_\ell$$

for $\ell = 1, 2, \ldots, K-1$ and $i = 1, 2, \ldots, N$. Thus,

$$p_{i,\ell} := P(G_i = \ell | X = x_i) = \frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $\ell = 1, 2, \ldots, K-1$ and

$$p_{i,K} := P(G_i = K | X = x_i) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $i = 1, 2, \ldots, N$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The model of the (softmax) logistic regression:

$$\log\left(\frac{P(G_i = \ell | X = x_i)}{P(G_i = K | X = x_i)}\right) = x_i^T \beta_\ell$$

for $\ell = 1, 2, \ldots, K-1$ and $i = 1, 2, \ldots, N$. Thus,

$$p_{i,\ell} := P(G_i = \ell | X = x_i) = \frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $\ell = 1, 2, \ldots, K-1$ and

$$p_{i,K} := P(G_i = K | X = x_i) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $i = 1, 2, \ldots, N$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The model of the (softmax) logistic regression:

$$\log\left(\frac{P(G_i = \ell | X = x_i)}{P(G_i = K | X = x_i)}\right) = x_i^T \beta_\ell$$

for $\ell = 1, 2, \ldots, K-1$ and $i = 1, 2, \ldots, N$. Thus,

$$p_{i,\ell} := P(G_i = \ell | X = x_i) = \frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $\ell = 1, 2, \ldots, K-1$ and

$$p_{i,K} := P(G_i = K | X = x_i) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}}$$

for $i = 1, 2, \ldots, N$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Thus, the
$[Y_1, \ldots, Y_K | X = x_i] \sim$multinomial$(1, p_{i,1}, p_{i,2}, \ldots, p_{i,K})$'s
likelihood function of the training data is

$$L(\beta) = \prod_{i=1}^{N} \frac{1}{y_{i,1}!\, y_{i,2}! \cdots y_{i,K}!} p_{i,1}^{y_{i,1}} p_{i,2}^{y_{i,2}} \cdots p_{i,K}^{y_{i,K}}$$

$$= \prod_{i=1}^{N} \prod_{\ell=1}^{K-1} \left(\frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}} \right)^{y_{i,\ell}} \left(\frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}} \right)^{1 - \sum_{\ell=1}^{K-1} y_{i,\ell}}$$

$$= \prod_{i=1}^{N} \frac{e^{x_i^T \sum_{\ell=1}^{K-1} y_{i,\ell} \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}}$$

which is maximized at the maximum likelihood estimator (MLE) $\hat{\beta}$ which also is the maximum of the log-likelihood function $\ell(\beta) = \log(L(\beta))$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Thus, the
$[Y_1, \ldots, Y_K | X = x_i] \sim$multinomial$(1, p_{i,1}, p_{i,2}, \ldots, p_{i,K})$'s
likelihood function of the training data is

$$L(\beta) = \prod_{i=1}^{N} \frac{1}{y_{i,1}!\, y_{i,2}! \cdots y_{i,K}!} p_{i,1}^{y_{i,1}} p_{i,2}^{y_{i,2}} \cdots p_{i,K}^{y_{i,K}}$$

$$= \prod_{i=1}^{N} \prod_{\ell=1}^{K-1} \left(\frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}} \right)^{y_{i,\ell}} \left(\frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}} \right)^{1 - \sum_{\ell=1}^{K-1} y_{i,\ell}}$$

$$= \prod_{i=1}^{N} \frac{e^{x_i^T \sum_{\ell=1}^{K-1} y_{i,\ell} \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}}$$

which is maximized at the maximum likelihood estimator (MLE) $\hat{\beta}$ which also is the maximum of the log-likelihood function $\ell(\beta) = \log(L(\beta))$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Thus, the
$[Y_1, \ldots, Y_K | X = x_i] \sim$multinomial$(1, p_{i,1}, p_{i,2}, \ldots, p_{i,K})$'s
likelihood function of the training data is

$$L(\beta) = \prod_{i=1}^{N} \frac{1}{y_{i,1}!\, y_{i,2}! \cdots y_{i,K}!} p_{i,1}^{y_{i,1}} p_{i,2}^{y_{i,2}} \cdots p_{i,K}^{y_{i,K}}$$

$$= \prod_{i=1}^{N} \prod_{\ell=1}^{K-1} \left(\frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}} \right)^{y_{i,\ell}} \left(\frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}} \right)^{1 - \sum_{\ell=1}^{K-1} y_{i,\ell}}$$

$$= \prod_{i=1}^{N} \frac{e^{x_i^T \sum_{\ell=1}^{K-1} y_{i,\ell} \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}}$$

which is maximized at the maximum likelihood estimator (MLE) $\hat{\beta}$ which also is the maximum of the log-likelihood function $\ell(\beta) = \log(L(\beta))$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Thus, the
$[Y_1, \ldots, Y_K | X = x_i] \sim$multinomial$(1, p_{i,1}, p_{i,2}, \ldots, p_{i,K})$'s
likelihood function of the training data is

$$L(\beta) = \prod_{i=1}^{N} \frac{1}{y_{i,1}!\, y_{i,2}! \cdots y_{i,K}!} p_{i,1}^{y_{i,1}} p_{i,2}^{y_{i,2}} \cdots p_{i,K}^{y_{i,K}}$$

$$= \prod_{i=1}^{N} \prod_{\ell=1}^{K-1} \left(\frac{e^{x_i^T \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_k}} \right)^{y_{i,\ell}} \left(\frac{1}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}} \right)^{1 - \sum_{\ell=1}^{K-1} y_{i,\ell}}$$

$$= \prod_{i=1}^{N} \frac{e^{x_i^T \sum_{\ell=1}^{K-1} y_{i,\ell} \beta_\ell}}{1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}}$$

which is maximized at the maximum likelihood estimator (MLE)
$\hat{\beta}$ which also is the maximum of the log-likelihood function
$\ell(\beta) = \log(L(\beta))$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The log-likelihood function

$$\ell(\beta) = \sum_{i=1}^{N} \left[x_i^T \left(\sum_{\ell=1}^{K-1} y_{i,\ell}\, \beta_\ell \right) - \log\left(1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}\right) \right]$$

For brevity, let $K = 2$ and $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$. The log-likelihood function

$$\ell(\beta) = \sum_{i=1}^{N} \left[y_i\, x_i^T \beta - \log(1 + e^{x_i^T \beta}) \right]$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The log-likelihood function

$$\ell(\beta) = \sum_{i=1}^{N} \left[x_i^T (\sum_{\ell=1}^{K-1} y_{i,\ell}\, \beta_\ell) - \log(1 + \sum_{k=1}^{K-1} e^{x_i^T \beta_K}) \right]$$

For brevity, let $K = 2$ and $\beta = (\beta_0, \beta_1, \ldots, \beta_p)^T$. The log-likelihood function

$$\ell(\beta) = \sum_{i=1}^{N} \left[y_i\, x_i^T \beta - \log(1 + e^{x_i^T \beta}) \right]$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Thus, the score function

$$\begin{aligned}\frac{\partial \ell(\beta)}{\partial \beta} &= \sum_{i=1}^{N}(y_i - p_i)x_i^T \\ &= X^T(y - p)\end{aligned}$$

when set equal to zero it gives a system of $p + 1$ nonlinear equations in $\beta_0, \beta_1, \ldots, \beta_p$, the first of which is $\sum_{i=1}^{N} y_i = \sum_{i=1}^{N} p_i$.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

In addition, the Hessian matrix

$$\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} = -\sum_{i=1}^{N} p_i(1-p_i)x_i x_i^T = -X^T W X$$

where W is a diagonal matrix whose i^{th} diagonal entry is $p_i(1-p_i)$. The information matrix is

$$-\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T}\bigg|_{\hat{\beta}} = X^T \widehat{W} X$$

where $\widehat{W}$ is the matrix W calculated at $\hat{\beta}$.

Based on asymptotic theorem $\hat{\beta} \sim MN_{p+1}(\beta, \frac{1}{N}(X^T \widehat{W} X)^{-1})$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

In addition, the Hessian matrix

$$\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} = -\sum_{i=1}^{N} p_i(1-p_i)x_i x_i^T = -X^T W X$$

where W is a diagonal matrix whose i^{th} diagonal entry is $p_i(1-p_i)$. The information matrix is

$$-\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T}\bigg|_{\hat{\beta}} = X^T \widehat{W} X$$

where $\widehat{W}$ is the matrix W calculated at $\hat{\beta}$.

Based on asymptotic theorem $\hat{\beta} \sim MN_{p+1}(\beta, \frac{1}{N}(X^T \widehat{W} X)^{-1})$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\ &= \beta^{old} + (X^T W X)^{-1} X^T (y - p)\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\ &= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\ &= \beta^{old} + (X^T W X)^{-1} X^T (y - p)\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T (y - p)
\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p)
\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p) \\
&= (X^T W X)^{-1} X^T W (X \beta^{old} + W^{-1} (y - p))
\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p) \\
&= (X^T W X)^{-1} X^T W \underbrace{(X \beta^{old} + W^{-1}(y - p))}_{z}
\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p) \\
&= (X^T W X)^{-1} X^T W \underbrace{(X \beta^{old} + W^{-1}(y - p))}_{z \text{ is called adjusted response}}
\end{aligned}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$
\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p) \\
&= (X^T W X)^{-1} X^T W \underbrace{(X \beta^{old} + W^{-1}(y - p))}_{z \text{ is called adjusted response}} \\
&= (X^T W X)^{-1} X^T W z
\end{aligned}
$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

The MLE $\hat{\beta}$ is found iteratively using Newton's algorithm

$$\begin{aligned}
\beta^{new} &= \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} \Big|_{\beta^{old}} \right)^{-1} \cdot \frac{\partial \ell(\beta)}{\partial \beta} \Big|_{\beta^{old}} \\
&= \beta^{old} + (X^T W X)^{-1} X^T (y - p) \\
&= (X^T W X)^{-1} (X^T W X) \beta^{old} + (X^T W X)^{-1} X^T W W^{-1} (y - p) \\
&= (X^T W X)^{-1} X^T W \underbrace{(X \beta^{old} + W^{-1}(y - p))}_{z \text{ is called adjusted response}} \\
&= (X^T W X)^{-1} X^T W z
\end{aligned}$$

which is referred to iterative re-weighted least squares (IRLS). Each iteration is nothing but the solution of weighted least squares

$$\beta^{new} = argmin_\beta (z - X\beta)^T W (z - X\beta).$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

First,

	Coefficient	Std. Error	Z Score
(Intercept)	−4.130	0.964	−4.285
sbp	0.006	0.006	1.023
tobacco	0.080	0.026	3.034
ldl	0.185	0.057	3.219
famhist	0.939	0.225	4.178
obesity	-0.035	0.029	−1.187
alcohol	0.001	0.004	0.136
age	0.043	0.010	4.184

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

Finally,

	Coefficient	Std. Error	Z score
(Intercept)	-4.204	0.498	-8.45
tobacco	0.081	0.026	3.16
ldl	0.168	0.054	3.09
famhist	0.924	0.223	4.14
age	0.044	0.010	4.52

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. make a scatter plot of the data for one or two inputs.
2. Use glm to fit a multiple linear regression (MLR) to the data.
3. Plot the best fit of the MLR to the data against those one or two, probably significant, inputs; e.g, age and tobacco. What do you see?
4. Check out the assumptions by performing residual analyses. What do you see?
5. Use glm and/or glmnet to fit a logistic regression (Log-R) to the data.
6. Plot the best fit of the Log-R to the data against those one or two, probably significant, inputs. What do you see?

Logistic Regression - Inference

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - CI

Since by asymptotic theorem

$$\hat{\beta} \sim MN_{p+1}(\beta, \widehat{V})$$

where $\widehat{V} = \frac{1}{N}(X^T \widehat{W} X)^{-1}$ then

$$c^T \hat{\beta} \sim N(c^T \beta, c^T \widehat{V} c)$$

Thus a $(1-\alpha)100\%$ CI for the log odds-ratio $\text{logit}(P(Y=1|X=x))$ is

$$x^T \hat{\beta} \pm z_{\alpha/2} \sqrt{x^T \widehat{V} x}$$

and a $(1-\alpha)100\%$ CI for the probability $P(Y=1|X=x)$ is

$$\left(\frac{exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}, \frac{exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})} \right)$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - CI

Since by asymptotic theorem

$$\hat{\beta} \sim MN_{p+1}(\beta, \widehat{V})$$

where $\widehat{V} = \frac{1}{N}(X^T \widehat{W} X)^{-1}$ then

$$c^T \hat{\beta} \sim N(c^T \beta, c^T \widehat{V} c)$$

Thus a $(1-\alpha)100\%$ CI for the log odds-ratio $\text{logit}(P(Y=1|X=x))$ is

$$x^T \hat{\beta} \pm z_{\alpha/2} \sqrt{x^T \widehat{V} x}$$

and a $(1-\alpha)100\%$ CI for the probability $P(Y=1|X=x)$ is

$$\left(\frac{exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}, \frac{exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})} \right)$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - CI

Since by asymptotic theorem

$$\hat{\beta} \sim MN_{p+1}(\beta, \widehat{V})$$

where $\widehat{V} = \frac{1}{N}(X^T \widehat{W} X)^{-1}$ then

$$c^T \hat{\beta} \sim N(c^T \beta, c^T \widehat{V} c)$$

Thus a $(1-\alpha)100\%$ CI for the log odds-ratio $\text{logit}(P(Y=1|X=x))$ is

$$x^T \hat{\beta} \pm z_{\alpha/2} \sqrt{x^T \widehat{V} x}$$

and a $(1-\alpha)100\%$ CI for the probability $P(Y=1|X=x)$ is

$$\left(\frac{exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} - z_{\alpha/2} \sqrt{x^T \widehat{V} x})}, \frac{exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})}{1 + exp(x^T \hat{\beta} + z_{\alpha/2} \sqrt{x^T \widehat{V} x})} \right)$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Wald's test:

$$H_0 : A\beta = b \text{ versus } H_A : A\beta \neq b$$

where A is $q \times (p+1)$ matrix of rank $q \leq p+1$, and b is an $q \times 1$ column vector.

Using Wald's test statistics

$$W_0 = (A\hat{\beta} - b)^T (A(X^T \widehat{W} X)^{-1} A^T)^{-1} (A\hat{\beta} - b) \sim \chi^2_q$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Wald's test:

$$H_0 : A\beta = b \text{ versus } H_A : A\beta \neq b$$

where A is $q \times (p+1)$ matrix of rank $q \leq p+1$, and b is an $q \times 1$ column vector.

Using Wald's test statistics

$$W_0 = (A\hat{\beta} - b)^T (A(X^T \widehat{W} X)^{-1} A^T)^{-1} (A\hat{\beta} - b) \sim \chi^2_q$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Wald's test:

$$H_0 : A\beta = b \text{ versus } H_A : A\beta \neq b$$

where A is $q \times (p+1)$ matrix of rank $q \leq p+1$, and b is an $q \times 1$ column vector.

Using Wald's test statistics

$$W_0 = (A\hat{\beta} - b)^T (A(X^T \widehat{W} X)^{-1} A^T)^{-1} (A\hat{\beta} - b) \sim \chi^2_q$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Likelihood-ratio test: (More powerful)

$$H_0 : D\beta = 0 \text{ versus } H_A : D\beta \neq 0$$

where D is $(p+1) \times (p+1)$ diagonal matrix of ones and zeros to select the parameters to test being equal to zero.

Using likelihood-ratio test statistics

$$LR_0 = -2\ell(\hat{\beta}_{\text{restricted}}) + 2\ell(\hat{\beta}_{\text{full}}) \sim \chi^2_{tr(D)}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Likelihood-ratio test: (More powerful)

$$H_0 : D\beta = 0 \text{ versus } H_A : D\beta \neq 0$$

where D is $(p+1) \times (p+1)$ diagonal matrix of ones and zeros to select the parameters to test being equal to zero.

Using likelihood-ratio test statistics

$$LR_0 = -2\ell(\hat{\beta}_{\text{restricted}}) + 2\ell(\hat{\beta}_{\text{full}}) \sim \chi^2_{tr(D)}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Inference - Test of Hypothesis

- Likelihood-ratio test: (More powerful)

$$H_0 : D\beta = 0 \text{ versus } H_A : D\beta \neq 0$$

where D is $(p+1) \times (p+1)$ diagonal matrix of ones and zeros to select the parameters to test being equal to zero.

Using likelihood-ratio test statistics

$$LR_0 = -2\ell(\hat{\beta}_{\text{restricted}}) + 2\ell(\hat{\beta}_{\text{full}}) \sim \chi^2_{tr(D)}$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Carry out the following test of hypotheses

$$H_0 : \beta_1 \text{ and } \beta_3 = 0 \textit{ versus } H_A : \text{either } \beta_1 \text{ or } \beta_3 \neq 0$$

2. Carry out the following test of hypotheses

$$H_0 : \beta_1 \text{ and } \beta_3 = 0 \text{ and } \beta_2 + \beta_4 = 2\beta_5$$

$$\textit{versus}$$

$$H_A : \text{either } \beta_1 \neq 0 \text{ or } \beta_3 \neq 0 \text{ or } \beta_2 + \beta_4 \neq 2\beta_5$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Carry out the following test of hypotheses

$$H_0 : \beta_1 \text{ and } \beta_3 = 0 \textit{ versus } H_A : \text{either } \beta_1 \text{ or } \beta_3 \neq 0$$

2. Carry out the following test of hypotheses

$$H_0 : \beta_1 \text{ and } \beta_3 = 0 \text{ and } \beta_2 + \beta_4 = 2\beta_5$$

versus

$$H_A : \text{either } \beta_1 \neq 0 \text{ or } \beta_3 \neq 0 \text{ or } \beta_2 + \beta_4 \neq 2\beta_5$$

Logistic Regression - Prediction

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Prediction

For prediction at x_*, the probability

$$\delta_1(x_*) = P(\hat{Y} = 1|X = x_*) = \frac{exp(x_*^T \hat{\beta})}{1 + exp(x_*^T \hat{\beta})}$$

with a $(1 - \alpha)100\%$ CI for the probability $P(Y = 1|X = x_*)$ is

$$\left(\frac{exp(x_*^T \hat{\beta} - z_{\alpha/2}\sqrt{x_*^T \widehat{V} x_*})}{1 + exp(x_*^T \hat{\beta} - z_{\alpha/2}\sqrt{x_*^T \widehat{V} x_*})}, \frac{exp(x_*^T \hat{\beta} + z_{\alpha/2}\sqrt{x_*^T \widehat{V} x_*})}{1 + exp(x_*^T \hat{\beta} + z_{\alpha/2}\sqrt{x_*^T \widehat{V} x_*})} \right)$$

Probability-based Classification Methods - Partitioning Methods

Logistic Regression - Prediction

How to choose the decision boundary?

- If $\delta_1(x_*) \geq .5$, then $\hat{G}(x_*) = 1$, otherwise $\hat{G}(x_*) = 2$.

  OR

- Use a cut-off point other than .5, that minimizes the mis-classification error in cross-validation or in the whole training data.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Make some predictions using a cutoff at .5.
2. Try to find a better cutoff point.
3. Use that better cutoff point for predictions.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Make some predictions using a cutoff at .5.
2. Try to find a better cutoff point.
3. Use that better cutoff point for predictions.

Probability-based Classification Methods - Partitioning Methods

Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Make some predictions using a cutoff at .5.
2. Try to find a better cutoff point.
3. Use that better cutoff point for predictions.

L_1 - Regularized Logistic Regression

Probability-based Classification Methods - Partitioning Methods

L_1-Regularized Logistic Regression

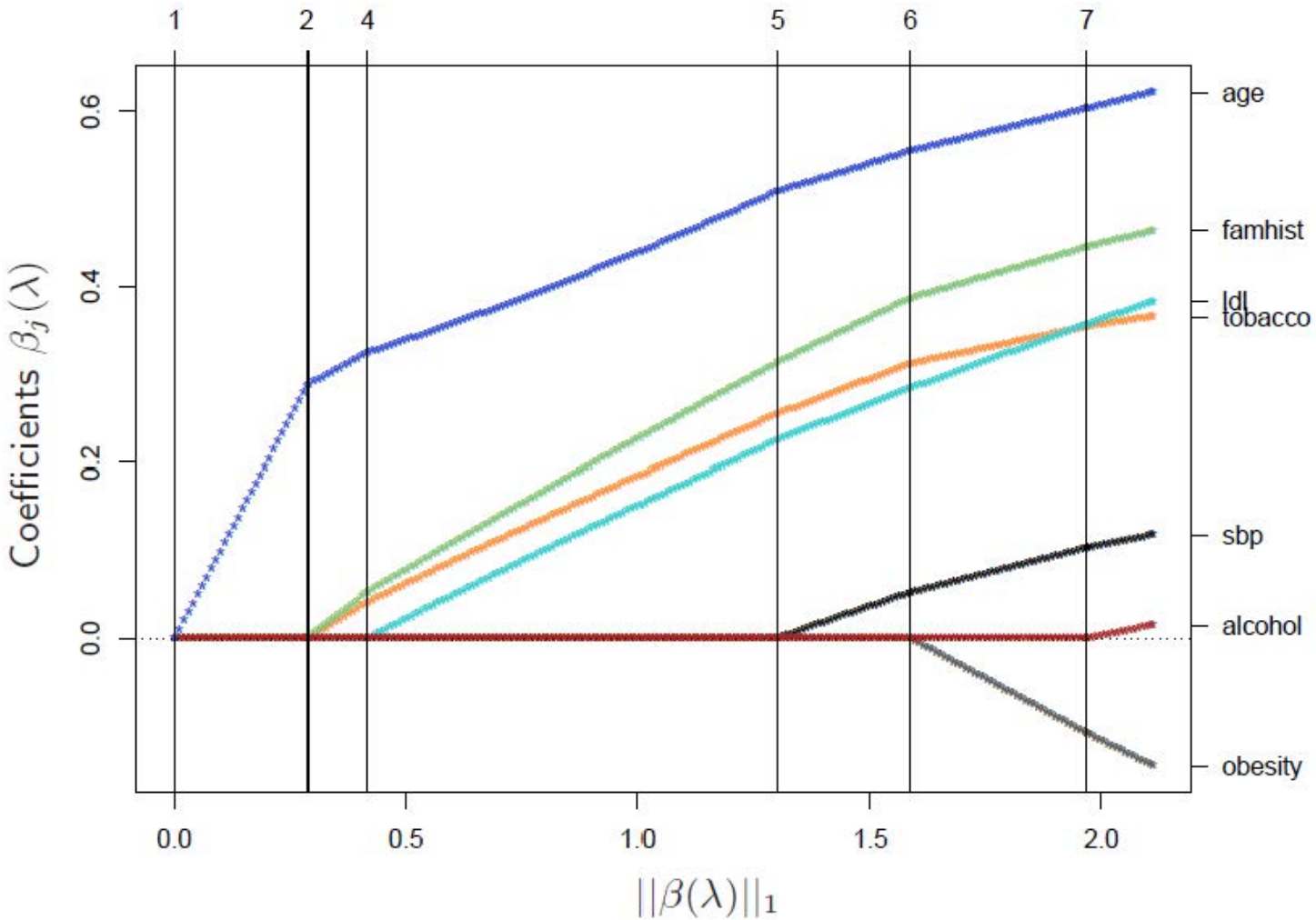
The idea is to shrink and select the inputs using standardized training data and so

$$\hat{\beta}^{L_1} = argmax_\beta \sum_{i=1}^{N} \left[y_i \, x_i^T \beta - \log(1 + e^{x_i^T \beta}) \right] - \lambda \sum_{j=1}^{p} |\beta_j|$$

Probability-based Classification Methods - Partitioning Methods

L_1-Regularized Logistic Regression

Example (SA Heart Disease)



Probability-based Classification Methods - Partitioning Methods

L_1-Regularized Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Carry out a L_1-Regularized Logistic Regression using *glmnet*.
2. Carry out the elastic-net Logistic regression using *glmnet*.

Probability-based Classification Methods - Partitioning Methods

L_1-Regularized Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Carry out a L_1-Regularized Logistic Regression using *glmnet*.
2. Carry out the elastic-net Logistic regression using *glmnet*.

Probability-based Classification Methods - Partitioning Methods

L_1-Regularized Logistic Regression

Example (SA Heart Disease)

DIY in R

1. Carry out a L_1-Regularized Logistic Regression using *glmnet*.
2. Carry out the elastic-net Logistic regression using *glmnet*.

Please study the different feature of the *glmnet* from `https://glmnet.stanford.edu/articles/glmnet.html`

Bayes-based Classification Methods

Bayes-based Classification Methods - Partitioning Methods

By Bayes theorem,

$$
\begin{aligned}
P(G = k|X = x) &= \frac{P(X = x|G = k)P(G = k)}{P(X = x)} \\
&= \frac{P(X = x|G = k)P(G = k)}{\sum_{\ell=1}^{K} P(X = x|G = \ell)P(G = \ell)} \\
&= \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell}
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

By Bayes theorem,

$$P(G = k|X = x) = \frac{P(X = x|G = k)P(G = k)}{P(X = x)}$$

$$= \frac{P(X = x|G = k)P(G = k)}{\sum_{\ell=1}^{K} P(X = x|G = \ell)P(G = \ell)}$$

$$\text{(even for continuous r.v. } X) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell}$$

Bayes-based Classification Methods - Partitioning Methods

By Bayes theorem,

$$P(G = k|X = x) = \frac{P(X = x|G = k)P(G = k)}{P(X = x)}$$

$$= \frac{P(X = x|G = k)P(G = k)}{\sum_{\ell=1}^{K} P(X = x|G = \ell)P(G = \ell)}$$

$$\text{(even for continuous r.v. } X) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell} \quad \text{for } x \in \mathbb{R}^p$$

Bayes-based Classification Methods - Partitioning Methods

By Bayes theorem,

$$P(G = k|X = x) = \frac{P(X = x|G = k)P(G = k)}{P(X = x)}$$

$$= \frac{P(X = x|G = k)P(G = k)}{\sum_{\ell=1}^{K} P(X = x|G = \ell)P(G = \ell)}$$

$$\text{(even for continuous r.v. } X) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell} \quad \text{for } x \in \mathbb{R}^p$$

where $\pi_k = P(G = k)$ are the prior probabilities such that $\sum_{k=1}^{K} \pi_k = 1$ and $f_k(x)$ is the density of X in class k.

Bayes-based Classification Methods - Partitioning Methods

By Bayes theorem,

$$P(G = k|X = x) = \frac{P(X = x|G = k)P(G = k)}{P(X = x)}$$

$$= \frac{P(X = x|G = k)P(G = k)}{\sum_{\ell=1}^{K} P(X = x|G = \ell)P(G = \ell)}$$

$$\text{(even for continuous r.v. } X) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell} \quad \text{for } x \in \mathbb{R}^p$$

where $\pi_k = P(G = k)$ are the prior probabilities such that $\sum_{k=1}^{K} \pi_k = 1$ and $f_k(x)$ is the density of X in class k.

We use the discriminant function

$$\delta_k(x) = \log(f_k(x)\pi_k)$$

after removing constants, and

$$G(x) = argmax_k \delta_k(x) = argmax_k \left[\log(f_k(x)) + \log(\pi_k)\right]$$

Bayes-based Classification Methods - Partitioning Methods

Therefore,

$$\frac{P(G = k|X = x)}{P(G = \ell|X = x)} = \frac{f_k(x)\pi_k}{f_\ell(x)\pi_\ell}$$

for any k and ℓ, and so

$$\log\left(\frac{P(G = k|X = x)}{P(G = \ell|X = x)}\right) = \log(\frac{\pi_k}{\pi_\ell}) + \log(\frac{f_k(x)}{f_\ell(x)})$$

Bayes-based Classification Methods - Partitioning Methods

The density f_k determines the method.

- Linear and quadratic discriminant analysis use normal/Gaussian distribution f_k
- Mixed discriminant analyses use a mixture of Gaussian distributions for f_k
- Naïve Bayes uses a product of probability distributions for f_k
- f_k could be any general non-parametric density

Bayes-based Classification Methods - Partitioning Methods

The density f_k determines the method.

- Linear and quadratic discriminant analysis use normal/Gaussian distribution f_k
- Mixed discriminant analyses use a mixture of Gaussian distributions for f_k
- Naïve Bayes uses a product of probability distributions for f_k
- f_k could be any general non-parametric density

Bayes-based Classification Methods - Partitioning Methods

The density f_k determines the method.

- Linear and quadratic discriminant analysis use normal/Gaussian distribution f_k
- Mixed discriminant analyses use a mixture of Gaussian distributions for f_k
- Naïve Bayes uses a product of probability distributions for f_k
- f_k could be any general non-parametric density

Bayes-based Classification Methods - Partitioning Methods

The density f_k determines the method.

- Linear and quadratic discriminant analysis use normal/Gaussian distribution f_k
- Mixed discriminant analyses use a mixture of Gaussian distributions for f_k
- Naïve Bayes uses a product of probability distributions for f_k
- f_k could be any general non-parametric density

Linear & Quadratic Discriminant Analyses

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The density f_k is assumed to be a multivariate normal (Gaussian) $MN_p(\mu_k, \Sigma_k)$, for $k = 1, 2, \ldots, K$. That is,

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}$$

for $x \in \mathbb{R}^p$.

1. If $\Sigma_k = \Sigma$ for all k, then it is linear discriminant analysis.
2. If Σ_k depends on k, then it is quadratic discriminant analysis.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The density f_k is assumed to be a multivariate normal (Gaussian) $MN_p(\mu_k, \Sigma_k)$, for $k = 1, 2, \ldots, K$. That is,

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)}$$

for $x \in \mathbb{R}^p$.

1. If $\Sigma_k = \Sigma$ for all k, then it is linear discriminant analysis.
2. If Σ_k depends on k, then it is quadratic discriminant analysis.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The density f_k is assumed to be a multivariate normal (Gaussian) $MN_p(\mu_k, \Sigma_k)$, for $k = 1, 2, \ldots, K$. That is,

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}$$

for $x \in \mathbb{R}^p$.

1. If $\Sigma_k = \Sigma$ for all k, then it is linear discriminant analysis.
2. If Σ_k depends on k, then it is quadratic discriminant analysis.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

For any k and ℓ,

$$\frac{f_k(x)}{f_\ell(x)} = \frac{\frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}}{\frac{1}{(2\pi)^{p/2}|\Sigma_\ell|^{1/2}} e^{-\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}}$$

$$= \frac{|\Sigma_\ell|^{1/2}}{|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)+\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}$$

and

$$exponent = -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

For any k and ℓ,

$$\frac{f_k(x)}{f_\ell(x)} = \frac{\frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}}{\frac{1}{(2\pi)^{p/2}|\Sigma_\ell|^{1/2}} e^{-\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}}$$

$$= \frac{|\Sigma_\ell|^{1/2}}{|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)+\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}$$

and

$$exponent = -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

For any k and ℓ,

$$\frac{f_k(x)}{f_\ell(x)} = \frac{\frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)}}{\frac{1}{(2\pi)^{p/2}|\Sigma_\ell|^{1/2}} e^{-\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}}$$

$$= \frac{|\Sigma_\ell|^{1/2}}{|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)+\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)}$$

and

$$exponent = -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell)$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$\begin{aligned} exponent = &-\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\ &+\frac{1}{2}(x-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_\ell) \end{aligned}$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$\begin{aligned} \textit{exponent} = & -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\ & + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \end{aligned}$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&\quad + \frac{1}{2}(\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) \\
&\quad + (x-\mu_k)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
\textit{exponent} &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&\quad + \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&\quad - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) \\
&\quad + x^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

$$
\begin{aligned}
exponent &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) \\
&+ \frac{1}{2}(x-\mu_k+\mu_k-\mu_\ell)^T\Sigma_\ell^{-1}(x-\mu_k+\mu_k-\mu_\ell) \\
&= \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) \\
&- \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) \\
&+ x^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Thus, in LDA

$$
\frac{f_k(x)}{f_\ell(x)} = e^{x^T\Sigma^{-1}(\mu_k-\mu_\ell)-\frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma^{-1}(\mu_k-\mu_\ell)}
$$

Bayes-based Classification Methods - Partitioning Methods

Linear discriminant analysis (LDA)

In linear discriminant analysis (LDA)...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = x^T\Sigma^{-1}(\mu_k-\mu_\ell) - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma^{-1}(\mu_k-\mu_\ell)+\log(\frac{\pi_k}{\pi_\ell})$$

for any k and ℓ. And the linear discriminant function is

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

and again

$$G(x) = argmax_k\delta_k(x)$$

Bayes-based Classification Methods - Partitioning Methods

Linear discriminant analysis (LDA)

In linear discriminant analysis (LDA)...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = x^T\Sigma^{-1}(\mu_k-\mu_\ell) - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma^{-1}(\mu_k-\mu_\ell) + \log(\frac{\pi_k}{\pi_\ell})$$

for any k and ℓ. And the linear discriminant function is

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

and again

$$G(x) = argmax_k\delta_k(x)$$

Bayes-based Classification Methods - Partitioning Methods

Linear discriminant analysis (LDA)

For $\ell = K$

$$\log\left(\frac{P(G=k|X=x)}{P(G=K|X=x)}\right) = a_k + \sum_{i=1}^{p} b_{k,i}x_i$$

for some a_k and $b_{k,i}$ functions in the priors and parameters.

What are the similarities and differences with logistic regression?

Bayes-based Classification Methods - Partitioning Methods

Linear discriminant analysis (LDA)

For $\ell = K$

$$\log\left(\frac{P(G=k|X=x)}{P(G=K|X=x)}\right) = a_k + \sum_{i=1}^{p} b_{k,i}x_i$$

for some a_k and $b_{k,i}$ functions in the priors and parameters.

What are the similarities and differences with logistic regression?

Bayes-based Classification Methods - Partitioning Methods

Quadratic discriminant analysis (QDA)

Whereas in quadratic discriminant analysis (QDA)...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) + x^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) + \frac{1}{2}\log(\frac{|\Sigma_\ell|}{|\Sigma_k|})$$

for any k and ℓ. And the quadratic discriminant function is

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

and again

$$G(x) = argmax_k\delta_k(x)$$

Bayes-based Classification Methods - Partitioning Methods

Quadratic discriminant analysis (QDA)

Whereas in quadratic discriminant analysis (QDA)...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = \frac{1}{2}(x-\mu_k)^T(\Sigma_\ell^{-1}-\Sigma_k^{-1})(x-\mu_k) + x^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma_\ell^{-1}(\mu_k-\mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) + \frac{1}{2}\log(\frac{|\Sigma_\ell|}{|\Sigma_k|})$$

for any k and ℓ. And the quadratic discriminant function is

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

and again

$$G(x) = argmax_k\delta_k(x)$$

Bayes-based Classification Methods - Partitioning Methods

Quadratic discriminant analysis (QDA)

For $\ell = K$

$$\log\left(\frac{P(G=k|X=x)}{P(G=K|X=x)}\right) = a_k + \sum_{i=1}^{p} b_{k,i}x_i + \sum_{j=1}^{p}\sum_{i=1}^{p} c_{k,i,j}x_i x_j$$

for some a_k, $b_{k,i}$, and $c_{k,i,j}$ functions in the priors and parameters.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

Generally speaking, the decision boundary $\mathcal{B}_{k,\ell}$ is given through $\delta_k = \delta_\ell$ Thus,

- In LDA,

$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : x^T \Sigma^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T \Sigma^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) = 0\}$$

- In QDA,

$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : \frac{1}{2}(x - \mu_k)^T(\Sigma_\ell^{-1} - \Sigma_k^{-1})(x - \mu_k) + x^T \Sigma_\ell^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T \Sigma_\ell^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) + \frac{1}{2}\log(\frac{|\Sigma_\ell|}{|\Sigma_k|}) = 0\}$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

Generally speaking, the decision boundary $\mathcal{B}_{k,\ell}$ is given through $\delta_k = \delta_\ell$ Thus,

- In LDA,

$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : x^T\Sigma^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T\Sigma^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) = 0\}$$

- In QDA,

$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : \frac{1}{2}(x - \mu_k)^T(\Sigma_\ell^{-1} - \Sigma_k^{-1})(x - \mu_k) + x^T\Sigma_\ell^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T\Sigma_\ell^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) + \frac{1}{2}\log(\frac{|\Sigma_\ell|}{|\Sigma_k|}) = 0\}$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

Generally speaking, the decision boundary $\mathcal{B}_{k,\ell}$ is given through $\delta_k = \delta_\ell$ Thus,

- In LDA,

$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : x^T\Sigma^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T\Sigma^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) = 0\}$$

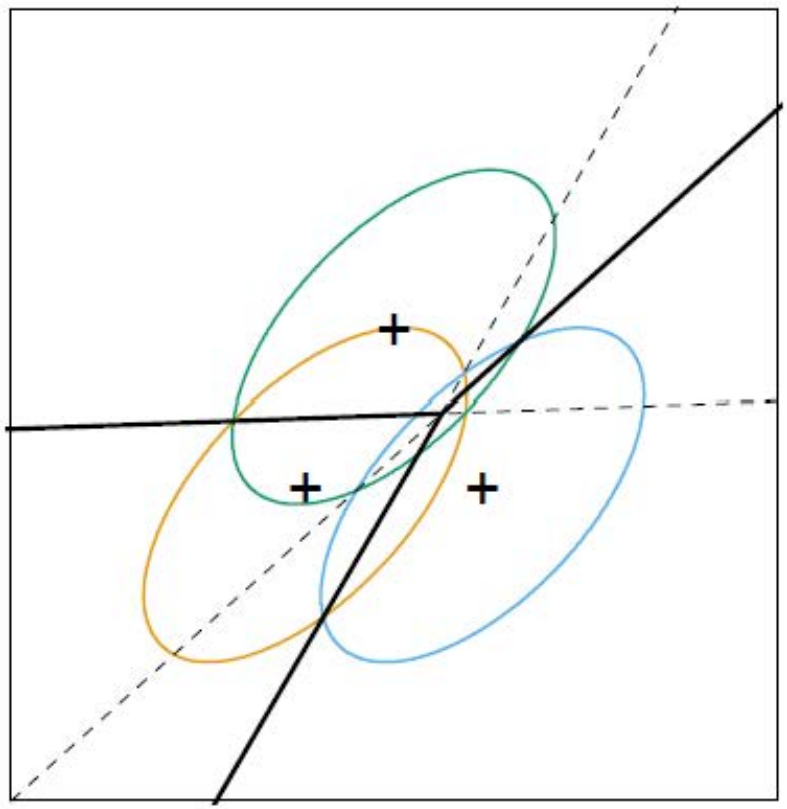
- In QDA,

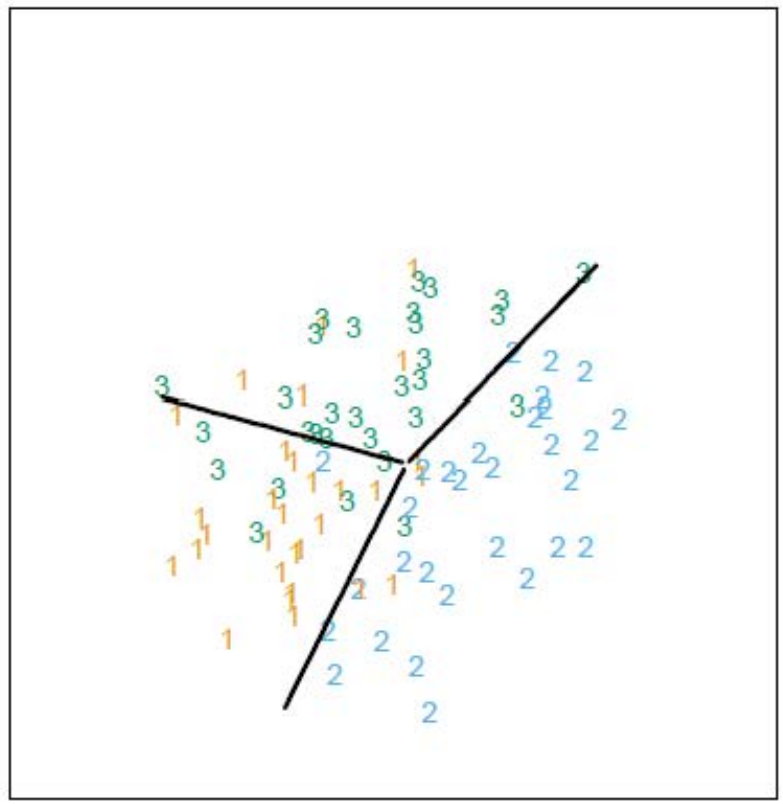
$$\mathcal{B}_{k,\ell} = \{x \in \mathbb{R}^p : \frac{1}{2}(x - \mu_k)^T(\Sigma_\ell^{-1} - \Sigma_k^{-1})(x - \mu_k) + x^T\Sigma_\ell^{-1}(\mu_k - \mu_\ell) - \frac{1}{2}(\mu_k + \mu_\ell)^T\Sigma_\ell^{-1}(\mu_k - \mu_\ell) + \log(\frac{\pi_k}{\pi_\ell}) + \frac{1}{2}\log(\frac{|\Sigma_\ell|}{|\Sigma_k|}) = 0\}$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: simulated data with $(X_1, X_2) \sim N_2(\mu_k, \Sigma)$ for $k = 1, 2, 3$. (Contours are for 95% volume.) (- - pairwise sep.)



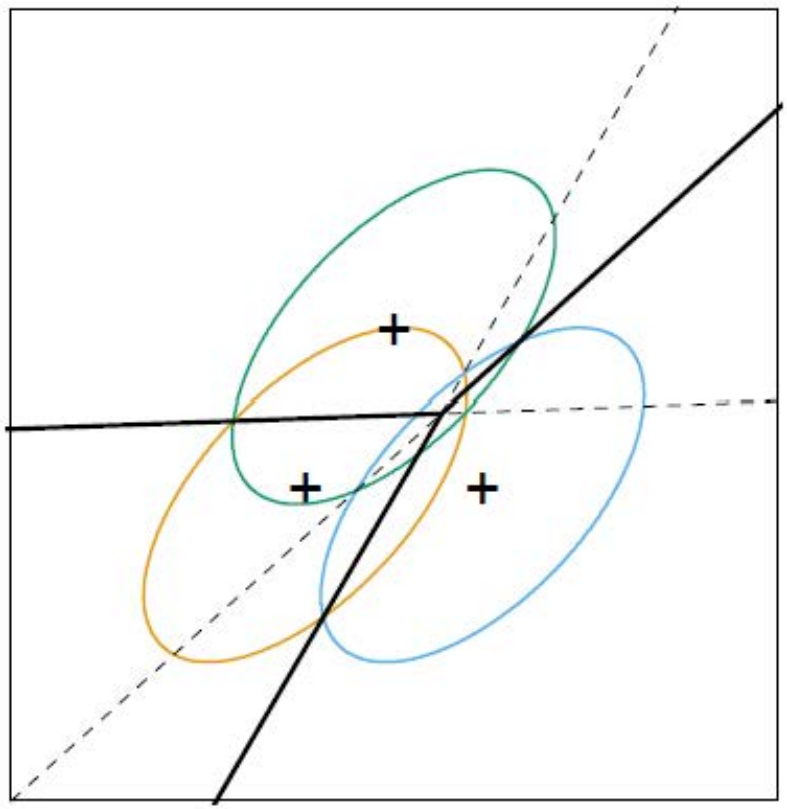


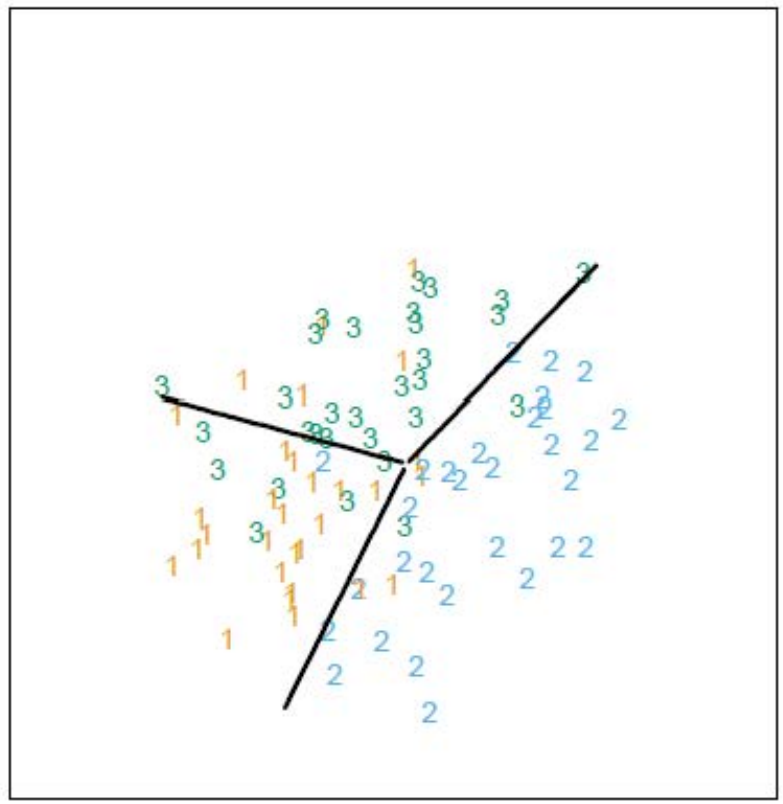
The right panel: decision boundaries are due to LDA in X_1 and X_2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: simulated data with $(X_1, X_2) \sim N_2(\mu_k, \Sigma)$ for $k = 1, 2, 3$. (Contours are for 95% volume.) (- - pairwise sep.)



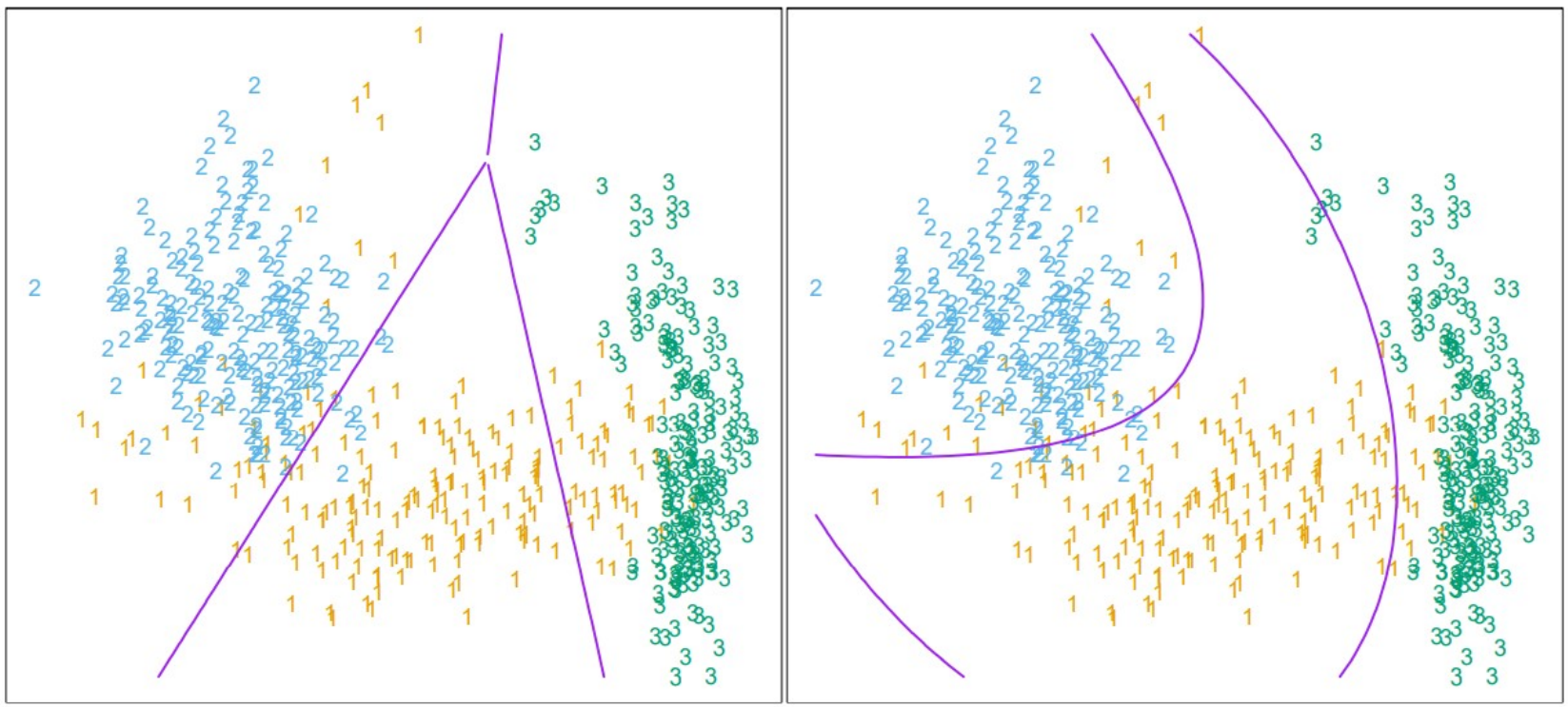


The right panel: decision boundaries are due to LDA in X_1 and X_2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: decision boundaries are due to LDA in X_1 and X_2. (Yet, not Gaussian.)

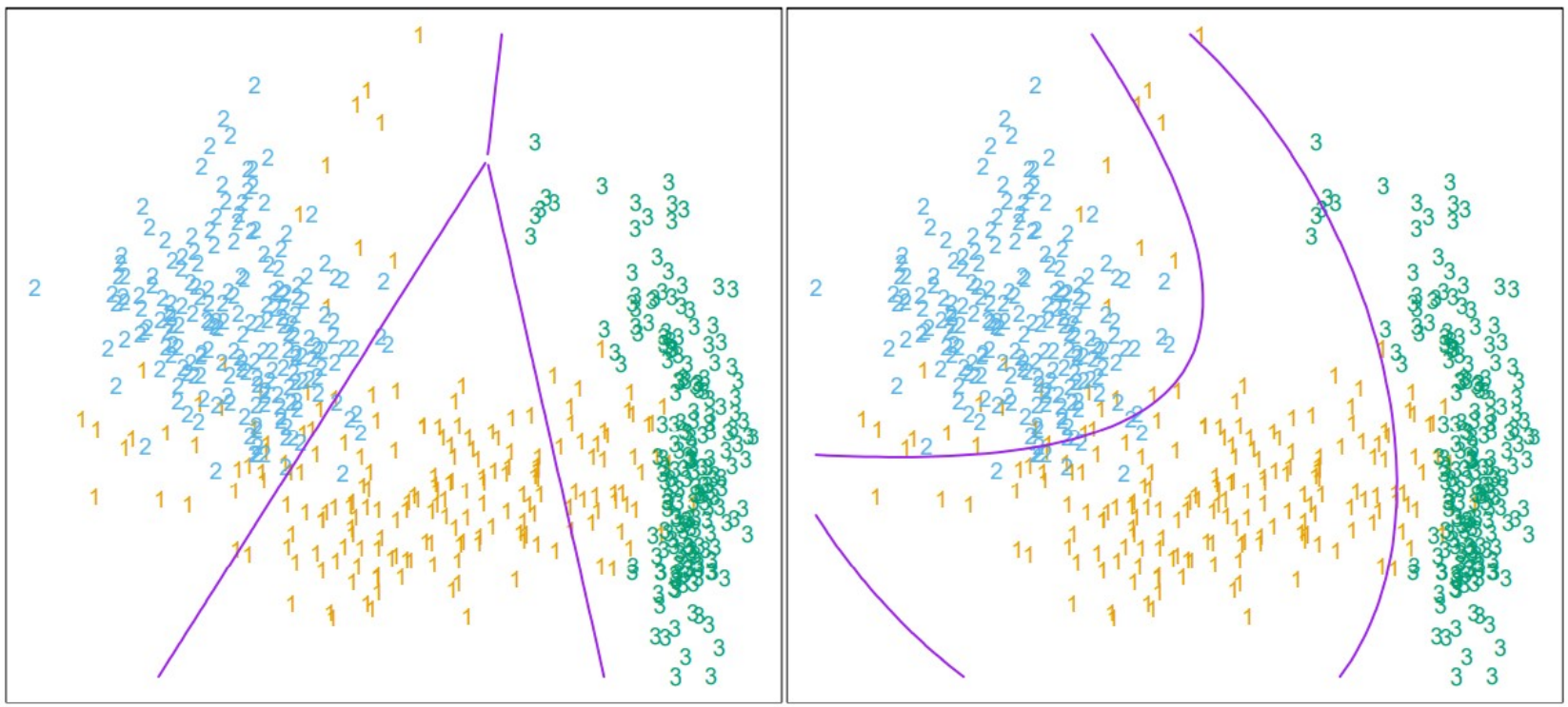


The right panel: decision boundaries are due to LDA in X_1, X_2, X_1X_2, X_1^2 and X_2^2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: decision boundaries are due to LDA in X_1 and X_2. (Yet, not Gaussian.)

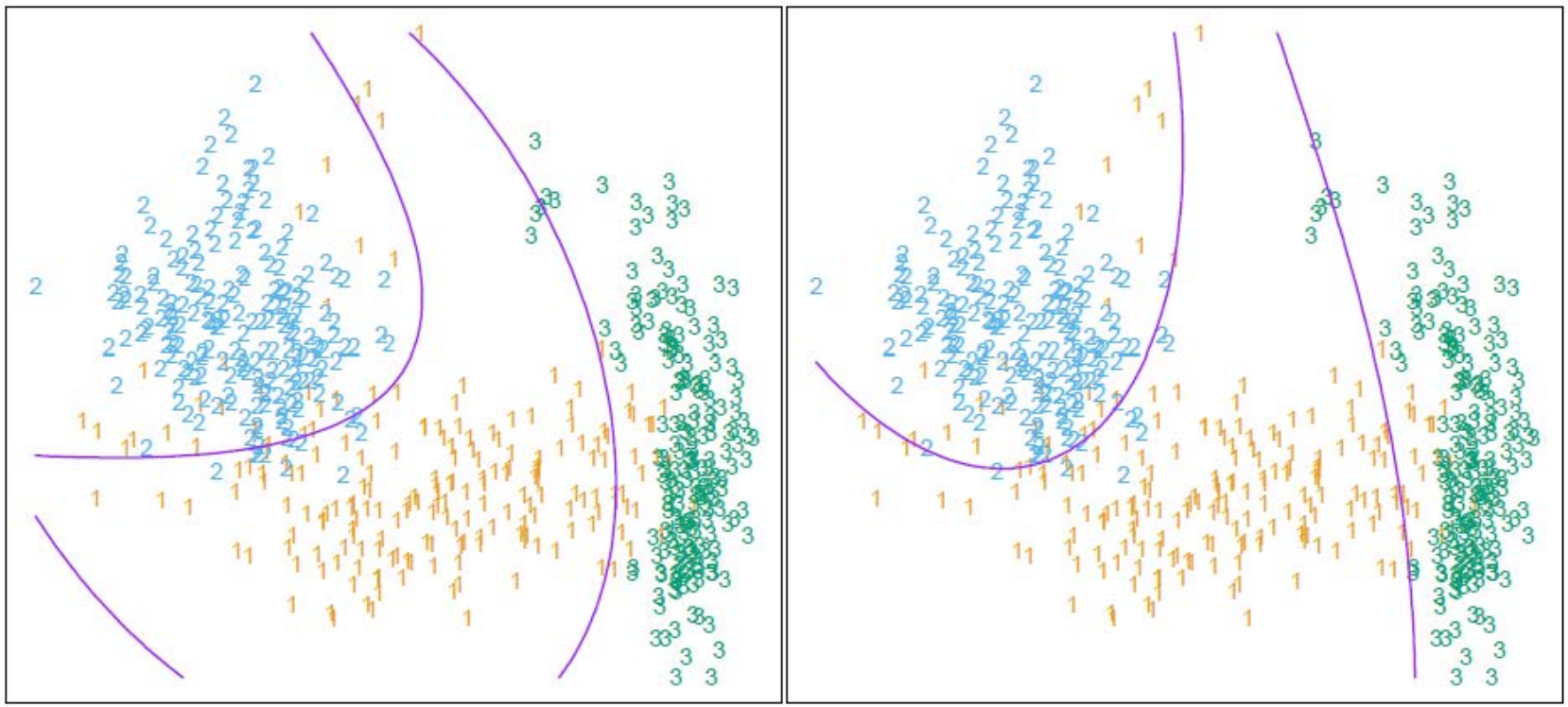


The right panel: decision boundaries are due to LDA in X_1, X_2, X_1X_2, X_1^2 and X_2^2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: decision boundaries are due to LDA in X_1, X_2, X_1X_2, X_1^2 and X_2^2.

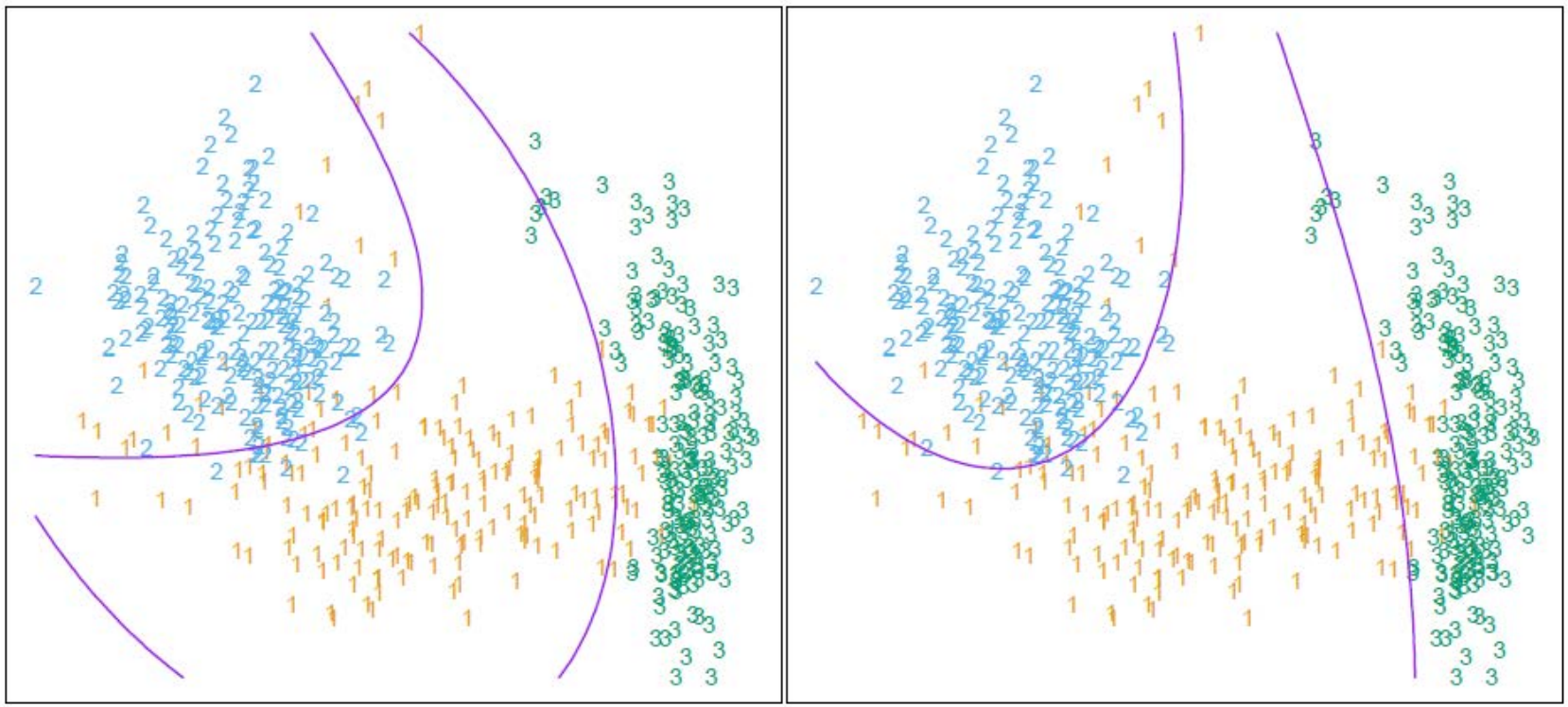


The right panel: decision boundaries are due to QDA in X_1 and X_2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

The left panel: decision boundaries are due to LDA in X_1, X_2, X_1X_2, X_1^2 and X_2^2.



The right panel: decision boundaries are due to QDA in X_1 and X_2.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

How can we specify the priors π_k and identify μ_k and Σ_k for all k?

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

- One way is to invoke Laplace principle

$$\pi_k = \frac{1}{K}$$

- Or better to empirically estimate them by

$$\hat{\pi}_k = \frac{N_k}{N}$$

where

$$N_k = \sum_{i=1}^{N} I(i \in \text{class}_k)$$

is the number of class k items in the training data.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

- One way is to invoke Laplace principle

$$\pi_k = \frac{1}{K}$$

- Or better to empirically estimate them by

$$\hat{\pi}_k = \frac{N_k}{N}$$

where

$$N_k = \sum_{i=1}^{N} I(i \in \text{class}_k)$$

is the number of class k items in the training data.

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

- The centers/means are estimated by

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{i \in \text{class}_k} x_i$$

- The covariance matrices are estimated by

$$\widehat{\Sigma}_k = \frac{1}{N_k - 1} \sum_{i \in \text{ class}_k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$$

- In case of LDA, the covariance matrix is the pooled estimate given by

$$\widehat{\Sigma} = \frac{1}{N - K} \sum_{k=1}^{K} (N_k - 1)\widehat{\Sigma}_k$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

- The centers/means are estimated by

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{i \in \text{class}_k} x_i$$

- The covariance matrices are estimated by

$$\widehat{\Sigma}_k = \frac{1}{N_k - 1} \sum_{i \in \text{ class}_k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$$

- In case of LDA, the covariance matrix is the pooled estimate given by

$$\widehat{\Sigma} = \frac{1}{N - K} \sum_{k=1}^{K} (N_k - 1)\widehat{\Sigma}_k$$

Bayes-based Classification Methods - Partitioning Methods

Linear and quadratic discriminant analyses

- The centers/means are estimated by

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{i \in \text{class}_k} x_i$$

- The covariance matrices are estimated by

$$\widehat{\Sigma}_k = \frac{1}{N_k - 1} \sum_{i \in \text{ class}_k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$$

- In case of LDA, the covariance matrix is the pooled estimate given by

$$\widehat{\Sigma} = \frac{1}{N - K} \sum_{k=1}^{K} (N_k - 1)\widehat{\Sigma}_k$$

Naïve Bayes Classifier

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The density f_k is assumed to be a the product of density functions of p independent random variables $f_{k,i}(x_i)$, for $k = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, p$. That is,

$$f_k(x) = \prod_{i=1}^{p} f_{k,i}(x_i)$$

for $x_i \in \mathbb{R}$. In which case the discriminant function is

$$\delta_k(x) = \sum_{i=1}^{p} \log(f_{k,i}(x_i)) + \log(\pi_k)$$

after removing constants.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

For NBC ...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = \log(\frac{\pi_k}{\pi_\ell}) + \sum_{i=1}^{p} \log\left(\frac{f_{k,i}(x_i)}{f_{\ell,i}(x_i)}\right)$$

for any k and ℓ. For $\ell = K$

$$\log\left(\frac{P(G=k|X=x)}{P(G=K|X=x)}\right) = \log(\frac{\pi_k}{\pi_K}) + \sum_{i=1}^{p} \log\left(\frac{f_{k,i}(x_i)}{f_{K,i}(x_i)}\right)$$

$$= a_k + \sum_{i=1}^{p} g_{k,i}(x_i)$$

which is a generalized additive model (GAM).

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

For NBC ...

$$\log\left(\frac{P(G=k|X=x)}{P(G=\ell|X=x)}\right) = \log(\frac{\pi_k}{\pi_\ell}) + \sum_{i=1}^{p} \log\left(\frac{f_{k,i}(x_i)}{f_{\ell,i}(x_i)}\right)$$

for any k and ℓ. For $\ell = K$

$$\log\left(\frac{P(G=k|X=x)}{P(G=K|X=x)}\right) = \log(\frac{\pi_k}{\pi_K}) + \sum_{i=1}^{p} \log\left(\frac{f_{k,i}(x_i)}{f_{K,i}(x_i)}\right)$$

$$= a_k + \sum_{i=1}^{p} g_{k,i}(x_i)$$

which is a generalized additive model (GAM).

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,j}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,j}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

in which cases, the parameters need to be estimated based on the training data (using MLEs for example).

4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Bayes-based Classification Methods - Partitioning Methods

Naïve Bayes Classifier

The densities $f_{k,i}(x_i)$ could be parametric like

1. $N(\mu_{k,i}, \sigma^2_{k,i})$ or
2. $Gamma(\alpha_{k,i}, \beta_{k,i})$
3. $Beta(\alpha_{k,i}, \beta_{k,i})$ for within class standardized data

   in which cases, the parameters need to be estimated based on the training data (using MLEs for example).
4. $f_{k,i}(x_i)$ could be empirically estimated as non-parametric.
   - If X_j is quantitative, then use the relative frequency histogram or better the kernel density estimator for the x_j within each class k to be $\hat{f}_{k,i}$.
   - If X_j is qualitative, then use the relative frequency discrete distribution the x_j within each class k to be $\hat{f}_{k,i}$.

Probability-based Classification Methods - Partitioning Methods

LDA, QDA, KNN, and naïve Bayes

Example (SA Heart Disease)

DIY in R

1. Carry out LDA and QDA using *MASS*.
2. Carry out naïve Bayes classification using *e1071*.
3. Carry out KNN classification using *class*.
4. To evaluate the performance of the classifiers: produce a confusion matrix (a table of predicted vs actual classes) for all of the previous methods. Then calculate sensitivity (% of true positive identified as positive) and specificity (% of true negative identified as negative).

Probability-based Classification Methods - Partitioning Methods

LDA, QDA, KNN, and naïve Bayes

Example (SA Heart Disease)

DIY in R

1. Carry out LDA and QDA using *MASS*.
2. Carry out naïve Bayes classification using *e1071*.
3. Carry out KNN classification using *class*.
4. To evaluate the performance of the classifiers: produce a confusion matrix (a table of predicted vs actual classes) for all of the previous methods. Then calculate sensitivity (% of true positive identified as positive) and specificity (% of true negative identified as negative).

Probability-based Classification Methods - Partitioning Methods

LDA, QDA, KNN, and naïve Bayes

Example (SA Heart Disease)

DIY in R

1. Carry out LDA and QDA using *MASS*.
2. Carry out naïve Bayes classification using *e1071*.
3. Carry out KNN classification using *class*.
4. To evaluate the performance of the classifiers: produce a confusion matrix (a table of predicted vs actual classes) for all of the previous methods. Then calculate sensitivity (% of true positive identified as positive) and specificity (% of true negative identified as negative).

Probability-based Classification Methods - Partitioning Methods

LDA, QDA, KNN, and naïve Bayes

Example (SA Heart Disease)

DIY in R

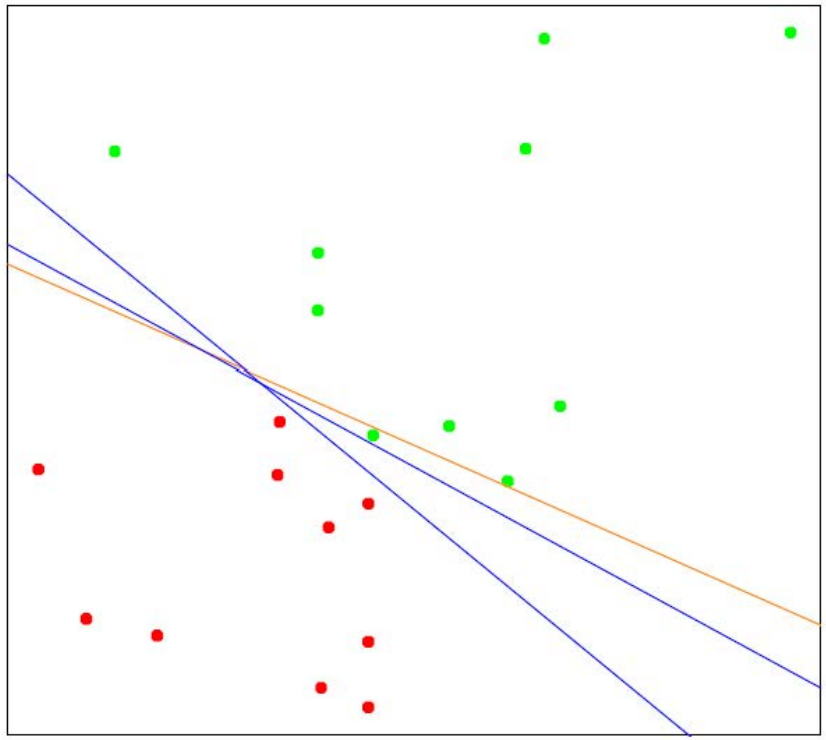
1. Carry out LDA and QDA using *MASS*.
2. Carry out naïve Bayes classification using *e1071*.
3. Carry out KNN classification using *class*.
4. To evaluate the performance of the classifiers: produce a confusion matrix (a table of predicted vs actual classes) for all of the previous methods. Then calculate sensitivity (% of true positive identified as positive) and specificity (% of true negative identified as negative).

Please study the different methods in the ISL book. See also Poisson regression using R in the ISL.

Separating Hyperplanes

Separating Hyperplanes

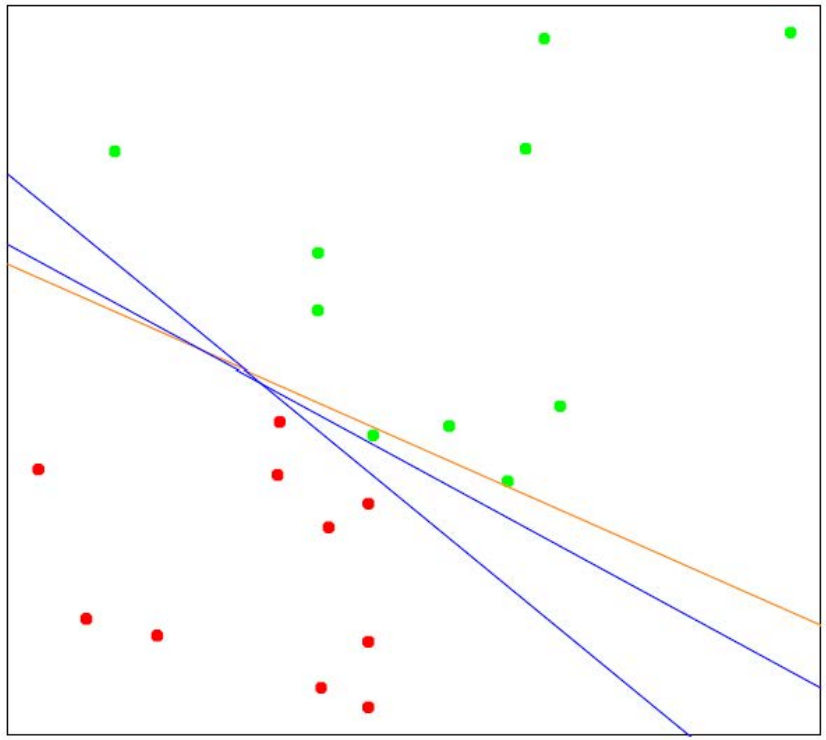
Example (Simulated data in $\mathbb{R}^2$)



The orange line is based on least squares which is also equivalent to LDA in that situation of two classes. It is not perfect.

Separating Hyperplanes

Example (Simulated data in $\mathbb{R}^2$)



The orange line is based on least squares which is also equivalent to LDA in that situation of two classes. It is not perfect.

Separating Hyperplanes

- Rosenblatt's Perceptron Learning Algorithm.

- Optimal Separating Hyperplanes (a step towards support vector machines).

Separating Hyperplanes

- Rosenblatt's Perceptron Learning Algorithm.

  For $y = \pm 1$,

  $$\text{perceptron} = \text{sign}(\beta_0 + x^T \beta)$$

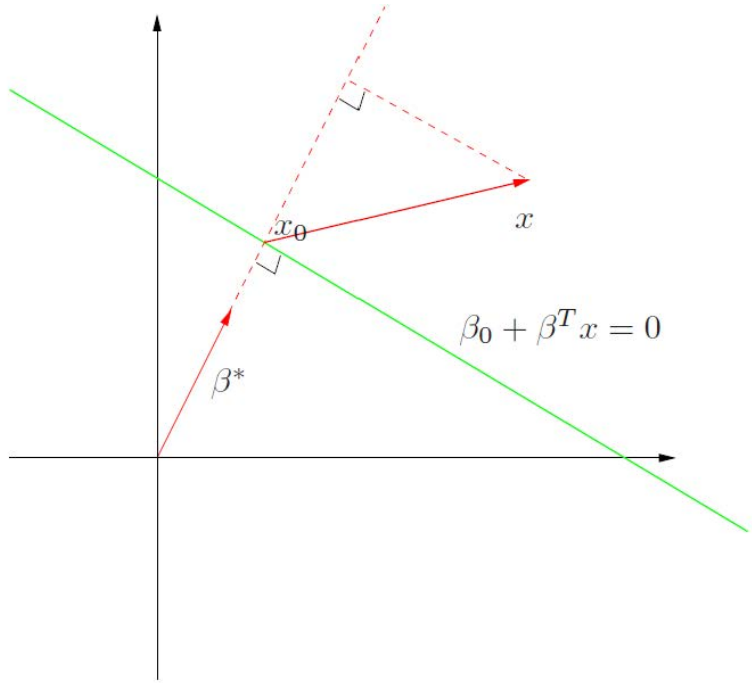
- Optimal Separating Hyperplanes (a step towards support vector machines).

Separating Hyperplanes

From linear algebra ...

- $\beta^* = \frac{\beta}{\|\beta\|}$ is orthonormal to the separating hyperplane

$$L = \{x : \beta_0 + x^T\beta = 0\}$$



Separating Hyperplanes

From linear algebra ...

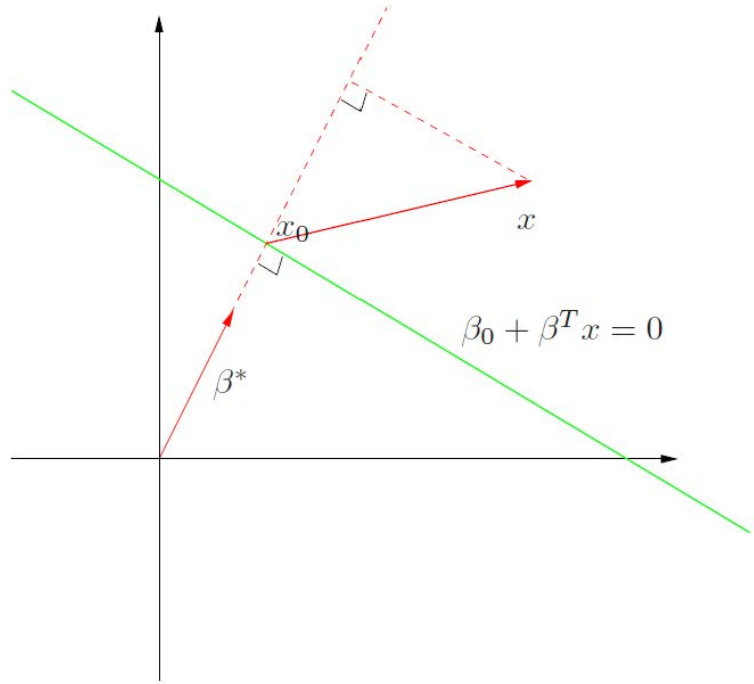
- $\beta^* = \frac{\beta}{\|\beta\|}$ is orthonormal to the separating hyperplane

$$L = \{x : \beta_0 + x^T\beta = 0\}$$

if

$$(x_1 - x_0)^T\beta^* = 0$$

for any $x_0, x_1 \in L$.



Separating Hyperplanes

From linear algebra ...

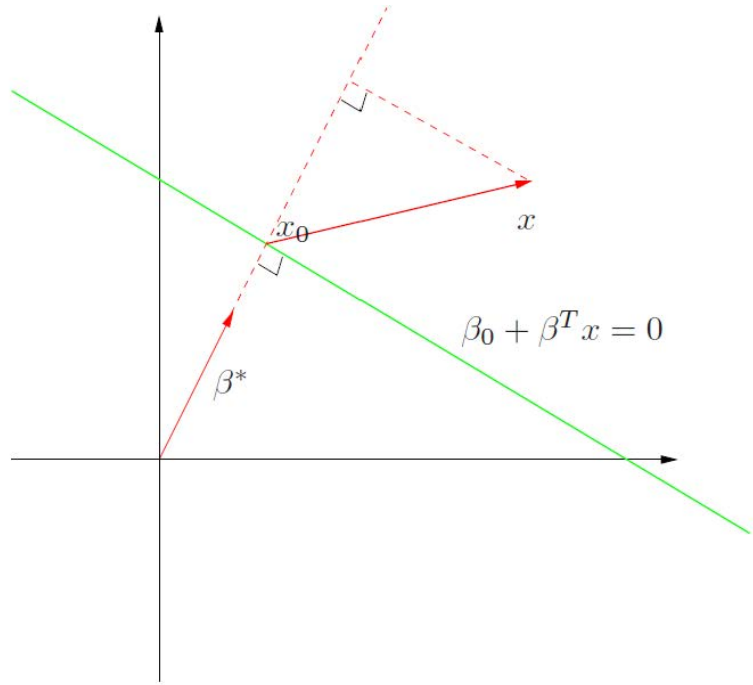
- $\beta^* = \frac{\beta}{\|\beta\|}$ is orthonormal to the separating hyperplane

$$L = \{x : \beta_0 + x^T\beta = 0\}$$

if

$$(x_1 - x_0)^T\beta^* = 0$$

for any $x_0, x_1 \in L$.



- For $x \notin L$, the signed distance of x to L is

$$(x - x_0)^T\beta^* = \frac{\beta_0 + x^T\beta}{\|\beta\|} \propto \beta_0 + x^T\beta$$

Separating Hyperplanes

From linear algebra ...

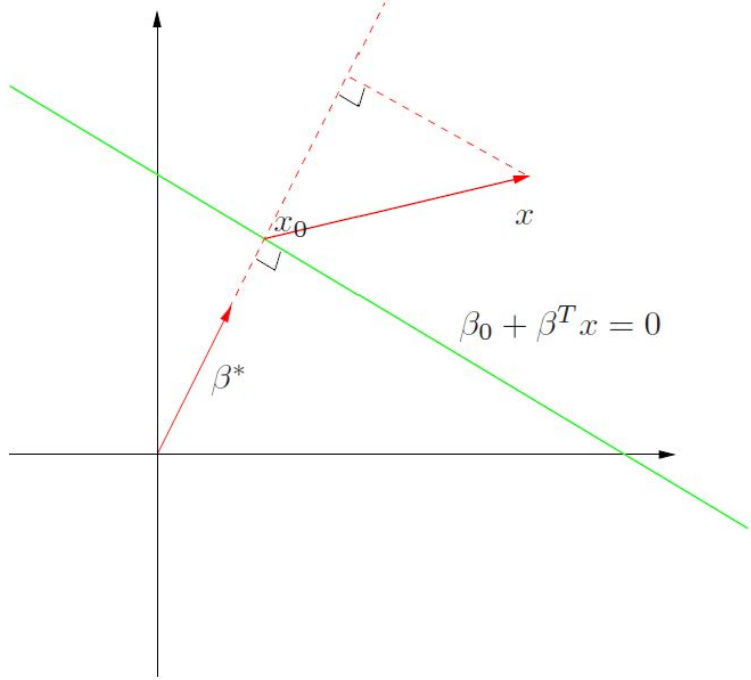
- $\beta^* = \frac{\beta}{\|\beta\|}$ is orthonormal to the separating hyperplane

$$L = \{x : \beta_0 + x^T\beta = 0\}$$

if

$$(x_1 - x_0)^T\beta^* = 0$$

for any $x_0, x_1 \in L$.



- For $x \notin L$, the signed distance of x to L is

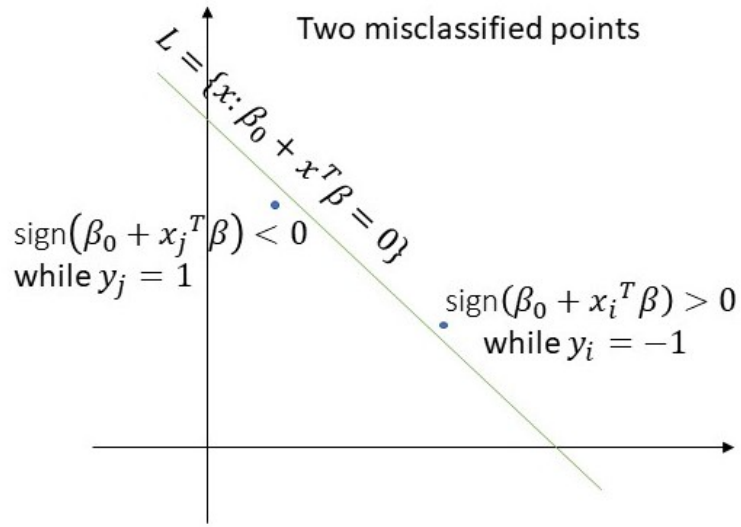
$$(x - x_0)^T\beta^* = \frac{\beta_0 + x^T\beta}{\|\beta\|} \propto \beta_0 + x^T\beta$$

- Note that, signed distance of $x_1 \in L$ is zero.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

RPLA looks only at those misclassified points, put in a set $\mathcal{M}$, and minimizes the signed distances to the decision boundary



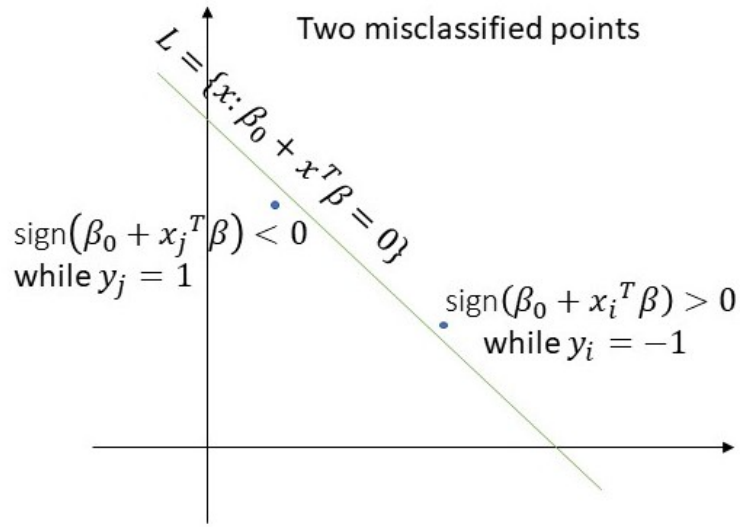
-
-

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

RPLA looks only at those misclassified points, put in a set $\mathcal{M}$, and minimizes the signed distances to the decision boundary

- Minimize
  $D(\beta_0, \beta) = -\sum_{i \in \mathcal{M}} y_i(\beta_0 + x_i^T \beta)$
-

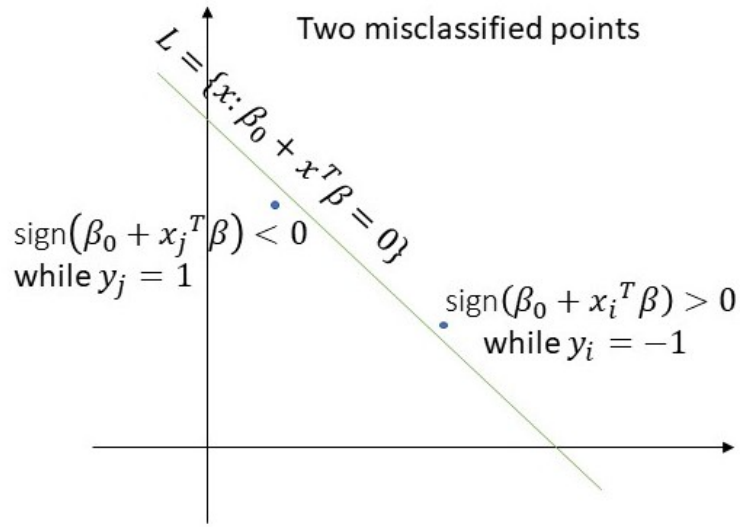


Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

RPLA looks only at those misclassified points, put in a set $\mathcal{M}$, and minimizes the signed distances to the decision boundary

- Minimize
  $D(\beta_0, \beta) = -\sum_{i \in \mathcal{M}} y_i(\beta_0 + x_i^T \beta) \geq 0$
-



Separating Hyperplanes

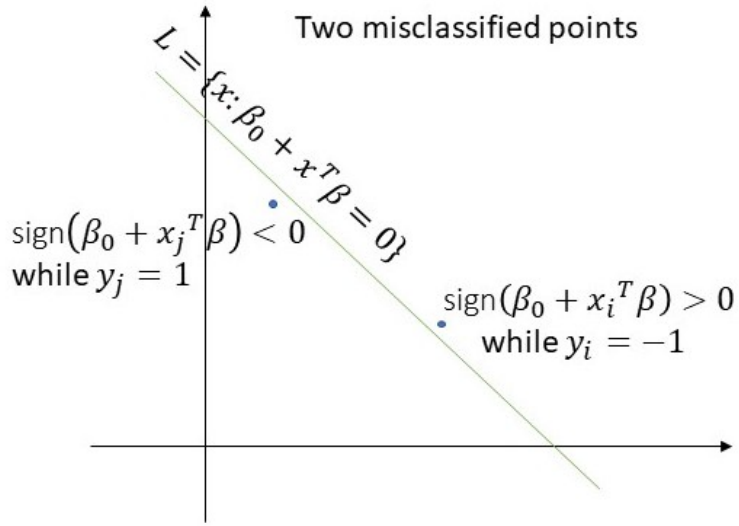
Rosenblatt's Perceptron Learning Algorithm.

RPLA looks only at those misclassified points, put in a set $\mathcal{M}$, and minimizes the signed distances to the decision boundary

- Minimize
  $D(\beta_0, \beta) = -\sum_{i \in \mathcal{M}} y_i(\beta_0 + x_i^T \beta) \geq 0$

- The gradient is

$$\frac{\partial D}{\partial \beta_0} = -\sum_{i \in \mathcal{M}} y_i$$



Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

RPLA looks only at those misclassified points, put in a set $\mathcal{M}$, and minimizes the signed distances to the decision boundary

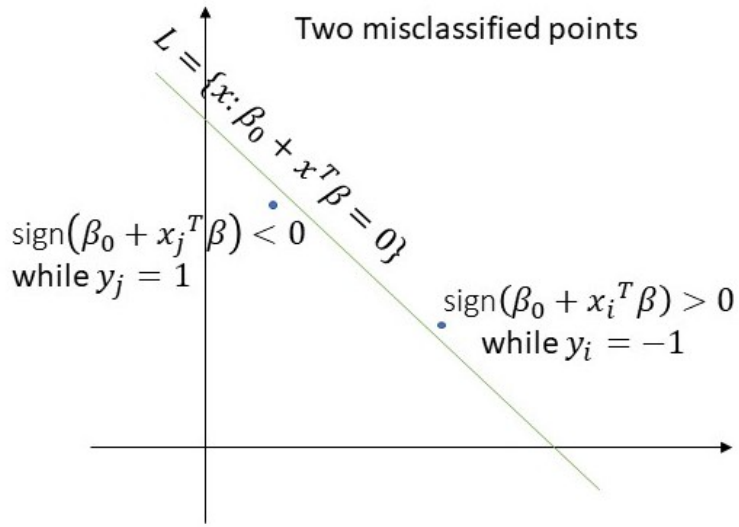
- Minimize
  $D(\beta_0, \beta) = -\sum_{i \in \mathcal{M}} y_i(\beta_0 + x_i^T \beta) \geq 0$

- The gradient is

$$\frac{\partial D}{\partial \beta_0} = -\sum_{i \in \mathcal{M}} y_i$$

and

$$\frac{\partial D}{\partial \beta} = -\sum_{i \in \mathcal{M}} y_i x_i$$



Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

But instead of using steepest decent in which

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}\sum_{i\in\mathcal{M}} y_i\\ \sum_{i\in\mathcal{M}} y_i x_i\end{pmatrix}$$

with learning rate $\eta > 0$.

RPLA proceeds using stochastic gradient descent algorithm and sequentially visits each point in $\mathcal{M}$

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}y_i\\ y_i x_i\end{pmatrix}$$

It revolve with i and update the vector after each visit to each point $x_i \in \mathcal{M}$.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

But instead of using steepest decent in which

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}\sum_{i\in\mathcal{M}} y_i\\ \sum_{i\in\mathcal{M}} y_i x_i\end{pmatrix}$$

with learning rate $\eta > 0$.

RPLA proceeds using stochastic gradient descent algorithm and sequentially visits each point in $\mathcal{M}$

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}y_i\\ y_i x_i\end{pmatrix}$$

It revolve with i and update the vector after each visit to each point $x_i \in \mathcal{M}$.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

But instead of using steepest decent in which

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}\sum_{i\in\mathcal{M}} y_i\\ \sum_{i\in\mathcal{M}} y_i x_i\end{pmatrix}$$

with learning rate $\eta > 0$.

RPLA proceeds using stochastic gradient descent algorithm and sequentially visits each point in $\mathcal{M}$

$$\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{new} = \begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}_{old} + \eta \begin{pmatrix}y_i\\ y_i x_i\end{pmatrix}$$

It revolve with i and update the vector after each visit to each point $x_i \in \mathcal{M}$.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

- It requires an initial vector and TOL for stopping.
- Solutions are not unique and depend on the initial vector.

Separating Hyperplanes

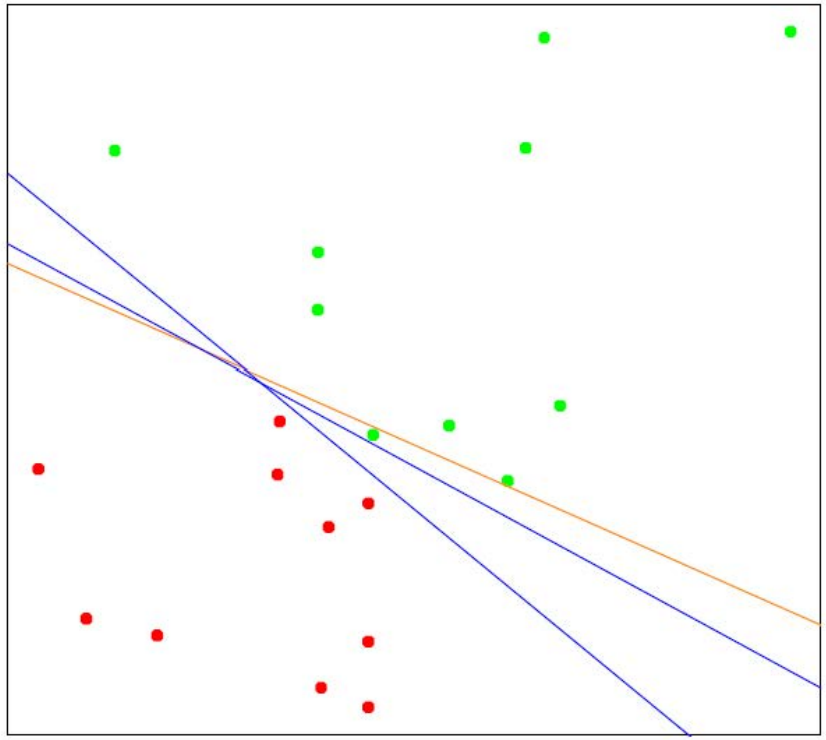
Rosenblatt's Perceptron Learning Algorithm.

- It requires an initial vector and TOL for stopping.
- Solutions are not unique and depend on the initial vector.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

Example (Simulated data in $\mathbb{R}^2$)

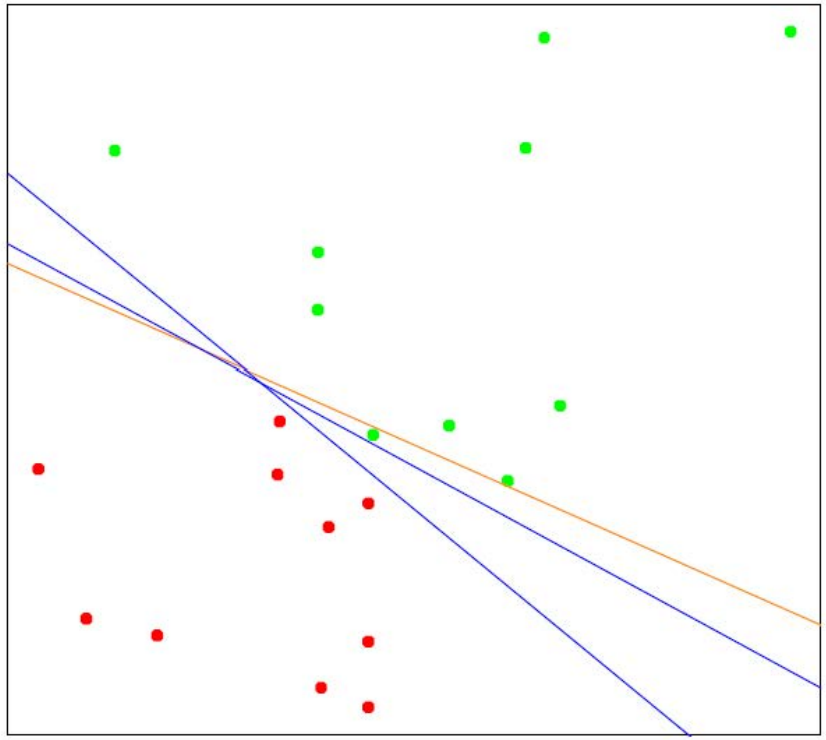


The two blue lines are two RPLA solutions for two different initial vectors. To be rectified using constraints.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

Example (Simulated data in $\mathbb{R}^2$)



The two blue lines are two RPLA solutions for two different initial vectors.To be rectified using constraints.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

- If classes are linearly separable, RPLA converges in a finite number of steps, possibly large.
- The smaller the gaps between the points in $\mathcal{M}$ and L, the larger the number of steps is.
- That makes a problem that might be mitigated by basis transformation with the chance of overfitting.
- If they are not separable, it goes into an infinite cycle.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

- If classes are linearly separable, RPLA converges in a finite number of steps, possibly large.
- The smaller the gaps between the points in $\mathcal{M}$ and L, the larger the number of steps is.
- That makes a problem that might be mitigated by basis transformation with the chance of overfitting.
- If they are not separable, it goes into an infinite cycle.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

- If classes are linearly separable, RPLA converges in a finite number of steps, possibly large.
- The smaller the gaps between the points in $\mathcal{M}$ and L, the larger the number of steps is.
- That makes a problem that might be mitigated by basis transformation with the chance of overfitting.
- If they are not separable, it goes into an infinite cycle.

Separating Hyperplanes

Rosenblatt's Perceptron Learning Algorithm.

- If classes are linearly separable, RPLA converges in a finite number of steps, possibly large.
- The smaller the gaps between the points in $\mathcal{M}$ and L, the larger the number of steps is.
- That makes a problem that might be mitigated by basis transformation with the chance of overfitting.
- If they are not separable, it goes into an infinite cycle.

Separating Hyperplanes

Optimal Separating Hyperplanes.

OSH maximizes the margins (signed distances M) of the slab

- Solve

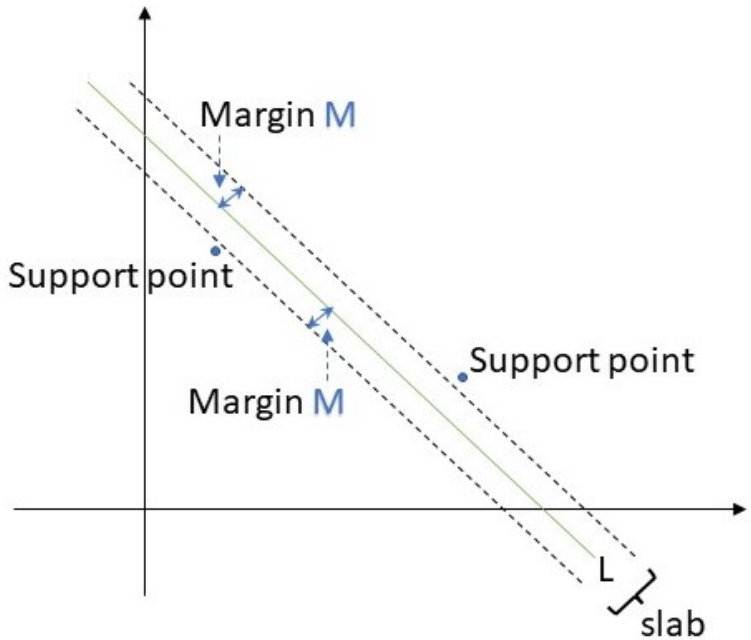
$$\max_{\beta_0,\beta} M$$

subject to

$$\frac{1}{\|\beta\|} y_i(\beta_0 + x_i^T \beta) \geq M$$

for $i = 1, 2, \ldots, N$.

- Set $\|\beta\| = \frac{1}{M}$



Separating Hyperplanes

Optimal Separating Hyperplanes.

OSH maximizes the margins (signed distances M) of the slab

- Solve

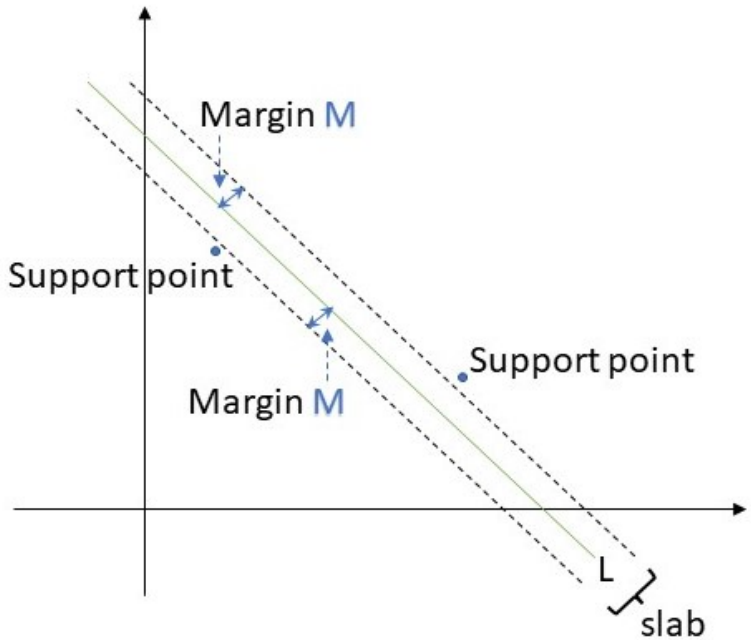
$$\max_{\beta_0,\beta} M$$

subject to

$$\frac{1}{\|\beta\|} y_i(\beta_0 + x_i^T \beta) \geq M$$

for $i = 1, 2, \ldots, N$.

- Set $\|\beta\| = \frac{1}{M}$



Separating Hyperplanes

Optimal Separating Hyperplanes.

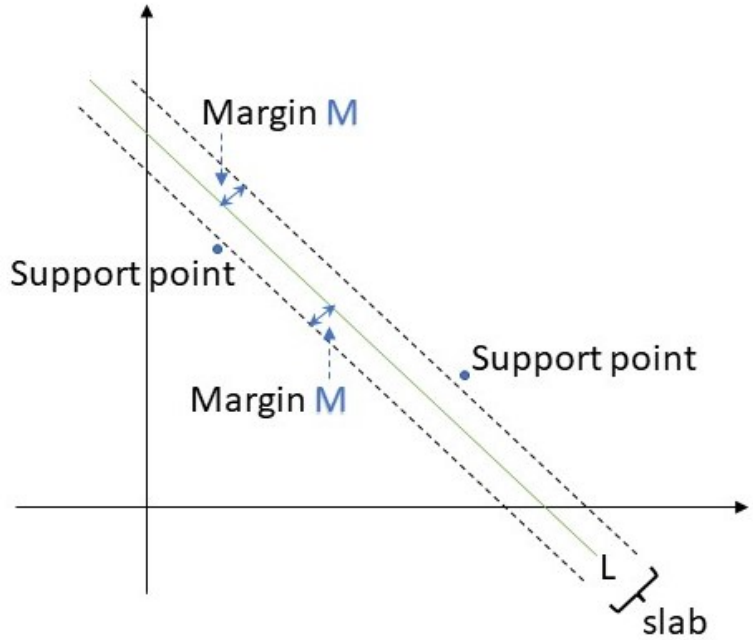
- Then the problem becomes equivalent to the convex optimization problem

$$\min_{\beta_0,\beta} \frac{1}{2} \|\beta\|^2$$

subject to

$$y_i(\beta_0 + x_i^T \beta) \geq 1$$

for $i = 1, 2, \ldots, N$.



Separating Hyperplanes

Optimal Separating Hyperplanes.

- Step 1: is the Lagrange problem to

$$\min_{\beta_0,\beta} L_p$$

where

$$L_p = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i (y_i(\beta_0 + x_i^T \beta) - 1)$$

- Setting derivatives equal to zero leads to

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \text{ and } \sum_{i=1}^{N} \alpha_i y_i x_i = \beta$$

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Step 1: is the Lagrange problem to

$$\min_{\beta_0,\beta} L_p$$

where

$$L_p = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i (y_i(\beta_0 + x_i^T \beta) - 1)$$

- Setting derivatives equal to zero leads to

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \text{ and } \sum_{i=1}^{N} \alpha_i y_i x_i = \beta$$

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Substituting with those into L_p we get

$$L_p = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Step 2: Using Wolfe dual optimization, the problem becomes

$$\max_{\alpha_i} L_D$$

where

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

subject to (the Karush-Kuhn-Tucker conditions)

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \text{ and } \sum_{i=1}^{N} \alpha_i y_i x_i = \beta$$

$$\alpha_i \geq 0$$

and

$$\alpha_i (y_i (\beta_0 + x_i^T \beta) - 1) = 0$$

for $i = 1, 2, \ldots, N$.

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Here, β depend of α through the KKT conditions.
- If $y_i(\beta_0 + x_i^T\beta) - 1 > 0$ then the point is not on the line and $\alpha_i = 0$.
- If $y_i(\beta_0 + x_i^T\beta) - 1 = 0$ then the point is on the line and $\alpha_i > 0$ which will contribute to the values of β that will make up the decision boundary based on this support points on the slab's boundaries.
- Separation will occur according to $\hat{G}(x) = sign(\hat{\beta}_0 + x^T\hat{\beta}_1)$.

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Here, β depend of α through the KKT conditions.
- If $y_i(\beta_0 + x_i^T\beta) - 1 > 0$ then the point is not on the line and $\alpha_i = 0$.
- If $y_i(\beta_0 + x_i^T\beta) - 1 = 0$ then the point is on the line and $\alpha_i > 0$ which will contribute to the values of β that will make up the decision boundary based on this support points on the slab's boundaries.
- Separation will occur according to $\hat{G}(x) = sign(\hat{\beta}_0 + x^T\hat{\beta}_1)$.

Separating Hyperplanes

Optimal Separating Hyperplanes.

- Here, β depend of α through the KKT conditions.
- If $y_i(\beta_0 + x_i^T\beta) - 1 > 0$ then the point is not on the line and $\alpha_i = 0$.
- If $y_i(\beta_0 + x_i^T\beta) - 1 = 0$ then the point is on the line and $\alpha_i > 0$ which will contribute to the values of β that will make up the decision boundary based on this support points on the slab's boundaries.
- Separation will occur according to $\hat{G}(x) = sign(\hat{\beta}_0 + x^T\hat{\beta}_1)$.

Separating Hyperplanes

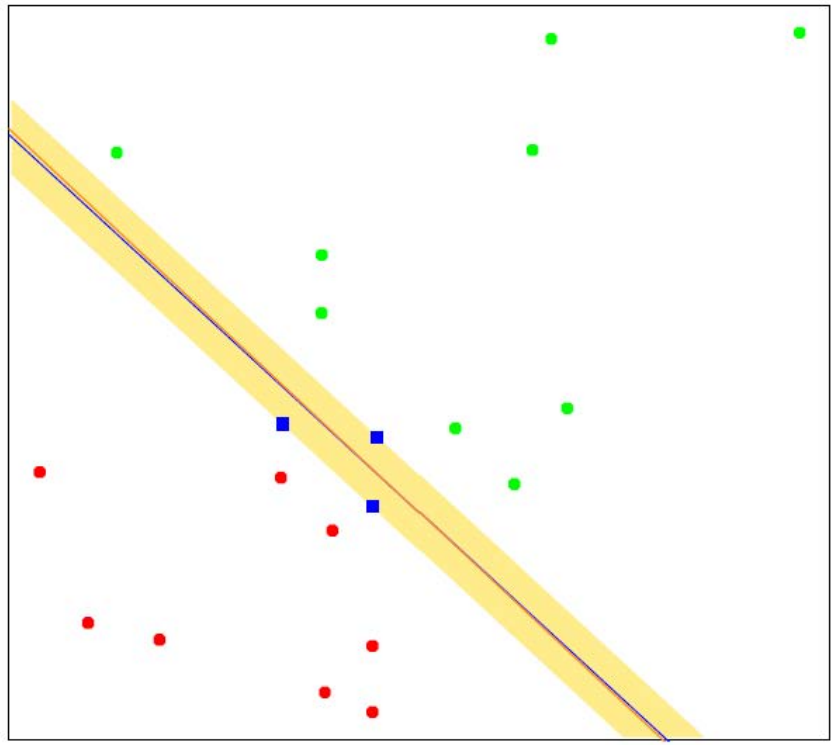
Optimal Separating Hyperplanes.

- Here, β depend of α through the KKT conditions.
- If $y_i(\beta_0 + x_i^T\beta) - 1 > 0$ then the point is not on the line and $\alpha_i = 0$.
- If $y_i(\beta_0 + x_i^T\beta) - 1 = 0$ then the point is on the line and $\alpha_i > 0$ which will contribute to the values of β that will make up the decision boundary based on this support points on the slab's boundaries.
- Separation will occur according to $\hat{G}(x) = sign(\hat{\beta}_0 + x^T\hat{\beta}_1)$.

Separating Hyperplanes

Optimal Separating Hyperplanes.

Example (Simulated data in $\mathbb{R}^2$)



The blue line is the OHS and the red line is due to logistic regression.

End of Set 4