See Hastie and Tibshirani, chapters 2 and 4.

7 LINEAR AND QUADRATIC CLASSIFICATION

7.1 INTRODUCTION OF CLASSIFICATION

In classification, the individuals in the population come from K different classes/groups. (E.g. healthy/unhealthy patients for $K = 2$)

☛ At our disposal: a **training** sample of individuals from the population for which we observe $(X_1, G_1), \ldots, (X_n, G_n)$, where

- $G_i \in \{1, 2, \ldots, K\}$ is the class/group label of the ith individual; an individual belong to exactly one group.
- X_i is a vector $X_i = (X_{i1}, \ldots, X_{ip})^T$ of explanatory variables, e.g. age, blood pressure, etc.

☛ A new individual comes in, for which we only observe $X = x = (x_1, \ldots, x_p)^T$ but not G. We want to find the class label $G \in \{1, \ldots, K\}$ of that new individual.

☛ Note: Without highlighting a particular sample, we use the notation (X, G) to denote a generic individual from the population.

7.2 MAIN IDEAS OF TECHNIQUES

☛ Main idea: compare the new value x with the X_i's from the training samples.

☛ If x is more similar to the X_i's from group k than the other groups $\Rightarrow$ classify the new individual in group k.

☛ How to decide if x is more "similar" to the X_i' s from group k than from other groups?

☛ For $k = 1, \ldots, K$, use the training sample of i.i.d. data $(X_i, G_i)_{i=1}^n$ to estimate the probability

$$P(G = k|X = x).$$

that an individual comes from group k given that his/her X is x:

☛ Denote the estimated value by

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

☛ Two classes of methods: **Regression** and **Bayes methods**.

☛ Then, classify new individual in group k if

$$\hat{P}(G = k|X = x) > \max_{j=1,\ldots,K, j\neq k} \hat{P}(G = j|X = x).$$

☛ Suppose $K = 2$. Then we classify a new individual with $X = x$ in group 1 if

$$\hat{P}(G = 1|X = x) > \hat{P}(G = 2|X = x).$$

☛ Idea: obtain the estimate $\hat{P}(G = k|X = x)$ is through regression with the training sample $(X_1, G_1), \ldots, (X_n, G_n)$.

☛ Define new variables from G. For $k = 1, 2$, let

$$Y_k = I\{G = k\} = \begin{cases} 1 & \text{if } G = k \\ 0 & \text{otherwise.} \end{cases}$$

☛ At the training sample level with $i = 1, \ldots, n$, we have

$$Y_{ik} = I\{G_i = k\}.$$

☛ Example: Suppose we observe a training sample of size $n = 7$ where

$$(G_1, \ldots, G_7) = (1, 1, 1, 2, 2, 2, 1).$$

Then, for $i = 1, \ldots, n$ and $k = 1, 2$, the Y_{ik}'s are given by:

$$(Y_{11}, \ldots, Y_{71}) = (1, 1, 1, 0, 0, 0, 1)$$
$$(Y_{12}, \ldots, Y_{72}) = (0, 0, 0, 1, 1, 1, 0).$$

☛ Note that

$$P(G = k|X = x) = E\big(I\{G = k\}|X = x\big) = m_k(x),$$

if we define the regression curve:

$$m_k(x) = E\big(Y_k|X = x\big).$$

☛ We can use regression techniques to estimate m_k.

☛ Since every individual must belong to one of the two groups, so that for all $x \in \mathbb{R}^p$

$$P(G = 1|X = x) + P(G = 2|X = x) = 1,$$

that is

$$m_2(x) = 1 - m_1(x)\,.$$

Thus we only need to estimate m_1 (we can deduce m_2 from it).

7.3.1 LINEAR REGRESSION CLASSIFIER

- When we use a *linear regression* classifier, we assume

$$m_1(x) = E\big(Y_1|X = x\big) = P(G = 1|X = x) = \beta_0 + \beta^T x.$$

- Use a least-squares (or PCR or PLS) estimates of (β_0, β) to construct

$$\hat{m}_1(x) = \hat{\beta}_0 + \hat{\beta}^T x.$$

- Deduce $\hat{m}_2 = 1 - \hat{m}_1$.

- Classify new x in group 1 if

$$\hat{m}_1(x) > \hat{m}_2(x) \iff \hat{\beta}_0 + \hat{\beta}^T x > 1 - \hat{\beta}_0 - \hat{\beta}^T x \iff \hat{\beta}_0 + \hat{\beta}^T x > 1/2\,;$$

otherwise classify x in group 2.

- Relies on the validity of the linear regression model; usually only an approximation in real life.

Example: Golub data (ESL, Section 18.4)

- Molecular Classification of cancer

- Reference: Class discovery and class prediction by gene expression monitoring, Science 286:531-7, Golub TR et al (1999).

- G: two types of leukemia ("ALL" and "AML").

- $X = (X_1, X_2)^T$: expression measurements for two genes connected to leukemia type.

- From the estimated linear model we have

$$\widehat{m}_1(x) = 0.9231 + 0.2454x_1 - 0.4800x_2$$
$$\widehat{m}_2(x) = 0.07691 - 0.24542x_1 + 0.47999x_2 \approx 1 - \hat{m}_1(x)$$

as expected (the $\approx$ is due to numerical errors).

Classification/discrimination boundary

- **Classification boundary**: the boundary between the regions in the X-space where we classify a new data point into the two groups.

- It is obtained by solving the equation

$$\widehat{m}_1(x) - \widehat{m}_2(x) = 0$$

which, for the case of linear regression classifier, is equivalent to

$$\widehat{m}_1(x) = 1/2.$$

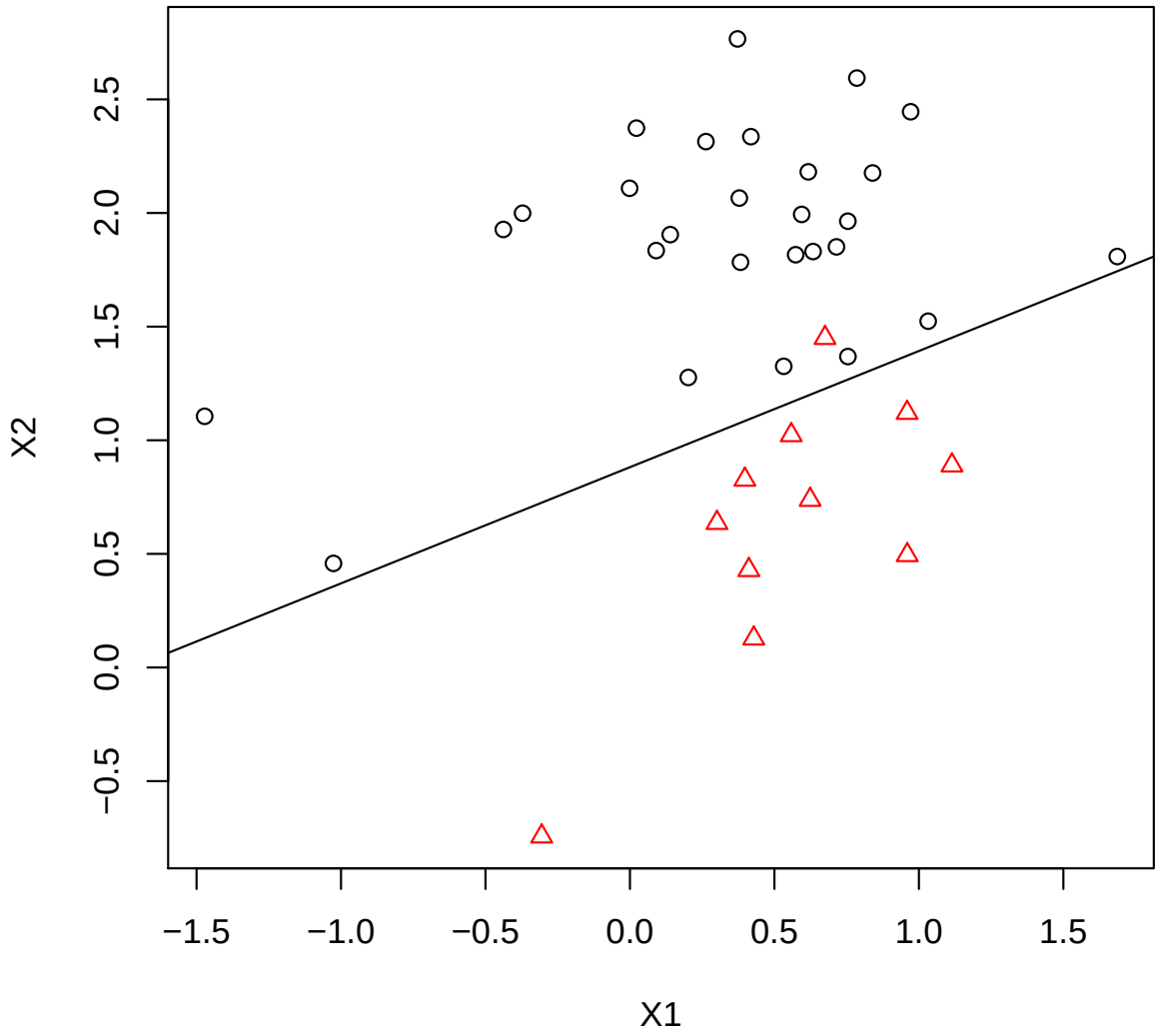
- E.g.: In the Golub data case, the boundary is obtained by solving

$$0.9231 + 0.2454x_1 - 0.4800x_2 = 1/2.$$

- This can be expressed by the line

$$x_2 = \frac{0.9231 - 0.5}{0.4800} + \frac{0.2454}{0.4800}x_1 = 0.8815 + 0.5113x_1.$$

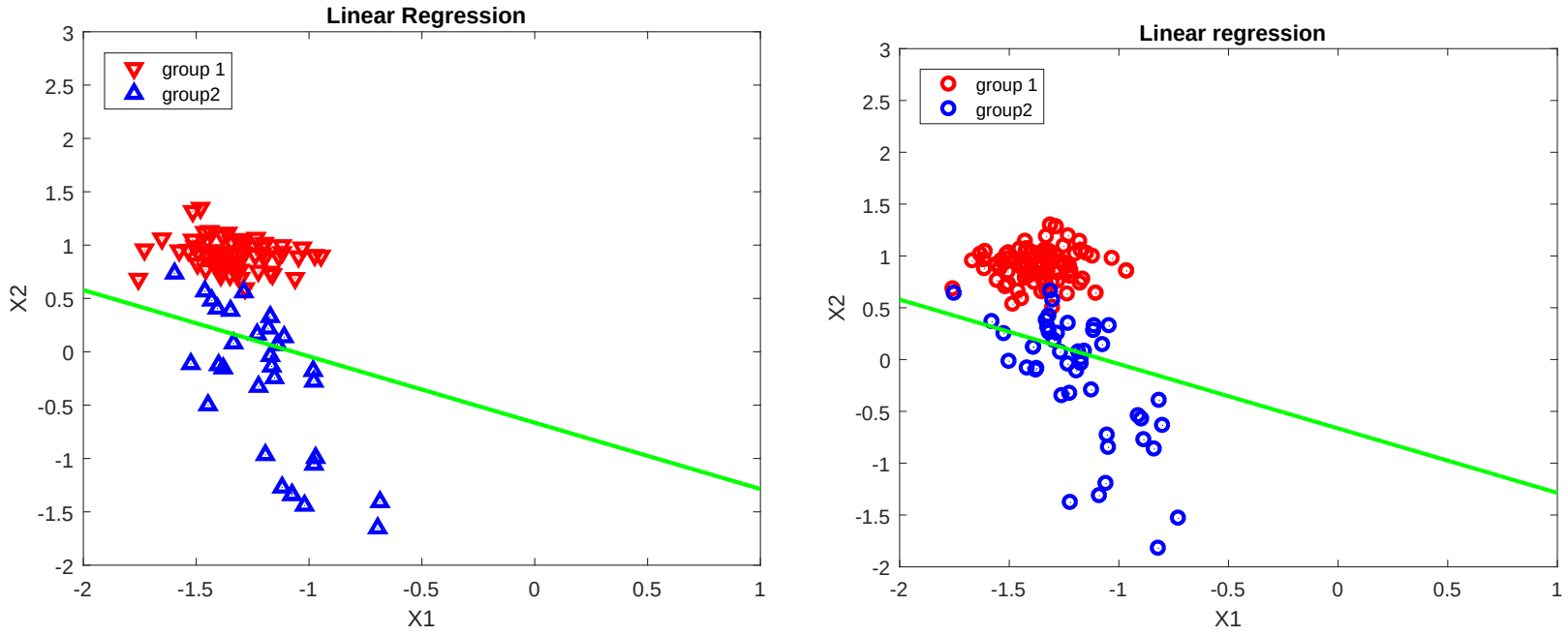
Classification boundary shown by the line. Training X_i's from two groups displayed in different colours. New observations falling below line will be classified in red group; others will be classified in black group.



Another example

☛ *Left*: training X_i's, different groups shown by blue/red. Green line shows decision boundary $\hat{\beta}_0+\hat{\beta}_1 x_1+\hat{\beta}_2 x_2 = 1/2$ constructed from training.

☛ *Right*: new data points (**test data**) to be classified. Unlike in real life, the truth (blue/red) is shown to illustrate how the classifier performs. Data classified in one group if below the line, in another if above the line. The blues above the line are the **misclassification errors**.



7.3.2 LOGISTIC REGRESSION CLASSIFIER

☛ Issue with the linear regression classifier $\hat{\beta}_0 + \hat{\beta}^T x$:

$\hat{P}(G = 1|X = x)$ is not guaranteed to be in $[0, 1]$,
but it tries to estimate a probability.

☛ Suggestion: Use *logistic regression*, i.e. assume

$$m_1(x) = P(G = 1|X = x) = E(Y_1|X = x) = \frac{\exp(\beta_0 + \beta^T x)}{1 + \exp(\beta_0 + \beta^T x)}.$$

☛ Estimate β_0, β using *maximum likelihood* (ML) estimators $\hat{\beta}_0, \hat{\beta}$ computed from the training data

$$(X_1, Y_{11}), \ldots, (X_n, Y_{n1}).$$

(or replace the data by their PCA or PLS components if needed).

☛ NOTE: Again, the method relies on the logistic model assumption which is only an approximation in real life.

- Since $m_2 = 1 - m_1$, in this model:

$$m_1(x) = P(G = 1|X = x) = \frac{\exp(\beta_0 + \beta^T x)}{1 + \exp(\beta_0 + \beta^T x)}$$
$$m_2(x) = P(G = 2|X = x) = 1 - m_1(x) = \frac{1}{1 + \exp(\beta_0 + \beta^T x)}.$$

- Classify new x in group 1 if

$$\hat{P}(G = 1|X = x) > \hat{P}(G = 2|X = x)$$
$$\iff \exp(\hat{\beta}_0 + \hat{\beta}^T x) > 1 \iff \hat{\beta}_0 + \hat{\beta}^T x > 0,$$

otherwise classify x in group 2.

- The classification boundary is obtained by solving the equation

$$\widehat{m}_1(x) - \widehat{m}_2(x) = 0 \iff \exp(\hat{\beta}_0 + \hat{\beta}^T x) = 1 \iff \hat{\beta}_0 + \hat{\beta}^T x = 0.$$

Golub data revisited

- From the estimated logistic model, we have

$$\widehat{m}_1(x) = \frac{\exp\left(0.9432 + 0.5487x_1 - 10.714x_2\right)}{1 + \exp\left(0.9432 + 0.5487x_1 - 10.714x_2\right)}.$$

- The classification boundary is obtained by solving

$$0.9432 + 0.5487x_1 - 10.714x_2 = 0,$$

which we can express by the line

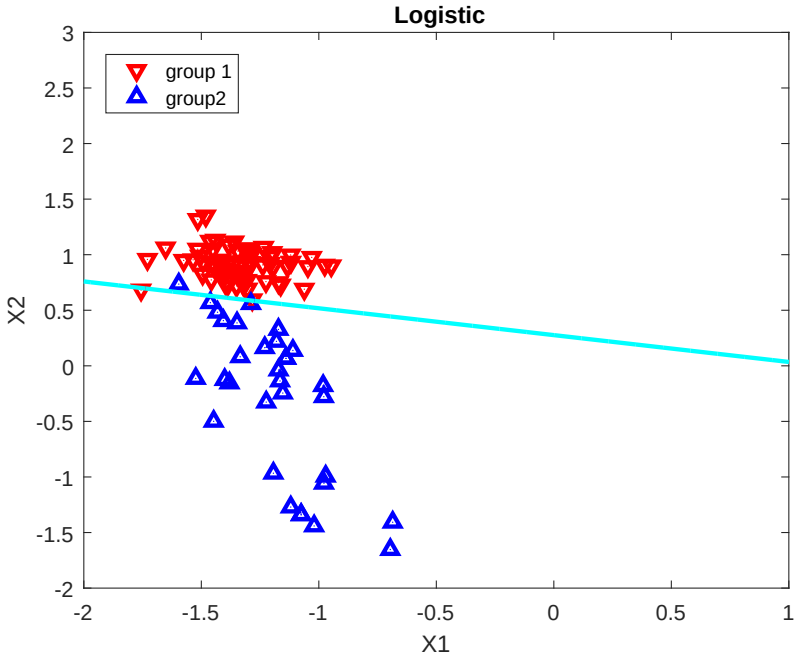
$$x_2 = \frac{0.9432}{10.714} + \frac{0.5487}{10.714}x_1 = 0.08803435 + 0.05121337x_1$$

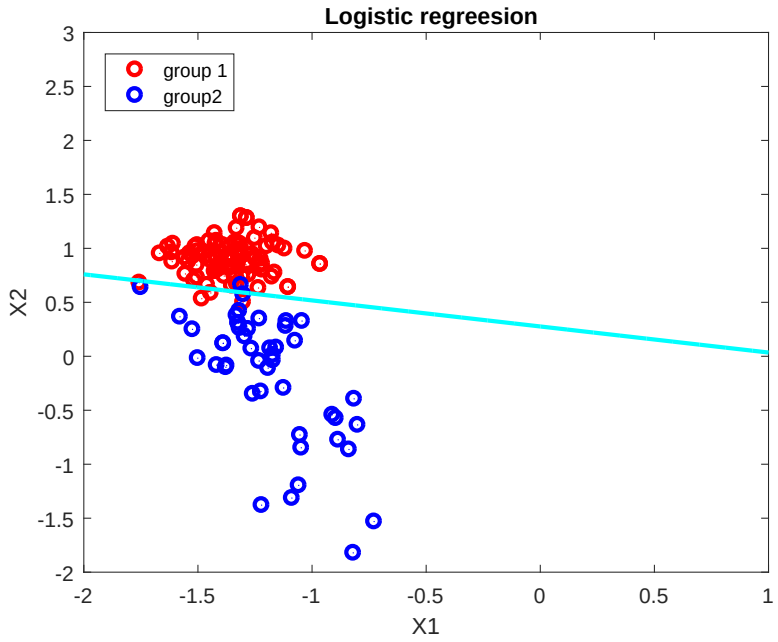
Another example revisited

☛Again, the left and right pictures correspond to the X_i's of the *training* and *test* data, with their group identities shown in red and blue. As before, the truth of the test data are revealed to check how the classifier performs.

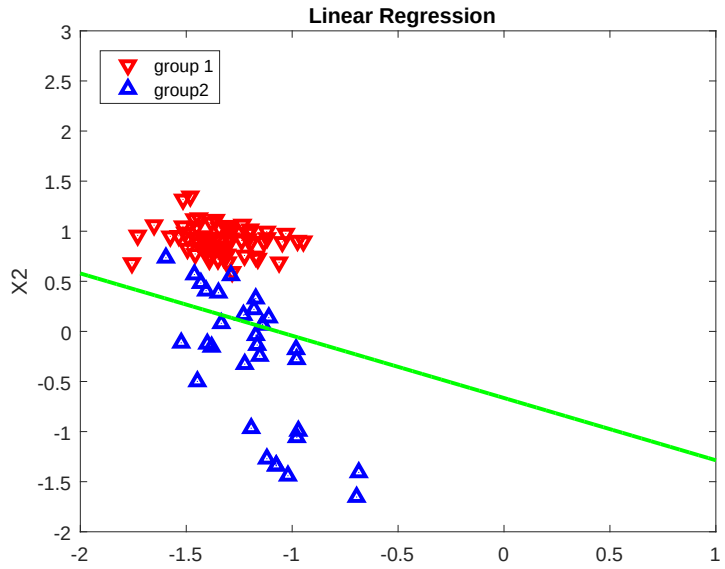
☛ Cyan: Decision boundary $\hat{\beta}_0+\hat{\beta}_1 x_1+\hat{\beta}_2 x_2 = 0$ constructed from training X_i's on the left based on *logistic regression*.

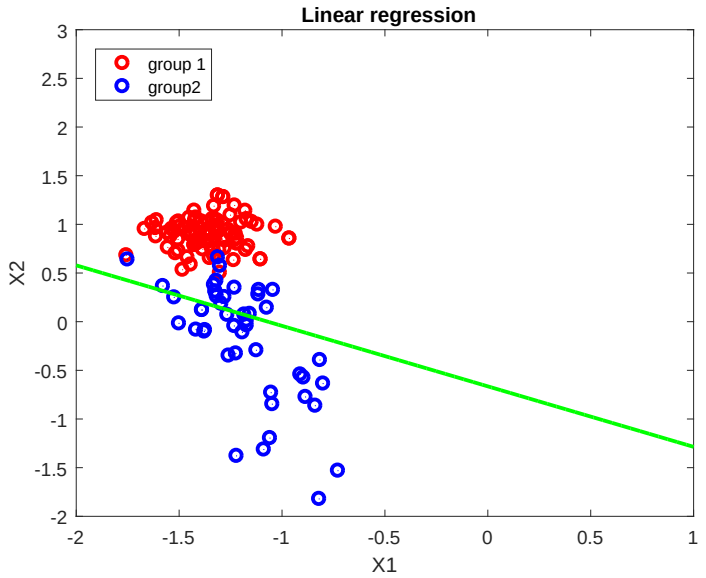
☛ *Right* picture: Data classified in one group if below the line, in another if above line

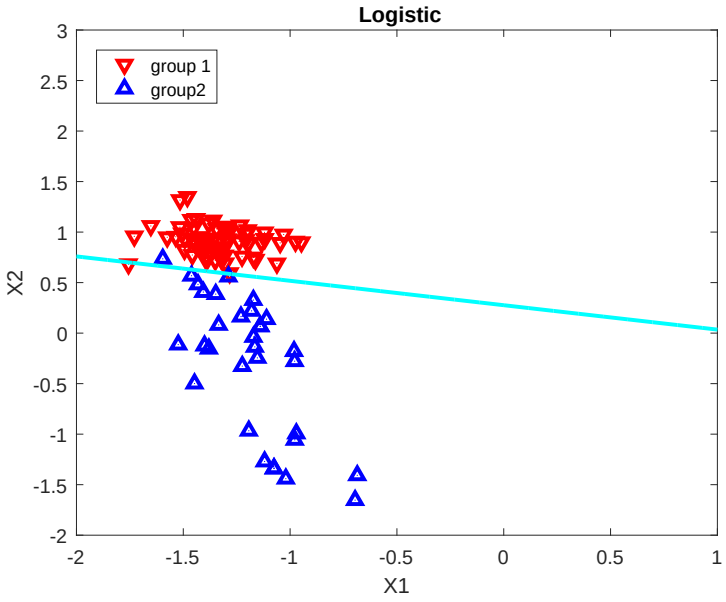


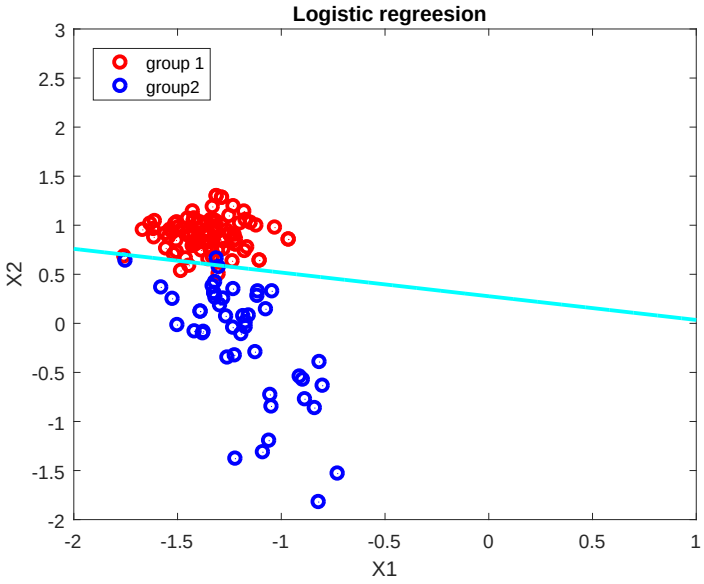


Comparison: Less classification errors for the logistic regression classifier









- Alternatively, we can classify by Bayes methods.

- Using **Bayes theorem** we have

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

where

$$\pi_k = P(G = k).$$

- Estimate these conditional probabilities from training data.

7.4.1 LINEAR DISCRIMINANT (LD)

☛ For the **linear discriminant** method, we assume that $P(X = x|G = k)$ is the density of a $N_p(\mu_k, \Sigma)$.

☛ Recall that p-dimensional normal density:

$$f_k(x) = (2\pi)^{-p/2}\{det(\Sigma)\}^{-1/2}\exp\{-0.5(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\}\,.$$

☛ Under the Bayes theorem, a new individual with $X = x$ is more likely to come from group 1 than group 2 iff

$$\begin{aligned}
&P(G=1|X=x) > P(G=2|X=x)\\
\iff\ &P(X=x|G=1)\pi_1 > P(X=x|G=2)\pi_2\\
\iff\ &\exp\{-0.5(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\}\pi_1 > \exp\{-0.5(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\}\pi_2\\
\iff\ &2\log(\pi_1/\pi_2) + 2x^T\Sigma^{-1}(\mu_1-\mu_2) - \mu_1^T\Sigma^{-1}\mu_1 + \mu_2^T\Sigma^{-1}\mu_2 > 0.
\end{aligned}$$

Classify x in group 1 if above is satisfied and in group 2 otherwise.

☛ The decision rule

$$2\log(\pi_1/\pi_2) + 2x^T\Sigma^{-1}(\mu_1 - \mu_2) - \mu_1^T\Sigma^{-1}\mu_1 + \mu_2^T\Sigma^{-1}\mu_2 > 0$$

is a **linear** function in x.

☛ NOTE: Relies on normality and homoscedasticity assumptions, i.e. Σ is the same for both groups, which are often not satisfied in practice

$\Rightarrow$ it only provides an approximation to the truth.

☛ In practice, we do not know π_k, μ_k and Σ; but we can estimate them using the training data, resulting in the classification boundary:

$$2\log(\hat{\pi}_1/\hat{\pi}_2) + 2x^T\hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_2) - \hat{\mu}_1^T\hat{\Sigma}^{-1}\hat{\mu}_1 + \hat{\mu}_2^T\hat{\Sigma}^{-1}\hat{\mu}_2 = 0.$$

☛ Estimation of π_k:

$$\hat{\pi}_k = n_k/n,$$

where n_k= number of training data from group k.

☛ NOTE: Estimating π_k with n_k/n assumes that the group sizes in the training data are *representative of the group sizes in the population*. If we are not sure about that, one may alternatively often use

$$\hat{\pi}_k = 1/K,$$

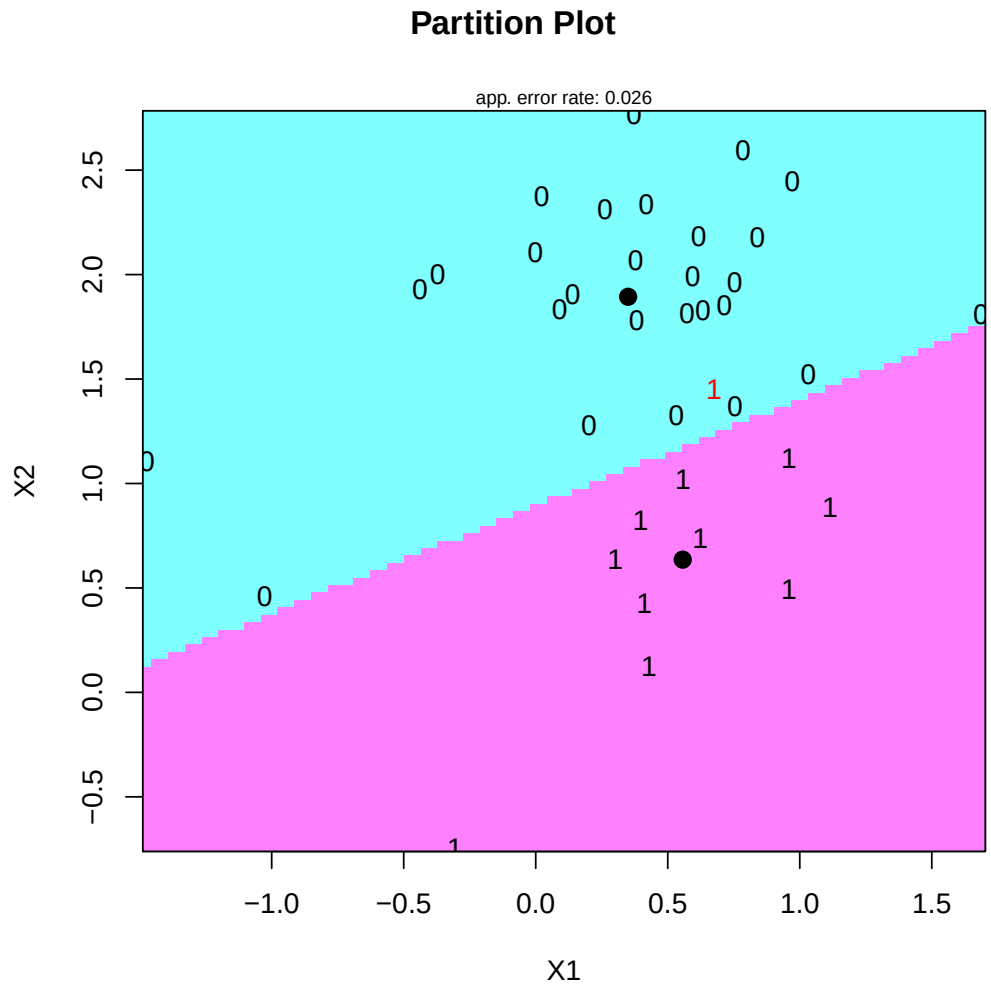
corresponding to an *objective prior* on the group probabilities.
(When $K = 2$, $\hat{\pi}_1 = \hat{\pi}_2 = 1/2$).

☛ Estimation of μ_k and Σ: let $X_{i,k}$, $i = 1, \ldots, n_k$ denote all the X_i's from group k. Then

$$\hat{\mu}_k = \overline{X}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} X_{i,k}\ , \quad \widehat{\Sigma} = \frac{1}{n_1 + n_2 - 2}\sum_{k=1}^{2}\sum_{i=1}^{n_k}(X_{i,k} - \widehat{\mu}_k)(X_{i,k} - \widehat{\mu}_k)^T.$$

Golub data revisited

☛ Two categories of patients, corresponding to two types of leukemia (”ALL” and ”AML”); $X = (X_1, X_2)^T$: expressions of two genes.

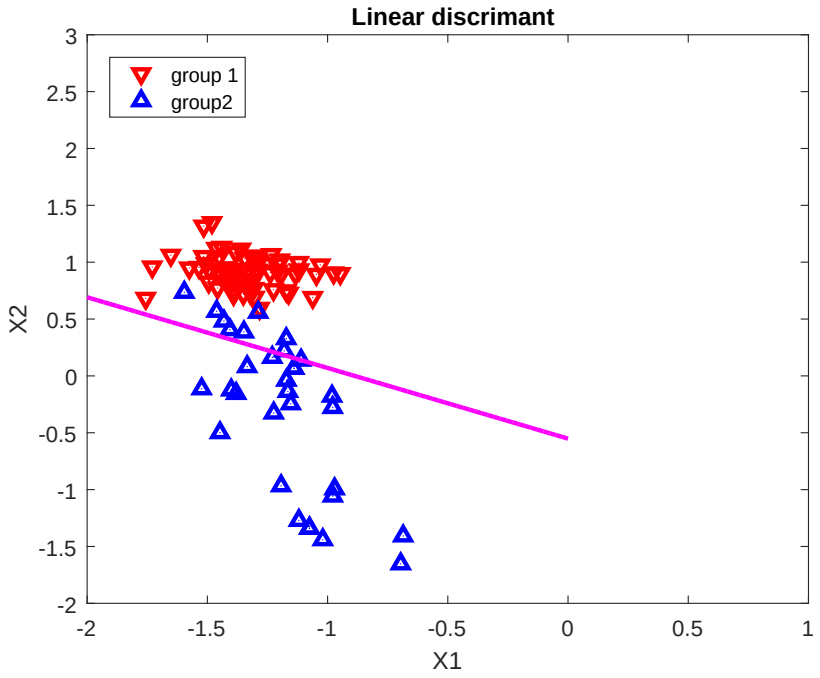


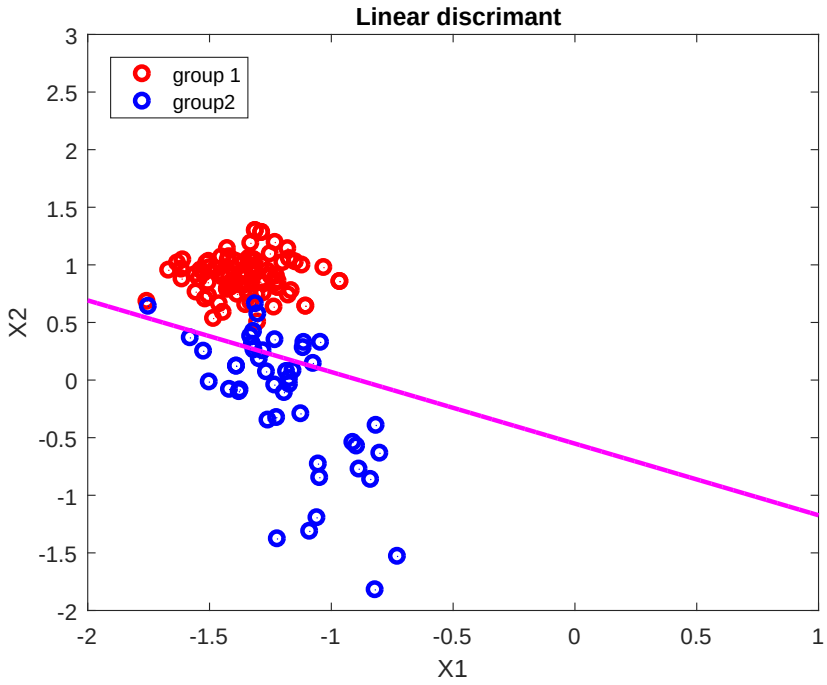
Another example revisited

☛ Again, the left and right pictures correspond to the X_i's of the *training* and *test* data, with their group identities shown in red and blue.

☛ Magenta: The classification boundary $2\log(\widehat{\pi}_1/\widehat{\pi}_2) + 2x^T\widehat{\Sigma}^{-1}(\widehat{\mu}_1 - \widehat{\mu}_2) - \widehat{\mu}_1^T\widehat{\Sigma}^{-1}\widehat{\mu}_1 + \widehat{\mu}_2^T\widehat{\Sigma}^{-1}\widehat{\mu}_2 = 0$ constructed from training X_i's.

☛ *Right* picture: Data classified in one group if below the line, in another if above line.





Another example revisited

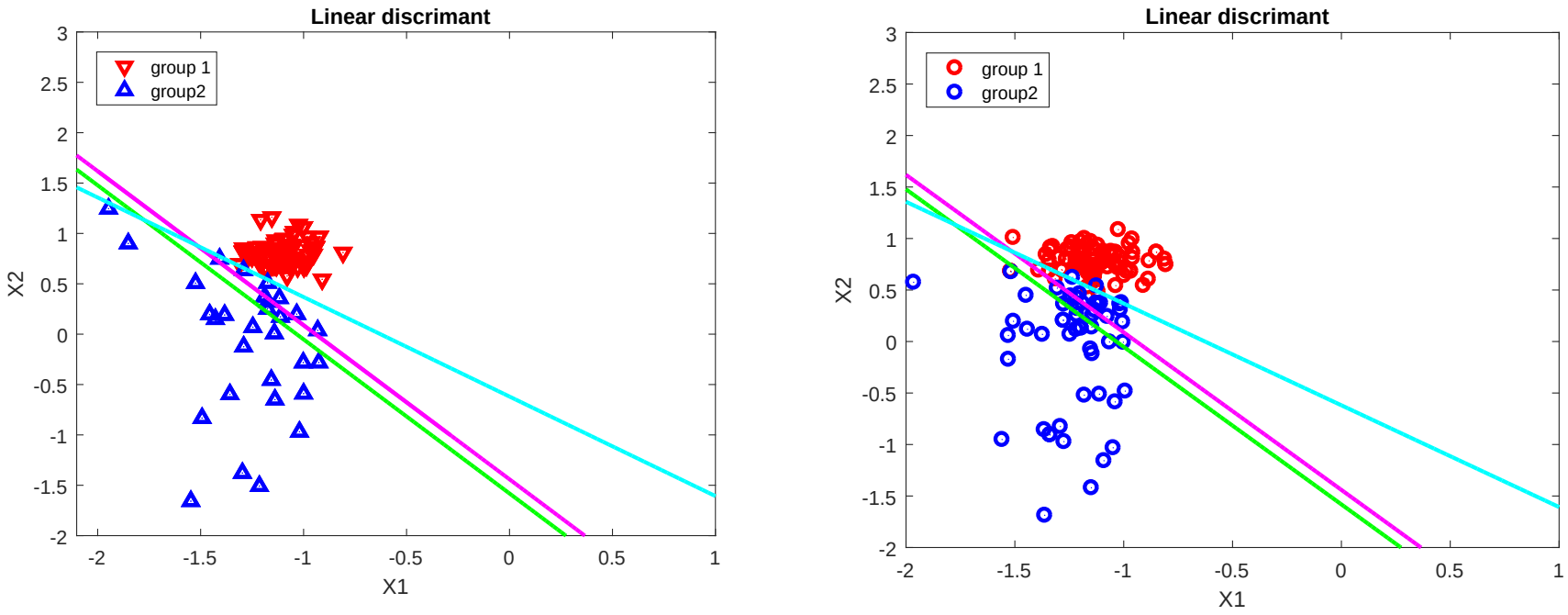


Figure 1: fdf

Logistic regression (cyan) performs the best for this "another example".

Classification boundaries

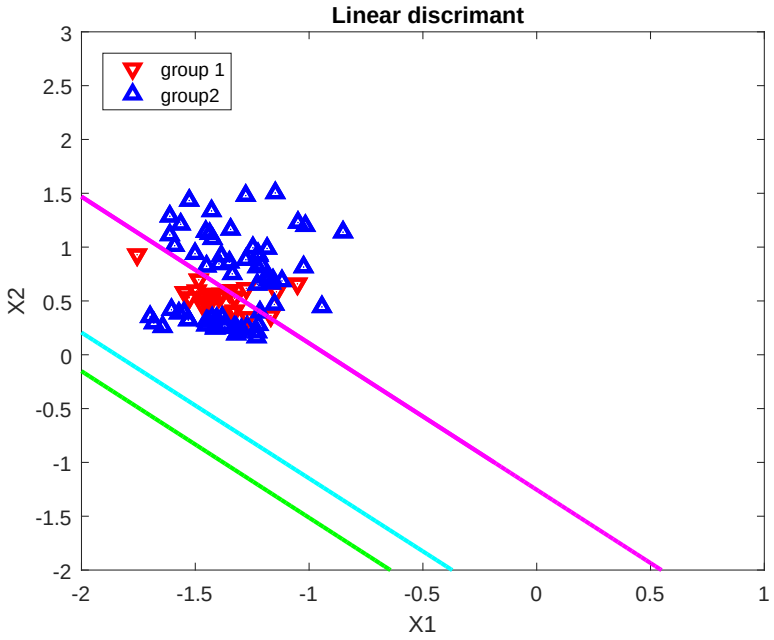
All three methods have *linear* classification boundary:

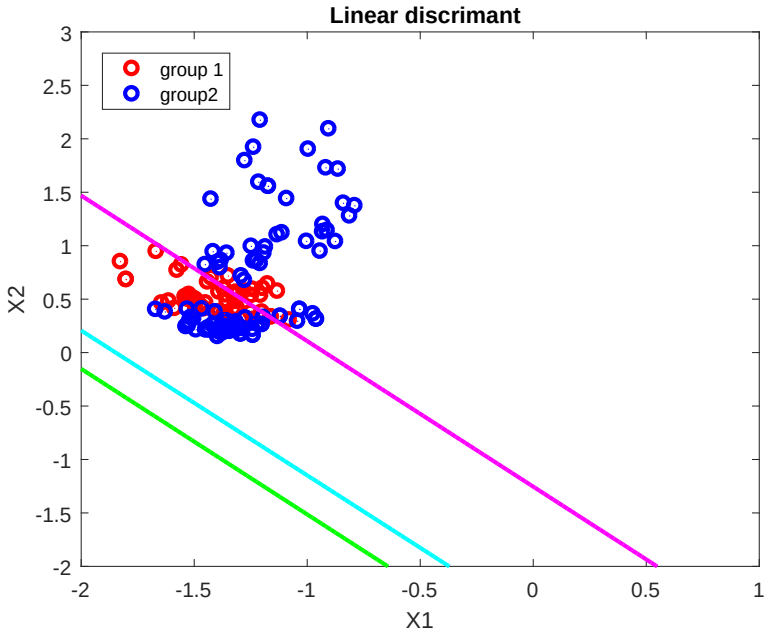
- LD case: two classes are on different side of
$$\hat{\beta}_0 + x^T \hat{\beta} = 0.$$
- Linear regression: two classes are on different side of
$$\hat{\beta}_0 + x^T \hat{\beta} = 1/2.$$
- Logistic regression: two classes are on different side of
$$\hat{\beta}_0 + x^T \hat{\beta} = 0.$$
- They differ through the way $\hat{\beta}_0$ and $\hat{\beta}$ are constructed.

- All three methods rely on model assumptions that only approximates the truth. Often, logistic regression works better than the other two methods.

Yet another example of linear methods NOT working:

- **left**: training data, **right**: test data
- Lines correspond to classification boundaries of
  - linear regression
  - logistic regression
  - linear discriminant
- None of the methods separate the two classes well on the right.





7.4.2 QUADRATIC DISCRIMINANT (QD)

☛ In the **quadratic discriminant** method, we assume that $X|G = k$ is a p-variate normal, but both the mean and the covariance are different for $k = 1$ and $k = 2$, i.e. Σ_k **heteroskedastic**

☛ $P(X = x|G = k)$, is the density

$$f_k(x) = (2\pi)^{-p/2}\{det(\Sigma_k)\}^{-1/2}\exp\{-0.5(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k)\}\,.$$

☛ Under the Bayes theorem, a new individual with $X = x$ is more likely to come from group 1 than group 2 iff

$$\begin{aligned}
&P(G=1|X=x) > P(G=2|X=x) \iff \\
&\iff P(X=x|G=1)\pi_1 > P(X=x|G=2)\pi_2 \\
&\iff \{det(\Sigma_1)\}^{-1/2}\exp\{-0.5(x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1)\}\pi_1 \\
&\qquad > \{det(\Sigma_2)\}^{-1/2}\exp\{-0.5(x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)\}\pi_2 \\
&\iff -2\log(\pi_1) + \log\{det(\Sigma_1)\} + (x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) \\
&\qquad < -2\log(\pi_2) + \log\{det(\Sigma_2)\} + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2).
\end{aligned}$$

☛ The decision rule

$$-2\log(\pi_1) + \log\{det(\Sigma_1)\} + (x-\mu_1)^T\Sigma_1^{-1}(x-\mu_1) \\ < -2\log(\pi_2) + \log\{det(\Sigma_2)\} + (x-\mu_2)^T\Sigma_2^{-1}(x-\mu_2)$$

is a **quadratic** function of x, and hence the name.

☛ In practice we estimate the unknown μ_k and Σ_k by

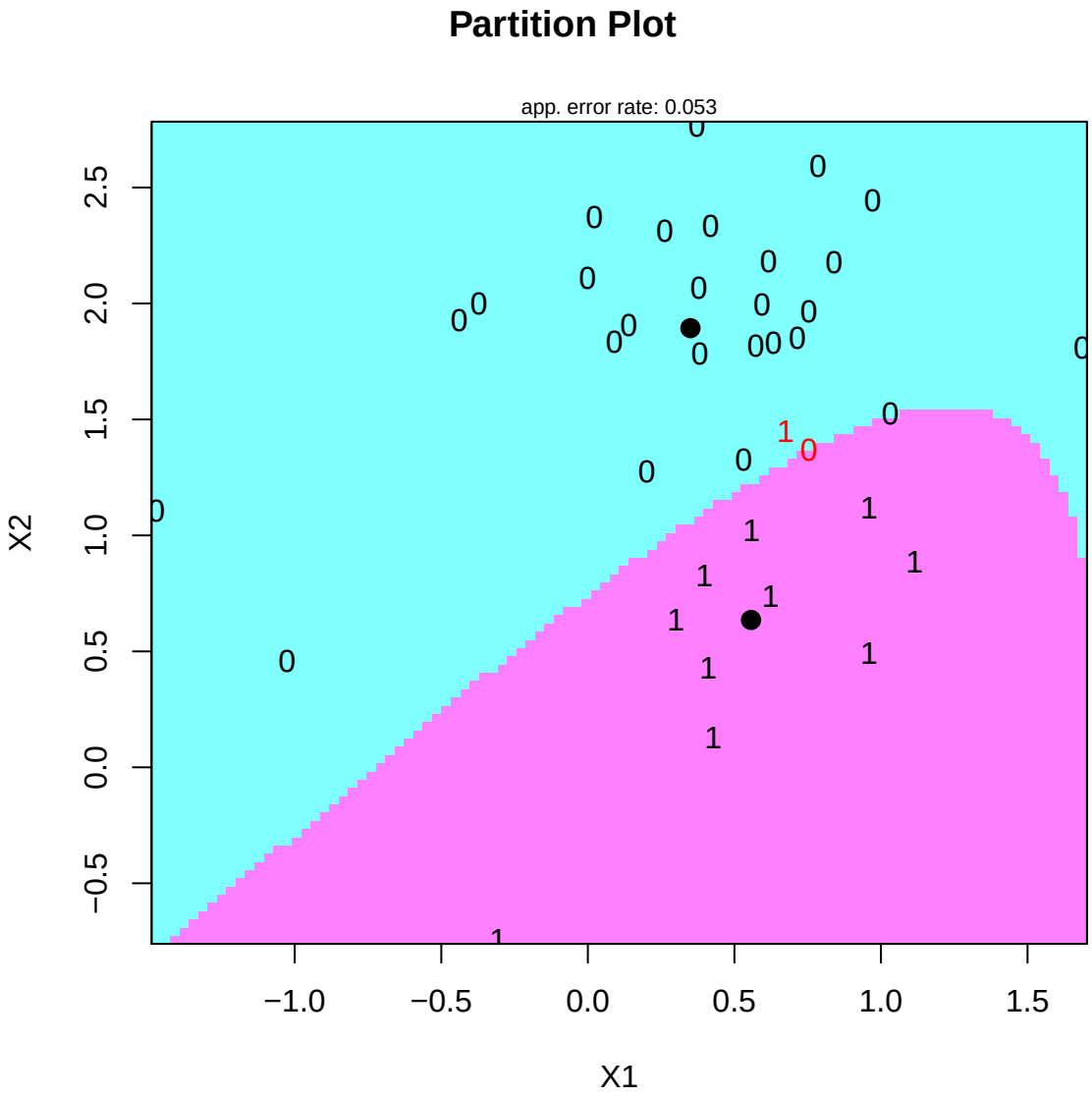
$$\hat{\mu}_k = \overline{X}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} X_{i,k}$$

$$\widehat{\Sigma}_k = \frac{1}{n_k - 1}\sum_{i=1}^{n_k}(X_{i,k} - \widehat{\mu}_k)(X_{i,k} - \widehat{\mu}_k)^T.$$

☛ If we know the group sizes in the training data reflect those in the population, we can estimate π_k by $\hat{\pi}_k = n_k/n$, where n_k= number of training data from group k. Otherwise, one can take $\hat{\pi}_k = 1/K$.

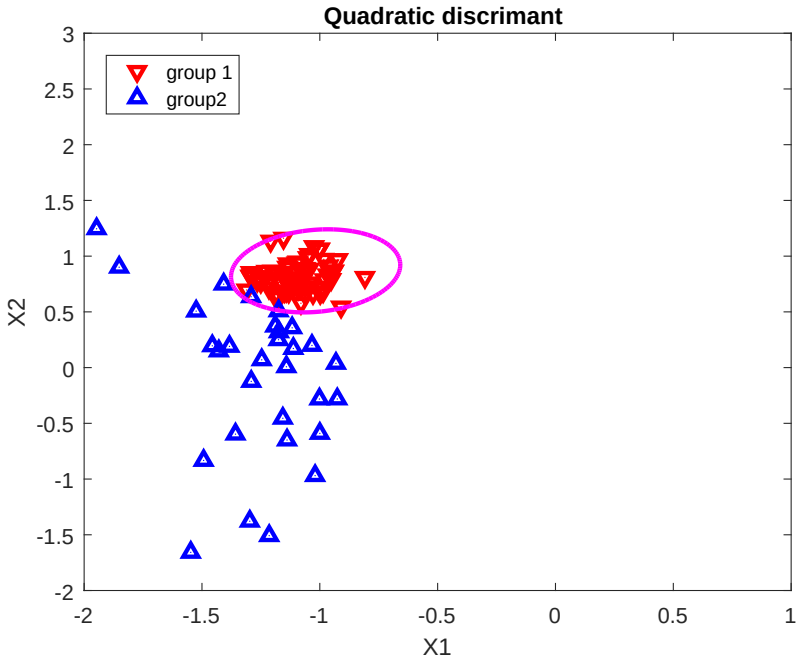
Golub data revisited

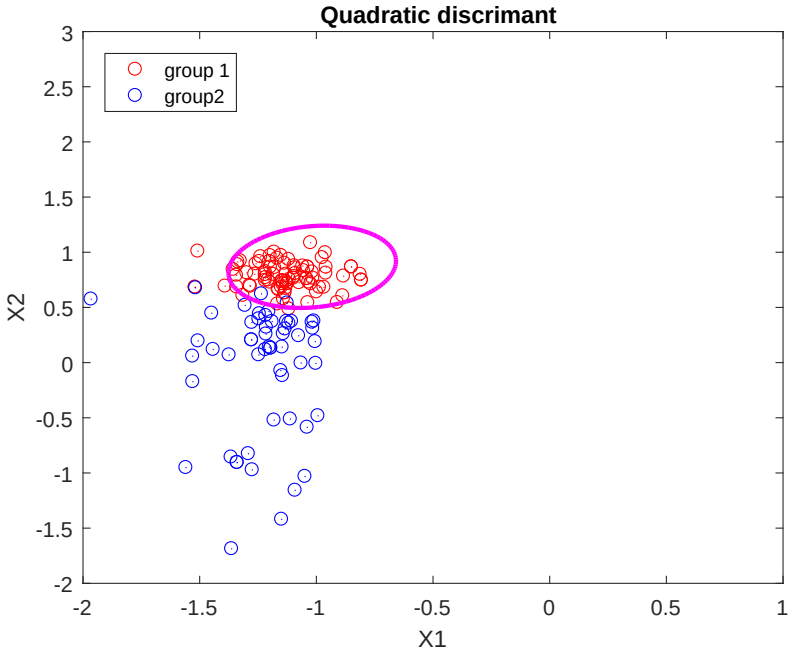
☛ Two categories of patients, corresponding to two types of leukemia ("ALL" and "AML"); $X = (X_1, X_2)^T$: expressions of two genes.



Another example revisited

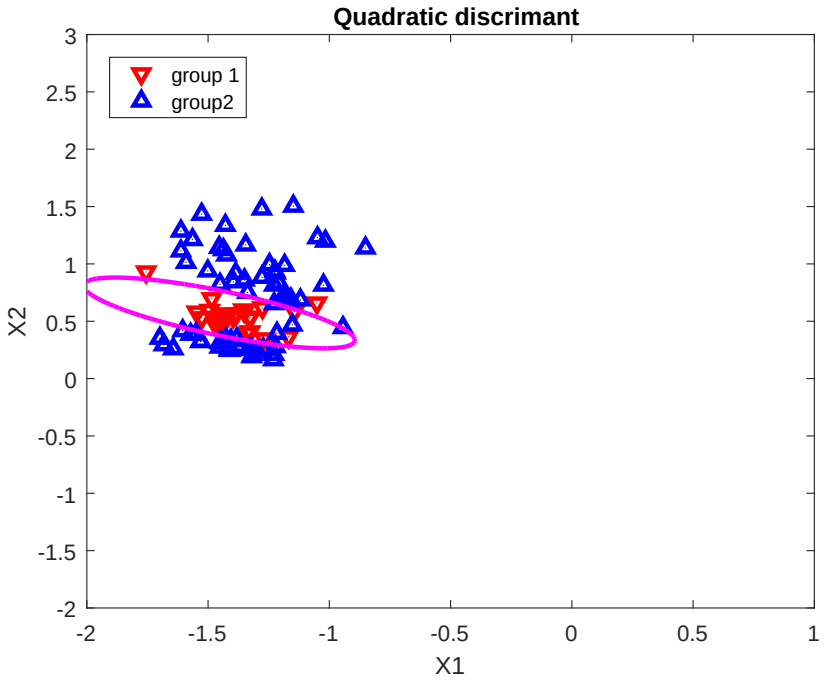
The eclipse shows the classification boundary of quadratic discriminant

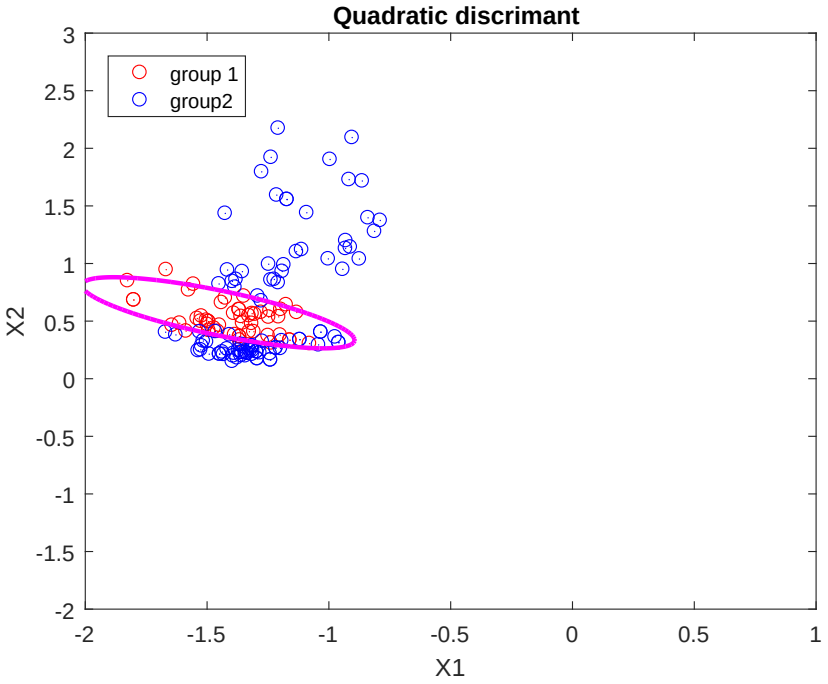




Yet another example of linear methods NOT working, revisited

The eclipse shows the classification boundary of quadratic discriminant





7.4.3 REGULARISED ESTIMATORS

☛ Looks like QD works better than the linear methods. Why not always use it?

☛ Difficulty with QD: need to estimate K covariance matrices.

☛ If p, the dimension of X, is large, estimating several covariance matrices implies estimating many more parameters

$\Rightarrow$ *noisy* (and hence unreliable) estimated decision boundary.

☛ **Regularised Discriminant analysis** (RD), make a tradeoff between LD and QD by taking, for group k,

$$\tilde{\Sigma}_k = \alpha\hat{\Sigma}_k + (1-\alpha)\hat{\Sigma},$$

where $\alpha \in [0,1]$ is a tuning parameter. Larger α means the resulting model is closer to QD than LD. Typically, α is chosen by cross-validation (*CV*).

☛ For LD: even Σ may involve too many parameters to estimate if p is large. One can further regularize to trade off between $\hat{\Sigma}$ and a constant variance for all components of X:

$$\tilde{\Sigma} = \alpha\hat{\Sigma} + (1-\alpha)\hat{\sigma}^2 I_p,$$

where $\alpha \in [0,1]$ is a tuning parameter to be chosen by CV.

☛ Alternative way to regularize: replace X with their PCA or PLS components; the number of components q can be chosen by CV.

7.4.4 CROSS-VALIDATION FOR CLASSIFICATION

☛ $G_1, \ldots, G_n$: the true group identities of the n training individuals.

☛ $\hat{G}$: a generic classifier that assigns an individual with X value to the group $\hat{G}$, which is a function of X. In particular, denote

- $\hat{G}^{(-i)}$: one such classifier trained *without* the i-th sample in the training data.
- $\hat{G}_i^{(-i)}$: the group classification of applying $\hat{G}^{(-i)}$ to the i-th training sample.

☛ Leave-one-out CV (**LOOCV**) estimator of the classification error:

$$CV = \frac{1}{n} \sum_{i=1}^{n} I(\hat{G}_i^{(-i)} \neq G_i).$$

☛ We can also use B**-fold cross validation**:

1. Randomly split the training data into B blocks of roughly equal sizes.

2. For $b = 1, \ldots, B$, train the classifier $\hat{G}^{(-b)}$ that does not use the b-th block of training data.

3. We apply $\hat{G}^{(-b)}$ to all data of the bth block: $\hat{G}_i^{(-b)} = \hat{G}^{(-b)}(X_i)$ denotes the classification of ith sample in the bth block

4. Sum over all blocks to get

$$CV = \frac{1}{n} \sum_{b=1}^{B} \sum_{i=1}^{n} I(\hat{G}_i^{(-b)} \neq G_i) \cdot I\{i\text{th sample is in block } b\}$$

☛ More *computationally efficient*: we can reuse the same classifier $\hat{G}^{(-b)}$ several times.

☛ Regardless of using "leave-one-out" or "B-fold", the estimated CV classification error is implicitly a function of the tuning parameter used in the trained classifier.

☛ For example, if a tuning parameter is used, then $CV = CV(\alpha)$. Similarly, if PCA/PLS is used as a means of regularization, $CV = CV(q)$.

☛ Our final choice for the classifier will be based on the tuning parameter that minimizes the CV estimates, e.g. If

$$\hat{\alpha} = \text{argmin}_{\alpha \in [0,1]} CV(\alpha),$$

then the final RD classifier will have $\tilde{\Sigma}_k$ based on this particular value $\hat{\alpha}$.

☛ We also pick the $\hat{q}$ similarly for PCA/PLS regularized classifiers.

- i.i.d data: $(X_1, G_1), \ldots, (X_n, G_n)$, $G_i = 1, 2, \ldots, K$ (class labels).
- Recode the G_i's: for $k = 1, \ldots, K$, let

$$Y_{ik} = I\{G_i = k\} = \begin{cases} 1 & \text{if } G_i = k \\ 0 & \text{otherwise.} \end{cases}$$

- For example, if $K = 3$ and $n = 5$ and we observe

$$(G_1, G_2, G_3, G_4, G_5) = (2, 3, 1, 1, 2)$$

then we create

$$\begin{pmatrix} Y_{11} \\ Y_{21} \\ Y_{31} \\ Y_{41} \\ Y_{51} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} Y_{12} \\ Y_{22} \\ Y_{32} \\ Y_{42} \\ Y_{52} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} Y_{13} \\ Y_{23} \\ Y_{33} \\ Y_{43} \\ Y_{53} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

- Define K regression curves:

$$m_k(x) = E\big(Y_k|X = x\big) = P(G = k|X = x),\ k = 1, \ldots, K.$$

- Since x must belong to one (and only one) of the K groups, we have

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

- Thus we have

$$m_K(x) = 1 - \sum_{j=1}^{K-1} m_j(x).$$

We only need to estimate $m_1, \ldots, m_{K-1}$ by $\hat{m}_1, \ldots, \hat{m}_{K-1}$, and take

$$\hat{m}_K(x) = 1 - \sum_{j=1}^{K-1} \hat{m}_j(x)$$

- Classify new x in group with largest $\hat{m}_k(x)$.

7.5.1 LINEAR REGRESSION FOR $K > 2$ CLASSES

☛ Recoded data: $(X_1, Y_{11}, \ldots, Y_{1K}), \ldots, (X_n, Y_{n1}, \ldots, Y_{nK})$.

☛ Define K linear regression curves: for $k = 1, \ldots, K$,

$$m_k(x) = E\big(Y_k | X = x\big) = \beta_{0k} + \beta_k^T x = \beta_{0k} + \beta_{1k} x_1 + \ldots + \beta_{pk} x_p.$$

☛ Estimate β_{jk}'s using LS, PLS, PCA or other. E.g. with standard LS, we have

$$(\hat{\beta}_{0k}, \hat{\beta}_{1k}, \ldots, \hat{\beta}_{pk})^T = (\mathcal{X}^T \mathcal{X})^{-1} \mathcal{X}^T Y_k$$

where

$$\mathcal{X} = \begin{pmatrix} 1 & X_{11} & \ldots & X_{1p} \\ \vdots & \vdots & & \vdots \\ 1 & X_{n1} & \ldots & X_{np} \end{pmatrix}, \quad Y_k = \begin{pmatrix} Y_{1k} \\ \vdots \\ Y_{nk} \end{pmatrix}.$$

☛ Compute $\hat{m}_1(x), \ldots, \hat{m}_K(x)$ where, for $k = 1, \ldots, K-1$,

$$\hat{m}_k(x) = \hat{\beta}_{0k} + \hat{\beta}_{1k} x_1 + \ldots + \hat{\beta}_{dk} x_p$$

and

$$\hat{m}_K(x) = 1 - \sum_{j=1}^{K-1} \hat{m}_j(x).$$

☛ Classify the new individual whose value of X is $x = (x_1, \ldots, x_p)^T$ in group $\hat{G}$ that gives the max value among $\hat{m}_1(x), \ldots, \hat{m}_K(x)$:

$$\widehat{G} = \arg \max_{k=1,\ldots,K} \widehat{m}_k(x).$$

7.5.2 MULTINOMIAL LOGISTIC REGRESSION FOR $K > 2$ CLASSES

- For a general $K > 2$, we assume

$$m_k(x) = E(Y_k|X = x) = P(G = k|X = x) = \frac{\exp(\beta_{k0} + \beta_k^T x)}{1 + \sum_{\ell=1}^{K-1} \exp(\beta_{\ell 0} + \beta_\ell^T x)},$$

for $k = 1, \ldots, K-1$

- Then we take

$$\begin{aligned} m_K(x) =& 1 - \sum_{k=1}^{K-1} m_k(x) = 1 - \frac{\sum_{k=1}^{K-1} \exp(\beta_{k0} + \beta_k^T x)}{1 + \sum_{\ell=1}^{K-1} \exp(\beta_{\ell 0} + \beta_\ell^T x)} \\ =& \frac{1}{1 + \sum_{\ell=1}^{K-1} \exp(\beta_{\ell 0} + \beta_\ell^T x)}. \end{aligned}$$

- It is easy to see that $m_1(x) + \ldots + m_K(x) = 1$, and m_k's are always non-negative.

☛ For $k = 1, \ldots, K-1$, the estimated parameters $\hat{\beta}_{k0}$, $\hat{\beta}_k$'s are obtained by maximizing the multinomial likelihood to construct the estimators $\hat{m}_k$ of m_k. Then the estimator for $m_K(x)$ is

$$\hat{m}_K(x) = 1 - \sum_{k=1}^{K-1} \hat{m}_k(x).$$

☛ Classify the new individual in group $\hat{G}$ which gives the max value among $\hat{m}_1(x), \ldots, \hat{m}_K(x)$:

$$\widehat{G} = \arg \max_{k=1,\ldots,K} \widehat{m}_k(x)$$

☛ In fact, when classifying, it suffices to compute

$$\log \frac{P(G=1|X=x)}{P(G=K|X=x)} = \log \frac{m_1(x)}{m_K(x)} = \beta_{10} + \beta_1^T x$$
$$\log \frac{P(G=2|X=x)}{P(G=K|X=x)} = \log \frac{m_2(x)}{m_K(x)} = \beta_{20} + \beta_2^T x$$
$$\vdots$$
$$\log \frac{P(G=K-1|X=x)}{P(G=K|X=x)} = \log \frac{m_{K-1}(x)}{m_K(x)} = \beta_{K-1,0} + \beta_{K-1}^T x.$$

☛ E.g. Suppose $K > 5$ and $P(G = 5|X = x)$ is the max. Then we will have

$$m_5(x) > m_K(x) \iff log\{m_5(x)/m_K(x)\} > 0$$

and for all $k \neq 5$,

$$m_k(x) < m_5(x) \iff \frac{m_k(x)}{m_K(x)} < \frac{m_5(x)}{m_K(x)} \iff \log \frac{m_k(x)}{m_K(x)} < \log \frac{m_5(x)}{m_K(x)}$$

so we are able to make our decision based only on these log ratios.

☛ As with $K = 2$, this is thus a linear method.

7.5.3 LD AND QD METHODS FOR $K > 2$ GROUPS

- Let $\pi_1 = P(G = 1), \ldots, \pi_K = P(G = K)$.

- LD: Classify new observed x in group k that maximises

$$\delta_k(x) = x^T \hat{\Sigma}^{-1} \hat{\mu}_k - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}^{-1} \hat{\mu}_k + \log \hat{\pi}_k.$$

- QD: Classify new observed x in group k that maximises

$$\delta_k(x) = \log(\pi_k) - \frac{1}{2} \log\{det(\hat{\Sigma}_k)\} - \frac{1}{2}(x - \hat{\mu}_k)^T \hat{\Sigma}_k^{-1}(x - \hat{\mu}_k).$$

☛ As with $K = 2$, in the training sample, let $X_{i,k}$, $i = 1, \ldots, n_k$ denote all the X_i's that are from group k;

☛ $\hat{\pi}_k = n_k/n$ is the proportion of training data that are from group k,

$$\hat{\mu}_k = \overline{X}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} X_{i,k}$$

$$\widehat{\Sigma}_k = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} (X_{i,k} - \widehat{\mu}_k)(X_{i,k} - \widehat{\mu}_k)^T$$

and

$$\widehat{\Sigma} = \frac{1}{n_1 + \ldots + n_K - K} \sum_{k=1}^{K} \sum_{i=1}^{n_k} (X_{i,k} - \widehat{\mu}_k)(X_{i,k} - \widehat{\mu}_k)^T.$$