Multivariate Statistics

Chapter 7: Classification Analysis

Pedro Galeano
Departamento de Estadística
Universidad Carlos III de Madrid
pedro.galeano@uc3m.es

Course 2017/2018

Master in Mathematical Engineering

Introduction

- The problem of classification is as follows:
  1. We have a set of objects (items, elements,...) that may come from two or more populations.
  2. We observe the value of a p-dimensional random variable $x = (x_1, \ldots, x_p)'$ on these objects.
  3. We want to classify a new object, with known values of the variables but with unknown population, in one of the populations.
- The techniques we will study here are also known as supervised classification, in order to indicate that we know a sample of well-classified objects that serves as information for the classification of subsequent objects.
- Supervised classification is a different problem than unsupervised classification or cluster analysis, seen in Chapter 6, where a sample of well-classified objects is not available.

Introduction

- Classification is one of the most popular problems in practical analysis nowadays:
  - In finance, the automatic credit scoring systems of financial institutions today are based on using many measurable variables (income, seniority in place of work, wealth,...) in order to predict future behavior.
  - In quality control, certain components should be classified as good or defective (lamps, televisions,...).
  - In engineering, it is important to design machines capable of automatic classification of voices, bills or coins, on-screen characters, or postal codes in letters.
  - Other examples are: assigning a written text of unknown origin to one of several authors using word frequency, assigning a musical score or painting to an artist, recognizing a tax declaration as potentially fraudulent or not, a business as a bankruptcy risk or not or a new manufacturing process as efficient or not.

Introduction

- The elements of a classification problem are the following:
  - A p-dimensional random variable, $x = (x_1, \ldots, x_p)'$, defined in a set of objects belonging to one out of G populations, P_g, for $g = 1, \ldots, G$.
  - An indicator variable, y, that takes value g, where $g \in \{1, \ldots, G\}$, if a randomly chosen object from one of the G populations belongs to group g.
  - The (unknown) probabilities, π_g, for $g = 1, \ldots, G$, that a randomly chosen object comes from the g-th population (obviously, $\pi_1 + \cdots + \pi_G = 1$).
  - A data matrix X of dimension $n \times p$, with observations $x_{i\cdot}$, for $i = 1, \ldots, n$, with known population membership, i.e., with known values of the indicator variable y.
- We are going to study the problem of classifying a new object with known values of x, say $x_0 = (x_{01}, \ldots, x_{0p})'$, in one of the G populations.

Introduction

- There are many possible classification techniques that one might use to classify an observation.
- Even if the problem has been traditionally analyzed by statisticians, several methods have been more recently proposed in the machine learning area.
- Here, we focus on three of the most widely-used statistical classifiers:
  - k-Nearest Neighbors (k-NN);
  - Bayes rule classifiers; and
  - Logistic regression.

k-Nearest Neighbors (k-NN)

- k-Nearest Neighbors (k-NN) is probably one of the simplest methods to perform classification.
- k-NN can be considered a non-parametric method because it does not require any distributional assumption on the random variable x.
- Indeed, k-NN can be used with data sets with any kind of variables, as long as a distance between observations exists.
- Moreover, this method does not require to estimate the probabilities $\pi_1, \ldots, \pi_G$.

k-Nearest Neighbors (k-NN)

- Given the data matrix X, the k-NN algorithm runs as follows:
  1. Define a measure of distance adequate for the observed random variable x.
  2. Compute the distance between the observation x_0, corresponding to the object to classify, and all the observations in the data matrix X.
  3. Select the k closest observations to x_0 and compute the proportion of the k observations that belongs to each population.
  4. Then, classify x_0 in the population with largest proportion (ties are broken at random).
- A key point in the algorithm is the selection of an adequate k.
- Several alternatives are available, but here we focus on leave-one-out cross-validation.

k-Nearest Neighbors (k-NN)

- Cross-validation is a general methodology useful to evaluate the performance of statistical methods.
- Given an observed sample and a certain method (model) that depends on certain parameters, the most general cross-validation procedure is as follows:
  1. Split the observed sample in two sub-samples.
  2. Use the first sub-sample to estimate the parameters of the method (model).
  3. Use the second sub-sample to validate the performance of the method (model) with the estimated parameters obtained from the first sub-sample.
- Leave-one-out cross-validation is a particular example of cross-validation when the second sub-sample consists only of a single observation of the whole observed sample.
- The idea is to extract conclusions with the results corresponding to apply leave-one-out cross-validation n times, one for each observation in the whole observed sample.

k-Nearest Neighbors (k-NN)

- k-NN with the leave-one-out cross-validation procedure runs as follows:

1. Repeat the following steps for $k = 1$ to $k = k_{\max}$, for certain upper bound $k_{\max}$:

   a. Given k, skip one observation of X and use k-NN to classify this observation.

   b. Repeat step 1.*a* skipping all the observations in X one time.

   c. Obtain a contingency table with the results:

	Classify in P_1	$\cdots$	Classify in P_G
Belongs to P_1	n_{11}	$\cdots$	n_{1G}
$\vdots$	$\vdots$	$\ddots$	$\vdots$
Belongs to P_G	n_{G1}	$\cdots$	n_{GG}

   where n_{ij} is the amount of observations that, coming from P_i, are classified in P_j.

   d. Compute the misclassification rate given by:

$$MR = \frac{n_{12} + \cdots + n_{G,G-1}}{n} = \frac{\text{Total misclassified observations}}{\text{Total number of observations}}$$

2. Select the optimal k as the one that gives the minimum misclassification rate or the one most equilibrated one among the best performances.

k-Nearest Neighbors (k-NN)

- Note that k-NN does not take into account the number of observations coming from each population, nor the probabilities $\pi_1, \ldots, \pi_G$.
- If the data set is highly unbalanced, i.e., one or more of the populations have much more elements than others, k-NN (as well as other classification methods) might classify most of the observations to the most represented populations.
- Note that this will lead to very small misclassification rates, but this is only reflecting the unbalanced situation.
- For instance, we have a problem with 2 populations with 100 observations, 90 coming from P_1 and 10 coming from P_2.
- If we get a misclassification rate of 0.1 is because the method always classify in P_1.

k-Nearest Neighbors (k-NN)

- There is no a gold-standard solution to this problem, but probably the best thing to do is resampling the data set.
- There are two options:
  1. Add sampled copies at random of observations from the populations less represented (over-sampling); or
  2. Delete observations at random from the populations more represented (under-sampling).

Illustrative example (I)

- We apply the k-NN algorithm to the Iris data set with cross-validation for $k = 1, \ldots, 20$.
- For that, we use the Euclidean distance because all the variables are quantitative and measured in the same units of measurements.
- The optimal k is 14, although different runs of the algorithm can lead to different optimal k's because of the existence of ties.

Illustrative example (I)

- The contingency table for $k = 14$ is given by:

	Clas. in Setosa	Clas. in Versicolor	Clas. in Virginica
Setosa	50	0	0
Versicolor	0	48	2
Virginica	0	1	49

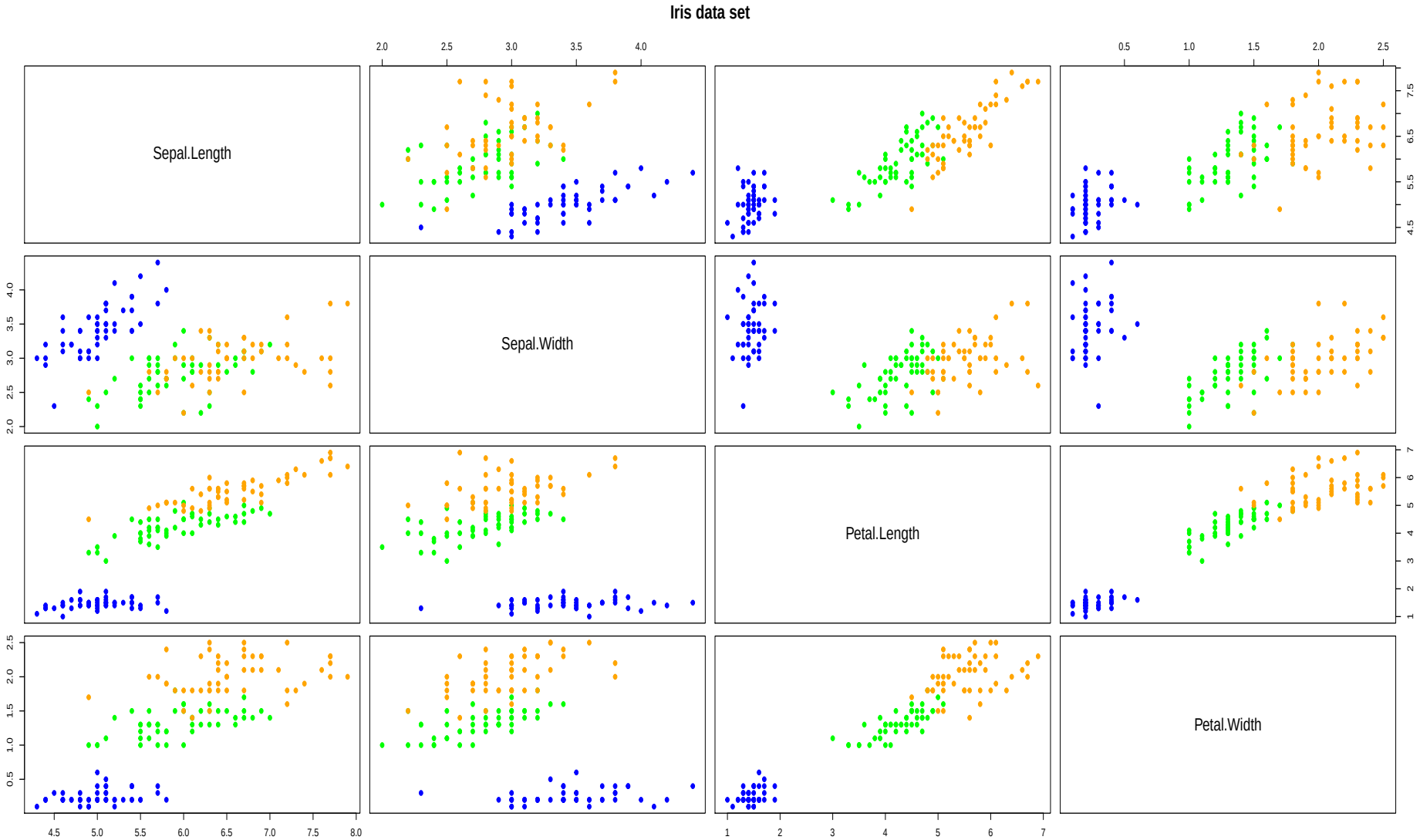
- Therefore, the misclassification error is estimated as:

$$\text{Error} = \frac{3}{150} = 0.02$$

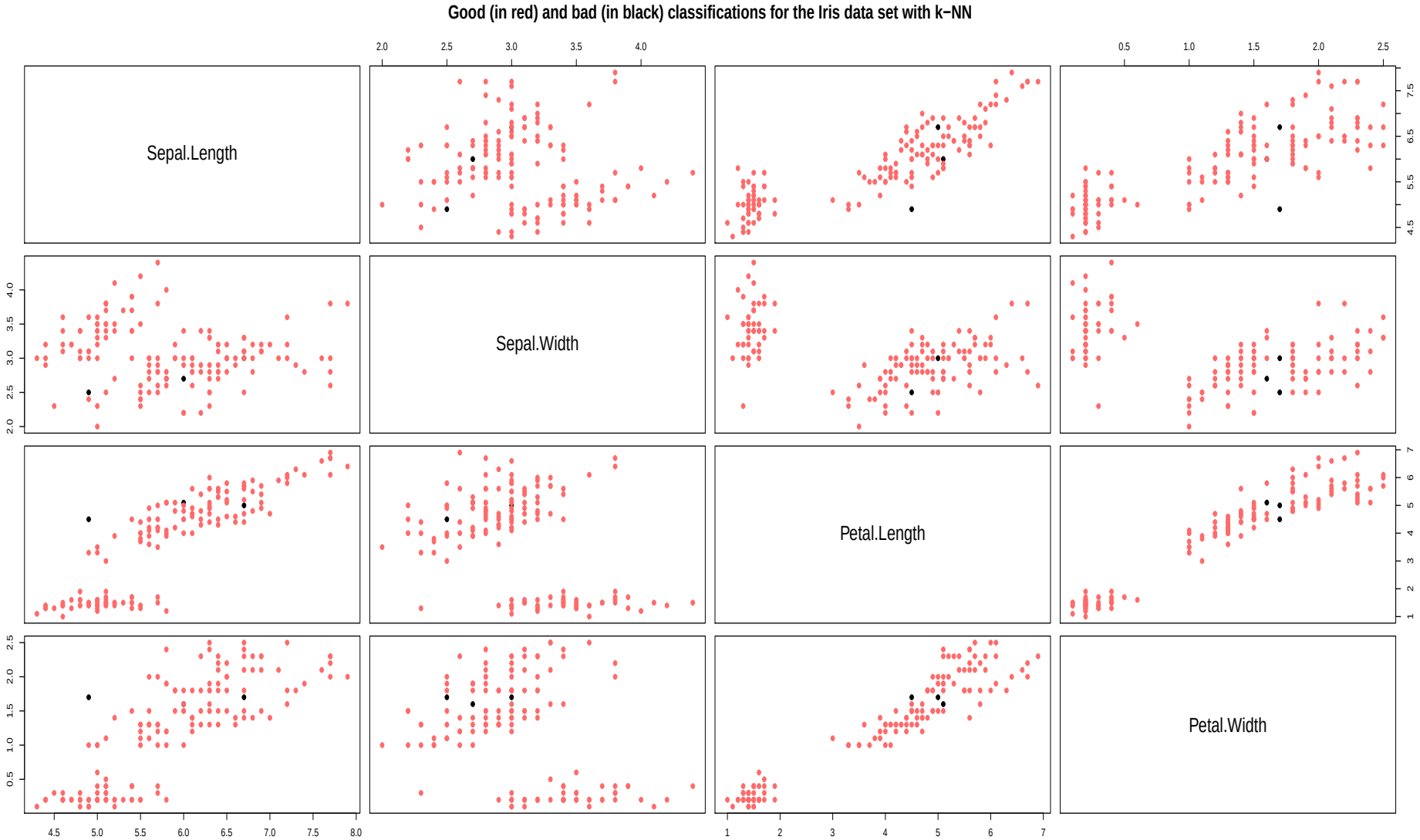
  which means that it is expected that the 2% of the new classifications are going to be wrong.

- The next two slides show the scatterplot matrix of the observations with the true populations and the scatterplot matrix of misclassified elements for $k = 14$.

Illustrative example (I)



Illustrative example (I)



Bayes rule classifiers

- Bayes rule classifiers are built up with the Bayes Theorem, thus they are under a probabilistic framework.
- The idea under the Bayes rule classifiers is to classify a new object in the population that has largest probability of having generated the associated observation.
- For easiness in presentation, assume that x is continuous and that the density functions of the G populations, denoted by $f_1, \ldots, f_G$, are known.
- Using the Bayes Theorem, the probability that the object with associated observation x_0 has been generated by the population P_g is given by:

$$\Pr(y = g | x = x_0) = \frac{\pi_g f_g(x_0)}{\sum_{k=1}^{G} \pi_k f_k(x_0)}$$

where π_g, for $g = 1, \ldots, G$, are the probabilities that a randomly chosen observation x_0 comes from the g-th population.

Bayes rule classifiers

- In other words, we classify the new object in P_g if $\Pr(y = g|x = x_0)$ is the maximum one among the G populations.
- Note that this is similar to the new object in P_g if $\pi_g f_g(x_0)$ is the maximum one among the G populations.
- In particular, if $\pi_1 = \cdots = \pi_G$, then, the condition for classifying in P_g is that $f_g(x_0)$ is the maximum one among the G populations, which means that we classify the new object in the population with largest density at the value x_0.

Bayes rule classifiers

- Next, we see what happens if we evaluate the Bayes rule classifier assuming that the density functions $f_1, \ldots, f_G$ are Gaussian.
- First, assume that $f_1, \ldots, f_G$ have different mean vectors, $\mu_1, \ldots, \mu_G$, but the same covariance matrix, Σ.
- Therefore, under P_g, x follows a $N(\mu_g, \Sigma)$ distribution.
- The optimal decision is, according to the Bayes rule, to classify the new element in the population P_g that maximize $\pi_g f_g(x_0)$, which is given by:

$$\pi_g f_g(x_0) = \pi_g (2\pi)^{-p/2} |\Sigma|^{-1/2} \exp\left(-\frac{(x_0 - \mu_g)' \Sigma^{-1} (x_0 - \mu_g)}{2} \right)$$

Bayes rule classifiers

- Taking logarithms and deleting some nuisance constants, this is equivalent to classify the new element in the population P_g that maximize:

$$2\log \pi_g - (x_0 - \mu_g)' \Sigma^{-1} (x_0 - \mu_g)$$

- Note that the last expression is equal to:

$$2\log \pi_g - x_0 \Sigma^{-1} x_0 + 2\mu_g' \Sigma^{-1} x_0 - \mu_g' \Sigma^{-1} \mu_g$$

- Consequently, the Bayes rule reduces to classify the new element in the population P_g that maximize:

$$p_g(x_0) = 2\mu_g' \Sigma^{-1} x_0 - \mu_g' \Sigma^{-1} \mu_g + 2\log \pi_g$$

because $x_0 \Sigma^{-1} x_0$ does not depend on the population P_g, so it can be skipped for classification purposes.

- As $p_g(x_0)$ depends linearly on x_0, the method is called the linear discriminant classifier.

Bayes rule classifiers

- Note that the linear discriminant classifier has an interesting interpretation if $\pi_1 = \cdots = \pi_G$.
- From the first formula in the previous slide, the linear discriminant rule classify the new element in the population P_g that maximize:

$$2 \log \pi_g - (x_0 - \mu_g)' \Sigma^{-1} (x_0 - \mu_g)$$

- If $\pi_1 = \cdots = \pi_G$, this is equivalent to classify the new element in the population P_g that minimize:

$$(x_0 - \mu_g)' \Sigma^{-1} (x_0 - \mu_g)$$

or, in other words, in the population whose mean vector is closest in terms of the squared Mahalanobis distance.

Bayes rule classifiers

- The linear discriminant classifier leads to the following expression of the probabilities $\Pr(y = g|x = x_0)$:

$$\Pr(y = g|x = x_0) = \frac{\pi_g \exp\left(-\frac{(x_0-\mu_g)'\Sigma^{-1}(x_0-\mu_g)}{2}\right)}{\sum_{k=1}^{G} \pi_k \exp\left(-\frac{(x_0-\mu_k)'\Sigma^{-1}(x_0-\mu_k)}{2}\right)} =$$

$$= \frac{\pi_g \exp\left(-\frac{1}{2} D_M^2(x_0, \mu_g)\right)}{\sum_{k=1}^{G} \pi_k \exp\left(-\frac{1}{2} D_M^2(x_0, \mu_k)\right)}$$

where $D_M^2(x_0, \mu_k) = (x_0 - \mu_k)' \Sigma^{-1} (x_0 - \mu_k)$ is the squared Mahalanobis distance between x_0 and μ_k.

Bayes rule classifiers

- Note that, in the development of the linear discriminant rule, we are assuming that we know the probabilities, $\pi_1, \ldots, \pi_G$, and the parameters of the Gaussian distributions, i.e., the means $\mu_1, \ldots, \mu_G$ and the common covariance matrix Σ.
- Obviously, in practice, this is not going to hold and, consequently, we need to estimate all these quantities.

Bayes rule classifiers

- For that, the data matrix X, of dimension $n \times p$, can be thought of as divided into G matrices corresponding to the G populations.
- Let x_{ijg} be the elements of these submatrices where i represents the individual, j the variable, and g the group or submatrix.
- Let $x_{i \cdot g}$ be the column vector that contains the p values of the variable x for the individual i in group g, that is, $x_{i \cdot g} = (x_{i1g}, \ldots, x_{ipg})'$.
- Let n_g be the number of elements in group g, such that the total number of observations is $n = \sum_{g=1}^{G} n_g$.

Bayes rule classifiers

- First, the probabilities, $\pi_1, \ldots, \pi_G$, can be estimated with the proportion of observed data in each group, i.e.,

$$\widehat{\pi}_g = \frac{n_g}{n}$$

- Second, the mean vector of x under P_g, i.e., μ_g, can be estimated with the sample mean vector within group g, i.e.

$$\overline{x}_g = \frac{1}{n_g} \sum_{i=1}^{n_g} x_{i \cdot g}$$

Bayes rule classifiers

- Third, the common covariance matrix of x, i.e., Σ, can be estimated with:

$$S_w = \sum_{g=1}^{G} \left(\frac{n_g - 1}{n - G} \right) S_g$$

where S_g is the sample covariance matrix for the elements of class g, i.e.:

$$S_g = \frac{1}{n_g - 1} \sum_{i=1}^{n_g} (x_{i \cdot g} - \overline{x}_g) (x_{i \cdot g} - \overline{x}_g)'$$

- Now, we can apply the linear discrimination rule and to estimate the probabilities $\Pr(y = g | x = x_0)$ after replacing the parameters of the Gaussian populations with their sample counterparts.

Bayes rule classifiers

- The question now is how to estimate the performance of the linear discrimination rule for a given data set.
- As with k-NN, we can use leave-one-out cross validation to achieve such goal.
- The idea is to classify each observation in the sample without including this element in the estimation step described before.
- Therefore, n classifications are performed from which we can obtain the corresponding contingency table and the associated misclassification rate.

Illustrative example (I)

- We use the linear discriminant rule to classify the flowers in the Iris data set using leave-one-out cross-validation.
- As with k-NN, there are only three misclassifications.

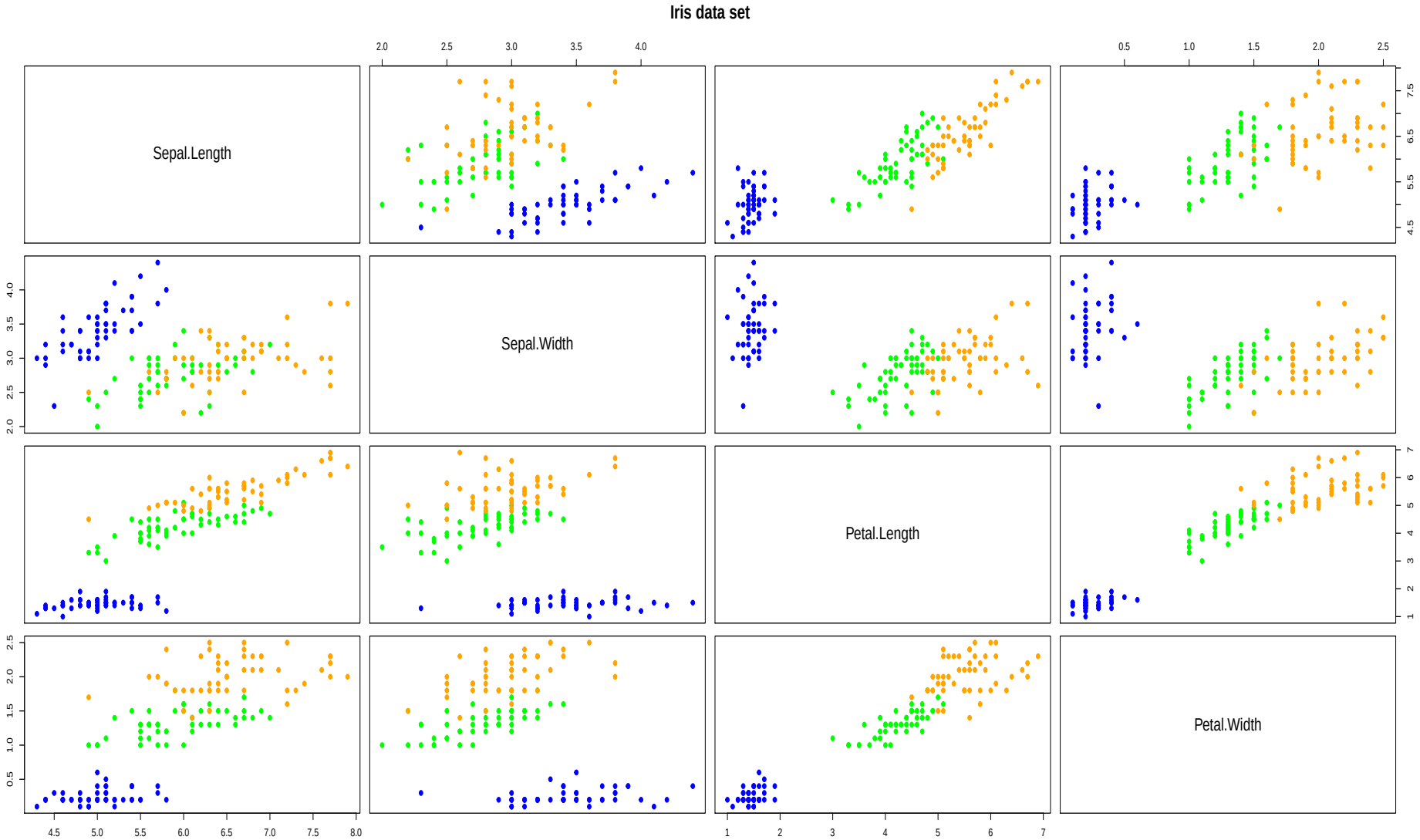
	Clas. in Setosa	Clas. in Versicolor	Clas. in Virginica
Setosa	50	0	0
Versicolor	0	48	2
Virginica	0	1	49

- Therefore, the misclassification error is again:

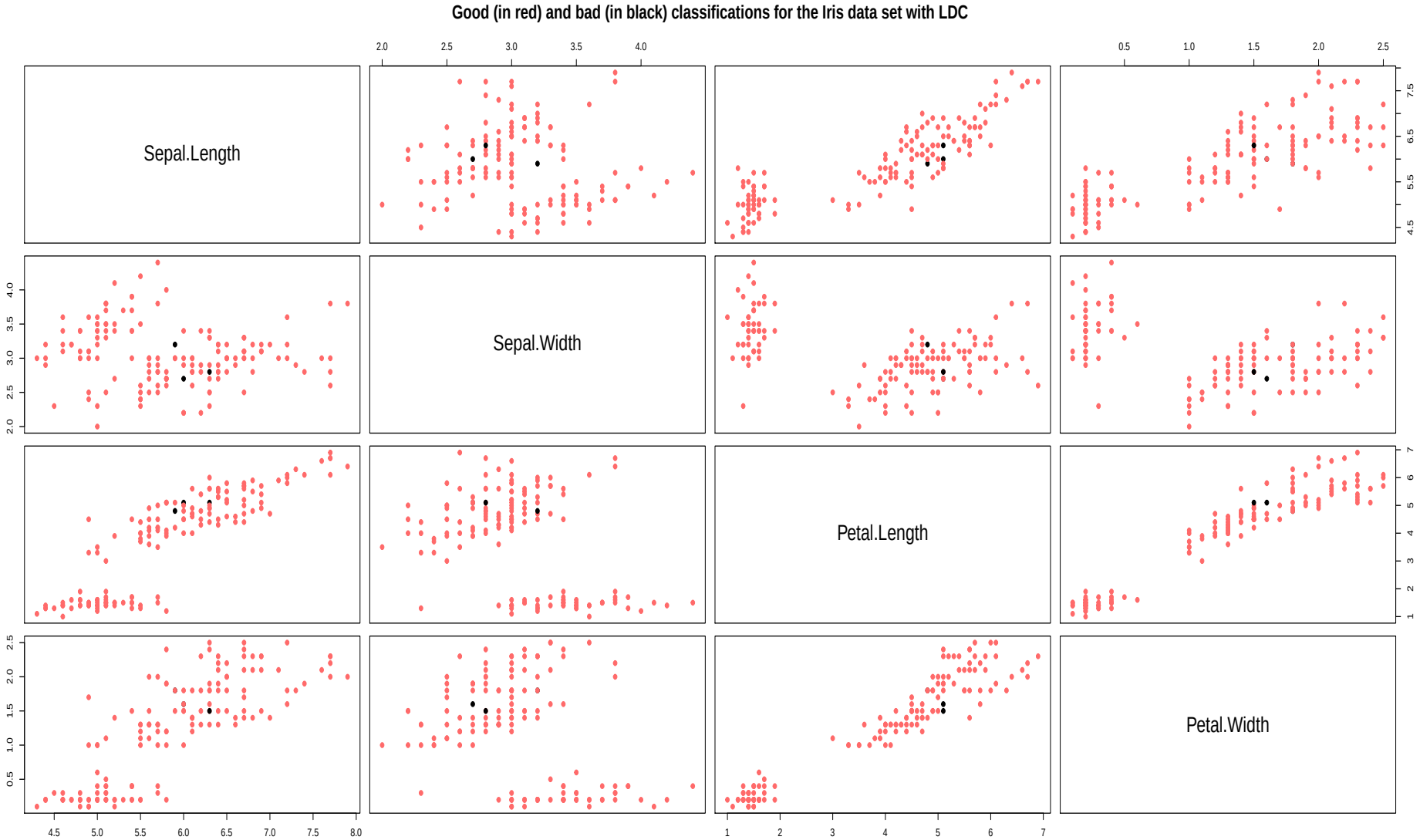
$$\text{Error} = \frac{3}{150} = 0.02$$

- The next slides show the scatterplot matrix of the classified elements, the scatterplot matrix of misclassified elements and the probabilities of the observations to belong to the three groups.

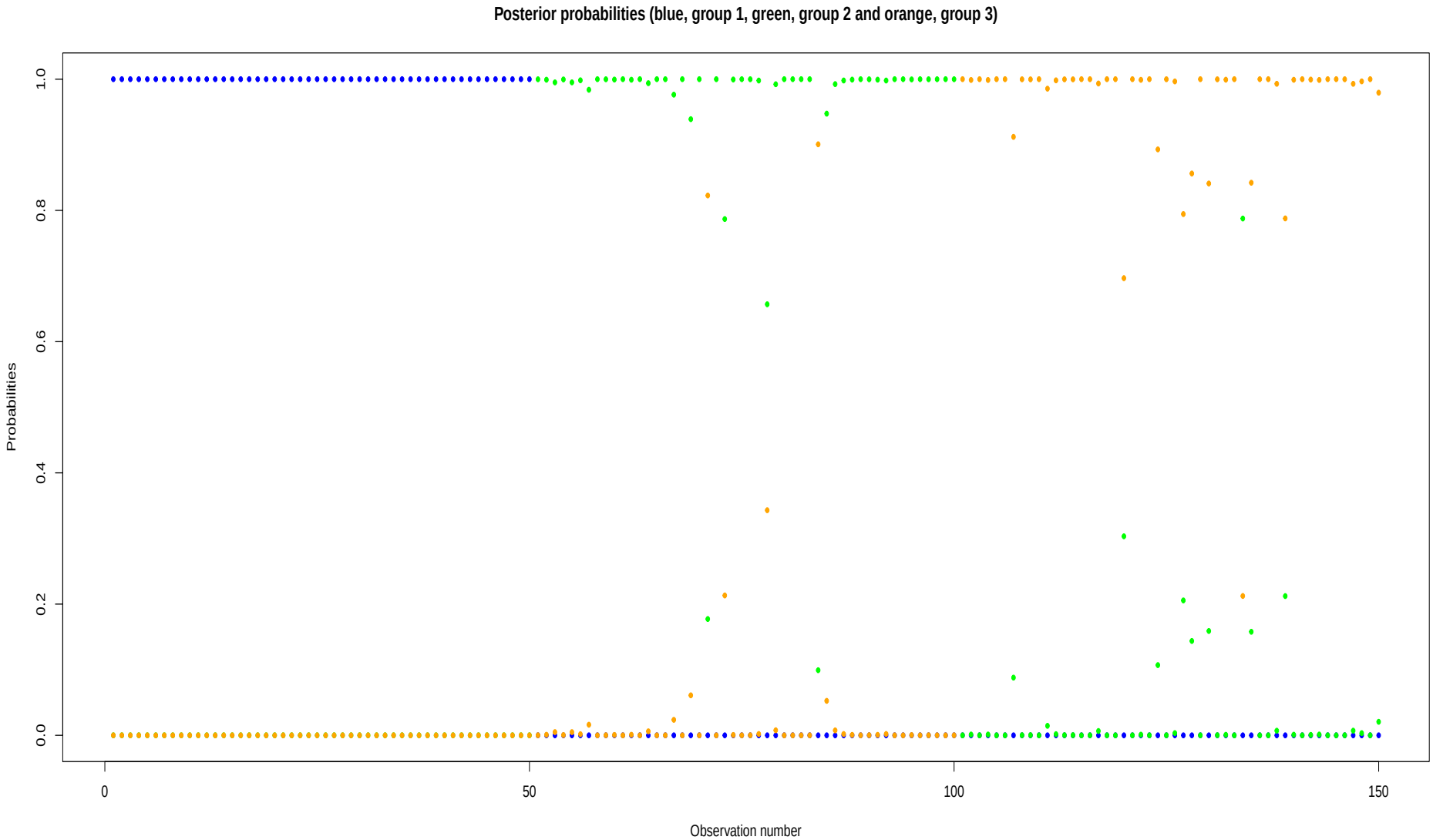
Illustrative example (I)



Illustrative example (I)



Illustrative example (I)



Bayes rule classifiers

- The second use of the Bayes rule classifier assumes that the density functions $f_1, \ldots, f_G$ are Gaussian with different mean vectors and different covariance matrices.
- Therefore, under P_g, x follows a $N(\mu_g, \Sigma_g)$ distribution.
- The optimal decision is, according to the Bayes rule, to classify the element in the population P_g that maximize $\pi_g f_g(x_0)$, which is given by:

$$\pi_g f_g(x_0) = \pi_g (2\pi)^{-p/2} |\Sigma_g|^{-1/2} \exp\left(-\frac{(x_0 - \mu_g)' \Sigma_g^{-1} (x_0 - \mu_g)}{2}\right)$$

Bayes rule classifiers

- Taking logarithms and deleting some nuisance constants, this is equivalent to classify the new element in the population P_g that maximize:

$$2\log\pi_g - \log|\Sigma_g| - (x_0 - \mu_g)'\,\Sigma_g^{-1}\,(x_0 - \mu_g)$$

- Note that the last expression is equal to:

$$2\log\pi_g - \log|\Sigma_g| - x_0\Sigma_g^{-1}x_0 + 2\mu_g'\Sigma_g^{-1}x_0 - \mu_g'\Sigma_g^{-1}\mu_g$$

- Consequently, the Bayes rule reduces to classify the new element in the population P_g that maximize:

$$p_g(x_0) = -x_0\Sigma_g^{-1}x_0 + 2\mu_g'\Sigma_g^{-1}x_0 - \mu_g'\Sigma_g^{-1}\mu_g + 2\log\pi_g - \log|\Sigma_g|$$

- As $p_g(x_0)$ depends quadratically on x_0, the method is called the quadratic discriminant classifier.

Bayes rule classifiers

- The quadratic discriminant classifier leads to the following expression of the probabilities $\Pr(y = g|x = x_0)$:

$$\Pr(y = g|x = x_0) = \frac{\pi_g |\Sigma_g|^{-1/2} \exp\left(-\frac{(x_0-\mu_g)'\Sigma_g^{-1}(x_0-\mu_g)}{2}\right)}{\sum_{k=1}^{G} \pi_k |\Sigma_k|^{-1/2} \exp\left(-\frac{(x_0-\mu_k)'\Sigma_k^{-1}(x_0-\mu_k)}{2}\right)} =$$

$$= \frac{\pi_g |\Sigma_g|^{-1/2} \exp\left(-\frac{1}{2} D_M^2(x_0, \mu_g)\right)}{\sum_{k=1}^{G} \pi_k |\Sigma_k|^{-1/2} \exp\left(-\frac{1}{2} D_M^2(x_0, \mu_k)\right)}$$

where $D_M^2(x_0, \mu_k) = (x_0 - \mu_k)' \Sigma_k^{-1} (x_0 - \mu_k)$ is the squared Mahalanobis distance between x_0 and μ_k, under population P_k.

Bayes rule classifiers

- As for the linear discrimination classifier, we are assuming that we know the probabilities $\pi_1, \ldots, \pi_G$ and the parameters of the Gaussian distributions, i.e. the means $\mu_1, \ldots, \mu_G$ and the covariance matrices $\Sigma_1, \ldots, \Sigma_G$.
- The prior probabilities can be estimated as in the previous case, i.e., $\widehat{\pi}_g = n_g/n$.
- The mean vector under P_g, i.e., μ_g, can be estimated with the sample mean vector within population P_g, i.e.

$$\overline{x}_g = \frac{1}{n_g} \sum_{i=1}^{n_g} x_{i \cdot g}$$

- The covariance matrix under P_g, i.e., Σ_g, can be estimated with the sample covariance matrix within population P_g:

$$S_g = \frac{1}{n_g - 1} \sum_{i=1}^{n_g} \left(x_{i \cdot g} - \overline{x}_g\right) \left(x_{i \cdot g} - \overline{x}_g\right)'$$

Bayes rule classifiers

- Now, we can apply the quadratic discriminant rule and to estimate the probabilities $\Pr(y = g|x = x_0)$ after replacing the parameters of the Gaussian populations with their sample counterparts.
- However, note that in the linear case we have to estimate $Gp + p(p+1)/2$ parameters, while in the quadratic case, we have to estimate $G(p + p(p+1)/2)$ parameters.
- Therefore, except for very large samples, the quadratic discriminant rule is relatively unstable and, although the covariance matrices are very different, we frequently obtain better results using the linear rule than the quadratic one.
- Also, the quadratic discriminant classifier is more sensitive to deviations from Gaussianity in the data than the linear classifier, so that it is recommended always to apply both rules and to check which of them have a better performance.
- Finally, we use leave-one-out cross validation to estimate the performance of the quadratic discriminant rule for a given data set.

Illustrative example (I)

- We use the quadratic discriminant rule to classify the flowers in the Iris data set using leave-one-out cross-validation.
- Here, we find four misclassifications:

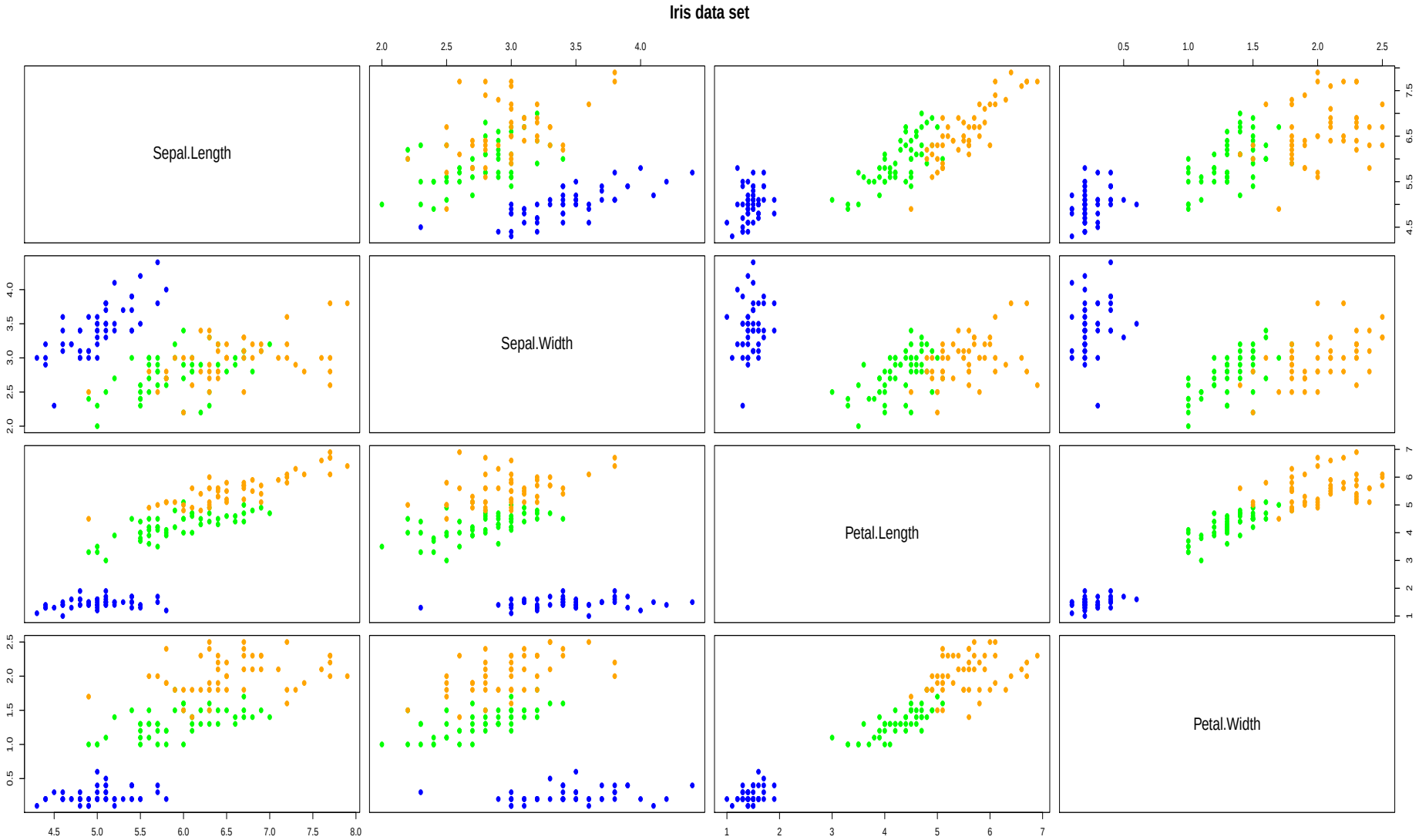
	Clas. in Setosa	Clas. in Versicolor	Clas. in Virginica
Setosa	50	0	0
Versicolor	0	47	3
Virginica	0	1	49

- Therefore, the misclassification error is again:

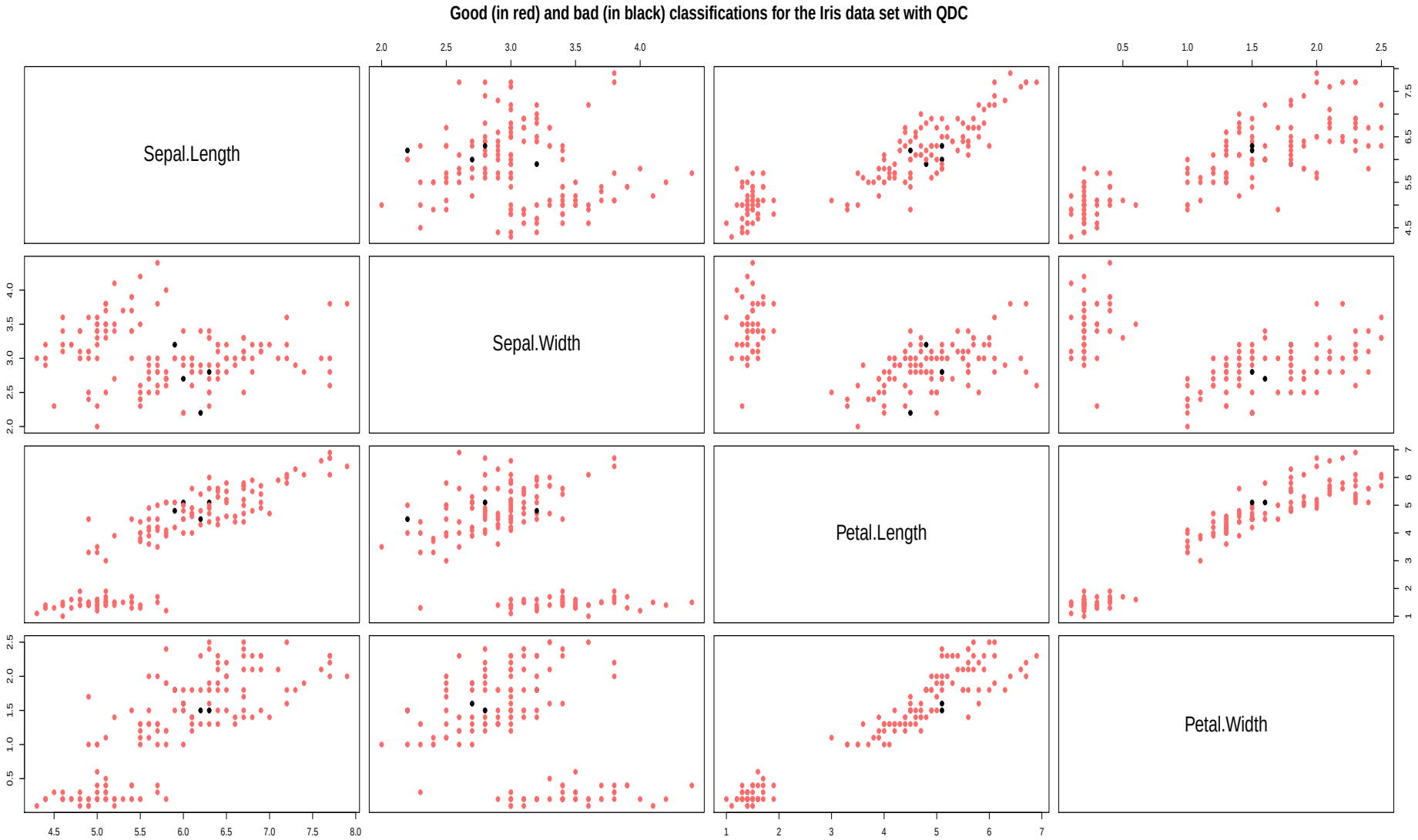
$$\text{Error} = \frac{4}{150} = 0.0266$$

- The next slides show the scatterplot matrix of the classified elements, the scatterplot matrix of misclassified elements and the probabilities of the observations to belong to the three groups.

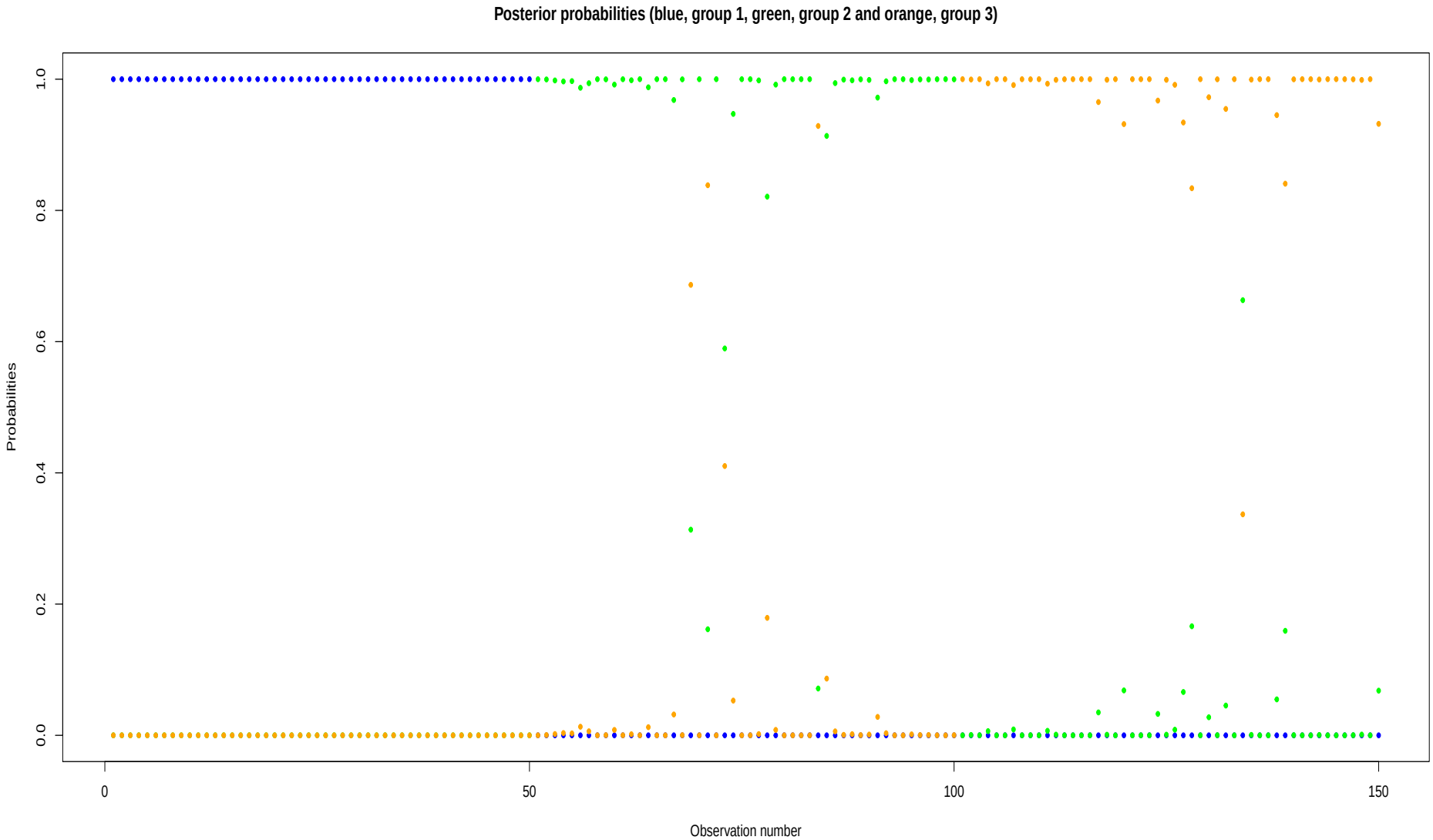
Illustrative example (I)



Illustrative example (I)



Illustrative example (I)



Logistic regression

- One of the main problems of Bayes rule classifiers is that it requires to assume that the random variable x has a certain distribution function under every population P_g.
- This restrict the type of variables that can be used with the linear and quadratic discriminant rules.
- Nevertheless, the probabilistic argument provides with strong support to the decisions taken by the rules.
- The question is whether is possible to compute the probabilities $\Pr(y = g|x = x_0)$, for $g = 1, \ldots, G$, without explicit knowledge of the densities.
- Logistic regression is a method to undertake such goal.

Logistic regression

- To avoid the use of the Bayes Theorem, one possibility is to assume that:

$$\Pr(y = g|x = x_0) = h_g(x_0)$$

  where $h_g(x_0)$ are certain positive functions of x_0 such that $h_1(x_0) + \cdots + h_G(x_0) = 1$.

- The question is which functions $h_1(x_0), \ldots, h_G(x_0)$ are the most appropriate to provide good classifications?

Logistic regression

- In logistic regression, these functions (probabilities) are defined as:

$$h_g(x_0) = \Pr(y = g | x = x_0) = \frac{\exp\left(\beta_{g0} + \beta'_{g1} x_0\right)}{1 + \sum_{k=1}^{G-1} \exp\left(\beta_{k0} + \beta'_{k1} x_0\right)}$$

for $g = 1, \ldots, G-1$, while for $g = G$:

$$h_G(x_0) = \Pr(y = G | x = x_0) = \frac{1}{1 + \sum_{k=1}^{G-1} \exp\left(\beta_{k0} + \beta'_{k1} x_0\right)}$$

where β_{g0}, for $g = 1, \ldots, G-1$ are real parameters and β_{g1} are p-dimensional vector parameters.

- Note that,

$$\sum_{g=1}^{G} h_g(x_0) = \sum_{g=1}^{G} \Pr(y = g | x = x_0) = 1$$

Logistic regression

- To understand the explicit form of the probabilities in logistic regression, consider the case of $G = 2$.
- In this case, we have:

$$\Pr(y = 1|x = x_0) = \frac{\exp(\beta_{10} + \beta_{11}' x_0)}{1 + \exp(\beta_{10} + \beta_{11}' x_0)}$$

and:

$$\Pr(y = 2|x = x_0) = \frac{1}{1 + \exp(\beta_{10} + \beta_{11}' x_0)}$$

respectively.

Logistic regression

- Then:

$$\frac{\Pr(y=1|x=x_0)}{\Pr(y=2|x=x_0)} = \exp(\beta_{10} + \beta_{11}'x_0)$$

- In other words,

$$\log\left(\frac{\Pr(y=1|x=x_0)}{1-\Pr(y=1|x=x_0)}\right) = \beta_{10} + \beta_{11}'x_0$$

i.e., the logit of $\Pr(y=1|x=x_0)$ is a linear function of x_0.

Logistic regression

- In practice, the parameters of the logistic regression method should be estimated.
- This can be achieved using maximum likelihood estimation (MLE).
- For that, the likelihood function is given by:

$$L\left(\beta_{10}, \ldots, \beta_{G0}, \beta_{11}, \ldots, \beta_{G1} | X\right) = \prod_{i=1}^{n} \Pr\left(y = g_i | x = x_i\right)$$

  where g_i is the population number corresponding to the observation x_i..

- The log-likelihood function is:

$$\ell\left(\beta_{10}, \ldots, \beta_{G0}, \beta_{11}, \ldots, \beta_{G1} | X\right) = \sum_{i=1}^{n} \log \Pr\left(y = g_i | x = x_i\right)$$

- The MLE of the parameters are obtained after maximizing the log-likelihood function.

Logistic regression

- Nevertheless, the MLE method has a drawback in terms of estimation.
- If some variables have more discriminant power that others, the parameter estimates might be largely biased.
- Several possibilities to solve this problem are the following:
  1. Skip variables without discriminant power from the analysis using graphical techniques.
  2. Use a variable selection procedure to select the variables most significant in the estimation procedure.
  3. Use a penalization method of the likelihood function, such as LASSO, to estimate the parameters of the model, and to shrink parameters associated with unimportant variables to 0.

Logistic regression

- In order to assess estimation error of the procedure we can also perform leave-one-out cross-validation as in the case of k-NN, the linear and the quadratic discriminant classifiers.
- Logistic regression can be applied to situations in which the observed variables are non-Gaussian, including discrete variables and categorical variables, that can be included in the model via dummy variables, as in multiple regression.
- However, under Gaussian populations, the linear and/or the quadratic discriminant classifiers are expected to have a better behavior.

Illustrative example (I)

- We use logistic regression to classify the flowers in the Iris data set using leave-one-out cross-validation.
- As with k-NN and the linear discriminant rule, there are only three misclassifications.

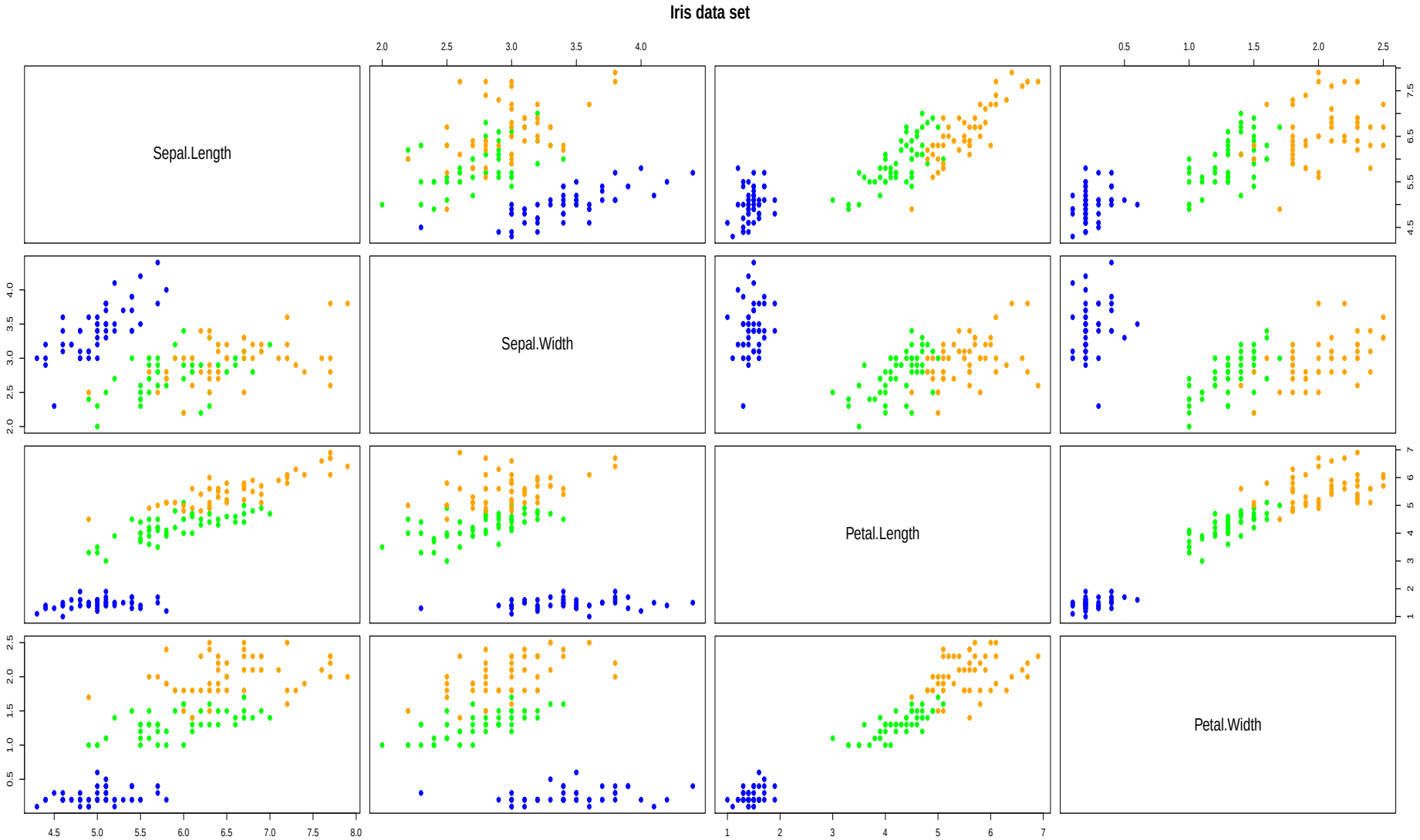
	Clas. in Setosa	Clas. in Versicolor	Clas. in Virginica
Setosa	50	0	0
Versicolor	0	48	2
Virginica	0	1	49

- Therefore, the misclassification error is again:

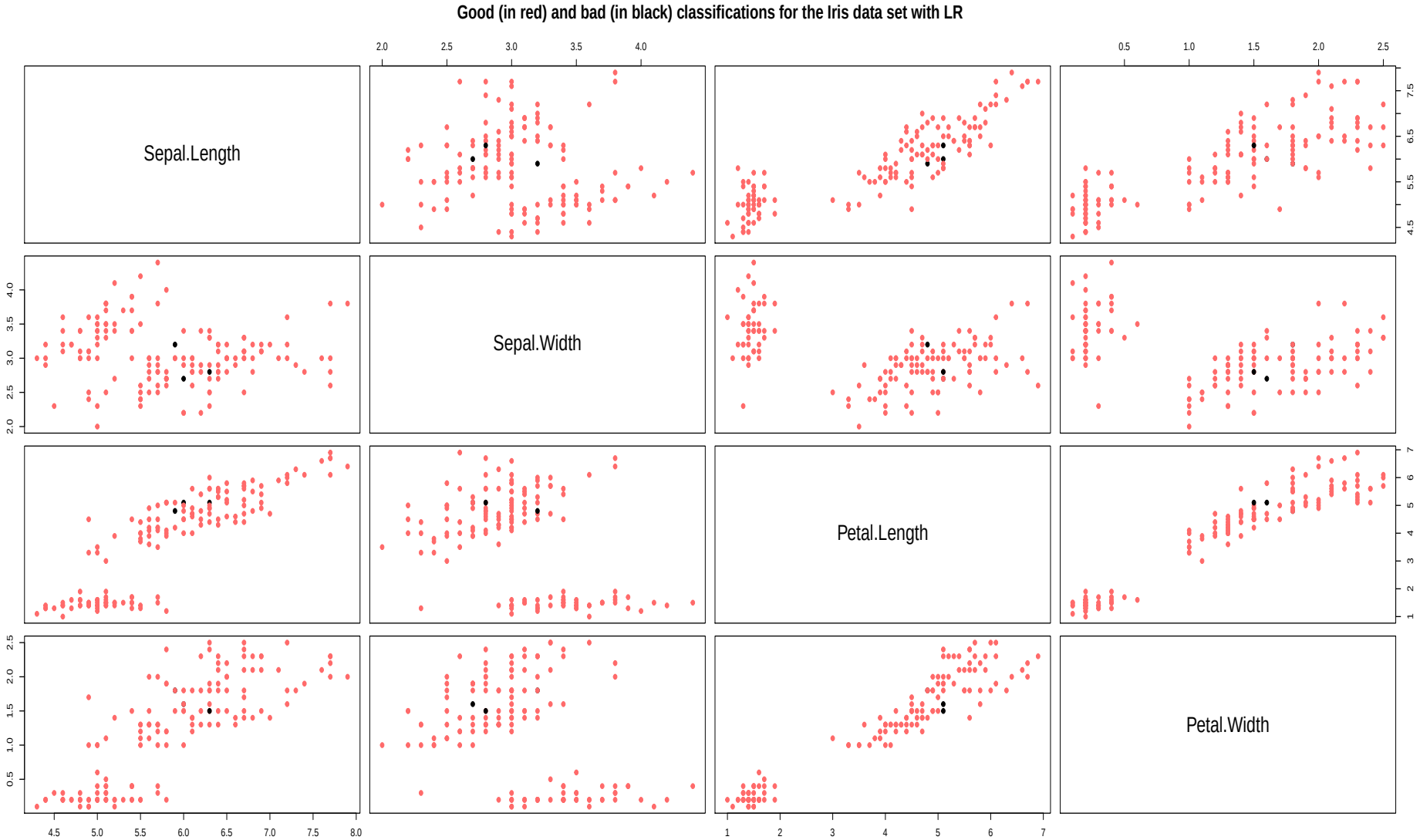
$$\text{Error} = \frac{3}{150} = 0.02$$

- The next slides show the scatterplot matrix of the classified elements, the scatterplot matrix of misclassified elements and the probabilities of the observations to belong to the three groups.

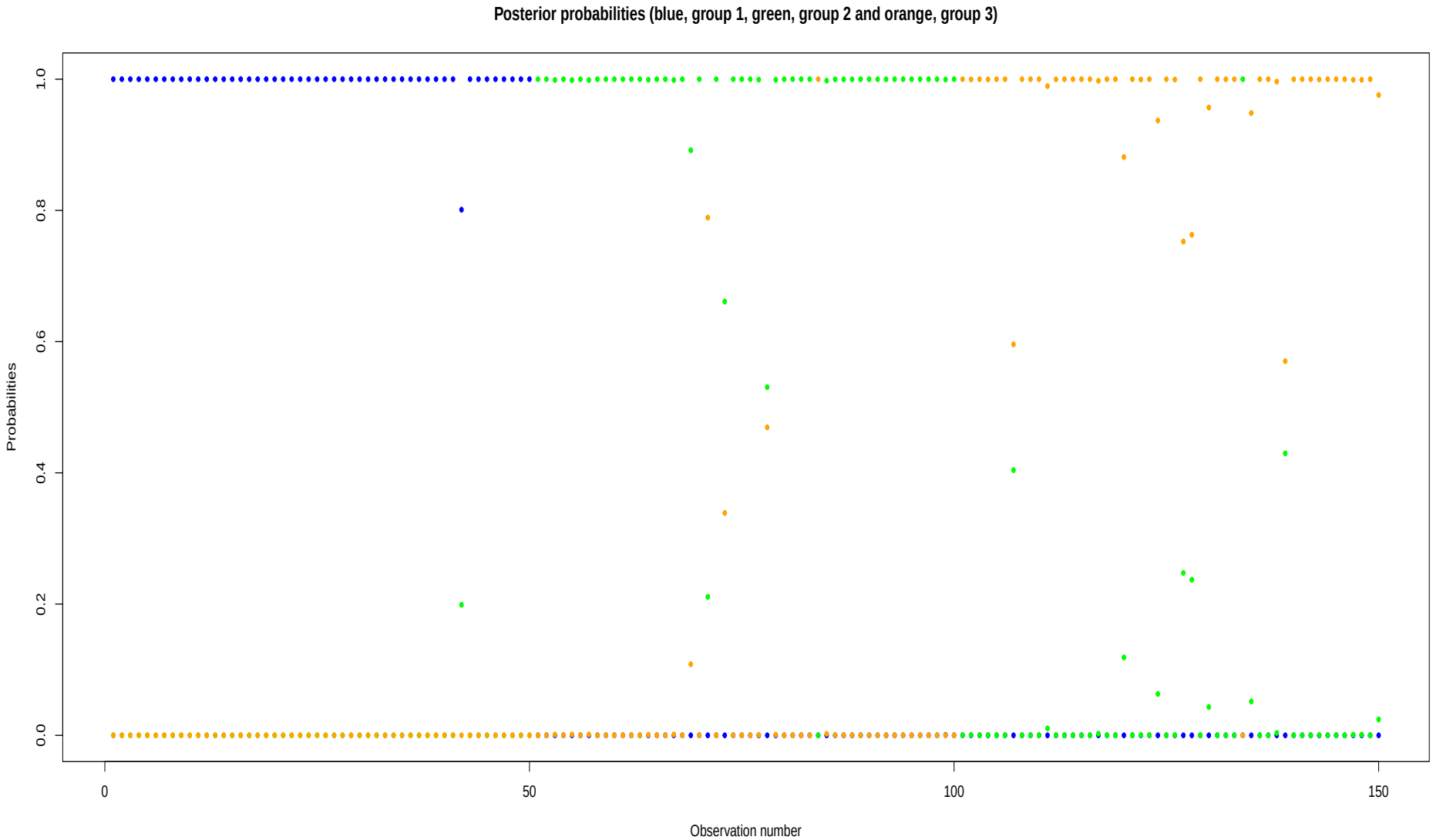
Illustrative example (I)



Illustrative example (I)



Illustrative example (I)



Alternative methods

- There are a large number of alternatives to the previous methods, some of them more computing-intensive methods (popular machine learning), including:
  - Generalized additive models.
  - Trees, random forests and boosting.
  - Support vector machines.

Chapter outline

We are ready now for:

The project and the exam