Multivariate Analytics
DATA 514 – Unit 3
Darius M. Dziuda, Ph.D.
CCSU, Spring 2025

Lecture 3:
Discriminant Analysis

1 Introduction

Discriminant Analysis is a classical supervised learning algorithm for classification. It is a powerful method that should be included in the toolbox of any data science professional. In addition to its classification capabilities, it also provides low-dimensional visualization of the classification space. The main assumptions under which discriminant analysis is performed include: the independence of training observations, no singularities or severe multicollinearities, and no extreme outliers. *Usually*, we also assume multivariate normality of variables; however, this assumption is —at least theoretically— not necessary. Fisher derived the original version of discriminant analysis without making this assumption (Fisher 1936). The goal of his design was to find such a solution to a classification problem that maximally separates the differentiated classes by maximizing the ratio of the variation between classes to the variation within classes. In contrast, the goal of the Bayesian design of discriminant analysis is to minimize the probability of misclassification; this design assumes that the variables follow a multivariate normal distribution in each of the differentiated classes.

There is a lot of confusion surrounding discriminant analysis, not only about whether the assumption of multivariate normality is necessary or not, but also about terminology and the relation between the two designs. To make it more transparent, we will refer to the original design as *Fisher's discriminant analysis* (FDA), and to the design using the Bayes approach, which assumes multivariate normality (Welch 1939), as *Gaussian* discriminant analysis. As already stated, the former maximizes the ratio of the variation between classes to that within classes, and the latter minimizes the probability of misclassification under the assumption of multivariate normality. Furthermore, Fisher's solution provides only linear boundaries between classes, while the Gaussian approach may identify either linear or quadratic boundaries. If the homogeneity (equality) of class covariance matrices is assumed, we have Gaussian *linear discriminant analysis* (LDA), if this assumption is not made, we have Gaussian *quadratic discriminant analysis* (QDA).

It is also worth stressing that both Fisher's approach and the Gaussian approach work well for multiclass classification problems, and that their good performance is attributed to the fact that simple —linear or quadratic— boundaries between classes are more likely to be supported by the available data than intricate alternatives; this is especially true for the linear decision boundaries. Furthermore, Fisher's discriminant analysis and Gaussian LDA lead to the same solution when the misclassification costs and the prior probabilities are the same for each class (Welch 1939, Hastie et al. 2009). Hence, both minimize the probability of misclassification when the differentiated populations are normally distributed (assumed or not).

Assuming that our classification problem involves J classes and p independent variables, calculations for discriminant analysis would require the inversion of a $p \times p$ matrix. This cannot be done when the number of variables p is greater than the number of observations N (for QDA, it would be the number of observations per class $N_j, j = 1,...,J$).[1] That will, however, in no way decrease the usability of this classification method, as it should be obvious that when dealing with high-dimensional $p > N$ data (and especially with $p >> N$ data), the feature selection step should be performed first[2], before building the final classifier. Although any supervised and multivariate feature selection method can be used for feature selection before performing discriminant analysis, it may be advantageous —in the context of classification with discriminant analysis— to use such a feature selection algorithm that will select an optimal subset of variables using an aligned criterion of maximizing the variation between classes in relation to the variation within classes. Furthermore, discriminant analysis can be used as a learning algorithm within a feature selection schema implementing either a forward or hybrid stepwise search.

2 Fisher's Discriminant Analysis

Fisher's discriminant analysis (FDA) identifies a set of discriminant functions (which are linear combinations of the independent variables) that will maximize the ratio of the variation between J classes, $J \geq 2$, to the variation within the classes (Fisher 1936, 1938). This means searching for such a projection that would maximally separate the class centers while simultaneously minimizing class variations. **No assumption about the distribution of variables is made** (so, in particular, there is no assumption that variables in each class have multivariate normal distribution). However, FDA makes an implicit assumption of the homogeneity of class covariance matrices.

[1] Thus, LDA would be a better choice than QDA when we have a relatively small number of training observations (provided LDA's assumption of the homogeneity of class covariances is not severely violated).

[2] Some software implementations of LDA include feature selection as the first step of the data analysis, before LDA is invoked.

Assume that in our training data we have p variables, J classes, and N observations. Each observation is represented by a $p \times 1$ vector $\mathbf{x}_i = \left[x_{1i}, \cdots, x_{pi} \right]^T$ and a class label y_i, $i = 1, ..., N$. Assume also that the class labels are $\mathcal{D}_1$, $\mathcal{D}_2$, … , $\mathcal{D}_J$, and that the number of training observations in class j is $N_j, j = 1, ..., J$, where $\sum N_j = N$. The class centers (mean vectors for observations from each class) are

$$\overline{\mathbf{x}}_j = \frac{1}{N_j} \sum_{i: y_i = \mathcal{D}_j} \mathbf{x}_i, \qquad j = 1, ..., J, \tag{1}$$

and the overall mean is

$$\overline{\overline{\mathbf{x}}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i. \tag{2}$$

We will first consider the simplest case of two-class Fisher's discriminant analysis (when $J = 2$), and then the general case of multiclass FDA (when $J \geq 2$).

2.1 Two-class Fisher's Discriminant Analysis

When we only have two classes ($J = 2$), the class centers will always lie on a line. Thus, to maximally separate the classes, we will look for such a projection $\mathbf{u}$, for which a linear function $z = \mathbf{u}^T \mathbf{x}$ will project training observations from their original p-dimensional space onto such a line (one-dimensional space) that would maximize the distance between the projected class means and simultaneously minimize the variation of each class. Since the difference between the projected class means, $\overline{z}_1 = \mathbf{u}^T \overline{\mathbf{x}}_1$ and $\overline{z}_2 = \mathbf{u}^T \overline{\mathbf{x}}_2$, may be negative, we will use the squared distance,

$$\begin{aligned} (\overline{z}_1 - \overline{z}_2)^2 &= (\mathbf{u}^T \overline{\mathbf{x}}_1 - \mathbf{u}^T \overline{\mathbf{x}}_2)^2 \\ &= \mathbf{u}^T (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^T \mathbf{u}. \end{aligned} \tag{3}$$

Observing that the variation *between* the two classes is represented by a $p \times p$ between-class scatter matrix

$$\mathbf{B} = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^T, \tag{4}$$

the distance between the projected class means can be presented as $\mathbf{u}^T \mathbf{B} \mathbf{u}$. Since the variation of class j can be represented by a $p \times p$ scatter matrix $\mathbf{W}_j$,[3]

[3] Recall that variance is calculated as variation divided by its degrees of freedom, and observe that the matrix of variation of class j is ($N_j - 1$) times the covariance matrix of the class, that is, $\mathbf{W}_j = (N_j - 1)\, \mathbf{S}_j$ (Duda et al. 2001).

$$\mathbf{W}_j = \sum_{i:y_i=\mathcal{D}_j} (\mathbf{x}_i - \overline{\mathbf{x}}_j)(\mathbf{x}_i - \overline{\mathbf{x}}_j)^T, \quad j = 1, 2, \tag{5}$$

the variations of the projected classes are $\mathbf{u}^T\mathbf{W}_1\mathbf{u}$ and $\mathbf{u}^T\mathbf{W}_2\mathbf{u}$; hence, to minimize them, we may minimize

$$\mathbf{u}^T\mathbf{W}_1\mathbf{u} + \mathbf{u}^T\mathbf{W}_2\mathbf{u} = \mathbf{u}^T(\mathbf{W}_1 + \mathbf{W}_2)\mathbf{u}. \tag{6}$$

Observing that the variation *within* the two classes can be represented by a $p \times p$ within-class scatter matrix **W**,

$$\begin{aligned} \mathbf{W} &= \mathbf{W}_1 + \mathbf{W}_2 \\ &= \sum_{j=1}^{2} \sum_{i:y_i=\mathcal{D}_j} (\mathbf{x}_i - \overline{\mathbf{x}}_j)(\mathbf{x}_i - \overline{\mathbf{x}}_j)^T, \end{aligned} \tag{7}$$

the projected within-class variation will be $\mathbf{u}^T\mathbf{W}\mathbf{u}$. Consequently, to maximize (3) and simultaneously minimize (6), we may maximize their ratio,

$$\underset{\mathbf{u}}{\text{maximize}} \ \frac{\mathbf{u}^T\mathbf{B}\mathbf{u}}{\mathbf{u}^T\mathbf{W}\mathbf{u}}. \tag{8}$$

Since **u** is a vector and we are only interested in its *direction* (which defines the direction of the discriminant function *z*), and not in its magnitude, we may replace (8) with the equivalent optimization problem,

$$\begin{aligned} &\underset{\mathbf{u}}{\text{maximize}} \ \ \mathbf{u}^T\mathbf{B}\mathbf{u} \\ &\text{subject to} \ \ \mathbf{u}^T\mathbf{W}\mathbf{u} = 1. \end{aligned} \tag{9}$$

To solve such a constrained optimization problem, we may introduce a nonnegative Lagrange multiplier λ, represent the problem by the Lagrangian

$$L(\mathbf{u}, \lambda) = \mathbf{u}^T\mathbf{B}\mathbf{u} \ - \lambda(\mathbf{u}^T\mathbf{W}\mathbf{u} - 1), \tag{10}$$

and solve it by setting its partial derivative to zero,

$$\frac{\delta L}{\delta \mathbf{u}} = 2\mathbf{B}\mathbf{u} - 2\lambda\mathbf{W}\mathbf{u} = 0, \tag{11}$$

which gives us

$$\mathbf{B}\mathbf{u} = \lambda\mathbf{W}\mathbf{u}. \tag{12}$$

This is a generalized eigenproblem, which for the case of binary classification has one eigenvalue λ and one eigenvector associated with this eigenvalue. We may check this by rewriting (12) as a regular one-matrix eigenproblem (which can be done under the assumption that **W** is not singular),

$$\mathbf{W}^{-1}\mathbf{B}\mathbf{u} = \lambda\mathbf{u}. \tag{13}$$

Although **B** and **W** are $p \times p$ matrices, the rank of **B** is only one (as it is the outer product of two vectors), thus matrix $\mathbf{W}^{-1}\mathbf{B}$ is also of rank one, which means that the solution to (13), as well as to (12), is the single eigenvector of matrix $\mathbf{W}^{-1}\mathbf{B}$. This $p \times 1$ vector $\mathbf{u} = \left[u_1, \cdots, u_p\right]^T$ defines the direction of the *linear discriminant function* line *z*. This discriminant function is a linear combination of *p* independent variables, $z = u_1 x_1 + \ldots + u_p x_p$, and it projects any observation, say $\mathbf{x}_{new}$, from the original *p*-dimensional space onto the one-dimensional discriminatory space (see Figure 1).

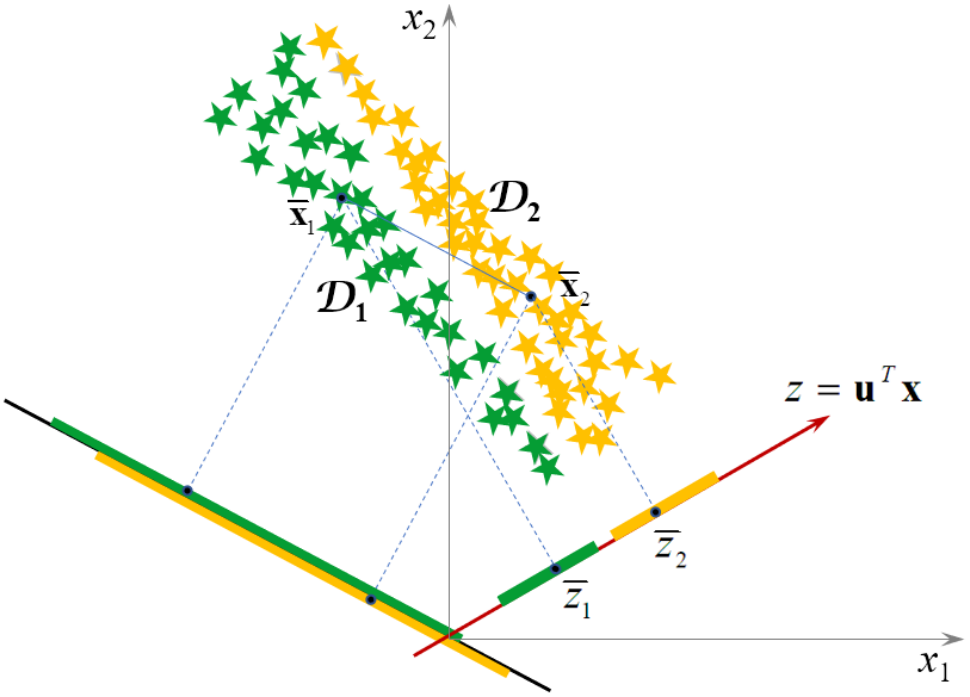

Figure 1: *Visualization of two-class Fisher's discriminant analysis (J = 2) for a toy example with only two independent variables (p = 2). With two classes, there is only one linear discriminant function,* $z = \mathbf{u}^T\mathbf{x}$, *whose direction is defined by vector* ***u*** *(red line). This direction maximizes class separation – in this example the training observations belonging to classes* $\mathcal{D}_1$ *and* $\mathcal{D}_2$ *are completely separated after they are projected on line z. Recall that maximal class separation means maximizing the ratio of the between-class to within-class variation, which corresponds to minimal overlap between classes, rather than the maximal distance between class centers (the black line represents direction that maximizes the distance between class centers; projecting training observations on this line would result in a very heavy overlap between classes).*

The only additional thing needed for classification is to decide about the threshold, that is, the point on line *z* (between the projected class centers $\bar{z}_1 = \mathbf{u}^T\bar{\mathbf{x}}_1$ and $\bar{z}_2 = \mathbf{u}^T\bar{\mathbf{x}}_2$), which

would define a boundary between the classes. If the projection of the observation, $z_{new} = \mathbf{u}^T \mathbf{x}_{new}$, is less than the threshold, the observation would be classified to one class, otherwise to another. If the midpoint between class centers, $(\overline{z}_1 + \overline{z}_2)/2$, is selected as the threshold, the observation will be classified to the class whose projected center is closer to z_{new} (Fisher 1936). Another option is to assume that the *projected* classes are normally distributed and then use this assumption to calculate an optimal threshold (that would minimize the probability of misclassification). Recall that Fisher's linear discriminant function has been identified without making the assumption of multivariate normality. However, since this discriminant function, $z = \mathbf{u}^T \mathbf{x}$, is the sum of many random variables, the central limit theorem provides some justification for making the assumption that the *projected* data is approximately normal in each class (Bishop 2006).

The main goal of this section is to describe FDA in a simple two-class situation, as an introduction to the multiclass case. However, if we were interested only in the two-class case, then we would not really need to solve eigenproblem (13). It has been shown (Duda et al. 2001, Rencher 2002) that, for a two-class case, we have

$$z = \mathbf{u}^T \mathbf{x} = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^T \mathbf{W}^{-1} \mathbf{x}, \tag{14}$$

and, consequently, classification could be performed the same way as described earlier, but z_{new}, $\overline{z}_1$, and $\overline{z}_2$ would be calculated using the rightmost part of Formula (14).

2.2 Multiclass Fisher's Discriminant Analysis

The centers of two classes lie on a line, the centers of three classes on a two-dimensional plane, and the centers of J classes will always lie on a $(J - 1)$-dimensional hyperplane. Hence, in a multiclass case,[4] FDA will identify $(J - 1)$ linear discriminant functions that would project observations from their original p-dimensional space into such a $(J - 1)$-dimensional space in which the classes will be maximally separated, that is, the ratio of the between-class to within-class variations will be maximized. Let us first consider these variations for a multiclass case. The *variation within classes* is based on the distances between each observation in the class and its class center, and is represented by a $p \times p$ scatter matrix $\mathbf{W}$,

$$\mathbf{W} = \sum_{j=1}^{J} \sum_{i:y_i = \mathcal{D}_j} (\mathbf{x}_i - \overline{\mathbf{x}}_j)(\mathbf{x}_i - \overline{\mathbf{x}}_j)^T. \tag{15}$$

Observe that the matrix of the within-class variation $\mathbf{W} = (N - J)\mathbf{S}$, where $\mathbf{S}$ is the common covariance matrix (see Formula 38). This means that FDA is performed under the *implicit* assumption of homogeneity of class covariance matrices and thus results in linear boundaries between classes.

[4] The multiclass case may also be considered a general case with $J \geq 2$.

The *variation between classes* is based on the distances between class centers. However, instead of calculating all $\binom{J}{2}$ pairwise distances for J classes, we may exploit the fact that the total variation $\mathbf{T} = \sum_{i=1}^{N} (\mathbf{x}_i - \bar{\bar{\mathbf{x}}})(\mathbf{x}_i - \bar{\bar{\mathbf{x}}})^T = \mathbf{B} + \mathbf{W}$. Since both **T** and **W** are easy to calculate, we may compute **B** as **T** – **W**; as a result, the variation between classes may be calculated in a much easier way by only using the distances between each class center and the overall mean, and represented by the $p \times p$ scatter matrix **B**,

$$\mathbf{B} = \sum_{j=1}^{J} N_j (\bar{\mathbf{x}}_j - \bar{\bar{\mathbf{x}}})(\bar{\mathbf{x}}_j - \bar{\bar{\mathbf{x}}})^T. \tag{16}$$

These three variations and their degrees of freedom are summarized in Table 1.

Table 1. *Partitioning total variation into variation between classes and variation within classes,* **T** = **B** + **W**. *Recall that variance is variation divided by its corresponding degrees of freedom. Thus, for example, the common within-class variance-covariance matrix* **S** = **W** / *(N - J).*

Source of Variation	Variation (Sum of Squares)	Degrees of Freedom
Between-class	$\mathbf{B} = \sum_{j=1}^{J} N_j (\bar{\mathbf{x}}_j - \bar{\bar{\mathbf{x}}})(\bar{\mathbf{x}}_j - \bar{\bar{\mathbf{x}}})^T$	$J - 1$
Within-class	$\mathbf{W} = \sum_{j=1}^{J} \sum_{i:y_i=\mathcal{D}_j} (\mathbf{x}_i - \bar{\mathbf{x}}_j)(\mathbf{x}_i - \bar{\mathbf{x}}_j)^T$	$N - J$
Total	$\mathbf{T} = \sum_{i=1}^{N} (\mathbf{x}_i - \bar{\bar{\mathbf{x}}})(\mathbf{x}_i - \bar{\bar{\mathbf{x}}})^T$	$N - 1$

The projection from the original *p*-dimensional space into a (*J* - 1)-dimensional space is represented by $\mathbf{z} = \mathbf{U}^T \mathbf{x}$, where **U** is now a $p \times (J-1)$ matrix, which will be used to map *p*-dimensional observations **x** into their (*J* - 1)-dimensional representations **z**. Thus, we search for the projection matrix **U** that maximizes the ratio of the projected between-class variation to the projected within-class variation, which will now be represented as[5]

$$\underset{\mathbf{U}}{\text{maximize}} \ \frac{tr(\mathbf{U}^T \mathbf{B} \mathbf{U})}{tr(\mathbf{U}^T \mathbf{W} \mathbf{U})}, \tag{17}$$

[5] Observe that $\mathbf{U}^T\mathbf{B}\mathbf{U}$ is not a scalar now, as we have *J*-1 projections; summing up the between-class variations in all *J*-1 dimensions is equivalent to calculating the trace of $\mathbf{U}^T\mathbf{B}\mathbf{U}$. The same is true for the within-class variations.

where *tr*() denotes the trace of a matrix (that is, the sum of its diagonal elements). Following the same reasoning as before, we may rewrite (17) as a constrained optimization problem,

$$\begin{aligned} &\underset{\mathbf{U}}{\text{maximize}} \quad tr(\mathbf{U}^T\mathbf{B}\mathbf{U}) \\ &\text{subject to} \quad \text{tr}(\mathbf{U}^T\mathbf{W}\mathbf{U}) = 1\ , \end{aligned} \tag{18}$$

introduce a vector of nonnegative Lagrange multipliers $\boldsymbol{\lambda}$, represent the problem in the form of the Lagrangian

$$L(\mathbf{U},\boldsymbol{\lambda}) = tr(\mathbf{U}^T\mathbf{B}\mathbf{U})\ - \boldsymbol{\lambda}[tr(\mathbf{U}^T\mathbf{W}\mathbf{U}) - 1], \tag{19}$$

and, finally, solve it by setting its partial derivative $\delta L / \delta \mathbf{U}$ to zero. This will result in the generalized eigenproblem

$$\mathbf{B}\mathbf{U} = \boldsymbol{\lambda}\mathbf{W}\mathbf{U}. \tag{20}$$

We can solve it directly in this form, or, alternatively, if matrix $\mathbf{W}$ is nonsingular, in the form of one-matrix eigenproblem $\mathbf{W}^{-1}\mathbf{B}\mathbf{U} = \boldsymbol{\lambda}\mathbf{U}$. Since the between-class scatter matrix $\mathbf{B}$ is the sum of J rank-one matrices, its rank, and thus the rank of matrix $\mathbf{W}^{-1}\mathbf{B}$, is $J - 1$.[6] Thus, as a solution to (20) we will have a $(J-1)\times(J-1)$ diagonal matrix $\boldsymbol{\lambda}$ of eigenvalues and a $p\times(J-1)$ matrix $\mathbf{U}$, whose columns will represent normalized eigenvectors $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_{(J-1)}$ associated with the eigenvalues,

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{(J-1)} \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1(J-1)} \\ u_{21} & u_{22} & \cdots & u_{2(J-1)} \\ \vdots & \vdots & \ddots & \vdots \\ u_{p1} & u_{p2} & \cdots & u_{p(J-1)} \end{bmatrix}. \tag{21}$$

Therefore, the projection from the original *p*-dimensional data space to the (*J* - 1)-dimensional space of the maximum class separation will be defined by the (*J* - 1) *linear discriminant functions*,

$$\begin{aligned} z_1 &= \mathbf{u}_1^T\mathbf{x}, \\ z_2 &= \mathbf{u}_2^T\mathbf{x}, \\ &\cdots \\ z_{(J-1)} &= \mathbf{u}_{(J-1)}^T\mathbf{x}. \end{aligned} \tag{22}$$

Since the nonzero eigenvalues are ranked, $\lambda_1 > \lambda_2 > ... > \lambda_{J-1} > 0$, the single direction that best separates the classes is the one associated with the first discriminant function, z_1. Consequently, if we want to visualize the discriminatory space when $J > 4$, we would use the first two or three discriminant functions. The relative magnitude of class separation

[6] Theoretically, it is min(*p, J* – 1), though it is unlikely to have fewer variables than classes.

provided by each of the discriminant functions can be calculated as the proportion of its eigenvalue to the sum of all eigenvalues (Rencher 2002),

$$\frac{\lambda_k}{\sum_{l=1}^{J-1}\lambda_l},\ k=1,...,J-1. \tag{23}$$

To classify a new observation $\mathbf{x}_{new}$, we would project it into the (J - 1)-dimensional discriminatory space,

$$\mathbf{z}_{new}=\mathbf{U}^T\mathbf{x}_{new}, \tag{24}$$

and then assign it using some classification criterion; for example, we may assign the observation to class j, whose projected center, $\overline{\mathbf{z}}_j=\mathbf{U}^T\overline{\mathbf{x}}_j$, is closest to the projected observation. This common practice exploits the fact that in a (J - 1)-dimensional space of uncorrelated features represented by the discriminant functions, the features have unit variances and zero covariances, which means that the within-class Mahalanobis distance[7] is the same as Euclidean distance (Ripley 1996). This also means that the class areas are represented by hyperspheres (rather than hyperellipsoids in the original p-dimensional space). Consequently, the *classification function* for Fisher's discriminant analysis may be written (Johnson and Wichern 2007) as:

$$Class(\mathbf{x}_{new})=\arg\min_j \sum_{k=1}^{J-1}[\mathbf{u}_k^T(\mathbf{x}_{new}-\overline{\mathbf{x}}_j)]^2 \qquad j=1,...,J. \tag{25}$$

Although the FDA's discriminatory space defined by the (J - 1) linear discriminant functions has been identified without assuming multivariate normality of classes in the original p-dimensional space, we may now —after transforming the data into this low-dimensional discriminatory space— assume multivariate normality of the (J - 1) features in each of the *projected* classes (justifying this, just like for the binary case, by invoking the central limit theorem).

For two, three, or four differentiated classes ($J<5$), the discriminatory space, training data, and classification results can be graphically presented in one-, two-, or three-dimensional space (respectively) that includes 100% of the discriminatory information (see Figure 2). For $J\geq 5$, we can visualize a subspace of the discriminatory space using the first two or three discriminant functions.

[7] If the variances of the features were not equal (as it would be for the original p variables), then to properly measure the distances between the classified observation and the class centers, Mahalanobis distance would need to be used as it takes into account such different variances.

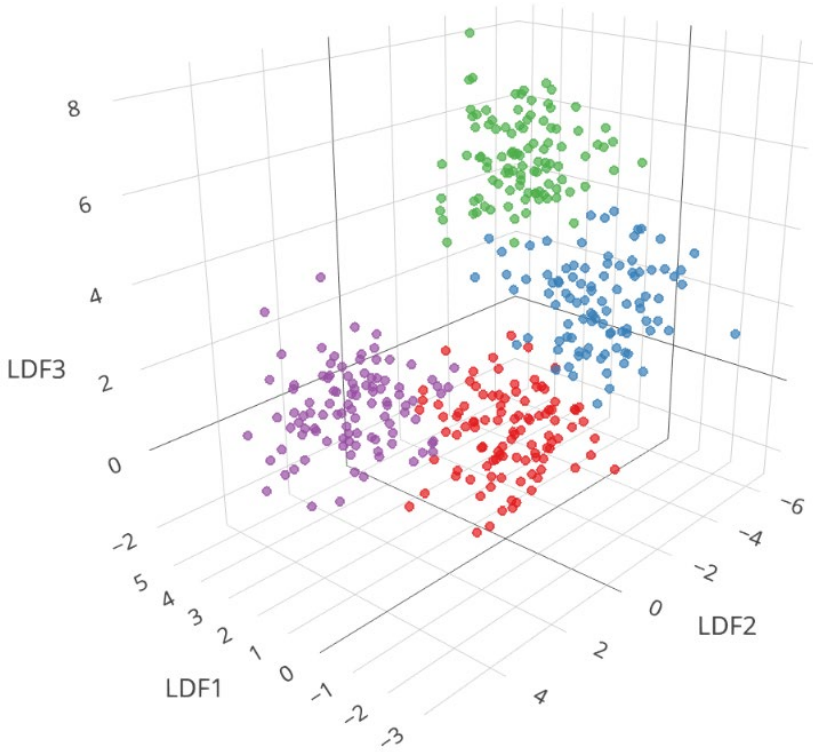

Figure 2: *An example of the discriminatory space for four classes (J=4). When four classes are differentiated, Fisher's discriminant analysis identifies three linear discriminant functions (LDF1, LDF2, LDF3); they define a three-dimensional discriminatory space, in which classes are maximally separated. This space includes 100 percent of the discriminatory information.*

3 Gaussian Discriminant Analysis

Fisher's discriminant analysis can be perceived as a nonparametric method employing the frequentist approach. In contrast, Gaussian discriminant analysis is based on a Bayesian approach, and is a parametric method since —for each class— it makes the assumption of multivariate normal distribution (that is, Gaussian densities) of the independent variables (Welch 1939). These lead to a solution for which the probability of misclassification is minimized.

Assume that our target population includes J classes, say, J disease states labeled $\mathcal{D}_1$, $\mathcal{D}_2$, … , $\mathcal{D}_J$, and that any observation, say, a patient under diagnosis, is represented by a $p \times 1$ vector $\mathbf{x}_i = \left[x_{1i}, \cdots, x_{pi} \right]^T$ corresponding to measured values of p variables (for example, gene expression levels). The core notion for a Bayesian classifier is the *posterior* probability $P(\mathcal{D}_j \mid \mathbf{x})$ of the observation belonging to class j (class with label $\mathcal{D}_j$) given that the vector representing the observation's results is $\mathbf{x}$. This posterior probability is calculated, for each class j, $j = 1, ..., J$, using Bayes' rule,

$$P(\mathcal{D}_j \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid \mathcal{D}_j)P(\mathcal{D}_j)}{\sum_{m=1}^{J} P(\mathbf{x} \mid \mathcal{D}_m)P(\mathcal{D}_m)}, \tag{26}$$

where:

- $P(\mathcal{D}_j)$ is the *prior* probability of class $\mathcal{D}_j$ (that is, the probability that a randomly selected observation belongs to class $\mathcal{D}_j$); denote it π_j and observe that $\sum_{j=1}^{J} \pi_j = 1$,
- $P(\mathbf{x} \mid \mathcal{D}_j)$ is the class-conditional density of the variables in $\mathbf{x}$ for class j, which under the assumption of a multivariate normal distribution of the variables in each class, with the class mean $\boldsymbol{\mu}_j$ and the class covariance matrix $\boldsymbol{\Sigma}_j$, will be represented by the Gaussian density function $f_j(\mathbf{x})$, [8]

$$f_j(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}\left|\boldsymbol{\Sigma}_j\right|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}. \tag{27}$$

We can now rewrite Formula (29) for posterior probability of class j, $j = 1,...,J$, as

$$P(\mathcal{D}_j \mid \mathbf{x}) = \frac{f_j(\mathbf{x})\pi_j}{\sum_{m=1}^{J} f_m(\mathbf{x})\pi_m}. \tag{28}$$

When classified based on this formula, the classified observation $\mathbf{x}$ will be assigned to class $\mathcal{D}_j$ with the largest posterior probability $P(\mathcal{D}_j \mid \mathbf{x})$. If posterior probabilities $P(\mathcal{D}_j \mid \mathbf{x})$, $j = 1,...,J$, are calculated under the assumption that all of the J class covariance matrices are equal (the homogeneity assumption), we will have Gaussian *linear* discriminant analysis, for which all boundaries between classes are linear. Otherwise, we will have Gaussian *quadratic* discriminant analysis.

3.1 Gaussian Linear Discriminant Analysis

Gaussian *linear discriminant analysis* (LDA), in addition to the multivariate normality assumption, makes the assumption of homogeneity of class covariances,

$$\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = ... = \boldsymbol{\Sigma}_J = \boldsymbol{\Sigma}. \tag{29}$$

[8] Where $\left|\boldsymbol{\Sigma}_j\right|$ denotes the determinant of covariance matrix $\boldsymbol{\Sigma}_j$ and $\boldsymbol{\Sigma}_j^{-1}$ its inverse.

To compare posterior probabilities of the J classes, we may first observe that the denominator of Formula (31) has the same value for all classes; thus, we need only to compare the $f_j(\mathbf{x})\pi_j$ terms. Remembering about the homogeneity of class covariances, from (30) we have

$$f_j(\mathbf{x})\pi_j = \frac{1}{(2\pi)^{p/2}\left|\mathbf{\Sigma}\right|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}\pi_j. \tag{30}$$

However, the first term on the right side of this formula is now constant with the same value for all classes, thus we may drop it. We may also take natural logarithm of the remaining part of Formula (33). As a result, finding the largest posterior probability $P(\mathcal{D}_j \mid \mathbf{x})$, $j = 1,...,J,$ is equivalent to finding the largest value of the following function (which can be considered a scaled posterior probability):

$$\begin{aligned}\delta_j(\mathbf{x}) &= -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_j) + \ln(\pi_j) \\ &= -\frac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x} + \mathbf{x}^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_j - \frac{1}{2}\boldsymbol{\mu}_j{}^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_j + \ln(\pi_j).\end{aligned} \tag{31}$$

Since the quadratic term $-\frac{1}{2}\mathbf{x}^T\mathbf{\Sigma}^{-1}\mathbf{x}$ is the same for all classes (the same covariance matrix and the same classified observation **x**), we may drop it and get a linear function in **x** for each class,

$$\delta_j(\mathbf{x}) = \mathbf{x}^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_j - \frac{1}{2}\boldsymbol{\mu}_j{}^T\mathbf{\Sigma}^{-1}\boldsymbol{\mu}_j + \ln(\pi_j), \qquad j = 1,...,J, \tag{32}$$

and thus a linear decision boundary between any two of J classes.

All that is left now in order to design an LDA classifier is estimating population parameters from the training data. Assume that the training data include N observations, each of them represented by a vector $\mathbf{x}_i = \left[x_{1i},\cdots,x_{pi}\right]^T$ and known to be from class y_i, $i = 1,...,N.$ Denote the class labels as $\mathcal{D}_1,...,\mathcal{D}_J,$ and the number of training observations in class j as $N_j, j = 1,...,J.$ If we do not know the prior probabilities of the classes in the target population, we estimate them from the proportions of training observations in each class,

$$\hat{\pi}_j = \frac{N_j}{N}, \qquad j = 1,...,J. \tag{33}$$

The mean for each class is estimated as

$$\overline{\mathbf{x}}_j = \frac{1}{N_j}\sum_{i:y_i=\mathcal{D}_j}\mathbf{x}_i, \qquad j = 1,...,J, \tag{34}$$

and the common covariance matrix as

$$\mathbf{S} = \frac{1}{N-J}\sum_{j=1}^{J}\sum_{i:y_i=\mathcal{D}_j}(\mathbf{x}_i - \overline{\mathbf{x}}_j)(\mathbf{x}_i - \overline{\mathbf{x}}_j)^T. \quad (35)$$

Consequently, the *linear classification functions* are

$$\delta_j(\mathbf{x}) = \mathbf{x}^T\mathbf{S}^{-1}\overline{\mathbf{x}}_j - \frac{1}{2}\overline{\mathbf{x}}_j{}^T\mathbf{S}^{-1}\overline{\mathbf{x}}_j + \ln(\hat{\pi}_j), \qquad j = 1,...,J, \quad (36)$$

and the observation represented by vector **x** will be classified to class *j* for which the classification function $\delta_j(\mathbf{x})$ has the largest value, that is

$$Class(\mathbf{x}) = \arg\max_j \delta_j(\mathbf{x}). \quad (37)$$

It is worth noting that if the prior probabilities are equal, that is, $\hat{\pi}_1 = \hat{\pi}_2 = ... = \hat{\pi}_J$, then — in terms of classification results— the LDA classification rule (40) is equivalent to Fisher's classification rule (28) (Johnson and Wichern 2007).

We may also look at this LDA classification rule from a geometrical point of view. With the LDA's assumption of equal class covariance matrices, the class areas are represented by equal-size hyperellipsoids centered about their class means, with hyperplane boundaries between the classes. If the prior probabilities of the classes are the same, an observation represented by vector **x** will be classified to class *j* whose center is nearest to **x**, when measured by the squared Mahalanobis distance $(\mathbf{x} - \overline{\mathbf{x}}_j)^T\mathbf{S}^{-1}(\mathbf{x} - \overline{\mathbf{x}}_j)$. In this case, a hyperplane boundary between any two classes crosses the line connecting the two class centers midway between the centers (although the hyperplane does not need to be orthogonal to this line, which would only be the case when the class areas are represented by hyperspheres). However, if the priors are not equal, the boundary will move towards the class with a lower prior probability (Duda et al. 2001).

Be aware that some authors call Functions (39) "linear *discriminant* functions". Yet, these *J* classification functions are very different from the (*J* - 1) Fisher's linear discriminant functions (described by Formula 22), which:

- define a projection from a *p*-dimensional space of the original variables into a (*J* - 1)-dimensional discriminatory space, in which the classes are maximally separated,
- have coefficients that are eigenvectors of $\mathbf{W}^{-1}\mathbf{B}$,
- are ordered by their discriminatory power, and
- facilitate low-dimensional visualization of the discriminatory space.

Since Functions (39) have none of the above characteristics, and since their purpose is to allow classification in the *p*-dimensional space of the original variables, it seems more appropriate to call them linear *classification* functions (Rencher 2002, Huberty and Olejnik 2006).

3.2 Gaussian Quadratic Discriminant Analysis

Gaussian *quadratic discriminant analysis* (QDA) assumes the multivariate normality, but does *not* make the assumption of homogeneity of class covariances. This means that instead of the LDA formula (33), we will have

$$f_j(\mathbf{x})\pi_j = \frac{1}{(2\pi)^{p/2}\left|\mathbf{\Sigma}_j\right|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}\pi_j. \tag{38}$$

Now we may only drop the constant $1/(2\pi)^{p/2}$; hence, after taking natural logarithm, we will have the QDA version of the scaled posterior probability,

$$\delta_j(\mathbf{x}) = -\frac{1}{2}\ln\left|\mathbf{\Sigma}_j\right| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j) + \ln(\pi_j). \tag{39}$$

Instead of estimating the common covariance matrix, we will now need to estimate separate covariance matrices for each class,

$$\mathbf{S}_j = \frac{1}{N_j - 1}\sum_{i:y_i=\mathcal{D}_j}(\mathbf{x}_i - \overline{\mathbf{x}}_j)(\mathbf{x}_i - \overline{\mathbf{x}}_j)^T, \quad j = 1,\ldots,J, \tag{40}$$

and have the following *quadratic classification functions*,

$$\delta_j(\mathbf{x}) = -\frac{1}{2}\ln\left|\mathbf{S}_j\right| - \frac{1}{2}(\mathbf{x}-\overline{\mathbf{x}}_j)^T\mathbf{S}_j^{-1}(\mathbf{x}-\overline{\mathbf{x}}_j) + \ln(\hat{\pi}_j), \tag{41}$$

which define quadratic boundaries between classes – that is, the boundary between classes k and l, $k,l = 1,\ldots,J$, $k \neq l$, is defined by $\delta_k(\mathbf{x}) = \delta_l(\mathbf{x})$. The form of the classification rule will be the same as described by Formula (40), but now observation **x** will be classified to class j corresponding to the largest value of the *quadratic* classification function.

3.3 Dimensionality Reduction in Gaussian Discriminant Analysis

In Fisher's version of discriminant analysis dimensionality reduction is intrinsic; that is, (J - 1) discriminant functions define a (J - 1)-dimensional discriminatory space, in which the ratio of between-to-within class variation is maximized, and in which classification is performed. This provides explicit dimensionality reduction from p to J - 1 (assuming, of course, that p is greater than J – 1). This, however, is far from obvious for Gaussian discriminant analysis. If we look at Gaussian classification functions —(39) for LDA and (44) for QDA— they are defined in a p-dimensional space of the original variables, and thus classification needs to be apparently performed in this space. Yet, we may recall that the centers of J classes will lie on a (J - 1)-dimensional hyperplane. Furthermore, even if the observation to classify is represented by a p-dimensional vector **x**, and we want to calculate distances between **x** and the class centers, we may ignore distances that are orthogonal to this hyperplane as they will be the same for each class. Hence, instead of

performing classification in the original p-dimensional space, we may project the observation's vector $\mathbf{x}$ onto the $(J - 1)$-dimensional subspace of class centers and perform classification there.

Furthermore, for LDA we may also find a sequence of linear discriminant functions that will be ordered by their importance for classification and thus, similarly as it was natively available for FDA, provide visualization of the classification space (and the data) in fewer than $J - 1$ dimensions (which would be particularly useful for $J > 4$). This approach is called reduced-rank LDA (Hastie et al. 2009), and will —surprisingly— result in the same discriminatory space as provided by the FDA, even if Fisher's analysis was done without the normality assumption. Consequently, LDA is equivalent to FDA. Moreover, although FDA does not make the multivariate normality assumption, if the classes *are* normally distributed, then the FDA solution will also minimize the probability of misclassification.

4 Partial Least Squares Discriminant Analysis

For discriminant analysis to work, we need to have more training observations than variables, and the more observations we have, the better it will work (well, the latter is true for any learning algorithm). For FDA and LDA, the total number of observations counts; whereas for QDA, the number of observations in the smallest class counts. For high dimensional data, this means that discriminant analysis cannot be used as a learning algorithm within feature selection schemas implementing a backward selection (for example, recursive feature elimination). However, it can be successfully used either after feature selection or within a feature selection schema using a stepwise hybrid search. Since, in multivariate predictive modeling, we are almost always interested in parsimonious solutions, these limitations of discriminant analysis are not really limiting its use. Consequently, there should be little reason for using this algorithm in situations when the number of training observations is smaller than (or similar to) the number of variables. Similarly, if high-dimensional data suffer from multicollinearity, properly designed multivariate feature selection should be performed instead of considering classification without the feature selection step. Nevertheless, one may perhaps imagine an unusual situation when the target variable really depends on very many independent variables (or, maybe, this was just assumed to be the case and no feature selection was performed), possibly with multicollinearities, and that there is a reason to use discriminant analysis in this situation. In such a case, the *partial least squares discriminant analysis* (PLSDA) could be the method of choice.

PLSDA combines the partial least squares (PLS) approach to supervised dimensionality reduction with classification via discriminant analysis. The PLS part works the same way it did for the partial least squares regression (PLSR) described in Lecture 1B, except that instead of having a continuous target variable as in PLSR, classes need to be encoded into the target. For binary classification, classes can be encoded with values 0 and 1 of a single response variable, represented by vector $\mathbf{y}$ of N values. When there are more than two classes ($J > 2$), we need J dummy response variables (alternatively, J classes can be

encoded by J - 1 dummy variables).[9] If we use J dummy response variables, they may be represented by a $J \times N$ matrix **Y,**

$$\mathbf{Y} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1N} \\ y_{21} & y_{22} & \cdots & y_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ y_{J1} & y_{J2} & \cdots & y_{JN} \end{bmatrix}, \tag{42}$$

whose rows represent J classes and columns represent N training observations. Each column will contain a single "1" value, corresponding to the class of this column's training observation (all other values in the column will be zero).[10]

The first stage of PLSDA is the same as for PLSR – a PLS algorithm is used to identify the projection of a p-dimensional space of the original variables (possibly with multicollinearities) into an m-dimensional space of orthogonal PLS components, where theoretically $m \leq p,$ but we are typically interested in having $m << p.$ As a result of this stage, we obtain a $p \times m$ matrix **C** whose columns represent m PLS components (see Lecture 1B). In the second stage, matrix **C** is used to project the training data into the space of the PLS components, and discriminant analysis (typically FDA or LDA) is performed in this m-dimensional space of those uncorrelated PLS components. An optimal number of PLS components m may be selected by cross-validation.

Note, however, that although the difficulties that the discriminant analysis would have had with high-dimensional or multicollinear data have been circumvented, no *feature selection* has been performed as the PLS components are linear combinations of all original p variables. Since it is typical for high-dimensional data to include many noisy or irrelevant variables, solutions provided by PLSDA, applied to such data, are likely to be suboptimal in terms of the predictive abilities of the PLSDA model as well as its interpretability. It is strongly recommended that —before deciding on using PLSDA — a serious consideration be given to multivariate feature selection.

References

Duda, R. O., Hart, P. E. and Stork, D. G. (2001). *Pattern classification* (2nd ed.). New York: Wiley.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics, 7*, 179–188.

[9] One of the classes will not have a dummy variable, but will be represented by zero values of all J-1 dummy variables.

[10] Software implementations of PLSDA usually perform the encoding of class information into dummy variables internally.

Fisher, R. A. (1938). The statistical utilization of multiple measurements. *Annals of Eugenics, 8*, 376–386.

Hastie, T., Tibshirani, R. and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction.* (Second ed.). New York: Springer.

Huberty, C. J. and Olejnik, S. (2006). *Applied MANOVA and discriminant analysis*. Hoboken, NJ: Wiley.

Johnson, R. A. and Wichern, D. W. (2007). *Applied multivariate statistical analysis* (6th ed.). Upper Saddle River, N.J.: Prentice Hall.

Rencher, A. C. (2002). *Methods of multivariate analysis* (2nd ed.). New York: Wiley.

Ripley, B. D. (1996). *Pattern recognition and neural networks*. Cambridge: Cambridge University Press.

Welch, B. L. (1939). Note on Discriminant Functions. *Biometrika, 31*(1/2), 218–220.

Required Reading

- **Lecture 3**

- **James et al:**
  - 4.4 Linear Discriminant Analysis
  - From 4.6 Lab:
    - *4.6.1 The Stock Market Data*
    - *4.6.3 Linear Discriminant Analysis*
    - *4.6.4 Quadratic Discriminant Analysis*

- **Kuhn & Johnson:**
  - 12.1: Case Study: Predicting Successful Grant Applications
  - 12.3 Linear Discriminant Analysis
  - 12.4 Partial Least Squares Discriminant Analysis
  - 12.7 Computing (Intro and sections: *Linear Discriminant Analysis* and *Partial Least Squares Discriminant Analysis)*

Suggested Reading

- **Izenman:**
  - 8. Linear Discriminant Analysis

- **Hastie et al:**
  - 4.3 Linear Discriminant Analysis (p. 106-112)

Research tasks for this unit (due April 5, 2025)

Task1:

Describe and discuss Fisher's and Gaussian discriminant analysis learning algorithms. Your narrative should reflect your understanding of the algorithms, thus, please try to refrain from using quotations.

Task 2:

Based on the information from the required reading assignments from James et al, perform the following discriminant analysis experiments:

a) Using *Smarket* data with six independent variables (***Volume* and five *Lag* variables**) perform LDA experiments; build an LDA classification model for predicting *Direction* (using training data), and then test it on the test data. (Please note that some objects, such as training and test subsets, are defined in 4.6.2).

b) Using the same *Smarket* data (as described for Task 2a) perform QDA experiments; build a QDA classification model (using training data), and then test it on the test data.

Task 3:

Based on the information from the required reading assignments from Kuhn & Johnson, as well as based on what you've learned from Task 2, perform the following:

a) Prepare *Grant* data for Tasks 3b) and 3c) experiments:

To prepare *Grant* data for these experiments, you need file ***unimelb_training.csv*** and script ***CreateGrantData.R***:

- save *unimelb_training.csv* in your RStudio working directory (I have uploaded the file to Blackboard, but you could also find it on *GitHub*).
- install the *AppliedPredictiveModeling* package and run the *scriptLocation()* command to locate *CreateGrantData.R;* then update (see below) and run this script in your RStudio environment.[11] Since the script is old and since R environment has changed, you need to make the following updates to the script:
  - add *stringsAsFactors=TRUE* to the *read.csv()* command,
  - add *options(expressions=15000)* near the beginning of the script.

  Please explain why these updates are needed to run the script under R 4.x.x.

[11] In my experiments with this script, parallel processing did not significantly decrease the running time, so to turn it off, you may change one line at the beginning of the script: from "`cores <- 3`" to "`cores <- 1`". However, if you want to run it in parallel, you need to use *doParallel()* instead of *doMC()*.

b) Using *Grant* data perform LDA experiments; build and test an LDA classification model.

c) Using *Grant* data perform *Partial Least Squares Discriminant Analysis* (PLS-DA) experiments: using the ***train*()** function identify a PLS-DA model with optimal number of PLS components, and then test this classification model.

Important instructions for your report:

For Tasks 2 and 3 experiments, provide description of all steps of your experiments, their results (including **confusion matrix** and at least the basic performance metrics: accuracy, sensitivity, and specificity; use the *confusionMatrix()* function from the *caret* package), and discussion of the results. Include — within your narrative for each step— the R code *used* at this step, as well as printouts of its most important results. The R code included at each step must be complete, that is, when copied from your report and executed, it has to work.

Note: **Set seed to 100** before any command that uses the random number generator (RNG), so your results are the same as expected.

Page limit: max 12 pages (plus references).

Very important: Make sure that you follow **all report requirements** as specified in Syllabus.