Report Assessment 2

Joana Levtcheva, CID 01252821

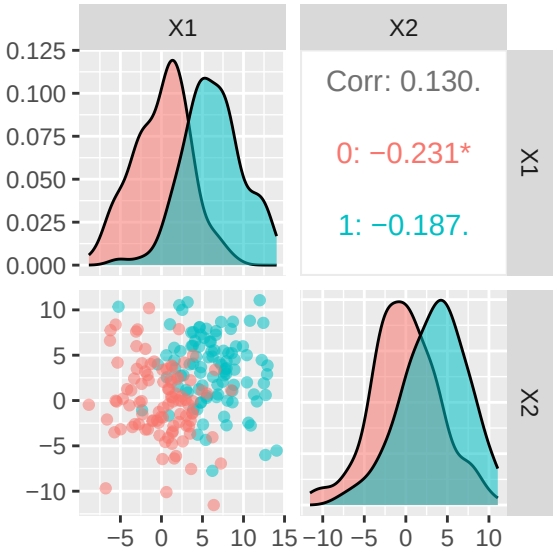
We are going to evaluate the behaviour of linear discriminant analysis (LDA) in settings where the underlying assumptions are violated, such as:the covariate data come from a distribution with a heavier tail than the normal, the sampling of classes is imbalanced, the data are `poisoned`.

Introduction to LDA

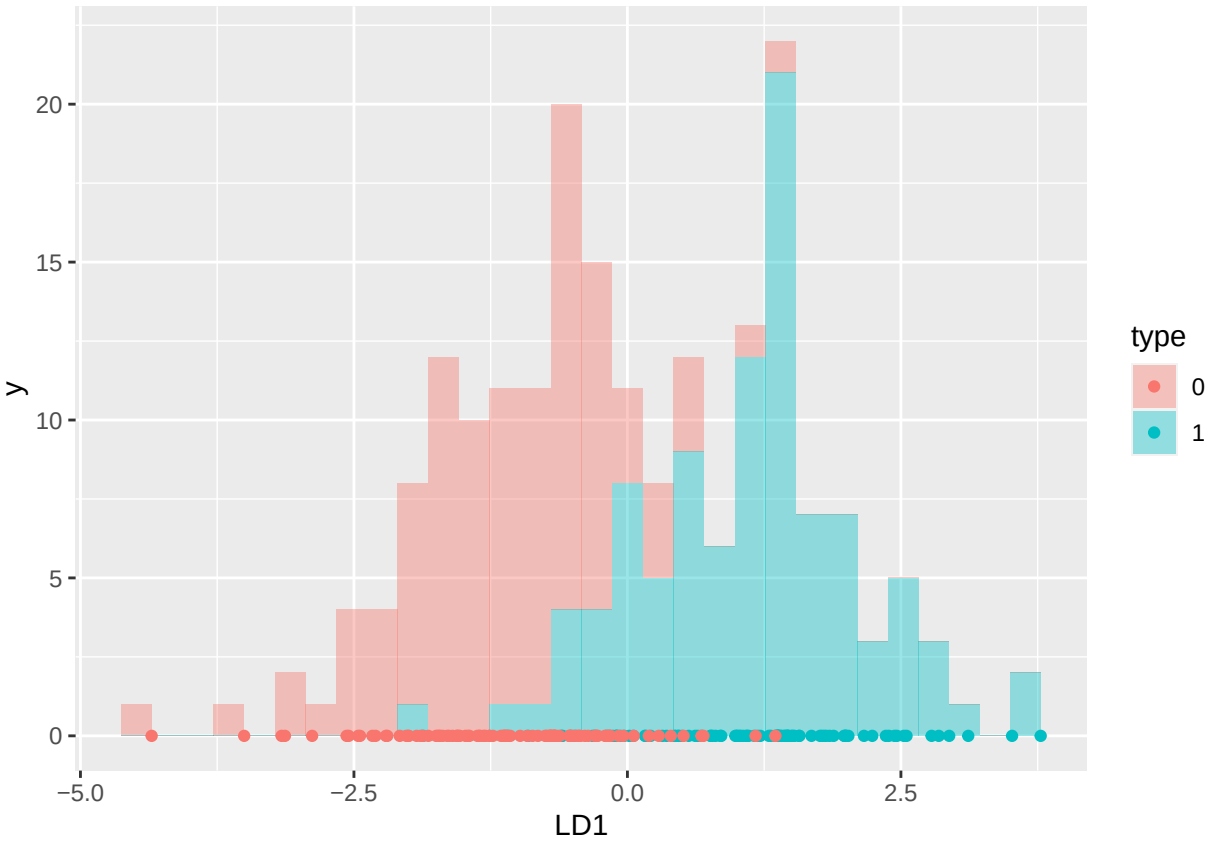
The idea of LDA is to maximise the separability among known classes.For example, let's have a dataset consisting of two covariate variables `X1` and `X2` where each one is assigned a `class` category of `1` or `0`. In our 2-dimensional case, the LDA will project the data onto a new axis, reducing the dimension from 2 to 1, in an attempt to maximise the separability. The new axes is constructed by maximising the distance between the two class means, and by minimising the variances within each class. Or, we want to maximise the expression $\frac{(\mu_1-\mu_0)^2}{\sigma_1^2+\sigma_0^2}$. To note, there are two important underlying assumptions: the mean vector μ_k may be different for each group, and the variance-covaraince matrix Σ is the same for all groups.

Behaviour of LDA following the underlying assumptions Let's analyse the case where no underlying assumptions are violated. We have simulated data such that the covariate variables `X1` and `X2` are drawn from multivariate normal distributions with sample size `200`, and whose mean vector depends on the value of the binary response variable `Y`, but with the same variance-covariance matrix. The response variable `Y`, named `class`, consist of only `1` and `0`, where each one has a count of 100 in the dataset.

Data summary:



After applying LDA in an attempt to classify the type of the class based on the covariates, we can plot a scatter plot of the data in the one-dimensional linear discriminant space, and the corresponding distributions.

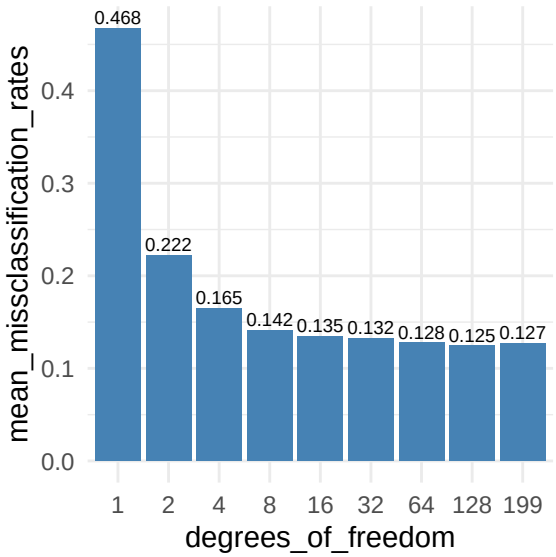


In order to evaluate the LDA performance we are going to divide the data into train (70% of the data) and test set, and calculate the missclassification error as $\frac{\text{False positives+False negatives}}{\text{total number of observations}}$ using the test set. If we evaluate the model based on one fixed simulated sample we would get `misclassification_rate = 0.1333`, but if we evaluate the model by taking a large number of simulated samples (1000) we would get a result of `misclassification_rate ~ 0.12`. We will use this value as e benchmark.

Now, for every setting with violated underlying assumptions we are going to simulate a large number of datasets that exhibit the misspecification, and evaluate the performance of the classifier on a withheld test subset. We are also going to show how the mean misclassification rate (estimated by averaging over many datasets) changes as the misspecification becomes more severe. For the purpose we are going to use the same method for simultaing the data as the one where the assumptions are not violated, but we are going to modify the parts where the assumptions should be violated.

Heavier tails: Student's t-distribution

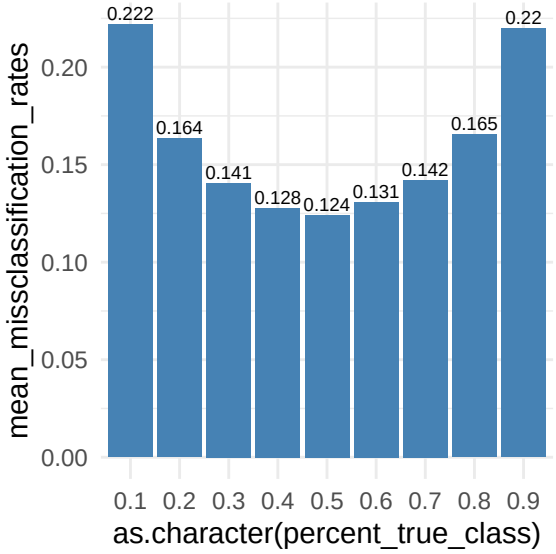
Here, the covariate variables `X1` and `X2` come from a distribution with a heavier tail than the normal. For example, the Student's t-distribution. In order to evaluate the misspecification behaviour we are simulating data from the distribution with different degree of freedom: `1, 2, 4, 8, 16, 32, 64, 128, 199`. For every degree of freedom we are simulating 1000 data sets, and calculating the mean missclassification error for each.



For the lowest degree of freedom, `df = 1`, the error is `~0.4` and it is the worst one amongst all mean classification rates. When the degrees of freedom become higher and are approaching 199 (initial sample size of 200 minus 1), the t-distribution should be approaching the normal distribution. The mean classification rates confirm this, as we can see the values for `df = 128` and `df = 199` are indeed `~0.12`, which is the benchmark value set earlier by the normal distribution.

Imbalanced class

The sampling of classes is imbalanced. This means that the proportions of observations in the two classes in the training sample are not representative of the proportions in the population. In this setting we are exploring how LDA performs if the percent of true class values are among `0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9`. In the initial setting the true values were 100 and the false values were also 100. In this case the true values are set as `30 + percent_of_true_values * 140` and the negative values as `30 + (1-percent_of_true_values) * 140`. In the initial setting the test data was set to contain 60 samples (30% of the data), that's why here we have the number 30.

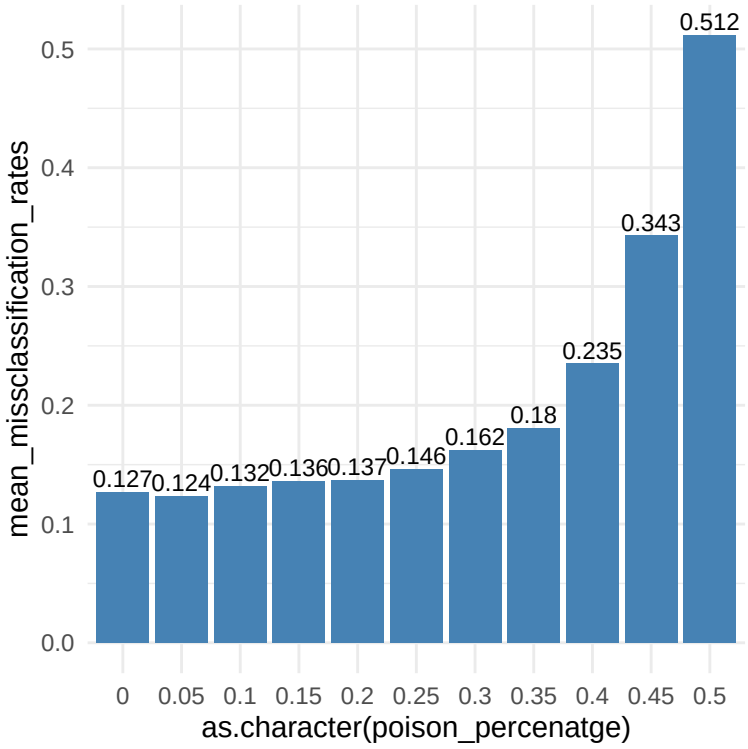


The results are showing that for the percent of true class values equaling 0.1 or 0.9, the performance is the worst. That's because the data is most imbalanced - the imbalance is 1:9 and 9:1 respectively. The case for 0.5 should be the optimal one, the data is considered balanced, and the mean classification rate of 0.124

confirms this, it's around the normal distribution benchmark. Therefore, the closer the percent of true class values to the optimal 50%, the better the LDA should be performing.

Poisoned data

In the last case, the data are `poisoned`. This means that the class labels of a small proportion of the training observations are switched. We are going to simulate 1000 data sets for each poison percenatge in `0, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.5`.



In this case, the best mean classification rate should be when the data has the lowest percenatge of poisoned data, or `0%`. The plot of the missclassification rates confirms this, and it is clear that the bigger the poisoned part of the data, the worse the rate. For `50%` of poisoned data we get `~0.5` error, whereas for the smaller percentages the rates are approaching the benchmark set from the normal distribution.