Multivariate Statistics – Homework 4

Note: Solve the tasks below using **R** and upload your solutions until Thursday, the 27th of May.

This exercise deals with **supervised learning**. As a classification method (categorical response) we use *linear discriminant analysis*, for regression (numeric response) we use *partial least squares regression*.

Task 1 – Classification: For this exercise we use the `OJ` (for *orange juice*) data set in the **R** package `ISLR`[1]. Our aim is to predict, which sort of orange juice (given in the variable `Purchase` with levels `CH` and `MM` for the two brands *Citrus Hill* and *Minute Maid*) a customer buys based on the other available variables in the data set.

1. **(2P)** Load the package and data set and get informed about the meaning of the variables on the help page. You will notice that some of the predictors contain numerical information (e.g. `PriceMM`), while others are categorical variables/factors (e.g. `Store7`). Are there variables contained in `OJ`, which should be factors, but are contained as numerical variables in the data frame? If yes, correct this.

2. **(2P)** Split the data in a training part ($\approx 3/4$) and test part ($\approx 1/4$). Do this randomly (but in a reproducible way). Use e.g. the `table()` function to determine the class proportions in the entire data set as well as in the training and test sets. Could there be problems, if the fraction of `MM` cases differs much between the subsets?

3. **(1P)** Set up a `LDA` model (`lda()` in package `MASS`, use `CV = TRUE`) on the training set to predict the purchase (`MM` or `CH`) using several other predictors. Which problem do you encounter?

4. **(2P)** We could group the $p = 17$ predictors into three subgroups:

   - predictors describing the time/location of the purchase: `WeekofPurchase`, `StoreID`, `Store7` and `Store`
   - predictors describing the price of the two products: `Price`, `Disc`, `Special`, `SalePrice` and `PctDisc` (all for `MM` and `CH` juices), `PriceDiff` and `ListPriceDiff`
   - a further predictor `LoyalCH`

   Go through the first two categories and determine predictors, which you could omit without loosing any information in the data **and** which might solve the problem above. For all further parts of the exercise, omit these predictors and only work with the remaining ones.

5. **(2P)** Set up a `LDA` model with your reduced predictor set (use the option `CV = TRUE`). Which percentage of misclassified observations do you get?

6. **(1P)** Depending on your choice (inclusion of predictor `LoyalCH` or not), recalculate the model (now with `CV = FALSE`) and apply it on the test set. Which fraction or percentage of misclassified objects do you observe?

Task 2 – Regression: For this exercise we use the `Boston` housing dataset in the **R** package `MASS`.

1. **(1P)** Load the data set, determine its dimensions and the number of missing values and use the `help` page to get information on the meaning of the columns. Are there variables contained in `Boston`, which should be factors, but are contained as numerical variables in the data frame? If yes, correct this.

[1] This is an **R** package accompanying the book *Introduction to Statistical Learning* by G. James, D. Witten, T. Hastie and R. Tibshirani.

2. **(1P)** We already know that *building a prediction model* usually involves some kind of optimization (*How complex/flexible shall the model be?*) and an estimation of the future prediction error we can expect, when this model is applied to new data. We also heard that performing these steps on the same data set will likely result in overfitting and too optimistic error estimates.

   Split the data randomly (but reproducible) into two parts – the **training set** (ca. $2/3$), on which we will build and optimize our models and the **test set** (ca. $1/3$), on which we will finally assess the model quality.

3. **(1P)** We start with a simple regression model for the median house value (`medv`) containing the average number of rooms per dwelling (`rm`) as only predictor. Create a bivariate scatterplot, add the regression line (in a different color) and determine, if this line describes the data well (why? why not?)[2].

4. **(3P)** Calculate the following two error measures: 1. $\text{RMSE}_{\text{train}}$, the root-mean-square error of this model on the training set and 2. RMSE_{CV}, the cross-validation RMSE. For calculating the latter, use the `cvFit()` function in the `cvTools` package. As the `help` page of the function tells us, it needs a fitted model as a first argument (`object`) as well as the arguments `data` (the data frame), `y` (the data column with the response values), `K` (the number of folds) and for reproducibility a `seed`.

5. **(3P)** Now apply `plsr` from **R** package `pls` to the training data. Have a look at the summary of the model. How do you interpret it? How many components would you choose?

6. **(2P)** What is the prediction error on the test data?

7. **(2P)** Finally let us repeat the analysis using argument `scale=TRUE`.

8. **(2P)** Visualize the measured versus the predicted values and interpret the results. What are your final conclusions on modeling the house prices in Boston?

[2] Keep in mind that for all model building and optimization steps the **training part** of the entire data set shall be used.