Guide to Assignment 1: Economic Growth

FEM11149 - Introduction to Data Science

Instructor: dr. A. Tetereva
TAs: L. de Wit & M. Praum

September 2025

Introduction

This is a guide for the first group assignment of the course Introduction to Data Science in the Data Science and Marketing Analytics Master program at the Erasmus School of Economics. Not all steps here have to appear in your final report. However, they are useful for you to learn and use in your life as data scientist whenever dealing with a dataset.

You should **not** use this file structure for your final report. Instead, you should organize your answers in the format of a business report, following the instructions from Canvas. This means that your team should **not** organize the document as 'question' followed by 'answer'! Instead, make a report and include the answers from the above questions in your text. Your report should be compiled using RMarkdown. Only the team captain will upload the report.

The grading for this group assignment follows the grading policy specified on Canvas (`https://canvas.eur.nl/courses/50901/pages/grading-policy-for-group-assignments-and-presentations`). Write after each name the percentage of the grade points that should be awarded to the person. If you have four team members who worked equally, write (25%) after each name. Check Canvas for more details and another example.

This document is structured as follows:

1. Motivation for the assignment
2. The task
3. R/R Markdown Appendix

Note: The dataset is different for each group. Hence, different groups will have different results.

1 Motivation for the assignment

You are part of a data science team that works as consultants for different governmental agencies on a wide range of projects. One of the agencies wants to know if, given a set of economic and demographic variables, one can predict the growth of a country's Gross Domestic Product (GDP). Namely, a country that wants to substantially increase its spending, may need to borrow money. Since the money also has to be paid back eventually, there is a need for economic growth. This economic growth can, for instance, be expressed in terms of the growth of the GDP.

Your manager believes that the Global Jobs Indicators Database from the World Bank would be a good place to start looking for data that can be used for this task. Furthermore, your manager has also recently learned from a powerful person in the government that the trade balance of a country is very important. Luckily, the World Bank also has data on the net trade in goods and services of different countries. The dataset **a1_data_group_x.csv** contains 150 rows, corresponding to different countries. In the 74 columns, one can find various economic and demographic variables, including the variable of interest: GDP growth. The meaning of the variables should be trivial from the names of the variables.

The data was obtained from World Bank: Jobs and World Bank: BoP, and was pre-processed for this assignment.

2 The task

After preparing the data (see Appendix), please answer the following questions:

1. Run a regression model using the variables you deem relevant from the dataset and include **Net trade in goods and services (BoP, current US$)** to explain **GDP growth (annual %)**. For that, use the **lm()** function explicitly writing the option **singular.ok = FALSE**.[1] You do not need to report all the process to run this model, but you should inform what is the model that you are working with and analyze why including **Net trade in goods and services (BoP, current US$)** to your old model is helpful, or not, and why. Make a connection to the theory behind OLS - explain in a way that your manager understands.[2]

2. Create a variable that gives a nonlinear or interaction effect. Does adding such a variable have an effect on the model diagnostics (see Lecture 2)? If so, what changes? Note that not putting all diagnostics in the main text will most likely cost too much space, so (some) part(s) of this question can be presented in the appendix.

3. Now run a penalized regression using LASSO to check if the model works. Do not mind adjustments at this stage, just run the model with the same variables as the regression you ran in the question before. Does the model work? Explain what is the difference in comparison to 2.

4. Why is it important to standardize predictors before applying penalized regression? Demonstrate with one example from your dataset.

5. After you show the results with LASSO, your manager is convinced that penalized regression is the way to go. Hence, she asks you to construct two additional penalized regression models to predict **GDP growth (annual %)**.

   - **even** teams work with ridge regression and elastic net.
   - **odd** teams work with LASSO and elastic net.

Warning: The **glmnet** package does not handle **data.frame** objects well (at least in prediction and cross validation). This may cause unexpected behavior. If this occurs, it means that you have to change your objects to have matrices. It can be interested to look into the **model.matrix()** function (simply using **as.matrix()** will not work because all your variables will be converted to strings).

6. Randomly divide your dataset into training samples and test samples of 110 and 40 observations, respectively.

7. For the training sample, run both methods, while choosing the optimal penalty parameter through cross-validation (for elastic net, you should optimize both α and λ). Report the final model for both methods (you can put the table in the appendix of the report). Give an interpretation, including which variables are more relevant for predicting GPD growth, and point out similarities and differences between the two models that you consider relevant.

[1] This guarantees that your model only fits if the matrix $\boldsymbol{X}$ is full rank, which can be important in situations where collinearity indicates a mistake in your data or model specification.

[2] If you load the data into R, the names of the columns can be slightly different compared to when you open the file of your team in, say, Excel. For instance, white spaces and percentage signs can be changed to dots. This should not lead to very significant problems, but it is something that you should keep in mind.

8. Compare the quality of the predictions from both models (using an appropriate metric). You will compute predictions using **lambda min** and **lambda 1se** that you found thanks to cross-validation, such that you have 4 models. Interpret the results.

9. Discuss why choosing λ purely based on minimizing cross-validation error may sometimes lead to overfitting. What is the role of the "1-SE rule" in mitigating this risk?

10. Even though your manager is capable of interpreting the results of the penalized regression, she is not sure if she understands it conceptually. Because why do we bother doing regularization in the first place, when we can just run a multiple linear regression model and remove all variables that are found to be insignificant. In this way, we obtain a perfectly fine model in only two steps, right? Argue why using penalized regression can be useful and why the reasoning of your manager is not really correct.

11. It might also be interesting to consider if a country grows more or less than most other countries. Create a new variable that classifies a country as **Growing more** in which you record values of GDP growth above 2.7% as 1 and below 2.7% as 0.

glm() also works with logistic regression. For that, you just need to add an extra parameter **family = "binomial"**. The same argument can be used with the function **cv.glmnet()**.

12. Construct a logistic model using Ridge regression that has the binary variable **Growing more** as response variable. Report your final model and compare the predictive performance on the test data, again comparing **lambda min** and **lambda 1se**.

13. For these particular models (that is, question 12) also examine the models once for a different train-test split. You do not have to report the final model. You only have to consider the predictive performance. Is the performance different? Is that something that you would have expected beforehand.

14. Wrap up based on what you have learned with this assignment. Some suggestions are:
    - Which model is the best model for predicting **GDP growth (annual %)**?
    - If you could acquire more data, would you prefer having more variables (columns), or more countries available (more rows). Why?
    - When is it interesting to predict GDP growth itself, and when would it be more interesting to consider a model that predicts the variable **Growing more**?

Some hints:

1. For penalized regression, you should use cross-validation whenever applicable. Remember that in elastic net you cross validate two parameters, there is a video on Canvas explaining how to do this.

2. Put your code at the end of the report, as an Appendix. Make sure your code is clear.

3. You saw in the lecture how the **glm()** function can be used for elastic net. Look into the documentation to discover how to use it for LASSO and Ridge regression. You will need the package that has the same name.

4. Whenever dividing your dataset for training and test sets, you need to do it properly: randomize without replacement. Remember to define a seed using **set.seed()** to make your results reproducible.

5. Start your code early, even if you do the written analysis later. This way you have time to ask for help in case something does not work in your RMarkdown file.

Your report should follow the guidelines specified on Canvas and should be in pdf format, compiled using RMarkdown. Do not change font size or margin sizes, otherwise you will loose points. Only the team captain will upload the report.

A Starting your R Markdown document

We are going to solve this assignments using the statistical software program RStudio based on the programming language R. RStudio is available in the PC labs at Erasmus University. You can also download R and RStudio for free from https://www.r-project.org/ and https://rstudio.com/products/rstudio/download/ respectively. Be aware of the fact that you have to install R first and then install RStudio. For more information, check the video on Canvas, R and RStudio installation.

A.1 R

For R beginners, the first operator you use is probably the assignment operator <- which evaluates the expression on its right side and assigns the evaluated value to the symbol on the left. You can also use the equal sign = instead of <-. In practice, they almost always work the same (except when declaring function arguments, but we will get there). Our recommendation is to stick with a convention to assign variables: either use <- or =.

R is a free software and the base program that you download when installing R does not come with all the resources that are available. The R community develops what we call packages, that are a set of files and functions that you load on R and can also use. R packages that are available on CRAN (The Comprehensive R Archive Network). CRAN is the network where all official versions of R and packages are available. If a package is available at CRAN, it means that it satisfies the basic requirements in terms of usability and functionality, but you can also find packages from people that just upload on their websites, for example. If this is your first time using R, it is best to stick with the packages from CRAN before using other tools.

A.2 R Markdown

In the R Markdown video available on Canvas you learn the structure of a R Markdown document. On top of the document there is the header, where you can add options including a bibliography file. In an RMarkdown file, you write text using plain text and Markdown (click here for a cheatsheet or check the files on Canvas), but you can also use equations in LaTeX. And, more important, you can add chunks (blocks) of code or call R inside the text (inline coding).

A.3 Calling Packages

Packages need to be installed when you are going to use for the first time and loaded every time you are going to use them. The most basic commands for that are **install.packages(“package_name”)** and **library(package_name)**, where **package_name** you replace by the actual name of the package. For example:

```r
install.packages("MASS")    # Note that you need the quotes
library(MASS)               # Now you don't need the quotes
```

The problem with this approach is that you end up (re)installing packages even when they are already installed. A way to avoid this is to use the **pacman** package. It will first check if a package is installed and then load it. If it is already installed, only the loading part is executed.

```r
# Verify if a package is installed, if not, download and install before loading
## If ind = 10 doesn't work, try changing to other number.
## See the full list by running chooseCRANmirror() in your console
## This is to choose the mirror to download the files
chooseCRANmirror(graphics = FALSE, ind = 10)
# This install pacman in case it is not installed
if (!require("pacman")) install.packages("pacman")
# This used the function p_load from the pacman package to load the other packages
pacman::p_load(plyr, knitr)
```

You do not need to load all the packages at the beginning of the document. In fact, you can load them as you need. However, loading them all together at the begining of the document helps to keep your code structured and organized.

B Reading external files in R

In the the R Markdown video, you can learn about various ways of loading files in R. Since we are dealing with a csv file, we will be using the **read.csv2()** function. This function comes in the **utils** package, which is already pre-installed and loaded in R. If you want to know more about this function, type **help(read.csv2)** in your console.

The dataset will be read from a directory in the computer, you can download the file from Canvas. The code below will first set as working directory the same folder as the one that your code is saved. So you first need to save the code file somewhere. Then, it will read the contents from the file 'and store in the variable **dfTrain**'. Notice that you will have to modify the code such that the file name matches the name of the file you have for working in the Assignment.

```r
# Directory setup
path = dirname(rstudioapi::getSourceEditorContext()$path) # Path is directory of this file
setwd(path) # Set working directory

dfTrain <- read.csv2("a1_data_group_30.csv", sep = ",", dec = ".", header = TRUE)
```

To check if you imported something, you can use **head()** and **tail()**. This allows you to check the first and last 6 lines of an object.

C Checking a data set: descriptive analysis

We first look at the summary statistics of the variables. Even if this is not asked for your assignment, you should do this data checking to see if the data corresponds to what you were expecting to have. Moreover, when dealing with 'real data', doing the descriptive analysis will help you in cleaning the file, as well as having insights of the analysis you can do with it. A large portion of your time working in data science projects is importing and cleaning data, even though this is not always a big part of the final report.

```r
# Give summary of the first five variables in
# dfTrain
summary(dfTrain[, 1:5])
```

The function **summary()** creates a summary table for each column of the dataset. Continuous variables will have displayed their minimum value, quartiles, mean and maximum. Qualitative variables are not very informative because nothing is displayed except the length (number of observations) and class. The exception will be qualitative variables that can be grouped as factors: for those we will be able to build frequency tables (we will see that in the recoding part, below).

D Having a bibliography file

In case you use references, you can include them in the RMarkdown file in an automatic way. You see on Canvas a file named **references.bib**, and the file is being called on the header of this rmd document. If you open this file (you can use either a TeX editor or notepad) you will see a single reference, that has a BibTeX format.

```
@book{ISL,
  title={An introduction to statistical learning},
  author={James, Gareth and Witten, Daniela and Hastie, Trevor and Tibshirani, Robert},
  volume={112},
  year={2013},
  publisher={Springer}
}
```

The first thing that appear after the bracket is the alias that you use to 'call' the book in your text. In RMarkdown, you need to use the @ sign, followed by this alias:[@ISL] results in (James et al., 2013). When you call a reference, RMarkdown will automatically add it in the very end of the document. So we recommend to end your file with a title format for references (even if not is being displayed in your code, it will appear after kniting the file).

To get a citation already in BibTeX format, you can search for the book or paper in Google Scholar, then click on the quotes that appear below the reference, and click in BibTeX on the pop-up. This will go to another page with only the code, that you can copy and paste in your **references.bib** file.

If you do not want to use a bibliography, then just don't call any reference in the file and erase in the header the command calling **references.bib**.

References

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani (2013). *An introduction to statistical learning: with applications in R*. Vol. 103. Springer.