Data Analysis Methods 2015
Assignment 3

Aasa Feragen, Christian Igel

In Assignment 3 you will work with regression and linear algebra.

Assignment 3 will be made available 8.3 and your report should be uploaded to Absalon no later than 18.3.

Guidelines for the assigment:

- **The assignments in DAM must be completed and written individually.** This means that your code and report must be written completely by yourself.
- You are, however, encouraged to discuss in small groups while working on the assignments, If you do this, please list your discussion partners in the beginning of your report.
- In DAM, we will be using an automatic code checking system for some of the exercises. These exercises are written in blue text below, and will be evaluated solely based on your uploaded code. Your solutions to these assignments should therefore be uploaded to the automatic code checking system. You can do this at any time, as many times as you want, until the deadline, and your auto-generated feedback will be immediately available.
- Upload your report as a single PDF file (no Word) named `firstname.lastname.pdf`.

Linear Regression: Prediction lung function

It is well known that smoking is not good for your lungs, but how can we prove this? In Assignment 1, you worked with the dataset `smoking.txt` which you can find on Absalon. This dataset contains data collected from 654 youth and children aged 3-19, and in Assignment 3 you will predict their lung function in terms of FEV1 using different regression models (see the Appendix for a description of FEV1).

In Assignment 1, you should have concluded that

i) in this dataset, smokers have better lung function than non-smokers, and
ii) this was caused by a confounding factor: The subjects in the study are children and youth, and while young children do not smoke, they also have smaller lungs and lower lung function than older children and youth.

In Assignment 3, you will predict lung function from the other measurements associated to the data using linear regression.

Exercise 1 (Linear regression with a single variable)**.** In this exercise, your task is to predict FEV1 from age using the univariate linear regression model

$$y = \alpha + \beta x + \varepsilon,$$

where y is the predicted ouput variable, x is the input variables, the w_i are free parameters and ε is a noise term that captures all other factors affecting y other than the input variables. The parameters α and β *define* the regression model, and once they have been estimated, the model can be used to predict outputs y associated to new input values x.

We generate the following model for our problem:

$$\text{FEV1} = \alpha + \beta \cdot \text{age} + \varepsilon.$$

a) Implement linear regression in a similar way as in Chapter 14 of the textbook [1], using the supplied template function `univarlinreg.py`. Your code should take as input the input vlues x and output values y of the data and output an estimate of the parameters α and β. You should not use built-in functions for regression. Please upload your code to the automatic code checking system.

b) Apply your implementation to the entire dataset, and record your α and β values. Plot your data points together with your estimated regression line. Remember to label your axes and your plot. Describe what you see and discuss briefly how the variable age relates to FEV1 according to your model.

c) Implement a function `rmse.py` using the supplied template, which computes the root-mean-square error

$$RMSE(\alpha, \beta) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \|y_i - (\alpha + \beta x_i)\|^2}$$

for a set of known input-output values $(x_1, y_1), \ldots, (x_N, y_N)$. Here, y_i is the recorded output value associated to the i^{th} data point x_i in the dataset, and $(\alpha + \beta x_i)$ is the output value associated to the input x_i as predicted by your regression model. Your code should take as input the input values x and output values y, as well as the α and β parameters defining your regression model. Please upload your code to the automatic code checking system.

d) Select 450 random subjects as a test set and use only these subjects to build the regression model. Use the remaining 204 subjects to compute $RMSE(\alpha, \beta)$ of your model.

Deliverables. a) Uploaded code, b) the α, the β, a plot of both your data and the regression line obtained through regression, and a one-liner, c) uploaded code, d) the resulting RMSE.

Exercise 2 (Multivariate linear regression)**.** In this exercise, you will predict FEV1 from *all* the other input variables given in the dataset using multivariate linear regression

$$y = w_0 + w_1 x_1 + \cdots + w_d x_d + \varepsilon,$$

where y is the predicted ouput variable, x_i are the input variables, the w_i are free parameters and ε is a noise term that captures all other factors affecting y other than the input variables. The parameters w_i *define* the regression model and, just like for univariate regression, once they have been estimated, the model can be used to predict outputs for new input values $x_1, \ldots, x_k$.

We generate the following model for our problem:

$$\text{FEV1} = w_0 + w_1 \cdot \text{age} + w_2 \cdot \text{height} + w_3 \cdot \text{gender} + w_4 \cdot \text{smoking status} + w_5 \cdot \text{weight} + \varepsilon.$$

a) Implement linear regression either as in the lecture slides or as in Chapter 15 of the textbook [1], using the supplied template function `multivarlinreg.py`. Your code should load the data matrix X containing the input variables, as well as the output vector y, and output an estimate of the free parameters in the model, that is the w_i in the form of a vector $\mathbf{w}$. Remember the offset parameter w_0. You should not use a built-in function for regression. Please upload your code to the automatic code checking system.

Hint: If you use the analytical solution of regression presented in the lecture slides, note that it is important to extend the data matrix with a column of coefficients 1 corresponding to the free parameter w_0 in the above model. Describe your result and discuss briefly how the different variables correspond to FEV1.

b) Run your regression function on the entire dataset and record your estimated parameters w_i. What do they tell you about how lung function relates to the different input variables? Does smoking affect FEV1?

c) Using the same randomly selected training and test sets as in the previous exercise, estimate the free parameters w_i for the model, and use these to predict output values on the test set. Run your RMSE function on the true and estimated output values for the test set. What do you see?

Deliverables. a) Uploaded code, b) Parameters w_i and at most three lines of discussion, c) your RMSE and a one-liner.

Logistic regression

In this exercise, you will compare the performance of logistic regression to the performance of your k-NN classifier implemented in Assignment 2. Please download the DD dataset found on Absalon in the file `DD.tar.gz`.

Exercise 3 (Logistic regression)**.** a) Implement logistic regression as described in Chapter 16 in the book [1] or in the lecture. Your function should take as input the training set data matrix, the training set label vector, and the test set data matrix. Please upload your code to the automatic code checking system.

b) Train the logistic regression on the training set and run it on the test set. Threshold the returned probabilities at 0.5 to obtain a binary classification, and compute the test error in percentage.

c) Run your implemented k-NN algorithm from Assignment 2 on the same dataset, selecting an optimal k by using cross validation on the training set as in Assignment 2. Which classifier works best on this dataset?

Deliverables. a) Uploaded code, b) the test error, c) the test error for the optimal k-NN, and a one-liner.

The data material

The smoking dataset

The file `smoking.txt`, which can be found in Absalon, contains a 654×6 matrix, where each column corresponds to (in the given order):

- age – a positive integer (years)
- FEV1 – a continuous valued measurement (liter)
- height – a continuous valued measurement (inches)
- gender – binary (female: 0, male: 1)
- smoking status – binary (non-smoker: 0, smoker: 1)
- weight – a continuous valued measurement (kg)

This data is collected from 654 youth and children and each row in the matrix can thus be considered as an observation describing one child/youth.

Measurement of lung function

Lung function can be measured using a *spirometry* test, and several parameters are computed based on the result. One of these parameters is the *forced expiratory volume in one second* (FEV1), which measures the volume that a person can exhale in the first second of a forceful expiration after a full inspiration. This measure will be used as an indicator of lung function in this assignment. A decrease in FEV1 generally indicates a decrease in lung function.

The D&D dataset

The D&D dataset consists of 1178 protein structures [2] represented as graphs, from which *degree distributions* are extracted. Each protein is represented by a graph whose nodes are amino acids, which are connected by an edge if they are less than 6 Ångstrøms apart. Some of the proteins are enzymes and some are not; we shall train a classifier to predict whether or not an unseen protein is an enzyme.

The degree distribution, which we take as a vector representation of each graph, is generated in the following way: For each node in each graph, record its *degree* – that is, how many edges connect that node to other nodes in the graph. For each graph G, generate a count vector $g \in \mathbb{R}^d$, where the coordinate g_i counts how many nodes of degree i there were in the graph G.

You are given the $D\&D$ dataset split into a *training* set with 450 proteins, and a *test* set with 204 elements, along with their corresponding enzyme/non-enzyme classification labels.

References

[1] J. Grus, *Data Science from Scratch*, 2015.

[2] P.D. Dobson and A.J. Doig, *Distinguishing enzyme structures from non-enzymes without alignments.*, J. Mol. Biol., 330(4):771-783, 2003.