Data Mining
Graded Exam Assignment 2013:
Sunspots and Seeds

Christian Igel
Department of Computer Science
University of Copenhagen

This is the graded exam assignment for the Data Mining part of the course *Databases and Data Mining* at the University of Copenhagen.

This assignment must be made and submitted individually. However, feel free to discuss the solution of the assignment with your fellow students. Submissions in English are preferred, but submissions in Danish are also accepted.

The assignment will be graded using the 7-point scale. This will be combined with the grades of the previous exams on databases to give the final grade for the course. To obtain the best grade of 12 in this assignment, you must fulfill all the learning objectives at an excellent level. In terms of the questions in this assignment, this means that you have to answer all questions with none or only a few mistakes or parts missing. To obtain the passing grade of 02, you need to fulfill the learning objectives at a minimum level.

Solution format

The deliverables for each question are listed at the end of each question. The deliverable "description of software used" means that you should hand in the source code you have written to solve the problem. If you have used a tool to solve the problem, this tool should be described and reasons for the particular choice of this tool should be given.

Thus, a solution should contain:

- A report with detailed answers to the questions. Describe the way you solved the problems. Your report should include graphs and tables with comments if needed (**max. 8 page of text** including figures and tables). Use meaningful labels, captions, and legends.

The way you solved the problems and your results must be comprehensible without looking a the attached source code.

- Your solution code (preferable Python scripts or C++ or Java code) with comments about the major steps involved in each question. The code must be submitted in original format (i.e., not as `.pdf` files). Use meaningful names for files, constants, variables, functions and procedures etc.

  Your code should also include a README text file describing how to compile (if necessary) and run your program. It should also contain a list of all required libraries. If you use the SHARK machine learning library, you need not include the library in your submission. If we cannot make your code run we have to consider your submission to be incomplete.

Database

All data considered in this assignment are stored in a single SQLite database named `DataMiningAssignment2013.db`.

There are two tables per data set. The tables ending with `_X` contain the input data and the tables ending with `_Y` the corresponding target (output) data. That is, the nth row of the table ending with `_Y` contains the label or response given the attributes in the nth row of the corresponding table ending with `_Y`.

The file `readData.py` gives an example how to access the data using Python.

The database as well as the source code are available from the course page in the Absalon system.

1 Sunspots Prediction

1.1 Background

Consider the data in the tables `Sunspots_Train_X` and `Sunspots_Train_Y` (years 1704–1903) as well as `Sunspots_Test_X` and `Sunspots_Test_Y` (years 1904–2011). They are based on the yearly sunspot number data provided by the Sunspot Index Data Center (SIDC), see `http://sidc.oma.be`. On `http://www.icsu-fags.org/ps11sidc.htm` we find the following short introduction to the sunspot data:

> Sunspots are extended regions on the Sun with a strong magnetic field. They have a lower temperature (3500-4500 K) than the surrounding photosphere (5800 K). The sunspots radiate less energy than the undisturbed photosphere of the Sun and are therefore visible as dark spots on the surface of the Sun. Sunspots are observed with

> some regularity since 1700 and on a strict daily basis since 1849; the relative number [...] (defined as ten times the number of groups + the number of spots) shows an 11 year cycle detected by Schwabe in 1843. The sunspot number reflects the magnetic activity of the Sun, which has a large impact to the magnetosphere of the Earth and is responsible for e.g. magnetic storms, polar lights.

In this assignment, the task is to predict the average number of sunspots in a year t based on the average numbers in the previous four years $t-1$, $t-2$, $t-3$, and $t-4$. The four predictor (input) variables are in the tables ending with `_X` and the corresponding response (output) variable in the tables ending with `_Y`. That is, each entry in a `_Y`-file is the average number of sunspots of a certain year and the corresponding four entries in the associated `_X`-file are the numbers from the previous four years.

1.2 Mean and sample variance

Let the output data in `Sunspots_Train_Y` be given by $y_1, \ldots, y_\ell$. Compute the sample mean

$$\hat{\mu} = \frac{1}{\ell} \sum_{i=1}^{\ell} y_i$$

and the *biased sample variance*

$$s^2 = \frac{1}{\ell} \sum_{i=1}^{\ell} (y_i - \hat{\mu})^2 \ .$$

Deliverables: mean and biased sample variance of the number of sunspots in the training data set

1.3 Linear regression

The goal of our modeling is to find a mapping $f : \mathbb{R}^4 \to \mathbb{R}$ for predicting the number of sunspots based on previous observations.

1.3.1 Build model

Build an *affine* linear model of the data using linear regression and the training data in `Sunspots_Train_X` and `Sunspots_Train_Y` only. Report the five parameters of the model.

1.3.2 Training error

Determine the training error by computing the mean-squared-error of the model over the complete *training* data set.

Compare this mean-squared-error with the biased sample variance calculated above. Have a look at the definitions of both quantities and briefly describe what it means if the mean-squared-error is below or above the biased sample variance.

1.3.3 Test error

Compute the mean-squared-error on the test data set (i.e., use `Sunspots_Test_X` and `Sunspots_Test_Y`). Comment very briefly on the result.

1.3.4 Visualize prediction

Plot the sunspots time series from year 1904 to 2011 (years on the x-axis, average number of sunspots on the y-axis) using the data in `Sunspots_Test_Y`. Add the prediction of your model (i.e., the predictions you used to compute the test error) to the plot to visualize the quality of your model. Comment briefly on the result.

Deliverables: description of software used; parameters of the regression model; mean-squared error on the training and test data set; brief discussion relating mean-squared-error to the biased sample variance, visualization of the test data (number of sunspots over time) and the corresponding model output; short discussion of results on the test set

2 Seeds

2.1 Background

This classification task addresses the problem of identifying a variant of wheat based on properties of its seed. The seed from which the wheat plant grows is called wheat kernel or wheat beery. We consider data available from the well-known UCI benchmark repository [Frank and Asuncion, 2010] (`http://archive.ics.uci.edu/ml/datasets/seeds`).

Charytanowicz et al. [2010] measured seven geometrical properties of kernels belonging to three different types of wheat, namely *Kama*, *Rosa*, and *Canadian*. The seven features are listed in see Table 1.

Consider the tables `Seeds_Train_X` and `Seeds_Train_Y` containing training data and the tables `Seeds_Test_X` and `Seeds_Test_Y` containing test data. The inputs

Table 1: Brief description of the input attributes for wheat classification, see the article by Charytanowicz et al. [2010] for details.

feature index	description
0	area A
1	perimeter P
2	compactness $C = 4\pi A/P^2$
3	length of kernel
4	width of kernel
5	asymmetry coefficient
6	length of kernel groove

are the seven geometric features and the corresponding label encodes the wheat variant by an integer.

2.2 Classification

Train a nearest neighbor classifier (1-NN) using the training data stored in the tables `Seeds_Train_X` and `Seeds_Train_Y`. Measure its performance on the test data stored in `Seeds_Test_X` and `Seeds_Test_Y`. How high is the classification accuracy on the test data?

Deliverables: description of software used; test accuracies of nearest neighbor classifier

2.3 Dimensionality reduction and visualization

Perform a principal component analysis of the input attributes of the training data in `Seeds_Train_X`.

Plot the eigenspectrum. How many components are necessary to "explain 90 % of the variance"? Visualize the data by a scatter plot of the first two principal components using a different color for each class. Briefly discuss the results.

Deliverables: description of software used; plot of the eigenspectrum; number of components necessary to explain 90 % of variance; scatter plot of training data projected on first two principal components of the training data; brief discussion of results

References

M. Charytanowicz, J. Niewczas, P. Kulczycki, P. A. Kowalski, S. Łukasik, and S. Żak. Complete gradient clustering algorithm for features analysis of x-ray images. In E. Piętka and J. Kawa, editors, *Information Technologies in Biomedicine*, volume 69 of *Advances in Intelligent and Soft Computing*, pages 15–24. Springer, 2010.

A. Frank and A. Asuncion. UCI machine learning repository, 2010. URL `http://archive.ics.uci.edu/ml`.