Lab Course Machine Learning
Exercise Sheet 2

Prof. Dr. Dr. Lars Schmidt-Thieme, Mohsan Jameel
Information Systems and Machine Learning Lab
University of Hildesheim

October 27th, 2016
Submission on November 3rd, 2016 at 11:55pm, (on moodle, course code 3112)

Exercise 1: Data Analysis with Pandas (10 Points)

Download *house.csv* from `http://jgscott.github.io/teaching/r/house/house.csv`. This dataset contains information about the sales price of houses along with other attributes. You will analyze this dataset using pandas library and plot some interesting information using matplotlib library.

1. Load the data using pandas
2. Summarize each field in the data, i.e. mean, average etc.
3. Group data by the field *nbhd*.
   (a) Give average *sqft*, average *price* and average *bedroom* of each group.
   (b) Plot for each field (sqft, bedroom, price etc). Use a boxplot that visualizes the statistical information about them.
   (c) For each group of *nbhd*, draw a prediction line for *price* vs *sqft* (similar to the one in Lab 1).

Set of methods for matplotlib `http://matplotlib.org/examples/index.html`

Exercise 2: Linear Regression via Normal Equations (10 Points)

In this exercise you will implement (multiple) linear regression using Normal Equations. See lecture (slides: 2-15) `https://www.ismll.uni-hildesheim.de/lehre/ml-14w/script/ml-02-A1-linear-reg.pdf`.The learning algorithm is given on the slide 9.

1. Reuse house.csv dataset from Excercise 1. Load it as X_{data}, [Hint:] from loaded data you need to separate y_{data} i.e. sales prices of houses, which is your target.
2. Choose those columns, which can help you in prediction i.e. contain some useful information. You can drop irrelevant columns. Give reason for choosing or dropping any column.
3. Split your dataset X_{data}, y_{data} into X_{train}, y_{data} and X_{test}, y_{test} i.e. you can randomly assign 80% of the data to a X_{train}, y_{train} set and remaining 20% to a X_{test}, y_{test} set.
4. Implement *learn-linreg-NormEq* algorithm and learn a parameter vector β using X_{train} set. You have to learn a model to predict sales price of houses i.e. , y_{test}.
5. Line 6, in *learn-linreg-NormEq* uses *SOLVE-SLE*. You have to replace *SOLVE-SLE* with following options. For each option you will learn a separate set of parameters.

(a) Gaussian elimination

(b) Cholesky decomposition

(c) QR decomposition

6. Perform prediction $\hat{y}$ on test dataset i.e. X_{test} using the set of parameters learned in steps 5 and 6 (Hint. you will have three different prediction models based on the replacement function from step 6).

7. Final step is to find how close these three models are to the original values.

   (a) plot residual $\epsilon = |y_{test} - \hat{y}|$ vs true value of y_{test} for each model.

   (b) Find the average residual $\epsilon = |y_{test} - \hat{y}|$ of each model.

   (c) Find the root-mean-square error (RMSE) = $\sqrt{\frac{\sum_{n=1}^{N}(y_{test}(n)-\hat{y}(n))^2}{N}}$ of each model.

Annex

1. You can use numpy or scipy in build methods for doing linear algebra operations.

2. You can use pandas to read and processing data

3. You can use matplotlib for plotting.

4. You should not use any machine learning library for solving the problem i.e. scikit-learn etc. If you use them you will not get any points for the task.