Lab Course Machine Learning
Exercise 3

Prof. Dr. Dr. Lars Schmidt-Thieme,
Mofassir ul Islam Arif
Information Systems and Machine Learning Lab
University of Hildesheim
Submission: 25.11.2018 LearnWeb 3112

November 18, 2018

1 Exercise Sheet 3

1.1 Data preprocessing (5 Points)

1.1.1 Datasets

You are required to pre-process given datasets.
Airfare and demand: http://www.stat.ufl.edu/ winner/data/airq402.dat [1]
Wine Quality: http://archive.ics.uci.edu/ml/datasets/Wine+Quality [2]
You are required to pre-process given datasets.

- 1. convert any non-numeric values to numeric values. For example you can replace a country name with an integer value or more appropriately use hot-one encoding. [Hint: use hashmap (dict) or pandas.get_dummies]. Please explain your solution.
- 2. If required drop out the rows with missing values or NA. In next lectures we will handle sparse data, which will allow us to use records with missing values.
- 3. Split the data into a train(80%) and test(20%) .

1.2 Linear Regression with Gradient Descent (15 Points)

Part A: (8 Points): Implement Linear Regression with Gradient Descent
In this part you are required to implement linear regression algorithm with gradient descent algorithm. Reference lecture https://www.ismll.uni-hildesheim.de/lehre/ml-16w/script/ml-02-A1-linear-regression.pdf
For each dataset given above

- 1. A set of training data $D_{train} = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(N)}, y^{(N)})\}$, where $x \in R^M, y \in R$, N is number of training examples and M is number of features

- Linear Regression model is given as $\hat{y}^n = \sum_{m=1}^{M} \beta_m x_m^n$
- Least square loss function is given as $l(x, y) = \sum_{n=1}^{N} (y^n - \hat{y}^n)^2$
- Minimize the loss function l(x, y) using Gradient Descent algorithm. Implement (learn-linregGD and minimize-GD algorithms given in the lecture slides). Choose i_{max} between 100 to 1000.
- You can choose three suitable values of step length $\alpha > 0$. For each value of step length perform the learning and record
  - In each iteration of the minimize-GD algorithm calculate $|f(x_{i-1}) - f(x_i)|$ and at the end of learning, plot it against iteration number i. Explain the graph.
  - In each iteration step also calculate RMSE on test set $RMSE = \sqrt{\frac{\sum_{q=1}^{T} (y_{test}^q - \hat{y}^q)^2}{T}}$ and at the end of learning, plot it against iteration number i. Explain the graph.

Part B: (7 Points): Step Length for Gradient Descent

This task is based on Part A. You have to implement two algorithms *steplength − armijo* and *steplengthbolddriver* given in the lecture slides. For each step length Algorithm

- In each iteration of the minimize-GD algorithm calculate $|f(x_{i-1}) - f(x_i)|$and at the end of learning, plot it against iteration number i. Explain the graph.
- In each iteration step also calculate RMSE on test set $RMSE = \sqrt{\frac{\sum_{q=1}^{T} (y_{test}^q - \hat{y}^q)^2}{T}}$ and at the end of learning, plot it against iteration number i. Explain the graph.

Compare different step length algorithms Compare the RMSE graphs of *steplength− armijo* and *steplengthbolddriver* and the three fixed step length. Explain your graph.

1.3 ANNEX

- You can use numpy or scipy in build methods for doing linear algebra operations
- You can use pandas to read and processing data
- You can use matplotlib for plotting.
- 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.

[1] Data taken from University of Florida [2] Original Owners:

Forina, M. et al, PARVUS - An Extendible Package for Data Exploration, Classification and Correlation. Institute of Pharmaceutical and Food Analysis and Technologies, Via Brigata Salerno, 16147 Genoa, Italy.