İstanbul Aydın University

Software Engineering Department

SEN 437 Machine Learning

Homework Assignment 1

Introduction to Python, Scientific Python, Linear Algebra and Fundamentals of Machine Learning

Due Date: 8 November 2020, Sunday 23:59

This is an individual homework assignment. You can't get help from your friends and help them. Copying solutions will not be tolerated, if detected, "0" grade will be given to everyone in this misbehavior. You can also fail the course.

We will use some practical books and online tutorials for practical applications of Machine Learning and Neural Networks. The book given below is one of these practical books:

Hands-On Machine Learning with Scikit-Learn, **Keras and TensorFlow**, Aurélien Geron (2019)
This book has Python Jupyter Notebook examples which can be downloaded from the address:
https://github.com/ageron/handson-ml

This first assignment using the first 4 chapters of this book guides you to Python, Scientific Python and Math and gives some of the first important concepts in Machine Learning.

In the last year, the fundamentals of Machine Learning were intoduced to most of the students taking this course. They took my another course. This semester there are some new students who did not take my previous course. I hope that this first homework assignmet will address the needs of both student groups.

If you finish the first 4 chapters of our practical text book and do this assignment, you will remember these important concepts and also gain some practcal hands-on experience on Machine Learning.

Questions

Chapter 1

1. (20 points) **Introduction to Machine Learning**

a) What are the typical steps and activities when using Machine Learning in real world problems? b) Compare Linear Regression with K-NN (K-Nearest Neighbors) regression approaches in following aspects: models, training/testing phases, speed, accuracy, application areas.

c) Explain the concepts of Underfitting, Overfitting, and Regularization using an example.

d) Explain the figure and the models given below.

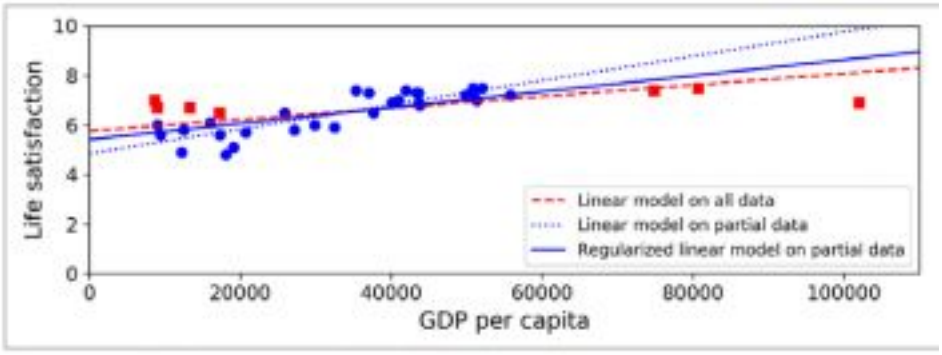


Figure 1-23. Regularization reduces the risk of overfitting

Chapter 2

2. (50 points) **Introduction to Data Preprocessing Techniques and Machine Learning Algorithms: Regression**

In this question, you will explore a **regression task**, **predicting housing values**, using various algorithms such as **K-NN**, **Linear Regression**, **Decision Trees**, and **Random Forests** as explained in chapter 1 and 2 using a different data set.

You will use the **Boston House Prices** dataset to do similar data preprocessing and machine learning analysis operations and produce a **Jupyter Notebook** given and explained in chapter 2. Your notebook will be similar to **02_end_to_end_machine_learning_project.ipynb**.

The Boston House Prices dataset has 13 input variables (home features) and 1 special attribute (or label) variable (house price).

See: **`sklearn.datasets.load_boston`**
https://ogrisel.github.io/scikit-learn.org/sklearn-tutorial/modules/generated/sklearn.datasets.load_boston.html

Boston House Prices data set:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33

Put the feature names to the column names

To get some insight you will perform several data preprocessing and visualization operations. Then the dataset will be divided into training and test sets by using a ratio of 80-20. A typical code example is given below. Then the model will trained and the model accuracy (error) on the test data will be calculated.

```
data = load_boston()                          #load dataset
X,Y = data["data"], data["target"]            #separate data into input and output features
#split data into train and test sets in 80-20 ratio
X_train,X_test,Y_train,Y_test = train_test_split(X, Y, test_size = 0.2)
```

Getting to Know Data and Preprocessing

a) (4 points) Explain the data set by learning it from the Internet sources. What are the 13 features in the data set (features), briefly explain them. How many training and test data samples are used in the program? First, plot the rows and data types of your dataset in the notebook.

b) (4 points) Write the training and test data sets in the program into two separate files (**boston_training.cvs** and **boston_test.cvs**) in CVS format. Also produce the file **boston.cvs** containing all data.

c) (4 points) Before applying your data to a Machine Learning algorithm, it is a good idea to get to know your data a little. Get the statistics of 13 columns in the dataset. That is: number of samples (count), mean (mean), standard deviation (std), min and max, and others (see example below).

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.593761	11.363636	11.136779	0.069170	0.554695	6.284634	68.574901	3.795043	9.549407	408.237154	18.455534
std	8.596783	23.322453	6.860353	0.253994	0.115878	0.702617	28.148861	2.105710	8.707259	168.537116	2.164946
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	2.900000	1.129600	1.000000	187.000000	12.600000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500	45.025000	2.100175	4.000000	279.000000	17.400000
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500	77.500000	3.207450	5.000000	330.000000	19.050000
75%	3.647423	12.500000	18.100000	0.000000	0.624000	6.623500	94.075000	5.188425	24.000000	666.000000	20.200000
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	22.000000

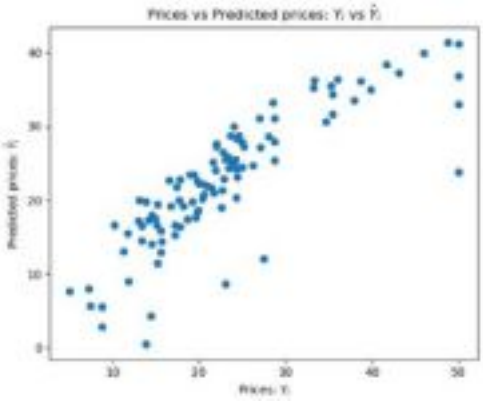
d) (4 points) Do other useful preprocessing and visualizations operations (such as plotting histograms of some attributes, correlation analysis, data cleaning, transformations, feature scaling, and so on, if necessary) similar to those given in chapter 2. Generate some graphs and briefly explain your operations and graphs in the Python notebook source file. Put proper headings into the notebook also.

e) (4 points) Find correlations: Loot at how much each attribute correlate with the Median House Value. You will choose 2 attributes having strong correlations with house value and produce 2 different models.

f) (4 points) Create 2 linear models for two attributes with Python using **`LinearRegression()`** and 80% of training data (X_train and Y_train). Then estimate house prices (**Y_pred**) for 20% of test data (X_test).

g) (4 points) Genera a scatter plot for test samples and your model line on the same graph for each model your have found.

h) (4 points) Generate a graph showing real prices (Y_test) on the X axis and estimated prices (**Y_pred**) on the Y axis and interpret this graph. You will obtain a graph similar to the one given below. Ideally, What should the scatter plot draw ideally? Explain your result and ideal case.



i) (4 points) Find coefficients and intercept points with python code, and write linear model equations in your report.

j) (4 points) Find the errors of your linear models with the MSE (Mean Squeare Error) method with Python code. What are the MSE values for both models you have found? Interpret the results. k) (10 points) Use K-NN, Decision Tree and Random Forest algorithms on the same data set, and compare the models and MSEs with the Linear Regression's results. Explain your findings.

Chapter 3

3. (20 points) **Classification**

Using some simple code segments, explain the following important concepts for classification:

a) What is binary classifier? How do you train a binary classifier?

b) How do you measure accuracy using cross-validation?

c) What is confusion matrix? Explain it using python code.

d) In what cases accuracy does not provide useful information about performance of a machine learning model? What is the problem? Give a specific example case.

e) What are precision and recall? What is the relation between these metrics? Explain using an example case. What is a ROC curve? Compare it with precision and recall curves.

Chapter 4

4. (10 points) **Training Models**

a) Explain the following: Batch Gradient Descent, Stochastic Gradient Descent, Mini-batch Gradient Descent

b) Explain the following 3 plots given in Figure 4-8 given below.

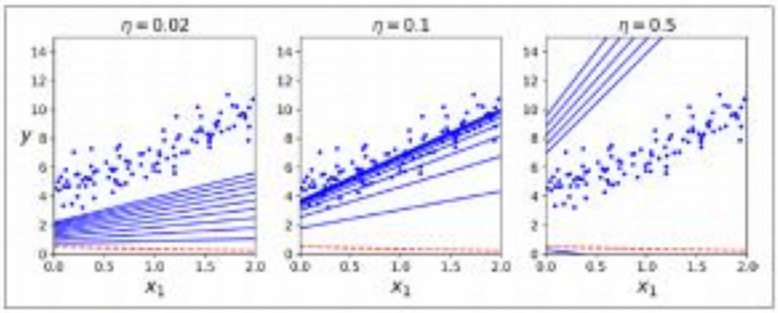


Figure 4-8. Gradient Descent with various learning rates

How to Submit the Assignment

Please follow the file naming conventions given below strictly. Otherwise, your assignment may not be graded fully. Because it is quite difficult to evaluate the files submitted in different formats.

You will upload **one zip file** to Google Classroom:

- HW-Assignment-1.zip
- HW1-Report.docx
- boston.zip

There will be one homework report **HW1-Report.docx** and one zip file (**boston.zip)** in the **HW Assignment-1.zip**.

boston.zip will have the following files:
boston.cvs
boston_training.cvs
boston_test.cvs
boston_pre-train.ipynb (main python program file)
- other data ve source files if necessary