

Linear Regression Project & Classification Tree Homework

New Attempt

Due Friday by 11:59pm Points 100 Submitting a file upload
Available Aug 30 at 12am - Dec 6 at 11:59pm

Dear Students,

TUTORIAL:

Please note the following for this Linear Regression assignment on how to run the Tutorial 5 where the Stock_Price_MAX data file is provided in a zip folder.

I also changed the code a bit so that it works! However, the data file contains only a sample of 5 records as a starter. So please add more random synthetic records before running.

For practice Tutorial-5 on Linear Regression, use Zip folder which contains the python and data files: Stock_Price_MAX.zip folder under:

Files --> LabHelp---> Labs/data

The data file contains only 5 records as a sample. You may synthetically generate about 5000 records before running the program!! Learn!

Run on Google Colab for this to work!

ASSIGNMENT:

(a) Regression: Using any of the data in previous assignment:

Intelligently choose any one of the data sets you used in your previous assignment and apply Linear Regression on the data. You had created Training and Test sets in the previous assignment by partitioning (splitting) your original data set. Apply Linear Regression on the Training set and test with the Test set.

If a previous dataset is not suitable for regression, you may use a different dataset of your own choice or add a new X or Y column of numerical data type as needed to your previous dataset creatively.

Use both Simple Linear Regression model and Multiple Regression Model to fit your data. For the simple linear regression model, select any two variables as x and y. Use the Test data to predict y for new x values. Make sure your Test data contains x values for within the Training range of x and outside the Training range of x (so you can test both interpolation and extrapolation and observe what happens).

Since you have two pairs of partitioned sets, do regression analysis separately on both pairs. Plot the data (and the line). Save the picture. Record your observations in a report.

Regression models may behave badly if correlations are not analyzed and if redundant correlated features (attributes, i.e. columns) are not dropped, or outliers are not analyzed and processed upfront. Also, if data is not standardized or normalized especially in the case of regularization. Understand the importance of standardizing or normalizing the data before performing your experiments. For those who like to dive one step deeper, you may refer to the two articles below on feature scaling for regularization.

[Feature scaling - Wikipedia](#) 

[6.3. Preprocessing data — scikit-learn 1.0.2 documentation](#) 

(b) Regression and Classification using a new data set provided in file:

Please use the following excel file from CANVAS:

Files --> LabHelp-->

Labs/data/Linear_Regression_data/Admission_Predict_Ver1.1_small_data_set_for_Linear_Regression.xlsx

There is also a description pdf file in the same directory. Also manually inspect and study the data. Make any random observations about the data. Now, do the same operations in (a) on this smaller data set. Use only one pair of training and test partitions of the original data.

Record your observations in a report. Plot the data (and the line). Save the picture.

Regression models may behave badly if correlations are not analyzed and if redundant correlated features (attributes, i.e. columns) are not dropped, or outliers are not analyzed and processed upfront. Also, if data is not standardized or normalized especially in the case of regularization. Understand the importance of standardizing or normalizing the data before performing your experiments. For those who like to dive one step deeper, you may refer to these two articles on feature scaling for regularization.

[Feature scaling - Wikipedia](#)

[6.3. Preprocessing data — scikit-learn 1.0.2 documentation](#)

I

Classification:

Discretize the last column "Chance of Admit" into three classes and create a classification tree with training data. Test the tree with test data and evaluate the results in Python.

Describe a few rules (3 to 5 valuable rules are sufficient). Which rules do you think are the most valuable?

You may combine both the reports into one and submit the Jupyter notebook file and the report.

(c) Classification Tree Homework:

View the Files-->Homework sheets-->Entropy_ID3_Exercise.pdf

and workout the exercise.

(i) create the solution.

(ii) understand what impact may happen to your created tree, if you later add a new missing attribute after creating the tree? You may add any new attribute, for example, "pattern of shirt". The values may be "checked", "striped", and one or two more. Be creative, add your own new favorite attribute or keep "pattern of shirt". The class column remains the same. What are some of the different possible changes you may expect to see on the classification decision tree you just created? Add this analysis to your solution document and submit. What if a data scientist provided his or her results with high confidence, by missing this attribute altogether? What if his or her results are used for decision making on how many million more shirts to produce for the next year? Do you think the data scientist surprises and makes an impact on the manager and CEO in case he or she discovers the new attribute and it's influence in getting more reliable results valuable to the company?

All of (a), (b) and (c) are team projects. Only one member submit. In your submission comment state the name of your team. Provide details of each team member's contribution. Please submit a PDF file only for your report.

(a) --> 30%

(b) ---> 20% regression 20 % classification

(c) ---> 30%

If you are inclined, here are links to additional reading material. Open the links and read only the first paragraph to get an idea what the article is about and see if you want to read further for any kind of deeper understanding. Also you may refer to articles in the Linear Regression section under Modules.

1. coding

<https://realpython.com/linear-regression-in-python/>

2. fitting and plotting

<https://stackoverflow.com/questions/50280694/how-to-fit-and-plot-a-linear-regression-line-in-python> ➞

<https://datascience.stackexchange.com/questions/27740/plotting-multivariate-linear-regression> ➞

3. preprocessing (normalizing and such)

<https://scikit-learn.org/stable/modules/preprocessing.html> ➞

4. lasso and ridge regression (using new loss functions)

https://www.analyticsvidhya.com/blog/2015/10/regression-python-beginners/?utm_source=blog&utm_medium=RideandLassoRegressionarticle ➞

<https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-ridge-lasso-regression-python/> ➞

5. types of variables

<https://statistics.laerd.com/statistical-guides/types-of-variable.php> ➞

6. classification trees

<https://www.datacamp.com/community/tutorials/decision-tree-classification-python> ➞

cheers,

;)

Jagan