CS 6220 Data Mining — Assignment 6

Regression

This assignment will require you to implement and interpret some of the regression concepts that were introduced in this module. Keep in mind that the main objective of this assignment is to highlight the insights that we can derive from applying these techniques—the coding aspect is secondary. Accordingly, you are welcome to consult any online documentation and/or code that has been posted to the course website, so long as all references and sources are properly cited. You are also encouraged to use code libraries, so long as you acknowledge any source code that was not written by you by mentioning the original author(s) directly in your source code (comment or header).
Using the Auto MPG Data Set, you will be asked to construct a linear regression using different features of the dataset to predict the target. You will then be asked to evaluate the performance of the regression with visual outputs.

Objectives:

1. Apply linear regression to a dataset containing numerical features
2. Evaluate the performance of linear regression using R-squared metrics and Mean Squared Error

Submission:

Through the assignment submission portal on Canvas, submit your ipynb with a pdf of your assignment solution; no need to zip the files.

Grading Criteria:
Follow the instructions in the pdf, and complete each task. You will be graded on the application of the modules' topics, the completeness of your answers to the questions in the assignment notebook, and the clarity of your writing and code.

Assignment Description

The Data The Auto MPG dataset contains information on various car models and their attributes. It is commonly used in regression analysis to predict the fuel efficiency (miles per gallon, or MPG) of a car based on its characteristics. The dataset includes both numerical and categorical attributes.

Each instance in the dataset represents a particular car model, and the attributes provide information about the car's specifications. You can know more about the dataset from here.

The dataset can be downloaded using
`https://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data`

The Idea: Linear Regression on the Auto MPG dataset

Here, we want to predict the fuel efficiency (MPG) of a car based on its other attributes. It allows us to explore relationships between the car's specifications and its fuel efficiency, and to build predictive models using regression techniques. In this assignment, we will predict the target feature using all numerical features ('Cylinders', 'Displacement', 'Horsepower', 'Weight', 'Acceleration', 'Model Year', 'Origin'). And then we will use single features to predict the target feature.

What to Do

First, download the Auto MPG dataset from here `https://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data`

You can load this dataset into a dataframe using the pandas library, then clean the data to remove any missing values.

To generate a linear regression model, you may use the *linear_ model.LinearRegression()* function available via the scikit-learn library. To run the model on the Auto MPG data, first divide the dataset into training and testing sets, then fit the model on the training set and predict with the fitted model on the testing set. scikit-learn provides several functions for dividing datasets in this manner, including *cross_ validation.KFold* and *cross_ validation.train_ test_ split.*

Several statistics can be generated from a linear model. Given a fitted linear model, the following code outputs the model coefficients (the parameter values for the fitted model), the residual sum of squares (the model error), and the explained variance (the degree to which the model explains the variation present in the data):

```
# The coefficients
print('Coefficients:', regr.coef )
# The mean squared error
print(’’Mean squared error:  %.2f’’ % np.mean((coly_pred  coly_test )  2))
# Explained variance score :  1 is perfect prediction
print('Variance score:  %.2f' % regr.score(colx_test , coly_test)

# The coefficients
print('Coefficients:', regr.coef )
```

```
# The mean squared error
print(’’Mean squared error:  %.2f’’ % np.mean((coly_pred  coly_test )  2))
# Explained variance score :  1 is perfect prediction
print(’Variance score:  %.2f’ % regr.score(colx_test , coly_test)
```

You can use these scores to measure the efficacy of a particular linear model.

What to do

Step 1 Split the dataset into training and test sets (80, 20).
Step 2(a) Use all the features (1-7) to fit the linear regression model for feature 8(MPG) using the training set.
Step 2(b) Report the coefficients, mean squared error and variance score for the model on the test set.
Step 3(a) Use each feature alone - to fit a linear regression model on the training set.
Step 3(b) Report the coefficient, mean squared error and variance score for the model on the test set. Also report the 7 plots of the linear regression models generated on each feature. Each plot should distinctly show the training points, test points and the linear regression line.
Step 4(a) Perform 10 iterations of (Step 1, Step 2(a), and Step 3(a)).
Step 4(b)

- During each iteration of Step4(a), gather the metrics - mean squared error and variance score for the 8 models on the test set (In 8th model all the features are used).
- For each feature, compute the average, over the 10 iterations, of each evaluation metric. Do the same for the metrics corresponding to ’all features’.
- To compare the model performance, provide the following plots
  1. mean square error vs features
  2. variance score vs features
- In the above-mentioned two plots, make sure to designate a point on the features axis for ’all 7 features’ so you can include the metrics corresponding to the models generated in the 10 iterations of Step 2(a). E.g., You may designate it as feature 0.

What to Provide

Your output should contain the following:

- Report generated in Step 2(b)
- 7 sets of metrics and plots generated with the regression line in Step 3(b)
- Two plots generated in Step 4(b)

- Given this output, respond to the following questions:

  1. Based upon the linear models you generated, which feature appears to be most predictive for the target feature? Note that you can answer this question based upon the output provided for the linear models.
  2. Suppose you need to select two features for a linear regression model to predict the target feature. Which two features would you select? Why?
  3. Examine all the plots and numbers you have, do you have any comments on them? Do you find any surprising trends? Do you have any idea about what might be causing this surprising trend in the data? This is a descriptive question meant to encourage you to interpret your results and express yourself.