IT 170 Programming Assignment 3

Answers to this assignment are due by ***the time specified on Reggienet***. You need submit your solution as required below to the ReggieNet.

No late submission!!!

Simple Linear Regression (35%)

In this programming assignment, you will implement a Simple Linear Regression model from scratch to analyze a dataset from a CSV file.

Objectives

- Be able to design and develop a solution to a relatively complicated problem (i.e., simple linear regression model)
- Be able to evaluate the performance of the predictive model
- Be able to apply various programming skills (e.g., for-loop, function, list, list comprehensions) to solve different problems

---

Background:

Linear regression

Linear regression assumes a linear or straight line relationship between the input variables (X) and the single output variable (y).

More specifically, that output (y) can be calculated from a linear combination of the input variables (X). When there is a *single input variable*, the method is referred to as a **Simple Linear Regression**.

In simple linear regression we can use statistics on the training data to estimate the coefficients required by the model to make predictions on new data.

The line for a simple linear regression model can be written as:

$y = b_0 + b_1 \times x$

where b_0 and b_1 are the coefficients we must estimate from the training data.

Once the coefficients are known, we can use this equation to estimate output values for y given new input examples of x. It requires that you calculate statistical properties from the data such as mean, variance and covariance. All the algebra has been taken care of and we are left with some arithmetic to implement to estimate the simple linear regression coefficients.

Briefly, we can estimate the coefficients as follows:

$$\begin{aligned}
b_1 = & \sum((x_i - \bar{x}) \times (y_i - \bar{y})) / \sum((x_i - \bar{x})^2) \\
= & \ Covariance(x, y) / Variance(x) \\
b_0 = & \ \bar{y} - b_1 \times \bar{x}
\end{aligned}$$

where i refers to the value of the i^{th} value of the input x or output y. $\bar{x}$ and $\bar{y}$ are the mean value of the input x or output y.

Please note, $\sum$ is a simplified version of $\sum_i^N$, where N is the total number of samples in a dataset.

Covariance & Variance

The covariance of two groups of numbers describes how those numbers change together.

Covariance is a generalization of correlation. Correlation describes the relationship between two groups of numbers, whereas covariance can describe the relationship between two or more groups of numbers.

We can calculate the covariance between two variables as follows:

$covariance = \sum((x_i - \bar{x}) \times (y_i - \bar{y}))/(n-1)$

$variance = \sum((x_i - \bar{x})^2)/(n-1)$

The calculation of covariance between x and y, and the variance of x is the estimation based on the samples of x and y.

Performance Evaluation

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit (i.e., the regression model). Root mean square error is commonly used in forecasting, and regression analysis to verify experimental results.

$$RMSE = \sqrt{\frac{\sum(y_i^{predicted} - y_i^{actual})^2}{n}}$$

Another commonly used performance metric is the Mean Absolute Error(MAE), which is the average of all absolute errors. The formula is:

$$MAE = \frac{\sum|y_i^{predicted} - y_i^{actual}|}{n}$$

Dataset

In this assignment, the data is saved in a CSV (Comma Separated Values) file called salary.csv.

The file is essentially a plaintext file, which contains 30 lines of information. Each line consists of two numbers (i.e., years of experience and salary) separated by a comma, such as "1.1,39343". Assuming an employee's salary (y) is approximately linearly increased with the number of years of working experience (x). Our task is to build a Simple Linear Regression (i.e., $y = b_0 + b_1 \times x$).

Here are the steps you can follow:

1. (5 points) Ceate a p3Lib.py file (module) and copy all previously developed relevant functions, including but not limit to: sumList; meanList; minMaxList, stdevList. For the new functions (e.g., varianceList, covarianceList) developed in this assignment, you can either include them to p3Lib.py, or simply include them in the main Python module: p3.py
2. (8 points) Develop a function called **train_test_split** to split the dataset (List) into a training set and a test set based on a ratio (e.g., 75% for training and 25% for testing).

```
# Hint: split "aList" into trainSet and testSet based on "ratio"
import random
def train_test_split(aList, ratio):

    Your Code Here

    return trainSet, testSet
```

3. (8 points) Develop two functions called **covarianceList** and **varianceList** to calculate Covariance and Variance of the training dataset. Develop another function called **coefficients** to find out coefficients (i.e., b_0, b_1)

```
# Hint: Calculate the variance of a list of numbers
def varianceList(aList):

    Your Code Here

    return variance


# Hint: Calculate the covariance of two lists of numbers
# The two lists should have the same dimension.
def covarianceList(aList, bList):


    Your Code Here
    return covariance


# Hint: Calculate the coefficients of the trainSet
# The input is a list, in which each item is also a list containg
# a sample [x_i, y_i]
def coefficients(trainingDatasetList):

    Your Code Here

    return b0, b1
```

4. (8 points) Develop a function to calculate RMSE used for evaluate the model performance.

```
# Hint: Evaluate the model performance using RMSE
# testResultList includes a series of test results
# E.g., [[y0_predict, y0_actual], [y1_predict, y1_actual], ...]
def evalRMSE(testResultList):

Your Code Here

return rmse
```

5. (6 points) Develop a Python application to

   a). open/read the CSV file to a List;

   b). split the list into trainSet and testSet;

   c). calculate the coefficients using trainset;

   d). using the model based on the coefficients, predict y (i.e., salary) for each x (i.e., years) in the testSet;

   e). evaluate the model performance using RMSE. Print the result of b_0, b_1, and RMSE.

Hint: each item in the List from CSV is also a List containing one pair of "years" and "salary". So after successfully loading the dataset, it will be saved to a List as [["1.1", "39343"], ["1.3", "46205"], ...]. This list is similar to a $[2 \times 30]$ array, representing 30 samples.

Please note: You still cannot directly use built-in functions max(); min(); sum(); and mean() (or **similar ones**) from statistics module.

● ✕