Data Analytics

CS/STAT 5525

LAB # 4

In this Homework, you will learn practice simple & multiple linear regression analysis for prediction. You will also practice polynomial regression for prediction. Feature selection using PCA & random forest technique will also discussed in this assignment.

The dataset that is selected for this assignment is 'Carseat.csv' which can be found on the course GitHub. All numbers must be displayed with 3-digit decimal precision. Add x-label, y-label, grid, title, and legend [if applicable] to all submitted plots. Us the 'seaborn-whitegrid' style for all the matplotlib plots.

The 'Carseat.csv' dataset contains several features (qualitative and quantitative) and the dependent variable 'Sales' (child car seat) in 400 locations. The goal is to predict the sales using regression model. Use random_state = 1 as needed.

1. EDA: Develop a python program that:[15 points]

    a. Read the dataset and plot the group horizontal bar plot that shows the 'ShelveLoc' versus Sales with respect to the 'US' (legend). Which Shelve location has the highest sales and whether the best sales are inside the US or outside? Hint: You need to aggregate using summation.

    b. Using the one-hot-encoding converts the qualitative features to quantitative feature and make it ready for regression analysis (zeros or ones). Display the first 5 rows of the converted features on the console.

    c. Standardize the dataset in the previous step and split the dataset into train-test 80-20. Turn the shuffle ON and use the random_state = 1. Display the first 5 rows of the train and test set. Hint: use

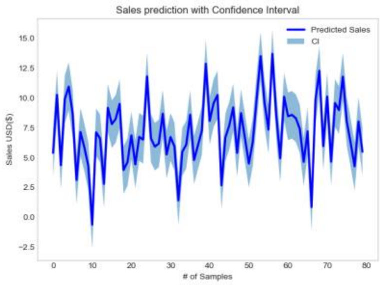
    ```
    from sklearn.model_selection import train_test_split
    ```

2. Feature selection & Prediction: Develop a python program that performs a backward stepwise regression analysis and eliminate features using p-value of t-test (threshold 0.05). Hint: Drop one feature at a time while monitoring Adjusted R-squared. [20 points]
    a. Display the eliminated and final selected features on the console.
    b. Drop the insignificant features and perform a regression analysis OLS on the selected features. Take a screen shot of the OLS summery and place it here.
    c. Make a prediction on sales and compare it with the test set and plot the original test set (sales without transformation) versus the predicted values (not a scatter plot). Hint: You need to perform a reverse transformation.
    d. Display the Mean Squared Error (MSE) on this prediction on the console.

3. Feature selection & Prediction: Develop a python program that performs a Principal Component Analysis (PCA). [15 points]
    a. How many features are needed that explains more than 90% of the dependent variance.
    b. Plot the cumulative explained variance versus the number of features. Hint: The number of features lie on the x-axis and must start off at 1.
    c. Does PCA tell you which feature(s) need to be eliminated? Explain your answer.
4. Feature selection & Prediction: Develop a python program that performs a Random Forest Analysis and eliminate features based on the importance. [25 points]
    a. Plot the horizontal bar graph (once random forest analysis is completed) that displays the features importance in the descending order. Hint: Make sure that the y-axis labeled according to the feature labels.
    b. Display the eliminated and final selected features on the console. Does the selected features base on random forest and step-wise regression method identical? Explain your answer.
    c. Drop the insignificant features (pick a threshold) and perform a regression analysis OLS on the selected features. Take a screen shot of the OLS summery and place it here.
    d. Make a prediction using the test set and plot the original test set (sales without transformation) versus the predicted values (not a scatter plot) . Hint: You need to perform a reverse transformation.
    e. Display the Mean Squared Error (MSE) on this prediction on the console.

5. Create a table that compares the R-squared, Adjusted R-squared, AIC, BIC and MSE of the prediction in step 2 and 4. Which method of feature selection will you recommend for this problem and what features do you recommend for elimination? Explain your answer. [10 points]

6. Prediction Interval: Based on the answer in the previous question and the selected features, find the prediction intervals for the regression model. Plot the predicted sales value (need to perform reverse transformation), the lower and upper prediction intervals (95%) in one graph. Hint: You can use .get_prediction() function. The result is shown below. [10 points]

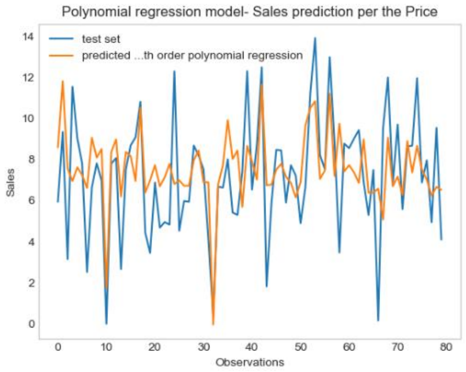


7. Polynomial regression and Grid Search: Let suppose we want to find a polynomial regression model that regresses Sales (dependent variable) versus the Price feature only. Hint: You may use the following packages: [25 points]

```
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.pipeline import make_pipeline
```

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \cdots + \beta_n x^n + \varepsilon.$$

a. Perform a grid search that minimized RMSE.
b. What is the optimum order for n?
c. Plot the RMSE versus the n order. The search intervals [1,15].
d. Split the dataset into train-test 80-20 random_state =1. Train the regression model using OLS and the optimum nth order derived from the previous section. Plot the test set (sales) versus the predicted sales values.
e. What is the MSE of this nth other polynomial regression prediction?



8. Bonus Question: In a simple linear regression with n observations [10 points]

$$y_i \approx \hat{\beta}_0 + \hat{\beta}_1 x_i$$

Prove the following:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}_1 \overline{x}$$

where $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $\overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ are the sample mean.

Upload the solution as a **report (as a single pdf)** plus **the .py file** through Canvas by the due date.