Exercise 4 : Linear Regression

Workflow

1. Create a folder on your Desktop and name it Cx1015_[LabGroup], where [LabGroup] is the name of your Group
2. Download the .ipynb files and data files posted corresponding to this exercise and store in the aforesaid folder
3. Open Jupyter Notebook (already installed on the Lab computer) and navigate to the aforesaid folder on Desktop
4. Open and explore the .ipynb files (notebooks) that you downloaded, and go through "Preparation", as follows
5. The walk-through videos posted on NTU Learn (under Course Content) may help you with this "Preparation" too
6. Create a new Jupyter Notebook, name it Exercise4_solution.ipynb, and save it in the same folder on the Desktop
7. Solve the "Problems" posted below by writing code, and corresponding comments, in Exercise4_solution.ipynb

Try to solve the problems on your own. Take help and hints from the "Preparation" codes and the walk-through videos. **If you are still stuck, talk to your friends in the Lab to get help/hints.** If that fails too, approach the Lab Instructor.

Note : Don't forget to import the Essential Python Libraries required for solving the Exercise. Write code in the usual "Code" cells, and notes/comments in "Markdown" cells of the Notebook. Check the preparation notebooks for guidance.

Preparation

M3 LinearRegression.ipynb Check how to perform Linear Regression on the Pokemon data (pokemonData.csv)

Objective

In the last Example Class, we have identified and analyzed some of the most relevant numeric variables in this dataset, which may affect the sale price of a house, and hence, will probably be most relevant in predicting "SalePrice". In this Example Class, we will extract those numeric variables one-by-one and perform Linear Regression to predict "SalePrice".

Problems

Download the dataset **train.csv** and the associated text file **data_description.txt** posted with this Exercise.

Problem 1 : Predicting SalePrice using GrLivArea

Import the complete dataset "train.csv" in Jupyter, as `houseData = pd.read_csv('train.csv')`

Extract the following Numeric variables from the dataset, and store as two new Pandas DataFrames.

```
houseGrLivArea = pd.DataFrame(houseData['GrLivArea'])        Above ground living area in SqFt
houseSalePrice = pd.DataFrame(houseData['SalePrice'])        Sale Price of house in US Dollars
```

a) Plot `houseSalePrice` against `houseGrLivArea` using standard jointplot, and note the strong linear relationship. Remember the correlation coefficient between these two variables from the last Example Class? Check again.

b) Import Linear Regression model from Scikit-Learn : `from sklearn.linear_model import LinearRegression`

c) Partition both datasets `houseGrLivArea` and `houseSalePrice` into Train (1100 rows) and Test (360 rows) sets.
Train datasets : `houseGrLivArea_train` and `houseSalePrice_train` (check both have 1100 rows)
Test datasets : `houseGrLivArea_test` and `houseSalePrice_test` (check both have 360 rows)

d) Training : Fit a Linear Regression model with `X = houseGrLivArea_train` and `y = houseSalePrice_train`

e) Print the coefficients of the Linear Regression model you just fit, and plot the Regression line on a Scatterplot of `houseGrLivArea_train` and `houseSalePrice_train` using the standard slope-intercept form of straight line.

f) Predict SalePrice for the test dataset `houseGrLivArea_test` using the Linear Regression model, and plot the predictions on the Scatterplot of `houseGrLivArea_test` and `houseSalePrice_test` to visualize the accuracy.

g) Find the Explained Variance (R^2) of the model on the Train set and on the Test set to check Goodness of Fit.

Problem 2 : Predicting SalePrice using Other Variables

Perform all the above steps on "SalePrice" against each of the variables "LotArea", "TotalBsmtSF", "GarageArea" one-by-one to perform individual Linear Regressions. Discuss with your Friends about the models, compare and contrast the Explained Variance (R^2), check the predictions, and determine which model is the best to predict "SalePrice".

Extra Resources

You may read more about the LinearRegression() model you will use in this exercise in the following references.

LinearRegression : https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html

Other Linear Models (Scikit Learn) : https://scikit-learn.org/stable/modules/classes.html#module-sklearn.linear_model

Bonus Problems

1. Note that LinearRegression() model can take more than one Predictor to model the Response variable. Try using this feature to fit a Linear Regression model to predict "SalePrice" using all the four variables "GrLivArea", "LotArea", "TotalBsmtSF", and "GarageArea". Find the Explained Variance (R^2) of this multi-variate model.

2. Fit a Linear Regression model to predict "SalePrice" using all the numeric variables in the given dataset. You may use all the numeric variables from Exercise 2. Find the Explained Variance (R^2) of this multi-variate model.

Is the Explained Variance (R^2) of a multi-variate model equal to the Sum of the Explained Variances (R^2) of the component univariate models? For example, if R^2 for "SalePrice" vs "GrLivArea" is 0.53 and R^2 of "SalePrice" vs "LotArea" is 0.22, will the R^2 for "SalePrice" vs ["GrLivArea", "LotArea"] be 0.75? Experiment, and think about it.