IC 272: Lab8: Prediction using Linear and Polynomial Regression

A dataset related to red variants of the Portuguese "Vinho Verde" wine is given as a csv file (`winequality-red.csv`). This dataset contains the values of different physicochemical tests from each samples of red wine [1]. The original goal of the dataset is to model wine quality based on physicochemical tests. The attributes of the dataset based on physicochemical tests are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH value, sulphates, alcohol content and last attribute is on quality. Each expert graded the wine quality between 0 (very bad) and 10 (very excellent).

1. Split the data from `winequality-red.csv` into **train data** and **test data**. Train data contain **70%** of tuples and test data contain remaining **30%** of tuples. Save the train data as `winequality-train.csv` and save the test data as `winequality-test.csv`
2. Build the simple linear regression (**straight-line regression**) model to predict wine quality given pH value.
   a. Plot the best fit line on the training data where x-axis is pH value and y-axis is quality
   b. Find the **prediction accuracy** on the training data using *root mean squared error*.
   c. Find the **prediction accuracy** on the test data using *root mean squared error*.
   d. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data. Comment on the scatter plot.
3. Build the simple nonlinear regression model using **polynomial curve fitting** to predict wine quality given pH value.
   a. Plot the best fit curve on the training data where x-axis is pH value and y-axis is quality.
   b. Find the **prediction accuracy** on the training data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).
   c. Find the **prediction accuracy** on the test data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).
   d. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data for the best degree of polynomial (p). Comment on the scatter plot and compare with that of in 2(d).
4. Build the multiple linear regression model to predict wine quality.
   a. Find the **prediction accuracy** on the training data using *root mean squared error*.
   b. Find the **prediction accuracy** on the test data using *root mean squared error*.
   c. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data. Comment on the scatter plot and compare with that of in 2(d) and 3 (d).
5. Build the multivariate polynomial regression model to predict wine quality.
   a. Find the **prediction accuracy** on the training data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).
   b. Find the **prediction accuracy** on the test data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).

c. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data. Comment on the scatter plot and compare with that of in 2(d), 3 (d) and 4(c).

6. Compute the Pearson correlation coefficient for every attribute with the attribute *quality* (dependent variable) on the training data. Select two attributes that are highly correlated (either positively or negatively correlated) with attribute *quality*.
   a. Build the multiple linear regression model considering only the selected two attributes to predict wine quality.
      i. Plot the best fit plane on the training data where x and y-axis are the two selected attributes z-axis is quality.
      ii. Find the **prediction accuracy** on the training data using *root mean squared error*.
      iii. Find the **prediction accuracy** on the test data using *root mean squared error*.
      iv. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data. Comment on the scatter plot.
   b. Build the multivariate polynomial regression model considering only the selected two attributes to predict wine quality.
      i. Plot the best fit surface on the training data where x and y-axis are the two selected attributes z-axis is quality.
      ii. Find the **prediction accuracy** on the training data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).
      iii. Find the **prediction accuracy** on the test data for the different values of degree of polynomial (p = 2, 3, 4, 5) using *root mean squared error (RMSE)*. Plot the bar graph of *RMSE* (y-axis) vs different values of degree of polynomial (x-axis).
      iv. Plot the scatter plot of *actual quality* vs *predicted quality* on the test data. Comment on the scatter plot.
7. Compare each of the regression models (all the cases from questions 2-6) based on *RMSE*.

Note:

A. **Simple and Multiple Linear Regression**:
Import the `LinearRegression` from `sklearn.linear_model`
A code snippet for prediction using linear regression:

```
regressor = LinearRegression()
regressor.fit(x, y)
```

`x` is set of univariate or multivariate training data used for building simple of multiple linear regression. `y` is corresponding dependent variable.

```
y_pred = regressor.predict(x)
```

B. **Polynomial Curve Fitting and Polynomial Regression**:
Import the `PolynomialFeatures` from `sklearn.preprocessing`
A code snippet for prediction using linear regression:

```
polynomial_features= PolynomialFeatures(degree=p)
x_poly = polynomial_features.fit_transform(x)
```

`x` is set of univariate or multivariate training data used for building simple of multiple polynomial regression.

```
regressor = LinearRegression()
regressor.fit(x_poly, y)
```

`x_poly` is set of polynomial expansions (monomials of polynomial up to degree `p`) training data used for building simple of multiple linear regression. `y` is corresponding dependent variable.

```
y_pred = regressor.predict(x)
```

Reference:

[1] P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. "*Modeling wine preferences by data mining from physicochemical properties*," In Decision Support Systems, Elsevier, vol. 47, issue 4, pp. 547-553, 2009.