E5 - ProgDS-2025-10-13 : Regression – Multiple Methods

This exercise is aimed at:

1. Getting introduced to and running various regression algorithms on a given data set and understanding their relative characteristics, performance, and advantages.
2. Calculating, documenting, and understanding various regression metrics and developing an approach towards using them.
3. Creating and consolidating multiple plots and Tables with the aim of comparing and contrasting the results of the regression algorithms.
4. Getting introduced to the relevant ML functions of the Python library: **sklearn**.

Tasks:

1) Review the Jupyter Notebook `multiple-regression-methods.ipynb`, along with the data file `multiple-regression-methods.xlsx`:
   a) Execute the code. Analyze the resulting plots and metrics.
   b) Choose the best algorithm for the given dataset. Justify your choice.
   c) What is your take-away from the way the code has been structured to generate the results?

2) Review **sklearn** documentation for each sklearn function used in the Notebook (eg. PolyNomialFeatures, LinearRegression, mean_squared_error, etc.) and create a summary of each to explain the functionality, the input parameters, and the outputs. Present this in the form of a two-column Table (Function name | Description).

3) You have been given six algorithms to study: Tree based (Random Forest and XGBoost), non-parametric (KNN) and parametric (Linear Regression, SVR, and Neural Network).
   a) Run the models by setting the degree of PolynomialFeatures() to 1, 6 and 10.
   b) Analyze the plots, metrics and state your conclusions.
   c) Have any of these models ***overfitted*** the training data? Justify your conclusions.
   d) Based on the plots, comment on *which ML algorithms seem to be affected the most by the polynomial degree, based on visual inspection? Why?*
      i) Are the train RMSE values in agreement with your observations and conclusions?

Standardization: This is a preprocessing step that rescales features to have zero mean and unit variance, ensuring that each feature contributes equally to the model:

$$\boxed{z = \frac{x_i - \mu}{\sigma}}$$

- In the above expression, μ is the mean and σ is the standard deviation of the feature.

---

4) Set the polynomial *degree* to 10 and tabulate the RMSEs (both train and test) of the six models, **with and without standardization**.
   a) Which models are affected the most, and which are not?
   b) Why?

5) Set the degree to 6 and **repeat the following before and after adding the outliers** (uncomment the outlier related code):
   a) Which models seem unaffected by outliers? Can you explain why?
   b) What does this tell you about the choice of models in situations where noisy data is expected?

6) Set the polynomial degree to 1, apply standardization, and run the Neural Network with 1, 2 and 3 hidden layers (each layer having 10 neurons).
   a) Analyze the plots, metrics and state your conclusions.

7) What are the advantages and limitations of the non-parametric methods?

8) Given the results, should *LinearRegression* be used at all, or would one of Random Forest / KNN be a better choice? Why, when? Justify your answer.

9) List your major learnings from this exercise.

oooOOOooo