School of Computer Science and Engineering, VIT Chennai.

BCSE209P Machine Learning

Lab-4 Revisiting Linear and Logistic Regression

Faculty : Dr. R. Jothi

**Due Date : 18/01/2024**

Submit your python code (Jupyter notebook): with output for all the questions.

Q1. Suppose you are asked to build a machine learning model for predicting the output of manufacturing machine based on its operational time as given in Machine.csv dataset file.

* Print all the predictor variables and dependent variables in the given dataset (Machine.csv)
* Understand the relationship between each predictor variable and the dependent variable; draw the plot.
* Implement (write your code for Gradient Descent algorithm and weight updation) linear regression algorithm with gradient descent optimization. Print the regression parameters after 2 epochs of training. Predict the output of the manufacturing if the machine runs for 13 hours. Also print the error of your regression model. (you may verify your answers with Scikit Linear regression function with number of epochs as 2)

Q2. Observe the collinearity/multi-collinearity between the predictor variables form the house price prediction dataset. (hint : use heatmap)

Q3. Use Scikit Library function **LogisticRegression** for IRIS species classification and heart disease prediction. Explore all the parameters as mentioned below.

***class*sklearn.linear\_model.LogisticRegression(*penalty****='l2'***,***\****, *dual****=False***, *tol****=0.0001***, *C****=1.0***, *fit\_intercept****=True***, *intercept\_scaling****=1***, *class\_weight****=None***, *random\_state****=None***, *solver****='lbfgs'***, *max\_iter****=100***, *multi\_class****='deprecated'***, *verbose****=0***, *warm\_start****=False***, *n\_jobs****=None***, *l1\_ratio****=None***)**

<https://scikit-learn.org/0.16/modules/generated/sklearn.linear_model.LogisticRegression.html>

For both the above classification problems observe the following results.

1. Whether accuracy changes if optimization algorithm to find the best-fit parameters that minimize the loss function is changed (hint : change solver)
2. Change number of epochs and verify your results.
3. Change in the order of input samples for training and observe (random\_state)