LAB WEEK 9 – GROUP 6

LINEAR REGRESSION

We fit a linear regression model using SkLearn. To measure the performance of the model we obtained the value of the Mean Square error of the fitted model. After fitting the model, the obtained the following theta (i.e., parameters of the regression):

```
array ([ 9.50440237e-02, 5.58065384e-02, -1.50850562e-01, 9.50440237e-02,
         5.58065384e-02, -1.50850562e-01, -3.89290828e+00, 4.14375416e+00,
        -8.12769890e-04, -2.41076136e-04, -1.03205112e-02, -5.42776786e-02,
         3.70283232e-01, -7.65295267e-02, -1.31274030e-02, 6.01862017e-03]))
```

The above values give us an idea of how much weight each feature carried in predicting the values of PE. The intercept of the model tells us the initial value of PE when all the features are equal to zero.

```
Intercept = 0.719443020714637
```

MEAN SQUARE ERROR = `0.2641669510659059`

As our main performance metric, the mean squared error tells us how far off our model is from the perfect model. The closer the value is to zero, the better the model.

LOGISTIC REGRESSION

The logistic regression model has the following coefficients.

```
array ([[-9.76426484e-02, 3.35899817e-01, -1.38305001e-01,
         -9.76426484e-02, 3.35899817e-01, -1.38305001e-01,
          7.86372402e-02, 1.01220761e-01, 4.06910129e-03,
         -3.09947014e-04, -1.35754383e-01, -2.44010009e-01,
         -2.81839549e-02, -8.17804511e-02, -5.88051259e-02,
          4.03803683e-02]])
```

Unlike the linear regression model, we measure the performance of the logistic regression using the usual performance metrics, ROC curves, accuracy, precision, and confusion matrix. Below shows the accuracy and precision of the logistic regression model.

```
Accuracy: 67.6056338028169%

Precision: 64.1025641025641%
```

EXPERIMENTING WITH SDG REGRESSOR

We then experimented with various regression parameters that that control learning. To do this we used SDG regressor as it has all the parameters, we were interested in experimenting with. *Stochastic Gradient Descent, SDG* picks a random instance in the training set at every step and computes the gradients based only on that single instance. Obviously, this makes the algorithm much faster since it has very little data to manipulate at every iteration. The fowling table shows the different results obtained from tunning the parameters of the SDG regressor.

Parameters	Intercept	Coefficients	Mean Squared Error
Number of iterations:50 Learning Rate(eta): 0.1 Penalty: none	2.7664277e+11	-1.64935440e+11, 6.28665754e+11, -1.87087544e+11, -1.64935440e+11, 6.28665754e+11, -1.87087544e+11, 9.68162191e+10, 1.02451534e+11, -4.12443713e+11, 4.68797324e+12, -1.68057516e+11, 1.73690107e+12, -1.63949366e+11, -1.96577820e+11, -2.83261392e+12, 6.44851091e+11	1.1892380805728238e+28
Number of iterations: 1000 Learning Rate(eta): 0.1 Penalty: none	-2.2673416e+10	9.54079465e+11, -1.5495077e+12, 5.72754860e+11, 9.5407947e+11, -1.54950774e+12, 5.7275486e+11, 5.74995630e+10, 3.93296375e+10, 1.11510561e+12, -1.5742650e+12, 1.18916595e+12, 8.57850415e+11, -1.42932036e+11, -3.500912e+11, 1.84379014e+12, -1.02865418e+12	1.1892380805728238e+28
Number of iterations: 1000 Learning Rate(eta): 0.01 Penalty: none	3.17129957e+10	-2.4708938e+10, 8.3075426e+10, -2.66534928e+10, -2.47089379e+10, 8.30754265e+10, -2.66534928e+10, 2.63466824e+10, 2.63027311e+10, 2.33188470e+11, -2.41521302e+10, 3.05834891e+10, -1.58227566e+11, -5.64427787e+09, 5.77282915e+10, -5.34278712e+10, 1.81899630e+10	1.1892380805728238e+28

CONCLUSION

By default, the SGD Classifier does not perform as well as the Logistic or Linear Regression. It requires some hyper parameter tuning to be done. The Error obtained is so large that the default linear regression model with all parameters set to default becomes preferable.