ACM Recruitments – Research Round 2

Raghav Sampath [21BEC0765]

Task:

Prepare a logistic regression model and an Artificial Neural Network to classify features from the Iris dataset. Explain in detail the various data-pre-processing techniques used, performance metrics for both the models and plot graphs for the same.

Pre-processing Steps:

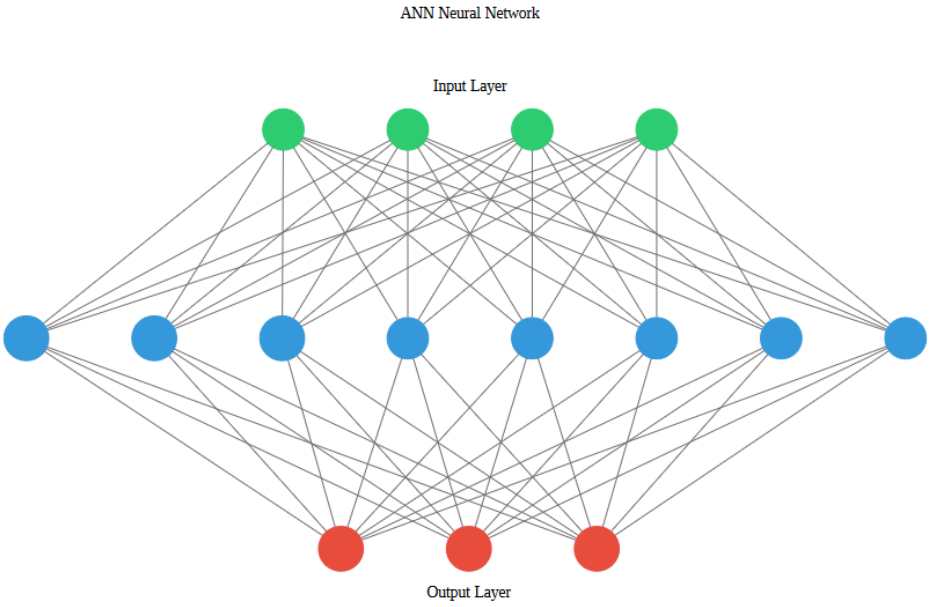
- Declare variable x for features.
- Declare variable y for classes.
- Split x and y using the train_test_split method in the ratio 7:3 (train:test)

Steps for Logistic Regression:

- Define a logistic regression model and set multi_class to 'ovr' (which stands for "one-vs-rest"). Used over multinomial as ovr performs better with small instances (50 for each in this case)
- Created model is then fit with the training split
- Model is then scored and used to predict using test data

Steps for ANN:

- The Training data used here is binarized using the label binarizer. to transform them into a binary format that can be understood by the algorithms. In this representation, the element corresponding to the true class label is set to 1, and all other elements are set to 0.
- Define a function for creating an ANN model using the keras library
- This ANN is a sequential model that
- First dense layer is added with 8 neurons. It takes an input tensor of size 4 (4 parameters) and returns an output tensor of 8. Using the ReLU activation function is used in the first dense layer to introduce nonlinearity into the model and to allow the model to learn complex patterns in the input data
- Second dense layer is added with 3 neurons. It takes an input tensor of size 8 (from above layer) and returns an output tensor of 3. It uses the softmax activation function which is commonly used in the output layer of multi-class classification problems, to produce probability scores for each of the output classes.
- Model is then compiled
- Number of parameters = (((current layer neurons)c * (previous layer neurons) p)+1*c)



- Model is then fit with train data and training begins for 200 epochs
- Model is then used to predict with testing data

Metric Calculation:

Confusion Matrix is plotting using the predicted and true values for both models and they yielded the following results.

$$recall = \frac{true\ positives}{true\ positives\ + false\ negatives} \qquad precision = \frac{true\ positives}{true\ positives + false\ positives}$$

- Recall: the ability of a classification model to identify all data points in a relevant class
- Precision: the ability of a classification model to return only the data points in a class
- F1 score: a single metric that combines recall and precision using the harmonic mean.
- MSE—that is, the average squared difference between the estimated values and the actual value. Lower the MSE, closer the predictions are to the actual value

1. **Logistic Regression:**

	Precision	**Recall**	**F1-Score**
setosa	1.0	1.0	1.0
versicolor	1.0	0.9375	0.96774
virginica	0.93333	1.0	0.96552

Accuracy: 97.78%
Mean-Squared Error: 0.0222

2. **ANN:**

	Precision	Recall	F1-Score
setosa	1.0	1.0	1.0
versicolor	0.88235	0.9375	0.85714
virginica	0.92308	0.85714	0.88889

Accuracy: 93.33%
Mean-Squared Error: 0.0667