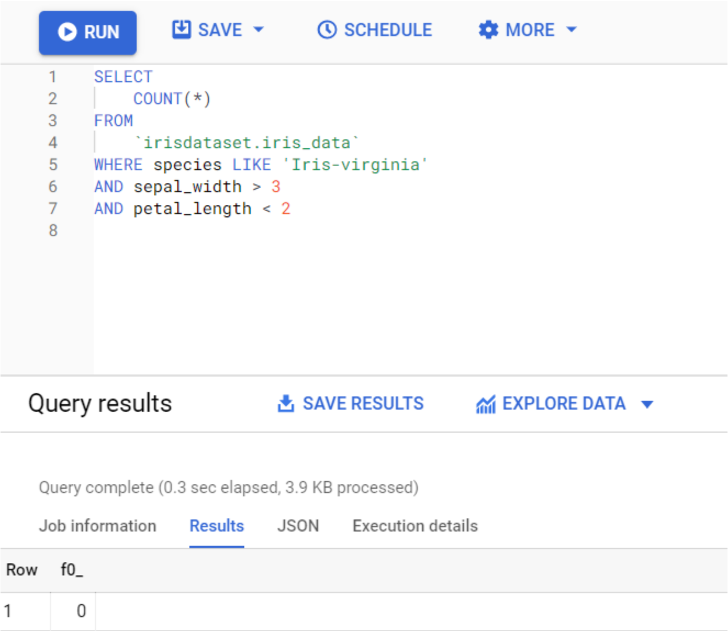
Big Data Lab - Lab 5

RA Keerthan
CH17B078

2) Count of sepal_width > 3 and petal_length < 2 is 0



3)Data Exploration and Feature Engineering

Firstly, we check if there are any NaN or Null values present in the data. Both the checks displayed a table as shown below (showing only one of them as the other table is also identical)

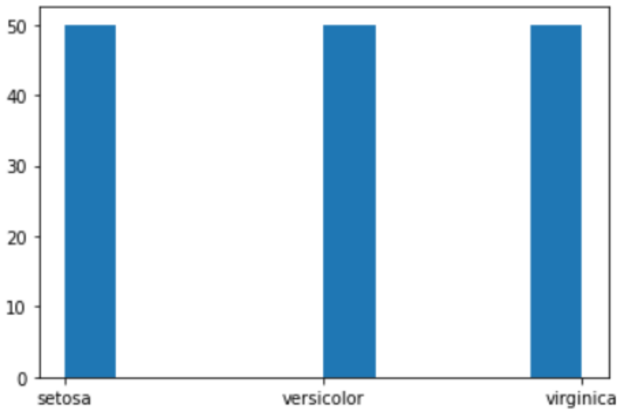
sepal_length	sepal_width	petal_length	petal_width	label
0	0	0	0	0

The table shows that there are no missing or NaN values in the dataset. Hence, data is already clean. As the next step, using spark dataframe's summary function, we get the following summary

summary	sepal_length	sepal_width	petal_length	petal_width
count	150	150	150	150
mean	5.843333326975505	3.0540000025431313	3.7586666552225747	1.198666658103466
stddev	0.8280661128539085	0.43359431104332985	1.7644204144315179	0.7631607319020202
min	4.3	2.0	1.0	0.1
max	7.9	4.4	6.9	2.5

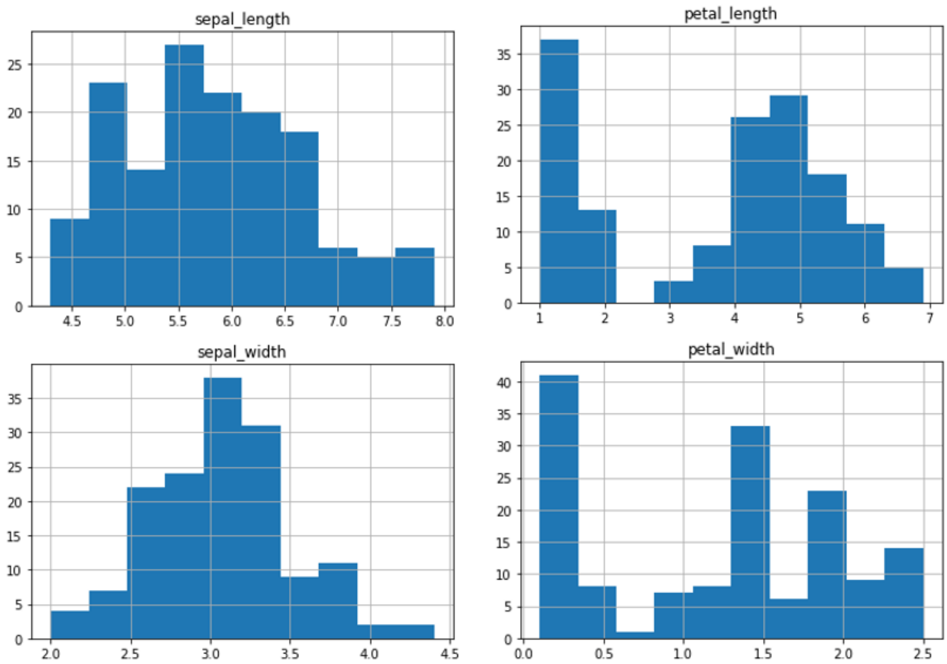
As we can see, the mean, min and max of the features are slightly shifted. Therefore, it is better to scale them using MinMax scaler.

Histogram of labels



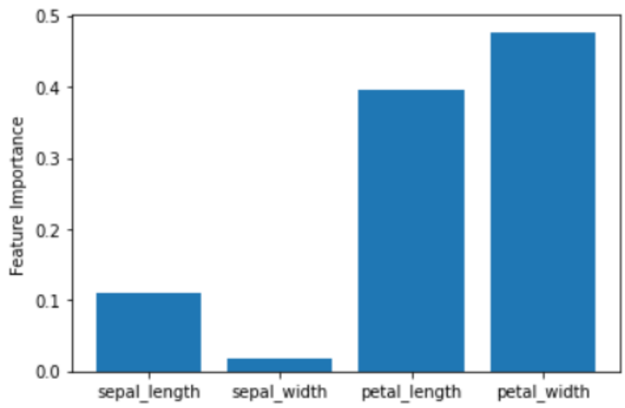
The histogram of labels shows us that the classes are balanced.

Histogram of features



Evaluating feature importance

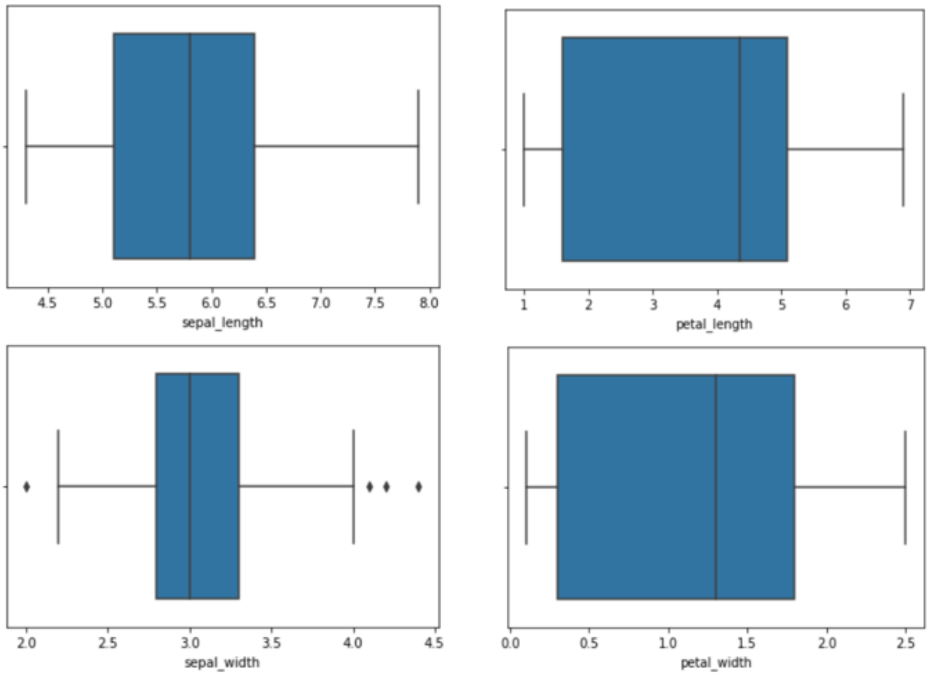
Feature importance were extracted after fitting a random forest model to the data.



As seen from the above plot, since sepal_width feature has a very low importance, we can remove that feature from our data.

Outlier detection and removal

Outliers are detected using box plot. Box plots uses the bounds provided by 1st quartile (Q1, 25th percentile), 3rd quartile (Q3, 75th percentile) and Inter Quartile Range (IQR) to determine outliers. If a given feature has a sample that does not lie in the range $[Q1 - 1.5IQR, Q3 + 1.5IQR]$, then that sample is regarded as an outlier and is eventually removed.



As seen from the above plots, there exists four samples in sepal_width feature that lies outside the bounds. Therefore, we remove the corresponding 4 datapoints.

All these plots will get saved in *gs://bdl_5/plots* directory after running *data_exp_and_feature_eng.py* as a DataProc job.

Pre-processing, fitting the model and results

The following are the pre-processing steps. We either do feature removal or outlier removal. A comparison of accuracy is provided between the two.

i. Convert categorical labels to numeric label
ii. Feature removal based on importance (or) Outlier removal.
iii. MinMax scaling
iv. Train-Test split of ratio 80:20

Model selection is carried out by 3-fold cross validation and is finetuned using gridsearch for respective parameters.

Models trained: Logistic Regression (LR), Random Forest (RF), Naïve Bayes (NB) and Decision Tree (DT)

Evaluation metric: Accuracy

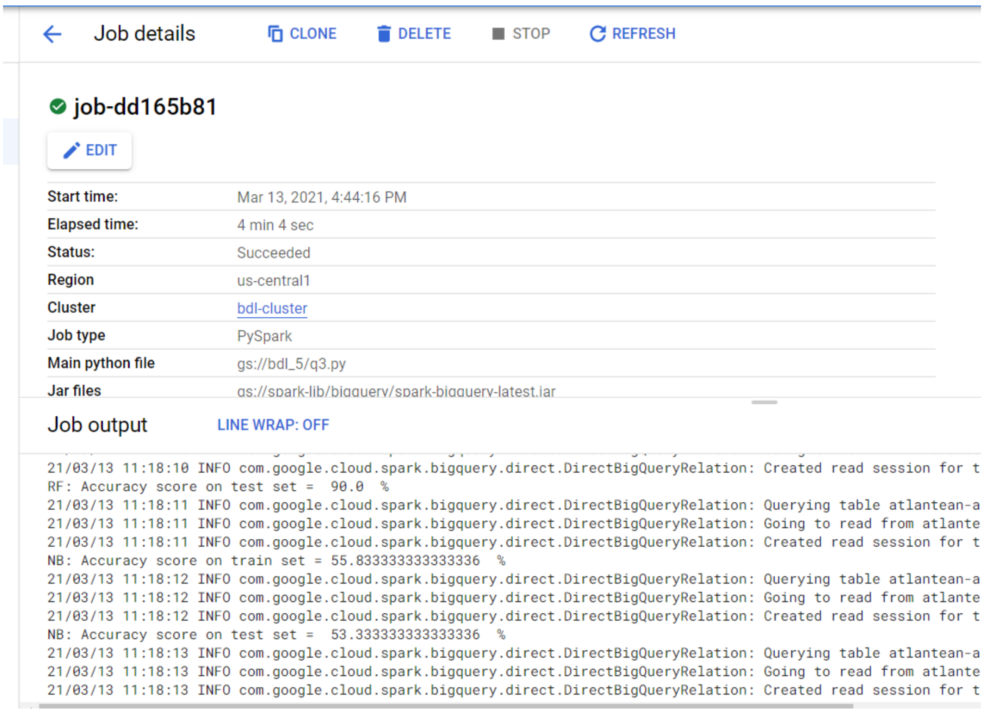
Since the outliers are because of 4 samples belonging to sepal_width feature, we can't have a case where we can do both outlier removal and feature removal.

Train\Test accuracy table

Technique\Model	LR	RF	NB	DT
Without feature removal and outlier removal	94.2\86.6	98.3\80	77.5\66.6	99.1\76.6
Without sepal_width	97.5\86.7	**99.9\90**	55.8\53.3	99.8\86.6
With outlier removal	88.8\80	99.5\83.3	63.8\66.6	99.5\83.3

The experiments related to the above table can be run by commenting out lines 52 to 54 in *q3.py* appropriately.

Screenshot of successful job run (for without sepal_width case)



From the table, we can see that **removing sepal_width feature and using random forest model** gave the highest combination of train and test accuracies.

The parameters of the random forest model that gave best results is below:

```
Number of trees for the best model: 10
Max Depth of best model: 10
Impurity of best model: entropy
```