Big Data Lab - Lab 7

RA Keerthan
CH17B078

1) Testing if producer.py works fine

producer.py reads each *iris.csv* as a dataframe and converts it to JSON which follows the same schema as that of *iris.csv*. Then it encodes each row and publishes it to the topic *irispred.* This is decoded back in subscriber.py and converted from json to dataframe following the same schema as that used in producer.py

Screenshot of working producer and subscriber:

At each instant, the row / batch of rows (depending on processing time) is printed on the console. Now that we have the rows received to the subscriber, we will move on to make real-time predictions!

2) Making real-time predictions

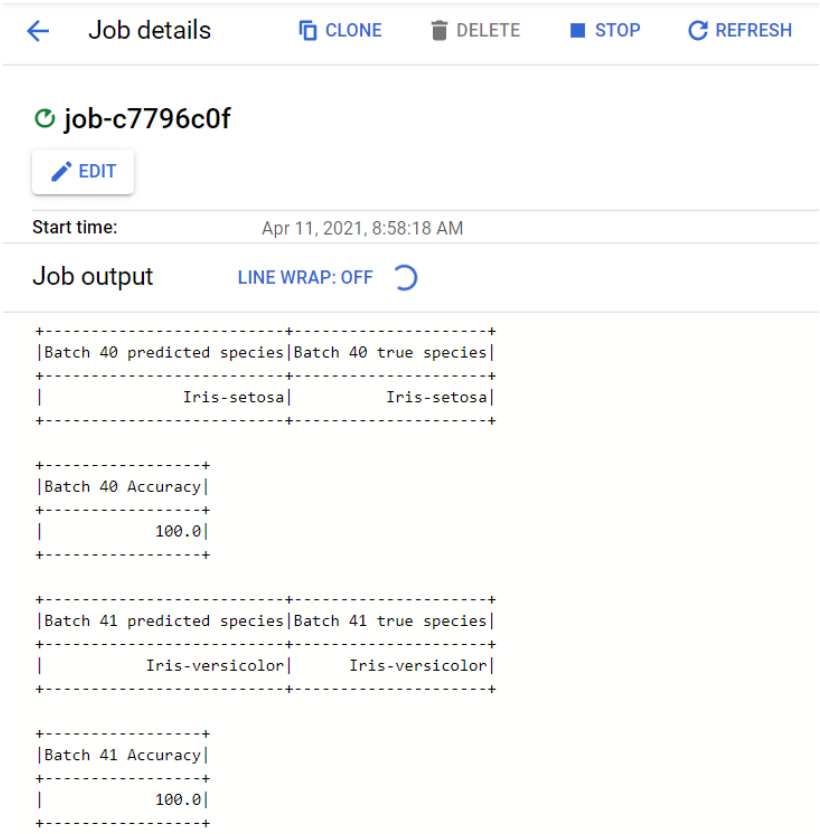
In order to make real-time predictions, we have to save the *pipeline*. The pipeline is saved using *.save()* method and loaded for real-time predictions using *.load()* method. We used the following pipeline:

$$string\ to\ index \rightarrow vector\ assembler \rightarrow minmax\ scaling \rightarrow random\ forest \rightarrow index\ to\ string$$

This pipeline is trained on a shuffled iris data with 80% training set and 20% test set. We also remove the *sepal_width* feature before giving the dataframe as input to the pipeline (reasoning given in lab5 report).

For finding accuracy, we use pyspark mllib's *MulticlassClassificationEvaluator* and apply this for every batch using *foreachBatch* method in *writeStream*.

Screenshot of real-time prediction with true label and accuracies:



Instance of misclassification: