Assignment 3

Vathana Him

November 25, 2021

1 Abstract

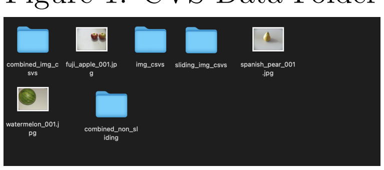
The purpose of this assignment is to use the data of the image created from assignment 1 to build classification models in a distributed system in databricks in order to classify the choosen images and compare the results and run time to that of the model built in our local machine. This assignment utilized images from UCI respository as sample datasets that will be used to train a machine learning model. Images that were processed represented three fruits spanish pear, fuji apple, watermelon. These images were labeled as Image0, Image1, and Image2 respectively and their dataset was processed and dervied in assignment1 for both the non-overlapping and overlapping layer. These labels was then encoded to take in the values of 0, 1, and 2. The primary machine learning that was used in order to classify these images was random forest. Prior to training the machine learning model, additional methods were taken into account in feature selection and data scaling in order to reduce the size of the data and train the model with a sufficient outcome. The random forest model in databricks was then used to compare to the random forest model that was dervied in our local machine in assignment2.

2 Task 1

In assignment1, the three images were divided into 8X8 pixel blocks, the grayscale images were then divided into sliding block of 8X8 pixels. Each feature was then assigned a label respectively to identify them. Two helper functions were used for this task, sliding blocks feature function which converts an image into sliding image blocks given an image object, and label feature function which create feature labels for each image given a list that contains the array of sliding images. The dimension of the gray images were resized to height of 256 and width of 344. The purpose for these choosen dimensions was to keep its aspect ratio. Additionally, the resized height and width must also be divible by eight since this project divided the targeted image into sliding blocks and non-sliding blocks of 8 by 8 height and width. T he feature vector that was constructed from these images created 6800 feature vectors for the sliding block and 3400 feature vectors for the non-sliding block and each feature vector lies a 8 by 8 pixels who's value lies between 0-255 of the gray image scale. The features of each feature vector was then flatten to 64 features for each respective feature vector. These datasets was exported into cvs files in the data folder. The evidence of this dataset can be shown in 2.1 Figure 1.

2.1 Assignment 1 Figure

Figure 1: CVS Data Folder



In assignment2, the random forest classifier was used to classify the images of different set in non-overlapping image01, overlapping image01, non-overlapping image012, and overlapping image012. Feature selection was also used in this model to increase the speed of the training time and reduce the computational power. The select from model feature selection from Sklearn compare the average importance of all features at a threshold value and dropped features that were below the threshold. Additionally the elastic net model will also be presented. However, the random forest models will only be used to compare with the databricks model.

In the two class classification for non-overlapping image0 and image1, the training accuracy score was 0.95 and the testing accuracy score was 0.92 based on 2.2 Figure 2. There was not a significant difference between the train and test score, this suggests that the train-test split provided a well balanced data between the two classes. The confusion matrix in 2.2 Figure 3 confirmed a true prediction value of 240 and false prediction value of 32 for class 0 and a true prediction value of 269 and false prediction value of 10 for class 1. This indicate that the accuracy rate and the precision rate for class 0 and class 1 was relatively as seen in 2.2 Figure 4 of the derived accuracy score and precision for both class 0 and class 1 respectively. Class 0 had a precision rate of 0.92, while class 1 had a precision rate of 0.89. This model provided a good accruacy for each of the predicted classes.

2.2 Assignment 2 Figure

Figure 2: Non-Overlapping Image01 RF Score

```
Final Training Accuracy: 0.9545661063153112
Testing Accuracy :  0.9237749546279492
```

Figure 3: Non-Overlapping RF Confusion-Matrix

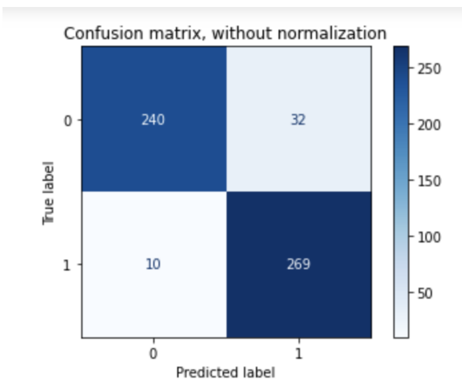


Figure 4: Non-Overlapping RF Derived Score

```
Accuracy : 0.7808716707021792
Precision class 0 : 0.8907563025210085
Precision class 1 : 0.7391304347826086
Precision class 2 : 0.7335640138408305
```

For the overlapping two-class classificaion of image0 and image1, the training accruacy score was 0.96 and the testing accuracy score was 0.92 based on 2.3 Figure 5. This small difference in accuracy score indicates that the train-test split provided an evenly balanced data for the test and train set for the random forest model. The confusion matrix on 2.3 Figure 6 provided the result of the test set as class 0 had 590 true prediction and 74 false prediction, while class 1 had 666 true prediction and 30 false prediction. This indicated a high precision value for both class 0 and class 1 because the model was able to make a prediction of the two images at a high accuracy rate. Based on the value of this confusion matrix, the hand calculation for accuracy score, precision for class 1 and precision for class 0 was derived. In 2.3 Figure 7, the accuracy score from the dervied calculation was 0.92 with a precision of 0.95 and 0.9 for class 0 and class 1 respectively. Based on these high precision values, it indicated that this model can be produced the same results when test with another dataset of the same characteristics. This model also performed significantly better than the elstaicnet for two-class classification of overlapping image0 and image1.

2.3 Assignment 2 Figure

Figure 5: Over-lapping Image01 RF Score

```
Final Training Accuracy: 0.9595588235294118
Testing Accuracy :  0.9235294117647059
```

Figure 6: Over-lapping RF Confusion-Matrix

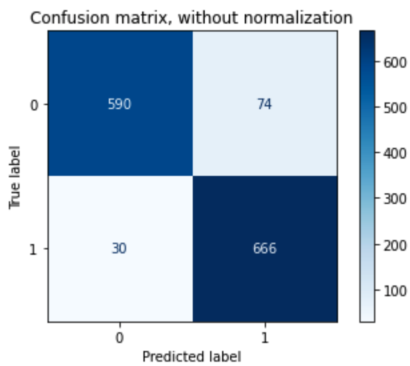


Figure 7: Over-lapping RF Derived Score

```
Accuracy : 0.9235294117647059
Precision class 0 : 0.9516129032258065
Precision class 1 : 0.9
```

In the non-overlapping elastic model, the data for non-overlapping image01 was used. The result of the model for non-overlapping image01 yielded a training accuracy of 0.69 and testing accuracy of 0.67 as shown in 2.4 Figure 8. A similar score in both the train and test set indicate that the data was balanced between the train and test set. The confusion matrix in 2.4 Figure 9 indicate that there were 197 predictions of True Positive for class 0 and 175 predictions of True Positive for class 1. These results was then used to manually derived the accuracy and precision. According to the manual derivated result in 2.4 Figure 10, the overall accuracy of the model was 0.72 with a precision of 0.72 for class 0 and 0.72 for class 1. This accuracy score indicate that there was a sufficient number of true postives for class 0 and class 1. However, the number of false postives was still indicative in affecting the accuracy score.

2.4 Assignment 2 Figure

Figure 8: Non-Overlapping Image01 Elastic-Net

```
Final Training Accuracy: 0.6942298955020445
Testing Accuracy :  0.6751361161524501
```

Figure 9: Non-Overlapping Image01 Elastic-Net Confusion-Matrix

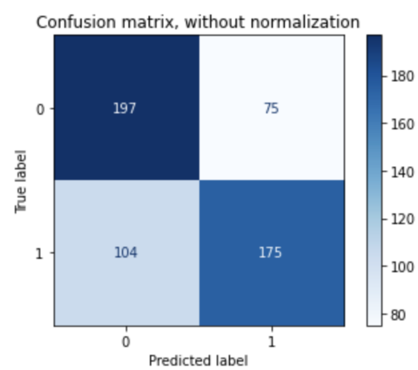


Figure 10: Non-Overlapping Image01 Elastic-Net Derived Score

```
Accuracy : 0.720508166969147
Precision class 0 : 0.7269230769230769
Precision class 1 : 0.7147766323024055
```

In the second elastic model, the data for overlapping image01 was used. The result of the model for overlapping image01 resulted in a training accuracy of 0.60 and a testing accuracy of 0.59. This inidcated that the data had a great degree of randomeses and it was balanced in the train-test set. The confusion matrix in 2.5 Figure 11 resulted in 493 prediction of true prediction and 314 of true prediction for class 0 and 1 respectively. However, in the derived precision score for class 0 was relatively lower than that of class 1 as shown in 2.5 Figure 12. This could indicate that there was an imbalance in the dataset between class 0 and class 1. Additionally, because of the nature of the image choosen, the black and white image of apple and pear had similar texture and texture. The overlapping nature of the dataset could distort the elastic-net loss function, when it attempted to classify the two images. This also impacted the overall dervied accuracy score of the model with a value of 0.65 as shown in 2.5 Figure 13. An accuracy score of 0.65 indicate that this model was not sufficient enough for making a prediction.

2.5 Assignment 2 Figure

Figure 11: Overlapping Image01 Elastic-Net

```
Final Training Accuracy: 0.6071691176470588
Testing Accuracy :  0.5933823529411765
```

Figure 12: Overlapping Image01 Elastic-Net Confusion-Matrix

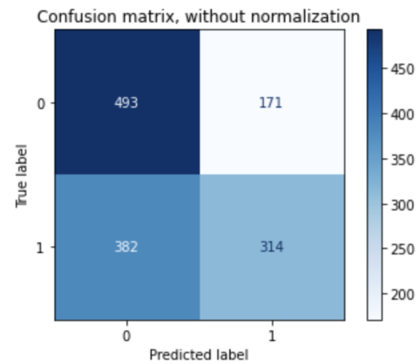


Figure 13: Overlapping Image01 Elastic-Net Derived Score

```
Accuracy : 0.6584615384615384
Precision class 0 : 0.5982800982800983
Precision class 1 : 0.7592592592592593
```

In contrast to the two-class non-overlapping classification random forest, the three-class classification of image0, image1 and image2 testing and training accuracy score deviate in larger degree. In 2.6 Figure 14, the training accuracy for this model is 0.89, whereas the testing accuracy for this model is 0.78. This may indicate an overfit in the model and that the train-test split set did not generate a well balanced enough data. The confusion matrix in 2.6 Figure 15 showed that a true prediction value of 212 for class 0, 221 for class 1, and 212 for class 2. These values was then used to dervied the caculated precision for each of the class. 2.6 Figure 16 indicated that class 0 had a 0.89 precision, class 1 had a precision of 0.74 and class 2 ha da precision of 0.73. The difference in this precision score can suggest that the model was overfitted to favor class 0.

2.6 Assisgnment 2 Figures

Figure 14: Non-Overlapping Image012 RF Score

```
Final Training Accuracy: 0.8994548758328286
Testing Accuracy :  0.7808716707021792
```

Figure 15: Non-Overlapping Image012 RF Confusion-Matrix

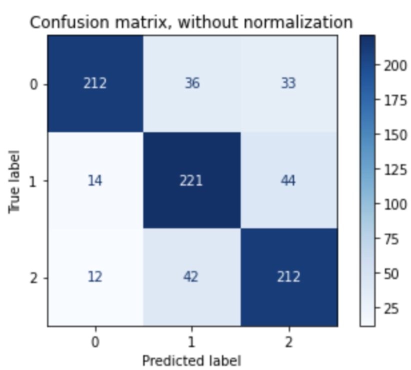


Figure 16: Non-Overlapping Image012 RF Derived Score

```
Accuracy : 0.7808716707021792
Precision class 0 : 0.8907563025210085
Precision class 1 : 0.7391304347826086
Precision class 2 : 0.7335640138408305
```

Similar to the three-class non-overlapping random forest model, the three-class classification of overlapping image0, image1 and image2 testing and training accuracy score also deviated to a noticeable extent. In 2.7 Figure 17, the training accuracy for this model is 0.91, whereas the testing accuracy for this model is 0.83. This may indicate a slight overfit in the model and that the train-test split set did not generate a well balanced enough data. The confusion matrix in 2.7 Figure 18 showed that a true prediction value of 552 for class 0, 608 for class 1, and 545 for class 2. These values was then used to dervied the caculated precision for each of the class. 2.7 Figure 19 indicated that class 0 had a 0.94 precision, class 1 had a precision of 0.77 and class 2 ha da precision of 0.81. The difference in this precision score can suggest that the model was overfitted to favor class 0, which is similar to that of the three-class random forest non-overlapping model. Although the dataset was randomly shuffled, the training set may have contained slightly more data for class 0 than that of class 2 and class 1.

2.7 Assisgnment 2 Figures

Figure 17: Overlaping Image012 RF Score

```
Final Training Accuracy: 0.9126225490196078
Testing Accuracy :  0.8357843137254902
```

Figure 18: Overlapping Image012 RF Confusion-Matrix

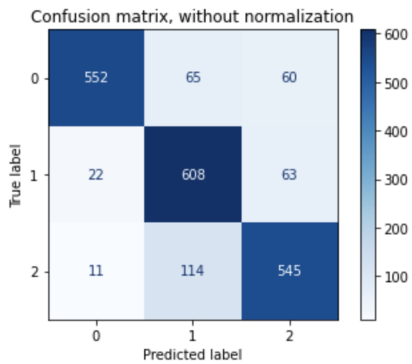


Figure 19: Overlapping Image012 RF Derived Score

```
Accuracy : 0.8357843137254902
Precision class 0 : 0.9435897435897436
Precision class 1 : 0.772554002541296
Precision class 2 : 0.8158682634730539
```

3 Task 2 (Knowledge Gained)

The purpose of this task is to explained the knowledge gained through databricks "Explore the Quickstart Tutorial Section".[1]. Databricks seems to be a cloud-based platform that can run ETL processes for processing and transforming large quantities of data for machine learning models. Databrick has a network of distributed systems that allows it to handle big quantities of data without any time loss through its system. These distributed systems are powered by third-party cloud providers such as Google Cloud, Azure Web Services, and Amazon Web Services. For example in the quick start tutorial, the dataset that was used was stored in a Databricks dataset directory, which used the storage engine of Amazon Web Services. Instead of storing it locally, this dataset is stored on the cloud that is provided by Amazon Web services as this method of storage can handle millions of terabytes of data in theory. In the quick start tutorial, the notebook exported the data from a dataset that was stored in an Amazon Web Services storage and loaded through a cloud computing cluster that is also provided by Amazon Web Services. When these services are integrated, the notebook works as if you're working on your local machine notebook.

You can load the dataset, process the dataset to draw insights and analysis as well as train machine learning models. Additionally, the notebook can be chosen to work with specific instances of a data language. These instances can include pyspark, python, sql and scala. The integration of multiple data processing languages in a singular cluster of computing resource that is provided by the users chosen cloud service provider can offer a seemly effortless workflow that allows users to work, integrate, and build machine learning model flows as well as per-

[1] https://docs.databricks.com/getting-started/quick-start.html

forming extract, load, and transformation processes to a data storage platform that is not dependent on a local computing power. This will not confine a user's big data project to a single local computer resource and will speed up the processing power of end-to-end machine learning model flows through the use of cloud computing clusters.

This method of processing and storing data can be coined the term "data lake" and "data warehouse" as databases provide a service of cloud data platform that leverages the cloud service of cloud providers such as Google Cloud, Azure Web Services, and Amazon Web Services. This type of architecture was based on an open source Apache Spark framework that allows users to query against semi-structured data without having to use the traditional database schema for the purpose of speed and efficiency. Since cloud storage and cloud computing (clusters) are used, the limited source of on-prem computing resources is nolonger a detriment to working with massive amounts of data.

In order to set up databricks for Task 3, the following steps was done. Databricks was connected to an Amazon Web Services account in order to use their storage system as well as their EC2 cloud computing to generate the needed custer of cpu resources. When Amazon Web Services was connected, the cpu cluster was configured. The configured clusters had the following specs 7.3 LTS (includes Apache Spark 3.0.1, Scala 2.12) as seen in 3.1 Figure 20. The data is then loaded into the DBFS, which is backed by AWS storage as shown in 3.1 Figure 21.

3.1 Task 2 Figures

Figure 20: Configured CPU Cluster

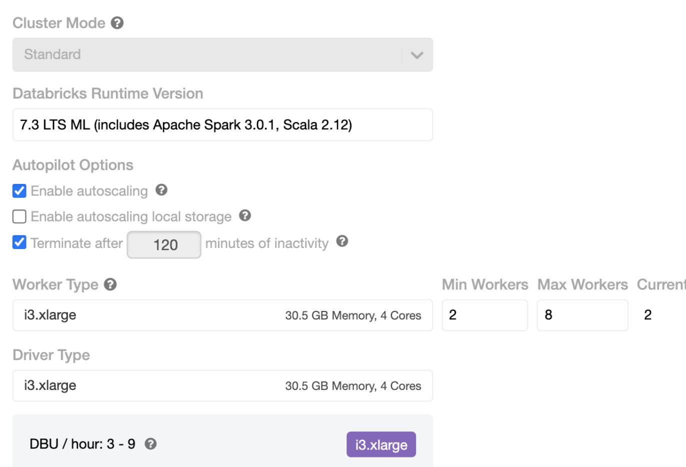


Figure 21: Loaded Data

4 Task 3

In Task 3, the random forest multi-class classifier was implemented in Databricks datastributed system with backend support from the cpu cluster and storage capcacity of Amazon Web Services EC2 and S3. The code that was implemented had charactertistics of the code from assignment2 through the use of random forest classification to classify three choosen images in fuji apple, pear, and watermelon. These images were labeled image0, image1 and image2 respectively. In the data preprocessing step of this task, an analysis of the dataset for non-overlapping image01, overlapping image01, non-overlapping image012, and overlapping image012 was done. It was seen that the number of class label for each image was equally distributed for both the overlapping and non-overlapping set as shown in 4.1 Figure 24 and Figure 25. This indicate that there was no imbalance of classes among the dataset. Subsequently, the distribution of the dataset was analzyed among all the feature space of feature 54 to determine if anything scaling is needed. It can be seen that in the 4.1 boxplots of Figure 23, Figure 25, Figure 27, and Figure 29, the data among the three classes of overlapping and non-overlapping images as well as the two classes of overlapping and non-overlapping images contain outliers and non-normally distributed data. From this finding, a scaling of min-max for each respective dataset was used before it was fed into the random forest machine learning model. Finally, since we are training on a large distributed system all features of the dataset was used since we're not limited by the computation powers of our local machine.

4.1 Task 3 Figures Preprocessing

Figure 22: Non-Overlaping Image01 distribution

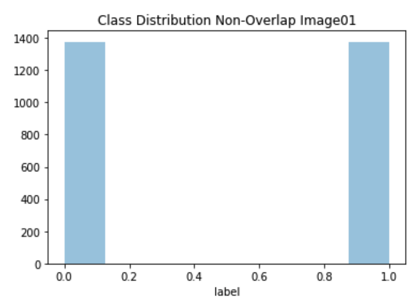


Figure 23: Non-Overlaping Image01 boxplot

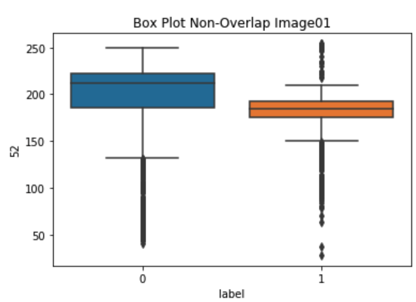


Figure 24: Non-Overlaping Image012 distribution

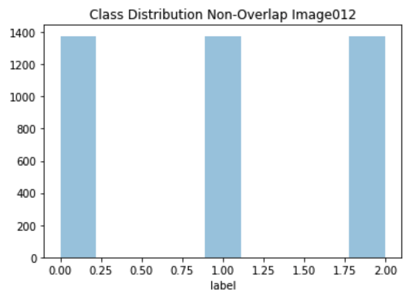


Figure 25: Non-Overlaping Image012 boxplot

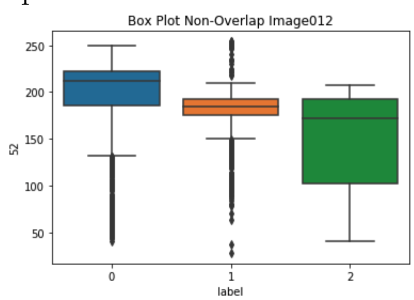


Figure 26: Overlaping Image01 distribution

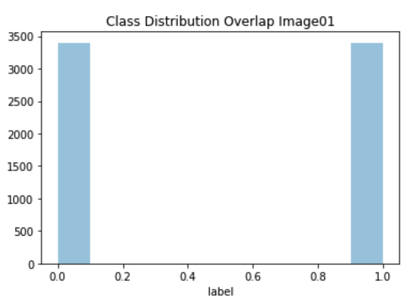


Figure 27: Overlaping Image01 boxplot

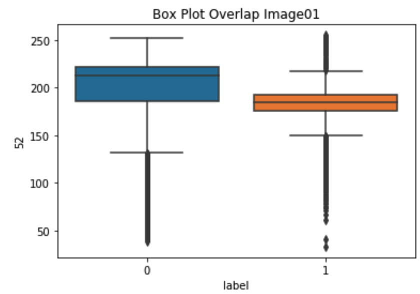


Figure 28: Overlaping Image012 distribution

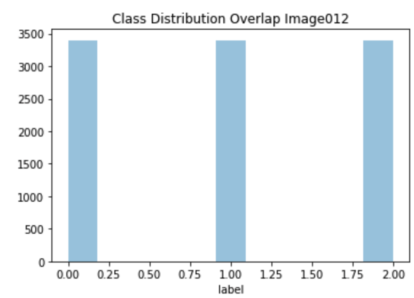
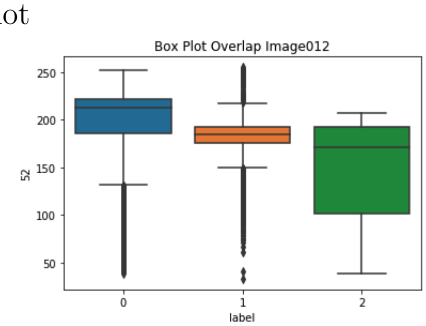


Figure 29: Overlaping Image012 boxplot



The random forest machine learning model implemented in databricks for non-overlapping image01 provided very good results in terms of numeric accuracy and precision measurements. In 4.2 Figure 30, the final training accuracy was 0.95 and testing accuracy was 0.92.

This suggested that the splitting of training and testing data provided an equally balanced data for the model to use. Additionally, class0 and class1 had a precision rate of 0.96 and 0.89 respectively, which produces very high true-poistives values for the respective classes. The ROC curve in 4.2 Figure 32 also provided a very high rate of true positive to low false postitive based on the auc value of 0.92. This indicate a very high rate of accurate prediction among the two classes.

4.2 Task 3 Figures Non-Overlapping Image01

Figure 30: Non-Overlaping Image01 Score

```
Final Training Accuracy: 0.955474784189005
Testing Accuracy :  0.9183303085299456
              precision    recall  f1-score   support

           0       0.96      0.88      0.91       272
           1       0.89      0.96      0.92       279

    accuracy                           0.92       551
   macro avg       0.92      0.92      0.92       551
weighted avg       0.92      0.92      0.92       551
```

Figure 31: Non-Overlaping Image01 Confusion Matrix

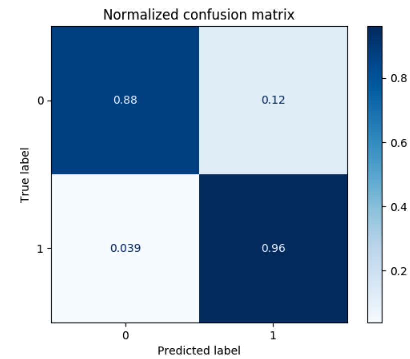
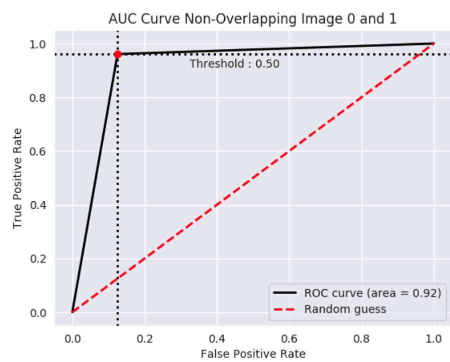


Figure 32: Non-Overlaping Image01 ROC Curve



In comparision, the random forest machine learning model implemented in databricks for overlapping image01 and overlapping image01 also provided very good results in terms of numeric accuracy and precision measurements. In 4.3 Figure 33, the final training accuracy was 0.97 and testing accuracy was 0.92. This suggested that the splitting of training and testing data provided an equally balanced data for the model to use. Additionally, class0 and class1 had a precision rate of 0.95 and 0.90 respectively, which produces very high true-poistives values for the respective classes. The ROC curve in 4.3 Figure 35 also provided a very high rate of true positive to low false postitive based on the auc value of 0.92. This indicate a very high rate of accurate prediction among the two classes. The result of the random forest model for the two class dataset for overlapping and non-overlapping images provided nearly identical results, which indicate that this model can be a good predictor of a two class fruit dataset.

4.3 Task 3 Figures Overlapping Image01

Figure 33: Overlaping Image01 Score

```
Final Training Accuracy: 0.9700367647058824
Testing Accuracy :  0.9205882352941176
              precision    recall  f1-score   support

           0       0.95      0.89      0.92       664
           1       0.90      0.95      0.92       696

    accuracy                           0.92      1360
   macro avg       0.92      0.92      0.92      1360
weighted avg       0.92      0.92      0.92      1360
```

Figure 34: Overlaping Image01 Confusion Matrix

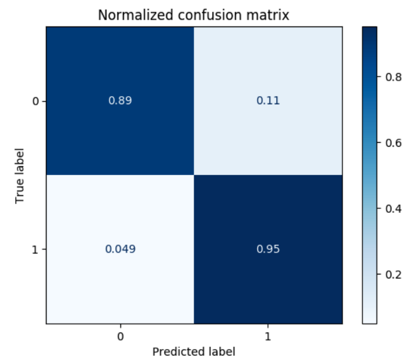
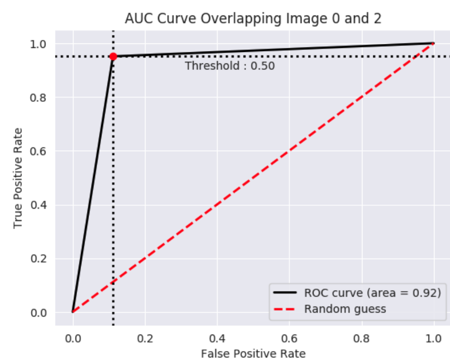


Figure 35: Overlaping Image01 ROC Curve



The random forest machine learning model implemented in databricks for non-overlapping image012 provided insufficent results in terms of numeric accuracy and precision measurements. In 4.4 Figure 36, the final training accuracy was 0.93 and testing accuracy was 0.80. This suggested that the splitting of training and testing data did not provided an equally balanced data for the model to use as indicated by the large differences in training and testing accuracy values. Class1 and class2 had a precision rate of 0.74 and

0.76 respectively, while class 0 had a precision rate of 0.92. This large differences in precision rate between class0 to class1 and class2 indicate that the model was more biased to class0. This can indicate that the split of training-testing set method may have unintentionally included more dataset from class0 in one of the sets than class1 and class2. The ROC curve in 4.4 Figure 39 and Figure 40 also provided a very low rate of true positive to high false postitive based on the auc value of 0.44 and 0.62 for class1 and class2 respectively. In 4.4 Figure 38, the ROC curve for class0 was sufficient with a good rate of true postive to false positive as supported by the auc value of 0.78. This large differences in the auc values among class0 to class1 and class2 the models bias towards class0 and thus, it may not be a good model for the three class image classfication.

4.4 Task 3 Figures Non-Overlapping Image012

```
Final Training Accuracy: 0.9279224712295578
Testing Accuracy :  0.7990314769975787
              precision    recall  f1-score   support

           0       0.92      0.77      0.84       281
           1       0.74      0.81      0.77       279
           2       0.76      0.82      0.79       266

    accuracy                           0.80       826
   macro avg       0.81      0.80      0.80       826
weighted avg       0.81      0.80      0.80       826
```

Figure 36: Non-Overlaping Image012 Score

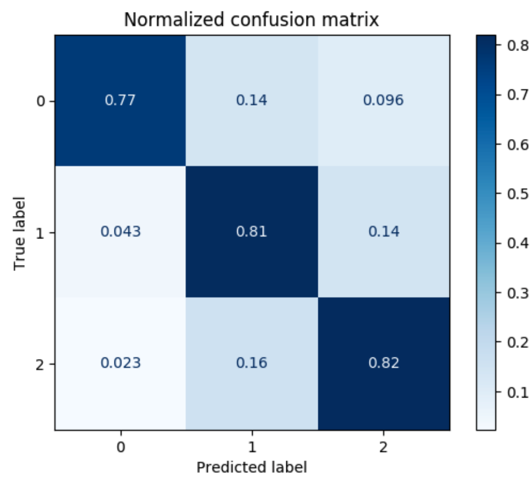


Figure 37: Non-Overlaping Image012 Confusion Matrix

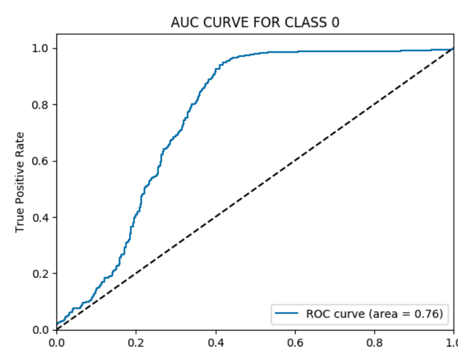


Figure 38: Non-Overlaping Image012 Class 0 ROC Curve

Figure 39: Non-Overlaping Image012 Class 1 ROC Curve

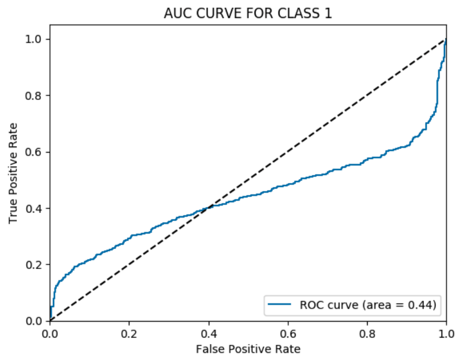
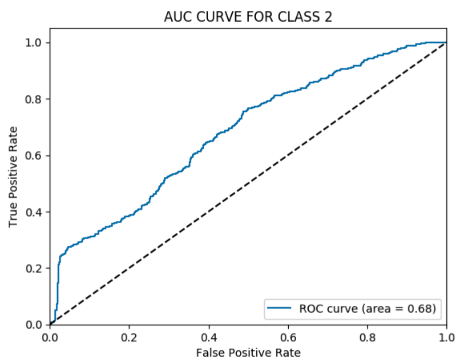


Figure 40: Non-Overlaping Image012 Class 2 ROC Curve



In contrast, The random forest machine learning model implemented in databricks for overlapping image012 provided a sufficent results in terms of numeric accuracy and precision measurements. In 4.5 Figure 41, the final training accuracy was 0.95 and testing accuracy was 0.85. This suggested that the splitting of training and testing data provided a slightly less balanced data for the model to use as indicated by the small differences in training and testing accuracy values. Class0, class1 and class2 had a precision rate of 0.95, 0.80, and 0.85 respectively, the result of this precision rate indicate that the model might had a small bias towards class0, also performed sufficiently when classify class1 and class2. The ROC curve in 4.5 Figure 43 and Figure 45 also provided sufficient rate of true positive to false postitive based on the auc value of 0.78 and 0.70 for class0 and class2 respectively. However, in 4.5 Figure 44, the ROC curve for class1 was insufficient as supported by the auc value of 0.58. This large differences in the auc values among class0 and class2 to class 0 as suggested by the ROC curve may suggests that the model is bias towards class0 and class2. Nevertheless, its prediction was sufficient enough to classify the three images as supported by the 4.5 Figure 42 confusion matrix as the true positive rate for class 0, class 1 and class 2 was 0.82, 0.90 and 0.86 respectively.

4.5 Task 3 Figures Overlapping Image012

Overall, the two class model for overlapping and non-overlapping dataset performed significantly better than that of the three class model as it had no bias towards one class versus the other class. However, in the scope of the three class model, the overlapping model performed better than the non-overlapping model as it resulted in less bias towards any of the three classes. It was also able to predict the three classes sufficiently as supported by the recall rates and the true positive rates for each of the classes shown in the 4.5 Figures.

Figure 41: Overlaping Image012 Score

```
Final Training Accuracy: 0.9504901960784313
Testing Accuracy :  0.859313725490196
              precision    recall  f1-score   support

           0       0.95      0.82      0.88       677
           1       0.80      0.90      0.85       693
           2       0.85      0.86      0.86       670

    accuracy                           0.86      2040
   macro avg       0.87      0.86      0.86      2040
weighted avg       0.87      0.86      0.86      2040
```

Figure 42: Overlaping Image012 Confusion Matrix

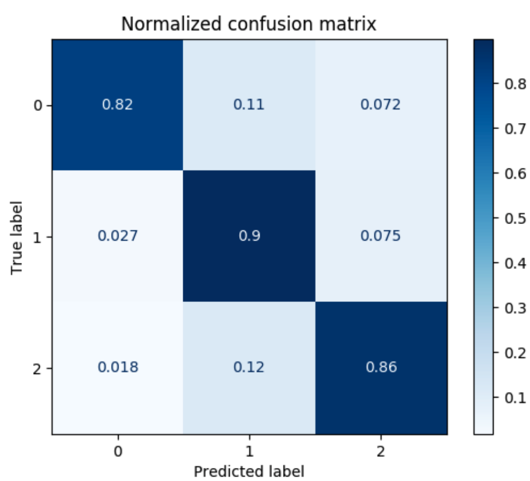


Figure 43: Overlaping Image012 Class 0 ROC Curve

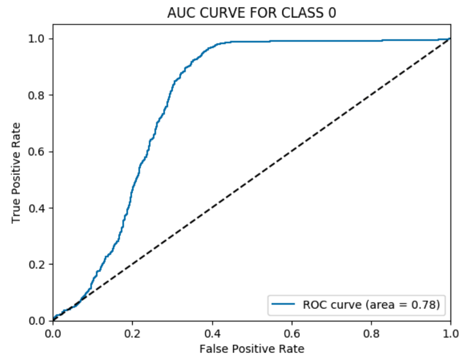


Figure 44: Overlaping Image012 Class 1 ROC Curve

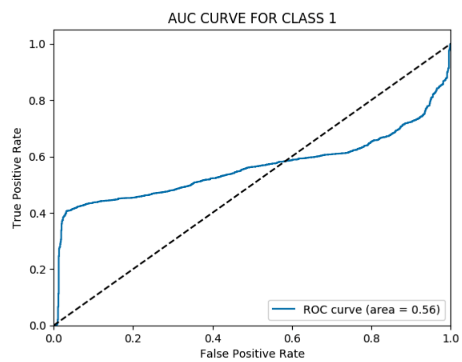
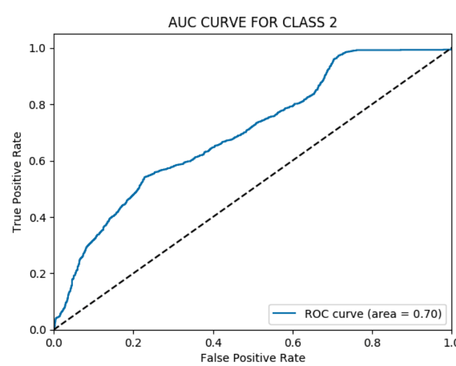


Figure 45: Overlaping Image012 Class 2 ROC Curve



5 Task 4

An overview of the results from the local machine and databricks powered by Amazon Web Services can be found in subsection 5.1 and 5.2 figures. There was quit a noticeable difference between the performance speed, when training the random forest model on the local machine in comparison to training in on the databricks platform. When training on the databricks platform, the model took about 1.5 seconds to train on a full set of features, whereas in the local machine it took aproximately 3.5 on just half of the features in the dataset. The computing power and performance speed of databricks over performed the local machine.

5.1 Task 4 Figures Local Machine Score

Figure 46: Non-Overlapping Image01 RF Score

Final Training Accuracy: 0.9545661063153112
Testing Accuracy : 0.9237749546279492

	precision	recall	f1-score	support
0	0.96	0.88	0.92	272
1	0.89	0.96	0.93	279
accuracy			0.92	551
macro avg	0.93	0.92	0.92	551
weighted avg	0.93	0.92	0.92	551

Figure 47: Overlapping Image01 RF Score

Final Training Accuracy: 0.9595588235294118
Testing Accuracy : 0.9235294117647059

	precision	recall	f1-score	support
0	0.95	0.89	0.92	664
1	0.90	0.96	0.93	696
accuracy			0.92	1360
macro avg	0.93	0.92	0.92	1360
weighted avg	0.93	0.92	0.92	1360

Figure 48: Non-Overlapping Image012 RF Score

Final Training Accuracy: 0.8994548758328286
Testing Accuracy : 0.7808716707021792

	precision	recall	f1-score	support
0	0.89	0.75	0.82	281
1	0.74	0.79	0.76	279
2	0.73	0.80	0.76	266
accuracy			0.78	826
macro avg	0.79	0.78	0.78	826
weighted avg	0.79	0.78	0.78	826

Figure 49: Overlapping Image012 RF Score

Final Training Accuracy: 0.9126225490196078
Testing Accuracy : 0.8357843137254902

	precision	recall	f1-score	support
0	0.94	0.82	0.87	677
1	0.77	0.88	0.82	693
2	0.82	0.81	0.81	670
accuracy			0.84	2040
macro avg	0.84	0.84	0.84	2040
weighted avg	0.84	0.84	0.84	2040

For the comparison of the dervied results of the model, an analysis of their results was done through the scope of the respective model. For the non-overlapping model for image0 and image1, the results derived was similar in both the local machine and databricks. In 5.1 Figure 46, the local model scored a training and testing accuracy of 0.95 and 0.89 and the databricks model scored a testing and training accuracy of 0.96 and 0.89 as shown in 5.2 Figure 50. The predicting precision for class 0 and class 1 was also similar for both models as both scored at 0.96 and 0.89.

For the model of overlapping image for image0 and image1, the results were nearly identical as the testing and training accuracy for both models were aproximately 0.96 and 0.95 respectively as seen in 5.1 and 5.2 Figure 47 and Figure 51. The precision rate for class0 and class1 was also identical as both models had a precision of aproximately 0.95 and 0.90 for the respective classes.

As for the model for non-overlapping image for image0, image1, and image2. The databricks model performed slightly better than that of the local model as it scored a 0.92 and 0.80 on the training and testing accuracy based on 5.2 Figure 52. However, the results were not significantly different enough to warrant a decision in deciding which model was fundamentally better as the local model training and testing accuracy came close with a score of 0.78 and 0.90 as shown in 5.1 Figure 48. The precision score of the databricks model for class0, class1, and class2 was also slightly higher with a score of 0.89, 0.74 and 0.73 respectively. However, this did not warrant that the justification that one model was better than the other as the differences in scores was only by a few percentage points.

Similary the results for overlapping image for image0, image1, and image2, the databricks model also performed slightly better than that of the local model as it scored a 0.95 and 0.86 on the training and testing accuracy based on 5.2 Figure 53. However, like the non-overlapping results, it was not significantly different enough to warrant a decision in deciding which model was fundamentally better as the local model training and testing accuracy came close with a score of 0.84 and 0.91 as shown in 5.1 Figure 49. The precision score of the databricks model for class0, class1, and class2 was also slightly higher with a score of 0.95, 0.80 and 0.85 respectively. However, this did not warrant that the justification that one model was better than the other as the differences in scores was only by a few percentage points.

5.2 Task 4 Figures Databricks Score

Figure 50: Non-Overlapping Image01 RF Score

```
Final Training Accuracy: 0.955474784189005
Testing Accuracy :  0.9183303085299456
              precision    recall  f1-score   support

           0       0.96      0.88      0.91       272
           1       0.89      0.96      0.92       279

    accuracy                           0.92       551
   macro avg       0.92      0.92      0.92       551
weighted avg       0.92      0.92      0.92       551
```

Figure 51: Overlapping Image01 RF Score

Final Training Accuracy: 0.9700367647058824
Testing Accuracy : 0.9205882352941176

	precision	recall	f1-score	support
0	0.95	0.89	0.92	664
1	0.90	0.95	0.92	696
accuracy			0.92	1360
macro avg	0.92	0.92	0.92	1360
weighted avg	0.92	0.92	0.92	1360

Figure 52: Non-Overlapping Image012 RF Score

Final Training Accuracy: 0.9279224712295578
Testing Accuracy : 0.7990314769975787

	precision	recall	f1-score	support
0	0.92	0.77	0.84	281
1	0.74	0.81	0.77	279
2	0.76	0.82	0.79	266
accuracy			0.80	826
macro avg	0.81	0.80	0.80	826
weighted avg	0.81	0.80	0.80	826

Figure 53: Overlapping Image012 RF Score

Final Training Accuracy: 0.9504901960784313
Testing Accuracy : 0.859313725490196

	precision	recall	f1-score	support
0	0.95	0.82	0.88	677
1	0.80	0.90	0.85	693
2	0.85	0.86	0.86	670
accuracy			0.86	2040
macro avg	0.87	0.86	0.86	2040
weighted avg	0.87	0.86	0.86	2040

Overall, the model resulted in a similar accuracy and precision scores from both the local machine and databricks. The only noticeable difference was the large difference in precision among the 3 class models. Class 0 had a significantly higher score of precision than that of class 1 and class 2. This inherent problem was consistent in both the local machine and databricks, and therefore can be pin point to the test-training method as it can create an imbalance set that favored class0 and in turn can cause a bias towards class0 in training the random forest models. In conclusion, in terms of results, there was not a noticeable difference between that of the models from databricks and local machine. However, in terms of speed and performance, the databricks platform significantly out perform the local models as databricks was able to train the random forest model in an efficient amount of time on a full set of data. The Databricks distributed system handled large sets of data and complex machine learning model well in comparison to the local machine through its speed and efficiency.