Exploring the Ability of Machine Learning Engineers to Predict the Best-Performing Machine Learning Model for the OLIVEs Data Set

Authors:

Garrett C Botkin and Jaime E Castro Cuevas

ECE 8803 – FunML – Spring 2023

Georgia Tech

An Introduction

- A large part of any machine learning problem is selecting a good model that is applicable to the problem
- Selecting this model is so important that machine learning models are being developed to predict which models would perform the best for a specific dataset
- However, papers exploring the ability of machine learning engineers' ability to predict the best model for a data set has been sparse so far
- Our paper seeks to explore the ability of machine learning engineers to predict the best model out of four for the OLIVEs data set

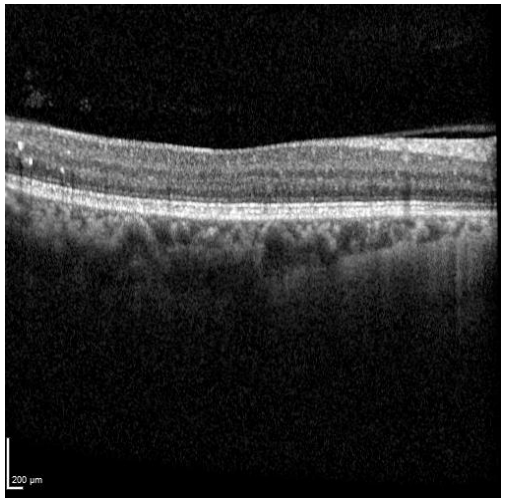
Visualization and Models

- To aid the machine learning engineers in predicting the best model, the OLIVEs data set is visualized through three methods: a 3d scatter plot, a kernel density estimator, and a pair plot

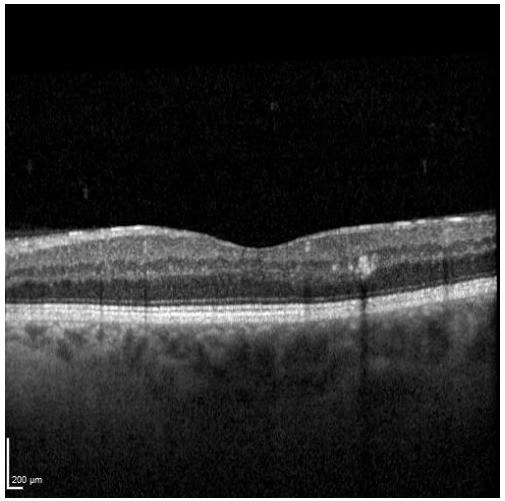
- Machine learning engineers examine each plot to determine if any hints within them existed to aid in predicting the best model

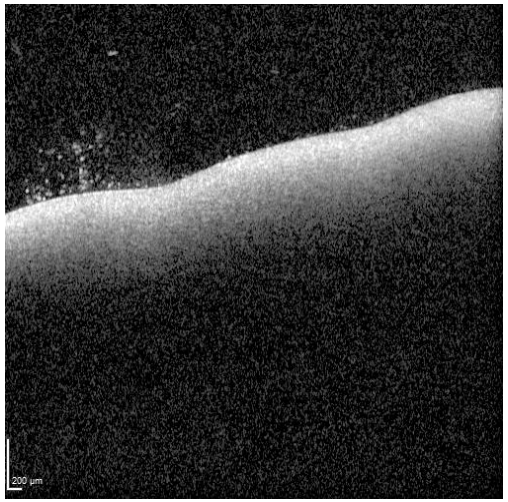
- The machine learning engineers select the best model from the four provided choices: K-Nearest Neighbors, Decision Tree, Gaussian Naïve Bayes, and Convolutional Neural Network.

The OLIVEs Dataset (Provided)



- 32,340 Samples of Medical Images
- Each sample is 504 x 496 Pixels
- Each sample is labeled with eight scores from the Diabetic Retinopathy Severity Scale.





- Each samples has metadata within the images that describe the patient in terms of age, gender, race, diabetes type, years as a diabetic, body mass index, best corrected visual acuity, and central sub field thickness.

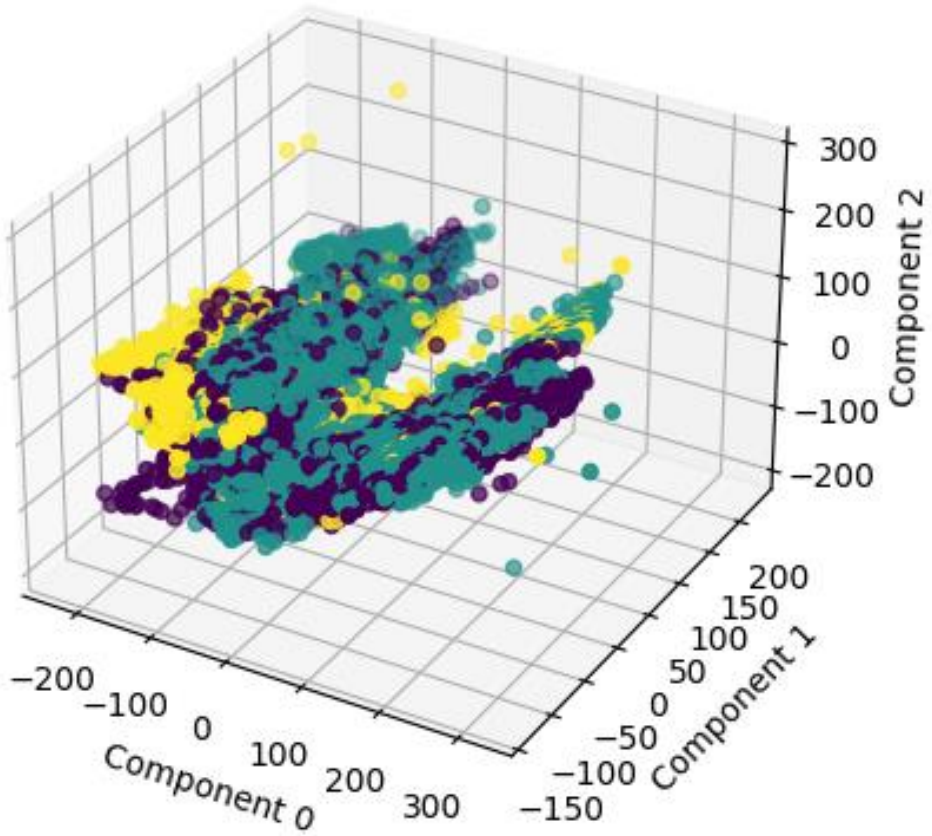
The OLIVEs Dataset (Transformed)

- Samples are relabeled with three severity levels based on the DRSS score: low (35, 43), medium (47, 53), and high severity (61, 65, 71, 85)
- Samples are center-cropped to 100 by 496 pixels
- Pixels are scaled to a range of 0 to 1 and normalized with a mean of 0.1706 and standard deviation of 0.2112
- Samples are splits into a train (19,403), a validation (4,851), and a test (7,987) data set.
- Samples' features are reduced using principal component analysis to three components

Visualization (The Scatter Plot)

- Small linear separation between label 0 and labels 1 and 2

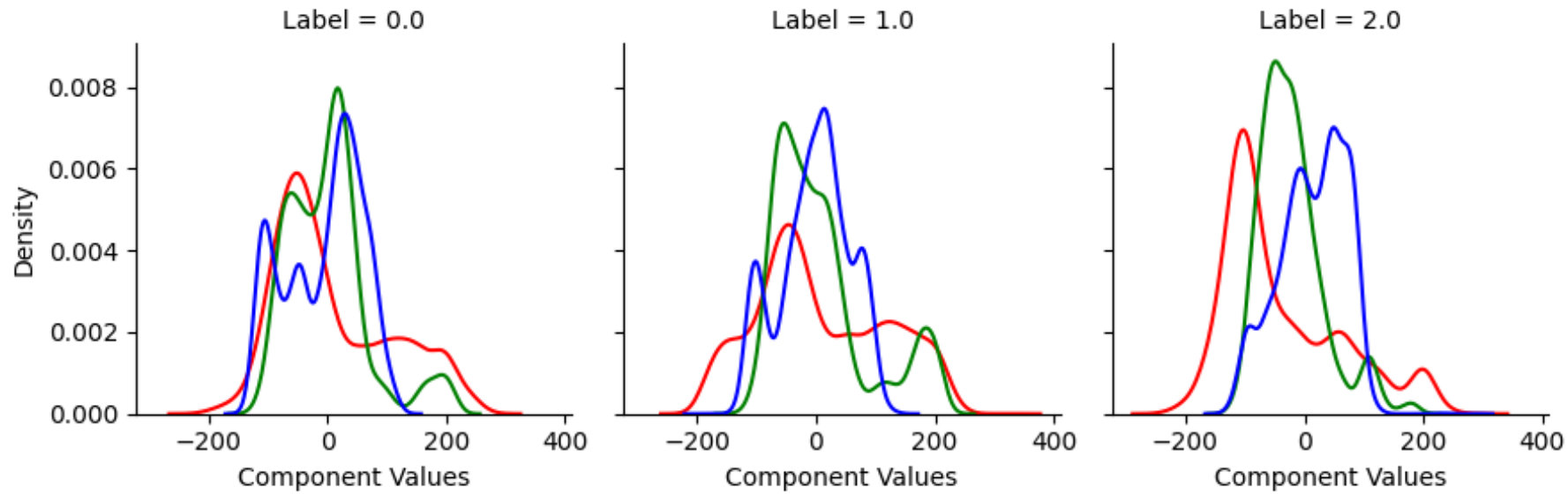
- Machine learning engineers believe that this gives a small edge to the decision tree and CNN models compared to other models



Visualization (Kernal Density Estimator)

- Similar distribution among the three labels

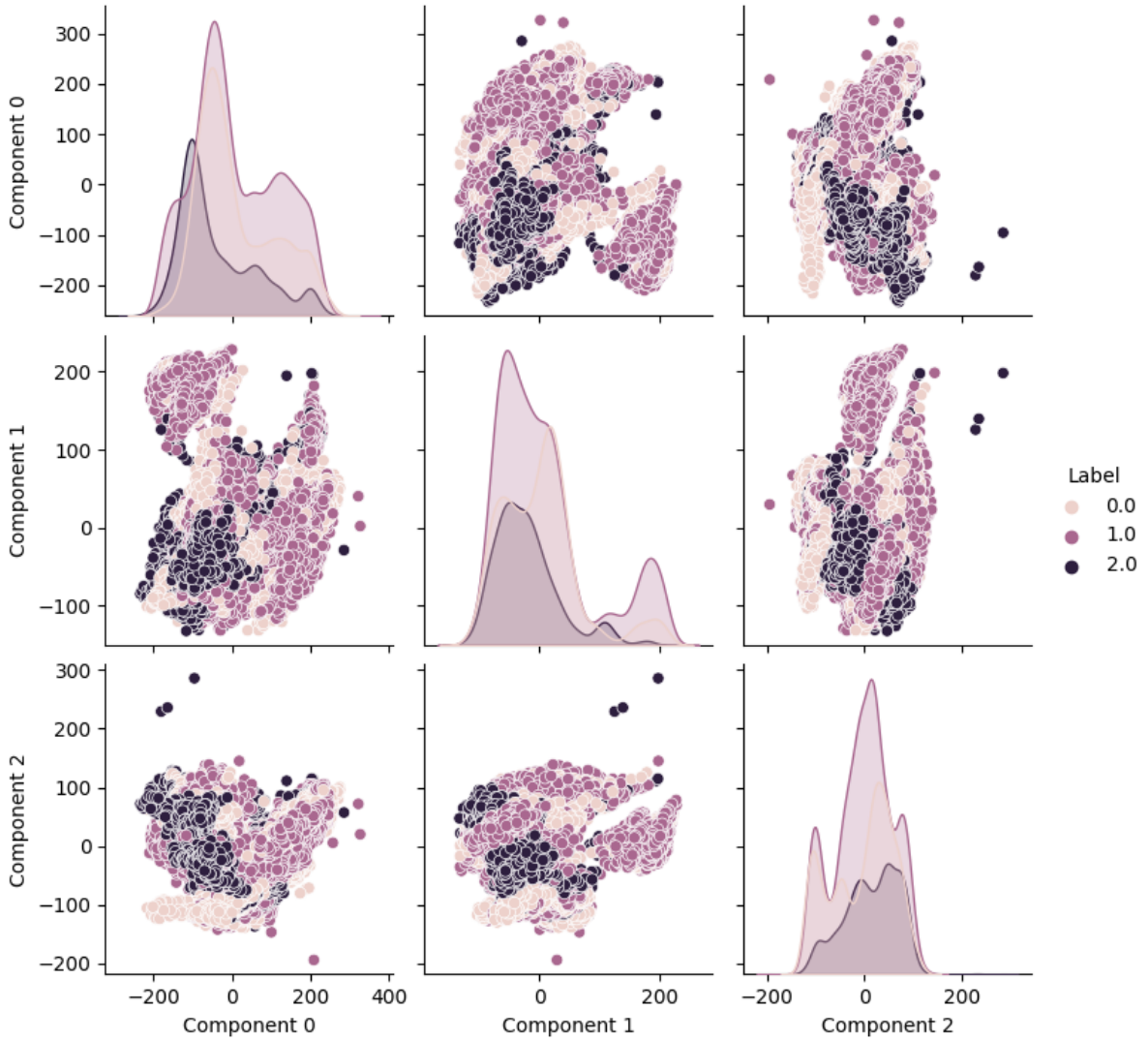
- Machine learning engineers believe that this would results in the Gaussian naïve bayes model to perform worse than the other three models



Visualization (Pair Plot)

- Does little to help machine learning engineers with narrowing down which model would perform the best.

- Machine engineers are unable to draw any conclusion about which model would perform the best from the pair plot



The Models (K-Nearest Neighbor)

- Is trained upon the PCA-reduced train data set
- "K" number of nearest neighbors is set to 50 neighbors and is experimentally determined through maximizing balanced accuracy with different k-values
- Minkowski distant metric with a power of two is used
- Predicted to perform the 2nd best

Model	Balanced Accuracy	Precision	Recall	F1
KNN	0.3081	0.3663	0.3857	0.3585

The Models (Decision Tree)

- Is trained upon the PCA-reduced train data set
- Gini Impurity Index is used to determine node splitting criteria
- Samples are equally weight, which prevents child nodes with zero or negative weight from being created
- Predicted to perform the 2nd worst

Model	Balanced Accuracy	Precision	Recall	F1
Decision Tree	0.3042	0.3115	0.4317	0.3330

The Models (Gaussian Naïve Bayes)

- Is trained upon the PCA-reduced train data set
- Setup without individual weights for samples
- Predicted to perform the worst

Model	Balanced Accuracy	Precision	Recall	F1
Gaussian NB	0.3359	0.3194	0.4674	0.3549

The Models (CNN)

- Is trained upon the non-PCA-reduced train data set
- Based upon the LeNet-5 Architecture, adapted for use with the OLIVEs dataset
- Uses Focal Cross Entropy Loss and Adam optimizer for training
- Predicted to perform the best

Layer	Layer Type	Operations	# of filters / Neurons	Filter size	Size of Feature map	Activation Function
Input	Input	-	-	-	100 x 496 x 1	-
C1	Convolutional	-	8	3 x 3	98 x 494 x 8	ReLU
S2	Sub-sampling	2D Max Pooling	-	4 x 4	24 x 123 x 8	-
C3	Convolutional	-	16	3 x 3	22 x 121 x 16	ReLU
S4	Sub-sampling	2D Max Pooling	-	2 x 2	11 x 60 x 16	-
F5	Fully connected	Flattening + Dropout	120	-	120	ReLU
F6	Fully connected	-	84	-	84	ReLU
Output	Fully connected	-	3	-	3	ReLU

Model	Balanced Accuracy	Precision	Recall	F1
CNN	0.4530	0.6686	0.3492	0.2356

Conclusion

- The machine learning engineers predict model performance as:
  - CNN > KNN > Decision Tree > Gaussian Naïve Bayes
- Actual model performance is:
  - CNN > Gaussian Naïve Bayes > KNN = Decision Tree

Model	Balanced Accuracy	Precision	Recall	F1
CNN	0.4530	0.6686	0.3492	0.2356
Gaussian NB	0.3359	0.3194	0.4674	0.3549
KNN	0.3081	0.3663	0.3857	0.3585
Decision Tree	0.3042	0.3115	0.4317	0.3330

- Only the best model was correctly guessed
- This implies that that while it is possible for machine learning engineers to predict the best performing model amongst a given group of machine learning models after a careful evaluation of the data set using visualization techniques, it is difficult to predict which models of similar complexity (such as KNN, Decision Tree and Gaussian NB) will perform best.