UC Berkeley MEng IEOR - FinTech Concentration

Final Capstone Project 2019

FinTech Group 3

Yue Hu
Chien-Hsun Huang
Ming Yu Chi
Nicolas Kardous
Yao-chieh HU

Introduction

This report serves as a solution proposal toward provision of a feasible approach for detection and diagnosis of diseases, especially, Diabetic Retinopathy (DR), though the similar contexts of visional categorization could be employed with the exact proposal of this report.

Impact and Severity

Diabetic Retinopathy presented itself as a core cause of blindness across the globe, specifically in east asian countries. The population who is suffered from the disease of diabetes and its complications are growing from 425 million (2017) to 629 million (2045), with a 48% surge within three decades. At the time of drafting this report, the population of Diabetic Retinopathy has reached 285 million, which is around 0.36% of the world population of 7.8 billion. Since people having diabetes are prone to be inflected with Diabetic Retinopathy compared to the general public. In th report of NCBI published in 2012, the prediction of the growth is set at 191 million by 2013, however, the number has been far surpassed, with a nearly 150% out of expectation within only a third of the time span given. Diabetic Retinopathy would definitely impose a negative impact to our future.

Visional Damage

Blood vessels inside the retina has high chance to be impaired by the threat of rising blood sugar levels of a patient, which can easily bleed and balloon in the scenario. The vessels could also possibly clog up to stop the flow of blood around the vicinity of retina. Moreover, the anomalous vessels could grow all over the retina to lead to even severe problems of vision.

Prevention and Treatment

Patients go through periodic inspection to conduct potential early detection and treatment on Diabetic Retinopathy, lessening the risk of blindness by 95 percent at the least.

Stages: Non-Proliferative and Proliferative

In particular, there are four stages of the development of Diabetic Retinopathy, starting with the stage of Mild Nonproliferative Retinopathy and ending with Proliferative Diabetic Retinopathy (PDR). For the first three stages of non-proliferative phases, patients will be suffered from a gradual change from mild, moderate, to severe, in regards of the vessels' state. In the beginning, several minor regions adjacent to the retina would have vessels start to sweet and swell. As the inflammation advancing, these vessels around retina might disform and grow in sizes. At the last stage of non-proliferative, some vessels begin to obstruct the flow and create shortage of blood supply within the area.

In the final stage, or specifically, the proliferative stage of Diabetic Retinopathy, the growth factor of retina may be triggered and causes a lot of new vessels to activate and grow. Those newly formed vessels can drill into the retina or cover the entire area of vision. Considering that the vessels might be frail and easily damaged, the broken part would bleed and leak fluid to impact the original functionality of the retina, leading to macular edema. More severely, the reformed

tissue of the vessels could variate and migrate to detach retina. As long as the retina are no longer attach to its adjoining flesh, the patient would suffer from permanent loss of vision.

Problem Description

This report aims at solving the problem of Diabetic Retinopathy stage detection with machine learning techniques, particularly through CNN and SVM. The conventional approach relies on manually skimming through the visual data of patients' eyes until a symptom of potential Diabetic Retinopathy has been recognized. Since the stages are divided into four stages, it is evident that human might find it hard to tell the subtle difference between stages. As the clarification given in Susan Ruyu Qi's "AI in Medicine — Majority decision isn't always right", the consistency between mutiple doctors who assess the visual data is as low as 60%.

This report claims that machine learning on the visual data of patients' eyes can resolve the problem and raise the accuracy as competitive as it could be to replace human doctors and manual efforts. There will be a set of 1500 photos of patients' eye captured, with allocation in five separated categories across (1) Healthy/None, (2) Mild Non-Proliferative, (3) Moderate Non-Proliferative, (4) Severe Non-Proliferative, and (5) Proliferative Diabetic Retinopathy, each with the number of photos as 1095, 120, 229, 35, and 21. Since the distribution of the photos across categories are not even, there could be additional works need doing on the machine learning data preprocessing to get rid of the factors of insufficient data provision or unbalanced.

Approach Description

According to the thesis, *Angelos Filos et al*. " ***Benchmarking Bayesian Deep Learning with Diabetic Retinopathy Diagnosis***", we convert the dependent variable (stage label of symptom) of images into 0 (stage 0 and stage 1) and 1 (stage 2, stage 3 and stage 4). We then processing the independent variables. First, we reshape each image into a size of 512 * 512 because all images are not uniform in size. Then, since the dimension of each observation is high (512 * 512 * 3), we first do the unsupervised PCA to transform the independent variables (the pixel array with 3 channels) of the data set into a set of data composed with only 100 dimension with a 98.42% of explained variance ratio. Lastly, we use the neural network approach with our PCA independent variables to train our model instead of the traditional convolutional neural network approach.

On the other hand, in the NN with PCA model, we have bootstrapped our training set due to the imbalance in the dependent variable. We reconstructed the training set with 600 images with DR disease and 600 images without disease. If we do not reconstruct the training set, the model will have an inclination to predict the images as 0 (no disease) on the test set because we use accuracy as our metrics when training the model and the majority of the original training set consists of 0 (no disease). Therefore, without reconstruction, the non-disease images group will have more weight on the training process and model due to the higher quantity of observations.

Models

1. Support Vector Machine (SVM):

After the feature extraction using PCA method to deal with dimension reduction, the extracted features of images are given as inputs to Support vector machines(SVM). SVMs are a set of related supervised learning methods used for classification and regression, and performs classification by constructing an N--dimensional hyperplane that optimally separates the data into two categories.

In this project, we fit an estimator to be able to predict the classes to which unseen samples belong. The estimator we use is the class sklearn.svm.SVC that implements SVM, and we set the value of gamma manually using tools such as grid search. The classifier is done by passing our training set to the fit method, and thus predict the values by determining the image from the training set that best matches.

Based on the output, it shows that we could obtain 66% of the overall predicted outputs that were correctly classified, and the confusion matrix also allows us to have a better picture of the performance of the model.

2. Convolutional Neural Networks (CNN)

We have to use multi layer neural networks as CNN. Due to the fact that our data's structure is Image, the best type of neural network satisfying our goal is Convolutional Neural Networks. After doing preprocessing and normalizing, prepared dataset could be used as input of our deep convolutional neural networks. Then CNN will be run and fit to our data and the result will be produced by that. This report will cover step by step how this deep convolutional network be implemented.

This neural network contained 6 hidden layers which will be described. The layer we used is Convolutional2D network with kernel size (3,3) and activation function relu. After preprocessing data ,our input shape is going to be (512,512, 3), and then we use MaxPooling to combine important features then Flatten and then dropout. At the end Dense is used because we have binary classes [is diabetic or not]. We also use adam as our optimizer to. batch_size is 100 to update the weights in batch mode. This may prevent our model from overfitting.

After doing all the steps above, we fit and run then by passing test data. Results were given that testing accuracy was 0.716, and will describe the prediction in the following section.

```
scores = model.evaluate(test_images_arr, y_testOneHot)
scores[1]
```

```
300/300 [==============================] - 1s 2ms/step
0.7166666658719381
```

3. Neural Networks (NN) - With PCA

We also tried using PCA to do dimension reduction to further improve our accuracy by transforming the independent variables of the data set into a set of data composed with only 100 dimension with a 98.55% of explained variance ratio. We have also bootstrapped our training set due to the imbalance in the dependent variable. We reconstructed the training set with 600 images with DR disease and 600 images without disease.

From the output, we can see that all accuracy scores are around 0.74, which is better performing than the previous version of 0.716.

```
scores = model.evaluate(x_test_pca, y_testOneHot)
scores[1]
```

```
300/300 [==============================] - 0s 87us/step
0.7400000007947286
```

Predictions

Looking at the results, we see that the Neural Network model including PCA has the highest accuracy of 0.74. Our next best model was the convolutional neural network that doesn't implement PCA, which had an accuracy of 0.716. We believe the CNN model with implementing the PCA performed a better accuracy because it was dealing with much less features, and thus this would lead to a lower variance on the training data, and predict a better performance on the testing set. A model that takes in a lot of features has the tendency to overfit on the training set, and not perform as well on the validation or testing set. One way to combat this is to incorporate some form of regularization such as LASSO or ridge regression. However, we saw that dimensionality reduction was a good means of tackling this problem because 100 dimensions of the data accounted for 98.42% of the variance.

We see that for both NN and SVM, the models had a higher precision compared to recall when predicting for Healthy eyes. A higher precision means that our models predicted less false positives. False positives is when our model would predict positive for healthy eyes, when in actuality it should be negative. Having less false positives is a very good thing, especially in the case of medical diagnosis, because we don't run the risk of telling a patient they don't have DR, when they actually do.

With regards to NN and SVM, the models had a higher recall compared to precision when predicting for DR disease. A higher recall means that the models predicts less false negatives. False negatives is when our model predicts negative for DR disease, when in actuality it should be positive. Having less false negatives in relation to DR disease is a good thing because we are not predicting negative for DR disease when the actual result is positive.

NN with PCA	precision	recall	f1-score	support
Healthy eyes	0.83	0.85	0.84	243
DR Disease	0.29	0.26	0.28	57
accuracy			0.74	300
macro avg	0.56	0.56	0.56	300
weighted avg	0.73	0.74	0.73	300

Our SVM model has performed very well, however, Neural networks are the industry standard when it comes to image classification.

SVM with PCA	precision	recall	f1-score	support
Healthy eyes	0.83	0.73	0.78	241
Dr Disease	0.26	0.39	0.31	59
accuracy			0.66	300
macro avg	0.55	0.56	0.54	300
weighted avg	0.72	0.66	0.69	300

Improvement

Our current model is a binary classification model. Initially, we tried to classify a multi-classification model, however, it turns out that the best accuracy we can get is when the model all predicts 0. We think this problem is because of the imbalanced dataset. So we first tried to balance the dataset by resampling, but this turned out to gain only about 0.35 accuracy. And then we tried to add the ‘class_weight’ parameter in our model, the model turns out to predict all to class 3 or 4.

We then considered whether our problem is that the loss function was stuck in a local minimum, so we changed our optimizer to stochastic gradient descent with momentum.

It seemed to work a little. But after it began to classify different classes instead of all to 0, the accuracy became lower and lower.

Since our dataset is imbalanced, we should apply some other metrics and loss functions instead of the accuracy. In the next step, we plan to try the kappa loss and cohen kappa metric. As it takes into account the possibility of the agreement occurring by chance, it is a proper metric for imbalanced datasets like ours.

Moreover, due to the limited dataset and computational resources, we couldn't afford to train the CNN model thousands of times. In our PCA model, we found that the accuracy might began to increase after training over 2000 times. So in the further improvement of our model, we would likely to train our model with a more powerful GPU resource and see whether there will be a difference.

Code

SVM:

https://github.com/MichaelHuang05/test-repo/blob/master/FintechDR-PCA-SVM.ipynb

CNN:

https://github.com/MichaelHuang05/test-repo/blob/master/FintechDR_0_and_1_CNN_with_resample.ipynb

NN with PCA:

https://github.com/MichaelHuang05/test-repo/blob/master/FintechDR_PCA_CNN_with_replacement.ipynb