Diagnosing Hepatitis and Diabetes using KNN and Decision Trees

COMP 551 Mini Project 1

Chloe Mills, Shania Wan-Bok-Nale, Khoi Nguyen

February 2022

Abstract

In this project, we were tasked to implement two different classification models to predict the diagnosis of two diseases; Hepatitis and Diabetes, using two datasets containing Messidor image sets and a Hepatitis symptomatic attributes dataset. The goal is to find the model with the highest accuracy and the best pair of features. We built the KNN models using hyper-parameter settings with distance functions and normalised the features so that they would have equal weight when calculating the distance which we found could potentially impact the performance significantly. Similarly we constructed the Decision tree over hyper-parameter settings on depth, cost function and minimum leaf instances. We evaluated our models using K-fold validation techniques to ensure the consistent performance of our models. We found that our KNN approach achieved a better accuracy which reported a test accuracy score of 93.8% on the final test set in the Hepatitis dataset and a test accuracy score of 69.6% on the Messidor dataset. In comparison to the decision tree model with a test accuracy score of 68.8% for the Hepatitis dataset and a test accuracy score of 65.2% for the Messidor dataset.

Introduction

Hepatitis and Diabetes are two prevalent diseases associated with increased morbidity and mortality that affect around 422 million and 355 million people worldwide. Diabetes is a disease in which your body cannot produce insulin while Hepatitis is inflammation of the liver. Many studies have found a two way association between the two diseases where they observed that patients suffering from Hepatitis C are more likely to develop type 2 Diabetes. Hence, the task of correctly classifying the diagnosis of a patient potentially suffering from Hepatitis or Diabetes is desirable. Here we are using Messidor image sets to predict whether an image contains signs of diabetic retinopathy and the level of symptomatic attributes to predict if someone lives or dies from Hepatitis. We will use both datasets over two Machine Learning models that we implemented: KNN and Decision tree to make a diagnosis of the disease.

Dataset

The Hepatitis dataset consists of two classes: Live, Die and nineteen features. The features are Age, Sex, Steroid, Antivirals, Fatigue, Malaise, Anorexia, Big Liver, Firm Liver, Spleen Palpable, Spiders, Ascites, Varices, Bilirubin, Alk Phosphate, Sgot, Albumin, Protime, and Histology.

The dataset has 155 rows, many of which were missing data in some or all of the features. These missing features were represented by '?'. We removed the incomplete rows by importing the data into pandas.DataFrame and replacing the question marks with NaN such that we can use the function pandas.DataFrame.dropna(). Eighty rows remain without any incomplete data. Further "cleaning" needs to be performed as the features are not in the correct type in the data frame. Most of the features are of type object rather than int64 or float64 so we converted them accordingly. The values of the features are similar to boolean values(i.e 1 or 0).

The Messidor dataset consists of 1150 rows. The original file contains superfluous data that was removed before importing into pandas.DataFrame. The features were of the correct type after importing into a data frame. The features of the Messidor dataset are Quality assessment, Retinal abnormality, Ma alpha 0.5, Ma alpha 0.6, Ma alpha 0.7, Ma alpha 0.8, Ma alpha 0.9, Ma alpha 1, Exudates 8, Exudates 9, Exudates 10, Exudates 11, Exudates 12, Exudates 13, Exudates 14, Exudates 15, Distance, Diameter and AM/FM. The two classes are Contains DR: Yes, No.

Ethical concerns: In the Hepatitis dataset, Males are overrepresented in the data compared to Females. This could cause problems if medical research is being done using this data as it would be male-biased and could potentially misdiagnose female patients.

signs of DR	mean	min	max	standard deviation
yes	30.473	1	99	20.759
no	45.473	2	151	27.411

Table 1: Messidor Distribution of feature MA ALPHA 0.5

Class	mean	min	max	standard deviation
Live	66.567	0	100	22.307
Die	41.615	29	90	17.652

Table 2: Hepatitis Distribution of feature PROTIME

Table 1 & 2 outlines the mean, min value, max value and standard deviation of the features MA alpha 0.5 and protime respectively. The value is calculated based on the class each entry belongs to. Before testing, we split the data into training set and test set. The difference of class imbalance between the training and test sets should be at a minimum in order to ensure the model is trained on similar instances as the test data. To find the minimum difference of imbalance,

we found an appropriate seed by searching through a list of seeds, and comparing the difference of imbalance on the sets using each seed. By setting the seed, this helped to control randomness having an impact on the testing results.

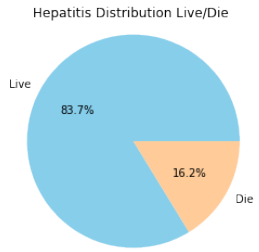


Figure 1: Hepatitis class distribution outlining if patients live or die

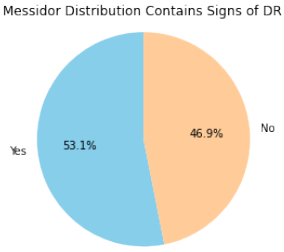


Figure 2: Messidor class distribution outlining if the entry contains signs of DR or no signs of DR

The imbalance of the Hepatitis dataset is 16.2% whereas the Messidor dataset yields an imbalance of 46.9%. The Hepatitis dataset has a much larger imbalance which suggests it could be better to use metrics such as precision or recall rather than accuracy. However, the data is not fit to calculate precision or recall.

To extract the most important features from each dataset, we used a few methods such as SHAP values (see Appendix) as well as Method 1 & Method 2 & Method 3 outlined below.

Method 1: split the nineteen features into trios. For each feature, add up the accuracy and take the mean of all trios that contain that feature.

Method 2: split the nineteen features into trios. Gather the trios that yield the highest accuracy after ten random runs. This experiment was done on both of the datasets.

Method 3: compute a correlation value table with the scalable features in order to find the correlation between every pair of scalable features. This allows us to use features that are strongly correlated. Table 3 & 4 outlines some of our results with highly correlated pair of features for each dataset respectively.

Implementation & Results for Decision Tree

Constructing the decision tree with the default hyperparameters (max depth=20, cost function=misclassification, min leaf instances=1) yields an accuracy of 63.9% for the Messidor dataset and 68.8% for the Hepatitis dataset.

feature 1	feature 2	correlation value
ALBUMIN	ALK PHOSPHATE	-0.410
BILIRUBIN	PROTIME	-0.362
ALBUMIN	BILIRUBIN	-0.344
ALK PHOSPHATE	PROTIME	-0.212
SGOT	PROTIME	-0.145
SGOT	ALBUMIN	-0.113
BILIRUBIN	SGOT	0.315
BILIRUBIN	ALK PHOSPHATE	0.317
ALK PHOSPHATE	SGOT	0.349
PROTIME	ALBUMIN	0.435

Table 3: Correlation value between some pairs of scalable features in Hepatitis dataset

feature 1	feature 2	correlation value
MA ALPHA 1	MA ALPHA 0.9	0.975
MA ALPHA 0.8	MA ALPHA 0.6	0.977
MA ALPHA 0.7	MA ALPHA 0.5	0.986
MA ALPHA 0.8	MA ALPHA 0.9	0.988
MA ALPHA 0.8	MA ALPHA 0.7	0.992
MA ALPHA 0.6	MA ALPHA 0.7	0.994
MA ALPHA 0.6	MA ALPHA 0.5	0.996

Table 4: Correlation value between some pairs of scalable features in Messidor dataset

We experimented with the maximum depth of the decision tree to see if it produces changes in the test accuracy. The Messidor dataset has a larger sample size so it was used to conduct this experiment.

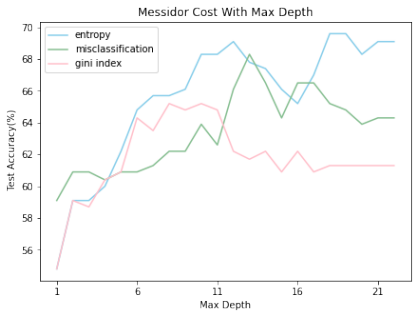


Figure 3: Messidor cost functions with maximum depth increasing. Note that The min leaf instances = 1.

Figure 3 outlines that the accuracy is low in all three of the cost functions when the maximum depth is small. As the maximum depth increases, a steady increase can be seen in the three cost functions until its peak, followed by a sharp drop. The entropy cost function has two peaks as outlined in figure 3. Upon further speculation, the tree has a maximum depth such that after it is surpassed, the tree no longer creates new branches. Thus the accuracy drops followed by a plateau. The max depth reflects under-fitting when low and over-fitting when high, hence the dip in accuracy.

Min leaf Experiment: Setting the max depth to equal nineteen (i.e. the peak accuracy), we can experiment with the

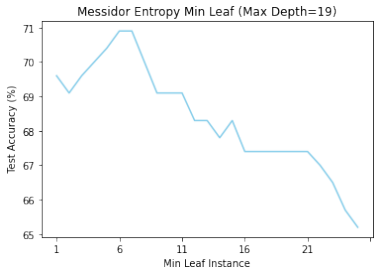


Figure 4: Messidor cost functions with maximum depth increasing

number of leaf instances to see how this affects the entropy cost of the Messidor dataset. As outlined in the figure 4, after the plot peaks, the accuracy continues to decrease as the number of min leaf instances increases.

Cross-Validation: In order to obtain the best hyper-parameters, cross validation was used. The datasets are not very large, hence cross-validation is appropriate as it supplies more training data to use with the model. 4-fold cross-validation was used for the Hepatitis dataset and 5-fold for Messidor. To select the best hyper-parameters, the mean and standard deviation of the validation accuracy's were computed.The hyper-parameters with the highest validation accuracy and the lowest standard deviation was selected, and tie breaks were implemented by choosing the lower max depth value or the lower number of min leaf instances. The data was split 64 (training) - 16 (validation) - 20 (test) in the 5-fold cross validation. By using the hyper-parameters found by the cross-validation for each of the data sets, we obtained the following plots.

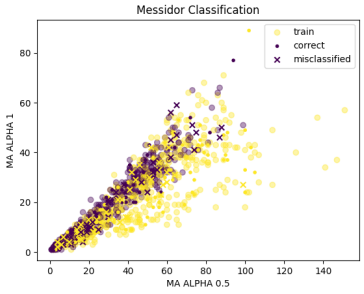


Figure 5: Decision Tree Messidor cross validation best features

The Messidor dataset's accuracy increased only slightly after using the best hyper-parameters found by cross-validation. A possible explanation is that we are taking the mean across all folds, so a hyper-parameter that performs well in one fold, could perform poorly in others, and thus it is not chosen by the model.

The experiments conducted so far, brings us to testing every combination of the maximum depth from [1,30] with every minimum leaf instance value from [1,15].

Our results are outlined in figure 7 and figure 8. They

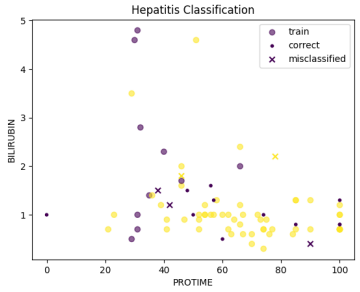


Figure 6: Decision Tree Hepatitis cross validation best features

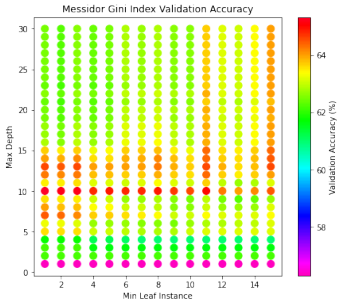


Figure 7: Messidor gini-index cost with max depth and min leaf instances

both use scatter plots with heat-map aspects such that the colour indicates whether the validation accuracy went up or down with the associated maximum depth and minimum leaf value.

The Messidor dataset's highest validation accuracy is 65.3% with standard deviation of 2.80, and test accuracy of 65.2%. To achieve this value it used gini-index with max depth = 10 and min leaf instances = 1. The Hepatitis dataset's highest validation accuracy is 87.5% with standard deviation of 7.65 and test accuracy of 68.8%. It also used the gini-index as the cost function with max depth = 3 and min leaf instances = 7. Note that in Figure 8, the Hepatitis data set has high validation accuracy for the majority of the cases where the minimum leaf instances is set to 7. Even accounting for the different values for depth, the validation accuracy is around 87% for that column. In Figure 7, we observe a similar situation in the Messidor data set, but with its highest validation accuracy being achieved when the max depth is 10, not changing very much until the minimum number of leaf instances reaches 11+.

The differences here spark many questions, such as why does the Messidor dataset experience higher accuracy related more to the maximum depth in contrast to the Hepatitis dataset which has higher accuracy related to the minimum number of leaf instances?

Although the Hepatitis dataset has a higher validation accuracy, its standard deviation is much larger. This could be a result of the dataset being much smaller and hence containing a larger imbalance in the classes compared to the

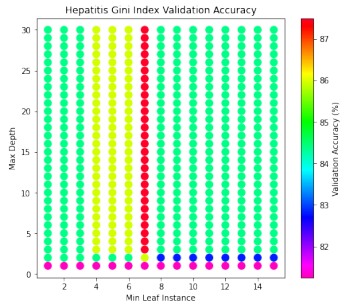


Figure 8: Hepatitis gini-index cost with max depth and min leaf instances

Messidor dataset.

Decision boundary for decision tree:

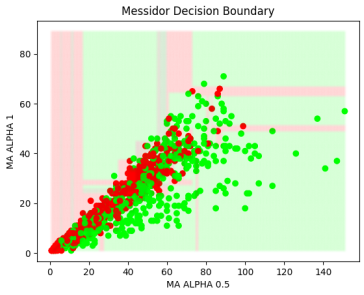


Figure 9: Messidor decision boundary

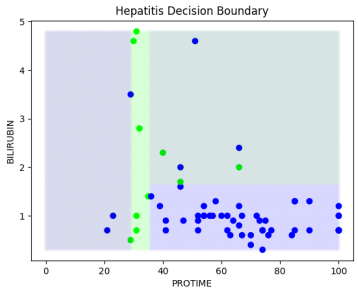


Figure 10: Hepatitis decision boundary

Implementation & Results for K Nearest Neighbours

Before running our KNN model, we normalised the features such that they have equal weight when calculating the distances between points. We did so by multiplying 'BILIRUBIN' and 'SGOT' in the hepatitis dataset. The features 'MA ALPHA' and 'EXUDATES' over the Messidor dataset, did not need normalisation as they are already appropriately scaled.

Constructing the KNN model with the default hyper-parameters with K = 3 and distance function = Euclidean distance, we got 87.5% accuracy on the Hepatitis dataset and 72.5% accuracy on the Messidor dataset. We wanted to experiment with the value of K nearest neighbours to see how it would affect the accuracy. Similar to approaches used in the decision tree model, we run KNN with the Messidor dataset over different values of K over two different cost functions of distance: Euclidean distance and Manhattan distance.

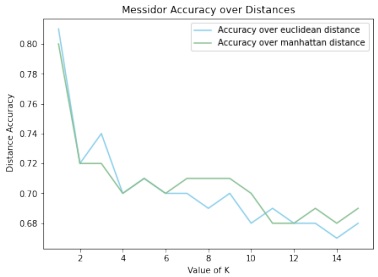


Figure 11: Messidor accuracy with different distances with increasing parameter K

Figure 11, shows that the accuracy peaks at K = 1 followed by a sharp drop. However, we observe that increasing K to a larger value does not increase the accuracy compared to the lower values of K. We suspect that this is the result from the highly correlated nature of the Messidor features as well as the nature of the train-test split. In general, small K values produce under-fitting and large values of K lead to over-fitting.

Cross-Validation for KNN: To find the best hyper-parameters for each dataset, we performed cross validation. We used 4-fold for the Hepatitis dataset and Messidor dataset over our KNN model. After experimenting with different values of K outlined in Figure 11, we opted to test every single combination of K values from [1,15] with each distance function.

We split the data 60 (training) - 20 (validation) - 20 (test) in 4-fold cross validation for each data set.

The mean and standard deviation of the accuracies obtained from running the KNN model was used to select the best hyper-parameters. We chose the value of K with the highest ratio of mean validation accuracy to lowest standard deviation of validation accuracy over the 4 fold. While doing so, we also choose the best pair of features that give the highest accuracy over the chosen hyper-parameters.

By using the hyper-parameters found by the cross-validation for each of the data sets, we obtained the following results.

For the Hepatitis dataset, we obtained the highest mean validation accuracy = 81.2% and the standard deviation equal = 2.71 over 4 fold. The hyper-parameters used were Euclidean distance function and K = 2. After testing different combinations of pairs of features, the best pair of features = (SGOT,PROTIME) and the test accuracy is 93.8%. See Figure 12 and 13

For the Messidor dataset we obtained the highest mean validation accuracy = 66.0% and the standard deviation equal = 0.65 over 4 fold. The best hyper-parameters is the

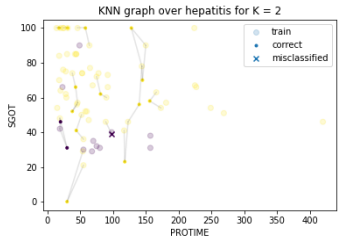


Figure 12: KNN Hepatitis cross-validation over best features

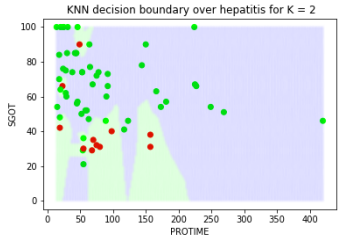


Figure 13: Decision boundary on Hepatitis best features

Manhattan distance function and K = 7. After testing different combinations of pairs of features, the best pair of features = (MA ALPHA 0.5, MA ALPHA 0.8). and the test accuracy is 69.6%. See Figure 14 and 15

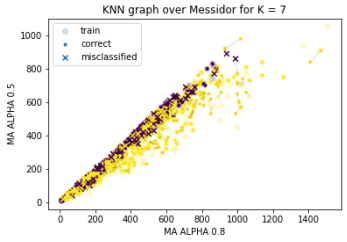


Figure 14: KNN Messidor cross-validation best features

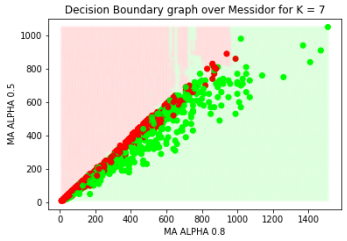


Figure 15: Decision boundary on Messidor best features

Discussion and Conclusion

In this project, we classified Diabetes and Hepatitis diagnosis using KNN and Decision Tree ML model. We examine some points worth discussing. In manually finding the best features in each data-set, the top features were the same as the features found using SHAP values, but we noticed a discrepancy in some of the others. We also tried this experiment using the best three features but noticed similar results as the test with two features. We came to the conclusion that the difference could be a result of the SHAP values using all nineteen feature simultaneously to predict, while our experiments only used two or three features at a time.

Nearest neighbours are affected by the existence of noise and irrelevant features. We added noise and saw its effects on the accuracy of our KNN classifier. See Appendix for our result. We observe that there is a gradual but steady decrease in accuracy as we scale up the noise.

After splitting the data into training, validation, and test, we examined the possibility that the tests might not be representative of the data in our splitting. The difference in the imbalance of classes were equally split into the training and test set, but the difference in the imbalance of features were split randomly. This could explain why the test accuracy did not improve as much when using the best hyper-parameters found during cross-validation.

We find that since our models are trained on small datasets, it is not very reliable. In the Hepatitis dataset, it uses less than eighty instances to train the model and less than twenty for validation. Thus we come to the conclusion that our model is not reliable for diagnosis for Hepatitis. Working on a larger dataset would make our model more reliable and our scores more accurate.

That being said, KNN model achieves a better accuracy than the decision tree model. KNN achieves test accuracy score of 93.8% in the Hepatitis dataset and a test accuracy score of 69.6% on the Messidor dataset. On the other hand, the decision tree model achieves test accuracy score of 68.8% for the Hepatitis dataset and a test accuracy score of 65.2% for the Messidor dataset.

As for future investigation, diagnosing diabetes patient from Messidor images using a Convolutional Neural Networks as a Machine Learning Model could be a possible solution. This CNN model is based on image classification.

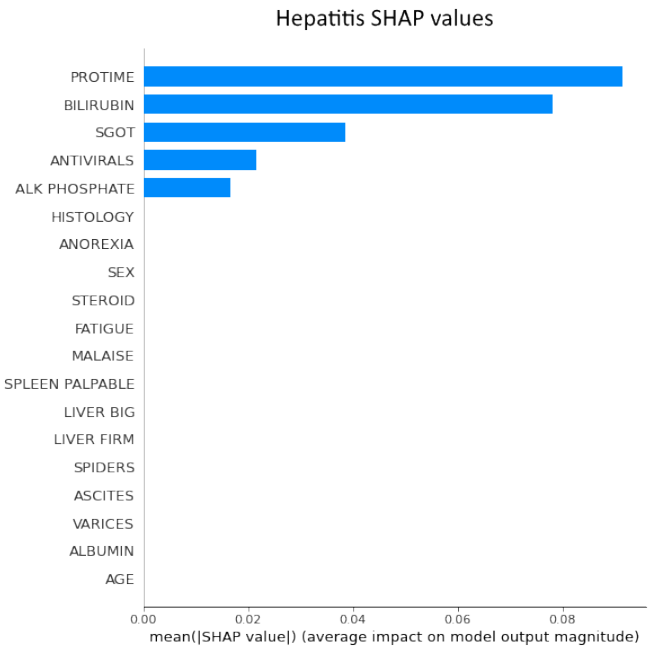
Statement of Contributions

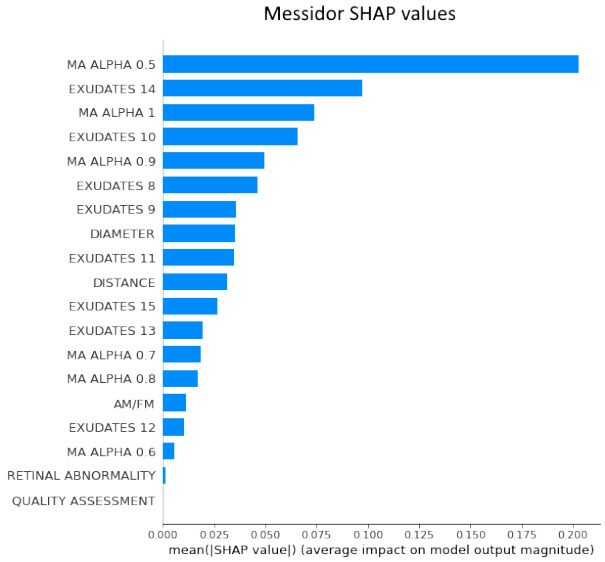
All three members of the group contributed equally to this project, we even had a fun time together and it was a great learning curve for all three of us. Nguyen worked on the Decision Tree model while Wan-Bok-Nale worked on the KNN model. Mills worked on the analysis and format of the datasets and results. We all contributed to the write up of the report of our respective work.

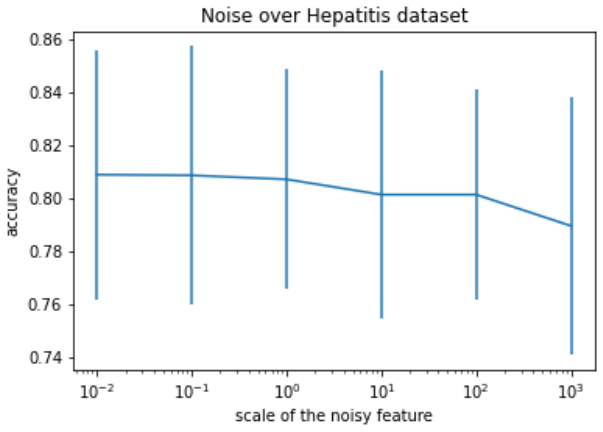
Reference

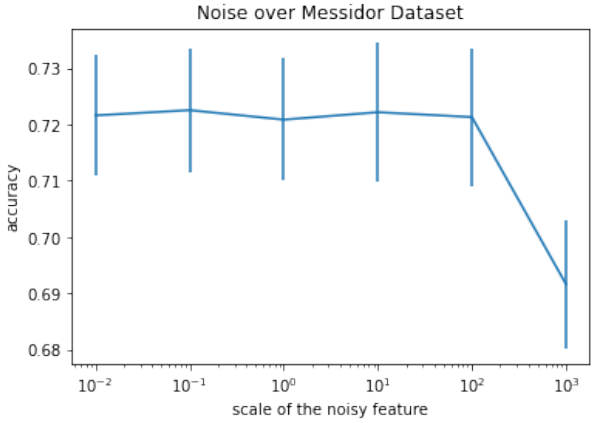
SimplicitySimplicity 1, Anthony EbertAnthony Ebert 1, IngmarIngmar 4, David CrosswellDavid Crosswell 34111 silver badge22 bronze badges, meduzmeduz 1, & NorbertNorbert 1. (1961, November 1). Putting two images beside each other. TeX. Retrieved February 8, 2022, from https://tex.stackexchange.com/questions/148438/putting-two-images-beside-each-other

Appendix









feature 1	feature 2	feature 3	test accuracy
MA ALPHA 0.5	MA ALPHA 0.8	MA ALPHA 0.9	0.678
MA ALPHA 0.5	MA ALPHA 0.8	EXUDATES 14	0.673
MA ALPHA 0.5	MA ALPHA 0.8	EXUDATES 15	0.670
MA ALPHA 0.5	MA ALPHA 0.8	AM/FM	0.670
MA ALPHA 0.5	MA ALPHA 0.6	AM/FM	0.667
QUALITY ASSESSMENT	MA ALPHA 0.5	MA ALPHA 0.8	0.666
MA ALPHA 0.5	MA ALPHA 0.8	EXUDATES 8	0.666
MA ALPHA 0.5	MA ALPHA 0.6	MA ALPHA 0.9	0.663
RETINAL ABNORMALITY	MA ALPHA 0.5	MA ALPHA 0.7	0.663
QUALITY ASSESSMENT	MA ALPHA 0.5	MA ALPHA 0.7	0.661

Table 5: Method 2: best trio of features by test accuracy