# Theory & Calculations:

The images used in the experiments were gathered using the Macular retinal images analysis tool (MRIA). Multiple images were taken of the fundus on multiple occasions. Only images of satisfactory quality were kept, images were removed if the MRIA picture changed dramatically between patient visits or if there was no consistency between the images, i.e. the patient couldn’t focus whilst the image was being taken. Each image is paired with a class that it fits into 1. Under 50 and healthy, 2. Over 50 and presumed healthy, 3. Over 50 with Macular Degeneration (MD). All images were in the form of an 81x81 pixel matrix with all values between 0 and 1 representing the macular pigment distribution.

Zernike polynomials are used as a basis function within this research. There are two types of function, even and odd. It is even when m is positive and odd when m is negative. They are formulated slightly differently.

Even:

Odd:

The variable p represents the radial distance where 0 ≤ p ≤ 1 and is the azimuthal angle. The radial polynomial is and is defined as follows.

Only certain combinations of integers produce valid basis functions. This is either when m and n are both even or where m odd and n is even. Using m and n values from 1 to 13 provided 105 basis functions which are the limit of what is used in this research.

A computer model was developed that produced a Zernike polynomial coefficient vector that represented the macular pigment data. It is formulated as an optimisation problem that finds the minimal error when the model is fitted to the data (Image). Using a combination of least square fitting and Gaussian elimination, and limiting the number of coefficients to 105, an optimal solution to the problem is found. Each element of the coefficient vector is paired with a specific Zernike basis function. This can be seen as a dimensionally reduced representation of the original image. Each Zernike basis function & coefficient combination represents features of the image that can be compared.

To reproduce the original image, perform:

* Equate where m and n represent the Zernike number matched with coefficient ‘I’.
* Normalise between 0 and 1.

will be a close approximation to the original image due a to degree of error being introduced due the optimisation technique.

Four techniques were developed to build classifiers for this problem: K-Nearest-Neighbours (KNN) (average error and 3-prioritised variant), Support Vector Machines (SVM), Softmax Layer Neural Network (SMNN), and Pattern Recognition Feedforward Neural Network (PRNN).

KNN is a technique that uses a distance function to calculate the neighbours of a given point and then defines the class of that point depending on the classes of its k nearest neighbours. In this research, it is one of the techniques used to test whether the use of only certain coefficients can produce better results. This is done by using the only the first 15 coefficients in the training stage and then using the powerset operator to compute every combination of coefficients. All combinations are then tested by the training data to see the efficiency of each combination. The most effective solution is kept and used and trained classifier model. When the testing data is used and has been converted into its coefficients only the coefficients that were identified as optimal in the training step are used to classify each test image.

Similar to KNN, SVM is used to test different combinations of coefficients. SVM constructs a hyperplane or set of hyperplanes in high dimensional space that can be used to separate the data. The difference in this classifier is that is can only separate between 2 classes at a time. The powerset operator is used in the same way as it is in KNN to get the most accurate combination of coefficients. Different to KNN this uses the created hyperplane to class the new input whereas KNN uses the training data every time. The Statistics and Machine learning toolbox within Matlab was used to perform the calculations.

The SMNN is an artificial neural network where the final layer consists of the softmax function. It can be trained using the cross entropy or mean squared error loss function. It will take an n-dimensional vector as an input which will be the first n coefficients of the images Zernike coefficient vector and will return a single value representing the class the image is classified into. Instead of using the powerset operator as the first 2 methods did this one is testing how many of the coefficients are required to achieve accurate results, from 15 up to 105 coefficients. The algorithm used is taken from the Matlab Neural Network toolbox.

The final method being used is PRNN is a feedforward neural network, where the number of hidden layer and hidden neurons can be specifically defined. Much like the SMNN, it can be trained using the 2 different loss functions and different numbers of coefficient available. This method also adds the control of the topology of the network. The code for the algorithm also utilises the same Matlab tool box.

For each classifier, input parameters alter and are changed to test different set ups to get the optimal setup. A hypothesis that is being investigated is that the classifiers will perform better if the images are centred around their peak MP distribution before the model calculated the Zernike Coefficient vector, thus tests are run on both centred and original image coefficient vectors. The parameters that are changed for each algorithm are defined below:

* KNN: number of neighbours used, the number of coefficients to consider, error calculation used, centred or not
* SVM: number of coefficients to consider, centred or not.
* SMNN: Loss Function, the number of coefficients to consider, centred or not.
* PRNN: Loss Function, Topology, the number of coefficients to consider. centred or not.

When testing how efficient the classifiers are there are multiple different runs of each algorithm consisting of different class combinations to satisfy different objectives. The four set ups are: Class 1 vs Class 2 vs Class 3 (basic class comparison), Class 1 & Class 2 vs Class 3 (testing to identify disease), Class 2 vs Class 3 (testing for disease in patients over 50), Class 1 vs Class 2 & Class 3 (testing to identify patients age).

In total 479 images were obtained from 90 patients with representations of each class being: Class 1 = 203 images, Class 2 = 153 images, Class 3 = 123 images. The data set is randomly distributed into 3 sets of training, validation and test data for use in building and evaluation of each of the classifier models upon every run to ensure the generalisation of the results.

# Results

*Experiment 1: Class 1 vs Class 2 vs Class 3 Classification Test Results*

Note: SVM is excluded in this experiment due to the algorithm only being able to split data into 2 separate classes.

Figure 1 demonstrates the best results for each algorithm using both the centred and the original coefficient vectors. When the images were centred before the coefficient vector was produced the error rate was consistently higher than when the original images were used. This could mean that the information that is allowing the KNN classifier to split the images is lost/less effective when the image is centred.

The error was fairly consistent between 3 of the classifiers, however, PRNN was much more effective producing an error rate of 28% (52 out of 72) misclassified. The input values that gave this solution were: Loss function = Cross Entropy, the network topology of [151,151] and the use of the first 15 coefficients. The confusion matrix seen in figure 1 is for this error rate.



Figure 1: Testing Classification between all 3 classes (top) and Confusion Matrix for PRNN bets result for Class 1 vs Class 2 vs Class 3 (Bottom).

The generalisation of the best result is demonstrated by the error being fairly evenly spread between the 3 classes shown by the fact an image of class 1 is misclassified 24% of the time, images of class 2, 24% of the time and class 3, 41% of the time. Images in class 3 are the main issue with this classifier. This is still a positive result altogether though.

All of the tests that were run using the PRNN technique and original image data produced results that were all below 35% misclassified which is lower than the best results of all the other techniques. This centred PRNN result data compared to this were consistently slightly worse giving more emphasis to the concern that important information is lost during the centring process.

A more in depth look at all the PRNN results shows that there no is little to no benefit of increasing the number of coefficients used. Specifically looking at the best error rate for the different number of coefficient: 15 = 28%, 60 = 31%, 105 = 31%, shows us that using a small number of coefficient provides the neural net enough dimensions for separating the data. Using fewer coefficients means the calculations are less complicated and the computation time will be reduced increasing the efficiency of the classifier.

*Experiment 2: Class 1 & Class 2 vs Class 3 Classification Test Results*

This experiment was performed to try and split the data purely by disease status ignoring the age of the patients.

Figure 2 shows the top results from original and centred images of each class on a receiver operation characteristic (ROC) graph to demonstrate the generalisation of each result. Having a result in the top left-hand corner would be seen as a perfect classifier.

The best results were again produced by the PRNN technique with an error rate of 14% (62 out of 72) by both original and centred images however the ROC graph demonstrates that the result using the centred image is less generalised due to the low value of specificity paired with a high sensitivity (white diamond). The result using original images, on the other hand, has much more equal values of

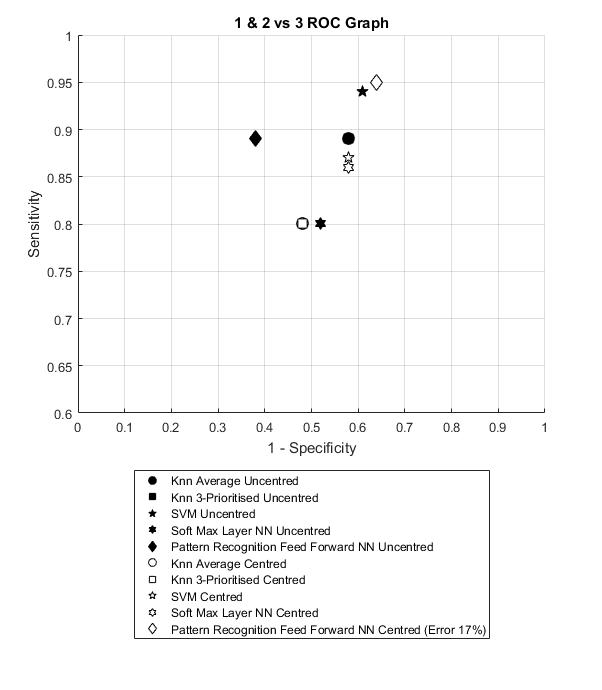
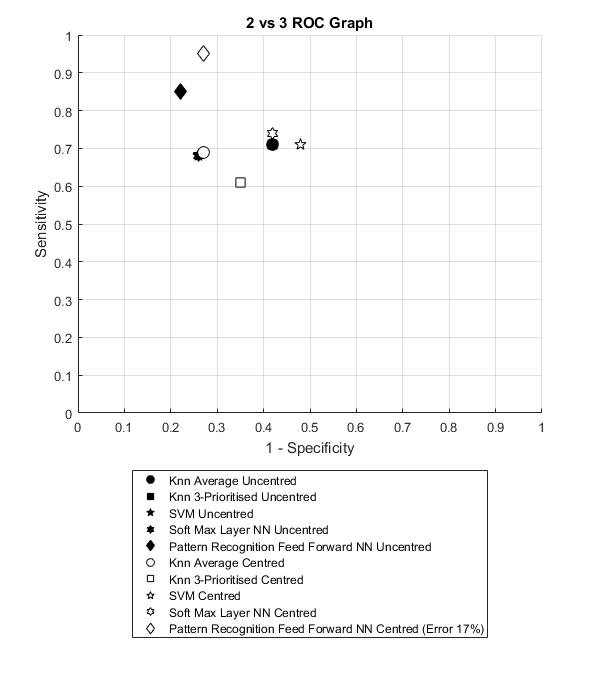
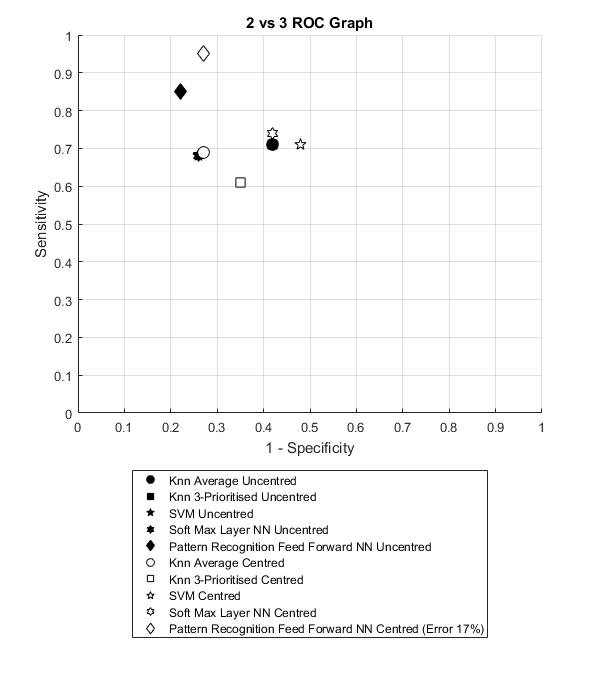


Figure 2: ROC graph for comparing results of tests run using data structure of Class 1 & 2 vs Class 3 (Left) and Class 2 vs Class 3 (Right)

sensitivity and specificity leading to the conclusion of it being the better solution of the two (black diamond). The input values that produced this result were: Loss function = Cross Entropy, available coefficients = 60, topology = [416].

Although the error rates appear to very low if the error is recalculated to take into account the size of each class the error is closer to 25% which is still promising but less accurate. The classifier still produces false negatives, at best 38% of the time which is not ideal.

Combing Class 1 and 2 created an issue when training the classifiers because now this combined class was almost 3 times bigger than the second class, this appeared to add bias to the training of the classifiers, this is evident in the fact that most of the results have high sensitivity and low specificity. This is clearly evident in the fact that the classifiers have mostly produced solutions that are overfitted to Class 1&2.



*Experiment 3: Class 2 vs Class 3 Classification Test Results*

A possible issue with the previous results that was considered was that the images taken from patients under 50 could be skewing the results so they were removed to see what effect this would have. This changed the size of the total number of images to 276 with the sizes of class 2 and 3 staying the same.

The tests run using the centred images and PRNN produced the best error of 17% (white diamond, 35 of 41), however, the when using the original images although the error rate was more at 19% (black diamond, 34 of 42) the resulting classifier was more generalised. This is shown by the position of the black diamond in the resulting ROC graph shown in figure 2.

The inputs that produced the best result were: Loss function = Cross Entropy, available coefficients = 105, topology = [424]. Similar to the observation made in the results of classifying Class 1 vs Class 2 vs Class 3 the best results for using the different values of available coefficients were the same in this case to 2sf (15 = 19%, 60 = 19%, 105 = 19%) backing up this statement.

Due to the size of data set changing the testing data set consisted of only 42 images. To improve the reliability of the result more images of these 2 classes would need to be obtained and the tests re-run using the proposed solution.

*Experiment 4: Class 1 vs Class 2 & Class 3 Classification Results*

Note: KNN 3-Prioritised results are excluded from this section due to no class 3 in test.

This test was done to test whether, irrespective of disease status, as patients age could be identified by the MRIA values. Figure 3 shows the top results of this test. The optimal error value was 13% (63 out of 72) from the PRNN using original images however the best result was 14% (62 out of 69) from the PPRN with centred images due to the fact the produced classifier was more generalised. This is visible in figure 3 by the positioning of the white six-pointed star compared to the black six-pointed star. The input values that produced this result were: loss function = Mean squared error, available coefficients = 15, topology = [361].

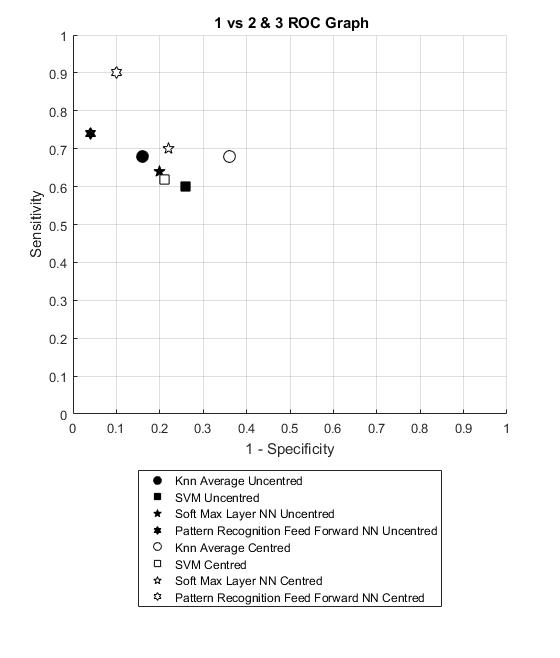


Figure 3: ROC graph for comparing results of tests run using data structure of Class 1 vs Class 2 & 3.

This was the only set of results where the centred images produced a better classifier. As the centring process focuses on the peak distribution it could me that this is more important in determining age and his harder to identify before the images are centred.

In this experiment, the observation about the number of coefficient also held as the best results for each number were: 15 = 13%, 60 = 13%, 105 = 13%. This increases the viability that it is the other values (topology & loss function) that have more of an effect on the accuracy of the model produced.

# Discussion

The success of each method can be gauged on how they performed across all the tests. Looking at the classifier models that were created using the methods KNN, SVM and SMNN was very easily skewed by the size of each data set, thus in the tests that have very different size data sets (Class 1 & 2 vs Class 3 and Class 1 vs Class 2 vs Class 3) the training error was very low because it focused on getting every member of one class correct but when using the test data, the testing error was much worse as there was a lack of generalisation in the model.

The algorithm that did consistently produce good results was the PRNN. The best error rates for each experiment were gained by testing the models produced by PRNN. The sensitivity and specificity of these results were also close to each as evident by the positioning of the points in the ROC graphs in figure 2 and 3.

A hypothesis that centring the data would improve the results were also tested and proved to be inconclusive. In 3 of the 4 experiments the best results were obtained using the original images but the 4th experiment did use the centred image. The reasoning behind this could be due to the class structure. The three classes that used the original data were involved in trying to identify diseased patients however the fourth experiment was only classifying between age. As mentioned, the centring process loses some of the data of the image, this would lead to the suggestion that the task of identifying a diseased individual requires all of the information in the outlying areas in the retinal image whereas classifying by someone’s age is able to be performed accurately using the peak macular pigment distribution value and its surrounding shape.

The observation recorded in the results that the number of Zernike coefficients in PRNN results is backed up in all four of the experiments completed. The more likely limiting factor in the topology that is used to structure the network. In the tests, the number of topologies that were tested was limited due to the time it takes to test each one. Fine tuning the network topology would be a good step to increase the accuracy of the promising results.

Results from the experiments showed that although it is possible to classify between the 3 classes in experiment 1 the models produced by experiments 2 to 4 show that the classifier models are a lot more accurate if only 2 classes are considered at a time. This is visible in the graph in figure 4

Figure 4: The optimal error values produced by PRNN for all of the experiments performed.

which shows the best results for each experiment.

Experiments 2 and 3 were focusing on identify diseased individuals. The issues raised in experiment 2 were confirmed as although the errors rates were larger (19% compared to 14%) the specificity & sensitivity values show that the generalisation of the model produced by experiment 3 was much better. Assuming age is already known meant that the classifier model would be much more effective and is more realistic. The result is promising, however, the optimal solution needs to be tested with another data set to ensure the results are accurate and that the generalisation holds.

Experiment 4 is interesting because it shows using MRIA is an accurate way of identifying a patients age. The error rate of 13% produced is very good and considering this can possibly be improved with the altering of the neural net variables it is a promising step in finding applications of the MRIA technique.

In conclusion of the 4 methods tested the Pattern Recognition neural network that was used proved to be the best companion to the Zernike coefficient representation of the retinal images. The errors that were produced for all of the experiments were very promising meaning that the use of Zernike polynomials in representing the images was able to produce an accurately separable set of data

**References**

1. Styles, I.B., Calcagni, A., Claridge, E., Orihucka-Espina, Fand Gibson, J .M. Multispectral retinal image analysis: a novel non-invasive tool for retinal imaging. www.nature.com/eye, 7, 2011
2. Eric C. Kintner, 1976. “On the Mathematical Properties of the Zernike Polynomials”. Optica Acta, vol. 23 (8): p.679-680.
3. Critianini, Nello; and Shawe-Taylor, John, 2000: “ An Introduction to Support Vector Machines and other kernel-based learning methods”, Cambridge University Press.
4. Mathworks. (2011). Statistics and *Machine Learning: Classification Support Vector Machine Classification*(r2011b). Retrieved January 10, 2017, from https://uk.mathworks.com/help/stats/fitcsvm.html
5. Mathworks. (2011).  Neural Network toolbox*: Pattern Recognition & Classification Support Vector Machine Classification*(r2011b). Retrieved July 15, 2017, from https://uk.mathworks.com/help/nnet/ref/trainsoftmaxlayer.html
6. Mathworks. (2011).  Neural Network toolbox*: Pattern Recognition & Classification Pattern Recognition Network*(r2011b). Retrieved July 15, 2017, from https://uk.mathworks.com/help/nnet/ref/patternnet.html