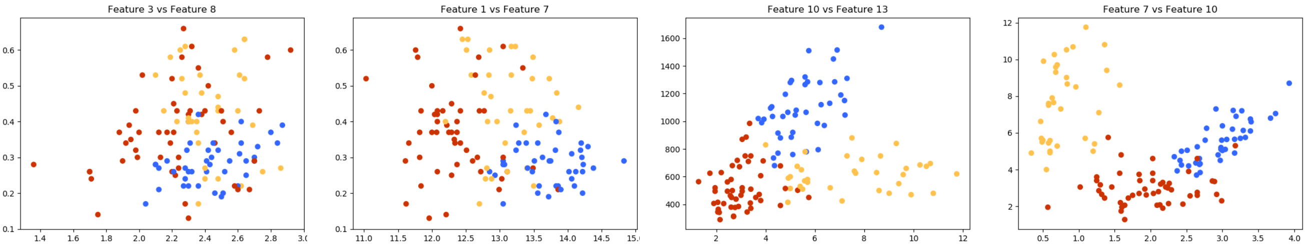
A Memory of Wine

Jack Healey - jh16031, Jamie Day - jd17070

Feature Selection

In order to classify the given set of samples into the 3 classes (defined by the three cultivars) a pair of discriminative features must be extracted. Ideally, these chosen features would separate the data into three linearly separable classes. Using plots of all the pairwise feature combinations, it can be seen on inspection that certain pairs of features separate the data better than others, therefore the task here is to chose the optimal result.



Feature 3 vs Feature 8 Feature 1 vs Feature 7 Feature 10 vs Feature 13 Feature 7 vs Feature 10

The figure above shows a variety of pairwise combination plots, strengthening in separation from left to right. The leftmost plot of feature 3 against feature 8 is an example of a combination with inadequate clustering. Here the classes overlap a great deal and it would be difficult to distinguish between the classes using the classifiers below. The middle two plots on the other hand have more distinguishable clusters, however these are weak compared to the combination of feature 7 and feature 10. There is also a large spread within each sample in both plots, therefore the classifiers would struggle to identify the class of a test sample that lies in the region between two clusters. The pair that has been chosen to be extracted here is feature 7 (Flavanoidsm) and feature 10 (Colour intensity). This is because one cluster can almost be separated linearly from the other two, where these classes overlap in only a few cases.

K-Nearest Neighbours Classifier

Initially, the dataset was split in a 70/30 proportion to create the training and testing sets. The training set can then be used to train the classifier in order to predict the class of the samples in the testing set. The K-NN algorithm takes a sample from the testing set, finds the 'K' nearest points on the feature plot and allocates the modal class of these points to this sample. If there is more than one modal class, then the average distance from the sample to the two modal classes is found, and the smallest of these is taken and the corresponding class then allocated.

Choosing K

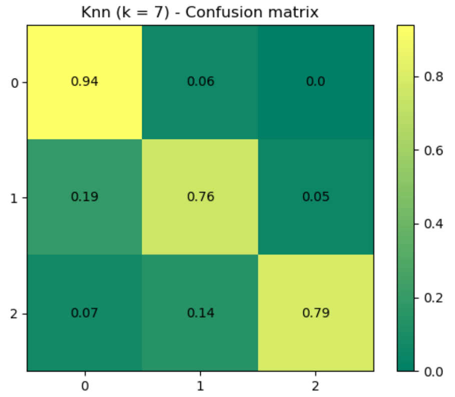
In machine learning, the optimal value of K for most datasets is between 3 and 10 when using the K-NN classifier (Ma et al. [2014]), and is highly data-dependent. There are pros and cons to smaller and larger values. In general, a small K value leads to overfitting resulting in a decision boundary that is not smooth. On the other hand, a higher K would minimise noise in the data, however this can be under-fitting and would result in a decision boundary that is imprecise. The aim in this case is to choose a value of K that is neither overfitting nor under-fitting, producing an acceptable accuracy in its results. Features 7 and 10 were chosen for their separability on inspection, therefore K does not have to be increased to take significant amounts of noise into account. To fit these criteria, $K = 7$ was selected for general use in this investigation.

The accuracy of the K-NN classifier can be seen above to vary with each K value, initially decreasing as K increases until minimum point $K = 4$ where after this, the accuracy of the classifier increases as K increases. The greatest accuracy lies where $K \in \{1, 2\}$, however it could be argued that theses results are not reliable as they are overfitted to the training set. Adding another nearest neighbour reduces the accuracy of the classifier as the decision boundary

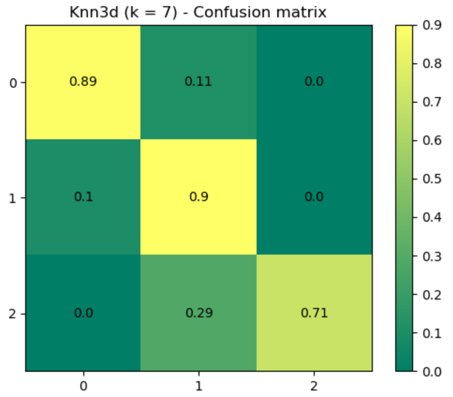
K-Nearest Neighbour	
K value	Accuracy [%]
1	90.566
2	90.566
3	79.245
4	73.585
5	79.245
7	83.019

K-Nearest Neighbour - 3D	
K value	Accuracy [%]
1	92.453
2	92.453
3	81.132
4	84.906
5	83.019
7	84.906

becomes generalised, until it reaches a large enough value ($K = 5$) where the boundary is smoother and also more precise, shown by a increase in accuracy up to $K = 7$.



Confusion Matrix for K-Nearest Neighbours where $K = 7$



Confusion Matrix where $K = 7$, using 3 features

The confusion matrices shown above have columns 0, 1, and 2 corresponding to classes 1, 2, and 3 respectively. The diagonal entries give the proportion of data points in the given class that are correctly classified. Class 1, 2, and 3 are represented by the blue, red, and orange data points respectively.
These matrices are useful as they provide accuracy's for each class in the the classifier. Comparing these values provides insight into under what circumstances the K-NN classifier is effective. Here it shows class 1 is classified very accurately with 94%, whereas the classifier only achieves 76% and 79% for class 2 and 3 respectively. Observation of the feature 7 vs 10 graph explains this. Class 1, shown by the blue points, is well separated from the orange and red points with a few points being classified as red, and none being classified as orange. However in class 2, the red points, some are closer to more orange points than than red. The same can be said for orange to blue. This gives class 2 the lowest accuracy. Class 3 has spread out points with some approaching class 2 points. This means there are more incorrectly assigned points in class 3 than class 1.

Adding a third dimension to the K-NN classifier can make it more or less accurate depending on the data. Adding a feature which is fairly inseparable may reduce the accuracy of the classifier. Therefore, careful analysis of the data is needed before the decision to add another feature can be made. In this case, feature 1 shows separable data which we thought may improve the K-NN classifier. Comparing the tabulated accuracy for $K = 7$ for 2D and 3D shows a slightly more accurate result from 3D overall. This increase is small and observation of the confusion matrix shows a reduced accuracy in classifying class 1 and 3. However, the accuracy in class 2 increases hugely. This is because class 2 sits between class 1 and 3 on the graph of feature 7 vs 10, meaning it has the lowest accuracy. Feature 1 from the the Feature 1 vs 7 graph shows class 2 is well separated from the other two. This has the effect of increasing the accuracy of class 2 classification significantly.

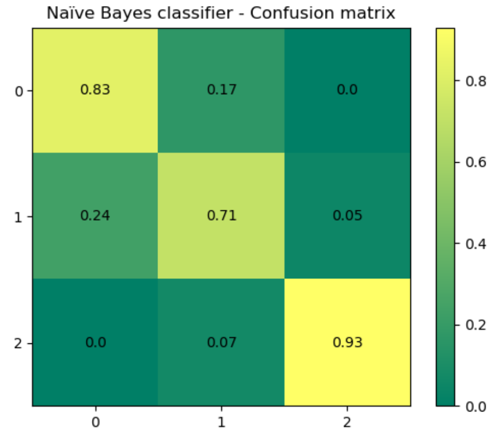
Naïve Bayes Classifier

Accuracy of Naïve Bayes classifier [%]: 81.132

To further our understanding of the data, an alternative classifier is implemented. The Naïve Bayes Classifier calculates the mean and variance of a class and applies a normal distribution to calculate the probability of a data point being in any of the three classes. The data point is assigned to the class with the highest probability. Naïve Bayes was chosen as the selected features can be assumed to be conditionally independent for a given class (Ashari et al. [2013]). This can be seen by the lack of correlation between feature 7 and 10 within each class, shown in the graph above. Furthermore, this classifier can obtain good results with relatively small amounts of data. As the test set comprises of only 50 data points, this property of the Naïve Bayes classifier makes it attractive. Moreover, this classifier can be easily implemented and will provide good results for most data sets.

The high accuracy of 81.132% obtained by the Naïve Bayes classifier is a result of considered feature selection. The features were chosen which appeared most linearly separable, greatly increasing the accuracy of the result. Additionally, the features appear to be almost completely conditionally independent from each other in two of the three classes. In the first class (blue points), there does appear to be a positive correlation between features, but the spread is very small so the accuracy is only slightly effected.

The lowest accuracy is found when classifying the red data points (class 2), as to be expected as this class sits in the middle of the parameter range. The orange data points (class 3) are classified more accurately than the blue (class 1) due to the data points in class 1 overlapping into the class 2 data points.



Classifier Results Comparison

An accuracy of 81.132 % is comparable to the K-NN classifier in two dimensions. Class 1 is classified with a 94% accuracy with the K-NN classifier, compared to 83% with Naïve Bayes. This can be explained by the spread of the data in this class. Class 1 has a cluster of points close to some of the class 2 data points. This results in some of these data points being assigned to class 2. This problem does not occur in the K-NN classifier because the spread of data is irrelevant. Those data points which are spread far out are close to others like it and so aren't classified into class 2 in almost all cases.

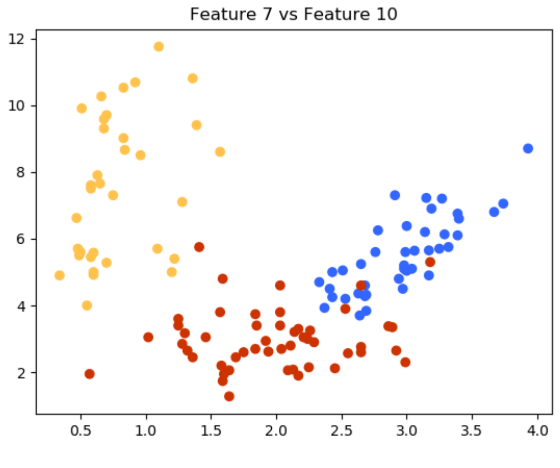
It is worth noting class 1 and 3 have no incorrect classifications into each other as the range in values of the features we selected does not make this possible.

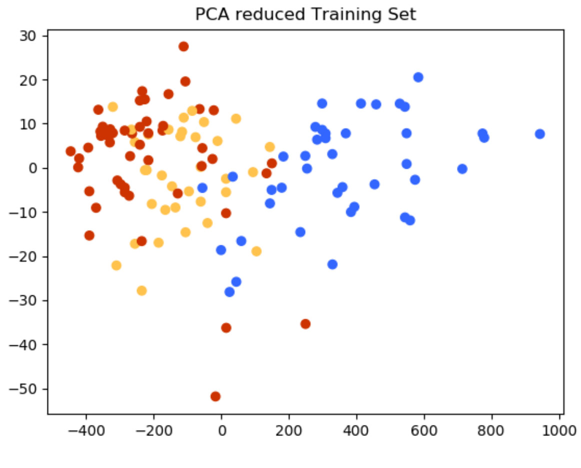
Class 2 is classified with similar accuracy for both classifiers, 76% and 71% for K-NN and Naïve Bayes respectively. This is not surprising as the data is neither tightly spread nor tightly packed. There are also no clusters of points close to another class. Therefore a roughly equal accuracy for the two classifiers is expected.

Class 3 shows a considerably higher accuracy in Naïve Bayes classification than K-NN, with 93% compared to 79%. Class 3 has a large variance; meaning, when using Naïve Bayes, far out points are still likely to be assigned to class 3. Furthermore, class 2 is fairly spread out with a high density of points being far from class 3 points. This means the probability of those class 3 points which come closest to the class 2 centroid being assigned to class 2 is reduced. As the K-NN classifier does not have holistic knowledge of the whole class, but rather only of the closest 7 points, many of the class 3 points which approach class 2 points are assigned to class 2.

A method that would be an interesting topic for further reading to improve the accuracy of the K-nearest neighbor classifier, is the idea of using local mean based and distance weight (K U Syaliman et al 2018 J. Phys.: Conf. Ser. 978 012047 [2018]).

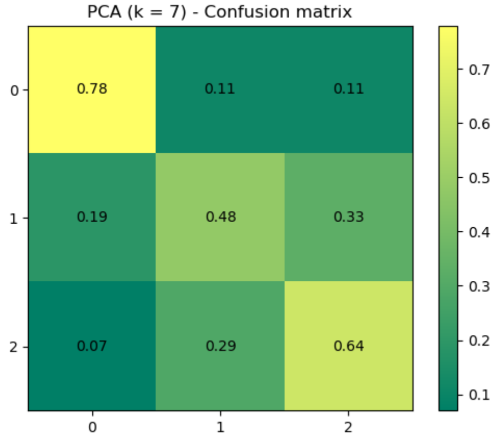
Principle Component Analysis (PCA)





Manual feature selection has been used so far in order to implement the classifiers above. The next step is to use Principal Component Analysis to reduce the dimensionality of the wine dataset and extract two features.

Principle Component Analysis	
K value	Accuracy [%]
1	71.698
2	71.698
3	64.151
4	64.151
5	64.151
7	62.264



The plots represent the training set reduced through the use of manual feature selection on the left, and the PCA reduced training set on the right. On inspection it can be seen that the three classes are separated significantly better using the manual feature selection. In the PCA reduced training set class 1 seems to be separated to an extent towards the right of the plot, whereas the other two classes overlap largely, with no obvious cluster formation.
The closer clusters lie together, the more likely it is that the nearest K neighbours would not share the same class, meaning it would be more common for classes to be mislabelled, affecting the accuracy of the classifier under a PCA reduced training set. It is expected, therefore, that the accuracy of the K-NN classifier (using K as listed above) using the PCA-transformed data would be significantly less than that of the manually reduced data, as the quality of the separation of the classes is significantly worse. This prediction is supported by the results shown in the accuracy table above, where for $K = 7$, the accuracy using a manual feature selection is 83.019% compared to the PCA's accuracy of 62.264%. Lower values of K provide better results here because the more nearest neighbours you take into account, the more likely it is that the samples further dispensed from their respective clusters are taken into account. This accuracy can be broken down further using the confusion matrix above. As class 1 is more detached from the other two classes, a good number of its samples are assigned the correct class, however as the samples approach the two integrated clusters, they are assigned to classes 2 and 3 with equal proportion, as the density of class 1 values is much lower towards the other classes. Class 3 is located in between the two other classes therefore any samples that drift away from its centroid are likely to be assigned incorrectly, where only 64% of samples where correctly assigned. Class 2, has a high density and is located on one side of the plot, however is greatly integrated with class 3, where the most dispersed values are closer to class 1, therefore only 48% of samples are correctly assigned.

References

Ahmad Ashari, Iman Paryudi, and A Min Tjoa. Performance comparison between naïve bayes, decision tree and k-nearest neighbor in searching alternative design in an energy simulation tool. https://thesai.org/Downloads/Volume4No11/Paper_5-Performance_Comparison_between_Na\unhbox\voidb@x\bgroup\let\unhbox\voidb@x\setbox\@tempboxa\hbox{\OT1\i\global\mathchardef\accent@spacefactor\spacefactor}\accent127\OT1\i\egroup\spacefactor\accent@spacefactorve_Bayes.pdf?fbclid=IwAR1KqBhtwpr0gPiwUFn4oXOSKk9FVEsyegPPS_EXkiHGJXQ5uqEd3vqKuPA, 2013.

K U Syaliman et al 2018 J. Phys.: Conf. Ser. 978 012047. Improving the accuracy of k-nearest neighbor using local mean based and distance weight. https://iopscience.iop.org/article/10.1088/1742-6596/978/1/012047/pdf, 2018.

Chih-Min Ma, Wei-Shui Yang, and Bor-Wen Cheng. How the parameters of k-nearest neighbor algorithm impact on the best classification accuracy. https://scialert.net/fulltextmobile/?doi=jas.2014.171.176#1180333_ja, 2014.