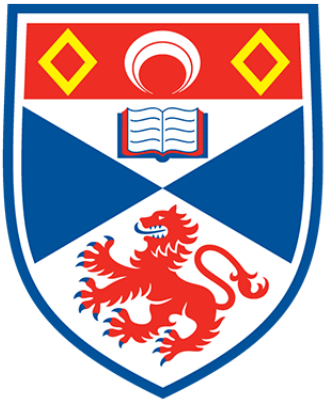


University of
St Andrews

CS5014 Machine Learning

Classification of object colour using optical spectroscopy

April 22, 2018

Lecturer:
David Harris-Birtill
Kasim Terzić

Submitted By:
140011146

1 Introduction

In this practical, two classification tasks were performed on an experimental dataset.

2 Binary Classification

2.1 Preprocessing

Before any preprocessing is done, the data was split into training and test datasets. This is to ensure no bias when looking at the data, even for visualisations. The split is also stratified over the output to ensure the training and test sets have the same proportion of classes that is found in the original data. This again is done to prevent bias and overfitting as much as possible so that the trained model is not overfit to a subset of the data with a different proportion of classes.

Nothing was done about the negative intensities found in the data as this came from issues during data collection where the distance of the spectrometer changed. It is not trivial to account for the errors as the distance moved is not known. Further, scaling to the negative values to keep the intensities going from 0-100% is likely to skew the data in an unknown way. Removing all data points where negative values exist would also hurt as the dataset is already quite small. TODO. Because of these reasons, the data was not cleaned for negative intensities. Furthermore the data was not normalised. This is because all the input features represented the same units of measurement and so there is not any difference between the ranges of each feature.

2.1.1 Data visualisation

First, the spectroscopy data can be visualised by plotting the intensities and wavelengths to see if there are visual patterns than can be seen in the data. As there are over nine hundred input features, it is likely many of them are either redundant or not well correlated to the input data.

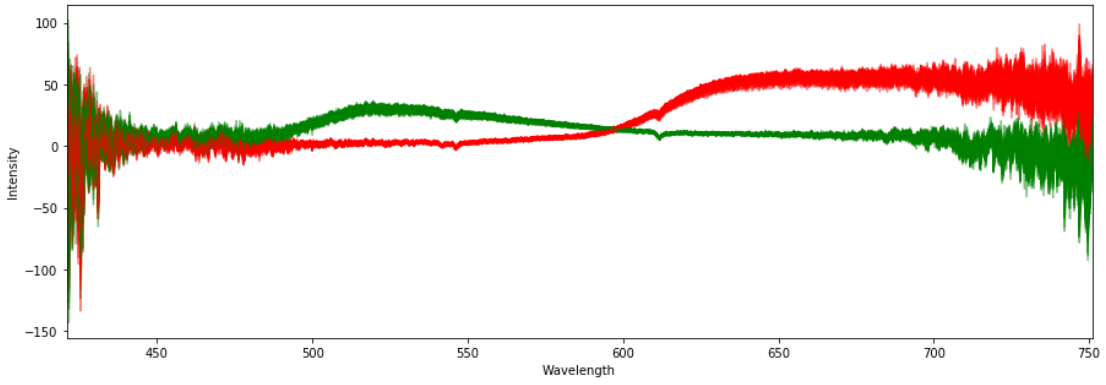


Figure 1: Initial plot of the data, showing the intensity of the wavelengths measured. The plot is colour coded with the output so any differences between the red and green data can be easily seen.

From the visualisation, it is interesting to see that for many wavelengths, there is a very clear distinction between the red and green intensities. This suggests that for those wavelengths, it would be very simple to classify. Further, the visualisation was created from all the training data, so it can be seen there are no large errors in the middle wavelengths where no data points incurred significant differences, which shows the consistency of the data. It can be see towards the start and end of the spectrum, the intensities are more varied, indicating possible errors during measurement.

As such, it would make sense when choosing input features to mostly include features from the middle wavelengths where less possible errors occurred.

2.1.2 Data correlation

Next, the correlation of each input feature wavelength can be calculated and plotted to see how the set of input features is correlated to the output. It is not feasible to plot each input feature's correlation separately due to the large number of features. Therefore all the features are put onto the same graph for comparison.

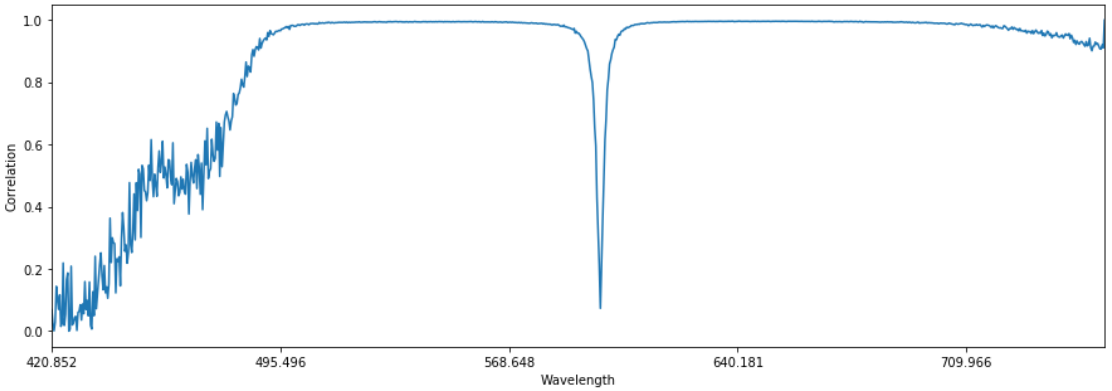
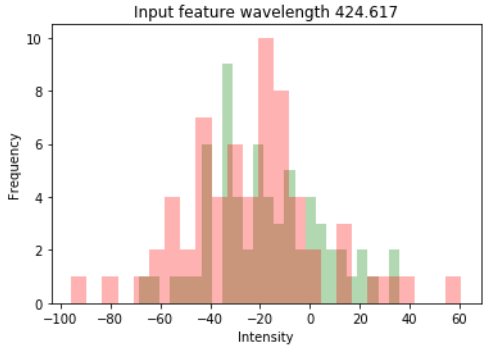


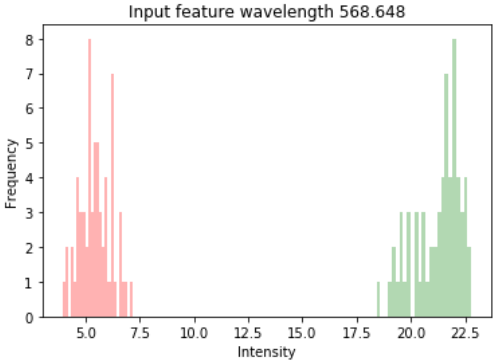
Figure 2: Correlation of each wavelength input feature to the output classification.

The curve of input feature correlation corresponds very closely to figure 1. At the beginning where there seems to be more noise, the correlation is low. At the wavelengths where the red and green curves start to become distinct from each other, the correlation of that wavelength to the output class tends to perfect correlation. At the center where the red and green curves cross over, a dip in the correlation can be seen as it becomes difficult to distinguish the two in the overlap. This is good news and expected, as the clear difference between the red and green curves corresponds to a clear classification.

To show this more clearly, the histograms of a single input feature can be visualised.



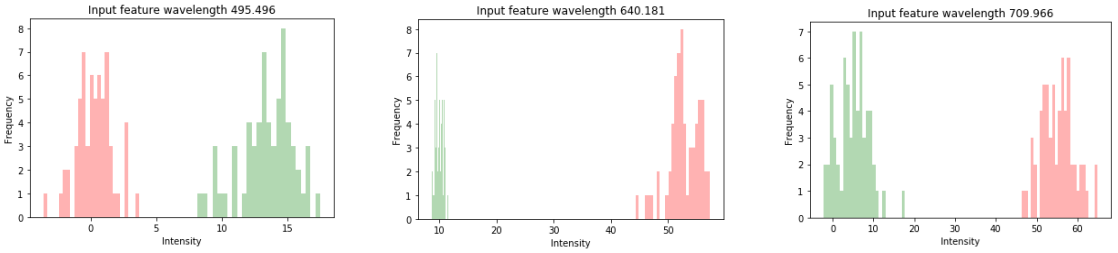
(a) Histogram of a wavelength where there is high overlap and low correlation between the red and green curves.



(b) Histogram of a wavelength where the red and green curves are distinct and there is perfect correlation with the output class.

Figure 3: Comparison of histograms of a single wavelength measurement. One taken at a low correlation wavelength and the other at a high correlation wavelength.

The histograms in figure 3 show the difference between an input feature with low correlation and an input feature with high correlation. The two wavelengths were arbitrarily chosen based on the correlation graph, but an obvious difference can still be seen. With low correlation, there is a lot of overlap between the two classes, which could also have been seen in figure 1. For the input feature with high correlation, the histogram shows very clearly the separation between the two classes. Figure 4 further supports the fact that all input feature wavelengths where the correlation is high and a clear distinction seen in figure 1 have clear, separate intensities. As the histograms contain all data points from the split training set, it is very likely that a single input feature can lead to a high classification rate.



(a) Wavelength 495.496 (b) Wavelength 640.181 (c) Wavelength 709.966

Figure 4: Further histograms of highly correlated wavelengths, showing the separation of the two classes occurs for many different input features.

2.2 Results

From the preprocessing stage, it was shown that there are multiple input features where a clear distinction using just that single input feature could be seen between the two classes. As such, a single input feature is tried first for training to see what classification rates could be obtained.

Input feature	Accuracy
Wavelength 421.228	0.5
Wavelength 495.496	1.0
Wavelength 568.648	1.0
Wavelength 640.181	1.0
Wavelength 709.966	1.0

Table 1: Results of training a logistic regression model on a single input feature.

Despite the indications that the data would be easy to classify, the predictions from the logistic regression model are suspiciously high. All single input features used that had a high correlation gave perfect accuracy during prediction. The first thought was that perhaps the model is overfitting, so the same training is done again with k-fold cross validation.

Input feature	5 fold accuracy	10 fold accuracy	25 fold accuracy
Wavelength 421.228	0.5	0.49	0.5
Wavelength 495.496	1.0	1.0	1.0
Wavelength 568.648	1.0	1.0	1.0
Wavelength 640.181	1.0	1.0	1.0
Wavelength 709.966	1.0	1.0	1.0

Table 2: Accuracy of using k-fold cross validation to show the logistic regression model is not overfitting.

Even with using a high k-fold cross validation, the strong input features continued to get 100% accuracy, showing it is unlikely due to overfitting but because the data and those specific input features lends themselves well to a binary classification. It was further confirmed how well defined the classification was by splitting the training data into an even smaller dataset with only 10% of the training dataset used for training and testing on the rest of the training data. Here there is a slight discrepancy between training and testing accuracy where certain input features obtained slightly lower accuracy when running on the test data. This tells us that even with a very well defined classes as seen from previous training models and visualisations, it is difficult for models to generalise well given very few samples. At the same time, the fact that such a high testing accuracy could still be obtained from such a low number of training samples once again shows the clear difference between the classes.

Input feature	**Training accuracy**	**Testing accuracy**
Wavelength 421.228	0.67	0.46
Wavelength 495.496	1.0	0.98
Wavelength 568.648	1.0	0.99
Wavelength 640.181	1.0	1.0
Wavelength 709.966	1.0	1.0

Table 3: Accuracy of training logistic regression on 10% of the training data and testing on the remaining 90%.

Two more approaches were used as a final way of checking that one input feature is enough for a good classification of this problem. First the accuracies of each single input feature is calculated by training a logistic regression model with that feature and plotting the accuracies. Finally, recursive feature elimination was used to get the accuracy score as the number of features used increased to show any more than two input features will always give 100% accuracy.

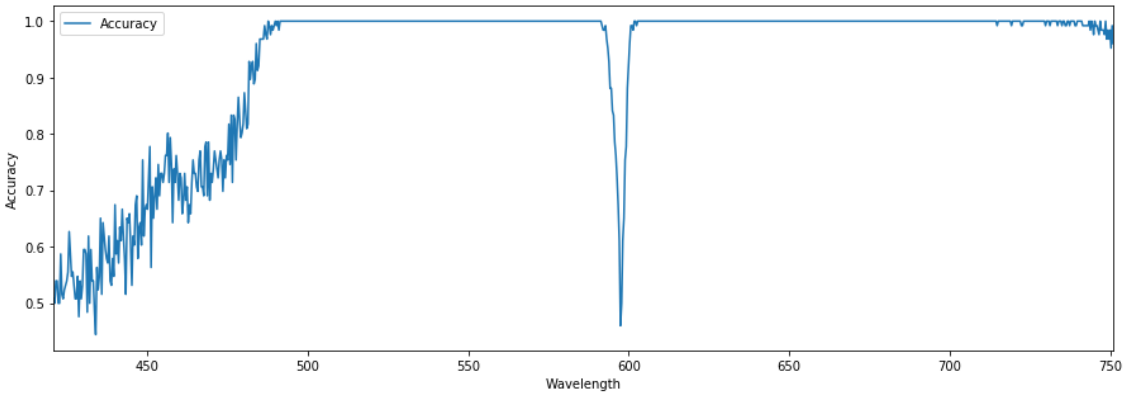


Figure 5: All input feature wavelengths trained individually and their accuracies graphed.

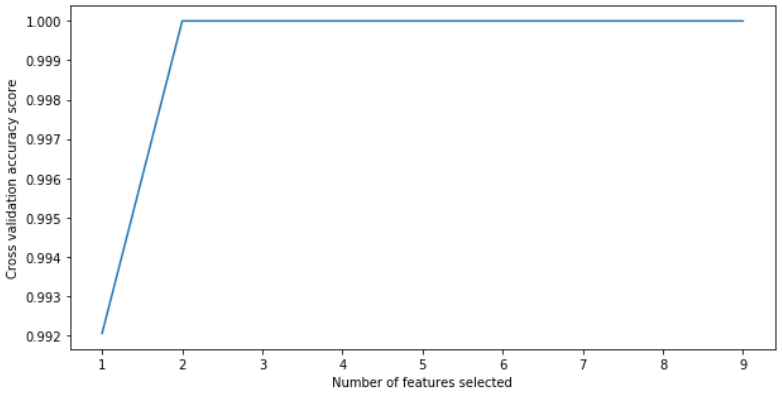


Figure 6: Cross validation accuracy score of increasing number of features using recursive feature elimination.

Figure 5 shows the accuracy of each input feature when used alone to train the logistic regression model. It can be seen to be very similar to figure 2 which showed the correlation of each input feature. This indicates the well correlated features are also the ones that have good accuracies with little noise disrupting this pattern. The graph of accuracy obtain from recursive feature elimination with cross validation does not confirm it is always the case the one input feature is enough for perfect classification. This is because of the wavelengths at the beginning and end of the data where there is more overlap. However, one input feature already gets a 0.992 classification rate on average and any more than one feature always gets 100%.

2.3 Evaluation

From the initial visualisations of the binary classification data, it could be seen that for a range of wavelengths, the two classes of red and green colours were easily distinguishable by eye. This was further confirmed by graphing the correlation of each wavelength to the output classification and shown clearly by plotting the histogram of a single wavelength, which showed a clear separation between the classes. Because of these observations, it was not a huge surprise that a simple logistic regression model was able to perform perfect classification on the training data, using specific single input features. The results were confirmed to not be due to overfitting, as k-fold validation was performed with a large value of k. Moreover, it was shown that with a training set size of just 10% (12 data points), the high classification rate was still retained when predicting the other 90% (114 data points).

The results make sense when coming back to what the actual classification problem is. The data represents spectroscopy data with red and green colours placed under the device. As red and green are very distinct colours with different wavelengths, it was natural that there would be a clear separation between the colours.

Because of the high accuracy obtained by the logistic regression model, no further machine learning models were used or tried on the dataset. This was done for a few reasons. First, little improvement in terms of accuracy could be gain from using different models if the accuracy from logistic regression is already at 100% using one input feature and 12 training data points, so there is little point to try and optimise further. Second, most other machine learning models involved more complexity than a logistic regression model so if little improvement can be made, it seemed more practical to stick with the simple model rather than try something more complex.

3 Multiclass Classification

For the multiclass classification, many of the same steps as the binary task were taken. Because of the added complexity of more classes, the results were not the same and hence the steps deviate from the binary task at a certain point.

3.1 Preprocessing

Again the data was split into training and test data at the very beginning before any further visualisation or analysis. Stratification was also used on the output to ensure the same proportion of classes in both the training set and testing set. Additionally, the data was not cleaned or normalised for the same reasoning explained for the binary classification.

3.1.1 Data visualisation

With the additional colour classes in the dataset, it can be seen that there are more areas of overlap and no clear wavelengths where all colours are distinct from each other. The pattern of having noisy input at the early and late wavelengths is still apparent in this data, which shows the measurement was relatively consistent to the data for binary classification. Because of the additional colours, it may be expected that a single input feature would not be enough to classify the colours. Particularly the pink and red wavelength intensities look like they follow a very similar pattern, which makes sense as they are the closest colours out of the five. These two colours may be one of the more difficult to determine between compared to the other classes.

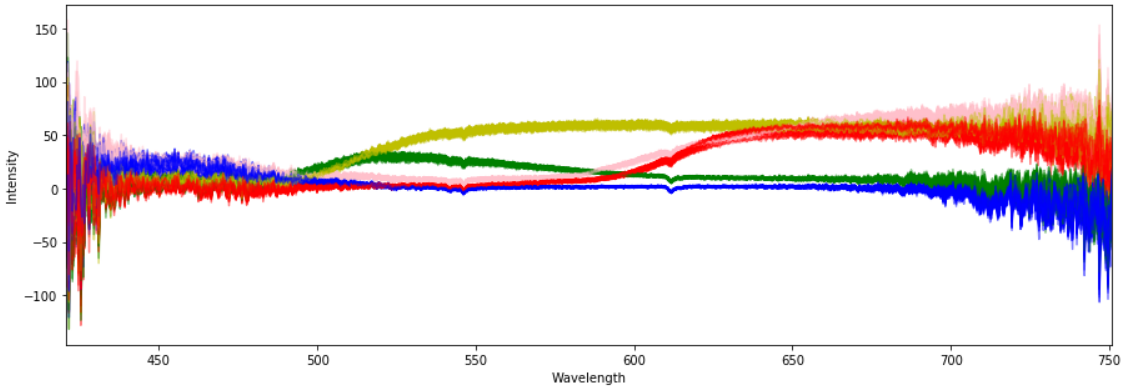


Figure 7: Plot of the data to show the intensity of wavelengths of the different colour classes.

3.1.2 Data correlation

The correlation of each input feature to the output class is graphed to see if similar patterns as the binary classification can be seen.

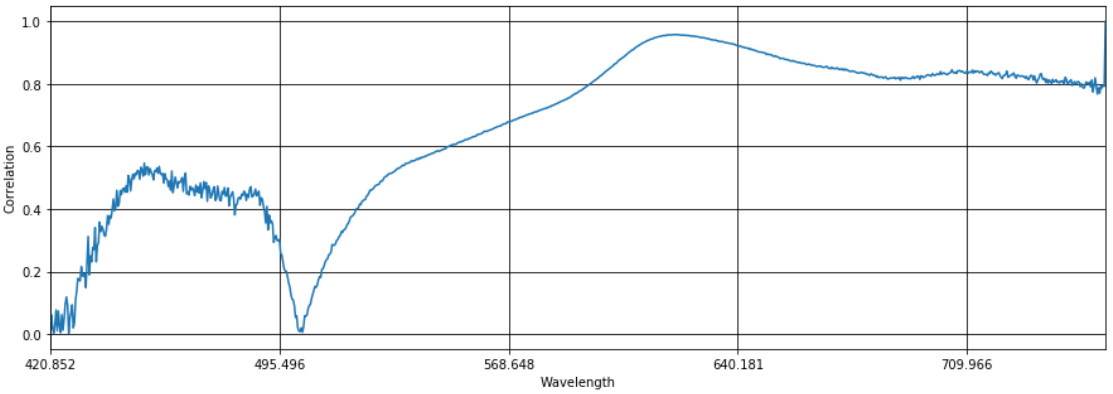
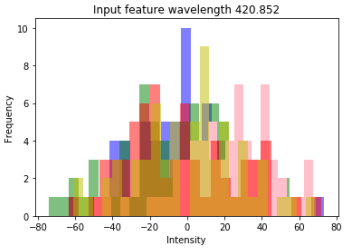
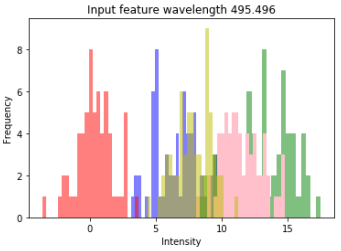


Figure 8: Correlation of each wavelength input feature to the colour classification.

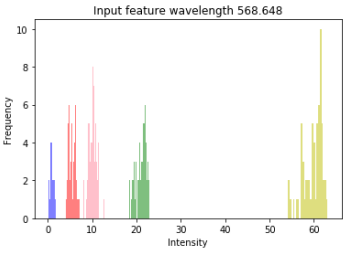
Unlike in binary classification, here there is never a single input feature where there is perfect correlation and this follows from the visualisation of the data in figure 7. Some ranges of wavelengths have a high correlation than others, but not enough to completely classify each of the five classes. To get a good idea of how the data is spread for each wavelength, a few histograms are created like in the binary task.



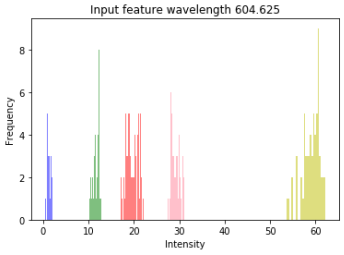
(a) Wavelength 420.852



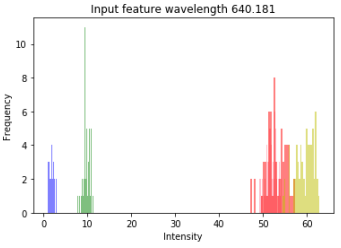
(b) Wavelength 495.496



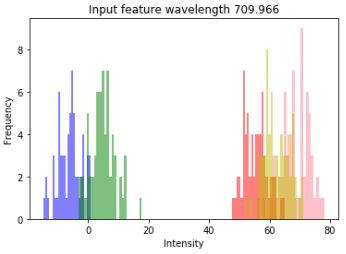
(c) Wavelength 568.648



(d) Wavelength 604.625



(e) Wavelength 640.181



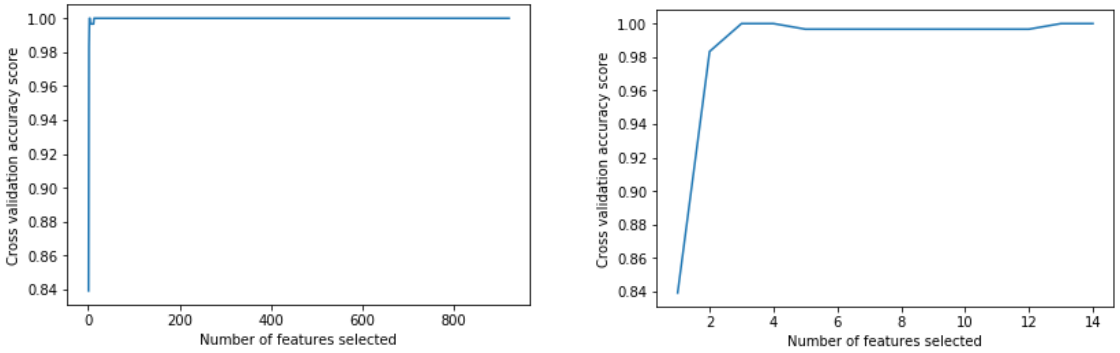
(f) Wavelength 709.966

Figure 9: Histograms of single input features for multiclass classification

Because of the additional classes, there is a much less clear distinction between all the colours. For example we can see for wavelength of 420nm, 495nm and 709nm there is significant overlap between at least two classes. The intensities at wavelength 568nm and 604nm show a clearer difference, but there is not a large margin of difference between some classes, for example red and pink. This suggests that one input feature is not likely to be enough to get perfect classification like in the binary case. Further, models could be more prone to overfitting, as the lack of large margin difference in a single input feature means any small outliers may cross into another class boundary more easily. With more than a single input feature, the combination of different boundaries for different wavelengths could uniquely identify the class.

3.2 Results

To get an idea of the optimal number of input features that should be used, recursive feature elimination is used. A logistic regression model is used first following the good results this simple model got during the binary classification task.



(a) Showing all features

(b) Showing only up to 15 features

Figure 10: Cross validation accuracy score of increasing number of features with recursive feature elimination for multiclass classification.

It can be seen there is an odd dip at the beginning when using fewer features. Upon closer inspection, at three and four input features the accuracy becomes 100%, but dips back down from including more features before going back to 100% after thirteen features. This suggests that between five and twelve input features, the addition of more features sometimes causes noise that is not well correlated to other features, causing some overfitting. It should be noted that the dip is a *very* small dip in accuracy, from 100% to 99.67%. Past twelve features, the accuracy stays at 100% up to using all features from the data. This shows that with enough wavelength information, it is possible to always classify the different colours. Adding additional features does not lead to overfitting in this case because there is little noise in the actual data and all the input features represent the same thing in wavelength intensity, so it does not become the case that having all the input features adds irrelevant information, but more the case it simply adds redundant information.

As it is expected using all the input features would get 100% accuracy, the goal pursued was to see the kind of accuracies that could be gained with as few features as possible. From the visualisations earlier, it may be possible with one feature, but fewer features may also make the model harder to generalise, especially with less training data.

No. of input features	**Training acc.**
1	0.73
2	1.0
3	1.0
4	1.0

Table 4: Accuracy of training with different number of input features for multiclass classification.

From training on all the data from the training set, it was possible to get 100% accuracy with 2 features. Again, we must do more to ensure this is not a case of overfitting. Both cross validation and training with significantly less sample points were done like in binary classification to confirm any absence of overfitting and to show that the model can generalise to larger volumes of unseen data with few features and few training examples.

No. of input features	10 fold acc.	50 fold acc.	10% training size test acc.
1	0.727	0.721	0.581
2	1.0	0.998	0.996
3	1.0	1.0	1.0
4	1.0	1.0	0.993

It can be seen that similar to the case with binary classification, where 100% accuracy could be obtained from training on all the training data, using additional methods confirms the fact that the result is not a case of overfitting. There is only a very small drop in accuracy from using a large k-fold validation and from using a small 10% training size. It is interesting to see that with 4 input features, no loss of accuracy can be seen despite the methods used, showing the logistic regression model with 3 input features is able to generalise very well to unseen data.

4 Conclusion and Evaluation