AO09: Determining volcanic ash type from satellite data

Candidate Number: 1032529

Supervisors: Dr I Taylor and Prof R G Grainger

Abstract

In this project, the feasibility of using machine learning to classify volcanic ash by ash-type is evaluated. The machine learning model was trained on a brightness temperature spectra dataset containing simulated ash spectra for a range of ash types and ash parameters. The model, once trained, was then validated, testing on a fraction of the dataset unseen during training, as well as a separate test dataset to determine it's efficacy. When validated against data in the test portion of the initial dataset, the model had a high accuracy rate, classifying up to 84% of cases correctly. However, when tested on a separate 2nd test dataset with different values of ash parameters to the ones seen in training, the machine learning model attained a maximum accuracy of 62%. This project shows that a machine learning model with a good degree of accuracy can be produced when tested on a sufficiently large amount of data, however questions regarding its applicability remain open and further work is necessary to evaluate whether the model can be used to classify ash type accurately, directly from satellite imagery.

1 Introduction

Volcanic eruptions pose a significant risk to human civilisation, affecting settlements, human health and the aviation industry. In the past, the aviation industry has been significantly impacted by volcanic eruptions, with the largest scale disruption caused by the Eyjafjallajökull eruption in 2010, which over the course of six days, led to an average of 48 percent of European flights being cancelled, and an economic loss to the aviation industry of 1.7 billion US Dollars [1] [2] [3].

Detailed observations and monitoring is vital to mitigate risk, and reduce disruption. In recent years, following advancements in satellite technology, it is now possible to quantify key information about the clouds, including ash, SO_2, and aerosol content in a process known as a 'retrieval' [4]. However, in order to obtain accurate information on the volcanic ash cloud, a set of wavelength-dependent refractive indices must be assumed. These refractive-indices can be thought of as an 'ash-type'. Assuming the most accurate ash-type will result in the most accurate retrieved quantities. [10]. Physically collecting samples in order to determine the refractive-indices can be difficult, and so currently estimations of the most appropriate ash-type are made from a set of existing ones determined from historically collected samples [9].

Machine learning is described as the process by which computers use statistical techniques to extract knowledge from data [12]. The use of machine learning in volcanology is an emerging field and has been used in the past to classify seismic event types and in the estimation of landslide susceptibility [21]. Machine learning methods are most successful when they can automate decision-making processes using known examples [12]. Instead of estimating the most appropriate ash-type, a machine learning model could be used to analyse the brightness temperature spectra (BTS) of a volcanic ash cloud to make a decision on which ash-type to assume to retrieve its properties most accurately. The method of predicting a class label, from a set of predefined possibilities is known as 'Classification' and is most suitable for this study [12].

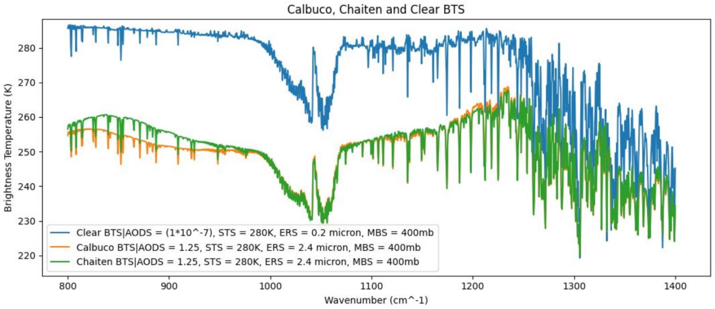

Figure 1: 'Calbuco', 'Chaiten' and 'Clear' BTS

Classification methods using machine learning requires the input data to be accurately distinguished in order to output the correct classes, with the input data being the BTS of a volcanic ash cloud. in Figure 1, BTS are simulated using the radiative transfer model RTTOV for three different ash-type classes: 'Calbuco', 'Chaiten' and 'Clear Sky', where 'Calbuco' and 'Chaiten' are ash-types representative of the ash expelled from each of these Chilean volcanoes, while the 'Clear' class is representative of a sky completely clear of ash or cloud [17].

These three classes have different wavelength-dependent refractive indices and therefore result in different spectral shapes when plotted with respect to wavenumber. The radiative model takes ash cloud parameters as inputs in order to simulate the BTS, namely ash optical depth (AODS), surface temperature (STS), ash effective radius (ERS), and pressure at cloud height (MBS).

The BTS of the the three ash-types has a general v-shape, with the spectra of 'Calbuco' and 'Chaiten', translated downwards in temperature over the 800 cm^{-1} to 1400 cm^{-1} spectral range. Fine ash particles in the atmosphere can block solar radiation, and absorb and re-emit infrared radiation explaining the lower brightness temperature recorded for the 'Calbuco' and 'Chaiten' spectra compared to the 'Clear Sky' BTS [5]. Additionally, there is a visible brightness temperature difference between the 'Calbuco' and 'Chaiten' spectra despite using the same values of ash parameters, which can be attributed to the difference in composition of the volcanic ash expelled from each of the two volcanoes. This difference between the spectral values for each of these ash-types is what the machine learning model should exploit in order to distinguish between spectra and classify the spectra of imaged volcanic ash clouds correctly.

Volcanic ash clouds can be viewed using the infrared atmospheric sounding interferometer (IASI) and a brightness temperature spectrum can be produced spanning a wide spectral range from 645 cm^{-1} to 2760 cm^{-1} with a fine spectral resolution of 0.5 cm^{-1} [4]. Ideally, running the machine learning model on the IASI spectra would act as a pre-processing step to determine which ash-type to assume when carrying out the subsequent retrieval. The aim of this project is to improve and expand on the machine learning model first developed by Martin Okánik, and evaluate its ability to accurately predict ash-type [9].

Section 2 in this report explains the methods used to develop and enhance the machine learning model, including those used to rectify errors made by the model. Section 3 documents the results and each of the alterations made to the model in more detail. Finally, Section 4 concludes the report, and evaluates further work that could add value.

2 Methods

2.1 Machine Learning Method

The machine learning model implemented in this study was programmed using Python and its in-built scikit-learn module.The type of machine learning model used is known as an MLPClassifier which is a type of neural network [13]. A family of algorithms modelled loosely on the human brain, the neural network here hopes to find a link between the output classes and the input spectra by writing the output classes as a weighted sum of the input spectra, similar to curve fitting but on a more complex scale [6].

While curve fitting typically fits data to a polynomial curve, here weights are applied to each of the input spectra in a linear sum. Unlike other linear models, neural networks have multiple layers known as 'hidden' layers (See Appendix for a visual schematic). The final class is then calculated as a sum of the outputs from each of the hidden layers. This 'sum of sums' is then passed through an activation function such as the 'tanh' function or 'relu' function which allows the outputs to be constrained to a given domain regardless of the input value. The added complexity of the neural network model makes it more powerful than basic linear machine learning models without hidden layers. [12].

$$\begin{aligned} h[0] &= w[0,0]x[0] + w[1,0]x[1] + .. \\ h[1] &= w[0,1]x[0] + w[1,1]x[1] + .. \\ y &= v + h[0] + h[1] + .. \end{aligned}$$

The weighted sum for the layers and the total sum which give the output is summarised in the equations, where $w[i,j]$ is the value of the weight in layer j for spectral value i, and $x[i]$ is the ith spectral value. y is the output class calculated by the model, using the hidden layer sums and v is an added bias [12].

Producing a machine learning model is typically split into two stages: training and testing. During training, the model iterates through different values of the weights (coefficients) in the equation to find values that minimise a log-loss error function for the input spectra [13]. The equations with the determined weights can then be applied to test data, not used during training, as part of the test stage to quantify the efficacy of the machine learning model. Typically, a full dataset will be split into training and testing datasets as part of the production of a machine learning model [13][12]. The working model can then be applied to different spectra to predict its class.

2.2 Making the Dataset

Prior to training and testing, a careful method of pre-processing must be carried out to ensure a dataset that can be effectively trained is produced, and in this section this method will be discussed [9] [14].

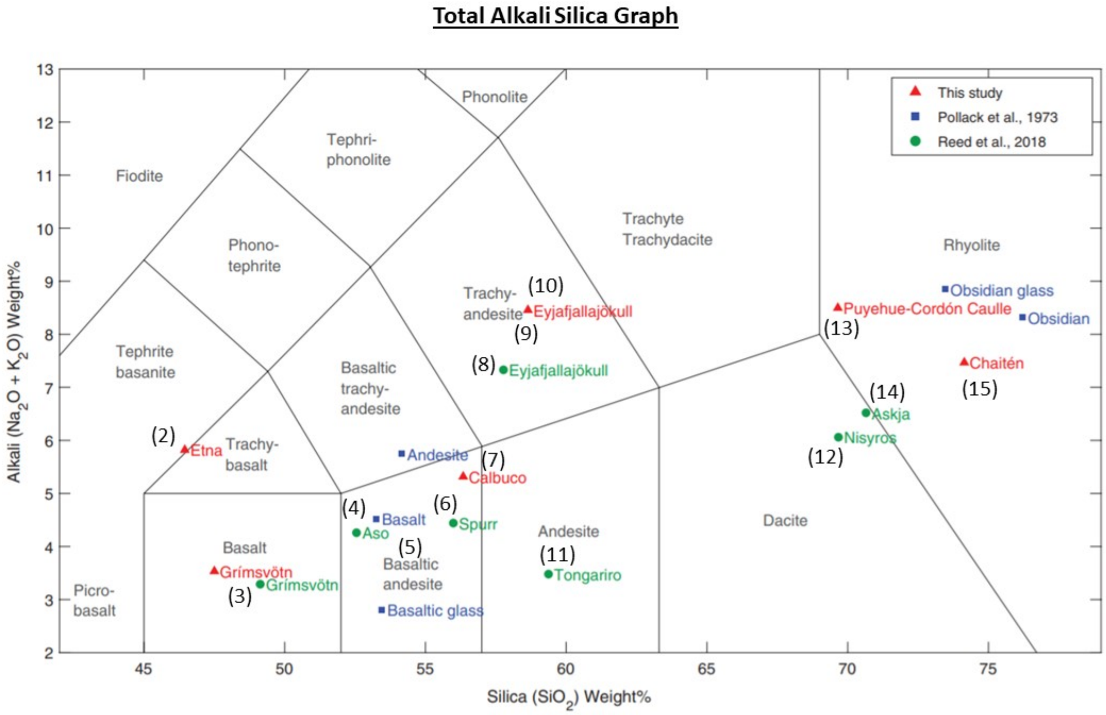

Figure 2: Total Alkali Silica diagram (with labels) [15]

Spectral Classes	
Class No.	Class Name
1	A83
2	Etna
3	Grimsvotn
4	Aso
5	Mayon
6	Spurr
7	Calbuco
8	Eyjafjallajökull-Reed
9	Eyjafjallajökull-Reed-2
10	Eyjafjallajökull-Deguine
11	Tongariro
12	Nisyros
13	Puyehue-Cordon-Caulle
14	Askja
15	Chaiten
16	H_2SO_4
17	Ice
18	Clear Sky
19	Cloudy Sky

Table 1: Classification labels

To produce a training dataset, simulation spectra files are produced for 19 total classes: 15 ash, 'Ice', 'Sulfuric Acid', 'Clear Sky' and 'Cloudy Sky', all of which are detailed in Table 1. In order to do this, complex refractive indices used by Reed et al., Deguine et al. and Patterson et al. are used along with the RTTOV radiative transfer model [9] [17] [15] [8] [16]. Each of the first 15 ash-types are based on samples from different volcanoes, detailed in Table 1, and have different compositions, and this is best visualised using the Total Alkali Silica (TAS) diagram which plots the alkali content of the ash with respect to its silica content, as plotted in Figure 2 with the relative positions of 14 of the 17 ash-types [15]. Through varying the properties of the ash in the simulations, initially 2000 spectra are created for each of the first 17 ash-types (including 'Ice' and 'Sulfuric Acid'). For the 'Clear Sky' class, initially 17 spectra are created, and for the 'Cloudy Sky' class 900 spectra are created. The number of spectra was later increased in Section 3.7.

2.2.1 Scaling

While the original spectra files span a range of wavenumbers from 800 cm^{-1} to 1400 cm^{-1}, water vapour absorption which begins to dominate for wavenumbers of approximately 1200 cm^{-1} means that it is better to scale the spectral data such that the maximum wavenumber is 1250 cm^{-1}. This was implemented by Okánik in his earlier study and was unchanged here [9].

2.2.2 Normalising

Normalising the data is an important pre-processing method. Normalisation or 'feature scaling' in this way adds value by transforming features to be on a similar scale. This can potentially improve performance and training stability of the model[18]. Okánik implemented an initial version of normalisation to the model, which was later altered in this study. Normalisation of a given spectra is carried out around a given wavenumber, chosen to be 950 cm^{-1}, such that the spectral value at this wavenumber is zero. The entire spectra is then divided by the standard deviation to give the normalised spectra. This is then repeated for every spectra in the dataset [9].

2.2.3 Smoothing

Smoothing of the data was applied to further improve the accuracy of the machine learning model. Both the clear and ash spectra follow a general trend but have many narrow absorption gas spectral lines. Smoothing can remove these gas spectral lines potentially resulting in better model performance [9]. Smoothing the data is carried out by firstly dividing the spectra into portions defined as 'windows'. In the smoothing program, this window size influences the effect to which the spectral data is smoothed, with a larger window size resulting in smoother spectra [20].

2.3 Training the Model

After making the spectral dataset, a portion of the dataset is set aside for training of the model, typically 80 percent, and the rest is set aside for validation/testing of the trained model. This process is known as a 'train-test-split' [13].

The weights are optimised using a numerical solver following a method known as stochastic gradient descent[13]. Two solvers, with largely equal accuracy, are used during this project, namely the Limited Memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) solver and ADAM solver, with the ADAM solver being used more frequently in this study as a result of its shorter computation times [13].

2.4 2nd Test Dataset

Applying the model to the test dataset, a proportion of the total spectral dataset, is the first step of evaluating the model efficacy. However, a complex model can occasionally become 'over-trained', 'memorising' the training dataset, resulting in a 'test score', the accuracy of the model at predicting classes of the test dataset, that is an overestimate of the true model performance [12].

In order to evaluate the true model performance, the model can be tested on an entirely new test dataset, where the ash parameters are different from those used in the training dataset. The model is applied to this 2nd test dataset in order to get a better estimate of the true model performance.

2.5 Enhanced Training

Following testing on the 2nd dataset, a large fall in accuracy was measured compared to the test score recorded when producing the model [12].

Several different methods were experimented with in order to try and improve the performance of the model:

- Reduce the complexity of the model by varying the 'alpha' parameter in the model training which changes the size of the weights in the weighted sum.
- Introduce an alternative method of training, applying a portion of the dataset to the model as it trains. This allows the log-loss and score of the model to be calculated as it trains.
- Increasing the amount of spectral training and testing data. When simulating the spectra, different values of ash properties are used by the radiative transfer model. Increasing the number of values of ash properties within the given range substantially increases the amount of training and testing spectral data. More spectra were also added to the 2nd test dataset which is applied to the model following training and initial testing.

3 Results

3.1 First Training Results

The first results in Table 2 came from training the model on a BASIC dataset (listed in the Appendix) made up of only 4 ash-types, with 2 volcanic ash refractive-indices and 'Ice' and 'Sulfuric Acid'. The first result comes from training a model largely unchanged from the model trained by Okánik. This was later expanded to the 17 ash-types listed in Table 1, but without the 'Clear Sky' and 'Cloudy Sky' classifications. The following initial smoothed and increased iteration models use enhanced models with changes made to model parameters and processing methods [9].

The results from the first training runs showed that the MLPClassifier is effective at training a model with a good degree of accuracy at predicting the ash type of its test dataset.

Spectral Classes	
Training Dataset	Train/Test Score (%)
BASIC (No Smoothing)	Train = 88 Test = 86
BASIC (1000 iterations)	Train = 91 Test = 88
BASIC (Window Size 51)	Train = 91 Test = 88
BASIC (Window Size 25)	Train = 95 Test = 90
All Ash (No Smoothing)	Train = 78 Test = 72
All Ash (Window Size 51)	Train = 75 Test = 72

Table 2: Training on first datasets

The iterations referred to in the table represent the number of times the LBFGS solver used here iterates to minimise the log-loss score. The default number of iterations used in the 1st run is 200, and increasing this to 1000 had a positive impact on the test score. This number of iterations was also followed for later training attempts in the table.

The initial training runs on the BASIC dataset were also less time-consuming and thus acted as a way to test the effect of smoothing, which was newly introduced, and in particular what window size should be used for training. The larger the window size, the smoother the data. A window size of 51 made no difference to the test score compared to the 1000 iteration training run carried out previously. Reducing the window size to 25 which reduces the smoothing effect seems to have a positive impact on test score. It is difficult to ascertain whether this is directly because of the smoothing window size or because of 'over-fitting' of the data which may have started to occur with such a small dataset. Over-fitting occurs when the model begins to memorise the particularities of the training set, performing well when applied to the training set, but poorly with new data [12]. For later training attempts, the smoothing window size was largely fixed constant at 25. The exact number used is relatively arbitrary and has little impact on smoothing for small changes of the window size. While each of the models show a good accuracy, despite very few changes made to the model parameters, the test scores recorded are not always a good metric for model performance and they are not always able to show over-fitting if it has occurred.

3.2 Adding Clear and Cloud Spectra

An key aim in this study was to introduce the 'Clear Sky' and 'Cloudy Sky' classifications to the model. In this section, the efficacy of predicting these additional classes will be evaluated.

Having amended the spectral dataset to add the 'Clear' spectra and 'Cloud' spectra, the first working model was produced with the following results.

Spectral Classes	
Training Dataset	Train/Test Score
All Classes	Train = 78% Test = 74%

Table 3: Training on all ash types and 'Clear' and 'Cloud' samples

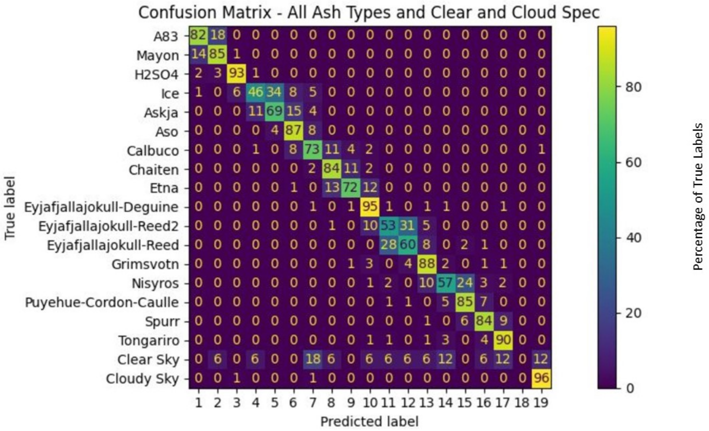

Figure 3: Confusion matrix for 1st model (Accuracy = 77%)

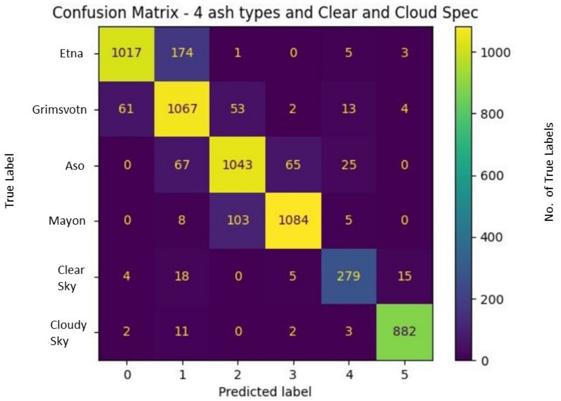

Figure 4: Confusion matrix with more 'Clear' Spectra using 6 classes (Accuracy = 89%)

The most effective way to visually display the model results is in the form of a confusion matrix, which displays how often the class corresponding to the row, is classified as the class corresponding to the column. The confusion matrix is an excellent way of visualising where the model is and isn't working and is far more informative than just an accuracy score [12] [13].

The confusion matrix in Figure 3 represents the distribution of true and predicted labels as a percentage of the number of true labels for each row. The model is clearly unable to correctly predict spectra from the 'Clear Sky' class using 17 spectra, with 0% predicted correctly. Substantially increasing the number of 'Clear Sky' spectra rectifies this as evident in Figure 4, which shows the majority of 'Clear Sky' spectra being classified correctly. Here the number of true labels are plotted instead to show the increased number of 'Clear' spectra.

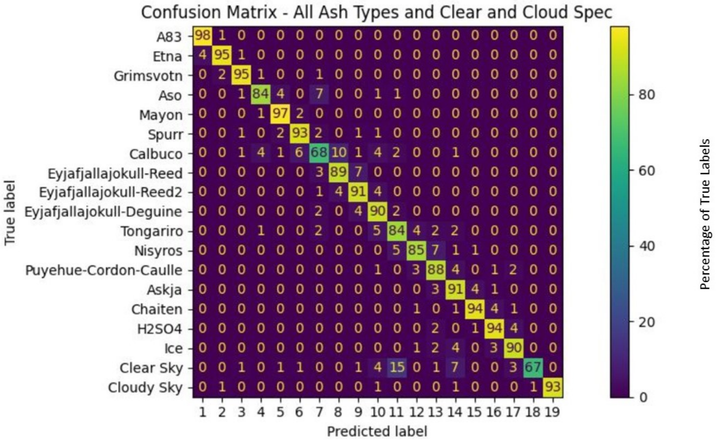

Figure 5: Confusion matrix for all classes with more 'Clear' spectra (Accuracy = 89%)

Expanding this to all classes shows that the model is capable of distinguishing 'Clear' spectra even when all the other classes are present.

Unfortunately due to an error in how spectra and labels were stored, the predicted labels using the first few models are from applying the model to the entire spectral dataset, including the training data, although this was fixed for the results in Section 3.6. These confusion matrices do not reflect how the model will perform with spectra of different ash parameters, but it does provide evidence that the model has been trained to a good degree of accuracy.

A combination of Equatorial and Standard atmosphere models are produced. For Figure 4 above, the model used is produced for a Standard Atmosphere. Meanwhile, the Equatorial profile was used for Figure 5 since this model was directly used for further testing on the 2nd test dataset which was initially created for the Equatorial atmosphere profile. The atmosphere used has little effect on scores.

3.3 1st validation with the 2nd dataset

The confusion matrices in Figures 4 and 5 only show that the model has been trained accurately, but not whether the model can be applied to spectra with any values of ash parameters. In order to get an indication of how 'generalised' the model is, which is essentially how versatile it is, it is important to apply the model to a 2nd test dataset. This dataset was created using the same method as in Section 2.2, however, the values of the ash properties were changed. Applying the model to the 2nd test dataset gives a better indication of how the model may perform when exposed to IASI spectra, which are not limited to given values of ash parameters.

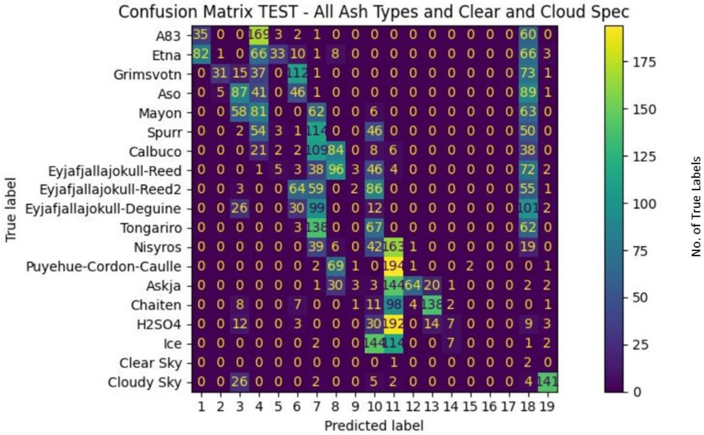

Figure 6: Confusion matrix tested on 2nd test dataset (Accuracy = 10%)

In Figure 6, applying the model to the 2nd test dataset is attempted. The model performed poorly with only a 10 percent accuracy, indicating a large amount of error. For such a matrix, the error can be evaluated as the percentage of incorrect classifications or (100-Accuracy). The 'Cloud' class is the only class for which there is a good agreement between true labels and predicted labels. There is a larger brightness temperature difference between the ash and 'Cloud' spectra, so this may explain the higher than average accuracy of prediction for these classes. For most of the ash classes however, the ash spectra are misclassified as 'Clear Sky'.

There are a couple of possible causes for these errors:

- Over-fitting may have occurred which makes the model ineffective with data that is not identical to the training dataset. This may explain the poor accuracy at predicting ash classes, but not why this is exclusive to the ash classes - 'Cloud' spectra have been predicted to a higher accuracy.

- The 2nd test dataset is made up of far fewer spectra than the training dataset which may explain why the model is struggling. Increasing the number of test spectra in this dataset may improve the accuracy. However, the full working model should be able to correctly classify a single IASI spectra.

3.4 Validating the Model and Changing Feature Scaling

In order to prevent over-fitting, a process known as 'regularisation' is carried out, to improve the model's ability to work with unseen data. Increasing the 'alpha' parameter in the model, which constrains the size of the weights that go into the machine learning model, is typically the first step. Several tests were carried out using different values of 'alpha' however there was little to no change in model accuracy, and as such this method has not been discussed in detail in this report. Results from one of the 'increased alpha' models can be viewed in the Appendix.

Since changing the 'alpha' parameter had little impact on the performance of the model, other methods were attempted. Firstly, the split of data used for training and testing is changed, and the normalisation is changed from the method proposed by Okánik to the default method within scikit-learn known as StandardScaler [13] [14].

Each spectral feature is standardised by removing the mean and scaling to unit variance. The general form of the standardisation formula of the StandardScaler is:

$$z = \frac{x - u}{s}$$

Where x is the feature, u is the mean and s the standard deviation [14].

Validated Model	
Training Dataset	Train/Val/Test Score (%)

First 4 Ash, Clear, Cloud	Train = 87 Val. = 83
	Test = 85

Table 4: Training on all ash types and 'Clear' and 'Cloud' samples

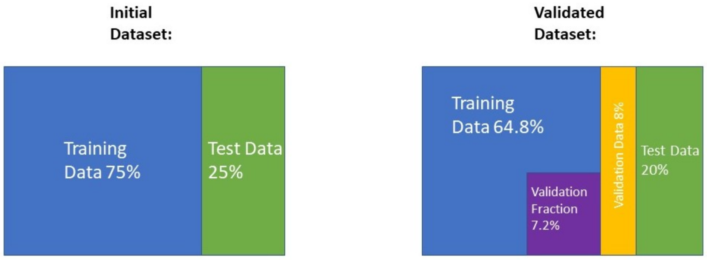

Figure 7: Train, test and validation split

Initially, at the start of this project, the machine learning model was trained on 75% of the total dataset, and tested on the remaining 25% of the data. By introducing validation, the fraction of data entering training is changed. 20% goes into Testing, 8% is also set aside for validation of the model after training, and 7.2% is used as validation data explicitly by the model during training. While it may appear there is no difference, the 7.2% fraction is actually used to validate the model as it runs. This feature of MLPClassifier is used to stop the model 'early' if the validation score (accuracy of applying the model to the 7.2% validation fraction) stops increasing by more than a tolerance value (0.0001 here) [13] [14]. The fractions of the dataset, that form the training, test and both validation datasets are visualised in Figure 7. The 8% used for validation is primarily to check the model, and is used to produce confusion matrices which do not include training data, an error made in the matrices in Section 3.2.

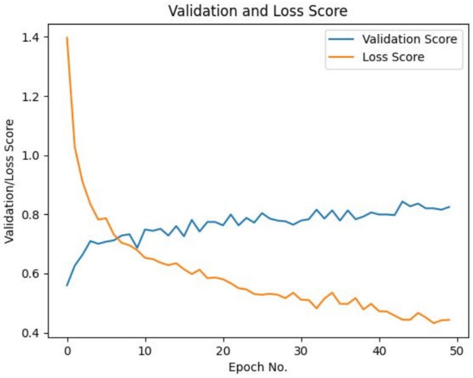

Figure 8: Validation score and model loss score

The validation score and log-loss can be recorded while applying the model to the validation fraction as it trains with respect to each epoch, or a complete viewing of the dataset by the model. The plot in Figure 8 shows a decreasing log-loss function beginning to plateau. The validation score increases and starts to reach an optimal value but shows no sign of decreasing. A decreasing validation score typically occurs when a model is overfit. This suggests that the number of epochs used for model training before stopping, is appropriate to produce a model with a sufficiently low log-loss, while not resulting in significant over-fitting. [14].

3.5 Applying a Validated Model

In this section, the newly validated model can be tested on the separate test dataset which the model has previously struggled with.

The validated model predicts the labels of the 2nd test dataset with a higher accuracy. This can potentially be attributed to the validation by the model as it runs, which allows the model to stop early and prevent overfitting, or the change in normalisation. However, there is misclassification between the ash types with 'Mayon' frequently misclassified as 'Aso'.

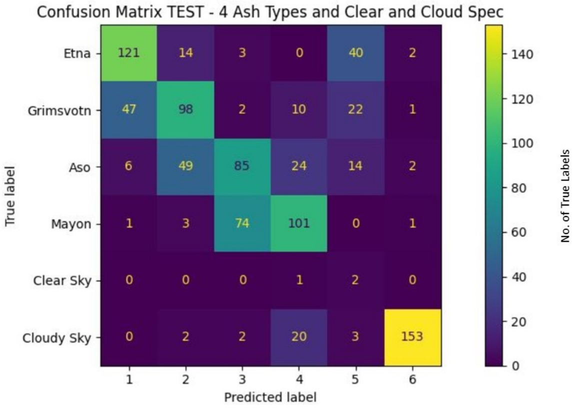

Figure 9: Confusion matrix using a validated model (Accuracy = 62%)

The same idea was extended to testing on all classes, however the model struggled to classify into all classes with a low accuracy of only 19%, although this is higher than previously observed. The confusion matrix for this test can be viewed in the Appendix.

Due to the close proximity on the TAS diagram of each of the classes in Figure 9, training and testing on a dataset with classes spaced further across the TAS diagram is carried out to see if this prevents the high rate of misclassifcation between ash classes. The results of this showed that training on ash types that are spaced further apart on the TAS diagram does not have a significant impact on the accuracy or the misclassification of ash classes (See Appendix), with spectra of adjacent classes in the list still being frequently misclassified.

The methods applied here have improved the 6-class model accuracy, and provide evidence that for the size of dataset used, careful monitoring of validation score and loss is necessary to produce a well trained model.

3.6 Increasing the number of spectra

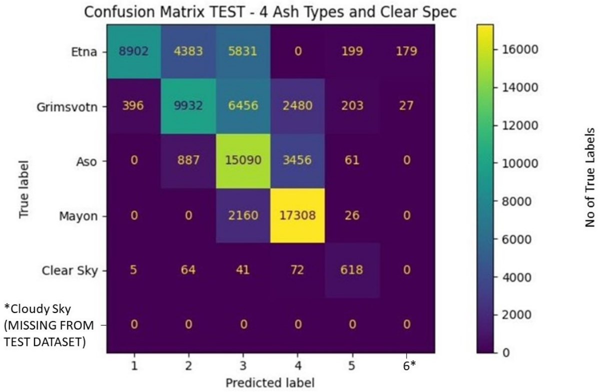

Figure 10: Confusion matrix using larger 2nd test dataset (Accuracy = 66%)

The validated model has improved the accuracy of the model predictions for the 2nd test dataset, albeit for only 6 classes. Similarly to Section 3.2, where more 'Clear' spectra were added, adding more ash spectra to both the training and the 2nd test datasets was attempted to study whether this prevented misclassification.

Increasing the number of ash spectra increased the computational time by a significant amount, and so training was attempted with little tuning of model parameters, and with a relatively small number of nodes compared to the number of spectra. A model with an 84% test score was trained, and when applied to the 2nd test dataset, also containing more spectra, there was an improvement in the accuracy up to 66% and less misclassification of the ash spectra.

The model created for Figure 10 was trained on 4 ash-types and 'Clear and 'Cloud' spectra, however due to an initial lack of simulation files of 'Cloud' spectra for the 2nd test dataset, the model was tested only on the ash spectra and 'Clear' spectra. This explains the unlabelled 6th row/column in Figure 10. Not having the 'Cloud' spectra in the 2nd test dataset actually served a purpose, indicating how many ash and 'Clear' spectra are misclassified as 'Cloudy Sky', which theoretically should not occur at all. Only around 1% of spectra are misclassified which is promising, and along with the improved classification of ash and 'Clear' spectra, these results helped justify re-training for all classes with these larger datasets.

A model with an initial test score of 87% was trained including all classes, with the results shown in Figure 11.

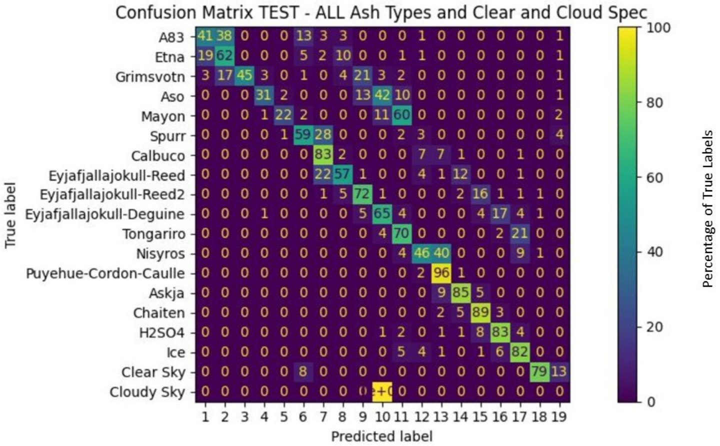

Figure 11: Confusion matrix using larger 2nd test dataset including all classes (Accuracy = 62%)

This model when applied to the larger 2nd test dataset was able to classify spectra with an accuracy of 62%. This has significantly improved the classification compared to earlier models, suggesting that the model requires a large training dataset in order to classify spectra to a good degree of accuracy. However, at the time of writing, there is an unexpected error in the prediction of the 'Cloudy Sky' class where all 'Cloud' spectra have been incorrectly predicted as the 'Tongariro' class, and unfortunately due to time constraints this could not be completed by the end of the project. The extent of misclassification suggests that there is an error in the training/test data as opposed to a genuine computational error by the model. Another feature is the 'secondary' diagonal above the leading diagonal. For certain classes, the majority of spectra are incorrectly classified in this 'secondary' diagonal. The cause of this is not immediately evident, and potentially further fine tuning of model parameters will be able to remove this. Nevertheless, by increasing the number of training spectra and the number of spectra in the 2nd test dataset, correct classification to a good degree of accuracy has been achieved for 14 out of 19 classes, resulting in an encouraging accuracy improvement from 13% to 62%. While these results are promising, it is yet to be seen whether this model is capable of predicting ash-type accurately of real spectra imaged using IASI data.

4 Conclusion

This project has been an investigation of the feasibility of machine learning in classifying spectra into a maximum of 19 classifications including 15 volcano based ash-types, 'Ice', 'Sulfuric Acid' and into 'Clear Sky' and 'Cloudy Sky' classes.

It is evident that it is feasible to train a spectral dataset to a good degree of accuracy with the final model attaining an accuracy in excess of 60%. However there are several complications when applying a machine learning model to a complex dataset. Without using a very large number of spectra, 19 classifications often proved too many for the model to accurately classify unseen data into. The reasonably well performing final model produced in Section 3.7 used more than 380,000 spectra for the initial training/testing dataset, leading to increased computational time and resources needed for model training.

With the existence of powerful tools such as sckit-learn, the applicability of machine learning is widespread, however during this project certain limitations were unearthed. Careful pre-processing and fine-tuning of model parameters is necessary to create an effective model, and this model will likely be very bespoke.

Further work in this area is strongly recommended however, firstly fixing the incorrect classification of 'Cloud' spectra. Using more versatile and more customisable machine learning models such as 'keras' or 'TensorFlow' as opposed to scikit-learn which is rather basic, may also add value [12]. Additionally, one of the key goals that existed for this project, to test the model on real world data, is one that should be explored in further work. The fully working version of the final 19-class model produced at the end of this project, in Section 3.7, could be applied to real retrievals to evaluate whether this model version benefits these processes.

While an accurate test of the model on IASI spectra was not carried out, there is evidence to suggest that with further work and a better understanding of how machine learning can be used for this particular purpose, that such a system can be implemented into the ash retrieval process, potentially leading to more accurate retrieval results, bettering our understanding of certain volcanic eruptions as well as how we should react to them.

4.1 Acknowledgements

RTTOV, Python and Scikit-learn were used frequently in this project.

Thanks to Dr Duncan Watson-Parris for his guidance with the Machine Learning, and a special thanks to my supervisors Dr Isabelle Taylor and Professor Roy Gordon Grainger for all their support.

References

[1] UK Civil Aviation Authority *A history of ash and aviation* Date Accessed : 13/03/2022 https://www.caa.co.uk/Safety-initiatives-and-resources/Safety-projects/Volcanic-ash/A-history-of-ash-and-aviation/

[2] BBC News *"Flight disruptions cost airlines 1.7bn, says IATA"*. 21 April 2010. Archived from the original on 12 May 2011. Retrieved 22 March 2022. http://news.bbc.co.uk/1/hi/business/8634147.stm

[3] Bye, Bente Lilja (27 May 2011). *"Volcanic eruptions: Science and Risk Management"*. Science 2.0. Archived from the original on 29 May 2011. Retrieved 3 Apr 2022.

[4] Thomas, H.E., Watson, I.M. *Observations of volcanic emissions from space: current and future perspectives* Nat Hazards 54, 323–354 (2010). https://doi.org/10.1007/s11069-009-9471-3

[5] A.J. Prata, C. Bernardo, *Retrieval of volcanic ash particle size, mass and optical depth from a ground-based thermal infrared camera* Journal of Volcanology and Geothermal Research,Volume 186, Issues 1–2, 2009,Pages 91-107,ISSN 0377-0273, https://doi.org/10.1016/j.jvolgeores.2009.02.007

[6] Larry Hardesty *Explained: Neural Networks* https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

[7] Le Bas, M.J., Le Maitre, R.W. Woolley, A.R. *The construction of the Total Alkali-Silica chemical classification of volcanic rocks.* Mineralogy and Petrology 46, 1–22 (1992). https://doi.org/10.1007/BF01160698

[8] Reed (2018) *The Complex Refractive Index of Volcanic Ash Aerosol Retrieved From Spectral Mass Extinction* https://doi.org/10.1002/2017JD027362

[9] Okánik (2021) *Classification of ash spectra in various environmental settings* Date Accessed: 13/03/22 http://eodg.atm.ox.ac.uk/eodg/vacation_reports/2021Okanik.pdf

[10] Mackie, S., Millington, S. and Watson, I.M. (2014), *"How assumed composition affects the interpretation of satellite observations of volcanic ash"* Met. Apps, 21: 20-29. https://doi.org/10.1002/met.1445

[11] Ishimoto, H., Hayashi, M., and Mano, Y.: *Ash particle refractive index model for simulating the brightness temperature spectrum of volcanic ash clouds from satellite infrared sounder* Atmos. Meas. Tech., 15, 435–458, https://doi.org/10.5194/amt-15-435-2022

[12] Muller, Guido (2016) *Introduction to Machine Learning with Python: A Guide for Data Scientists* O' Reilly Media

[13] Scikit-learn: *Machine Learning in Python* Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[14] Scikit-learn Documentation (2022) https://scikit-learn.org/stable/index.html Date Accessed: 03-04/2022

[15] Alexandre Deguine, Denis Petitprez, Lieven Clarisse, Snævarr Gumundsson, Valeria Outes, Gustavo Villarosa, and Hervé Herbin, *Complex refractive index of volcanic ash aerosol in the infrared, visible, and ultraviolet* Appl. Opt. 59, 884-895 (2020) https://opg.optica.org/ao/abstract.cfm?URI=ao-59-4-884

[16] Edward M. Patterson, *Optical absorption coefficients of soil aerosol particles and volcanic ash between 1 and 16 µm, in the 2nd Conference on Atmospheric Radiation* American Meteorological Society, 177-180, 1975.

[17] RTTOV Transfer Model Version 9 https://nwp-saf.eumetsat.int/site/software/rttov/

[18] *Data Preparation and Feature Engineering for Machine Learning* (2021) https://developers.google.com/machine-learning/data-prep/transform/normalization

[19] Bottou, Léon; Bousquet, Olivier (2012). *"The Tradeoffs of Large Scale Learning"*. In Sra, Suvrit; Nowozin, Sebastian; Wright, Stephen J. (eds.). Optimization for Machine Learning. Cambridge: MIT Press. pp. 351–368. ISBN 978-0-262-01646-9.

[20] SciPy *Savitzky-Golay Filter* https://scipy.github.io/old-wiki/pages/Cookbook/SavitzkyGolay

[21] Carniel, R., Guzmán, S. R. , 2020, *'Machine Learning in Volcanology: A Review'*, in K. Németh (ed.), Updates in Volcanology - Transdisciplinary Nature of Volcano Science, IntechOpen, London. 10.5772/intechopen.94217.

A Graphs and Tables

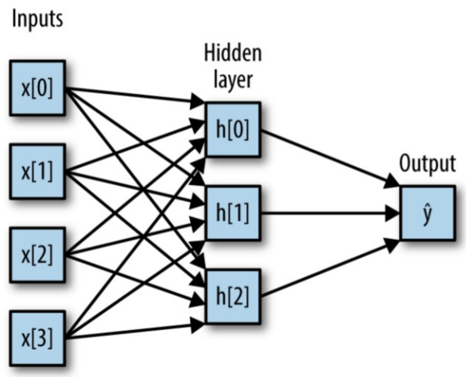

Figure 12: Neural network schematic with a hidden layer [12]

BASIC Dataset Spectral Classes	
Class No.	Class Name
1	A83
2	Mayon
3	H_2SO_4
4	Ice

Table 5: BASIC classification labels

First 4 Ash, Clear, Cloud Dataset Spectral Classes	
Class No.	Class Name
1	Etna
2	Grimsvotn
3	Aso
4	Mayon
5	Clear Sky
6	Cloudy Sky

Table 6: First 4 Ash, Clear, Cloud classification labels

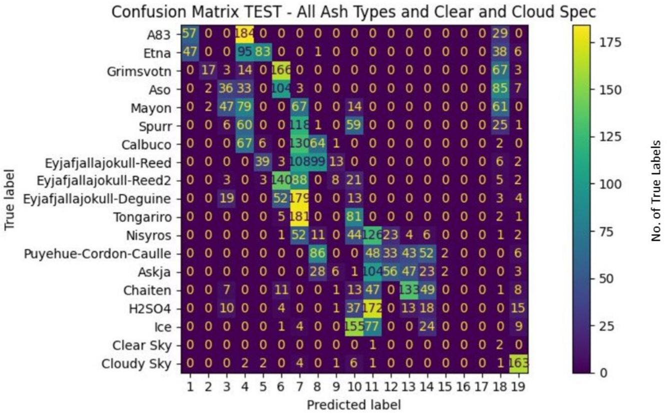

Figure 13: Confusion matrix for ‘alpha’ = 0.5 (Accuracy = 13 percent)

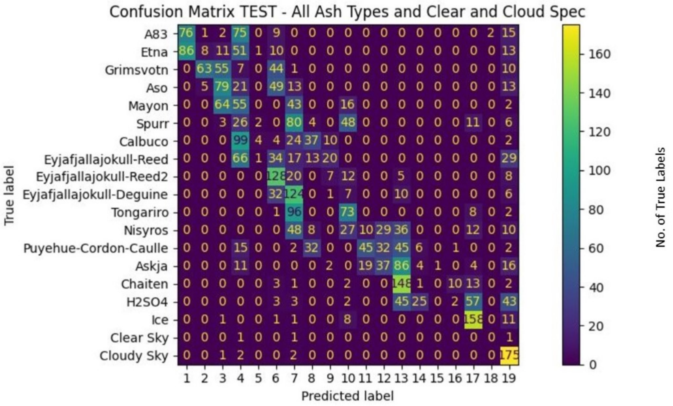

Figure 14: Confusion matrix for validated model using all 19 classes

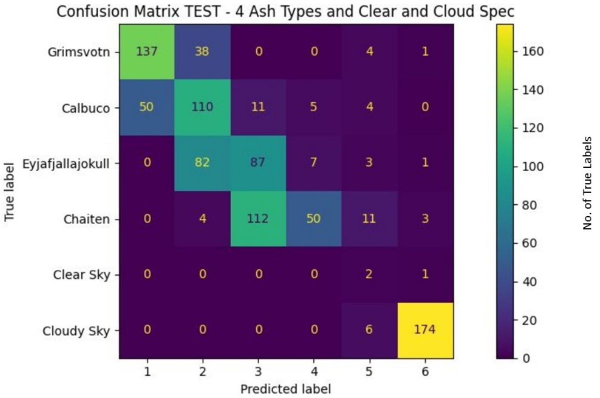

Figure 15: Confusion matrix for classes along the TAS diagram (Accuracy = 62 percent)