Data Engineering

Prof. Angelo Pio Rossi and Dr. Carlos Henrique Brandt

Master Thesis

Use of Machine Learning Techniques for Computational Spectral Index Extraction and Classification

Spectral Index Extraction and Classification

Supervisor: Prof. Angelo Pio Rossi and Dr. Carlos Henrique Brandt

---

Handed in by: Date:May/05/2020

Feven Legesse Mammo

Im Dorfe 21

28757, Bremen

Matriculation Number: 30002359

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter gives an outline of the problem statement, the aim of the work, and the structure of this work. The general introduction about the topic and the motivation to implement a machine learning technique for computational spectral index extraction and classification is explained in the first section. Thereafter, the aim of the work is described. The chapter ends with an overview of the general structure of the work.

General introduction and motivation of the work

Minerals are inorganic solid materials which we use in a different format for various purposes in our daily life routines. For example, a mineral called “gypsum” for building a house and a mineral “diamond” for jewellery and so much more. Mainly, minerals are the building block of rocks (Department, 2009). According to the International Mineralogical Association in 2018, there are 5,400 minerals are found on earth (America, Collector's Corner, 2020). Mars the second planet behind Earth, which is known as the “red planet”, its surface is rocky and mostly covered by red dust which is the reason for a thick layer of oxidized iron (Wild, 2015). Since the surface of Mars is somewhere among the basalt or andesite rocks on earth, the formation of minerals on mars has similarities with what is found on earth (Wikipedia, n.d.). CRISM (Compact Reconnaissance Imaging Spectrometer for Mars) is an instrument that is an imaging spectrometer with a scannable field of view that can cover wavelengths from 0.362 to 3.92 microns at 6.55 manometers/channels so that can identify a broad range of minerals on the Martian surface (Webmaster, n.d.). Spectral imaging is a sampling of data in many wavelength bands and it yields a three-dimensional data that is called data cube. The third dimension of the data cube represents the spectrum of each pixel. Spectral imaging can be divided into two, one is multispectral that samples data from different and discrete but disconnected wavelengths the other is hyperspectral imaging which samples enough data to reconstruct almost continuous or a connected spectrum over a given spectral range. The image

data (a data cube) I am going to work with is gathered through the CRISM instrument with its 9 summery products. (ALABAMA, 2020).

On the surface of mars below Syrtis Major (“an extended plateau on the planet Mars which is the darkest and noticeable formation pyroxene and olivine are stored up material in the solidified system (wikipedia.org, 2012). Again, there is a suggestion that Ni–Cu and PGE ores can be found there also. Mars pathfinder figured out that iron contains one-fifth of the weight of the soil which is an indication of the importance of iron-rich material on Mars. In the low land of Mars, Infrared Mineralogical Mapping Spectrometer detected the sulphate mineral kieserite (Mg-sulphate). But some mineral deposits are not found on Mars. However, recently nickel-iron meteorites (in number “three”) discovered by the Mars exploration rovers. Which those minerals are principal for steel manufacture (West, 2010).

Choosing an appropriate methodology or algorithm is necessary to obtain an accurate mineral potential map. Accuracy depends only on the capacity of the algorithm to learn multiform relationships among the input evidential features and mineral deposit occurrence. Also, two things must be considered: transparency and interpretability as seen in a journal under Ore Geology Reviews titled “Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines” (Rodriguez-Galiano, 2019). There must be attentive selection out of machine learning classification algorithms to do modelling, most importantly depending on the data entitled to trained by the machine learning algorithm. Based on this article explains about finding the best tool for the classification model out of Random Forest, Decision Tree, Support vector machine and Artificial Neural Network (ANN) algorithms the SVM and RF perform better. But still, most studies did not seek to understand the machine learning algorithms' performance during (at the time of) luck of training data. (V. Rodriguez-Galiano, 2015).

In Feb.2016 Institute of Electrical and Electronics Engineer (IEEE) published paper expressed that the Neural network ensemble (NNE) classifier is the most robust and efficient classification algorithm for Multispectral image classification. As it is described on the paper NEE is the best model comparison with Bayes Maximum-Likelihood Classifier and K-NN

(K-nearest Neighbours) Classifier based on the accuracy of each model on the multi-spectral image data took from SPOT image of Calcutta in India (Fu, 2016).

In general, since minerals have huge relevance to create something useful out of it for humankind or a whole living thing in the world, it is essential to discover more about it. On the point of mineralization on mars there is not so much known, yet which is an indication of a lot to discover. According to the exploration made by a machine (Robot) spacecraft; the knowledge about "mars mineralogy" is less than about mars geology (wikipedia.org, 2012). This research project expected to give a valuable contribution towards the Asteroid mining and Planetary Geology field through implementing and identifying a more robust and efficient machine learning classifier out of random forest and support vector machine.

Aim of the work

The main aim of this thesis is to understand about a machine learning technique for computational spectral index extraction and classification. For this purpose, a spectral image data cube which is gathered from a spectrometer called CRISM is used. On the way, the goal of this work in addition to this is to identify which area on the surface of mars accumulated minerals through analysing and using machine learning techniques. The specific goal of the work is described below:

- In the scientific aspect: to have a better understanding of the evolution and content of the universe.

- On the mining side: for extraction of minerals from the planet mars such as; iron.

- From a resource point of view: to upgrade the knowledge about what are the resources on planet mars.

Structure of the work

The thesis is structured as follows:

Chapter 2: Section 2.1 illustrates a general background about machine learning such as type of algorithm and learning mechanisms. Subsequently, the concept of different technique of the state of the art are described. Those are hyperparameters, model selection, underfitting, and overfitting, pre-processing. In Section 2.2 the idea of Imaging and its relation to machine learning and finally, the approaches of the two machine learning methods are explained. First, the Random forest in Section 2.3 followed by a support vector machine in Section 2.4

Chapter 3: This chapter illustrates the details of the implementation, such as software packages and frameworks and programming languages. Section 3.1 performs exploratory data analysis (EDA) of the data which has a subsection of data description and pre-processes. Afterward, Section 3.2 discusses the implementation of the two machine learning algorithms into data, and the evaluation procedure, then the chapter concludes by following the result and discussion part and summarizing the whole processes and some recommendation.

Chapter 2

Machine Learning

In this chapter overall about machine learning concepts such as supervised and unsupervised learning are covered. In the previous time throughout a long-time research activity related to this subject, the area is covered widely. Here, in this thesis, only a small section is considered. However, it covers the topics which are required to understand the specific machine learning approaches which are evaluated and to better determine the findings of the thesis. Firstly, for the reason of good understanding of the different type of the technique, a review about learning types is given. Afterward about the option of hyper-parameters is discussed because they must be chosen for a method which is evaluated within this work. To be able to select the

best model out of the methods within this work and it is necessary to know about how to evaluate and choose the best method, this topic is described in Section 2.1.3.inorder to explain and understand the results, also the challenges of the training model, the issue of underfitting and overfitting topic is explained under section 2.1.4. Thereafter, since pre-processing is a common task to accomplish before applying a machine learning model, major pre-processing techniques are demonstrated in section 2.1.5.

Above the general part, this chapter is focussing on the machine learning algorithms which are entitled to be used within this thesis. Those algorithms or methods are Random Forest and Support vector machine. These two machine learning algorithms are expressed briefly to give a clear view of how they perform and in what kind of data is favourable. And, it is necessary to know the basic idea, functionality, and algorithmic details of each method.

Machine learning overall

Machine learning algorithm uses training sets of real-world data to generalize models that are more specific and refined or practiced than humans. The major goal is to search a mapping or aligning from input patterns to an output. For example, let us consider that there is a data collected which can be useful for the prediction of being eligible or not eligible to be taken or considered as a graduate student in one of internationally recognized university or not. Then, the main aim of the algorithm is to learn from the pattern of the input data and label the new data to the corresponding output. This is to demonstrate that we humans teach a machine by using previously observed or gathered data and use the model to predict the new input data directly and accurately as possible to corresponding labels. This machine learning or training machine strategy is very different in a different task or according to the problem to be solved. In general what we humans do to make a machine operate as we want is, we just feed the machine a data which is called “training data” to teach the machine to learn different patterns and structure of the given data and the machine use that experience to predict or classify to the respective labels accurately as possible when we input a new data to that specific machine learning model. There are several methods of machine learning technics and algorithms and currently, this technology area is modifying by experts and scientists who are involving in it to be able to predict more efficiently and precisely (Ben-David, 2014).

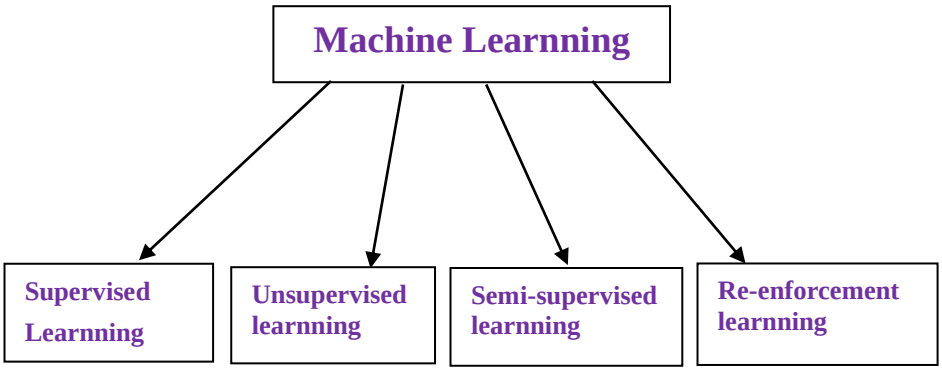
Certain cases are necessary to use machine learning algorithms or techniques; related to a task which is too complex to program, one is tasks performed by humans through a natural cycle which doesn't have a clear explanation how it is done but a machine learning programs for example in the course of speech recognition and image understanding it learns from past data and produce satisfactory results, and the other case is when the task is above human capabilities which means the ability of a machine to analyse very large and complex datasets very efficiently than humans, such as astronomical, weather prediction, electronic commerce datasets and so forth. In conclusion, one more thing about this technique is; a machine learning models have adaptive behaviour which implies when the program has been written and loaded it remains unchanged until redoing it. Nevertheless, many problems change over time, but that model must be reusable only for that specific task or which it is entitled to get a kind of new input data (has similarity with the training data which was given on) and predict. (Ben-David, 2014). Machine learning has a relation with other fields like computer science, Artificial intelligence, Statistics and probability, Data mining, and mathematical optimization. Naturally, machine learning is a discipline of computer science so human program a machine so it can learn. And, machine learning can be considered as a department of Artificial intelligence as a machine can detect meaningful patterns in complex data which is a basis of human intelligence. In case of probability, machine learning utilizes these theories to examine algorithms for pattern recognition which is the building block of machine learning. When we see data mining relations with machine learning both use the same approach to perform or accomplish different objectives. Based on available data machine learning focuses on making predictions whereas data mining discovering anonymous aspects. At last, there is a close connection between machine learning and optimization techniques. Predictions of machine learning are made on the root of available data. So, since optimization is the one who searches the best solution from all solutions for any task, mathematical optimization methods can be utilized to provide the right idea or premise to be used (Learnning, 2010).

Types of Algorithms

In simple words, learning means the capability to change depending on the external motive and recall most of our experiences. A machine learning algorithm can infer common laws and

learn their form with relatively high accuracy, but only if they influence the real data. All methods of machine learning mechanisms which are shown in figure 2.1 below are explained under it (Bonaccorso, 2018).

Figure: 2.1. Subdivision of machine learning



A. Supervised learning

Supervised learning includes building a statistical model for predict, or forecast, an output according to the stated one or more inputs. The trained system or the model must work with samples that have never been seen or used earlier. What it means is it is necessary to allow the model to generate a generalization capability and evade a common problem called overfitting. The concept of overfitting and underfitting will be discussed below in section 1.4.5. In general, supervised learning uses labelled data to train the machine. So, when the data is fitted to the machine learning algorithm the machine learns which feature associate with which label (Gareth James, 2013). There are two types of mechanisms in supervised learning: one is a regression which is the output variable is a continuous or real value. A change in one variable has a relation with a change in another variable. An example of this might be trying to predict a salary based on experience. The other is a classification method that identifies which category on object belongs to and generally this type of supervised learning output variable is categorical with two or more classes. For instance, when one tries to find out or predict if there will be a rise in salary for next year or not. Frequent application of these learning is Pattern recognition, spam detection, Natural language processing, Automatic image classification, Automatic sequence processing, sentiment analysis, and of course predictive analysis based on categorical classification or regression (Bonaccorso, 2018).

Major algorithms

- ♦ Decision trees
- ♦ Boosting
- ♦ K Nearest Neighbour
- ♦ Support Vector Machines
- ♦ Naïve Bayes
- ♦ Linear Regression
- ♦ Logistic Regression
- ♦ Neural Networks (Neves, 2015)

B. Unsupervised learning

In case of unsupervised learning, there will be inputs but there is no supervising output; even so, it gives an insight into the interrelation and formation of such data. In unsupervised learning, there is no labelled data, which implies a lack of a response variable that can supervise the analysis. When data is fitted to the machine, the machine identifies the pattern of the given data accordingly. In unsupervised learning, the algorithm grouped the data according to their patterns into clustering which is the machine make a group based on the behaviour of the data and association which discovers the probability of the co-occurrence of items in a collection (Silva, 2019). These are rule-based learning to find a relation between large data sets and variables. For example, “Clustering”: In the business area if one wants to know which customer made the same product purchase. “Association”: When we look at in business perspective; when one wants to find out which products purchase together. (Gareth

James, 2013) Common applications of unsupervised learning are in word clustering: speech recognition, machine translation, named entity recognition … etc.in document clustering: "text classification, information retrieval … etc".

Learning algorithms

- A prior algorithm for association rule learning algorithm
- Clustering algorithms

- Flat

+ K-means

- Hierarchical

+Top-down (Divisive)

+ Bottom-up (Agglomerative) (Neves, 2015).

C. Semi-supervised learning

Semi-supervised learning concept is based on the idea of supervised and unsupervised learning. Mostly there are an inadequate labelled data and adequate unlabelled data. The algorithm tries to find an analogy between the unlabelled and labelled data and based on their similarity predicts the labels of unlabelled data. This means, in other words, training the classifier by feeding labelled data and predicts labels of unlabelled data. An example of a real-world application of semi-supervised learning is webpage classification (Neves, 2015).

D. Reinforcement Learning

Reinforcement learning is about taking acceptable or satisfactory action to boost reward in a specific situation. Here there is no answer key or labelled data, but the reinforcement agent determines what to do to execute in the given problem. So, without a training dataset, it is

bound to learn from its occurrence or experience. That means the machine learns from the given feedback and try again to label or classify the data correctly. There are two types of reinforcement learning positive and negative both have advantages and disadvantages. Some practical applications of reinforcement learning are robotics, data processing and it can be used in a big environment when the model of the environment is studied but an analytic result is not accessible and just a reflection model of the environment is given. (Bajaj, 2020).

Hyper Parameters

Hyperparameters are mainly used to optimize the performance of the machine learning model in the automated machine learning task and all machine learning method has hyperparameters. Here two types of hyperparameter optimizations are discussed one is black box hyperparameter optimization and the other is multi-fidelity optimization. In black-box optimization mostly global optimization algorithms are preferred but to make some changes within some function evaluations that are often available some locality in the optimization activity is necessary. Black-box HPO has two methods: those methods are model-free black-box optimization method and Bayesian optimization methods. Both methods are discussed below.

Model-Free Blackbox Optimization Methods

Grid search or full factorial design is the main HPO technique. Here, the user defines a set of values for every hyperparameter, then grid search determines the cartesian product of these sets. One problem of grid search is the required number of function evaluations rises accordingly with increasing the resolution of discretization. Another way of search is a random search and this method works better than grid search in the situation when certain hyperparameters are more important than others and, they use easier parallelization and flexible resource allocation. The common population-based techniques are covariance matrix adaption evolutionary strategy (CMA-ES) and this strategy is one of the most competitive black-box optimization algorithms (Hutter, Hyperparameter Optimization).

Bayesian Optimization

Bayesian Optimization is a cutting-edge optimization outline for black-box function. It tunes deep neural networks for image classification, speech recognition, and neural language modelling. This optimization algorithm has two main components: A probabilistic surrogate model and an acquisition function to determine on which part should evaluate next. In contrast to evaluating the black-box function, the acquisition function is cheap to compute and can be completely optimized (Hutter F., 2019).

Multi-Fidelity optimization

In the area of HPO, the increment of dataset sizes and increment of complex models are the main difficulties in HPO since when backbox performance evaluation is very high or excessive. Nowadays single parameter training on large datasets consumes several hours and days. And, a usual method to boost manual tuning is an algorithm or hyperparameter configuration on a modest subgroup of the data. Multi-Fidelity optimization changes the manual way of hyperparameter configuration to a formal algorithm.

There are different applications of hyperparameter optimization methods to automate machine learning. The earliest adaptive optimization methods applied to HPO were greedy depth-first search and pattern search. Two of them developing over default hyperparameter configurations and pattern search developing over grid search too (Hutter, 2018/2019).

Model Selection and evaluation

Model selection and evaluation are a principal aspect of a machine learning workflow, and it is a good idea to understand a machine learning model evaluation technique and why it is very necessary. The importance of model evaluation is to answer the following two main questions.

- First, how and based on what criteria one should select an effective model for a specific task?

- To better see and understand how the model generalizes efficiently to unseen data.

It is obvious that before handling data it is recommended to plan before and use methods that match our needs. Here two types of model evaluation techniques are discussed those are holdout and cross-validation methods. Both techniques are performed in data not seen before which is called test data by the learning algorithm (Raschka, 2018).

Holdout

This technique works by testing a model on unseen data which means, unlike the trained data. So, in this procedure, the dataset divided into three parts: training set, testing set, and validation set (Stephen Coggeshall, 2019).

Training set: - a subdivision of a data set used to build a predictive model.

Test set: - some part of a data set that is unseen by the learning algorithm during data training that used for evaluating or determining how the predictive model performance is.

Validation set: - Also some parts of the dataset that used to evaluate the model performance during the training stage. Here parameter tuning is performed and will allow us to choose the best model. But the thing is all learning algorithms do not need a validation set (Gareth James, 2013).

Holdout approach of model evaluation mostly related to variability because dissimilarity among training and testing sets leads to a significant difference in the estimated accuracy. This method is applicable because of its speed, adaptability, and simplicity.

Cross-validation

This technique divides or partitions the real data set into training sets and an individual set of data that evaluates the analysis. The main and standard approach is k-fold – cross-validation. K implies the number of partitions in a given real data sets that are used for several cycles when one is used as a testing set and the rest uses as a training set and continues this routine until all or each k number of the data sets perform as a test set individually. Finally, the best model which performs better is chosen. This method is valuable because swapping the

training and test sets rises the effectiveness of the method (Mutuvi, Introduction to Machine Learning Model Evaluation, 2019).

To specify the model performance, evaluation matrices are needed. The method of model evaluation techniques is different according to the task. For example, the task might be regression or classification or clustering…etc. Some useful matrices for supervised machine learning problems are in the classification side; classification accuracy, confusion matrix, F-measure, logarithmic loss, and area under the curve. Whereas in regression; the common matrices are root mean squared error (RMSE) and mean absolute error (MAE) (Mutuvi, 2019).

Underfitting and Overfitting

Before discussing overfitting and underfitting let us consider what signal and noise mean in predictive modelling. The signal means the true underlying pattern or our real need for the dataset we want the learning algorithm to learn from. Whereas noise is unnecessary or irrelevant information in the dataset. So, A machine learning algorithm is useful to distinguish the signal from the noise in a dataset. The concept of noise data means when the algorithm is too complicated or flexible which means: if there are many input features or not well regularized. Then the algorithm learns from the noise data and instead of finding the signal or the true underlying pattern it reminds the noise data pattern. According to statistics goodness of fit means, how is the similarity between the model prediction with the true or observed values? A machine learning model who learned from noise data rather than the signal is taken as “overfit” because of the reason that the model fits the training dataset but has a bad fit over the new dataset. On the other hand, underfitting appears if the model is too simple or flexible which means a dataset with a small number of features or well regularized. In machine learning, bias and variance are prediction errors. A bias error is a reducible error were as a variance error is an irreducible and both are a reason for underfitting and overfitting (MAFIADOC, 1998).

Bias: Has a concept of how far is the predicted or the forecasted value from the actual or the real value. The bias error increases when the predicted values far off from the real values.

This kind of error occurs when the model is too simple and can't get the complexity of the data and this is what we call it underfitting data. to fix this problem or error one can try different methods. For instance: by adding more features to the dataset, by decreasing regularization and by adding more complexity through introducing polynomial features (MAFIADOC, 1998).

Variance: Has a concept of how dispersed are the predicted or the forecasted value from the real value. High variance is an indication of overfitting. So, in this case, a model shows a good performance on the trained dataset but not good in an unseen data set. One can solve the problem or the variance error by increasing the amount of training data, and by reducing input features and increase the regularization term (EliteDataScience.com, 2019).

EDA (Exploratory data analysis)

Data exploration is the very first step in any data analysis workflow. Any techniques of data visualizing and exploring except machine learning models and inferential statistics are called exploratory data analysis. This method of data analysis is very useful because we humans by nature find pictorial expression of data or things very understandable and self-expressive rather than registered data in number or any other structural format. EDA is used for:

- To better understand the data and point out any mistakes in it
- To find interrelation between given input variables in the dataset
- To determine the connection between the input and output variables
- Helping to choose the best model to the data from the beginning

In the dataset, the collected data might have categorical (qualitative) or numerical (quantitative) data. Mostly collected data are represented in row and column format together and the columns express subject identifiers whereas the rows express sample data collected based on the subject identifiers (Trevor Hastie, 1993).

Univariate: A dataset having one column.

Multivariate: A dataset having multiple columns.

Graphical: Summarizes or represents the data in a pictorial or diagrammatic way.

Non-graphical: Summarizes or reparents the data in numerical summary statistics way.

Numerical data: Represents in number and divided in to discrete and continues. Discrete refers to a countable item and continues to refer to value measured as intervals in a real numbers line.

Categorical Data: This is also called the nominal variable has two or more categories or groups. For example, A gender variable has two categories (Male and Female). A categorical data which is in an ordered form such as; “small, medium, large” is called ordinal data.

Graphical EDA for univariant numerical and categorical data: When we consider numerical data, histogram is a common way that tells a lot about the data. One can get information related to a central tendency, spread, modality, shape, outliers and generally to see the shape of the distribution. Instead of histogram one can use stem and leaf plot. This plot doesn’t hide any information and it shows the shape of the distribution with all data values. Another important graphical method is the boxplot. Boxplot is a robust and effective way of expressing outliers, central tendency, symmetry, and skew of the data. To get the most out of the box plot it is a good idea to plot side-by-side. One more plot of univariate graphical EDA is quantile-normal plots (QN) or quintile quantile (QQ) plot). This plot has a different interpretation and is useful for the diagnosis of kurtosis and skewness and the detection of non-normality. Most importantly it is applicable for analysing “residuals”. (Statistics point of view, residual means “result error in linear regression model”). The major method to represent categorical data in graphical form is a histogram which is a bar plot of registered data in table form and a pie chart is also used but not often like a histogram.

Non-Graphical EDA for univariant numerical and categorical data: The exploratory analysis is a first calculation about the population distribution of the feature or the variable. Some

calculations which are going to perform here are the variable central tendency, spread, modality, shape, and outliers. Additionally, statistical measurement can be performed. Such as mean, variance, standard deviation, skewness, and kurtosis. The main central tendency measurement is mean and after that the median and mode. But to see the outliers the median would be chosen. To find out about how far the data from the center is (to measure the spread of the data) calculating variance, standard deviation and interquartile range (IQR) is the major method. IQR is more robust for spread measurement. Kurtosis and skewness are the measurements of peaked-ness and asymmetry of the data respectively. For categorical data putting the frequency of each category in a table, the format is a good way of expressing the data.

Graphical EDA for multivariate numerical and categorical data: Common way of representing multivariant categorical data graphically is a bar plot. Besides the bar plot, a side-by-side box plot is a good way of analysing the connection between a categorical feature and a quantitative feature, furthermore the quantitative variable distribution at each degree of the categorical variable. In the condition of two or more quantitative variables scatter plot is the most useful technique.

Non-Graphical EDA for multivariate numerical and categorical data: In general, non-graphical EDA methods are useful to see the relationship between variables and a dataset. Cross-tabulation, correlation, covariance, and correlation matrix is the common way of expressing the data numerically. Cross-tabulation is very necessary for categorical data with a few, unlike variables. When we take quantitative data, calculating correlation and covariance of the data is the major way of a non-graphical multivariant dataset. This method checks the relationship between each variable and gives a value between +1 and -1. The out came of this technique tells if there is a strong positive(direct) or negative(indirect) relationship or in between of two which is neutral (zero) or no relationship at all among the variable. (Seltman, 1999).

Data Pre-processing

After EDA is performed, the next step would be preparing the data or in other words, after exploring the data and have a piece of good information about it such as identifying what kind of data it is and the behaviour of the data in general, the pre-processing stage continues. Data pre-processes mean cleaning the data to make it clear from noise, inconsistency and different problems to be useful for the intended use. Especially nowadays data are collected from different sources and are very messy so not easy to understand and use them for the specific task. To be able to use collected data properly and efficiently the data should be stored inappropriate way. Then one can extract the wanted data and use it for the work of interest. The use of data pre-processing in different aspects is described below (R, 2018).

Data quality point of view: The data have a quality when it fully feels the condition of the predetermined task. Based on the accuracy, completeness, consistency, timeliness, believability, and interpretability of the dataset one can check the quality of the data. Data with low quality have low-quality results. Machine learning point of view: For achieving better results from the applied model through improving the overall performance of the machine learning model by preparing the data as per the specific need of the machine learning model (G. Goos, 2004). Data mining point of view: - to improve the effectiveness and simplicity of the mining processes.

Main tasks in data pre-processes

A. Data Cleaning: - involves smoothing noisy data, recognize and removing outliers, sort out, and clarify inconsistencies and filling in missing values.

B. Data Integration: - This means combining data from one or more data stores. Proper integration is good to reduce or avoid inconsistencies and redundancies in the data set. This can improve the speed and accuracy of data mining processes.

C. Data Reduction: - Acquire a minimized or reduced representation of the data set but approximately show the same result. It involves dimensionality reduction and

numerosity reduction. Dimensionality reduction is to get a compressed representation or form of the original data set by applying data encoding schemes. Some of the data compression techniques are principal component analysis and wavelet transforms. On the assumption of numerosity reduction, the concept is replacing the data by a smaller version of it using parametric (E.g., log-linear models or regression) and non-parametric models (e.g., data aggregation or sampling, histograms, clusters).

D. Data Transformation: - Consists of data normalization, discretization, and concept hierarchy generation. Data transformation has a huge contribution according to successful data mining processes (Jiawei Han, 2011).

Digital Image Processing

Introduction

Digital image processing is processing digital images by using a digital computer. A digital image is made of a finite number of elements called pixels and each pixel element has its location and value. Some operations are applied to digital images: to extract valuable information, to analyse it and make decisions, and to improve the quality of an image. Digital image processing earliest application was in the newspaper industry where pictures sent by submarine cable among London and New York. Nowadays digital image processing can be applied in so many areas for instance: robotic vision, automotive safety, artificial intelligence, medical image analysis, image restoration, and enhancement and so much more.An image is a matrix of pixels organized in columns and rows, which means represented in 2-dimension (h*w) array of pixel values and each pixel has information of intensity and color and has an individual value ranging from 0 to 255. Image data are can be formatted into a raster image (e.g. jpg, gif, png, tiff) and vector image (e.g. pdf, eps, emf). Raster (bit map) images are pixel-based graphics and resolution-dependent whereas vector images are curve-based graphics and resolution-independent. There are RGB images that have three channels and a grayscale image with one channel. The three channels of RGB are red, green, and blue whereas in grayscale in one channel from the pure white to the pure black and in between

different greyscale levels are encounters. Those image representations are based on human vision. Representative colour images that lie outside human vision are taken through imaging machines that cover the whole electromagnetic spectrum starting from gamma to radio waves that are exactly invisible to us. That electromagnetic spectrum can be utilized on images such as computer-generated images, ultrasound, and electron microscopy. In general, there are three parts of computerized processes: low-level processes, mid-level processes, and high-level processes. Low-level processes consist of operations for example Image pre-processing to reduce noise, image sharpening, and contrast enhancement. Mid-level processes contain image segmentation (dividing or partitioning an image into objects) so, here the input is an image, but the output can be the identity of an individual object or any other attributes taken out from those images. High-level processes include recognition of individual regions or objects in an image. The electromagnetic energy spectrum is the principal energy source for the image. In addition to this, another useful source of energy is ultrasonic, electronic (like electron beams used in electron microscopy) and include. An image generated by a computer that used for visualization and modelling is a synthetic image. Images built on radiation from the EM spectrum are common mainly in the X-ray and visual bands of the spectrum (Woods, 2002).

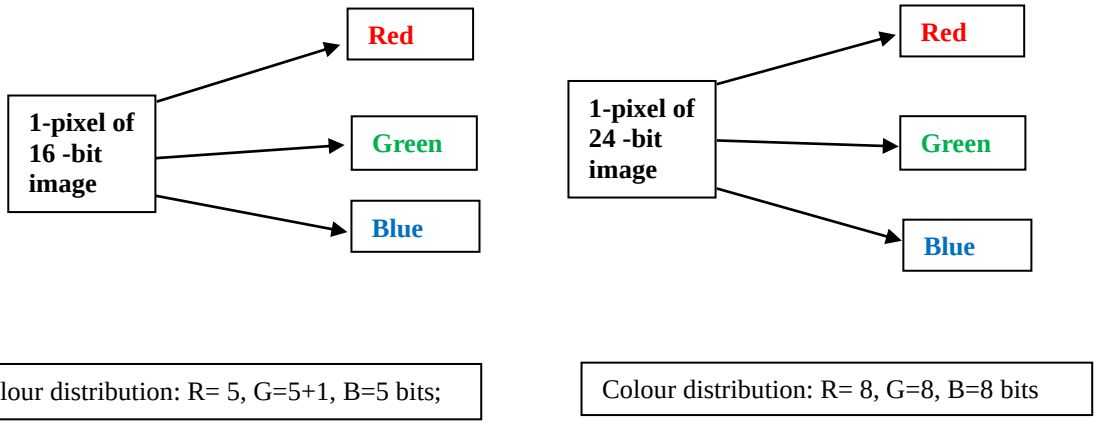
Types of image

A. Binary image: is a 1-bit image, it takes one binary digit to represent each pixel and it has two-pixel values. Only black and white color no gray level or in between. Mainly created from a gray-level image by using threshold operation.

B. 8-bit color format image: known as grayscale (1 color) image and have 256 various shades of color.

C. 16-bit color format image: high color image format and contains 65,563 different colors in it.

Figure: 2.2 Colour Image



As it is shown in the above fig. 2.3, A 24-bit and the 16-bit image have three different matrices of R, G, B which is red, green, and blue respectively. Considering a 16-bit image, the color distribution might vary, and green has a plus 1 because out of all three colors green is the quietest or soothing to eyes. The 24-bit color format is a high color image format (Tutorialpoint, 2020).

D. Spectral images: Before defining or explaining what spectral imaging is let's define some terms related to it.

Electromagnetic spectrum: - Contains all forms of electromagnetic radiation and each electromagnetic radiation belongs to a different band of the spectrum. This means when the spectral bands are grouped, they form spectrum ranging from highest energy (gamma rays) to lowest energy (radio waves) and called the electromagnetic spectrum. Electromagnetic bands are not distinct they are transformed smoothly from one to the other. Below describe all bands of the electromagnetic spectrum (B.B. Singh, 2005).

a. Gamma-Ray Imaging: - The main imaging based on gamma-Rays is astronomical observations and nuclear medicine. In nuclear medicine, the image is used to locate the area of bone pathology, for instance, tumours or infections.

b. X-Ray Imaging: - Mostly it is used for medical diagnostic and sometimes in industry area and astronomy.

c. Ultraviolet Band: - Can be applied in various areas. For instance: microscopy, biological imaging, astronomical observations, lasers, industrial inspection, and lithography.

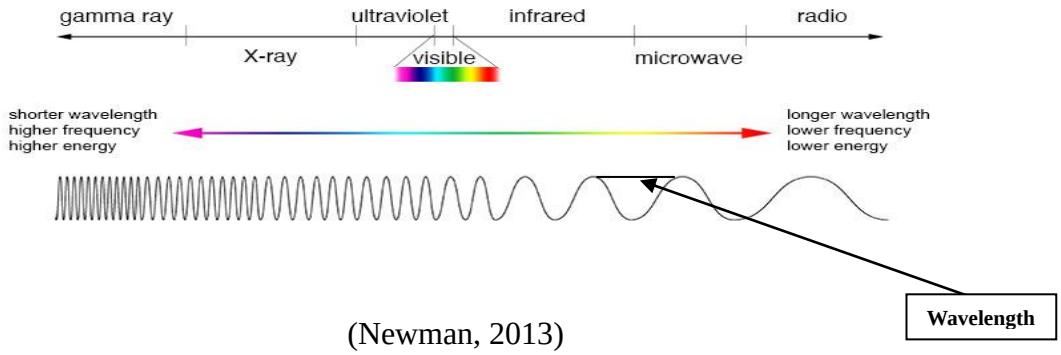
d. Visible and Infrared Bands: - Mostly applied in the area of astronomy, remote sensing, industry, law enforcement, and light microscopy.

e. Microwave Band: - Imaging application of microwave band is radar and the distinctive thing about imaging radar is its ability to gather data over virtually at any region at any time in any case of weather condition.

f. Radio band: - Imaging in the radio band of the main application is in medicine (MRI) and astronomy (Woods, 2002).

Wavelength: - Forms or designs of electromagnetic radiation (like radio wave) make quality patterns as they travel through space and those waves have their shape and length. So, the range or distance between the wave picks is called wavelength (Woods, 2002). Below Fig 2.4 shows the electromagnetic spectrum with all bands.

Figure: 2.3 Electromagnetic spectrum



(Newman, 2013)

Imaging that uses several bands through the electromagnetic spectrum is called spectral imaging and it combines two things spectroscopy and photography (spectrometer) to sample

image data at various wavelength bands. Spectroscopy focuses on investigating how light behaves in the target and identifies materials based on their distinct spectral signature. On the other hand, a spectrometer is an apparatus that splits the incoming light into a spectrum. Spectral imaging is commonly divided into hyperspectral and multispectral imaging. The difference between them is; over a given spectral range hyperspectral imaging samples data from a continuous spectrum whereas multispectral imaging samples data in a distinct spectrum Spectral imaging provides 3-dimensional data called data cube which the third dimension represents the wavelength and it is designed or created from many wavelengths. The wavelengths individually have their information that provides. Generally, spectral imaging (hyperspectral and multispectral imaging) gives more and detailed information about the target material or object than the RGB image give (specim, 2020).

The main purpose of Image processing is visualization, image sharpening, and restoration, measurement of a pattern (measures different objects in an image), image recognition and image retrieval (retrieve the image of interest or see of the needed image) (Rebeca González, 2020).

Nowadays machine learning in image processing has a huge benefit by making a complex image more understandable. Mostly machine learning is helpful when the processing data is huge which has a high dimension. Some uses of machine learning in Image analysis are to conclude the necessary information of the visual data. for instance: segmentation, registration, recognition … etc. and when the image data has a large variation and complex. Machine learning uses statistical methods to get an approximate answer or solution for some problems related to image data; such as to the reconstruction of 3D from a single image, and when it is hard or very complex to model e.g. scene classification. Some of the applications of machine learning in image processing are; image segmentation and classification, object recognition in an image such as face recognition, and object detection. Below the most popular machine learning supervised learning for classification algorithms are discussed briefly (Anon., 2016).

Image pre-processing

Image pre-processing is done before feeding the image to the machine learning algorithm. This is because mostly the data collected from different sources and appears in different ways which is messy. So, they should be cleaned and standardized then the learning algorithm can learn the necessary or right pattern from the data. In general, cleaned and pre-processed data have a high impact when it comes to efficiency and accuracy of a learning algorithm which means when a cleaned data fits into the algorithm the produced model will be more accurate and efficient. Techniques and algorithms of image pre-processing with its application and performance depends on the device used, type of image measured, and the fact or information expected to get from the analysis. There are two disciplines in data pre-processing; data cleaning and data augmenting.

Data cleaning: pre-processes the data in the way that the algorithm can take easily and understand the real pattern of the data. Common applications during the data cleaning step are; rescaling, grey scaling, sample wise std normalization, feature-wise centering, and feature-wise std normalization.

Data augmenting: Occasionally dataset shortage occurs for a deep model to learn sufficiently. In this case, augmenting the data method is important to solve the problem by transforming or changing every individual data in many possible ways it can be, and adding those transformed data into the data set. So, the dataset can increase effectively. Mostly the data are chosen randomly during transformation is applying. Common data augmentation techniques are rotation, horizontal shift, vertical shift, shearing, zoom, horizontal flip, vertical flip, and combination (zone, 2017).

Spectral image pre-processing

As image pre-processing, in general, is the most important and required task to do before starting any image data analysis to treat many issues in an image such as dead pixels, compression of the images, spiked points, and background removal. It is a good idea to see the main methods to solve the above problems to the spectral image. Most of the software

packages provided with the hyperspectral devices include different methodologies for image pre-processing (data cleaning') (Vidal, Pre-processing of hyper spectral images. Essential steps before image analysis, 2012). Let us see the common techniques of image pre-processing.

a. Image compression: - the two types of spectral images hyperspectral and multispectral images mainly composed of thousands or even millions of data. So, these data need huge storage space and to put the data in a manageable way the image should be compressed into a small size (Vidal, Pre-processing of hyper spectral images. Essential steps before image analysis, 2012).

b. Background removal: - during the selection of regions of interest (ROI) geometry of the samples in the acquiring of the image plays an essential role. When the acquired sample image fails to cover the scanned area, the left-out area must be discarded. This uncovered area is noisy spectra. Erasing of this area would be performed in different techniques for example manual selection of ROI, by using a histogram and manual selection of the threshold value (Vidal, Preprocessing of hyperspectral images. Essential steps before image analysis , 2012).

c. Dead pixel: Mainly dead pixel is created by anomalies in the detectors and those pixels in hyperspectral image determined as "dead" pixels. So, because of this distortion of the multivariate model happens. To handle this problem, dead pixels should be located using genetic or evolutionary techniques and thresholding.

d. Identifying and handling spiked points: Spikes mean a sharp rise succeed by a sharp decline in the spectrum and it happens because of abnormal characteristics of the detector or environmental conditions like cosmic ray events. The main spike detection technique is by manual supervision and this technique is harder and time-consuming. The other technique is based on the nearest neighbouring pixel comparison (Vidal, Preprocessing of hyperspectral images. Essential steps before image analysis , 2012).

e. Spectral pre-processing: majorly to eliminate the effect of irrelevant phenomena affecting the spectral measurement, such as detector art facts and surface roughness. The common way of spectral pre-processing methods is derivatives, de-noising, and scatter correction.

f. Outlier detection: outliers are observations or data points that are distinct from most of the data. When data are more heavy-tailed distribution good detection can be obtained by Gaussian distribution of outlier detection (Vidal, 2012)

Radom Forest for classification

Random forest is one of a tree-based method that is a supervised machine learning algorithm for regression and classification problems. Decision-tree techniques have a set of dividing rules that used to classify the predictor space able to be summarized in a tree. Decision tree methods are simple to explain and applicable. In respect of prediction and accuracy, tree-based methods are not competitive or not a good choice over the other best-supervised learning approach. In general, there are three types of decision tree-based algorithms or methods: bagging, boosting, and random forest (MAFIADOC, 1998). These techniques are producing various trees which they combined to produce a single agreement prediction. Since this thesis is focusing on classification techniques, the discussion below is focusing on a classification approach of random forest algorithm. A classification tree has high similarity with a regression tree and the difference between them is the classification tree is to predict a qualitative response whereas a regression tree is to predict a quantitative response (MAFIADOC, 1998). A classification tree predicts the observation where it belongs, by extracting or collecting the most commonly occurring classes of training observations in the region to which it belongs. The advantages of decision trees are they are easy to describe to people and can easily understand and explain by a non-expert. Additionally, qualitative predictors can be handled simply without creating dummy variables. On the other hand, the disadvantage of trees relative to the other regression and classification methods is; predictive accuracy is not higher than the others. Nevertheless, by combining decision trees with other techniques like bagging, random forest, and boosting it is possible to upgrade the predictive performance of it (Gareth James, 2013).

Since the random forest method is improving on top of the decision tree, before exploring random forest it's a good idea to understand the technique of decision tree. Below a sample of 10 patients' datasets with 5 features is provided. By implementing a decision tree algorithm to the dataset, classification techniques of the algorithm are explained clearly. "Please note that the dataset is not real, instead just created to explain how decision tree algorithm works (classifies)".

Terminologies in decision trees

- The very top node is called "root node"
- The mechanism of dividing a node into two or more sub-nodes is called splitting.
- The sub-nodes divide into other sub-nodes is called a decision node.
- The reverse processes of splitting are pruning which implies, removing sub-nodes of a decision node.
- A Member of the entire tree is called sub-tree or branch.
- The parent node is when the node divided into sub-node; on the other hand, sub-nodes are the child of a parent node (Chauhan, 2020).

Task A: Based on the below categorical training dataset of "COVID-19", classify whether a new patient's condition into serious or not. In the dataset, the dependent (response) variable is "COVID-19 condition serious" and the independent (predictor) variables are "Fever, Cough, Tiredness and Breathing difficulty". In another word, the task is creating a decision tree that uses Fever, Cough, Tiredness, and Breathing difficulty to predict whether a patient is in serious or mild condition.

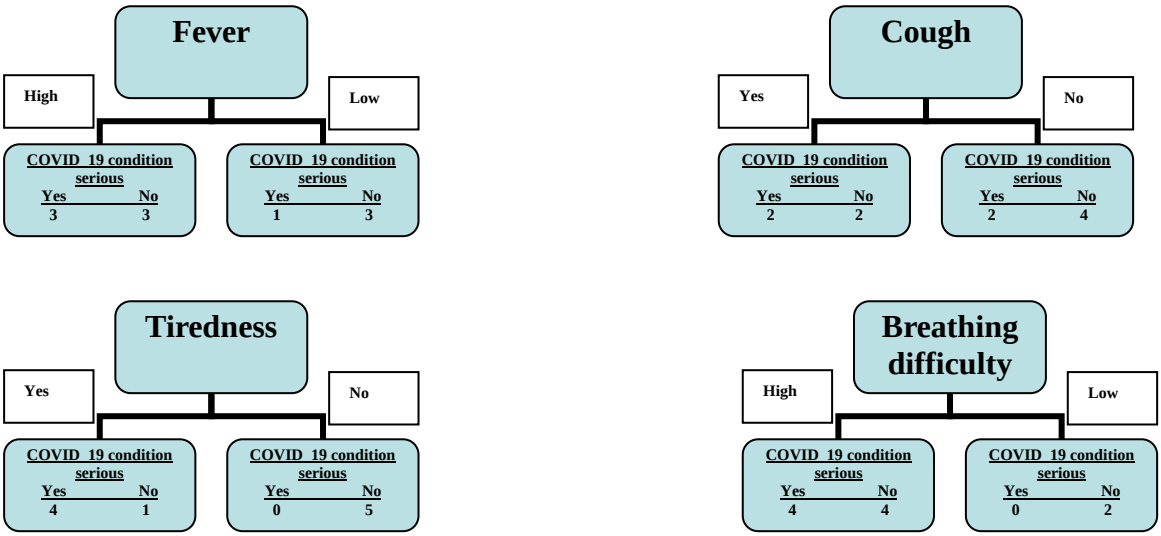
Table: 2.1 COVID-19 Patients Dataset

Person	Fever	Cough	Tiredness	Breathing difficulty	COVID_19 condition serious
P_1	High	Yes	Yes	High	Yes
P_2	Low	No	Yes	High	No
P_3	High	Yes	No	Low	No
P_4	High	No	Yes	High	Yes
P_5	Low	Yes	No	High	No
P_6	Low	Yes	Yes	High	Yes
P_7	High	No	No	Low	No
P_8	High	No	Yes	High	Yes
P_9	High	No	No	High	No
P_10	Low	No	No	High	No

Steps to solve the task

1st step. check how each independent variable classifies the sample patient's condition individually.

Figure: 2.5 Decision tree of each predictor variable



As we can see from the above figure none of the leaf nodes are 100% "Yes" /serious condition or "No"/mild condition, so they all are considered "impure". There are six different ways of determining the best attribute (leaf node) which separates the patient's into "serious" and "mild" condition. Six of the methods with their short definition is described below.

1. Entropy: it measures the randomness of the information processed and the higher the entropy the stronger the concluded information. A branch with entropy "0" can be

determined as a leaf node whereas a branch with "< 0" determined to be needed further splitting.

2. Information Gain (IG): focussing on attributes that yield the highest information gain with the smallest entropy.

3. Gain Ratio: Out of the attribute it chooses who has a larger number of different values. It improves information gain by considering the inherent information of split into account.

4. Reduction invariance: mostly for regression (continuous) problems and uses the formula of variance to select the best split. A split with a lower variance would be chosen as a principle to split the population.

5. Chi-Square: this is the oldest of tree classification techniques and it is preferable to work with the categorical target variable like; True/False or Yes/No.

6. Gini Index: Easy to implement and best for categorical target variables like; Male/Female.

So, by doing operations using one of the techniques described above additionally considering the data type (categorical or continuous) the algorithm determines which attribute should place at the root or on different levels of the tree as internal nodes (Chauhan, 2020).

For this specific task (example) we are going to use the simplest and easy to understand method called Gini Index. The formula to calculate Gini is described below.

Gini = 1-(probability of "yes")^{2} - (probability of "no")^{2}

By using the above formula let's calculate the Gini impurity of all the attributes one by one.

Fever => High = 1- $(3/3+3)^{2}$ – $(3/3+3)^{2}$ = 0.5

Low = 1- $(1/1+3)^2 – (3/1+3)^2 = 0.375$

Total Gini impurity = (6/6+10)0.5 + (4/4+10)0.375 = 0.29

Cough => Yes = 1-$(2/2+2)^2 – (2/2+2)^2 = 0.5$

No = 1- $(2/2+4)^2 – (4/2+4)^2 = 0.3$

Total Gini impurity = (4/4+10)0.5 + (6/6+10)0.3 = 0.25

Tiredness => Yes = 1-$(4/4+1)^2 – (1/1+4)^2 = 0.32$

No = 1-$(0/0+5)^2 – (5/0+5)^2 = 0$

Total Gini impurity = (5/5+10)0.32 + (5/5+10)0 = 0.032

Breathing difficulty=> High = 1-$(4/4+4)^2 – (4/4+4)^2 = 0.5$

Low = 1-(0/0+2)2 – (2/0+2)2 = 0

Total Gini impurity = (8/8+10)0.5 + (2/2+10)0 = 0.4

Out of those total results, the attribute with the lowest value should be chosen as a root node and has an assumption of value with the lowest value separates each sample better than the other attributes to their categories accordingly. And to put the rest of the attributes as a leaf node also the algorithm follows the same trend. So, in this case, “tiredness” attribute is the lowest of all with the value of “0.032” and it can be a root node.

As we can see from the tree of “tiredness” the left side it classifies properly as “when if the patient is not filling tired it can be determined as the patient is in a” mild” condition which is not serious. On the other hand, the tree didn’t classify the left side (when the patient is feeling tired) properly. It shows 4 serious conditions and 1 mild (not serious). The next step would be classifying 5 (p1, p2, p4, p6, p8) of the patients which feels tired using the rest of the attributes. To decide which attribute should be used as a leaf node on the right side of

tiredness Gini impurity should be calculated. Since, how to calculate Gini impurity is explained above no need to show the calculation any more so, here is the Gini impurity values of the three attributes: Cough = 0.1875, Fever = 0.78, Breathing difficulty = 0.064. As we can see Gini impurity of Breathing difficulty is the smallest of all, so it should be the leaf node that will classify 5 of the patients.

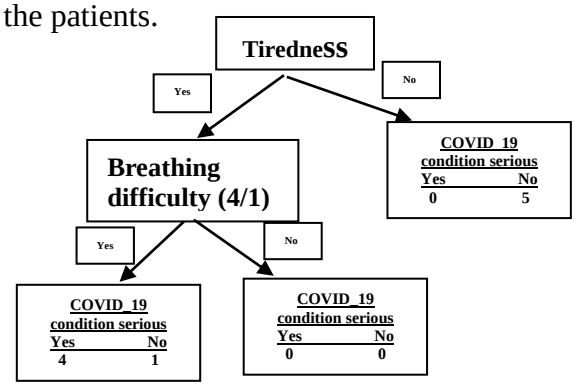


Figure: 2.5 DT_1

Since Breathing difficulty classifies the same as "tiredness" in the right leaf, no need to do Gini impurity to decide which attribute should be taken as a leaf node on the left side of breathing difficulty because it comes with the same result. When we compare the Gini impurity result of cough and fever, cough result is the smallest so cough attribute will be the leaf node to classify the left side of breathing difficulty result.

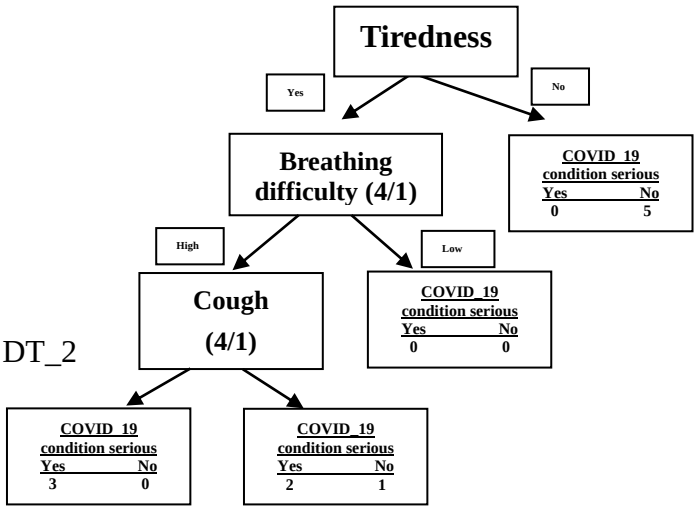


Figure: 2.6 DT_2

Here, as we can see from the tree above "Cough" classifies properly on the left side and it shows impurity on the right side. Now, the "Fever" attribute is left to classify the rest patients. Let's see how Fever will classify.

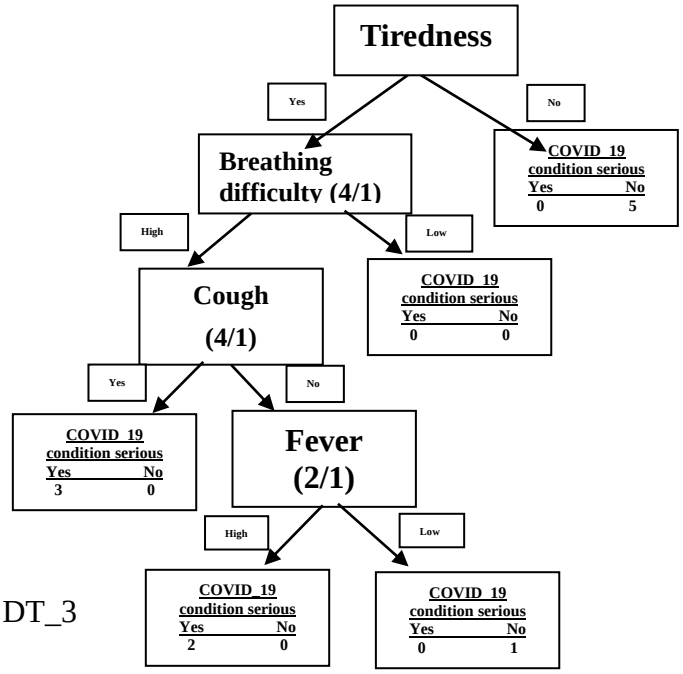


Figure: 2.7 DT_3

As we can see from the above tree, “Fever” classified the rest 3 patients properly and now we can say the decision tree model is built already and we can test how it predicts when there is new data.

First, the model checks if the new patient felt tired, if so then it asked again if the patient had breathing difficulty or not, if so then checks if the person was coughing if so, the patient is in serious condition and if no the model checks if the person had a fever or not and if so he/she is in a serious condition and if not the patient is in mild (not serious) condition. From the beginning, if the patient didn’t feel tired the model predicts the patient is in mild condition. In “Breathing difficulty” if the patient didn’t have difficulty, the model goes to check in the “yes” condition because at this stage the algorithm shows no answer. In the "cough” stage, if the person does experience coughing the model predicts the patient is in serious condition.

Based on the description above how the new decision tree model performs the table below shows the sample prediction of the model in new data, the model predicts a new patient data which is “Yes, in serious condition” in case of P-12 and “No, in mild condition” in case of P-11.

New patient data for prediction

Person	Fever	Cough	Tiredness	Breathing difficulty	Covid_19 condition serious
P-11	High	No	Yes	Low	Yes
P-12	Low	No	Yes	High	No

So, let's see how the decision tree performs when there is missing data and in feature selection processes.

Feature selection

For example, Imagine attribute "Cough" didn't give a reduction in impurity score so, in such a case the algorithm is not going to use "Cough" to split the patients and is not going to be part of the tree and this called automatic feature selection. Alternatively, a threshold can be used such that to make a big difference the impurity minimization has to be large enough. So, in this way finally, a simple decision tree model with not overfit is what we obtain. As a result of a decision tree, overfit means when the decision model does well in the real or original data which we have used to build the tree but not with new data. In general, this is what we call it to feature selection when the algorithm doesn't use one or a couple of attributes in a data set because of the above-explained impurity reasons.

Missing data

When there is missing data in one or more of the given attributes in a dataset, we can handle this in a couple of methods. One way is using the most common recorded value in that specific attribute and fill it in the missed place. The other way is finding another column (attribute) that has the highest correlation with the column (attribute) which has missing data and use that as a guide to filling in (Trevor Hastie, 1993).

Random forest

Decision trees are easy to use, build, and interpret but not an excellent algorithm for predictive learning, which implies that there is an overfitting problem that makes the model inaccurate. One of the overfitting handling mechanisms of the decision tree is a random forest. The random forest method is very useful to get better predictive performance. This

technique combines various machine learning algorithms for better accuracy. The idea behind the name “random” is because; when building trees, it uses a random sampling of training data set, and during splitting nodes, a random subset of features is considered (Chauhan, 2020). Let us see its mechanism during a classification task. For this task we are going to use half of the same dataset with the decision tree (**Table 2**), half because for simplicity reason and same task (**Task A**).

Table: 2.2 COVID_19 for random forest

Person	Fever	Cough	Tiredness	Breathing difficulty	COVID_19 condition serious
P_1	High	Yes	Yes	High	Yes
P_2	Low	No	Yes	High	No
P_3	High	Yes	No	Low	No
P_4	High	No	Yes	High	Yes

Steps to build a random forest are:

Step_1. Setup a “bootstrapped” dataset: randomly selecting a sample data from the original dataset to create the same size as the original. Note that, it is possible to choose the same sample more than once.

Table 2.3. Bootstrapped dataset

Person	Fever	Cough	Tiredness	Breathing difficulty	COVID_19 condition serious
P_2	Low	No	Yes	High	No
P_4	High	No	Yes	High	Yes
P_1	High	Yes	Yes	High	Yes
P_1	High	Yes	Yes	High	Yes
P_5	Low	Yes	No	High	No

Step_2. Choose a random subset of attributes and build a decision tree using the bootstrapped dataset. Here, 2 randomly selected attributes are used at each step, and for example, we choose “Fever and Tiredness” and let’s say “Fever” did a good job so it is a root node. Next to choose a leaf node we again going to choose two candidate attributes except “Fever”. Let’s say “Breathing difficulty and cough” and continue building the tree as usual by considering a random subset of attributes at each step. In general, these two steps are called bagging (Geer, 2019).

And, go back and build a new bootstrapped dataset and repeat step two. In general, this whole process might be repeated more than 100 times. The resulting different type of tree makes the random forest more accurate and effective than the normal decision method tree. Once the model is created, we can predict with new data; first, run the data to the first tree, then the second tree and up to all the trees record all the results from the tree. To decide whether the patient is in a serious or not serious condition we will choose the most frequent answer from the decision trees. That means if it is yes, take “yes” and if it no takes “no” as a final answer. Then we can evaluate the built random forest model if it performs well in a new dataset or not. First, as we saw the whole process of building a random forest, ¼ of the dataset didn’t use in the bootstrapped dataset so there will be one or more (if the dataset is larger) data that is not part of the bootstrapped dataset. This is called “Out-Of-Bag-Dataset” and mainly we fit this dataset to the built random forest and compare the result with the original answer. If it comes the same, that implies the model predicts well in a new dataset. The incorrectly (wrongly) predicted data is called Out-Of-Bag-Error. To get the most accurate random forest model, we can compare the result of the “Out-Of-Bag-Error” for the random forest model with different amounts of variables per step which means; once with 2-variable, again with 3-variable, in general with a value above and below the first selected variables. In this way, one can obtain the most accurate random forest model.

Support vector machine for classification

A support vector machine is one of the popular supervised classification methods. Here three different ideas of the algorithm discussed below.

Maximal margin classifier

Here we are going to focus on hyperplane and optimal separating hyperplane. Hyperplane means a flat affine subspace of n-dimensional subspace. For example, in two dimensions a hyperplane is a line, in three dimensions a hyperplane is a plane. If a dataset can be separated by a hyperplane there will be a lot of hyperplanes. So, in this case, there should be a method to choose the most optimal separating hyperplane out of all. The way we decide which

boundaries are ideal to take as a classification boundary out of all is by the processes of taking the nearby data points and measure the distance between that boundary line to the data points. The distance between the data points and the boundary lines(hyperplanes) is called margin. Then, we can decide based on the margins such that: depending on the idea of the boundary which has a higher margin is better than the lower margin. This is because in higher-margin misclassification error will be minimized and the decision boundary classifies the two groups more appropriate. That is exactly what support vector machine tries to do: to maximize the margins between the nearby data points and the boundary line itself. There is a case that is non-separable classes, in such a case we are going to use a hyperplane that approximately splits or separates the class by using soft margin. A support vector classifier means who concludes the maximal margin classifier to the non-separable case. Here we are talking about a multiclassification case which means there are more than 2 classes to classify into. In the multiclassification task "Support vector machine draws a hyperplane in N-dimensional space such a way that it maximizes the margin between classification groups". In the matter of that, the maximal margin hyperplane is very sensitive to a change in a single or one data point says that it may have overfitted the training data. Anyways we must consider a hyperplane classifier that doesn't ideally separate the two classes for the interest of high robustness to individual observations and greater classification in most of the training data points. In this way, there might be a misclassification within a few training data points to perform better in classifying the remaining data points. Above explanation tries to demonstrate how soft margin classifier (support vector classifier) does (Gareth James, 2013).

The below figure shows how the algorithm works in binary classification. The red and blue full-coloured shapes are considered as nearby data points or support vectors. And, the green line is a boundary line (a hyperplane which tries to maximize the margins).

Figure: 2.9 Maximal margin classifier in binary (2D) and 3D classification

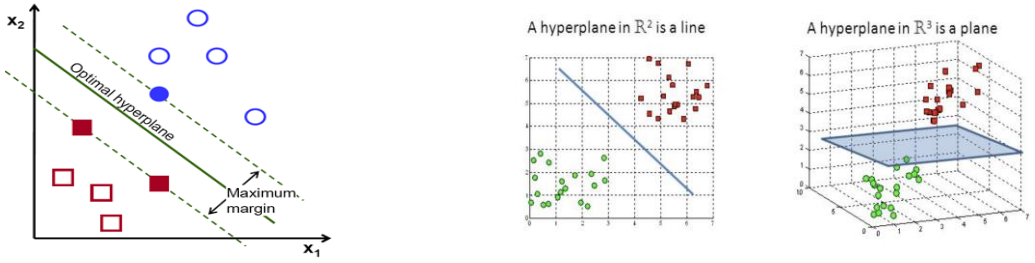
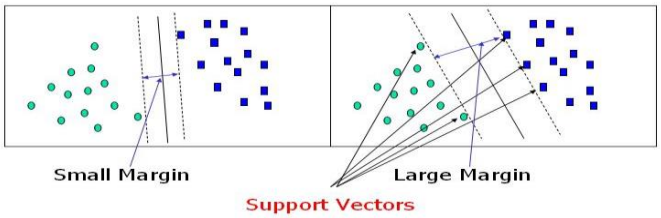


Figure: 2.10 Support vectors, small margin and large margin (Gandhi, 2018)



(Gandhi, 2018)

Above figure 3.5 tries to show nearby data points called 'support vectors', and what are small margins and large margins means.

Up to now, we were discussing a classification that has a linear relationship between predictor and outcome. Let's see how the algorithm performs when the relationship between the predictor and the outcome is non-linear. In such a case. An extension of a support vector classifier called support vector machine (SVM) uses kernels that enlarging the feature space using quadratic, cubic, and higher-order polynomial functions on the predictors (Sirohi, 2019).

Chapter 3

Problem statement: A binary classification problem on an image data cube which is collected by spectrometer called CRISM. The task is classifying the image pixels based on pixel information associated with the cube indexes. If the pixel has data labelled it as '0' or no data labelled it as '1'. In other words, the machine learning model will output '0' if it detects a mineral or output '1' if not. So, this can be determined as when the machine learning

algorithm model classifies the pixel to ‘0’ we can assume there is a mineral content in that specific area of the mars surface and whereas when the model outputs ‘1’ we can assume that there is no mineral content in that surface area of mars.

Workflow

Materials and Methods

Used material and methods to perform the spectral index (mineral) extraction and classification problem is described below.

- CRISM (Compact Reconnaissance Imaging Spectrometer for Mars) Instrument

It is an imaging spectrometer and it covers wavelengths from 362 to 3920 manometers. Because CRSIM able to detect light in these wavelength ranges the team can detect or identify a wide range of minerals on the Martian surface and as a result of infrared wavelength coverage CRISM has a high capability of mapping composition of Mars surface. This instrument has (OSU) optical sensor unit: that has features like optics, a gimbal, detectors (one for infrared images and the other is for visible images), cryocooler, and radiators, (GME) Gimbal Motor Electronics: this powers and commands the gimbal and in a feedback loop from its angular position encoder analyses data, (DPU) Data Processing Unit: commands from the spacecraft and data from the OSU goes here and processes to communicate it to the spacecraft. Generally, the aim of CRISM’s science is firstly, by using of minerals spectral fingerprints which form liquid water, searching where Mars had a persistently wet place or environment, Secondly, trying to understand what processes formed and changed the Martian crust through mapping the composition and layering of rock formations. Thirdly, it helps to better understand Mars modern climate by measuring the changing amounts of ice, water vapor, dust, and other gases in the atmosphere (Webmaster, n.d.).

- Jupiter notebook Python 3: - An open-source web application including data cleaning statistical modelling, data visualization, machine learning, and much more OS.

- Spectral python (SPy): python module for handling and processing hyperspectral image data.
- Pandas: - A library for data structures and data analysis.
- NumPy: - A package for scientific computing with python.
- Scikit-learn: - An open-source library that is an efficient tool for data analysis and data mining.
- Support Vector Machine and Random Forest machine learning algorithms for classification.
- Scikit-image: open source library which is useful for image processing in python.
- Matplotlib: an open-source library for interactive visualization in python.
- Pillow (PIL): is an imaging library in python.

Dataset

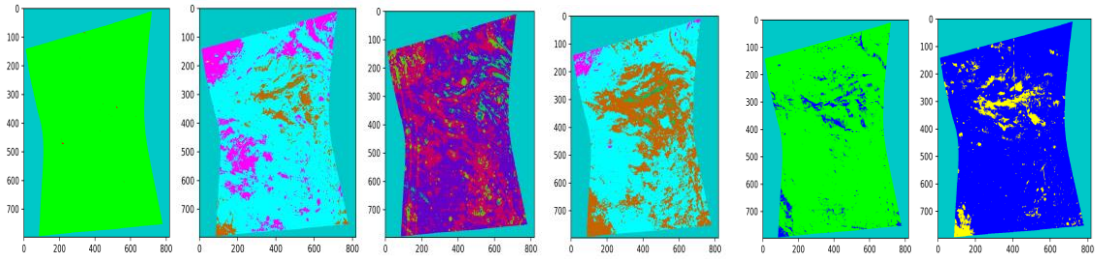
Datasets are (MTRDR) Map-Projected Targeted Reduced Data Records that involves TER data map-projected using terrain models of the Martian surface and consists of spectral summary parameters that are band maths calculations that specify the diagnostic or indicative spectral structure and collectively capture the mineralogical diversity of the surface. To use the full hyperspectral sampling of CRISM targeted observations, the summary parameters have been updated from earlier multispectral formulations [Pelkey et al. 2007]. Additionally, to avoid uncertainty and to better capture the target mineral spectral signature, the parameter wavelength selection has been updated (Webmaster, n.d.). The dataset is retrieved from the Mars Orbital Data Explorer portal (*https://ode.rsl.wustl.edu/mars/*) and to each pack there are numerous files divided into two main folders. The below table shows the dataset-specific spectral summary product.

Table 3.1: CRISM data summery product formulations.

NAME	PARAMETER	FORMULATION (nm)	STRETCH LIMITS*	PHASES DETECTED	CAVEATS
VNA	VNIR albedo proxy	7-channel median at 770 nm	**0.1% linear stretch**	VNIR apparent reflectance	--
BD530	0.53-µm band depth	1 – (R530/(a*R709+b*R440))	**0.18-0.25**	Extremely fine grained crystalline hematite	Slightly higher values from high atmospheric dust opacities
BD920	0.92 µm band depth	1 – (R920/(a*R800+b*R984))	**0.005-0.02**	Crystalline ferric minerals	Also detects low-Ca pyroxene
BDI1000VIS	1-µm integrated band depth; VNIR wavelengths	divide VNIR wavelengths by RPEAK1 then integrate over (1 – normalized radiances) to get integrated band depth	**0.01-0.022**	Minerals with Fe, especially in ferrous form	--
IRA	IR albedo proxy	7-channel median at 1330 nm	**0.1% linear stretch**	IR apparent reflectance	--
OLINDEX3 (beginning with MTRDR)	Olivine index 3	RCBD1080 × 0.025 + RCBD1152 × 0.025 + RCBD1210 × 0.025 + RCBD1250 × 0.025 + RCBD1263 × 0.05 + RCBD1276 × 0.05 + RCBD1330 × 0.10 + RCBD1368 × 0.15 + RCBD1395 × 0.15 + RCBD1427 × 0.20 + RCBD1470 × 0.20 where RCBD#### denotes the relative band depth [*i.e.*, ((RC####-R####)/(RC####))], where RC#### denotes the value of a point at a wavelength of #### nm along a modeled line that follows the average slope of the spectrum.	**0.025-0.1**	Olivine and Fe-bearing phyllosilicate	Weakly sensitive to dust
LCPINDEX2 (beginning with MTRDR)	Sensitive to ~2.0 µm feature associated with LCP	RCBD1690 × 0.10 + RCBD1750 × 0.20 + RCBD1810 × 0.35 + RCBD1870 × 0.20 + RCBD2120 × 0.10 + RCBD2140 × 0.05	**0.0-0.02**	Low-calcium pyroxene	Weakly sensitive to illumination effects on shaded slopes
HCPINDEX2 (beginning with MTRDR)	Sensitive to ~2.0 µm feature associated with HCP	RCBD2120 × 0.10 + RCBD2140 × 0.20 + RCBD2230 × 0.35 + RCBD2250 × 0.20 + RCBD2430 × 0.10 + RCBD2460 × 0.05	**0.0-0.15**	High-calcium pyroxene	--
BD1900(2)	1.9-µm H2O band depth	1 – ((R1930) / (a*R1850+b*R2046))	**0.005-0.025**	Most minerals with bound molecular H2O	Weakly sensitive to atmospheric ice hazes especially at low solar incidence angles
BD2100(2)	2.1-µm shifted H2O band depth	1 – ((R2132) / (a*R1930+b*R2250))	**0.008-0.03**	H2O in monohydrated sulphates	Also sensitive to pyroxenes, and atmospheric ice hazes especially at low solar incidence angles
D2300	2.3-µm dropoff	1 – ((R2290/ RC2290)+ (R2320/RC2320)+ (R2330/RC2330))/ ((R2120/RC2120)+ (R2170/RC2170)+ (R2210/RC2210)) where RC#### denotes the value of a point at a wavelength of #### nm along a modeled line that follows the average slope of the spectrum.	**0.01-0.03**	Minerals with OH bound to Fe, Mg	Weakly sensitive to shaded slopes and strong brightness boundaries.
SINDEX2	Convexity at 2.29 µm due to absorptions at s2.1 µm & 2.4 µm	1 – ((a*R2120 + b*R2400) / (R2290))	**0.01-0.03**	Hydrated sulphates	Sensitive to dust and ice hazes especially at low solar incidence angles

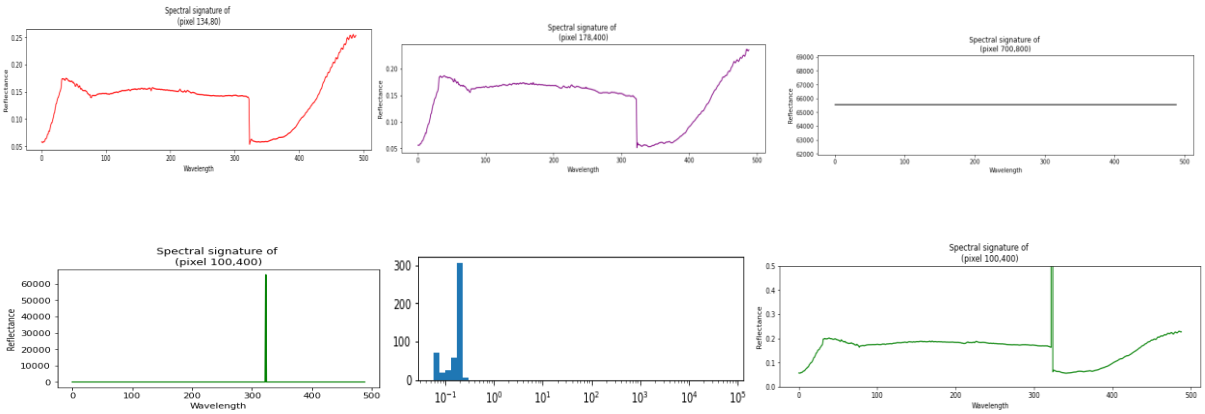
The above dataset which is described in the table is original datasets whereas the processed dataset that are extracted spectral parameters are arranged in PNG images which is an original image and thresholded image and NumPy array that is original data, thresholded data and Boolean data. The data cube used as a predictor variable which means our sample is the pixel elements which is 798 rows by 819 columns with their 489 wavelength bands or indexes. Those summery products (the 9 bands) used as a target variable during training the machine. So, we can take them as our ground truth datasets which we can use to validate our model. The whole dataset or image cube has a header file and a disc image file (.img). The header file is represented as '. hdr' and is metadata that holds the parameters of "number of rows, several columns and wavelengths" that are necessary to read the data cube in '. cube', '.dat' and '. raw' files whereas the .img file hold the image cube. From the data cube, the 489 wavelength bands are our feature or predictor variable and each pixel of the data cube which means 653562 are samples or observations that we going to train the machine with to create a model that detects a mineral in the image pixels. So, now we are assuming that this image pixel represents a specific place on the surface of Mars proportionally to it which means in the same dimension and position. Based on this image pixel information and the trained machine learning model one can identify where the minerals are accumulated in the surface of mars. The 9 bands (summery products) which are used as a target variable are the size of 798 rows by 819 columns image file and thresholded by threshing value 10. In our case, we are going to use one of the summery products as the ground truth. And those summery products provide an information, i.e. on the image pixel, the collection of details at the specific location that makes the image data to be associated with real materials and features on the ground. In this case, the ground material associated with the image data is the minerals or the area covered by the minerals. This data cube is useful to approach the image classification performance where each pixel of the image is contrasted with interrelated ground truth data to find a match that fits into it. Here the goal is, minimizing the error between the CRISM image and the ground truth. The less error between them the more the classification algorithm performs. The below figure shows some of the extracted indexes in different wavelength bands from the image data cube. All the images have a shape of 798 rows by 819 columns and at an index value of 0, 300, 488, 200, 10, and 400 respectively.

Figure: 3.1 Data cube image indexes in different wavelength



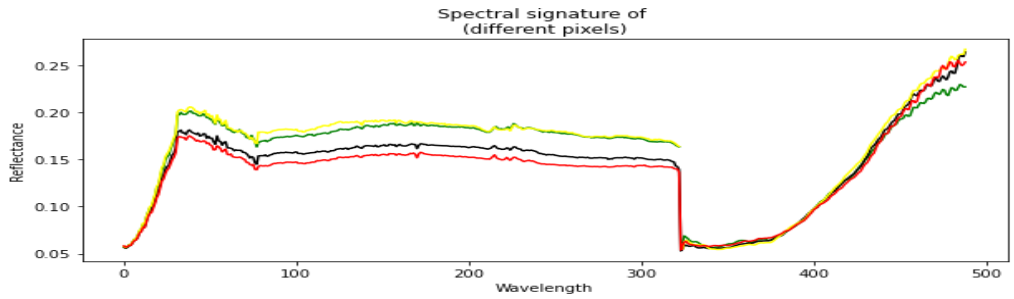
As we can see from the above figure, it is clearly seen that all of them have the same shape and gives a piece of information like; on the difference between the area of image pixel with data (information) and area of image pixel without data (information). Even if they are in the same shape the details of the image show difference and this difference might tell us how the materials in the image are detected differently in different wavelength bands. So, we can conclude that each wavelength band provides its information and, the combination of those different wavelength bands is useful to be able to detect minerals on the surface of Mars. Based on spectra information from each pixel one can even detect the exact mineral type (which mineral) are accumulated in that specific area of mars. Below figure 3.2 shows the sample extracted image pixel out of the data cube with its spectral signature. The spectral signature shows information of image data cube in different pixel and the x-axis is the wavelength and the y-axis are reflectance. Since CRISM spectroscopy captures an image data through electromagnetic spectrum up to 500 nanometres, the x-axis shows 0 up to 500 wavelength bands. The spectral signature information is extracted from a specific pixel of the image from 134 by 80, 178 by 400, 700 by 800, and 100 by 400, respectively. As it is shown in the figure, some of the pixels show spectral information and some are not. Pixel spectral signature information has its own pattern or characteristics based on the light of different wavelengths that helps to determine what kind of mineral is accumulated in that specific image pixel. Which means, according to the spectrum, it is possible to detect which mineral concentration is there and based on the location of the pixel element able to determine the place where is the specific mineral located on the surface of Mars.

Figure: 3.2 Spectral signature of different pixels



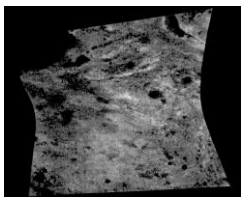
The above image describes how the spectra information can be visualized from pixels. From this; it is possible to generalize that, the rest of the pixel information which is not provided here is like one of those pixels. When we see the first two image pixel it is obvious that there is a spectra information or signature that is useful for mineral extraction and one can simply determine there is a pixel information (mineral) in that specific pixel. When we see the second image it is shown that there is no spectra information and from the straight line which is shown in the image it is possible to determine the representation of “None” or “no data” value numerical representation which is “65535.0”. When we go to the second raw of the spectral image at the first image it shows that there is pixel information but not specific enough to determine it. To be able to sure if there is data in that specific pixel or not, we can see the second histogram of that pixel and the third spectral signature. And from the histogram, it is possible to say that there is an information in this pixel and in the third plot it shows that there is spectral information in some wavelength bands and there is no information gathered in between some wavelength bands. This kind of spectral signature with not full pixel information might happen because of the result of the spectrometer, in this case CRISM or some other technical issues. This means as a reason of the equipment or human-made mistakes the spectrometer might not capture the image fully with the 489 wavelength bands like others. The image below shows the individual spectral signature all together in one plot.

Figure: 3.3. Spectral signature of different pixels in one plot.



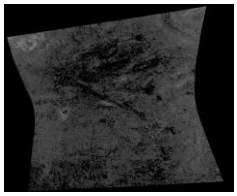
Below figure 3.3 shows a sample 4 of the indexes out of 9 bands which are processed image (thresholded image in a thresh value of 10) that is used as a target variable in the machine learning classification processes.

Figure: 3.4 Sample 4 out of 9 summery products of the CRISM image cube.



1_BD1900_2

9_SINDEX



7_OLINDEX3

2_BD2100_2

Data Pre-processing

The data cube image is captured in (798, 819, 489) which implies that 789 rows by 819 columns with 498 wavelength bands. The numerical representation of the image is in float32 data type. In general, during machine learning implementation the pixel size (in number 653562) is used as a sample of observations and the 489 wavelength bands are the feature or the columns of our data frame. Our target variable is one of the summery product images which is “7_OLINDEX3”. Out of 489 wavelength bands we are going to use 50 randomly selected indexes. Before implementing or starting the machine learning processes the “no data” values represented numerically as ‘65535.0’ should be extracted from the samples. Those “no data” values are not useful during training the machine because they do not have

information which is necessary for extracting spectral information. They just provide valuable information which is 'there is no data' value in that specific spectral signature. Then, we flatten the image pixels. To prepare the target value the image is binarized and thresholded by the threshing value of '0.37509602'. This threshold value is obtained by taking the mean of the pixel's intensity value. And we should flatten this image also. For the machine-learning algorithm to learn the patterns from the data, both the images (sample pixel and target pixel) length should be equal. Which implies that; when the no data value of '65535.0' erased from the data set, the columns or rows which have fully no data value should be dropped from both the target and sample pixels at the same position and time. So, in this way we can be sure that the length of both the sample and the target value is the same. For demonstration purposes, the cleaned dataset with the first five rows and five columns with the target variable is provided in table format.

Table: 3.2 Sample dataset.

	436.13(nm)	**494.68(nm)**	**559.78(nm)**	**624.92(nm)**	Target
8088	0.056497	0.072392	0.110154	0.150998	1
8089	0.056497	0.072392	0.110154	0.150998	1
8090	0.057104	0.071551	0.109121	0.14916	1
8091	0.057104	0.071551	0.109121	0.14916	1
8902	0.056057	0.071477	0.108903	0.147823	1

What we have now is the cleaned dataset which is 653562 rows by 51 columns and the next step is preparing the data for the machine learning algorithm. Out of 51 columns, the last column which is the target variable is our response variable so before preparing the data to fit in the classifier algorithm we must exclude it. So, we split the data without the target variable into training and testing. As it is a common rule to split the data in 80/20 training and testing set, we give 80% of the data for training the machine and the rest 20% of the data to test the created model performance. The machine learning classifier algorithm that we are going to use for this classification problem is support vector machine and random forest. Let's start with a support vector machine.

Support vector machine

Before fitting the data to the learning algorithm let us understand some terminologies related to the algorithm parameters.

Regularization (C): -In support vector machine algorithm there are two types of regularization, one is: High regularization which means when we draw a classification boundary very carefully to avoid any classification error. So, during this time, the boundary might be very zigzag or wiggly (try to overfit the model) shape especially with nD (N-features) which is a complex dataset. The second one is low regularization: In this case, we just draw a boundary line and probably have a classification error which might be okay. During this, the boundary looks very smoother. In high regularization there might be a high variance (overfitting error) because the decision boundary tries to fit most of the training dataset) but low bias. In low regularization there might be high bias (underfitting error) because of model might be too simple, but low variance.

Gamma: - Here also there are 2 different kinds of drawing boundaries. One way is when we consider the nearest data points (support vectors) then calculate the margin. This what we call high gamma. The other one is when we consider the faraway data points, and we draw the decision boundary called low gamma. The fact that the algorithm considers a lot of data points during low gamma, we can say that the algorithm performs better and expect good accuracy (more efficient). But both approaches are right. In high gamma, there might be low variance but high bias (underfitting error) because the model will be simple which doesn't capture the complexity of the data. In low gamma, there might be high variance (overfitting error) the model tries to fit most of the training data but low bias. Rarely in low gamma, the decision boundary might be accurate also (MAFIADOC, 1998).

Kernel: - Assume that we have complex data that might not be easy to draw a classification boundary. At this time, one approach might be creating a third dimension called 'Z'.

$$Z = X^2 + Y^2 \; Z = X^2 + Y^2$$

So, we are transforming our basic features (x and y) and creating new feature Z which holds the addition of X^2 and $Y^2(Z = X^2+Y^2)$ x^2 and $y^2(Z = X^2+Y^2)$, with this we will be able to draw

a new decision boundary(Z) which is called a Kernel. (Basically, the transformation (Z) is called a kernel). We need a kernel to transform the current features to be able to draw a decision boundary in a complex dataset. Fundamentally, to draw a decision boundary or a hyperplane in a complex dataset, there are four types of kernels that we can use in support vector machine. Those listed kernel types described below are using a function instead of applying a high-cost transformation.

1. Linear Kernel
2. RBF (Radical Basis Function) Kernel
3. Polynomial Kernel
4. Sigmoid kernel

The default kernel is RBF (Radical Basis Function) which is the default kernel. However, for model tuning, we are going to use Linear kernel. This kernel computes similarity in the input space. It doesn't precisely define a transformation to higher dimensions so, every hyperplane is straight lines.

Random Forest

Basic parameters that boost the model performance are n_estimators which indicates the number of trees in the forest and highly recommended to take a higher number as our computer can handle. The more we choose a higher number of it the more the model performs better, max_features which indicates the number of features to take account when searching for the split, and a minimum number of samples in newly created leaves and it is a good idea to take the higher number out of the trial. The other parameters of random forest algorithm which makes the learning processes more robust and efficient are n_jobs, random_state, and oob_score. n_jobs determines the number of jobs to run parallel or to train and test the model if different processors should be used. If we set n_jobs to -1, it means it uses all the processors and it is recommended when many trees are trained. The random state is used when building trees to control both the randomness of the samples and when searching for the principal split at each node to observe the sampling of the features. This implies that by giving the same

random state value to the models their outcome mostly or approximately the same. When we came to OOB_score which means out of bag scores, it is the most important parameters of random forest that work as a validation method for the creating random forest model. (Bonaccorso, Machine Learning Algorithms, 2020).

Main Results and Discussion

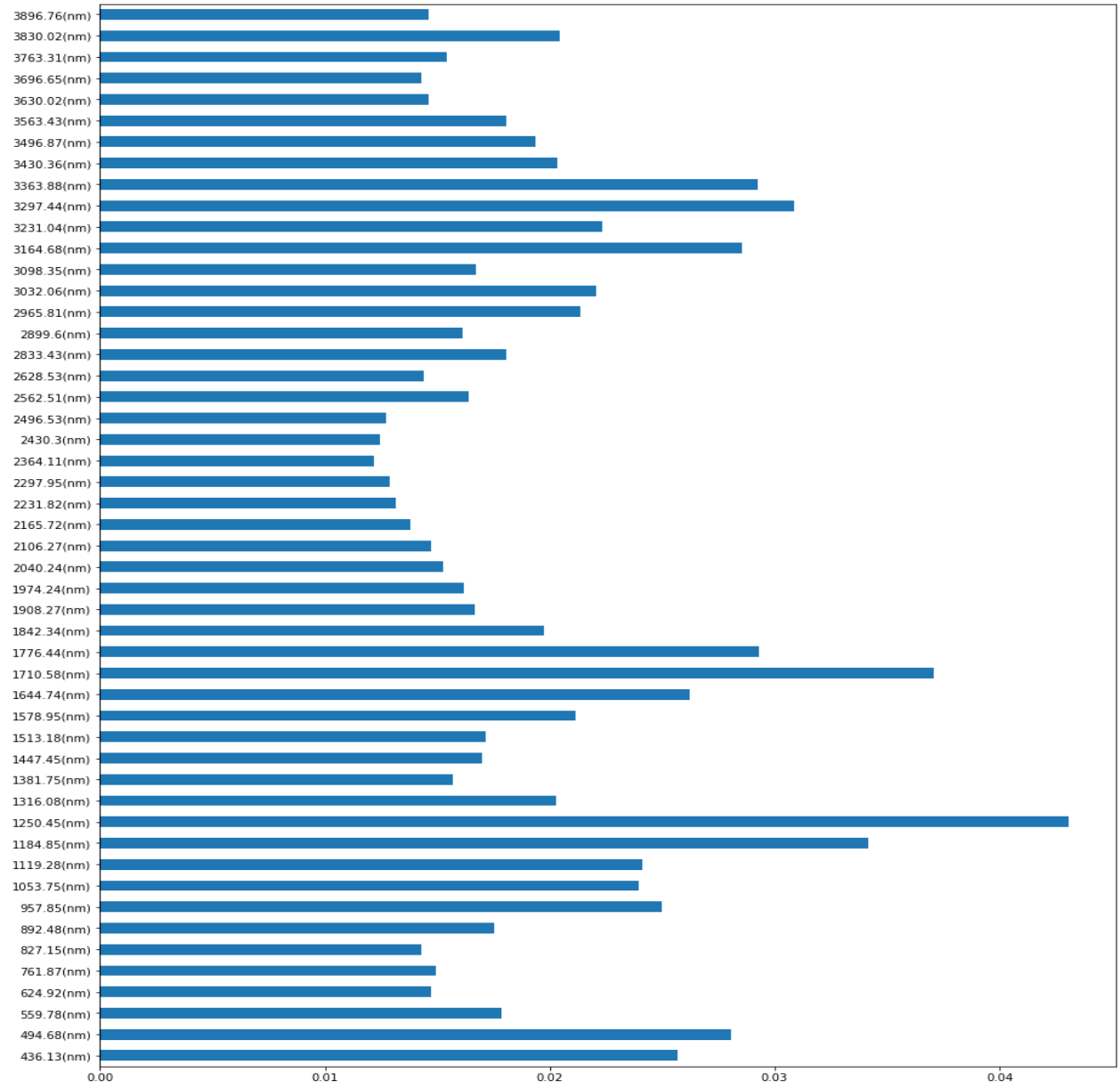
In our first try on support vector machine classifier we have used 50 randomly selected wavelength bands as our feature and 653562-pixels associated with it, the model classifies the image pixels properly with a classification accuracy score of 80%. And when we tune the model through one by changing the regularization of the model parameter to 10 the model performance is the same which is 80% and with kernel liner, the model shows similarity with a score of 80%. But when we see the model performance during a gamma value of 20 the model prediction accuracy performance increases to 87 %. This implies that the algorithm performs better when the classification boundary considers only the nearby data points or support vectors.

When we came to random forest classifiers the model did a good job which means the model classification accuracy upgrades to 94%. On the first try with 50 randomly selected indexes with 653562-pixels associated with its wavelength bands as our sample, the model classified the classes properly with an accuracy score of 0.94. During model tuning, we used 10 randomly selected wavelength bands out of 489 indexes as our feature and 653562-pixel values associated with the wavelength bands. This is because of the computer speed performance issue. So, in a random forest model tuning we trained the data within three n_estimators values which means the number of trees each model can take. Those numbers are 500, 1000, and 2000 number of trees. As we have discussed earlier taking a higher number of n_ estimators are a better choice than choosing fewer numbers during training the machine but here in this case as we can see the accuracy result didn’t show that much difference. In all cases, the accuracy result came up approximately the same result which is 94%. And three of the model’s accuracy score have a very little difference. When we put those scores from higher to lower which means on 1000 Trees: 0.9457417737527948 scores, on 2000 Trees:

0.9456966056143996 scores and on 500 Trees: 0.9456966056143996 scores. It is shown that the classifier classifies a little bit better when the number of trees in the forest is 1000 but in according to 500 and 2000, the accuracy score is almost the same. Again, this 94% accuracy is also the same with the accuracy score of the model when the feature is 50 wavelength bands. The only difference is during 50 wavelength bands the accuracy level is a little bit higher (0.9491998155583297). So, from this result, we can conclude that the more the feature the higher the accuracy will be but with little difference. And here again there are things we have to consider, that is; when we train the model with those different tree values the other parameters with their values was the same which means "oob_score = True, n_jobs = -1, random_state = 42". Those similarity of the parameters contributed as the models to have approximately the same result. Especially because of 'random state' value is 42.

The below bar plot shows that when the features of the model are 50 randomly chosen wavelength bands, which features were highly contributed the model accuracy level to be 94%. That means which variable is very important. As it is shown in the plot the variables which is important are accumulated at the beginning, in the middle and at last position. When we put a thresh value of o.o4 and we take features more than (> 0.04) as an important variable we will get wavelengths of around 35 as important features. Those indexes are '436.13(nm)', '827.15(nm)', '1184.85(nm)', '1974.24(nm)', '2231.82(nm)' '2496.53(nm)', '2899.6(nm)', '3164.68(nm)', '3430.36(nm)', '3696.65(nm)', '494.68(nm)', '2040.24(nm)', '2106.27(nm)', '1776.44(nm)', '1842.34(nm)', '1513.18(nm)', '1578.95(nm)', '1250.45(nm)', '1316.08(nm)', '892.48(nm)', '957.85(nm)', '494.68(nm)', '559.78(nm)', '2628.53(nm)', '3032.06(nm)', '3297.44(nm)', '3563.43(nm)', '3830.02(nm)', '624.92(nm)', and 1053.75(nm)'.This extraction or selection of features based on their contribution on the accuracy of the model is called feature engineering. Those extracted features are very important for model tuning. This means, after identifying the most important features, then by training the machine with only those features, it is possible to increase the model performance (increasing the classification accuracy).

Figure: 3.5 Features of the 50 wavelength bands.



Comparison between the two learning algorithms: Based on the out came of our models on those two algorithms we can conclude that random forest did a better job than support vector machine for spectral index extraction and classification problem. In general, random forest classifier model classifies the classes with an accuracy level of approximately 94%. Whereas support vector machine classifies the classes with the accuracy level of 87% when gamma is 20 and around 80% when tuning the parameters and with default parameters. One thing we

can pick or consider as a strength of random forest classifier is OOB (out of bag) score which is the random forest classifier way of validating the predictions. This feature makes the classifier most efficient and robust. To explain simply about OOB score: as we know, final prediction decision of random forest is depending on those bootstrap sampling trees and during the processes that means sampling the data out of whole dataset to train the machine, there are left out samples and those are called out of bag sample. After all the sampled diction trees are trained, those tree models are used to predict the left-out sample datasets. Each left out datasets are predicted by using of all decision tree models one by one. And the final decision of the prediction would be the majority vote or the most predicted value. So, which means OOB score is calculated as the number of accurately predicted rows from the out of bag sample. When we came to support vector machine, it has also a good feature we can pick and discuss about for instance the kernel, support vector machine supports different kernels to classify the classes more accurately. The selection of the kernel should be based on the data behaviour or our kind of dataset. Above all, by summarising our findings we can say that random forest classifier is more robust and efficient than support vector machine classifier for spectral index extraction and classification.

Conclusion and recommendations

In conclusion so far, we have discussed about the pre-processes and exploring of data before using the machine learning processes overall and specifically for image data cube. In addition to this we have explained the most important aspects related to machine learning like; types of machine learning and their general idea, bias and variance, hyperparameter tuning, model validation and in general pre-processes and post-processes of the data before and after training the machine. Then we have also discussed about image types and their processes with their application area focusing on spectral image. Overall, we have examined how random forest and support vector machine algorithms performs in general and we have tried how they operate for spectral index extraction and classification problems. In summary, the use of machine learning techniques for spectral index extraction and classification based on support vector machine and random forest classifier we have seen that both algorithms are classified with an accuracy level above 80% which is good. But random forest shows more accurate and

efficient result than support vector machine. So, random forest is a better machine learning classifier algorithm for spectral index extraction and classification problem. After this research depending on what we have seen, and cover related to the topic I would like to recommend trying other machine learning classifier algorithm like neural network and check how the result will be. In addition to this, it is a good idea to go back to data pre-processes and processes the data then use more wavelength bands as a feature and try to train the machine again. The link of the coding part of this thesis is provided at the end of the references.

References

ALABAMA, U. O. (2020). *Spectral Imaging*. Retrieved 2020, from https://www.southalabama.edu/centers/bioimaging/spectral_imaging.html

America, M. S. (1997). *Collector's Corner*. Retrieved 2020, from http://www.minsocam.org/msa/collectors_corner/faq/faqmingen.htm

America, M. S. (2020, 05). *Collector's Corner*. Retrieved from FREQUENTLY ASKED QUESTIONS MINERALOGY - GENERAL: http://www.minsocam.org/msa/collectors_corner/faq/faqmingen.htm

B.B. Singh, B. A. (2005). *On Pulsed emission of TeV γ -rays from Crab Pulsar.* Mumbai: Indian Institute ofAstrophysics.

Bajaj, P. (2020). *GeeksforGeeks.* Retrieved from https://www.geeksforgeeks.org/what-is-reinforcement-learning/

Ben-David, S. S.-S. (2014). *UNDERSTANDING MACHINE LEARNING From Theory to Algorithms* (1st ed.). New York: CAMBRIDGE UNIVERITY PRESS.

Bonaccorso, G. (2018). Machine Learning Algorithms . In G. Bonaccorso (Ed.), *Popular algorithms for data science and machine learning* (p. 501). Birmingham: Packt Publishing Ltd.

Bonaccorso, G. (2020). Machine Learning Algorithms. In G. Bonaccorso, *A reference guide to popular algorithms for data science and machinelearning* (p. 453). BIRMINGHAM : Packt Publishing Ltd.

Chauhan, N. S. (2020, January). *KDnuggets*. Retrieved 04 09, 2020, from https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html

Department, C. E. (2009). *Hong Kong Geology*. Retrieved January 08, 2019, from https://hkss.cedd.gov.hk/hkss/eng/education/GS/eng/hkg/chapter1.htm

EliteDataScience.com. (2019). *Overfitting in Machine Learnning: What it is and how to prevent it.* Retrieved March 27, 2020, from https://elitedatascience.com/overfitting-in-machine-learning

Fu, X. (2016). Eighth International Conference on Advanced Computational Intelligence (ICACI). *Multispectral image classification based on neural network ensembles*, 275-277.

G. Goos, J. H. (2004). *Advances in Information Retrieval* . Retrieved from INTERNET : https://archive.org/stream/springer_10.1007-b96895/10.1007-b96895_djvu.txt

Gandhi, R. (2018). *towards data science*. Retrieved 04 12, 2020, from https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

Gareth James, D. W. (2013). *An Introduction to Statstical Learnning* (1st ed.). London: Springer Science+Business Media New York 2013.

Geer, Z. (2019, November 08). *A Comprehensive Guide to Random Forest in R.* Retrieved from DZone: https://dzone.com/articles/a-comprehensive-guide-to-random-forest-in-r

Hutter F., K. L. (2019). SpringerLink. *Hyperparameter Optimization*, https://link.springer.com/chapter/10.1007/978-3-030-05318-5_1#citeas.

Hutter, M. F. (2018/2019). Machine Learnning Lab. In M. F. Hutter (Ed.), *Hyperparameter Optimization* (p. 35). Freiburg-Hannover: AutoML.org.

Hutter, M. F. (n.d.). Hyperparameter Optimization. In M. F. Hutter, *AutoML_Book_Chapter 1*. Creative Commons.

Jiawei Han, ,. M. (2011). Data minning. In T. Edition (Ed.), *Concepts and Techniques* (p. 740). Massachusetts: Elsevier Science & Technology.

Julian Koch1, H. B. (2019, Nov 15). *HESS*. Retrieved from Modelling of the shallow water table at high spatial resolution using random forests: https://www.hydrol-earth-syst-sci.net/23/4603/2019/

Learnning, I. C. (2010). *International Conference on Machine Learnning*. Retrieved June 21-24, 2010, from http://www.icml2010.org/related-fields-study.html

Machine Learning in Image Analysis - Theory and Practice. (2016). Retrieved April 04, 2020, from http://www.zib.de/MLIA

MAFIADOC. (1998). Retrieved from Theoretical Bioinformatics and Machine Learning (pdf, 8 MB): https://mafiadoc.com/theoretical-bioinformatics-and-machine-learning-pdf-8-mb_59a010741723dd0f406eeeb1.html

Mutuvi, S. (2019, April 16). *Introduction to Machine Learning Model Evaluation.* Retrieved 03 26, 2020, from https://heartbeat.fritz.ai/introduction-to-machine-learning-model-evaluation-fa859e1b2d7f

Mutuvi, S. (2019, April 16). *Introduction to Machine Learning Model Evaluation.* Retrieved from HEARTBEAT: https://heartbeat.fritz.ai/introduction-to-machine-learning-model-evaluation-fa859e1b2d7f

Neves, D. M. (2015). Natural Language Processing SoSe 2015. In I. S. Potsdam (Ed.), *Machine Learning for NLP* (p. 84). Potsdam: Hasso Plattner Instituite.

Newman, P. (2013). *National Aeronautics and space administration*. Retrieved 04 04, 2020, from https://www.google.com/search?q=electromagnetic+spectrum&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjL-pHLvM7oAhUpxqYKHXcPBt8Q_AUoAXoECBMQAw&biw=1536&bih=722#imgrc=zwm48m-mGrYD6M

R, V. (2018, September 11). *Feature selection — Correlation and P-value.* Retrieved from Towards Data Science: https://towardsdatascience.com/feature-selection-correlation-and-p-value-da8921bfb3cf

Raschka, S. (2018). Machine Learning. In S. Raschka (Ed.), *Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning* (p. 49). Madison, Wisconsin: University of Wisconsin–Madison.

Rebeca González, H. G.-L. (2020). Image Processing in Python. DataCamp.

Rodriguez-Galiano, V. S.-C.-O.-R. (2019). Machine learning predictive models for mineral prospectivity:. *An evaluation of neural networks, random forest, regression trees and support vector machines, Ore Geol*, https://www.hydrol-earth-syst-sci.net/23/4603/2019/.

Seltman, H. (1999). CMU statstics. In H. Seltman (Ed.), *Exploratory Data Analysis* (p. 40). Carnegie Mellon University.

Silva, L. D. (2019, March 02). *Overfitting and Underfitting.* Retrieved from StackExchange: https://stats.stackexchange.com/questions/395197/overfitting-and-underfitting/395205

Sirohi, K. (2019, August 06). *Support Vector Machine (Detailed Explanation).* Retrieved from Towards Data Science: https://towardsdatascience.com/support-vector-machine-support-vector-classifier-maximal-margin-classifier-22648a38ad9c

specim. (2020). *Spectral Imaging*. Retrieved 04 03, 2020, from https://www.specim.fi/what-is-hyperspectral-imaging/

Stephen Coggeshall, B. X. (2019, August 03). Scientific Research. *Predicting Credit Card Transaction Fraud Using Machine Learning Algorithms*.

Trevor Hastie, R. T. (1993). *Data Mining, Inference, and Prediction* . Retrieved from Springer Series in Statistics : https://archive.org/stream/Machine_Learning_201705/Elements_of_Statistical_Learning_print10_djvu.txt

Tutorialpoint. (2020). *Digital Image Processing.* Retrieved 04 02, 2020, from https://www.tutorialspoint.com/dip/index.htm

V.Rodriguez-Galiano, M.-C. M.-O.-R. (2015). Ore Geology Reviews. *Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines, 71*, 804-818.

V.Rodriguez-Galiano, M.-C.-O.-R. (2015). Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. *ScienceDirect*, https://www.sciencedirect.com/science/article/pii/S0169136815000037.

Vidal, M. a. (2012). Pre-processing of hyper spectral images. Essential steps before image analysis. *Chemometrics and Intelligent Laboratory Systems*, 138-148.

Vidal, M. a. (2012). Preprocessing of hyperspectral images. Essential steps before image analysis . *Chemometrics and intellegent laboratory system, 117*, 138-148.

Webmaster, J. (n.d.). *Compact Reconnaissance Imaging Spectrometer for Mars*. Retrieved from http://crism.jhuapl.edu/instrument/design/overview.php

West, M. a. (2010). Potential martian mineral resources: Mechanisms and terrestrial analogues. *Planetry and Space Science, 58*(4), 574-582.

Wikipedia. (n.d.). *Mineralogy of Mars*. Retrieved October 27, 2019, from https://en.wikipedia.org/wiki/Mineralogy_of_Mars

wikipedia.org. (2012). *Syrtis major*. Retrieved December 21, 2017, from https://de.wikipedia.org/wiki/Syrtis_Major

Wild, F. (2015). *NASA TV*. Retrieved August 07, 2017, from https://www.nasa.gov/audience/forstudents/5-8/features/nasa-knows/what-is-mars-58.html

Woods, R. C. (2002). Digital Image Processing. In P. I. Edition (Ed.), *Digital Image Processing* (p. 976). of Tenness: Pearson Education.

zone, I. s. (2017, October 13). *Hands-On AI Part 14: Image Data Preprocessing and Augmentation.* Retrieved April 05, 2020, from https://software.intel.com/en-us/articles/hands-on-ai-part-14-image-data-preprocessing-and-augmentation

https://github.com/fmammo/Mas_thesis/tree/fmammo-patch-1

Statutory Declaration

Family Name, Given/First Name	Mammo, Feven Legesse
Matriculation number	30002359
What kind of thesis are you submitting? Bachelor-, Master- or PhD-Thesis	Masters

English: Declaration of Authorship

I hereby declare that the thesis submitted was created and written solely by myself without any external support. Any sources, direct or indirect, are marked as such. I am aware of the fact that the contents of the thesis in digital form may be revised with regard to usage of unauthorized aid as well as whether the whole or parts of it may be identified as plagiarism. I do agree my work to be entered into a database for it to be compared with existing sources, where it will remain in order to enable further comparisons with future theses. This does not grant any rights of reproduction and usage, however.

This document was neither presented to any other examination board nor has it been published.

German: Erklärung der Autorenschaft (Urheberschaft)

Ich erkläre hiermit, dass die vorliegende Arbeit ohne fremde Hilfe ausschließlich von mir erstellt und geschrieben worden ist. Jedwede verwendeten Quellen, direkter oder indirekter Art, sind als solche kenntlich gemacht worden. Mir ist die Tatsache bewusst, dass der Inhalt der Thesis in digitaler Form geprüft werden kann im Hinblick darauf, ob es sich ganz oder in Teilen um ein Plagiat handelt. Ich bin damit einverstanden, dass meine Arbeit in einer Datenbank eingegeben werden kann, um mit bereits bestehenden Quellen verglichen zu werden und dort auch verbleibt, um mit zukünftigen Arbeiten verglichen werden zu können. Dies berechtigt jedoch nicht zur Verwendung oder Vervielfältigung.

Diese Arbeit wurde noch keiner anderen Prüfungsbehörde vorgelegt noch wurde sie bisher veröffentlicht.

May/19/2020 

Date , Signature