**UFCFMJ-15-M**

**Machine Learning and Predictive Analytics**

**Student ID: 18012587**

**Module Leader: Dr Hisham Ihshaish**

**Word Count: 2105**

**GitHub Repository:** [**https://gitlab.uwe.ac.uk/a2-coutodeoliv/machine\_l\_assignment/-/tree/main**](https://gitlab.uwe.ac.uk/a2-coutodeoliv/machine_l_assignment/-/tree/main)

**Introduction**

Mushroom hunting, otherwise known as “shrooming”, is a popular activity in Europe, Asia, and certain parts of the Middle East. Consequently, identification is essential to prevent fatalities, which in retrospect can be a challenging task as there are no simple rules which can determine the edibility of a mushroom. For example, the Death cap, the deadliest of all mushrooms, is distributed throughout Europe and look similar to edible mushrooms such as the Straw and Caesar’s mushrooms. Its heat-stable amatoxins can endure high temperatures when cooked and if eaten, it harms cells all over the body rapidly (Patowary, 2010).

The features of a mushroom can help determine its toxicity. The odour, texture and colour when combined can give the person hunting, clues on the nature of the plant. In this sense, the use of machine learning is desirable to accurately make predictions as poor judgement from people could lead to death. In fact, a study conducted in America found that between 1999 and 2016, there were 133 700 cases of mushroom exposure. 83% of the cases being unintentional and 86% of them causing little or no harm to the person (Bradenburg, 2018).

Consequently, this report will investigate the most used supervised machine learning model, to find how it performs on the dataset used and which are the most indicative features of a poisonous mushroom. Specifically, the KNN model is going to be assessed and compared. Lastly, the report will give recommendations as to which learning algorithm is less suitable for my predictive model.

**Data Analyses and Manipulation**

Chart, pie chart

Description automatically generatedThis Kaggle (2017) dataset was originally contributed to the UCI Machine learning repository in 1987. There are 23 attributes and 8124 observations. The dataset is based on hypothetical samples corresponding to 23 species of gilled mushrooms which are part of the Agaricus and Lepiota family and they can be classified as either edible or poisonous. The dataset did not have any missing values, and the distribution of the two types was almost even, with 51.8% of the data belonging to the Edible category while the 48.2% belonged to the other.

Table

Description automatically generated

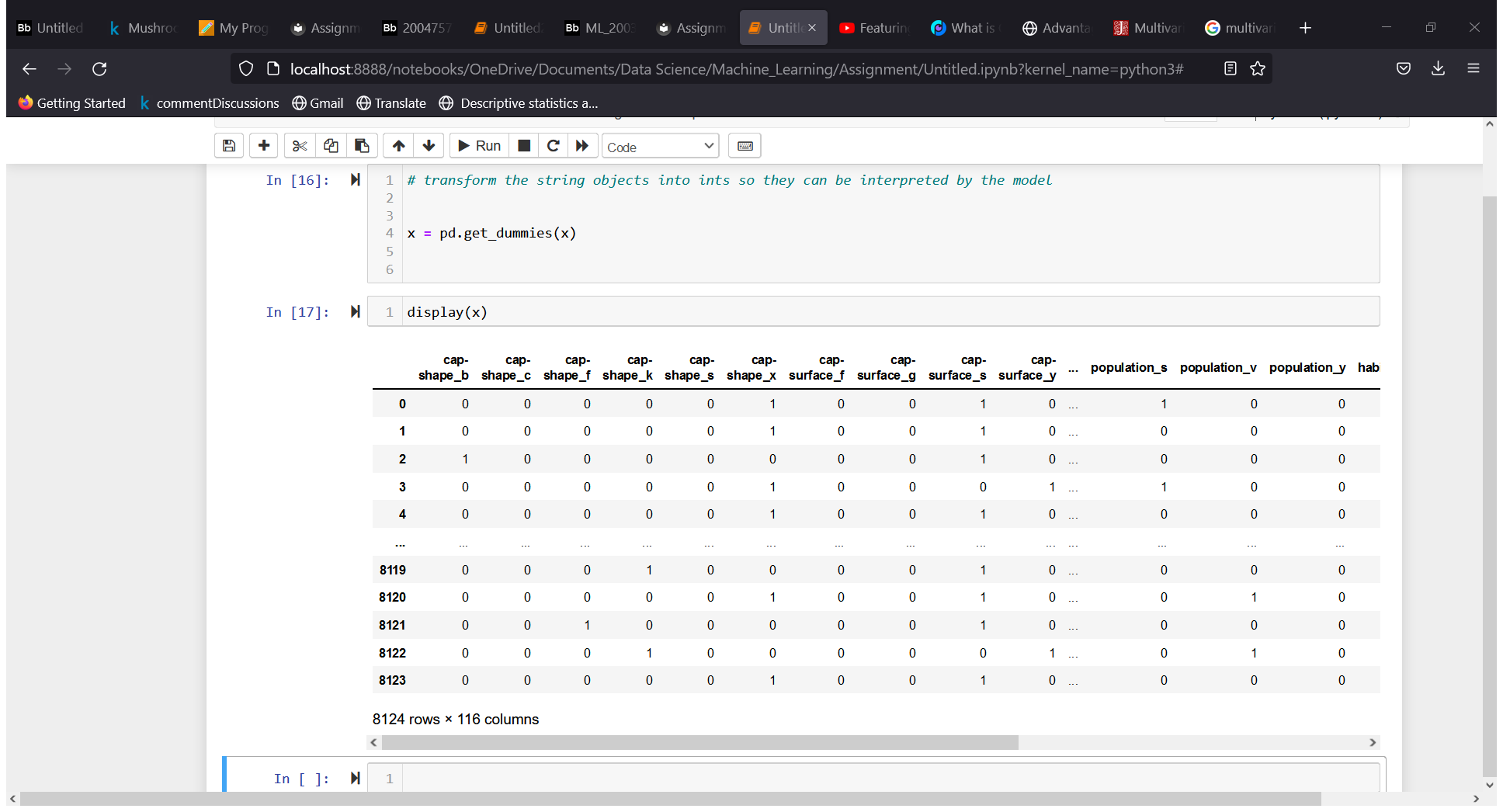
When analysing table 2 we can see how the features all hold object data types, which can be considered nominal variables. The feature which has the highest number of unique instances is Gill, holding 12 different types of mushroom colours. On the other hand, Veil type only held one unique instance. Consequently, it was removed from the dataset as it did not help determine the nature of the mushrooms.

Chart, bar chart

Description automatically generated

The next process of the project was to transform the categorical data. This is because KNN can only work with numeric values, unlike decision trees. One-Hot Encoding was used for this process, specifically to assess how it would affect the performance of the KNN model.

Dummy variable encoding works by creating a new categorical column and assigning a binary value of 1 or 0 to those columns. The number of additional features is dependent on the number of unique values in the categorical feature. This type of encoding is advantageous to our data set that has nominal values, meaning no relationship to each other. This is because, machine learning models tend to give meaning to the order of numbers, placing importance on the high values. However, this can also be an issue in certain ordinal situations where data that has no ranking, leading to issues with predictions and poor performance (Brownlee 2017).



After the One-hot encoding the dataset had 116 features. I predicted this to influence the KNN in a negative way. This is because KNN can be sensitive to the introduction of new features. This is because it performs better with a lower number of features. Furthermore, the distribution has been altered, before there were 23 features with 8124 rows. However, now there 116 features but the number of rows is still the same. This, however, does not mean the model is likely to under perform since it does not translate to high dimensionality(Soni 2020).

Nevertheless, other encoding methods which were considered but not used because of their unsuitability were, Ordinal, and Label Encoding. In ordinal encoding, every unique category in the features is assigned a numeric value. The issue with this is that it creates an ordinal relationship between categorical variables where none previously existed. Label encoding also suffers from the same issue as it assigns numeric values based on alphabetical order (Choi 2021).

**Analyses Type**

Multivariate analysis was the best fit for my type of research, this is because we are comparing the relationship of multiple features (independent variables) to determine whether each specific data entry corresponds to either a poisonous or edible mushroom (dependent variables). Specifically, we are conducting a classification analysis. In this type of multivariate analyses, we are constructing a discriminating function that separates objects based on certain measures. These objects are then classified based on the value of those measures. The advantage of using multivariate analysis is that it offers a deeper examination by looking at all the possible factors. Furthermore, the higher the quality of the data the more accurate the prediction will be. Multivariate analyses can be more demanding than bivariate analyses. This is because it can require more computational power depending on the number of variables you want to analyse. Also, for the prediction to be accurate there must be complete data (Kent 2021).

A bivariate would not be suited for this type of analyses because it would only evaluate the relationship between two variables to determine the class of the mushrooms. This would not be possible due to how it is dependent on multiple features. (Shah 2021)

**Learning Algorithms**

KNN is considered a supervised learning algorithm. Supervised learning methods take a known set of input data and the known response to the data, to form a model. The model makes future predictions based on new input data and they can be either regression or classification. The reason why you would want to use them is because they are simple to implement, highly effective when it comes to predicting new variables and it allows you to collect data or produce data out-put based on previous experience. However, there are some weaknesses such as, it is easy for them to overfit, pre-processing of data can be difficult and computational time can be vast (Singh, 2016).

**KNN**

KNN operates by assuming that similar objects exist in proximity; it is classified as a “lazy learner”, in the sense that it does not learn a discriminative function from the training data but instead it memorises the training dataset. Furthermore, there is no training time with KNN but as a drawback, this means that all the computation is performed when a classification is made. KNN is also known as an instance based or memory-based learning method since it heavily relies on memory to store its training data.

The advantages of using KNN is that it is simple to implement and intuitive to comprehend. The model constantly evolves with new data, as new information arises, the prediction is adjusted without having to retrain a new model (Guo 2003). Furthermore, there are a plethora of distance metric to implement which can affect the performance of the model. For my specific model, Minkowski, was the metric used. This is the generalised form of Euclidean and Manhattan distance. Euclidean is the straight-line distance between two points. Manhattan is the sum of absolute difference between the measures in all dimensions of two points (Fiori 2020).

**KNN Performance**

The data set was split into 70% training and 30% test. Nevertheless, an important step when it comes to KNN is selecting a K value for the test, for my model the number 3 was chosen at first. This is due to majority voting; an even number could lead to a tie when it comes to the classification. The results show that the model had an accuracy score of 100%. The performance of the model can be seen in the following confusion matrix. The matrix is a grid composed by four parameters, these being: True Positive, True Negative, False Positive and False Negative. The grid below shows how the model did not make a single mistake in the classification problem (Tyagi, 2020).

Application

Description automatically generated with medium confidence

Consequently, the precision, recall and F1 score also had perfect scores. Precision quantifies the number of positive class predictions that belong to the positive class. On the other hand, recall quantifies the amount of positive class predictions made from all the positive example in the dataset. While the F1 score gives a single score that balances both values together (Brownlee 2020).

Table

Description automatically generated

However, there seems to be some overfitting with the model. This could be caused by the small value of K, if the K value is set too low, the model becomes too specific and fails to generalise well. Furthermore, it becomes sensitive to noise. However, if we select a K value that is too large then the model becomes generalised and fails to predict that data points. This also known as underfitting. Consequently, as the next part of my Chart, line chart

Description automatically generatedproject, I tried to use an array of K values to test the different accuracy levels (Guo 2003).

The table above showcases how different values of K perform, the values of 1, 3 and 5 tend to outperform all the other with 100% accuracy. After that there is a drop in performance, with 29 to 39 being the worst classifiers with 99.70% in precision rate. Nevertheless, even with high K values, the rate does not fall under the 99%. Based on this graph, 5 seems to be the best number, as it can avoid some of the overfitting but still provide a high accuracy.

Nevertheless, another way we can try to reduce the overfitting of the model is through cross validation. This a model assessing technique which is used to test the reliability of the algorithm and to test how the model will generalise to unseen data (Brownlee 2020).

The cross validation was performed by a simple 5-fold method. This means that the data was split in 5 parts, where each fold was used as a testing set at some point. For the first part I cross validate the performance of the 3 K model.

Text

Description automatically generated with medium confidence

Chart, line chart

Description automatically generatedThe model performance dropped by 8% when looking at the mean score. The last part of the project was to cross validate (5-fold) the K values ranging from 1 to 39 and see how they compared. To do this I plotted the mean score for each K number.

The graph clearly shows a drop in performance ranging from 5% to 9% across the board. This however should be considered a more reliable prediction because the model has been tested on unseen data.

**Feature Importance**

When looking at the feature importance for the two types of mushrooms we can see how they both have opposite characteristics. For example, mushroom which were edible were classified with no smell, odor\_n, standing for none in the data set, while poisonous had odor\_f, standing for full. Nevertheless, edible mushrooms had no smells, had a ring type of pendant shape, a broad gill size, no bruises and smooth stalk. On the other hand, Poisonous mushrooms were foul in their smell, had silky stalk surface and their gill was narrow and buff.

A picture containing text, receipt, screenshot

Description automatically generatedA picture containing text, silhouette, screenshot

Description automatically generated

**Models less suitable**

Linear regression models are not suitable when it comes to classification. This is because they deal with continuous values, while classification problems mandate discrete values. Another reason is because with binary classification, the variance is a function of the mean and in particular is not constant as the mean changes. This contradicts one of the basic assumptions of linear regression: that the variance of residual errors is constant.

Another reason why linear regression is not suitable for the job is because for a binary classification, the mean is equal to 1 or success. If we use a linear regression to model a binary outcome it is possible that we will have a fitted regression which gives predicted values that are outside of the 0 or 1 range (Jing 2019).

**Conclusion**

As we have seen KNN can be a reliable machine learning model that can be implemented in real life, with the cross validation, the model was still able to perform a 92% rate of accuracy with a K number of 3 or 91% with a K value of 5. Although KNN can have it disadvantages when faced with big data sets or high dimensionality. It was very reliable with the mushroom data set due to its smaller size. Furthermore, the project was able to identify the top 5 features which would determine the nature of a mushroom.

**References**

Bradenburg, E., W. (2018) Mushroom poisoning epidemiology in the United States. *Mycologia* [online]. 110. Pp 637 – 641. Available from: <https://www.tandfonline.com/doi/full/10.1080/00275514.2018.1479561> [Accessed 12 May 2022]

Brownlee, J.(2017) Why One-Hot Encode Data in Machine Learning?. *Machine Learning Mastery* [online]. 28 July. Available from: <https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/> [Accessed 15 May 2022]

Brownlee, J.(2020) A Gentle Introduction to k-fold Cross-Validation. *Machine Learning Mastery* [online]. 3 August. Available from: <https://machinelearningmastery.com/k-fold-cross-validation/> [Accessed 19 May 2022]

Brownlee, J.(2020) How to Calculate Precision, Recall, and F-Measure for Imbalanced Classification. *Machine Learning Mastery* [online]. 2 August. Available from: <https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/> [Accessed 19 May 2022]

Choi, L.(2021) How and When to Use Ordinal Encoder. *Medium* [online]. 15 April. Available from: <https://leochoi146.medium.com/how-and-when-to-use-ordinal-encoder-d8b0ef90c28c> [Accessed 15 May 2022]

Fiori, L.(2020) Distance metrics and K-Nearest Neighbor (KNN). *Medium* [online]. 22 May. Available From: <https://medium.com/@luigi.fiori.lf0303/distance-metrics-and-k-nearest-neighbor-knn-1b840969c0f4> [Accessed 18 May 2022]

Guo, G.(2003) KNN Model-Based Approach in Classification. *OTM Confederated International Conferences* [online]. Pp 986-996. Available from: <https://link.springer.com/chapter/10.1007/978-3-540-39964-3_62> [Accessed 16 May 2022]

Jing, H.(2020) Why Linear Regression is not suitable for Classification. Medium [online] 7 May. Available from: <https://towardsdatascience.com/why-linear-regression-is-not-suitable-for-binary-classification-c64457be8e28> [Accessed 20 May 2022]

Kaggle, (2017) Mushroom Classification*. Kaggle*[online]. Available from: <https://www.kaggle.com/datasets/uciml/mushroom-classification> [Accessed 12 May 2022]

Kent, A., R. (2021) Multivariate Analysis. *Sage* [online]. Pp 151-182. Available from: <https://methods.sagepub.com/base/download/BookChapter/analysing-quantitative-data/i1251.xml> [Accessed 16 May 2022]

Patowary, B.S. (2010) Mushroom Poisoning-an overview. *Journal of college of Medical Sciences-Nepal* [online]. 6(2). Pp 56 – 61. Available from: <https://www.nepjol.info/index.php/JCMSN/article/download/3619/3118> [Accessed 12 May 2022]

Shan, K. (2021) Exploratory Analysis Using Univariate, Bivariate, and Multivariate Analysis Techniques. *Analytics Vidha*[online]. 19 April. Available from: <https://www.analyticsvidhya.com/blog/2021/04/exploratory-analysis-using-univariate-bivariate-and-multivariate-analysis-techniques/> [Accessed 16 May 2022]

Singh, A. (2016) A review of supervised machine learning algorithms. *IEEE* [online].Available from: <https://ieeexplore.ieee.org/abstract/document/7724478> [Accessed 16 May 2022]

Soni, A.(2020*) Advantages And Disadvantages of KNN. Medium* [online]. 03 July. Available from: <https://medium.com/@anuuz.soni/advantages-and-disadvantages-of-knn-ee06599b9336> [Accessed 15 May 2022]

Tyagi, N., 2020. What is Confusion Matrix*?. Analytics steps* [online]. 22 December. Available at: <https://www.analyticssteps.com/blogs/what-confusion-matrix> [Accessed 19 May 2022]