**Machine Learning Methods for Mushroom Classification**

**Akshatha R V**

|  |  |
| --- | --- |
| ***Abstract:* In this paper it gives about different methods of implementing classification of mushrooms into ‘edible’ or ‘poisonous’. Algorithms like multinomial Naïve bayes, Gaussian Naïve bayes, SGD, KNN, Stacking, Logistic regression, Decision tree and Random forest are used for train and testing of the data. Random forest is giving the accuracy of up to 96%.**  **I Introduction**  Mushroom is a type of fungi which is grown in large wet areas. It is an umbrella shaped fruiting body of order ‘Agaricales’  in the phylum ‘Basidiomycota’ .  Commercially important, edible mushrooms include *portobellos* (*Agaricus bisporus*), whose forms include button mushrooms, cremini, and baby bellas, and shiitake (*Lentinula edodes*). The morels (*Morchella*, *Verpa*) and false morels or lorchels (*Gyromitra*, *Helvella*) are popularly included with the true mushrooms because of their shape and fleshy structure; they resemble a deeply folded or pitted conelike sponge at the top of a hollow stem. Some are among the most highly prized edible fungi (e.g., *Morchella esculenta*). Edible truffles (various *Tuber* species), which hardly resemble mushrooms, are also popularly labelled as such. These and other edible mushrooms and fungi are free of cholesterol and contain small amounts of essential amino acids and B vitamins. However, their chief worth is as a specialty food of delicate, subtle flavour and agreeable texture. By fresh weight, the common commercially grown mushroom is more than 90 percent water, less than 3 percent protein, less than 5 percent carbohydrate, less than 1 percent fat, and about 1 percent mineral salts and vitamins. | Poisoning by wild mushrooms is common and may be fatal or produce merely mild gastrointestinal disturbance or slight allergic reaction. It is important that every mushroom intended for eating be accurately identified.  This paper focuses on creating a model to detect poisonous or edible mushroom which will distinguished based on it’s features like shape, surface, colour, odor etc. this classifies whether the mushroom is eatable or not.  **II Methodology**  In this research we used following steps:  Data Collection  Data Preparation  Visualization  Model Development  Model Evaluation   1. Data Collection   In this research we took dataset from Kaggle ([Mushroom Classification](https://www.kaggle.com/datasets/uciml/mushroom-classification)). It consists of 23 features like class, cap-shape, |

|  |  |
| --- | --- |
| cap-surface, cap-colour, bruise, odor gill-attachment, gill-spacing, gill-size, gill-color, stalk-shape, stalk-root, stalk-surface-  above-ring, stalk-surface-below-ring,  stalk-color-above-ring, stalk-color-below-ring, veil-type, veil-color, ring-number, ring-type, spore-print-color, population, habitat. It has 8124 records. Class has two values of p (Poisonous) and e (Edible).   1. Data preparation   In this dataset there is no any null values. Then applied label encoder to convert categorical data into numeric data. Then applied standardization method to scale the data. Then removed veil-type feature as it contains all the value as 0.   1. Visualization   Checked for correlation between the data by plotting heatmap and found there is a strong correlation between gill-attachment and veil-color of value 9.  Then plotted distribution graph to check the value distribution and with respect to class. All feature are in multi-modal distribution. So kept all the features for our model.   1. Model Development   In development of model, used various algorithms of all classes such as linear, non-linear, local and generative.  Linear algorithms used are:   * Logistic regression   It uses Sigmoid function to map the data between 0 and 1.  *σ*(*z*)=1/1+*e*−*z*1​  First it calculates the linear combination of all the features  *z*=(∑*i*=1*n*​*wi*​*xi*​)+*b* | then applies the sigmoid function to it. Then classifies the input as 0 or 1 based on threshold value usually as 0.5.  *P*(*y*=1) = *σ*(*z*)  *P*(*y*=0) = 1−*σ*(*z*)  It uses Gradient descent to calculate the loss function.   * SGD   In this algorithm it gradually minimize the error. It will repeat the process until it get the minimum error.  Non-linear algorithms used are:   * Decision Tree   It is like a flowchart where it consists of root, internal node and leaf node. In every step it classify based on each feature. Branches will have Yes/No. The depth of decision tree also can be decided for best result to avoid overfitting   * Random Forest   It will construct the multiple decision trees and decides based on the voting of each decision trees. It uses bagging method to build multiple decision trees and combines the output for the better result   * Stacking classifier   This will uses multiple models for the better prediction. It uses **meta-model** to learn how to best combine the predictions of base models.  Local type algorithm used is:   * K-Nearest Neighbour   It predicts the new data point by looking at the ‘K’ closest point. It chooses K points and calculates the distance between each K points |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| for new data points. Then finds the nearest neighbours and decides based on majority votings.  Euclidean Distance formula is used to calculate the distance.  *d*=∑i=1*n*​(*x*2i​–*x1i*​)2​  Generative type algorithms used are:   * Naïve Bayes Multinomial   It assumes all the features of independent of each other. It classifies based on the frequency.  To calculate the probability of a message belonging to a certain category Multinomial Naive Bayes uses the **multinomial distribution**:    In this two methods are used for training and testing.   1. Train-test-split   Here 50% of data is used for training and 50% is used for testing. And applied the algorithms such as Naïve bayes multinomial, Gaussian naïve bayes, KNN, SGD and Stacking classifier.   1. Cross validation   10 fold cross validation is use for training and testing of data and applied algorithms such as Naïve bayes multinomial, Gaussian naïve bayes, KNN, SGD and Stacking classifier, Logistic regression, Decision tree and Random forest.   1. Model Evaluation   Accuracy, Recall, Precision, F1-Score are used for evaluating model.    **Accuracy:**  Accuracy represents the number of correctly classified data instances over the total number of data instances. | **Precision:**  *Precision* should ideally be 1 (high) for a good classifier. *Precision* becomes 1 only when the numerator and denominator are equal i.e *TP = TP +FP*, this also means *FP* is zero. As *FP* increases the value of denominator becomes greater than the numerator and *precision* value decreases.    **Recall:**  *Recall* should ideally be 1 (high) for a good classifier. *Recall* becomes 1 only when the numerator and denominator are equal i.e *TP = TP +FN*, this also means *FN* is zero. As *FN* increases the value of denominator becomes greater than the numerator and *recall* value decrease.    **III Results**  In this experiment, we divided the data into 50% training and 50% testing. It gives the result as follows:   |  |  | | --- | --- | | Multinomial NB | 79.99% | | Gaussian NB | 93.05% | | SGD | 96.45% | | KNN | 99.68% | | Stacking | 100% |   In this experiment, we are getting 100% accuracy with Stacking model. In stacking model, Random forest and SVC are considered as a base learner and Logistic regression as a meta learner. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| As an other experiment, we took 10-fold cross validation for training and testing. It gives the result as follow:   |  |  | | --- | --- | | Multinomial NB | 75.76% | | Gaussian NB | 84.64% | | SGD | 88.49% | | KNN | 95.02% | | Stacking | 93.28% | | Logistic regression | 95.69% | | Decision tree | 96.11% | | Random forest | 96.57% |   In this experiment, we got 96.57% mean accuracy in random forest. In this n\_estimator is considered as 150 for the better prediction.    After conducting two experiments which are by splitting and cross validation, we compared the accuracy of different models with splitting and cross | validation. Even though splitting is giving more accuracy with 100% in stacking, cross validation is giving 93.28% which means with splitting as it uses part for training and testing, it may cause some bias. Also it is a sign of overfitting of the model which is not acceptable. In cross validation, it convers all pattern in the data as it takes parts of data and do both training and testing. So with this experiment, we will consider Random forest model with accuracy level of 96.57% as the best.  Other than this, we tried with feature extraction methods like filter-CFS method and wrapper method but we couldn’t able to reach better accuracy as the model giving by considering all features.  **Conclusion:**  In our work, focused on building a better model with less bias and better accuracy model. It classifies mushroom class into ‘edible’ or ‘poisonous’. The dataset from Kaggle has been taken and reached up to mean accuracy of 96.57% accuracy in Random forest model. Other than accuracy we checked for Precision, Recall. F1-Score and support.  References:  [1] Narumol Chumuang1, Kittisak Sukkanchana2, Mahasak Ketcham3, Worawut Yimyam4, Jiragorn Chalermdit5, Nawarat Wittayakhom6, Patiyuth Pramkeaw7 “Mushroom Classification by Physical Characteristics by technique of k-Nearest Neighbor”  [2] Julian White a, Scott A. Weinstein a, Luc De Haro b, Regis Bédry c, Andreas Schaper d, Barry H. Rumack e, Thomas Zilker f “Mushroom poisoning: A proposed new clinical classification”  [3] John Heland Jasper C. Ortega1, Ace C. Lagman2, Lizel Rose Q. Natividad3, Emilsa T. Bantug4, Michael R. Resureccion5, LanzJimuel O. Manalo6 “Analysis of Performance of Classification Algorithms in Mushroom Poisonous Detection using Confusion Matrix Analysis” |