Mushroom Quality Prediction using Machine Learning Classification

Ankita Samantara
Asst. Professor
MITS, BBSR,
Odisha

Tapaswini Nayak, PhD
Asso. Professor
MITS, BBSR,
Odisha

Bijayalaxmi Parida
Asst. Professor
MITS, BBSR,
Odisha

ABSTRACT
For machine learning applications, classification is the first step in grouping, dividing, categorization and separation of dataset based on the feature vectors. Mushrooms are the most familiar delicious food which is cholesterol free as well as rich in vitamins and minerals. Though nearly 45000 species of mushrooms having known throughout the world, most of them are poisonous and few are lethally poisonous. In this Project we focus on the use of classification Techniques such as Bayes and Functions classifiers to predict the quality of mushroom for its edibility. For performing the experiment we will use a mushroom dataset which is available in UCI Machine Learning Repository, which includes descriptions of hypothetical samples corresponding to 23 species of Gilled Mushrooms in Agaricus and Lepiota family. We will use different Machine Learning algorithms to check the quality of mushroom and analyze which algorithm performs better. For experimentation purpose, we have used WEKA tool and the Mushroom dataset used in our work is downloaded from Mushroom Dataset Datahub. The performance of different techniques is evaluated using different parameters such as accuracy, MAE, and kappa statistics.

Keywords
Machine learning Techniques, WEKA tool, Machine learning Classifiers.

1. INTRODUCTION
Mushroom is one type of fungus type plant containing no chlorophyll and it has become so popular presently because of having numerous significant nutrition like niacin, riboflavin, selenium, potassium and vitamin D which are precluding of hypertension, Alzheimer, Parkinson and high risk ofstroke [1]. It is a natural agent that helps to promote the environment of the world. It also helps in the recovery of contaminated damaged habitats and acts as natural pesticide and also supplies sustainable fuel Econol. Mushrooms are a good source of proteins, vitamins, fats, carbohydrates, amino acids and minerals. It is apparent from the present study that the Agaricus and Lepiota mushrooms are well represented in the North India. Many of the investigated mushrooms are edible and are being collected by local inhabitants for consumption. When evaluated for their nutritional and nutraceutical constituents, these mushrooms have been documented to possess substantial amount of protein, which is much more than the common vegetables. Presence of good amount of macro and micro nutrients and meager amount of heavy metals, which are well within the prescribed limits of edibility parameters as fixed by FAO, is yet another property of these mushrooms which accounts for their nutritional credentials. Amongst the various antioxidants evaluated, the amount of phenols is much more in the mushrooms as compared to other antioxidants like β–carotene, lycopene, alkaloids, flavonoids and ascorbic acid. Besides, these mushrooms are also an excellent source of other valuable nutritive constituents like Vitamin C, Vitamin B_1, and Vitamin B_2 as is the case with other mushrooms. There is a need to domesticate these mushrooms. Documentation of information on edible, medicinal and poisonous mushrooms as well as the different social and cultural practices associated with their use in ethno medicinal practices in different parts of India is very important to sensitize the communities about the value of these mushrooms. Various mushrooms have been highly valued as food, as tonics and, in some cases, as medicine for a long period of time. Mushrooms have become more popular in recent years, as can be witnessed by the increased demands for higher production volumes. Their popularity is derived from three highly desirable characteristics as food [2] they have remarkable taste and flavor [3] they are nutritious, not only because they contain high contents of protein with significant amounts of lysine and methionine (which are low in plants), fibers, minerals, and vitamins, but also for what they do not have (high calories, sodium, fat, and cholesterol) [4] they can be easily processed, dried, pickled and canned to allow maximum storage and transportation.

The ethno mycological knowledge among communities that forage mushrooms is based on oral communication handed down from generation to generation which is not a reliable safeguard. In countries where mushrooms are highly consumed, a number of intoxications are reported every year mainly due to misidentification of mostly wild species. Hazardous toxins are present in these species and are able to cause different syndromes that can be fatal depending on the amount ingested. Accidental ingestion of mushrooms is difficult to avoid especially in countries where eating wild species is common. Proper identification is important to avoid accidents and the identification ofsymptoms and signs of intoxication as soon as possible enables the success of treatment.

Cultivating mushroom in scientific way reduces the probability to occur poison in mushroom yield. In our country 4 kinds of mushroom are available namely button mushroom, oyster mushroom, paddy Straw mushroom and milky mushroom. There are around 45000 type funguses available in the world. Among them around 2000 fungus are edible vegetable foods [5]. Unexpectedlyidentifying the edibility of mushroom manually is a too difficult task. Because maximum poisonous mushrooms look like edible mushrooms due to color and shape. So automation is very important in this field to reduce time and labor.

There are many classification approaches exist in machine learning. The main objectiveof thesis is to study the impact of Sequential minimal optimization (SMO), Naïve Bayes classification algorithm on the mushroom classification dataset in WEKA. The parameters for judging the algorithms are correctly classified instances, incorrectly classified instances, error rate and precision. These are helpful when training data is used instead of testing data and comparing them to know the correctly classified instances, incorrectly classified instances, error rate and precision of the particular algorithm. This thesis is

conducted by using an experimental method and assisted tool of WEKA. Thistool has small size with faster loading time with a simple interface and do not require many sources ofdata in the data processing.

2. LITERATURE SURVEY

There are different researches using different techniques that are used for mushrooms classification, a mushroom diagnosis assistance system (MDAS) was proposed by [6], which involves three components of web application (server), unified database and mobile phone application (client) which is used on mobile phone devices. The naïve Bayes and decision tree classifiers are used to determine the mushroom types. Firstly, the suggested system chooses the most known mushroom attributes. Secondly, specify the mushroom type. The experiment result shows that decision tree classifier is better than naïve Bayes classifier in correct and incorrect classified instances and error measurements.

Lavanya et al. [7] used a different kind of classification algorithms to identify whether the mushroom is edible or not. These algorithms are evaluated using accuracy, mean absolute error and kappa statistic. Bayes net, naïve Bayes and ZeroR are used for classification. But the classifier's accuracy rate is low when the dataset is small and their performance increase with the increasing dataset. The conclusion is Bayes Net has the best result in this scenario And ZeorR has the worst performance.

In paper [8], mushroom classification is done using a different kind of features of mushrooms such as gill's type or color shape or size, color of the cap, population, and odor. Here, Principal Component Analysis (PCA) is used to identify the mushroom type and gives the highest accuracy to differentiate between poisonous and edible mushrooms by applying decision tree algorithm. J48 is used to produce a decision tree. PCA is applied to the decision tree and for ranking the features. The dataset which is used here has 22 attributes, 3916 poisonous mushrooms and 4208 edible mushrooms. After applying PCA the highest-ranking attribute is an odor that means among those 22 attributes the contribution of odor is highest to classify the mushroom.

Agung wibowo et al. [9] compared the performance among three data mining algorithms: C4.5 based decision tree, naïve Bayes and SVM. For performing the experiment, dataset is taken from Audubon society field guide to North American mushrooms, available in the UCI machine learning repository [10] which includes Agaricus and Lepiota families of mushroom. Both C4.5 and SVM have better accuracy than naïve Bayes. Between C4.5 and SVM, C4.5 is faster than SVM by 0.02 seconds. Therefore C4.5 is considered as the best among these three algorithms. In addition, C4.5 discards 5 from 22 attribute and classify based on these five attributes which are the odor, spore - print- color, gill-size, gill-spacing and population.

3. PROPOSED WORK

3.1 Methodology

The following are the steps included in the classification process carried out in this work:

- We retrieved the mushroom dataset from mushroom dataset datahub [11] for the classificationprocess.
- The ARFF file is downloaded and is pre-processed to meet the requirements of the type ofanalysis that we are seeking.
- Two different classifiers namely sequential minimal optimization (SMO) and naïve Bayes arechosen for the classification process.
- After running the 2 classifier their accuracy, correctly classified instances, incorrectly classified instances, error rate and precision of each classifier are calculated.
- We are running this dataset in the WEKA tool of version 3.9.4.
- The results of the two classification algorithm are compared to determine which algorithmgives the best accuracy.
- Finally the result are analysed and the best suited algorithm for the chosen dataset is found and the performance is analysed.

3.2 Description About the Mushroom Dataset

The mushroom dataset is retrieved from the mushroom dataset datahub [11]. It contains descriptions of hypothetical samples which are corresponding to 23 species of the gilled mushroom in the Agaricusand Lepiota family. Each one of those species is identified as the definitely edible, definitely poisonous, unknown edibility, or is not recommended at all. The latter class has been combined with the poisonous and edible based on 22 physical attributes as recorded in [12].

3.2.1 Data Distribution

The dataset was distributed into two different classes:

- Class 1 = edible with the number of 4208 instances (51.28%).
- Class 2 = poisonous with the number of 3916 instances (48.2%).

There are total 8124 instances of mushroom.

3.2.2 Setting a Target

The main target of this project is determining if mushroom is edible (Y) or poisonous (N).

3.3 Mushroom Attributes

The 22 attributes of mushroom in the dataset are represented in the table 1. These attributesrepresent well known mushroom attributes. Hence a string of size 22 bytes(22 characters)is required to describe each mushroom. All the mushroom description strings are of the same size because each attribute has only one possibility [12].

There are 8124 instances in dataset. These instances are distributed as 4208 edible mushrooms and 3916 poisonous mushrooms. Stalk-root attribute, which is numbered 11, misses 2480 attributes. The sign "?" is used to note the missing attribute in dataset and same sign is used in string describing of the mushroom when attribute is missing [12]. The binary vector 0-1 can be used to describe the mushroom instead of using string because the data of mushroom is nominal.

TABLE 1 Attributes Of Mushroom Description In The Dataset

ATTRIBUTE NUMBER	ATTRIBUTE NAME	POSSIBILITES
1	cap-shape	bell=b, canonical=c, convex=x, flat=f, knobbed=k , sunken=s

2	cap-surface	fibrous=f, grooves=g ,scaly=y , smooth=s
3	cap-colour	Brown=n, buff=b, cinnamon=c, gray=g, green=r, pink=p, purple=u, red=e, white=w, yellow=y
4	bruises?	bruises=t, no=f
5	odor	almond=a, anise=l , creosote=c, fishy=y, foul=f, musty=m , none=n, pungent=p, spicy=s
6	gill-Attachment	attached=a, descending=d , free=f ,notched=n
7	gill-spacing	close=c, crowded=w, distant=d
8	gill-size	broad=b, narrow=n
9	gill-color	black=k, brown=n, buff=b, chocolate=h, gray=g, green=r, orange=o, pink=p, purple=u, red=e, white=w, yellow=y
10	stalk-shape	enlarging=e, tapering=t
11	stalk-root	Bulbous=b, club=c, cup=u, equal=e, rhizomorphs=z, rooted=r, missing=?
12	stalk-surface-above-Ring	fibrous=f, scaly=y ,silky=k, smooth=s
13	stalk-surface-below ring	fibrous=f, scaly=y, silky=k, smooth=s
14	stalk-color-above-ring	brown=n, buff=b, cinnamon=c, gray=g, orange=o, pink=p, red=e, white=w, yellow=y
15	stalk-color-below-ring	brown=n,buff=b, cinnamon=c,gray =g,orange=o,pink =p,red=e,whi te=w,yellow=y
16	veil-type	partial=p, universal=u
17	veil-color	brown=n, orange=o, white=w, yellow=y
18	ring-number	none=n, one=o, two=t
19	ring-type	cobwebby=c, evanescent=e, flaring=f, large=l, none=n, pendant=p, sheathing=s, zone=z
20	spore-print-color	black=k, brown=n, buff=b, chocolate=h, green=r, orange=o, purple=u, white=w, yellow=y
21	population	abundant=a, clustered=c, numerous=n, scattered=s, several=v, solitary=y
22	habitat	grasses=g, leaves=l, meadows=m, paths=p, urban=u, waste=w, woods=d

When an attribute includes a value 1 is used and 0 is used when the attribute does not include values. The attributes have the definite length because there are only certain possibilities for describing the attribute i.e. for attribute 21 – population, there 6 options- abundant (a), clustered (c), numerous(n), scattered (s), several (v), solitary (y). The feature scattered (s) can be represented as the binary string 000100 [12]. The proposed system based on these attribute in taking the decision. For example – Mushroom is most likely poisonous if: spore-print-color is green. While mushroom is most likely edible if: odor is almond, anise, or no smell at all.

3.4 Preprocess of Data in Weka

The data that is collected from the field contains many unwanted things that lead to wrong analysis. Thus the data must be preprocessed to meet the requirements of the type of analysis that we are seeking. We retrieved the mushroom's ARFF file from mushroom dataset datahub [11]. After downloading the ARFF file we stored it in the download folder. After opening the WEKA we will click on open file option which is under the preprocess tag and will select the mushroom_arff.arff file.
After opening the file the screen looks like this:

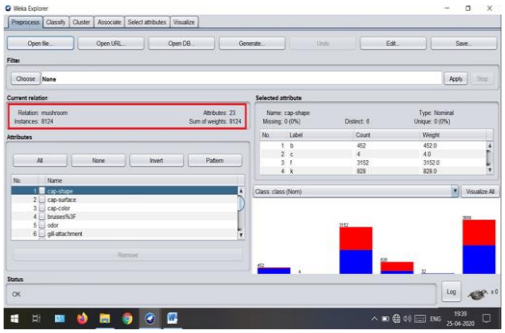

Figure 1 The highlighted part is the current relation sub-window

In the above Figure 1 the highlighted part is the current relation sub-window that shows the name of the database that is currently loaded. There are 8124 instances which represent the number of rows in the table. The mushroom database contains 23 attributes, when we select an attribute from this list; further details on the attribute itself are displayed on the right hand side.
Let us select the cap-shape attribute first, when we click on it, we would see the following screen

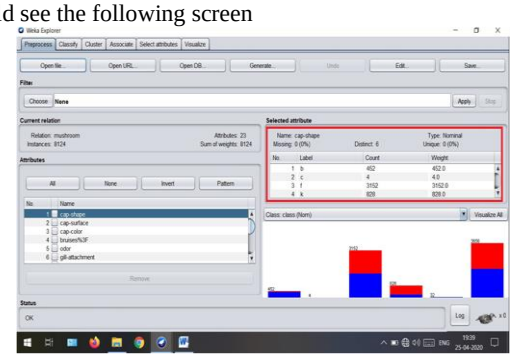

Figure 2 The highlighted part is selected attribute sub-window

In the above Figure 2 the highlighted part is selected attribute sub-window. In the selected attribute sub window we can observe the following:

- The name and the type of the attribute are displayed.
- The type for the cap shape attribute is nominal; the number of missing values is zero.
- There are 6 distinct values with no unique value.
- The table underneath this information shows the nominal values for this field as b = bell, c = canonical, f= flat, k = knobbed, x = convex, s = sunken.
- It also shows the count and weight in terms of a percentage for each nominal value.

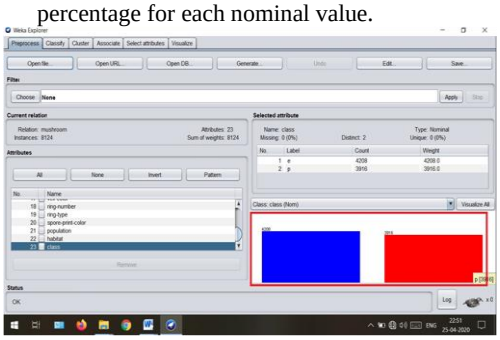

Figure 3 The highlighted part is class values which are visually represented

In the above Figure 3 the highlighted part is the visual representation of the class values that shows the following things:

- At the bottom of the window, we can see the visual representation of the class values.
- If we select class attribute then we can see at the visualize box that there are blue and red color histogram which shows that 4208 instances are edible which are represented as blue color and 3916 instances are poisonous which are represented as red color.

3.5 Classification in Weka

The concept of classification is basically to distribute data among the various classes defined in a dataset. Classification algorithms learn this form of distribution from a given set of training and then try to classify it correctly when it comes to test data for which the class is not specified. The values that specify these classes on the dataset are given a label name and are used to determine the class of data to be given during the test [17]. So after processing data now we will classify the data.

3.5.1 Setting Test Data

We will use the pre-processed mushroom data file. We will click on classify tab which is next to the preprocess tab and would see the following screen.

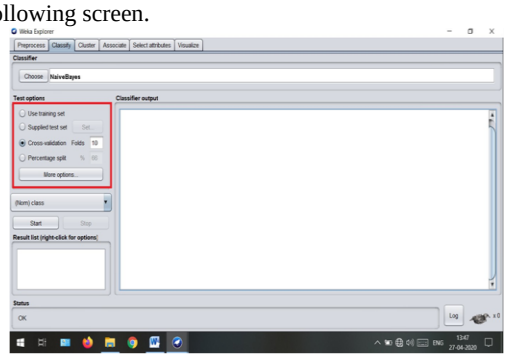

Figure 4 The highlighted part shows the following test option

In this we can notice the four testing options:

- Use Training set: - It classifies our model based on the dataset which we originally trained our model with.
- Supplied test set: - It controls how our model is classified based on the dataset that we supplied from externally. We can select dataset file by clicking the set bottom.
- Cross-validation
  - The cross-validation option is widely used one, especially if we have limited amount of datasets.
  - The number we enter in the fold section are used to divide our dataset into fold numbers.
  - The original dataset is randomly partitioned into 10 subsets.
  - After that WEKA uses set1 for testing and 9 sets for training for the first training. Then use set 2 for testing and the 8 sets for training and repeats that 10 times in total by incrementing the set number each time. In the end, the average success rate is reported to the user.
- Percentage split
  - It divides our dataset into train and test according to the number we enter.
  - By default the percentage values is 66% it means 66% of our dataset will be used as training set and other 33% will be our test set.

3.5.2 Selecting Classifier

We can select the classifier which we want to use by clicking on the choose bottom and can select any classifier.

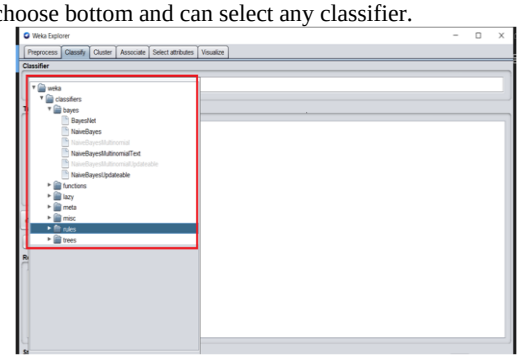

Figure 5 The highlighted part shows the following classifiers present in the WEKA tool

After selecting the naïve bayes we click on the start button to start the clasification process. After a while , the classification result would be presented on our screen as shown below.

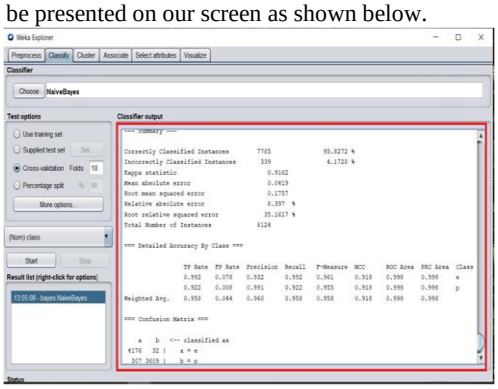

Figure 6 The highlighted part shows the output of the classifier chosen

3.6 Machine Learning Classification Algorithms

3.6.1 Naive Bayes

Naïve Bayesian classification is a supervised learning process. In addition, it is a statistical method for classification purposes. This classification is based on Bayesian theorem. Bayes Naïve Bayesian classifiers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence. It is made to simplify the computations involved, in this sense, is considered "naïve" [13]. It looks attractive on when the dimensionality of the supplied inputs is high. The process of maximum likelihood is being used for parameter estimation in naïve Bayesian models. Let D be a training set of tuples and their associated class labels. The representation of every tuple is being done by an n- dimensional attribute vector, X= (X_1,X_2,X_3, ….., X_n). Let there are m Classes C= (C_1,C_2,C_3,….., C_m). Under given dataset, the Naïve Bayesian classifiers will predict that given a tuple X belongs to the class having the highest posterior probability, conditioned on X. That is, the naïve Bayesian classifier predicts that tuple X belongs to the class Ci if and only if,

$$\mathbf{P(C_i / X) > P(Cj/ X) \text{ for } 1<= j <= m, j \neq i}$$

Thus, we maximize P (C_i / X). The class C_i for which P (C_i / X) is maximized is called the maximum posteriori hypothesis [13]. By Bayes' theorem

$$\mathbf{P(Ci / X) = P(X / Ci)\, P(Ci) / P(X).}$$

Only P (X / Ci) P (Ci) is maximized since P(X) is constant. Under given data sets with many attributes, it would be extremely computationally expensive to evaluate P (X / Ci). For reduction of computation, the naïve assumption of class-conditional independence is made.

$$\mathbf{P(X / C_i) = \prod_{k=1}^{n} P(x_k / C_i)}$$

$$\mathbf{= P(x_i / C_i) \times P(x_2 / C_i) \times P(x_3 / C_i) \times \ldots\ldots\ldots P(x_n / C_i)}$$

Bayesian classifiers have the minimum error rate in comparison to all other classifiers [13].

3.6.2 Sequential Minimal Optimization (Smo)

It is the extension of Support Vector Machines (SVM). It is a method for the classification of both linear and non-linear data [13]. SMO was developed by John Caplet of Microsoft research and ibis used to train the SVM faster. The goal of SVM is to search the linear optimal separating hyper plane (i.e. "decision boundary") that can separate two classes with the largest distance (i.e. "gap" or "margin") within a border line (support vectors) [14] .If data are linearly inseparable, original input data is transformed into a higher dimensional space with the help of a nonlinear mapping constructed through mathematical projection ("kernel trick") where separating decision surface is found. After that, a linear separating hyper plane is searched in new space. The maximal marginal hyper plane found in the new space corresponds to a non-linear separating hyper surface in the original space [13] .The training time of SVM might be slow but because of their ability to model complex nonlinear decision boundaries it is accurate to learn both simple and high complex classification models, and avoids over fitting by using complex mathematical principles.

Sequential Minimal Optimization (SMO) algorithm which is the new efficient technique for training SVMs. SMO breaks the very large quadratic programming (QP) optimization problem occurred in SVM training into a sequence of minimal possible QP problems involving only two variables, and each of these problems is solved analytically. SMO repeats until all the patterns satisfy the optimality conditions [15].The SMO algorithm needs less amount of memory, thus very large SVM training problem can accommodate in the memory of a personal computer, as a result, large matrix computation is avoided .SMO algorithm selects two Lagrange multipliers α1 and α2 and optimizes the objective value for both these α's. Finally it adjusts the b parameter based on the new α's. This process is repeated until the α's converge [16] .SMO update two Lagrange multipliers as a SMO Step as shown below

Given two examples E1 and E2:

$$\boldsymbol{\alpha_2^{new} = a_2 + y_z(E2-E1)\div n}$$

Where $\eta = K(\vec{x}_1, \vec{x}_1) + K(\vec{x}_2, \vec{x}_2) - 2K(\vec{x}_1, \vec{x}_2)$ Clips the value at the end of the segment:

$$a_2^{new,clipped} = \begin{cases} H & if \quad a_2^{new} \geq H; \\ a^{new}_2 & if \quad L < a_2^{new_2}; \\ L & if \quad a_2^{new} \leq L; \end{cases}$$

If $y_1 = y_2$ then:

$L = \max(0, \alpha_2 + \alpha_1 - C)$

$H = \min(C, \alpha_2 + \alpha_1)$

Otherwise:

$L = \max(0, \alpha_2 - \alpha_1)$

$H = \min(C, C + \alpha_2 - \alpha_1)$

$\alpha new = \alpha_1 + s\ (\alpha_2 - \alpha^{new\ ,clipped})$
Where $s = y_1\ y_2$
Major components of SMO are an analytical method to solve for two Lagrange multipliers.

4. RESULTS AND DISCUSSION

The performance of the SMO classifier and Naïve Bayes classifier on mushroom dataset is evaluated using the accuracies obtained in each fold of the cross-fold validation. The 10 – fold cross validation is performed on mushroom data which means that the whole training dataset is divided into 10 subsets of equal length , each of which was in turn used as an independent test dataset. In each subset 75% data is used for training the classifier and 25 % of data is used for testing the classifier.

TABLE 2 Comparative Result Analysis Of SMO And NAÏVE BAYES On Mushroom Dataset

Datasets	Classifiers	Accuracy in %	Kappa Statistics	MA Error	RMS Error	RA Error	RRA Error
MUSHROOM	SMO	100 %	1	0	0	0	0
	NAIVE BAYES	95.8272	0.9162	0.0419	0.1757	8.397	35.1617

=== RUN INFORMATION ===
Scheme: weka.classifies.bayes.naive Bayes
Relation: mushroom
Instances: 8124
Attributes: 23
Cap-shape
Cap-surface
Cap-color
Bruises%3F
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
Class

Test mode: 10 fold cross validation
Run information gives us the following information

1. The algorithm which we have used: naïve Bayes.
2. The relation name -> "mushroom".
3. Number of instances in the relation – 8124 and the list of attributes are given.

===SUMMARY===
Correctly classified instances: 7785 95.8272%
Incorrectly classified instances: 339 4.1728%
After using the naïve Bayes algorithm the result shows that the correctly classified instances are 7785 out of 8124 instances which means that 7785 are edible. So the accuracy of naïve-Bayes algorithm is 95.8%. Incorrectly classified instances are 339 out of 8124 which means that 339 are poisonous.
The performance of algorithms can be evaluated using the parameters given below:

1. **KAPPA STATISTICS -**

- Kappa statistics play a significant role in term of classification in mushroom dataset. It is a chance corrected measure of agreement between the classifications and the true classes of the entire dataset.
- Kappa is actually calculated by taking the agreement expected by chance away from the observed agreement and dividing by the maximum possible agreement. When a value is greater than 0, that means the classifier is doing better than chance.
- The kappa statistic (or value) is a metric that compares an observed accuracy with an expected accuracy (random chance). The kappa statistic is used not only to evaluate a single classifier, but also to evaluate classifiers amongst them. In addition it takes into account random chance (agreement with a random classifier), which generally means it is less misleading than simply using accuracy as a metric.
- The equation for k is:

$$K = P_o - P_e / 1 - P_e$$
$$= 1 - 1 - P_o / 1 - P_e$$

- Where Po is the relative observed agreement among rates, and Pe is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly saying each category. If the rates are in complete agreement then $\kappa = 1$. If there is no agreement among the rates other than what would be expected by chance (as given by Pe), $\kappa \leq 0$.

- A kappa value of 0 means that the result is the same as would be expected by chance.

2. **MEAN ABSOLUTE ERROR -**

- Another important protagonist works here in this study is mean absolute error (MAE). It measures the average magnitude of the errors in a set of forecasts, without considering their direction thereof. It measures accuracy for continuous variables. It is the average over the verification sample of the absolute values of the differences between forecast and the corresponding observation.
- Mean absolute error is the quantity which is used to measure how close forecasts or predictions are to the eventual outcomes.
  The mean absolute error is given by:
  $MAE = \mathbf{1}\ \sum \boldsymbol{n} |\mathbf{fi} - \mathbf{yi}| = \mathbf{1}\ \sum \boldsymbol{n}$
  $|\mathbf{ei}|\ \boldsymbol{ni}=\mathbf{1}\boldsymbol{ni}=\mathbf{1}$
  As the name suggests, the mean absolute error is an average of the absolute errors.

|ei| = |fi - yi| Where fi is the prediction and yi is the true value [18].
The mean absolute error is like the variance, but rather than squaring the difference we use its absolute value.

3. **ROOT MEAN SQUARED ERROR -**

- Root mean squared error (RMSE) measures the average magnitude of the error by using quadratic scoring rule. Here the difference between forecast and corresponding observed values are each squared and then averaged over the sample.
- It is the measure of the differences between values predicted by a model or an estimator and the values actually observed. It represents the sample standard deviation of the differences between predicted values and observed values. It aggregates the magnitudes of the errors in predictions for various times into a single measure of predictive power. It is a good measure of accuracy, but only to compare forecasting errors of different models for a particular variable and not between variables as it is scale dependent. It is also called the root mean square division, RMSD.
The RMSD = $\sqrt{\sum_{t=1}^{n}} (\hat{y}_t - y)^2 \div n$
- The RMSD of predicted values yt for times t of a regression dependent variable y is computed for n different predictions as the square root of the mean of the Squares of the deviations [19].

4. **RELATIVE ABSOLUTE ERROR -**

- Relative absolute error is a way to measure the performance of a predictive model. It is expressed as a ratio comparing a mean error to errors produced by a trivial or naïve model [20].
- RAE = mean of the absolute value the actual forecast errors / mean of the absolute values of the naïve model's forecast errors.

5. **ROOT RELATIVE SQUARED ERROR -**

- This predictor is just the average of the actual values.
- The relative squared error takes the total squared error and normalizes it by dividing by the total squared error of the simple predictor.
- By taking the square root of the relative squared error one reduces the error to the same dimensions as the quantity being predicted [21].
- The difference between relative and absolute is that the absolute error is how much our result deviates from the real value which relative error is measure in percent compared to the real value.

4.1 Confusion Matrix

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrected predictions are summarized with count values and broken down by each class. The confusion matrix shows the ways in which our classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.

	Class a predicted	Class b predicted
Class a actual	TP	FN
Class b actual	FP	TN

Here, Class a: edible
Class b: poisonous

- True positive (TP):- observation is positive and is predicted to be the positive.
- False negative (FP):- observation is positive, but is predicted negative.
- True negative (TN):- observation is negative, and is predicted to be negative.
- False positive (FP):- observation is negative, but is predicted positive.

4.2 Accuracy

Accuracy is measured in terms of correctly classified instances. It is calculated and represented in percentage.
Accuracy = TP + TN / TP +TN + FP + FN

= 4176 + 3609 / 4176 + 3609 + 307 + 32

= 7785 / 8124

≅ 0.9582

= 95.82%

4.2.1 Tp Rate

It is the rate of true positive instances correctly classified as a given class. It is the number of examples predicted positive that are actually positive. It is the proportion of example which was classified as class x, among all examples which truly have class x, i.e. how much of class was captured correctly. We can get the TP rate by seeing the confusion matrix.

a b <-- classified as

4176 32 | a = e

307 3609 | b = p

TP rate for edible (TPR) = TP / TP + FN

= 4176 / 4176 + 32

= 0.992

TP rate for poisonous (TNR) = 1-FPR

= 1 – 0.078

= 0.922

4.2.2 Fp Rate

It is the rate of false positives instances falsely classified as a given class. It is the number of example predicted positive that are actually negative. The FP rate is the proportion of the examples which were classified as class x, but belong to a different class among all examples which are not of class X.
FP rate for edible (FPR) = FP /FP+TN = 307/307+3609 = **0.078**
FP rate for poisonous (FNR) = 1- TPR = 1- 0.992 = **0.008**

4.3 Precision

To get the value of precision divide the total number of correct classified positive examples by the total number of predicted positive examples.

Precision value for edible = 1 – FDR = 1- 0.068 = **0.932**
(Where FDR = FP/ FP + TP = 0.068)
Precision for poisonous = TN / predicted poisonous = 3609 / 3641 = **0.991**

RECALL
It can defined as the ratio of total number of correctly classified positive examples divide to the total number of positive examples divide to the total number of positive examples. High recall indicates the class is correctly recognized.
Recall for edible = TP / actual edible = 4176 / 4208 = **0.992**
Recall for poisonous = TN / actual poisonous = 3609 / 3916 = **0.922**

F- MEASURE
It is a combined for precision and recall calculated as
F- Measure for edible = 2* (precision * recall / precision + recall)
= 2 * (0.932 * 0.992 / 0.992 + 0.932)
= **0.961**
F- Measure for poisonous = 2* (0.992 * 0.922 / 0.992 + 0.922)
= **0.955**

MCC
MCC is used as a measure of the quality of binary (two- class) classifications. It takes into account true and false positives and negatives and is generally regarded as a balanced measure which can be used even if the classes are the very different sizes.
It is a correlation coefficient between the observed and predicted binary classifications; it returns a value between -1 and +1.
The MCC can be calculated directly from the confusion matrix using the formula:

MCC = (TP * TN) – (FP * FN) / $\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}$

MCC for edible = 0.918
MCC for poisonous = 0.918

ROC AREA (RECIVER OPERATING CHARACTERISTICS)
It is one of the most important values output by WEKA. They give us an idea of how the classifiers are performing in general. The purpose of ROC area is to examine the performance of a binary classifier by creating a graph of the true positive vs. false positive for every classification threshold.

PRC AREA (PRECISION RECALL)
The precision recall plot is more informative than ROC plot when evaluating binary classifier on imbalanced datasets. The PRC area is calculated separately for each class by treating instances of the class as "positive" instances and instances of all other classes as "negative" instances.

4.4 Weighted Average

When evaluating multi class classification models, WEKA outputs a weighted average of per class precision, recall, and F-measure. It computes these statistics for each class individually, treating the corresponding classes as the "positive class" and the union of the other classes are "negative class" and computes a weighted average of these per class statistics with a per class weight that is equal to the proportion of data in the class.
Weighted average = [(value for E class * no. of instances from E class) + (value for P class * of instances from P class) / total instances in dataset]

Number of instances from edible class = 4208 Number of instances from poisonous class = 3916 Total instances in dataset = 8124

4.4.1 Weighted Average Of Naïve Bayes

TP RATE

EDIBLE = 0.992 = 0.992*4208 = 4174.336
POISONOUS = 0.992 = 0.992*3916 = 3610.552
Weighted Avg. of TP rate = 4174.336 + 3610.552 / 8124 = 7784.888 / 8124 = **0.958**

FP RATE
EDIBLE = 0.078 = 0.078*4208 = 328.224
POISONOUS = 0.008 = 0.008*3916 = 31.328
Weighted Avg. Of FP Rate = 328.224 + 31.328 / 8124 = 359.552 / 8124 = **0.044**

PRECISION
EDIBLE = 0.932 = 0.932*4208 = 3916.856
POISONOUS = 0.991 = 0.991*3916 = 3880.756
Weighted average of precision = 3921.856 + 3880.756 / 8124 = 7802.612 /8124 = **0.960**

RECALL
EDIBLE = 0.992 = 0.992*4208 = 4174.336
POISONOUS = 0.922 = 0.922*3916 = 3610.552
Weighted average of Recall = 4174.336 + 3610.552 / 8124 = 7784.888 / 8124 = **0.958**

F-MEASURE
EDIBLE = 0.961 = 0.961*4208 = 4043.888
POISONOUS = 0.955 = 0.955*3916 = 3739.78
Weighted average of F-measure = 4043.888 + 3739.78 /8124 =7783.668 /8124 = **0.958**

MCC
EDIBLE = 0.918
POISONOUS = 0.918
Weighted average of MCC = **0.918**

ROC AREA
EDIBLE = 0.998
POISONOUS = 0.998
Weighted average of ROC area = **0.998**

PRC AREA
EDIBLE = 0.998
POISONOUS = 0.998
Weighted average of PRC area = **0.998**

4.4.2 Weighted Average Of Smo

TP RATE
EDIBLE = 1.000 = 1.000 * 4208 = 4208
POISONOUS = 1.000 = 1.000* 3916 = 3916
Weighted average of TP Rate = 4208+3916 / 8124 = 8124/8124 = **1**
FP RATE
EDIBLE = 0.000
POISONOUS = 0.000
Weighted average of FP Rate = 0.000 PRECISION
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of precision = **1.000**

RECALL
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of recall = 1.000 F-MEASURE
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of F-Measure = 1.000

MCC
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of MCC = **1.000**

ROC AREA
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of ROC area = **1.000**

PRC-AREA
EDIBLE = 1.000
POISONOUS = 1.000
Weighted average of PRC Area = **1.000**

COMPARTIVE STUDY AMONG ALGORITHMS

TABLE 3 Weighted Average Of Detailed Accuracy Results For NAÏVE BAYES And SMO Algorithms On Mushroom Dataset

Classifier output	TP Rate	Precession	Recall	F-Measure	MCC	ROC Area	PRC Area	Precession	Recall
Algorithms	NAÏVE BAYES	0.958	0.044	0.960	0.958	0.958	0.918	0.998	0.998
	SMO	1	0.000	1.0000	1.000	1.000	1.000	1.000	1.000

Here the difference between forecast and corresponding observed values are each squared and then averaged over the sample. Finally the Square root of the average is taken into consideration. Since the errors are squared before they are averaged; the RMSE gives a relatively high weight to large errors. Both MAE and RMSE are used to diagnose the variations in the errors in a set of forecasts. Logically RMSE will always be larger or equal to the MAE. A comparative study among the classification algorithms used here has been shown in the table 3.

So after comparing these two algorithms it has been investigated that sequential minimal optimization classifier has become finally more preferable because of less prone to error among all different types of error rate during this experimental analysis. Sequential minimal optimization (SMO) classification algorithm performs best with accuracy 100% with less error rate.

5. ACKNOWLEDGEMENT

We are very obliged and thankful to all those who helped me to complete this paper work successfully. We wish to express my gratitude to project's team of head office who has given the opportunity to publish a paper. I express my sincere thanks to our project guide Dr. Tapaswini Nayak, Associate Professor, MITS SCHOOL OF BIOTECHNOLOGY, Bhubaneswar. for her not only providing us the technical as well as the non-technical support ,but also for her valuable guidance and the encouragement throughout the process to complete each. Last but not the least, i would like to thank to God and my parents who has always helped me and sheltered me under their divine blessings.

6. REFERENCES

[1] M.E Val Verde, T.Hernandez-perez and o.paredeslopez "Review article edible mushrooms: Improving human health and promoting quality life", 2015.

[2] Hawksworth DL (2001) Mushrooms: the extent of the unexplored potential. Int J Med Mushrooms3:333-337.

[3] Chang ST, Miles PG (1992) Mushroom biology: a new discipline. Mycologist 6: 64-65.

[4] Barbato MP (1993) Poisoning from accidental ingestion of mushrooms. Med J Aust 158: 842-847.

[5] D.R. Choudhury and S.ojha, " an empirical study on mushroom disease diagnosis : a data mining approach," 2017.

[6] R.Labarge, "Distinguishing poisonous from edible wild mushroom , "2008.

[7] Beniwal, Sunita and Bishan Das. "Mushroom classification using data mining techniques." International journal of pharma and Bio Science, Vol 6, issue 1, pp.1170-1176, 2015.

[8] 1smail, shuhaida, Amy rosshaida Zainal and Aida Nustapla. "In 2018 IEEE symposium on computer application and Industrial Electronics (ISCAIE), pp.412-415." IEEE, 2018.

[9] Agung widow, Yuri Rahayu, Andi Riyanto and Tauflik HIdayatulloh. "Classification algorithm for edible mushroom identification." In Information and communications technology (ICOIACT), 2018 International conference IEEE 2018. pp. 250 – 253.

[10] "UCI Machine Learning repository: mushroom dataset." [Online]https://archive.ics.uci.edu/ml/datasets/mushroom

[11] 'Mushroom dataset datahub' [online] http://datahub.io/machine-learning/mushroom

[12] G.H Lincoff (pres.) New York, Alfred A. Knopf "Mushroom dataset", mushroom records drawn from Audubon society field guide to North American mushroom (1981).http://archive.ics.uci.edu/ml/machine-learning-databases/mushroom/agaricus-lepiota.names .

[13] Jiwai Han, Michelin Kambe, Jian Pei, Han- "Data mining concepts and Techniques", 3rd edition, Norgan Kaufmann

publishers, imprint of Elsevier, ISBN 078-0-12-381479-1, 2012.

[14] Richard Enyinnaya, "Predicting Cancer-Related Proteins in Protein-Protein interaction Network using Network approach and SMO-SVM algorithm", International Journal of computer Applications, pp.5-9, vol.115, No.3, 2015.

[15] Dmibry Pavlov, Jianchang Mao, Byron Dom, "Scaling-up support vector machines use boosts algorithms", proceeding. 15th International conference on pattern Recognition, vol.2, 2000.

[16] "The simplified SMO algorithm", pp. 1-5, CS-229, Aubumn 2009.

[17] "Beginning to WEKA step by step AYse bilge gunduz [online]" https://code.likeagirl.io/beginning-to-weka-step-by-step-93f6564d9f2

[18] http://en.wikipedia.org/wiki/mean_absolute_error

[19] http://en.wikipedia.org/wiki/root-mean-square deviation [Online]

[20] http://www.statisticshowto.com/relative-absolute-error [Online]

[21] http://www.gepsoft.com/gxpt4kb/chapter10/section1/ss07.html [Online]