**Methodological Mycology: Increased Safety Through Machine Learning**

Tammy Liu, Curtis Von Allman, E R Mudiyanselage Samanthi Chathurani Ekanayake

**ABSTRACT**

Appropriate identification of poisonous mushrooms is a crucial step in food safety with life-impacting results. This study goes beyond traditional approaches and introduces a systematic methodology in the design of experiments. The study aims to contribute to a deeper understanding of machine learning methods, and how to continually refine the method to obtain the best results possible. This study reveals the need for meticulous evaluation of the 11 different machine learning methods used to guide future research in mycology. Data visualizations and multivariate charts are used to further accentuate the results found in this study.

**Keywords**

Mycology; Design of Experiments; Machine learning; Multivariate charts; Naïve Bayes; Mushrooms

**1. Introduction and literature review**

Over 5,000 years ago, mushrooms were used for medicinal purposes (Halpern, 2010). For millions of years, safe edible mushrooms constituted a significant part of human diets (Wang et al., 2014) and have important implications in nutrition, medicine, and biodiversity. Accurate and efficient identification of poisonous mushrooms is important, not only from a public health perspective but also for the potential medicinal benefits that might arise from the fungi. It is common for people to consume wild mushrooms as they can be easily found in forests and meadows, and it is observed that there is a rising interest in eating mushrooms as part of the back-to-nature and organic food trend. According to (Roy Chowdhury & Ojha, 2017), there are approximately 45,000 known species of mushrooms documented globally while only around 2,000 species are classified as edible.

Mushroom poisoning is a global issue that countries suffer from worldwide. China (Chen et al., 2014), Iran (Soltaninejad, 2018), Japan (Gonmori & Yoshioka, n.d.), Turkey (Sönmez, 2015), North America (Broussard, 2001) and Europe (Schmutz et al., 2018) have all reported toxic mushroom species. Mushroom poison cases have also been observed in countries including Czechia (Kieslichova et al., 2018), China (Sun et al., 2018), Mexico (Ruiz-González et al., 2017), Poland (Krakowiak et al., 2017), etc. which shows mushroom poisoning has never been a regional issue.

Mushroom classification has been the subject of extensive research, with the primary goal of distinguishing between poisonous and edible varieties. Various classification algorithms have been employed in these studies to achieve accurate and reliable results. This literature review aims to explore the diverse approaches used in mushroom classification and compare the performance of different algorithms in accurately categorizing mushrooms as either poisonous or edible. By summarizing the findings of these research efforts, valuable insights can be gained into the most effective methods for mushroom classification.

Sisodiya et al. (2023) revealed that there are different types of mushrooms as poisonous, edible, and medicinal mushrooms based on how they impact the human body after their ingestion. According to Govorushko (2019), mushrooms can be poisonous because of improper cooking techniques, long-term storage of freshly collected mushrooms, certain types of mushroom consumption with alcohol, recurrent consumption of mushrooms from Morchellaceae family, storing pre-prepared mushrooms for an extended period. The use of hallucinogenic fungi has the potential to accumulate harmful substances like heavy metals and radioactive materials during their growth. Pests, particularly flat-footed flies, can cause damage to fungi, including mushrooms.

Chelliah et al. (2018) have used supervised machine-learning methods for the classification of mushrooms. According to their study, some machine learning models such as the default decision tree model perform exceptionally well in accuracy and reliability. Using a confusion matrix, C. Ortega (2020) has discovered that the decision tree method has the highest computed accuracy result compared to other classification algorithms such as Logistic Regression, Naïve Bayes, and KNN.

According to the study done by Paudel & Bhatta (2022), the random forest method gains high values to identify the poisonous mushroom based on accuracy, precision, recall, and F-measure. A solution for identifying mushroom edibility proposed by Jahan Pinky et al. (2019) mentioned that the Random Forest method consumes the least amount of time out of the three methods - bagging (Naïve Bayes, dissimilarity) and Boosting (AdaBoost) are the other two.

Wang et al. (2020) used intuitive and appearance data, such as the shape and color of the mushroom caps, as features in their study. Three pattern recognition methods, namely logistic regression, support vector machine, and multigrained cascade forest, were employed to develop three distinct toxicity classifiers for mushrooms. The multigrained cascade forest classifier exhibited superior performance compared to the logistic regression and support vector machine classifiers, achieving an accuracy of approximately 98%.

An alternative approach to classifying the edibility of mushrooms was proposed by Gangu (2022) which generated more accurate results using a hybrid algorithm. They have used Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), and other classifiers. Tarawneh et al. (2022) suggested the same kind of approach for mushroom classification. Based on their study, the researchers found that the Decision Tree (DT), Artificial Neural Network (ANN), and Support Vector Machine (SVM) classifiers outperformed other classifiers in terms of accuracy. Motivated by this observation, Tarawneh et al. (2022) put forth a hybrid model that combines the top-performing classifiers to enhance classification accuracy. This hybrid approach aims to leverage the strengths of these individual classifiers and achieve improved overall performance in classification tasks. The conclusion from the study mentioned ‘Odor” as the most important feature of mushroom classification and mushrooms with any odor are poisonous.

Ismail et al. (2018) employed the Principal Component Analysis (PCA) algorithm to identify the optimal features for the classification experiment, utilizing the Decision Tree (DT) algorithm. They assessed the classification accuracy, coefficient metric, and time required to construct a classification model using a well-known Mushroom dataset. Through this analysis, they discovered that the 'odor' attribute exhibited the strongest influence on achieving a high classification accuracy. This finding also highlights the significance of the 'odor' feature in accurately classifying mushrooms.

The utilization of machine learning methods is not new to the field of mycology; however, this study goes beyond traditional approaches. The robust design of experiments leverages 3 crucial aspects in 11 different machine learning methods for a comprehensive analysis of the data. The systematic approach produced intriguing results compared to current literature reviews, particularly with the optimizable Naïve Bayes method.

The following sections of this paper will explain our entire process. The methodology section will showcase the methods chosen in descriptive analysis, feature selection and design of experiments. The results section shows the result analysis of descriptive analysis and MATLAB. The findings and additional considerations are discussed and explained to implement improvements in the field for further study to mushroom identification.

**2. Methodology**

*2.1 Dataset Description*

The Mushroom Data Set was obtained on Kaggle (UCI MACHINE LEARNING, 2016). The Data Set is a collection of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family. The dataset contains information about 8,123 mushrooms which are evenly split between edible (4,208) and poisonous (3,916) (48% poisonous cases). Ring number[[1]](#footnote-1) and Poisonous[[2]](#footnote-2) are integers, and the rest 21 variables are all categorical. Since Poisonous is the target, the ring number is the only numeric variable with data ranges in 0, 1, and 2 (*Figure 1*).

The dataset is a valuable resource for machine learning researchers who are interested in developing algorithms for classifying mushrooms. The dataset is well-curated and includes a variety of features that can be used to distinguish between edible and poisonous mushrooms. The attributes of the mushrooms are coded as follows.

A screenshot of a computer program

Description automatically generated with low confidence

*Figure 1. Attributes of Mushroom Dataset*

*2.2 Descriptive Analysis*

R programming script and RStudio (version 2023.03.0) were used to draw early findings. A high-level process can be found in *Figure 2*. First, all 22 independent variables have been transformed into 117 independent variables via dummy variables transformation. Second, independent variables are removed if they are either have low correlation (<10%) with target variable (Poisonous) or are highly correlated to each other (multicollinearity). 37 out of 117 remained after step 2. Third, Ridge Regression and Chi-Square take dimension reduction further to select the top 15 variables. Finally, 6 variables are selected based on significance after applying ANOVA and linear model.

A grey and white chart with black text

Description automatically generated

*Figure 2. High-level Process for Descriptive Analysis*

*Figure 3* outlines one example to walk-through. Spore\_point\_color, 1 of the 22 independent variables in the original database, was transformed into 9 independent variables via dummy transformation as spore\_point\_color\_black, spore\_point\_color\_brown and etc.*Figure 3* also visually displays the relationship between Poisonous and various dummy variables of spore\_point\_color. As the data indicates, mushrooms with spore\_point\_color in buff, orange, purple and yellow are 100% categorized as non-poisonous while all mushrooms with green spore\_point\_color are poisonous. Though it is helpful to convey early findings and trends via visualization, it is not scientific to take as a causational relationship. For example, a conclusion cannot be reached that all mushrooms in the world with green spore\_point\_color are poisonous.

A graph of different colors

Description automatically generated

*Figure 3. Poisonous vs Spore\_point\_color*

Multiple visualization approaches have been explored between each variable and target variables such as bar charts, scatterplots, and bar plots (details can be found in the descriptive analysis document). Dimension reduction was introduced and 80 out of 117 were removed due to low correlation (< 0.1) with target variables (*Figure 4*). Ridge regression was implemented to select the top 15 variables (*Figure 5*) that contribute most to the target variables based on coefficient values.

A picture containing diagram, plot, line, text

Description automatically generated

*Figure 4. IVs of Low Correlation with Target Variable*

A screenshot of a computer

Description automatically generated with medium confidence

*Figure 5. Ridge Regression Top Variable Names (R output)*

A correlation plot (*Figure 6*) was used to further investigate relationships between the top 15 variables and the target variable. Positive relationships (higher probability as poisonous mushrooms) and negative ones (higher probability as non-poisonous) are displayed as darker gray. Darker color represents stronger relationships such as odor\_none (-0.79), odor\_anise (-0.22), odor\_almond (-0.22), odor\_creosote (0.16), odor\_pungent (0.19), stalk\_root\_club (-0.22), ring\_type\_pendant (-0.54). Based on the correlation plot, the characteristic **mostly correlated to a poisonous mushroom** is **pungent odor** while the characteristic with highest probabilities indicating a **non-poisonous** mushroom are mushrooms with **no odor** and **pendant rings**.

*A close-up of a graph

Description automatically generated with low confidence**Figure 6. Correlation between the top variables and the target*

Data partition was set with 80% of training data. Logistic models (*Figure 7*) and ANOVA (*Figure 8*) were used to determine significant factors among the top list respectively and most variables including **gill\_color\_green, odor\_none, stalk\_color\_below\_ring\_yellow, odor\_anise and odor\_almond** show up in both approaches.

A screenshot of a computer

Description automatically generated with medium confidence

Figure *7*. LM

A screenshot of a computer

Description automatically generated with low confidence

Figure *8*. ANOVA

2.3 Predictive Analytics

MATLAB was used to conduct the design of experiments to identify the right set of features using various Machine Learning (ML) methods. The results from the design of experiments are compared against each other using two main approaches – the visual approach including the use of 15 multi-variate charts and the metric approach including the Interquartile Range (IQR)/Median/Range values. A conclusion is drawn based on the insights from both approaches.

**3. Experimental setup and data collection**

The design of experiments (DOE) is set up in MATLAB and all tests are run with 12 cores, original cost and no PCA enabled. The methods considered in MATLAB are MRMR, Chi-squared test, ANOVA, KK, LM, CORRPLOT, Ridge Regression and Stepwise.

The DOE comprises 27 runs across 3 sets of selected features (No. of features ranging in 3, 6, 9), test size (10%, 15%, 20%), and a number of cross-validations (CV) (4, 6, 8) (*Table 1*). Cross-validation was chosen over holdout as the database is relatively small and therefore all the data should be utilized.

Each run of the 27 runs contains MATLAB results for 11 ML methods against 20 computing criteria. Optimizable ML methods are chosen when available to reduce the running complexity as the optimizable method chooses the algorithm hyperparameters with the best running results.

|  |  |  |  |
| --- | --- | --- | --- |
| **RUN #** | **NO OF CV** | **TEST SIZE %** | **NO OF FEATURES** |
| RUN 1 | 4 | 10 | 3 |
| RUN 2 | 4 | 10 | 6 |
| RUN 3 | 4 | 10 | 9 |
| RUN 4 | 4 | 15 | 3 |
| RUN 5 | 4 | 15 | 6 |
| RUN 6 | 4 | 15 | 9 |
| RUN 7 | 4 | 20 | 3 |
| RUN 8 | 4 | 20 | 6 |
| RUN 9 | 4 | 20 | 9 |
| RUN 10 | 6 | 10 | 3 |
| RUN 11 | 6 | 10 | 6 |
| RUN 12 | 6 | 10 | 9 |
| RUN 13 | 6 | 15 | 3 |
| RUN 14 | 6 | 15 | 6 |
| RUN 15 | 6 | 15 | 9 |
| RUN 16 | 6 | 20 | 3 |
| RUN 17 | 6 | 20 | 6 |
| RUN 18 | 6 | 20 | 9 |
| RUN 19 | 8 | 10 | 3 |
| RUN 20 | 8 | 10 | 6 |
| RUN 21 | 8 | 10 | 9 |
| RUN 22 | 8 | 15 | 3 |
| RUN 23 | 8 | 15 | 6 |
| RUN 24 | 8 | 15 | 9 |
| RUN 25 | 8 | 20 | 3 |
| RUN 26 | 8 | 20 | 6 |
| RUN 27 | 8 | 20 | 9 |

*Table 1 Design of Experiment Setup*

**11 ML Methods adopted** including the following:

* Optimizable Tree: decision tree algorithm with adjustable hyperparameters
* Binary GLM Logistic Regression: generalized linear model for binary outcomes using logistic regression.
* Optimizable Naïve Bayes: Naïve Bayes algorithm with tunable parameters
* Optimizable SVM: support vector machine with adjustable parameters
* Efficient Logistic Regression: logistic regression method optimized for computational efficiency
* Efficient Linear SVM: linear support vector machine algorithm streamlined for efficient computation
* Optimizable KNN: k-nearest neighbors’ algorithm with tunable parameters
* SVM Kernel: support vector machine algorithm using a kernel function for non-linear classification
* Logistic Regression Kernel: logistic regression model utilizing a kernel for complex, non-linear patterns
* Optimizable Ensemble: group of machine learning models whose parameters are tunable for optimization
* Optimizable Neural Network: artificial neural network with adjustable parameters to optimize learning

**20 Computing criteria** include the following:

* Accuracy %: Percentage of correct predictions over total predictions
* TPR %: True Positive Rate; percentage of actual positives correctly identified ​
* TNR %: True Negative Rate; percentage of actual negatives correctly identified ​
* PPV %​: Positive Predictive Value; percentage of predicted positives that are correct
* NPV %: Negative Predictive Value; percentage of predicted negatives that are correct ​
* FNR %: False Negative Rate; percentage of actual positives incorrectly identified as negative​
* FPR %: False Positive Rate; percentage of actual negatives incorrectly identified as positive ​
* FDR %: False Discovery Rate; percentage of predicted positives that are false ​
* FOR %: False Omission Rate; percentage of predicted negatives that are false ​
* PT %: Prevalence Threshold; cut-off below which NPV exceeds PPV ​
* CSI %: Critical Success Index; ratio of correct predictions over sum of predictions​
* Prevalence %: Proportion of actual positive cases in total population ​
* BA %: Balanced Accuracy; average of TPR and TNR ​
* F1 %: F1 Score; harmonic mean of precision (PPV) and recall (TPR) ​
* MCC %: Matthews Correlation Coefficient; measures quality of binary classification ​
* FM %: Fowlkes-Mallows index; geometric mean of precision and recall ​
* BM %: Bookmaker Informedness; sum of TPR and TNR minus 1
* MK %: Markedness; sum of PPV and NPV minus 1 ​
* F2 %: F2 Score; measure that weighs recall higher than precision ​
* F.5 %: ​F0.5 Score; measure that weighs precision higher than recall

**Weighted Feature Selection**:

Since MATLAB does not need dummy transformation to analyze classification problems, all 22 variables are run directly with no transformation. A scientific weighted average (*Table 2*) was adopted to identify the top features in MATLAB based on the following chart. A variable will be rated 1 if they show up as the top variable under the method from both MATLAB and R and 1.5 if the variable has achieved a more significant value than its peers. The ranking is sorted based on the sum of the values. **Odor, spore point color, and bruises are the top 3 features** selected based on the weighted approach. In this way, 3 selected features include odor, spore point color, bruises. 6 selected features include the first 3 as well as gill color, ring type, stalk surface above ring. 9 selected features include the first 6 as well as gill size, stalk color above ring and stalk color below ring. Odor is unquestionably the most important variable consistent across descriptive analysis and predictive analysis. The other variables, however, are not entirely the same from the descriptive analysis results.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Rank** | **Method** | **MRMR** | **Chi2** | **ANOVA** | **Krustal Wallis** | **Ridge Regression (top 15)** | **Corrplot (all)** | **MATLAB Stepwise (all)** | **R stepwise** | **Sum** |
| 1 | Odor | 1.5 | 1.5 |  | 1 | 1.5 | 1.5 | 1 | 1.5 | 9.5 |
| 2 | Spore point color | 1 | 1.5 | 1.5 | 1.5 |  |  | 1 | 1.5 | 8 |
| 3 | Bruises | 1 | 1.5 | 1.5 | 1.5 |  |  | 1 | 1.5 | 8 |
| 4 | Gill color | 1 | 1.5 | 1 |  | 1.5 | 1 |  | 1.5 | 7.5 |
| 5 | Ring type |  | 1.5 | 1 | 1 | 1 | 1.5 |  | 1.5 | 7.5 |
| 6 | Stalk surface above ring | 1.5 | 1.5 |  | 1 | 1 | 1 | 1 |  | 7 |
| 7 | Gill size | 1 | 1.5 | 1.5 | 1.5 |  |  |  | 1.5 | 7 |
| 8 | Stalk color above ring |  | 1.5 | 1 |  | 1.5 |  | 1 | 1.5 | 6.5 |
| 9 | Stalk color below ring |  | 1.5 | 1 | 1 | 1 | 1 |  |  | 5.5 |
| 10 | Stalk surface below ring | 1 | 1.5 |  |  |  |  | 1 | 1.5 | 5 |
| 11 | Stalk root |  |  |  |  | 1 | 1.5 | 1 | 1.5 | 5 |
| 12 | Population |  | 1.5 | 1 | 1 |  |  | 1 |  | 4.5 |
| 13 | Habitat |  |  |  |  | 1 | 1 | 1 | 1.5 | 4.5 |
| 14 | Gill spacing | 1 |  | 1 | 1 |  |  |  |  | 3 |
| 15 | Veil Color | 1.5 |  |  |  |  |  |  |  | 1.5 |
| 16 | Cap shape |  |  |  |  |  |  |  | 1.5 | 1.5 |
| 17 | Cap surface |  |  |  |  |  |  |  | 1.5 | 1.5 |
| 18 | Cap color |  |  |  |  |  |  |  | 1.5 | 1.5 |
| 19 | Stalk shape |  |  |  |  |  |  |  | 1.5 | 1.5 |
| 20 | Ring number |  |  |  |  |  |  |  | 1.5 | 1.5 |

*Table 2 Weighted Feature Selection*

*Table 3* shows a DOE example of RUN 1 for 4 CV, 10% test size and 3 selected features.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RUN 1​** | **FEATURES​** | | | **MATLAB (TESTING CONFUSION MATRIX)​** | | | | | | **COMPUTED CRITERIA​** | | | | | | | | | | | | | | | | | | | |
| **Method**​ | **# of Cross Validation**​ | **Test Data Size ()**​ | **# of Features**​ | **Time (sec)**​ | **Hyperparameter/min classification error**​ | **TP (Top Left)**​ | **TN (Bot Right)**​ | **FP (Bot Left)**​ | **FN (Top Right)**​ | **Accuracy %**​ | **TPR  %**​ | **TNR %**​ | **PPV  %**​ | **NPV %**​ | **FNR %**​ | **FPR %**​ | **FDR %**​ | **FOR %**​ | **PT %**​ | **CSI %**​ | **Prevalence %**​ | **BA %**​ | **F1 %**​ | **MCC %**​ | **FM %**​ | **BM %**​ | **MK %**​ | **F2 %**​ | **F.5  %**​ |
| Optimizable Tree​ | 4​ | 10​ | 3​ | 17.898​ | Ginni/0.0054703​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Binary GLM Logisitic Regression​ | 4​ | 10​ | 3​ | 6.2386​ | ​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Optimizable Naïve Bayes​ | 4​ | 10​ | 3​ | 17.084​ | Gaussian/0.020512​ | 418​ | 376​ | 15​ | 3​ | 97.78​ | 99.29​ | 96.16​ | 96.54​ | 99.21​ | 0.71​ | 3.84​ | 3.46​ | 0.79​ | 16.43​ | 95.87​ | 97.78​ | 97.73​ | 97.89​ | 95.60​ | 97.90​ | 95.45​ | 95.74​ | 98.72​ | 97.07​ |
| Optimizable SVM​ | 4​ | 10​ | 3​ | 45.89​ | Cubic/0.0054703​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Efficient Logistic Regression​ | 4​ | 10​ | 3​ | 7.7652​ | ​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Efficient Linear SVM​ | 4​ | 10​ | 3​ | 6.9636​ | ​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Optimizable KNN​ | 4​ | 10​ | 3​ | 56.463​ | City block/0.0054703​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| SVM Kernel​ | 4​ | 10​ | 3​ | 11.548​ | ​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Logistic Regression Kernel​ | 4​ | 10​ | 3​ | 4.9913​ | ​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Optimizable Ensemble​ | 4​ | 10​ | 3​ | 148.4​ | Bag/494/3018/0.0054703​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |
| Optimizable Neural Network​ | 4​ | 10​ | 3​ | 172.16​ | 1/Than/45/0.0054703​ | 421​ | 383​ | 8​ | 0​ | 99.01​ | 100.00​ | 97.95​ | 98.14​ | 100.00​ | 0.00​ | 2.05​ | 1.86​ | 0.00​ | 12.51​ | 98.14​ | 99.01​ | 98.98​ | 99.06​ | 98.04​ | 99.06​ | 97.95​ | 98.14​ | 99.62​ | 98.50​ |

*Table 3 DOE RUN 1 Example*

**4. Results**

*4.1 Descriptive Analyses*

Comparing the results among approaches in Descriptive Analysis (*Table 4*), **odor\_none** is selected among all approaches. Others agreed among most approaches are **gill\_color\_green, stalk\_color\_below\_ring\_yellow, odor\_anise and odor\_almond**.

|  |  |  |  |
| --- | --- | --- | --- |
| **Ridge Regression** | **Correlation Plot** | **LM\*\*\*** | **ANOVA\*\*\*** |
| Gill\_color\_green | Stalk\_root\_club | Gill\_color\_green | Gill\_color\_green |
| **Odor\_none** | **Odor\_none** | **Odor\_none** | **Odor\_none** |
| Stalk\_color\_below\_ring\_yellow | Ring\_type\_pendant | Stalk\_color\_below\_ring\_yellow | Stalk\_color\_ below \_ring\_yellow |
| Stalk\_color\_above\_ring\_yellow | Odor\_anise | Odor\_anise | Odor\_anise |
| Stalk\_surface\_above\_ring\_scaly | Odor\_almond | Odor\_almond | Odor\_almond |

*Table 4 Results Comparison in Descriptive Analysis*

*4.2 Predictive Analyses – Quantitative outputs*

After running the design of experiment (DOE), plot, IQR, median and range were used to analyze **5 key metrics** out of 20 total metrics including **training time or CPU usage, accuracy (%), False Negative Rate (%), F2 (%) and BA (%).** The 5 key metrics are selected because they place higher cost on false negative detection. The cost of misclassifying false negative is high as it may increase mushroom poison cases.

The comparison summary of median (*Table 5*, *Table 6*, *Table 7*, *Table 8 and Table 9*) and IQR (*Table 10, Table 11, Table 12*, *Table 13 and Table 14*) of all tests. Lighter gray indicates the best performance and darker gray indicates the worst performance.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Median** | **Training Time (Sec)** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 12.12 | 21.34 | 19.71 |  |
| Binary GLM Logisitic Regression | 3.44 | 8.44 | 7.48 |  |
| Optimizable Naïve Bayes | 13.61 | 18.38 | 25.03 |  |
| Optimizable SVM | 51.74 | 74.18 | 88.99 |  |
| Efficient Logistic Regression | 3.63 | 4.72 | 5.79 |  |
| Efficient Linear SVM | 4.75 | 5.22 | 5.65 |  |
| Optimizable KNN | 30.16 | 37.62 | 55.94 |  |
| SVM Kernel | 22.29 | 39.92 | 48.52 |  |
| Logistic Regression Kernel | 6.56 | 10.16 | 12.73 |  |
| Optimizable Ensemble | 76.30 | 129.07 | 201.66 |  |
| Optimizable Neural Network | 172.16 | 365.04 | 271.46 |  |
| **Median1** (calculated excluding ONB) | **13.61** | **21.34** | **25.03** |  |
| max | 172.16 | 365.04 | 271.46 |  |
| min | 3.4431 | 4.7204 | 5.6518 |  |
| difference | 168.72 | 360.32 | 265.81 |  |

*Table 5 Median Values for All Criteria Across 27 Runs – Training Time*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Median** | **Accuracy %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 100.00 | 99.88 | 99.88 |  |
| Binary GLM Logisitic Regression | 100.00 | 99.88 | 99.88 |  |
| Optimizable Naïve Bayes | 96.80 | 96.55 | 96.09 |  |
| Optimizable SVM | 100.00 | 99.88 | 99.88 |  |
| Efficient Logistic Regression | 99.75 | 99.82 | 99.88 |  |
| Efficient Linear SVM | 100.00 | 99.84 | 99.88 |  |
| Optimizable KNN | 100.00 | 99.88 | 99.94 |  |
| SVM Kernel | 99.75 | 99.82 | 99.88 |  |
| Logistic Regression Kernel | 99.75 | 99.82 | 99.88 |  |
| Optimizable Ensemble | 100.00 | 99.88 | 99.88 |  |
| Optimizable Neural Network | 99.88 | 99.88 | 99.88 |  |
| **Median1** | **100.00** | **99.88** | **99.88** |  |
| Max (calculated excluding ONB) | 100.00 | 99.88 | 99.94 |  |
| Min (calculated excluding ONB) | 99.75 | 99.82 | 99.88 |  |
| difference | 0.25 | 0.06 | 0.06 |  |

*Table 6 Median Values for All Criteria Across 27 Runs – Accuracy*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Median** | **FNR %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 0.00 | 0.00 | 0.00 |  |
| Binary GLM Logisitic Regression | 0.00 | 0.00 | 0.00 |  |
| Optimizable Naïve Bayes | 1.43 | 1.43 | 0.01 |  |
| Optimizable SVM | 0.00 | 0.00 | 0.00 |  |
| Efficient Logistic Regression | 0.00 | 0.00 | 0.00 |  |
| Efficient Linear SVM | 0.00 | 0.00 | 0.00 |  |
| Optimizable KNN | 0.00 | 0.00 | 0.00 |  |
| SVM Kernel | 0.00 | 0.00 | 0.00 |  |
| Logistic Regression Kernel | 0.00 | 0.00 | 0.00 |  |
| Optimizable Ensemble | 0.00 | 0.00 | 0.00 |  |
| Optimizable Neural Network | 0.00 | 0.00 | 0.00 |  |
| **Median1** | **0.00** | **0.00** | **0.00** |  |
| Max (calculated excluding ONB) | 0.00 | 0.00 | 0.00 |  |
| Min (calculated excluding ONB) | 0.00 | 0.00 | 0.00 |  |
| difference | 0.00 | 0.00 | 0.00 |  |

*Table 7 Median Values for All Criteria Across 27 Runs – FNR %*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Median** | **F2 %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 100.00 | 99.95 | 99.95 |  |
| Binary GLM Logistic Regression | 100.00 | 99.95 | 99.95 |  |
| Optimizable Naïve Bayes | 97.96 | 97.98 | 97.84 |  |
| Optimizable SVM | 100.00 | 99.95 | 99.95 |  |
| Efficient Logistic Regression | 99.91 | 99.93 | 99.95 |  |
| Efficient Linear SVM | 100.00 | 99.94 | 99.95 |  |
| Optimizable KNN | 100.00 | 99.95 | 99.98 |  |
| SVM Kernel | 99.91 | 99.93 | 99.95 |  |
| Logistic Regression Kernel | 99.91 | 99.93 | 99.95 |  |
| Optimizable Ensemble | 100.00 | 99.95 | 99.95 |  |
| Optimizable Neural Network | 99.95 | 99.95 | 99.95 |  |
| **Median1** (calculated excluding ONB) | **100.00** | **99.95** | **99.95** |  |
| Max | 100.00 | 99.95 | 99.98 |  |
| Min | 99.91 | 99.93 | 99.95 |  |
| Difference | 0.09 | 0.02 | 0.02 |  |

*Table 8 Median Values for All Criteria Across 27 Runs – F2 %*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Median** | **BA %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 100.00 | 99.87 | 99.87 |  |
| Binary GLM Logistic Regression | 100.00 | 99.87 | 99.87 |  |
| Optimizable Naïve Bayes | 96.72 | 96.47 | 95.98 |  |
| Optimizable SVM | 100.00 | 99.87 | 99.87 |  |
| Efficient Logistic Regression | 99.74 | 99.81 | 99.87 |  |
| Efficient Linear SVM | 100.00 | 99.83 | 99.87 |  |
| Optimizable KNN | 100.00 | 99.87 | 99.94 |  |
| SVM Kernel | 99.74 | 99.81 | 99.87 |  |
| Logistic Regression Kernel | 99.74 | 99.81 | 99.87 |  |
| Optimizable Ensemble | 100.00 | 99.87 | 99.87 |  |
| Optimizable Neural Network | 99.87 | 99.87 | 99.87 |  |
| **Median1** (calculated excluding ONB) | **100.00** | **99.87** | **99.87** |  |
| Max | 100.00 | 99.95 | 99.98 |  |
| Min | 99.91 | 99.93 | 99.95 |  |
| Difference | 0.09 | 0.02 | 0.02 |  |
|  |  |  |  |  |

*Table 9 Median Values for All Criteria Across 27 Runs – BA %*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IQR** | **Training Time (Sec)** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 3.19 | 4.08 | 4.91 |  |
| Binary GLM Logistic Regression | 1.89 | 3.17 | 5.06 |  |
| Optimizable Naïve Bayes | 4.77 | 2.74 | 5.96 |  |
| Optimizable SVM | 7.65 | 27.19 | 27.68 |  |
| Efficient Logistic Regression | 1.37 | 2.71 | 2.37 |  |
| Efficient Linear SVM | 1.86 | 3.01 | 4.80 |  |
| Optimizable KNN | 4.81 | 9.67 | 15.70 |  |
| SVM Kernel | 14.63 | 31.44 | 33.95 |  |
| Logistic Regression Kernel | 1.78 | 5.75 | 6.19 |  |
| Optimizable Ensemble | 28.86 | 64.19 | 125.67 |  |
| Optimizable Neural Network | 166.29 | 914.50 | 399.97 |  |
| **Average1** | **21.55** | **97.13** | **57.48** |  |
| Max (calculated excluding ONB) | 166.29 | 914.50 | 399.97 |  |
| Min (calculated excluding ONB) | 1.37 | 2.71 | 2.37 |  |
| Difference | 164.92 | 911.79 | 397.60 |  |

*Table 10 IQR Values for All Criteria Across 27 Runs – Training Time*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IQR** | **Accuracy %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 1% | 1% | 1% |  |
| Binary GLM Logistic Regression | 1% | 1% | 1% |  |
| Optimizable Naïve Bayes | 3% | 3% | 2% |  |
| Optimizable SVM | 1% | 1% | 1% |  |
| Efficient Logistic Regression | 0% | 1% | 0% |  |
| Efficient Linear SVM | 1% | 1% | 1% |  |
| Optimizable KNN | 1% | 1% | 1% |  |
| SVM Kernel | 0% | 0% | 0% |  |
| Logistic Regression Kernel | 0% | 0% | 0% |  |
| Optimizable Ensemble | 1% | 1% | 1% |  |
| Optimizable Neural Network | 1% | 1% | 1% |  |
| **Average1** | **1%** | **1%** | **1%** |  |
| Max (calculated excluding ONB) | 1% | 1% | 1% |  |
| Min (calculated excluding ONB) | 0% | 0% | 0% |  |
| Difference | 0% | 0% | 0% |  |

*Table 11 IQR Values for All Criteria Across 27 Runs – Accuracy*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IQR** | **FNR %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 0% | 0% | 0% |  |
| Binary GLM Logistic Regression | 0% | 0% | 0% |  |
| Optimizable Naïve Bayes | 1% | 1% | 1% |  |
| Optimizable SVM | 0% | 0% | 0% |  |
| Efficient Logistic Regression | 0% | 0% | 0% |  |
| Efficient Linear SVM | 0% | 0% | 0% |  |
| Optimizable KNN | 0% | 0% | 0% |  |
| SVM Kernel | 0% | 0% | 0% |  |
| Logistic Regression Kernel | 0% | 0% | 0% |  |
| Optimizable Ensemble | 0% | 0% | 0% |  |
| Optimizable Neural Network | 0% | 0% | 0% |  |
| **Average1** | **0%** | **0%** | **0%** |  |
| Max (calculated excluding ONB) | 0% | 0% | 0% |  |
| Min (calculated excluding ONB) | 0% | 0% | 0% |  |
| Difference | 0% | 0% | 0% |  |

*Table 12 IQR Values for All Criteria Across 27 Runs – FNR %*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IQR** | **F2 %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 0% | 0% | 0% |  |
| Binary GLM Logistic Regression | 0% | 0% | 0% |  |
| Optimizable Naïve Bayes | 2% | 2% | 1% |  |
| Optimizable SVM | 0% | 0% | 0% |  |
| Efficient Logistic Regression | 0% | 0% | 0% |  |
| Efficient Linear SVM | 0% | 0% | 0% |  |
| Optimizable KNN | 0% | 0% | 0% |  |
| SVM Kernel | 0% | 0% | 0% |  |
| Logistic Regression Kernel | 0% | 0% | 0% |  |
| Optimizable Ensemble | 0% | 0% | 0% |  |
| Optimizable Neural Network | 0% | 0% | 0% |  |
| **Average1** | **0%** | **0%** | **0%** |  |
| Max (calculated excluding ONB) | 0% | 0% | 0% |  |
| Min (calculated excluding ONB) | 0% | 0% | 0% |  |
| difference | 0% | 0% | 0% |  |

*Table 13 IQR Values for All Criteria Across 27 Runs – F2 %*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IQR** | **BA %** | | | **Sparklines** |
|  | Set1 | Set2 | Set3 |  |
| Optimizable Tree | 1% | 1% | 1% |  |
| Binary GLM Logistic Regression | 1% | 1% | 1% |  |
| Optimizable Naïve Bayes | 3% | 3% | 4% |  |
| Optimizable SVM | 1% | 1% | 1% |  |
| Efficient Logistic Regression | 1% | 1% | 0% |  |
| Efficient Linear SVM | 1% | 1% | 1% |  |
| Optimizable KNN | 1% | 1% | 1% |  |
| SVM Kernel | 1% | 1% | 0% |  |
| Logistic Regression Kernel | 1% | 1% | 0% |  |
| Optimizable Ensemble | 1% | 1% | 1% |  |
| Optimizable Neural Network | 1% | 1% | 1% |  |
| **Median1** (calculated excluding ONB) | **1%** | **1%** | **1%** |  |
| Max | 1% | 1% | 1% |  |
| Min | 1% | 1% | 0% |  |
| Difference | 0% | 0% | 0% |  |
|  |  |  |  |  |

*Table 14 IQR Values for All Criteria Across 27 Runs – BA %*

**The best feature set is Set 1.** In *Table 15*, Set 1 was the only set to achieve **100% median accuracy**. The parameters indicate the number of cross validations, the % of test size and the number of features selected. For example, 4/10/3 indicates that there are 4-fold cross-validation, test size is 10% of the data and 3 features are selected in Run 1.

The highest performing tests in this set are **Run 5 and 8** where **100% accuracy** was achieved in all methods except optimizable naïve bayes. Run 5 and 8 include the same 6 features. The difference is the test holdout percent where Run 5 is 15% and Run 8 is 20%. Run 8 has **lower total training time** when compared to Run 5. According to the Run 8 ensemble hyperparameters the best performance was with **Gentleboost** which is different than what found in the literature review which shows Adaboost. *Table 15* also includes **optimizable KNN** as a top performing method which is the best method overall on average across all tests and performance metrics.

Therefore, **Run 8 is the best test out of Set 1.** Run 8 has 4-fold cross-validation, 20% test holdout and 6 features (Odor, Spore point color, Bruises, Gill color, Ring type and Stalk surface above ring).

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Accuracy (%)** | | | | | | | | | | |
| **Parameters** | 4/10/3 | 4/10/6 | 4/10/9 | 4/15/3 | 4/15/6 | 4/15/9 | 4/20/3 | 4/20/6 | 4/20/9 |
| **Algorithms** | Run 1 | Run 2 | Run 3 | Run 4 | **Run 5MAX** | Run 6 | Run 7 | **Run 8MAX** | Run 9 |
| Optimizable Tree | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| Binary GLM Logistic Regression | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| Optimizable Naïve Bayes | 97.78 | 94.46 | 96.92 | 98.28 | 94.17 | 95.57 | 98.21 | 94.83 | 96.80 |
| Optimizable SVM | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| Efficient Logistic Regression | 99.01 | 99.75 | 99.88 | 99.34 | 100.00 | 99.75 | 99.45 | 100.00 | 99.94 |
| Efficient Linear SVM | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| Optimizable KNN | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| SVM Kernel | 99.01 | 99.75 | 99.88 | 99.34 | 100.00 | 99.75 | 99.45 | 100.00 | 99.94 |
| Logistic Regression Kernel | 99.01 | 99.75 | 99.88 | 99.34 | 100.00 | 99.7 | 99.45 | 100.00 | 99.94 |
| Optimizable Ensemble | 99.01 | 99.75 | 100.00 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.00 |
| Optimizable Neural Network | 99.01 | 99.75 | 99.88 | 99.34 | 100.00 | 100.00 | 99.45 | 100.00 | 100.0 |

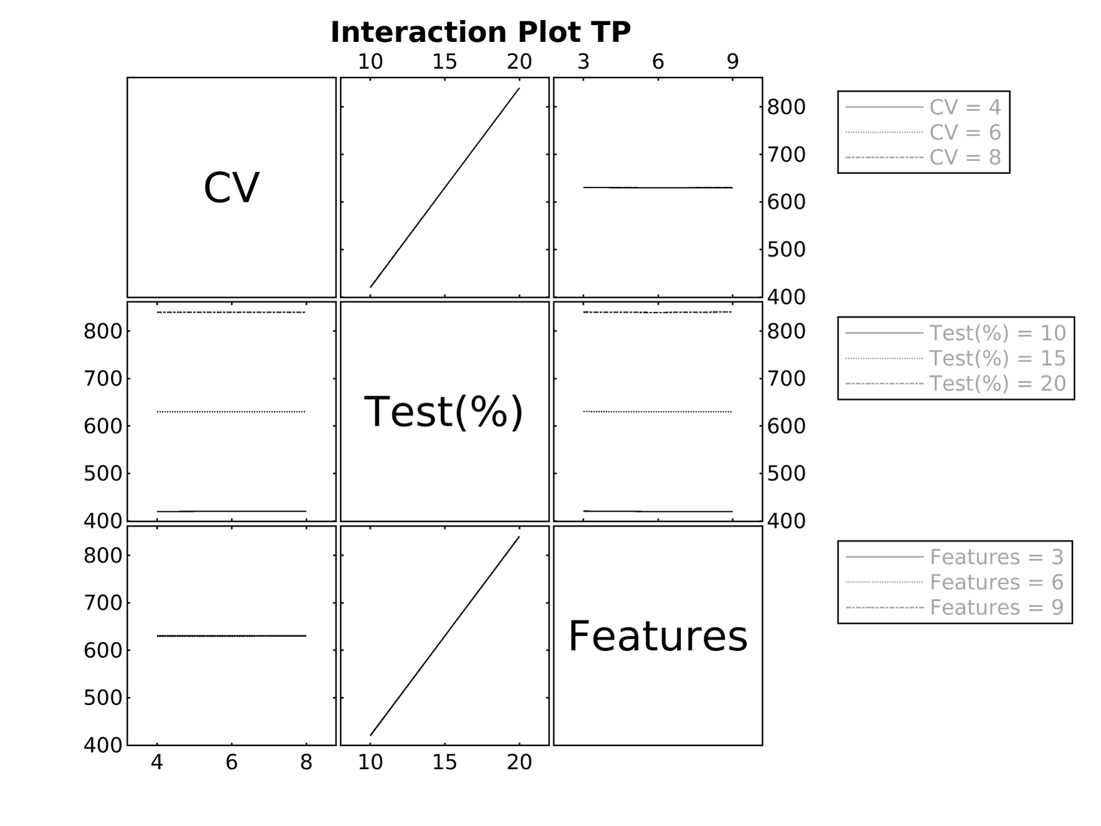
*Table 15. Comparison of Set 1 tests*

*4.3 Predictive Analyses – Visual outputs*

Another approach to compare run results is a visual approach. Based on CV, Test % and number of features 15 different charts were generated for True Positive (*Figure 9, Figure 10, Figure 11*), True Negative (*Figure 12, Figure 13, Figure 14*) , False Positive (*Figure 15, Figure 16, Figure 17*), False Negative (*Figure 18, Figure 19, Figure 20*), CPU Time (*Figure 21, Figure 22, Figure 23*).

Multivariate charts are essential for properly identifying relationships from the study’s DOE. In a quick concise format, averages in multiple dimensions can be analyzed. *Figure 9* shows high degree of correlation between number of CVs and number of features as well as a positive correlation between test size % and number of features. In *Figure 10*, number of CVs and features are very steady whereas test size % indicates a positive correlation with true positive observations. In *Figure 11*, there is a significant correlation across all dimensions regarding the average true positive observations. Similar synopses are observed in *Figure 12, Figure 13, Figure 14* for the true negative observations.

**True Positive (TP)**



*Figure 9 Interaction Plot -TP*

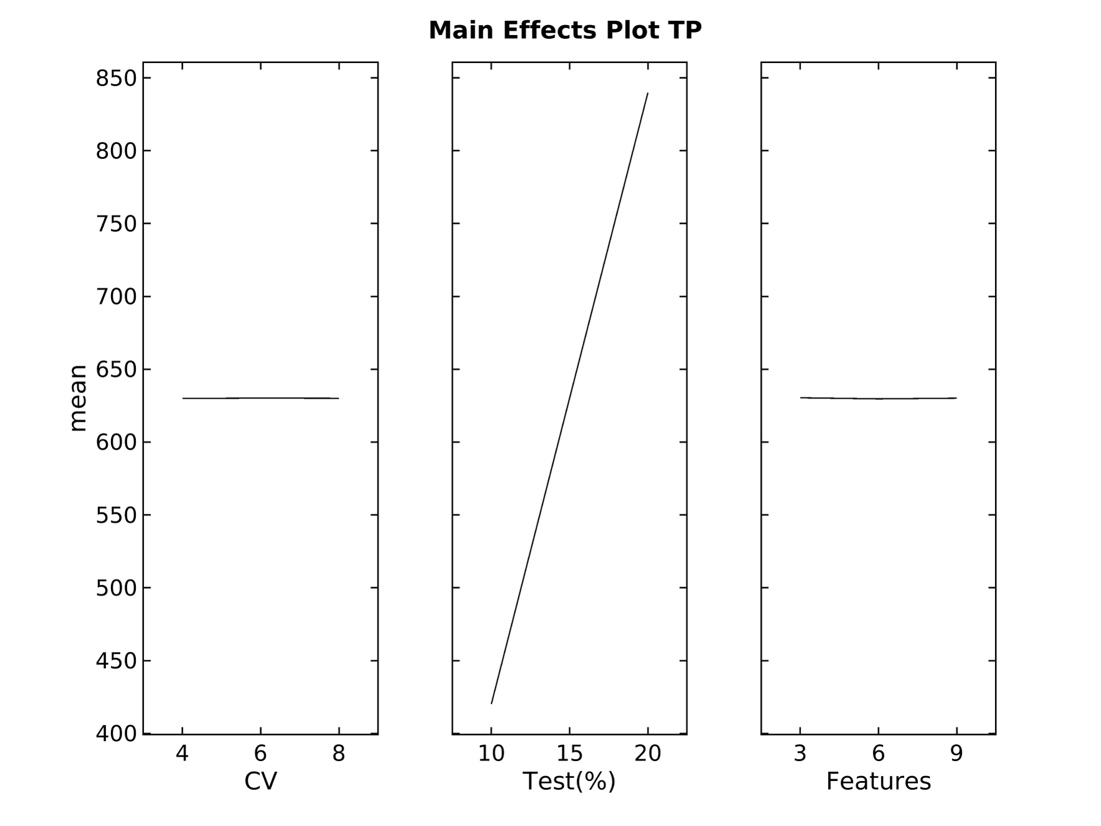


Figure 10 Main Effects Plot TP

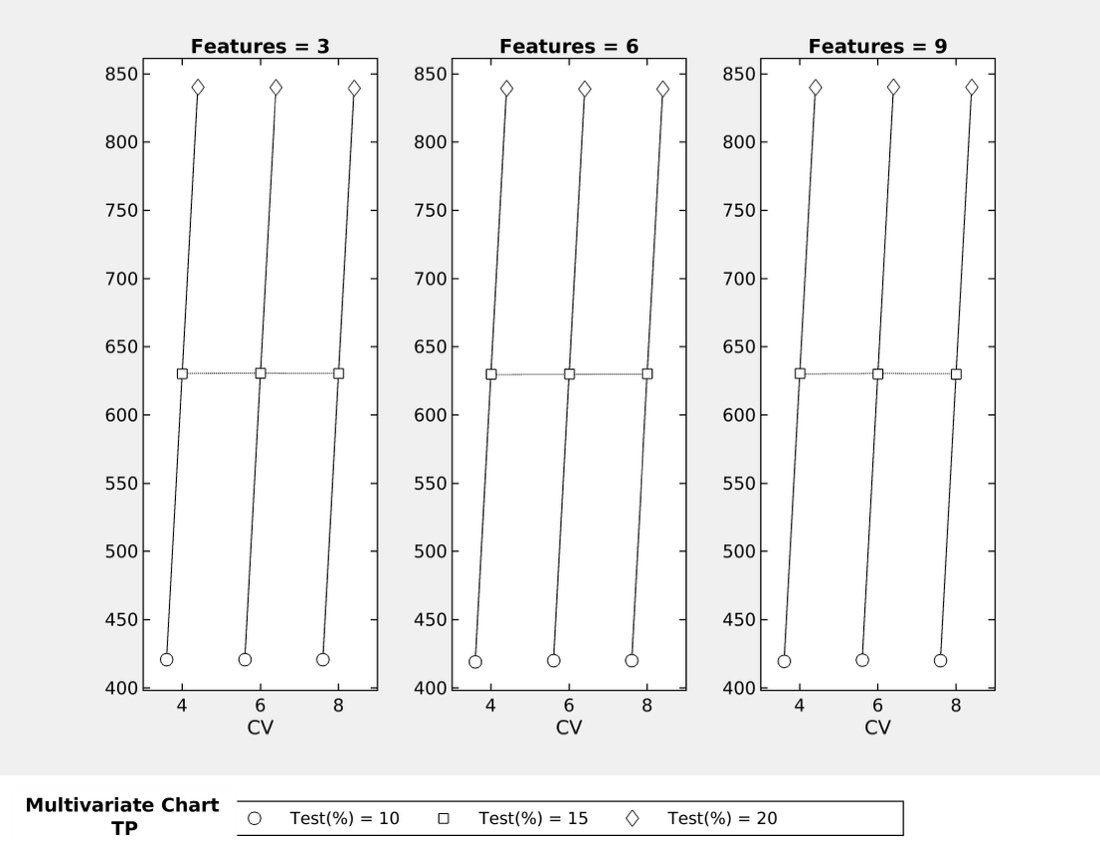


Figure 11 Multivariate Chart -TP

**True Negative (TN)**

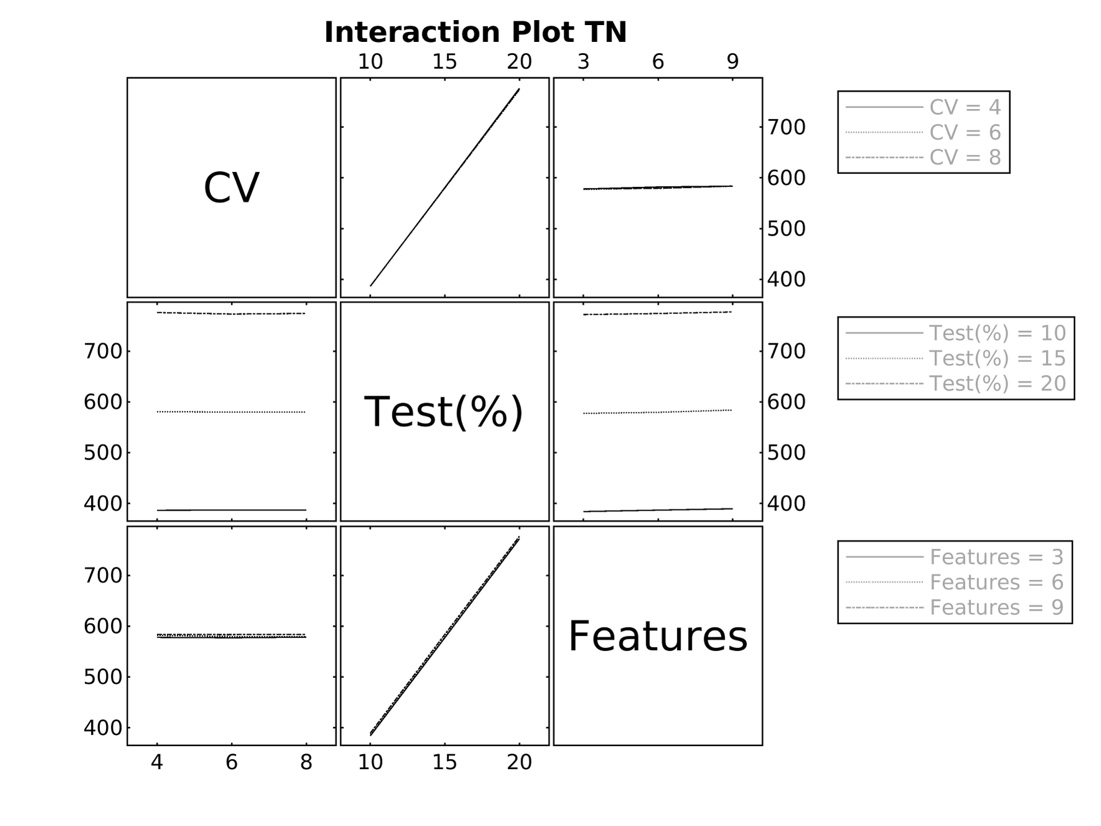


Figure 12 Main Effects Plot -TN

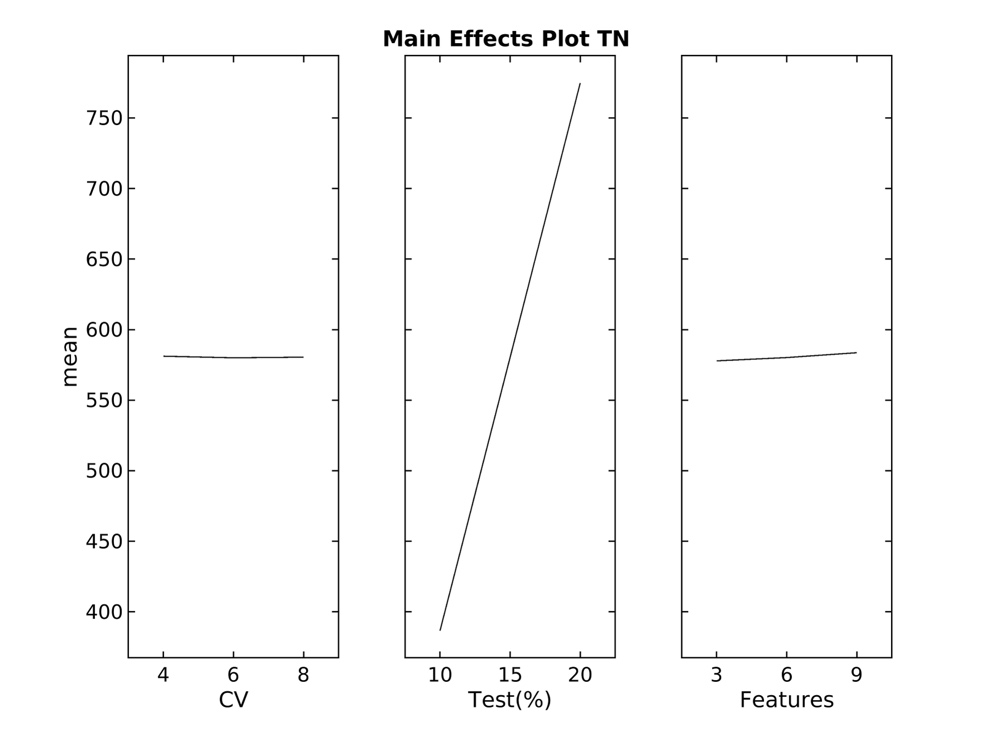


Figure 13 Interaction Plot -TN

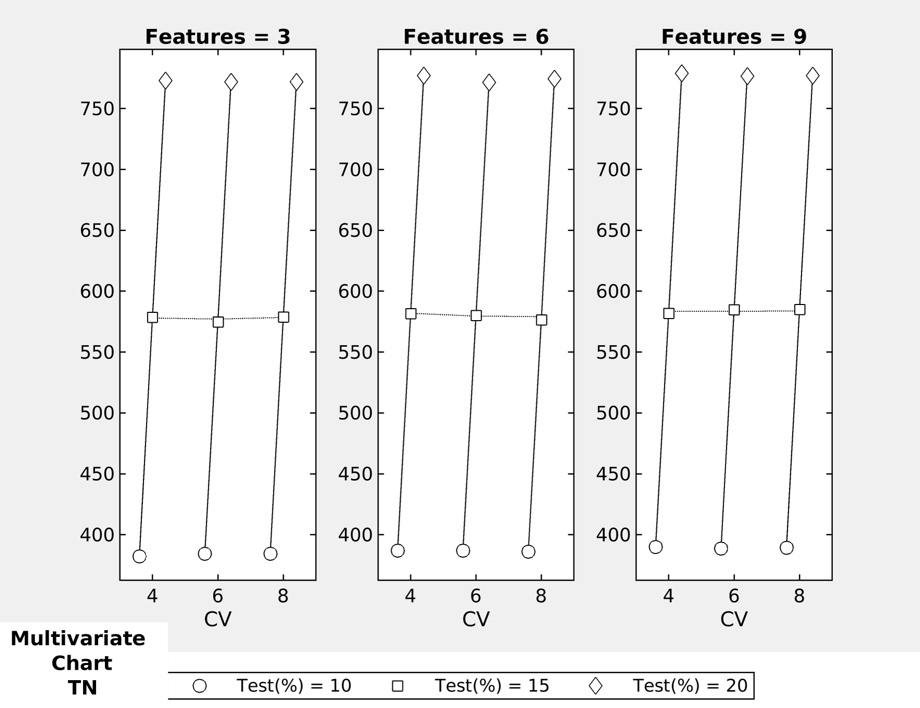


Figure 14 Multivariate Chart -TN

In *Figure 15*, correlation is still evident across test size % and number of features; however abnormal behavior is noted when considering **6 folds of CV**. *Figure 16* again emphasizes the 6 CV increased mean as well as a negative correlation between features and false positive observations. *Figure 17* indicates a notable trend: as the number of features increases, the number of false positive observations drop due to more accurate predictions.

**False Positive - (FP)**

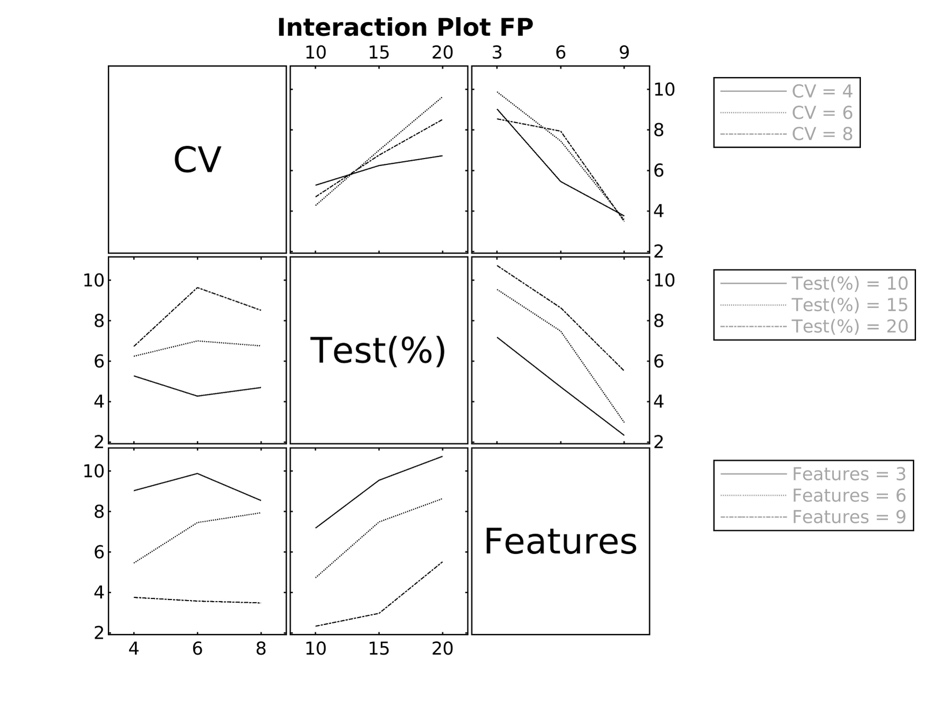


Figure 15 Interaction Plot -FP

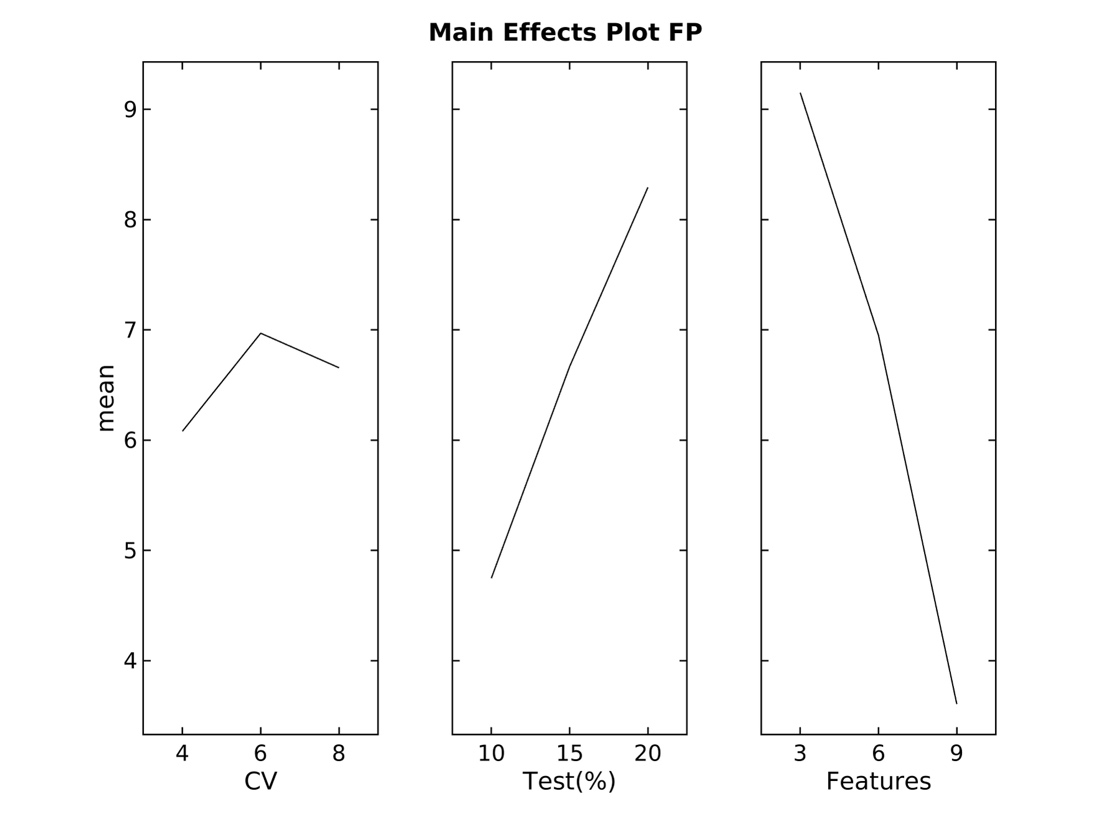


Figure 16 Main Effects Plot -FP

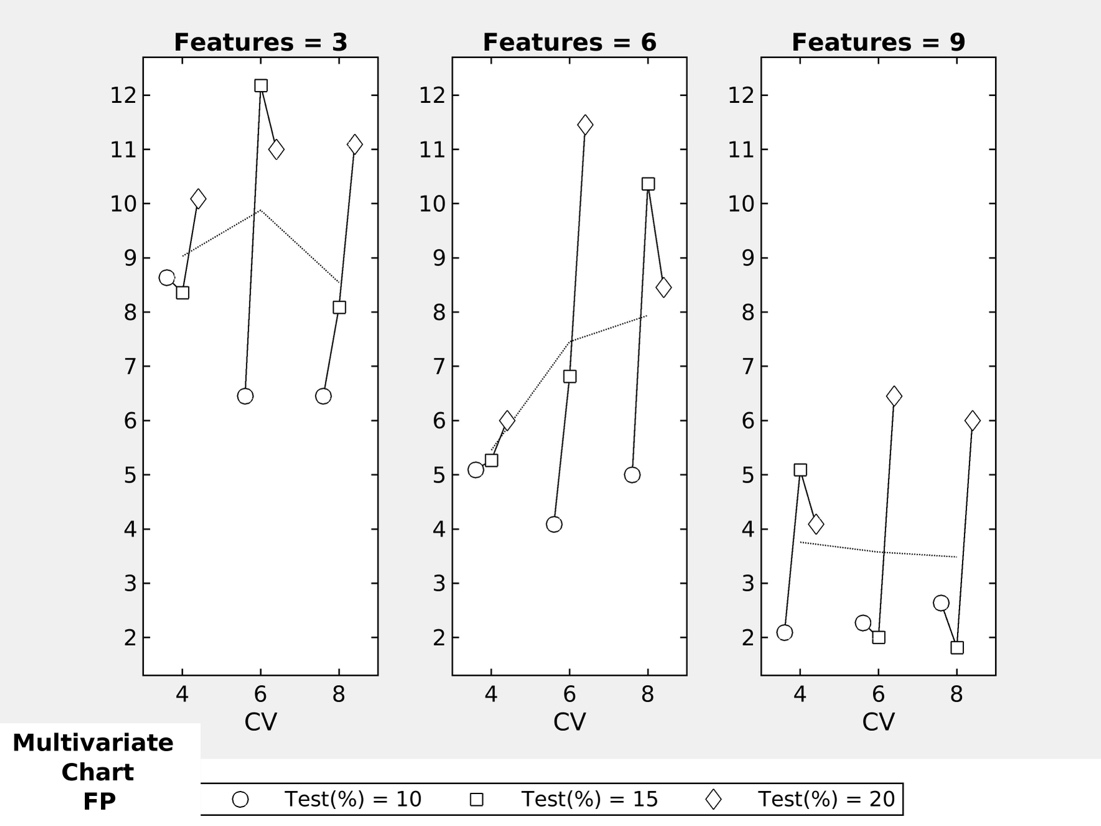


Figure 17 Multivariate Chart -FP

*Figure 18* shows irregular behavior with the interaction between 6 features and 20% test size regarding false negative observations. The abnormal effect of 6 features on false negative observations is evident in *Figure 19*. In *Figure 20* the volatility of 6 features and 20% test size is a stark contrast to the stability nine features brings to false negative observations.

**False negative- (FN)**

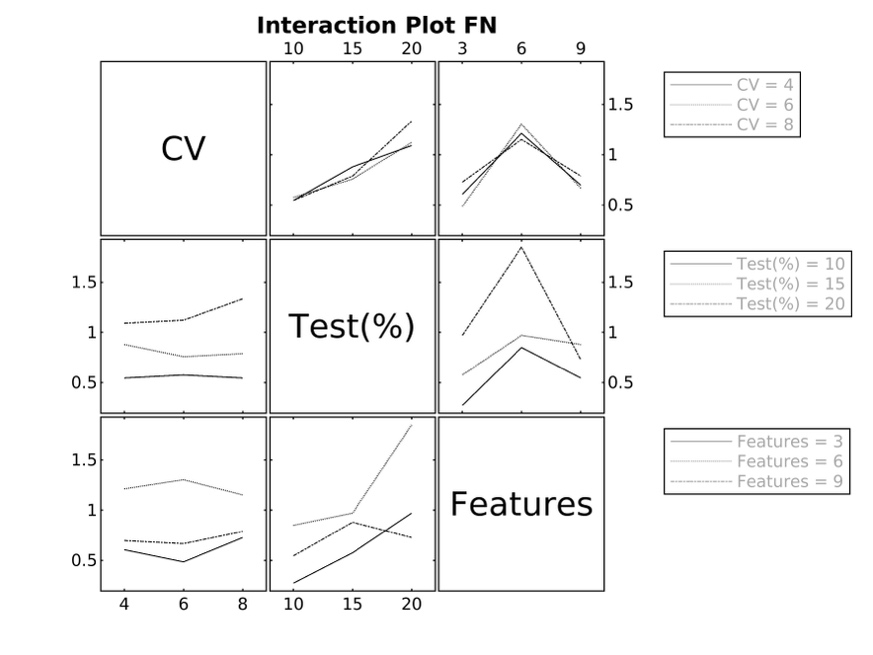


Figure 18 Interaction Plot -FN

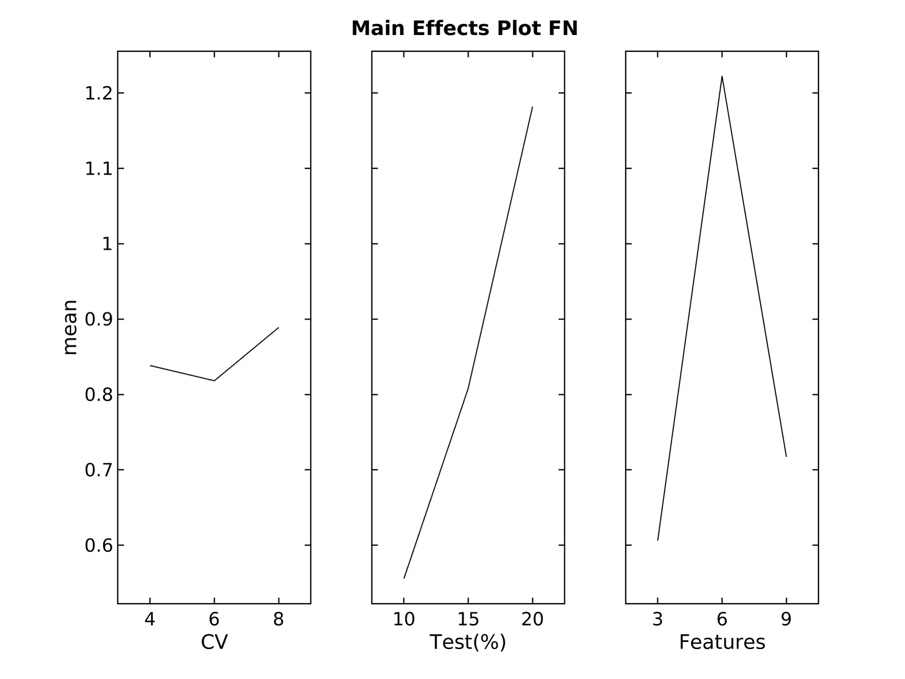


Figure 19 Main Effects Plot -FN

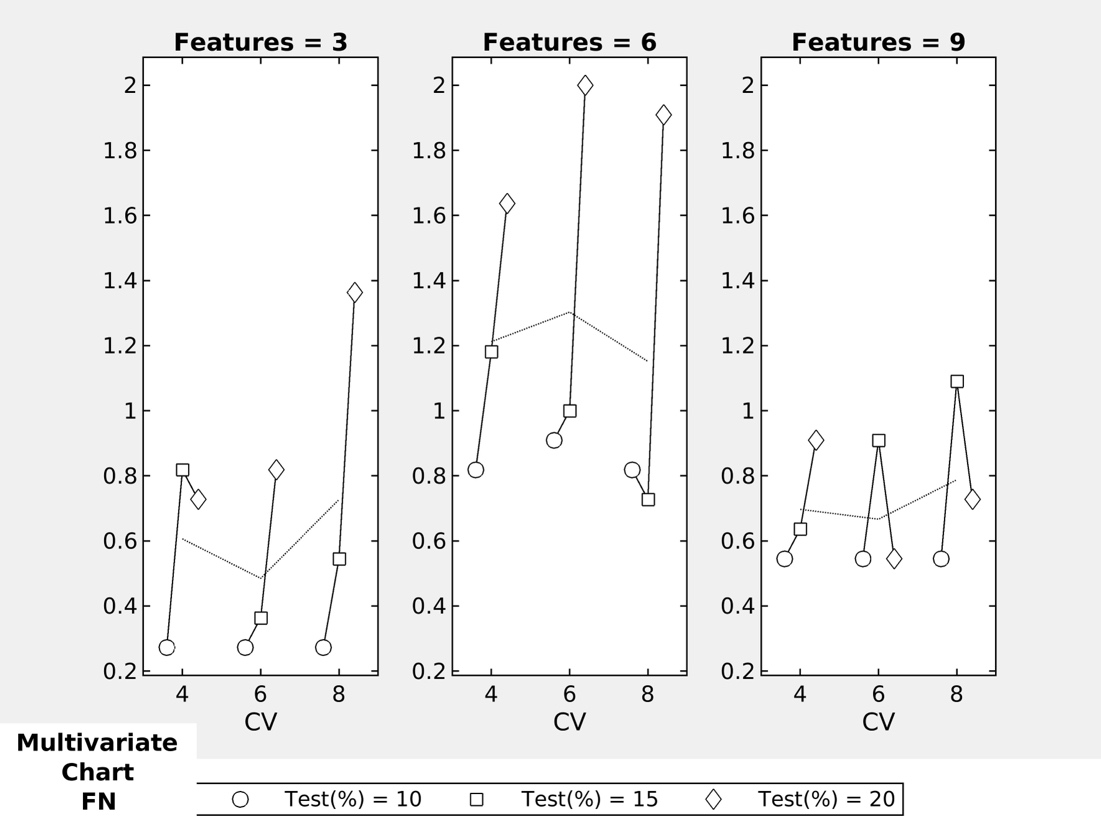


Figure 20 Multivariate Chart -FN

When looking at processing time, *Figure 21* and *Figure 22* illustrate the abnormal interactions with 6 features. *Figure 21* clearly indicates an outlier in the 20% test run; however consistent run times were not obtained during the study.

**CPU Time**

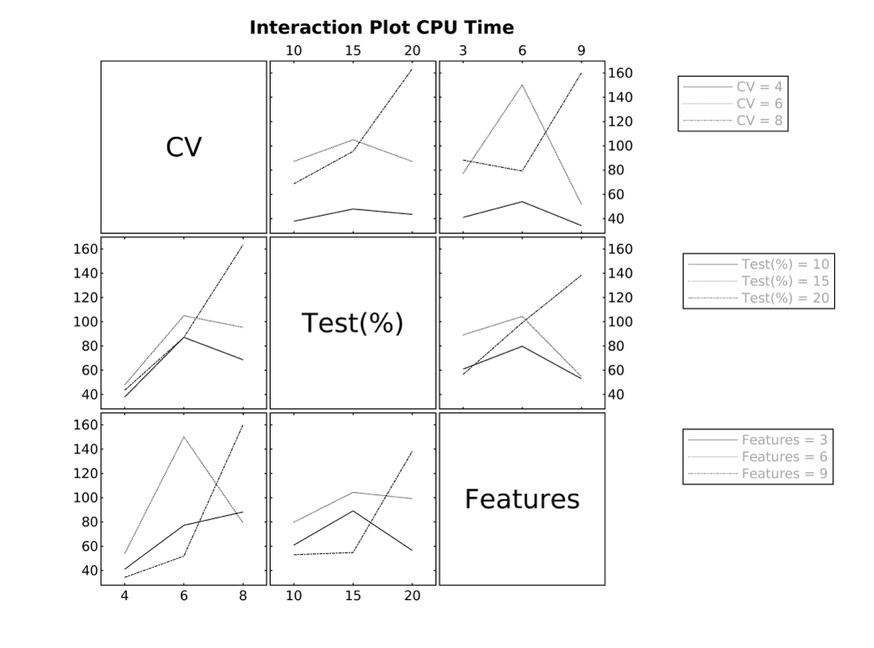


Figure 21 Interaction Plot -CPU Time

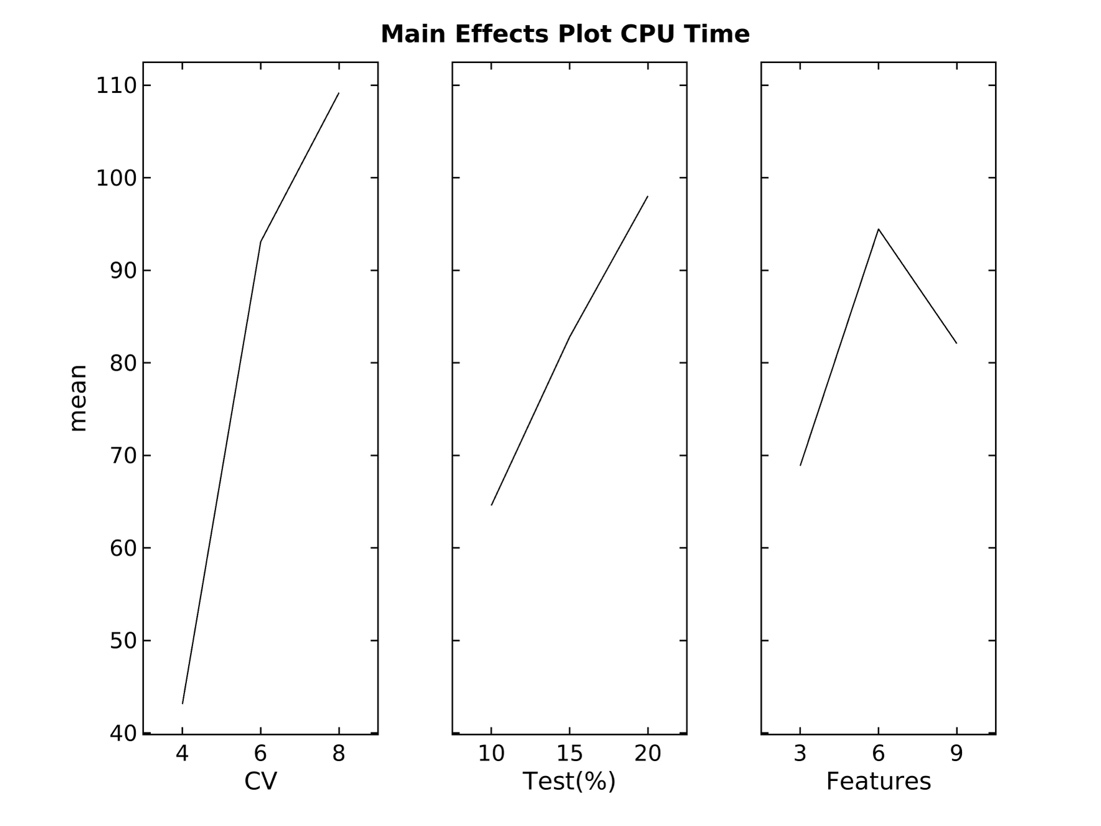


Figure 22 Main Effects Plot -CPU Time

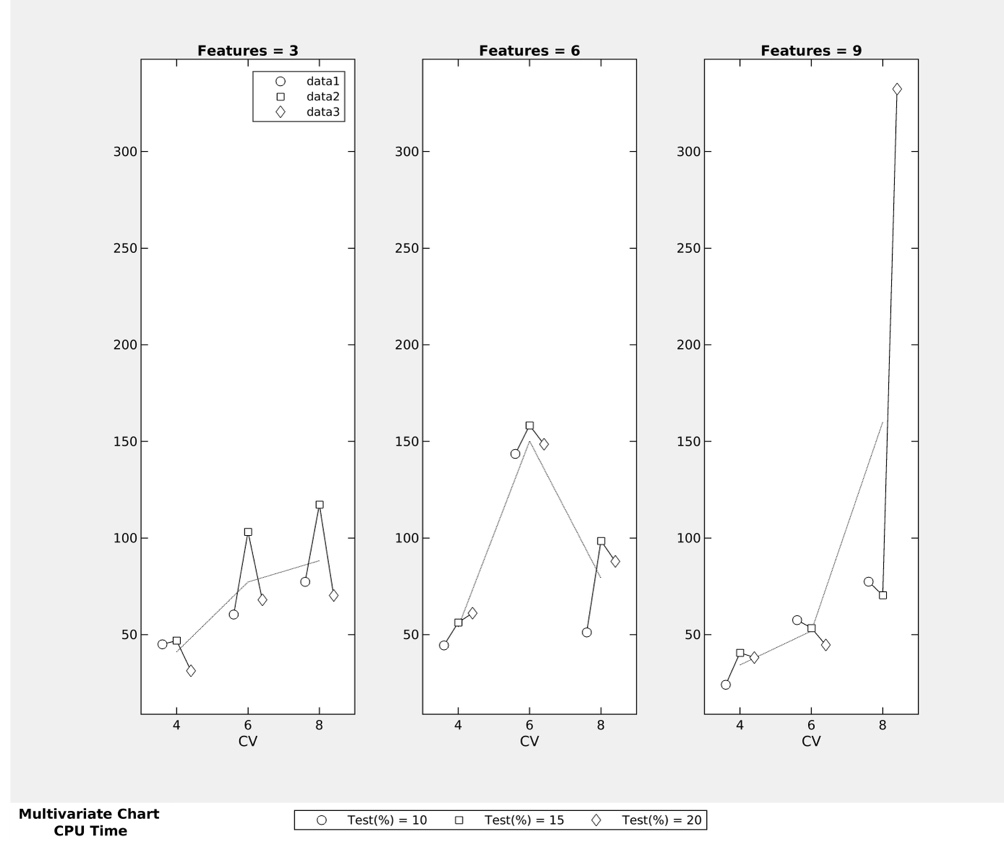


Figure 23 Multivariate Chart -CPU Time

**5.Discussion and Conclusion**

The findings of this study offer valuable insights into the power and limitations of 11 different ML methods in identifying patterns of the mushroom dataset. Notably, a consistency of below par performance was identified in the Naïve Bayes results in the robust design of experiments. The multivariate charts clearly illustrate irregular behavior with top 6 selected features. Even though the top nine features are all independent based on the infinite importance score from the chi-squared test, the Naïve Bayes method is identifying some correlation within that set of features.

Another observation from this study’s initial investigation indicates inconsistency/ potential negative impact from the bruise feature. Results inconsistency reduced after removing bruise from the 6-feature set but there were no conclusive findings. Additional research is required to determine the cause to the irregular behavior in the 6-feature set.

The study identified a set of well performing machine learning methods including Optimizable Trees, Binary GLM Logistic Regression, Optimizable SVM, Efficient Linear SVM, Optimizable KNN and Optimizable Ensemble. The Logistic Regression family outrun the other algorithms on the CPU time consistently across all sets. A key observation was in the achievement of zero false negative ratios for the well performing machine learning methods. The study also confirmed that odor as the most important variable in mushroom classification. The study proved that poisonous mushrooms could be accurately predicted by utilizing specific machine learning methods and common mushroom characteristics.

**Bibliography**

Broussard, C. (2001). Mushroom poisoning—from diarrhea to liver transplantation. *The American Journal of Gastroenterology*, *96*(11), 3195–3198. https://doi.org/10.1016/S0002-9270(01)03845-X

C. Ortega, J. H. J. (2020). Analysis of Performance of Classification Algorithms in Mushroom Poisonous Detection using Confusion Matrix Analysis. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(1.3), 451–456. https://doi.org/10.30534/ijatcse/2020/7191.32020

Chelliah, B. J., Kalaiarasi, S., Anand, A., Janakiram, G., Rathi, B. & Warrier, N. K. (2018). Classification of Mushrooms using Supervised Learning Models. 6(4), 229–232.

Chen, Z., Zhang, P., & Zhang, Z. (2014). Investigation and analysis of 102 mushroom poisoning cases in Southern China from 1994 to 2012. *Fungal Diversity*, *64*(1), 123–131. https://doi.org/10.1007/s13225-013-0260-7

Chowdhury, D. R. & Ojha, S. (2017). An Empirical Study on Mushroom Disease Diagnosis: A Data Mining Approach. International Research Journal of Engineering and Technology (IRJET), 4(01), 529–534.

Gangu, S. C. (2022). Edibility Detection of Mushroom Using Logistic Regression and Pca. International Journal of Advanced Research in Computer Science, 13(3), 30–34. https://doi.org/10.26483/ijarcs.v13i3.6830

Gonmori, K., & Yoshioka, N. (n.d.). Mushroom toxins. In *Drugs and Poisons in Humans* (pp. 469–480). Springer-Verlag. https://doi.org/10.1007/3-540-27579-7\_53

Govorushko, S., Rezaee, R., Dumanov, J. & Tsatsakis, A. (2019). Poisoning associated with the use of mushrooms: A review of the global pattern and main characteristics. Food and Chemical Toxicology, 128(April), 267–279. https://doi.org/10.1016/j.fct.2019.04.016

Halpern, G. M. (2010). Medicinal mushrooms. *Progress in Nutrition*, *12*(1), 29–36.

Holsen, D. S., & Aarebrot, S. (1997). [Poisonous mushrooms, mushroom poisons and mushroom poisoning. A review]. *Tidsskrift for Den Norske Laegeforening : Tidsskrift for Praktisk Medicin, Ny Raekke*, *117*(23), 3385–3388.

Ismail, S., Zainal, A. R. & Mustapha, A. (2018). Behavioural features for mushroom classification. ISCAIE 2018 - 2018 IEEE Symposium on Computer Applications and Industrial Electronics, 412–415. https://doi.org/10.1109/ISCAIE.2018.8405508

Jahan Pinky, N., Mohidul Islam, S. M. & Sharmin Alice, R. (2019). Edibility Detection of Mushroom Using Ensemble Methods. International Journal of Image, Graphics and Signal Processing, 11(4), 55–62. https://doi.org/10.5815/ijigsp.2019.04.05

Kieslichova, E., Frankova, S., Protus, M., Merta, D., Uchytilova, E., Fronek, J., & Sperl, J. (2018). Acute Liver Failure due to Amanita phalloides Poisoning: Therapeutic Approach and Outcome. *Transplantation Proceedings*, *50*(1), 192–197. https://doi.org/10.1016/j.transproceed.2017.11.032

Krakowiak, A., Piekarska-Wijatkowska, A., Kobza-Sindlewska, K., Rogaczewska, A., Politański, P., Hydzik, P., Szkolnicka, B., Kłopotowski, T., Picheta, S., Porębska, B., Antończyk, A., Waldman, W., Sein Anand, J., Matuszkiewicz, E., & Łukasik-Głębocka, M. (2017). Poisoning deaths in Poland: Types and frequencies reported in Łódź, Kraków, Sosnowiec, Gdańsk, Wrocław and Poznań during 2009–2013. *International Journal of Occupational Medicine and Environmental Health*. https://doi.org/10.13075/ijomeh.1896.01063

Paudel, N. & Bhatta, J. (2022). Mushroom Classification using Random Forest and REP Tree Classifiers. Nepal Journal of Mathematical Sciences, 3(1), 111–116. https://doi.org/10.3126/njmathsci.v3i1.44130

Roy Chowdhury, D., & Ojha, S. (2017). An Empirical Study on Mushroom Disease Diagnosis: A Data Mining Approach An Empirical Study on Mushroom Disease Diagnosis: A Data Mining Approach View project An Empirical Study on Mushroom Disease Diagnosis: A Data Mining Approach. In *International Research Journal of Engineering and Technology*. www.irjet.net

Ruiz-González, L. E., Vázquez-Zea, J. A., Vega-Villasante, F., Guzmán-Dávalos, L., & Guerrero-Galván, S. R. (2017). Evaluación del efecto tóxico de hongos Basidiomycota en la eclosión de quistes de Artemia franciscana. *Revista Iberoamericana de Micología*, *34*(4), 220–224. https://doi.org/10.1016/j.riam.2017.03.007

Schmutz, M., Carron, P.-N., Yersin, B., & Trueb, L. (2018). Mushroom poisoning: a retrospective study concerning 11-years of admissions in a Swiss Emergency Department. *Internal and Emergency Medicine*, *13*(1), 59–67. https://doi.org/10.1007/s11739-016-1585-5

Sisodiya, A. S., Kadicheeni, A. S., Das, G., Kumari, T., Archana, T. S. & Kumar, D. (2023). How to tell the mushroom you are holding will kill you or heal you: A mini review. 12(1), 109–122.

Soltaninejad, K. (2018). Outbreak of Mushroom Poisoning in Iran: April–May, 2018. *The International Journal of Occupational and Environmental Medicine*, *9*(3), 152–156. https://doi.org/10.15171/ijoem.2018.1380

Sönmez, B. M. , Y. F. , K. M. E. , D. T. , Ş. T. , C. H. , B. İ. , Y. M. (2015). Demographic and Clinical Characteristics of Mushroom Poisonings Presenting to Emergency Department. *Annals of Clinical and Analytical Medicine*, *06*(04). https://doi.org/10.4328/JCAM.2176

Sun, J., Li, H.-J., Zhang, H.-S., Zhang, Y.-Z., Xie, J.-W., Ma, P.-B., Guo, C., & Sun, C. (2018). Investigating and analyzing three cohorts of mushroom poisoning caused by *Amanita exitialis* in Yunnan, China. *Human & Experimental Toxicology*, *37*(7), 665–678. https://doi.org/10.1177/0960327117721960

Tarawneh, O., Tarawneh, M. & Sharrab, Y. (2022). Mushroom classification using machine-learning techniques. July.

Wang, X.-M., Zhang, J., Wu, L.-H., Zhao, Y.-L., Li, T., Li, J.-Q., Wang, Y.-Z., & Liu, H.-G. (2014). A mini-review of chemical composition and nutritional value of edible wild-grown mushroom from China. *Food Chemistry*, *151*, 279–285. https://doi.org/10.1016/j.foodchem.2013.11.062

Wang, Y., Du, J., Zhang, H. & Yang, X. (2020). Mushroom Toxicity Recognition Based on Multigrained Cascade Forest. Scientific Programming, 2020. https://doi.org/10.1155/2020/8849011

Zheng, Q., Wang, L., He, J., & Li, T. (2023). KNN-Based Consensus Algorithm for Better Service Level Agreement in Blockchain as a Service (BaaS) Systems. *Electronics (Switzerland)*, *12*(6), 1–21. https://doi.org/10.3390/electronics12061429

1. Ring\_number- a variable of the dataset (Figure 1) [↑](#footnote-ref-1)
2. Poisonous – a variable of the dataset (Figure 1) [↑](#footnote-ref-2)