Estefan Gonzales,
Brian McCollum,
Cyrus McCormick
02/19/2023
CS49 - Random Forests

Machine Learning Project 1:
Random Forests for Mushroom Classification

I. Introduction

Random forests is a machine learning method useful for performing classification tasks. Through using random forests for classification, we can limit potential issues which are commonly encountered with decision trees. Random forests create multiple decision trees with different subsets of training data and features which reduces the likelihood of overfitting when compared to performing classification with a single tree. Random forests are also more robust as they are less sensitive to outliers and missing data.

Our goal is to utilize random forests in the classification of mushrooms, where a mushroom in our data set is either edible or poisonous. The dataset we used for training our random forest is composed of 7,126 data samples which are described by a set of 23 attributes surrounding the mushroom's characteristics.

II. Method

A. Building Random Forest

Our random forest is constructed with a set of decision trees which are built using an implementation of the ID3 algorithm. Our random forest is built utilizing two sources of randomness. To ensure that there is some variation in the training sets used for our decision trees, we use a bagging algorithm which builds training sets through samples with replacement. The training set was split into an 80/20 split for training and validation.

By bootstrapping samples to train our decision trees on, we can ensure that there is variation between each of the trees in our random forest. Variation in our training data reduces the correlation found between our decision trees by helping us prevent overfitting to the training data.

B. Information Gain Criteria

When constructing the decision trees for our random forest, we specify which criteria should be used for defining information gain. The three methods we used to determine information gain for our decision trees includes entropy, gini index, and misclassification error. We compare the classification accuracy of each of these decision trees

Entropy

$$\sum_{i-1}^{m} - p_i log_2 p_i$$

Entropy is used to measure the degree of randomness which exists within our dataset where p_i is the proportion of data samples in class i. In regards to information gain within the context of our decision trees, entropy is used for measuring the level of impurity found for each attribute in our tree. Our goal when forming decision trees with entropy is to reduce the amount of impurity found at each split by minimizing the amount of entropy.

Gini Index

$$1 \ - \sum_{i-1}^{m} p_i^{\ 2}$$

Similar to entropy, gini index allows us to measure the amount of impurity found within our data set where low gini index values represent a more pure data set. Gini index is also minimized when many data samples at a specific node belong to the same class.

Misclassification error

$$1 \ - \ max_k p_k$$

Misclassification error measures impurity through the proportion of misclassified data samples and attempts to maximize the above value. If class distribution is skewed or unbalanced, misclassification error may not accurately reflect the true impurity of the node as it’s sensitive to changes in class probabilities.

C. *Chi-Square*

For this project our implementation of the Chi statistic termination rule works as a means to eliminate insignificant attributes. When constructing the tree we test each attribute and its significance relative to its classification. After testing each attribute we verify whether it qualifies as significant based on our value of alpha, if insignificant we remove the attribute from our set. With this implementation we found that the attributes that would be excluded rarely ended up impacting our tree. For this reason we found the Chi Square test-statistics did not have a significant impact on our trees or their accuracy apart from anomalous behavior.

III. Results

Red nodes represent poisonous classification
Green nodes represent edible classification
Blue nodes represent a node with children (Contains attribute name as well as gini rating)
Edges are labeled with attribute values

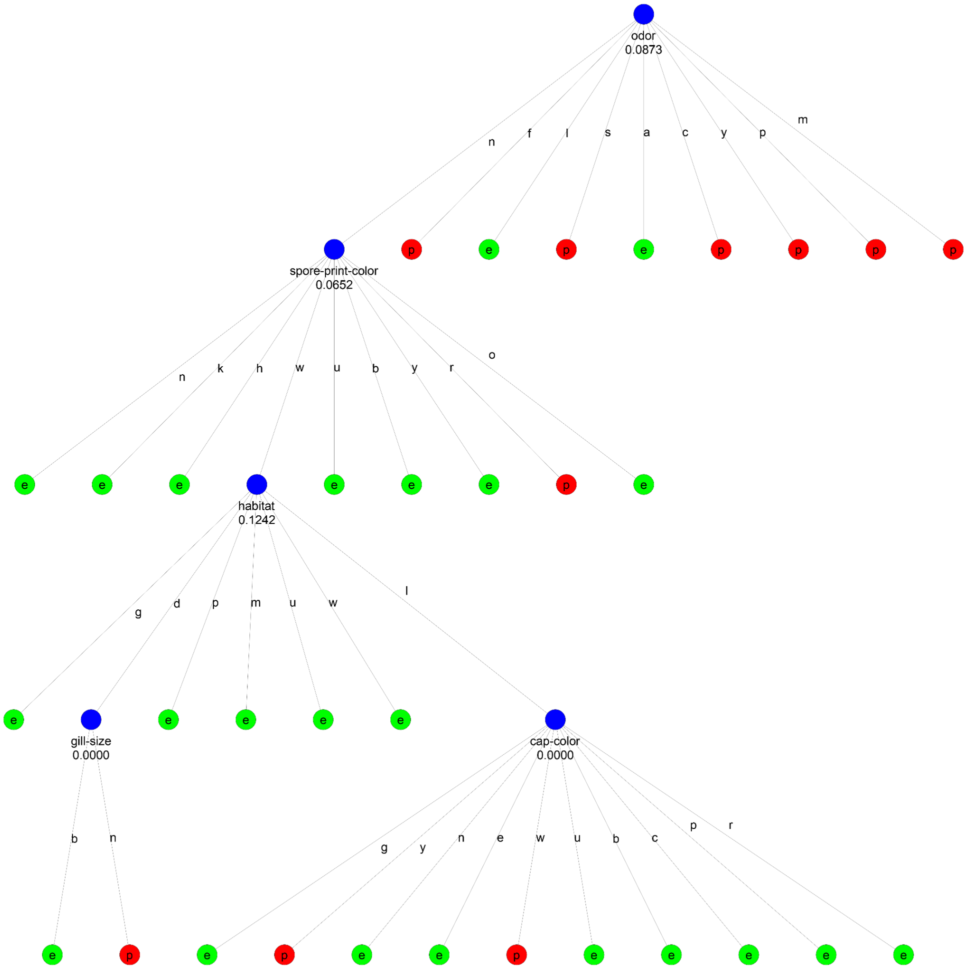


Figure 1. Visualization of a decision tree created with entropy

A. Chi-square results for different significance levels

When testing our decision trees with a set of alpha values for different information gain criteria, we found that our decision trees performed equally well when training with a large data set. If we decrease the size of our training data set significantly, we find that there is a small amount of variance between the accuracy levels for our decision trees.

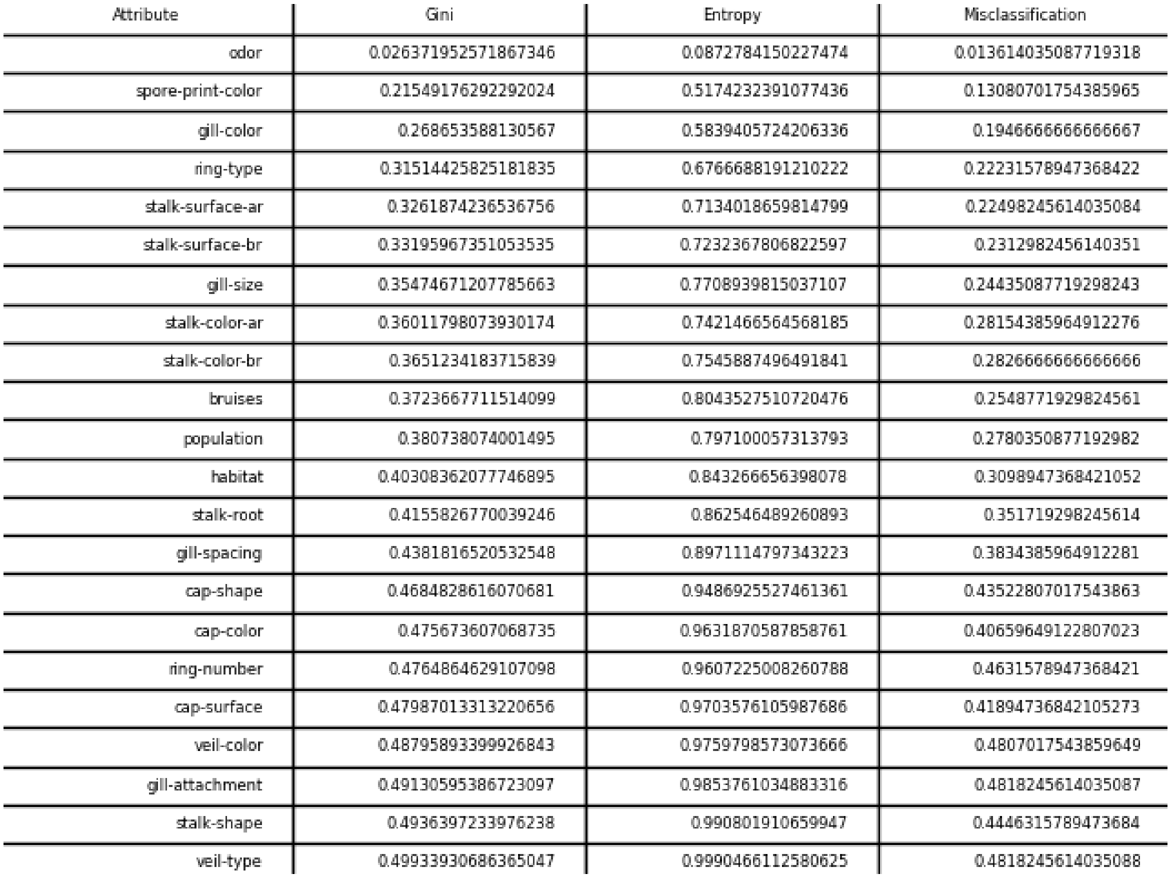
Criteria	**Significance level (α)**	**Accuracy level**
Entropy	0.00	99.9%
	0.05	99.9%
	0.95	99.9%
	0.99	99.9%
Gini Index	0.00	99.8%
	0.05	99.8%
	0.95	99.8%
	0.99	99.8%
Misclassification Error	0.00	100%
	0.05	100%
	0.95	100%
	0.99	100%

Figure 2. Comparison of decision tree accuracy by α value

Despite the small variance found when working with a small training set, all of our decision trees performed equally well with 100% accuracy levels when training with the entirety of the provided training set.

B. Split Criteria Performance

When evaluating the validity of attributes for splitting our tree we found uniformity in the highest performing split attributes. When analyzing the trees we find that odor allows for a large portion of our training examples to be directly classified. This is due to high homogeneity for the majority of the values that the odor attribute can take.



Attribute	Gini	Entropy	Misclassification
odor	0.026371952571867346	0.0872784150227474	0.013614035087719318
spore-print-color	0.21549176292292024	0.5174232391077436	0.13080701754385965
gill-color	0.268653588130567	0.5839405724206336	0.1946666666666667
ring-type	0.31514425825181835	0.6766688191210222	0.22231578947368422
stalk-surface-ar	0.3261874236536756	0.7134018659814799	0.22498245614035084
stalk-surface-br	0.33195967351053535	0.7232367806822597	0.2312982456140351
gill-size	0.35474671207785663	0.7708939815037107	0.24435087719298243
stalk-color-ar	0.36011798073930174	0.7421466564568185	0.28154385964912276
stalk-color-br	0.3651234183715839	0.7545887496491841	0.2826666666666666
bruises	0.3723667711514099	0.8043527510720476	0.2548771929824561
population	0.380738074001495	0.797100057313793	0.2780350877192982
habitat	0.40308362077746895	0.843266656398078	0.3098947368421052
stalk-root	0.4155826770039246	0.862546489260893	0.351719298245614
gill-spacing	0.4381816520532548	0.8971114797343223	0.3834385964912281
cap-shape	0.4684828616070681	0.9486925527461361	0.43522807017543863
cap-color	0.475673607068735	0.9631870587858761	0.40659649122807023
ring-number	0.4764864629107098	0.9607225008260788	0.4631578947368421
cap-surface	0.47987013313220656	0.9703576105987686	0.41894736842105273
veil-color	0.48795893399926843	0.9759798573073666	0.4807017543859649
gill-attachment	0.49130595386723097	0.9853761034883316	0.4818245614035087
stalk-shape	0.4936397233976238	0.990801910659947	0.4446315789473684
veil-type	0.49933930686365047	0.9990466112580625	0.4818245614035088

Figure 3. Attribute Scoring by Gain Scoring Method

When considering the different criteria for splitting whether that be Gini, Entropy, or Misclassification Error, the scoring and information provided from each is nearly identical when considered only in the ordering of the attributes by their scores. As a result the trees formed with the various split criteria end up mirroring each other closely. This result is shown in Figure (2) when comparing the accuracy performance for each of the trees when excluding any input from the Chi test statistic.

In further attempting to confirm tree similarity we looked into the depth and average depth of the resultant trees for the various feature selection types and their accuracy. Figure (4) shows the results of training with 500 test samples and testing against all ~7,000 training samples.

Split Criteria	Tree Depth	Average Tree Depth	Accuracy
Gini	5	3.6	0.9880
Misclassification	5	3.6	0.9935
Entropy	6	3.7	0.9956

Figure 4. Tree Depth Test Results

Based on the surface level results we went for a more general benchmark on tree differences. To find this benchmark for each criteria we created 100 trees with random samples from the test data and took their average depth and accuracy against all provided training examples. These results are shown in Figure (5).

Split Criteria	Tree Depth	Accuracy
Gini	3.81	0.9976
Misclassification	4.15	0.9979
Entropy	3.91	0.9978

Figure 5. Depth and Accuracy for 100 Trees trained on random subset

Attribute	Misclassification Tree Attribute Count	Gini Tree Attribute Count	Entropy Tree Attribute Count	Standard Deviation
odor	100	100	100	0
spore-print-color	97	95	96	1
cap-color	50	48	50	1.154700538
cap-surface	32	20	26	6
stalk-surface-br	28	25	18	5.131601439
stalk-surface-ar	27	23	17	5.033222957
cap-shape	22	23	24	1
stalk-color-br	18	20	8	6.429100507
habitat	17	25	48	16.09347694
gill-size	11	19	22	5.686240703
stalk-root	7	9	9	1.154700538
bruises	3	4	3	0.5773502692
gill-color	1	NA	1	0

Figure 6. Attribute frequency over 100 random trials

During this test we recorded the frequency with which attributes were included in the final tree made from the random sample of our training data. These results are shown in Figure (6). Through these split criteria tests we found that our trees end up following a very similar path of least resistance. As you can see in the random result attribute frequency table the different split criteria have minimal deviation when accounting for anomalous attributes being included in the tree.

C. Features and Target Variable

In our testing we confirmed the expectation that the most informative attributes were selected early on in tree creation. When analyzing this result we were able to find the clear correlation between the attributes and data homogeneity. In Figure (2) we can see the clear impact of high scoring attributes. Odor provides a number of immediate classifications based on its value. When excluding the top performing attributes we found highly variable maximum and average tree depth. The maximum we were able to accomplish is a depth of 17 for 6 excluded attributes.

IV. Conclusion

Considering the high accuracy levels of our decision trees on our validation data set as well as the accuracy of our random forest on the testing dataset, we can safely conclude that the task of classifying mushrooms was a perfect application of the random forest machine learning model. We found through our testing that odor provided the highest level of information gain by far, but that spore print and gill color attributes were effective predictors as well.

Another finding from our analysis regards the effectiveness of the criteria we used for measuring the impurity of our data splits. We found that regardless of the criteria used when forming our decision trees, there was no significant difference in the accuracy levels of our classifications. The lack of variation between the criteria was likely due to a limited variety of samples in our dataset as well as having a few key attributes which provided a significant amount of information gain for our trees.