



Introduction to Data Science – Methods and Tools 60070

Final Project

Lecturer : Dr. Jonathan Schler

Submitted by Adi Brill

Date: 3-4-2020

# Content

Abstract.....	3
1. Introduction .....	4
2. Background .....	6
3. Experiments .....	9
4. Results.....	10
4.1. Mushrooms.....	10
4.1.1. Expert Data.....	10
4.1.2. Amateur data .....	12
4.1.3. Amateur data with 10 percent.....	13
4.1.4. Amateur data with 30 percent.....	15
4.1.5. Amateur data with 50 percent.....	16
4.1.6. Layman data.....	17
4.1.7. Layman data with 10 percent .....	19
4.1.8. Layman data with 30 percent .....	20
4.1.9. Layman data with 50 percent .....	21
4.2. Cancer Dataset.....	23
4.2.1. Expert .....	23
4.2.2. Amateur .....	26
4.2.3. Amateur with 10 percent.....	27
4.2.4. Amateur with 30 percent.....	29
4.2.5. Amateur with 50 percent.....	30
4.2.6. Layman.....	31
4.2.7. Layman with 10 percent .....	32
4.2.8. Layman with 30 percent .....	33
4.2.9. Layman with 50 percent .....	34
5. Conclusions .....	36
6. Appendix 1 – Mushrooms Parameters List.....	38
7. Appendix 2 – Cancer Tumor Parameters List.....	47
Bibliography .....	48

## Abstract

Information gain is the main key that is used by Decision Tree Algorithms to construct a Decision Tree. Decision Trees algorithm will always try to maximize Information gain. An attribute with highest Information gain will split first.

Decision tree learning includes the gathering of a dataset from samples of the same type and harnessing it to find the attribute with the highest info gain. That is because the variable with the highest info gain is usually the one to subset the data into meaningful groups.

Information gathering is not a simple process. In most fields, one must either have common knowledge, partial-professional knowledge, or complete professional knowledge to determine an attribute's value. That is because some parameters are measured by universal scales and some are measured in scales relevant only to the data domain.

The current research examines two policies. Policy one – whenever constructing a decision tree, include only variables which fit the classifier's level.

Policy two – whenever constructing a decision tree include all relevant variables, taking the risk that the classifier will guess the value of a variable he does not recognize.

The above two policies were examined using two well known datasets from UCI website.

# 1. Introduction

Information gain is the main key that is used by Decision Tree Algorithms to construct a Decision Tree. Decision Trees algorithm will always try to maximize Information gain. An attribute with highest Information gain will split first.

Decision tree learning includes the gathering of a dataset from samples of the same type and harnessing it to find the attribute with the highest info gain. That is because the variable with the highest info gain is usually the one to subset the data into meaningful groups.

Information gathering is not a simple process. In most fields, one must either have common knowledge, partial-professional knowledge, or complete professional knowledge to determine an attribute's value. That is because some parameters are measured by universal scales and some are measured in scales relevant only to the data domain. The hypothesis is - the higher the required knowledge level is, the higher the probability for mistakes to occur (by none-expert). Our mission in this project would be to integrate the inaccurate measuring probability into the ID3 algorithm (Iterative Dichotomiser 3).

The main problem is that occasionally, some of the collected data may be false or inaccurate, and the decision-makers have no way of knowing whether the data is reliable. Which brings up the following question – is there a way to minimize the probability of false classification due to inaccurate measuring or prior knowledge?

Suppose there are three expertise levels, one being a total layman, two being an amateur in the relevant field, and three an educated expert.

ID3 algorithm uses us to generate a decision tree from a dataset. The classification results are divided into four groups:

- True Positive (TP) – Correctly classified as Positive.
- False Positive (FP) -Falsely classified as Positive.
- True Negative (TN) - Correctly classified as Negative.
- False Negative (FN) - Falsely classified as Negative.

In this project, we will research the mushroom's dataset<sup>1</sup>, and the Breast Cancer Wisconsin (Diagnostic) Data Set<sup>2</sup> from the UCI website. We will divide the data attributes into three levels of expertise in ascending order. Our underlying assumption would be that non-experts would have to either decide to rely on their knowledge entirely (i.e. build classification model based only on variables he/she is familiar with) or build decision tree based on the entire set of parameters and guess the values of the unfamiliar ones. We will try to determine how would different inaccuracy levels affect the decision-making process based on these two options.

[1] [mushroom's dataset](#)

[2] [Breast Cancer Wisconsin \(Diagnostic\) Data Set](#)

Thus, the dilemma is the following:

Who will be more successful?

- A nonexpert with a model fitting his/her capabilities, or
- A nonexpert who uses the expert model and guesses some values.

Similarly, we will take this dilemma one level below and ask with regards to the dummy decision makers:

Who will be more successful?

- A dummy with a model fitting his/her capabilities, or
- A dummy who uses the experts model and guesses some values.

## 2. Background

Suppose we are traveling in green meadows and run across some mushrooms. Being the mushrooms lovers that we are, we would like to harvest some for dinner, but how do we know if these mushrooms are edible? In order to make the most accurate decision, we need our decision tree to be as short, accurate, and easy as possible with the least chance of mistakes. Nevertheless, we are no mushroom experts, and we might recognize some of the parameters wrong. In response to that, our goal is to incorporate the FN and accuracy rate, to select our ideal decision tree and answer the question –

### **can a data scientist recover for the absence of an expert?**

The mushrooms dataset contains over eight thousand records describing twenty-two parameters of wild mushrooms. The parameters include odor, gill-color, population, habitat, and more. This dataset was chosen due to its variety of parameters complexity. Some of them are quite easy to classify, while others may require amateur or professional experience. Ultimately the dataset divides the mushrooms into two groups: poisoned (Positive) and edible (Negative).

The second dataset describes Samples of cancer tumors. The reason why this dataset was chosen is that much like the mushrooms' dataset, it has various parameters, some of them are easy to recognize, and some require significant medicinal experience.

What we are most concerned about is the FN option. The reason why FN options are more concerning rather than FP options is FP describes cases where we have classified an edible mushroom as poisoned. In that case, no harm caused. In contrast, FN describes cases where we have classified a poisoned mushroom as edible. Thus, we are at risk of being poisoned.

Similarly, FP cases of cancer refer to a negative patient tested as positive. In that case, further examinations will be applied, and if the patient is, in fact, healthy, that would eventually come out in later examinations. In cases of FN, the patients will be notified to return home instead of starting treatment. That is precisely why we would like to avoid that option as much as possible.

First, we must explain the dataset's attributes (The detailed list of parameters is in appendix 1). Bear in mind that not all mushrooms parameters are alike, and neither are cancer tumors. Some are very easy to recognize as they rely on measuring skills anyone has, such as perspective(wide, narrow) or colors (red, white, yellow).

However, some parameters require a much broader knowledge. For example, the mushroom's population parameter requires knowing whether a mushroom is common or rare; that kind of information is only at the hand of a real expert.

We hypothesize that if a person tries to classify parameters beyond his level, he will have to guess their value. We will try to determine what approach is better and under what circumstances. Is it better to decide only by parameters one is a hundred percent positive about, or is it better to make a guess, as wild as that guess would be? What would be the worst-case scenario? How would that answer change depending on the percentage of errors one might make?

We have categorized the parameters into three complexity levels based on how difficult it will be to determine their value—one being the simplest (layman) level(1), two being the amateur level(2), and three the expert level(3).

For the mushrooms' dataset:

Table 1: Mushroom dataset Level definition

No.	Name	Level
1	cap-shape	2
2	cap-surface	3
3	cap-color	1
4	bruises	1
5	odor	3
6	gill-attachment	3
7	gill-spacing	1
8	gill-size	2
9	gill-color	1
10	stalk-shape	1
11	stalk-root	2
12	stalk-surface-above-ring	3
13	stalk-surface-below-ring	3
14	stalk-color-above-ring	1
15	stalk-color-below-ring	1
16	veil-type	2
17	veil-color	1
18	ring-number	1
19	ring-type	2
20	spore-print-color	1
21	population	3
22	habitat	1

To conclude, we have :

11 Layman parameters, 6 Amateur parameters, and 5 Expert parameters

For the Cancer dataset:

Table 2: Cancer dataset Level definition

No.	Name	Level
1	Clump-Thickness	1
2	Uniformity-of-Cell-Size	1
3	Uniformity-of-Cell-Shape	1
4	Marginal-Adhesion	2
5	Single-Epithelial-Cell-Size	2
6	Bare-Nuclei	1
7	Bland-Chromatin	3
8	Normal-Nucleoli	3
9	Mitoses	3

(The parameters of this dataset are described in appendix 2).

To conclude, we have :

4 Layman parameters, 2 Amateur parameters, and 3 Expert parameters



### 3. Experiments

First, we will retrieve the mushrooms database from the UCI website. Since it contains categorical parameters, we had to number each parameter's options in order to simplify the analysis process.

On each expertise level, we will generate a tree and compare the FN rate of both trees, the tree with level compatible parameters, and the tree with guessing parameters. We repeat this process for each error percentage; 10%,30% and 50%, and decide what is the best tree in each situation. We will use ID3 and KNN visualizations for comparison.

After that stage, we have created designated files for the layman and amateur levels. For the amateur level, we have created a file that contains only amateur and layman parameters (those he is familiar with – level 1 and 2) and a file that contains all parameters but with 10,30 and 50 percent of errors in expert parameters. For the layman, we have done the same but with errors in both amateur and expert levels in 10-30 and 30-50 percent, respectively.

The cases are as follows:

- An amateur can be wrong in 10, 30, or 50 percent of the unknown expert parameters. Ten percent means the amateur is making educated guesses, so most of them are right, and 50 percent means the amateur is making wild guesses based on no knowledge whatsoever. Thirty percent is somewhere in between the two.
- A non-specialist can be wrong in either 10,30 or 50 percent of amateur parameters, and either 30,50 or 70 percent of expert parameters.

We assume the expert parameters would be more difficult for the layman than the amateur parameters. The assumption is trying to determine the worst-case scenario; Thus, those are arbitrary limits chosen for experiment sake only. If the cases described will not provide a clear answer to the question at hand, we will create additional files with different percentages of mistakes and examine the results once again.

We are trying to test a methodology we could assist in a case of no expert in hand. We will test this methodology for the cancer tumors domains as well. Our goal is to reach the lowest FN rate in case of a lousy classification.

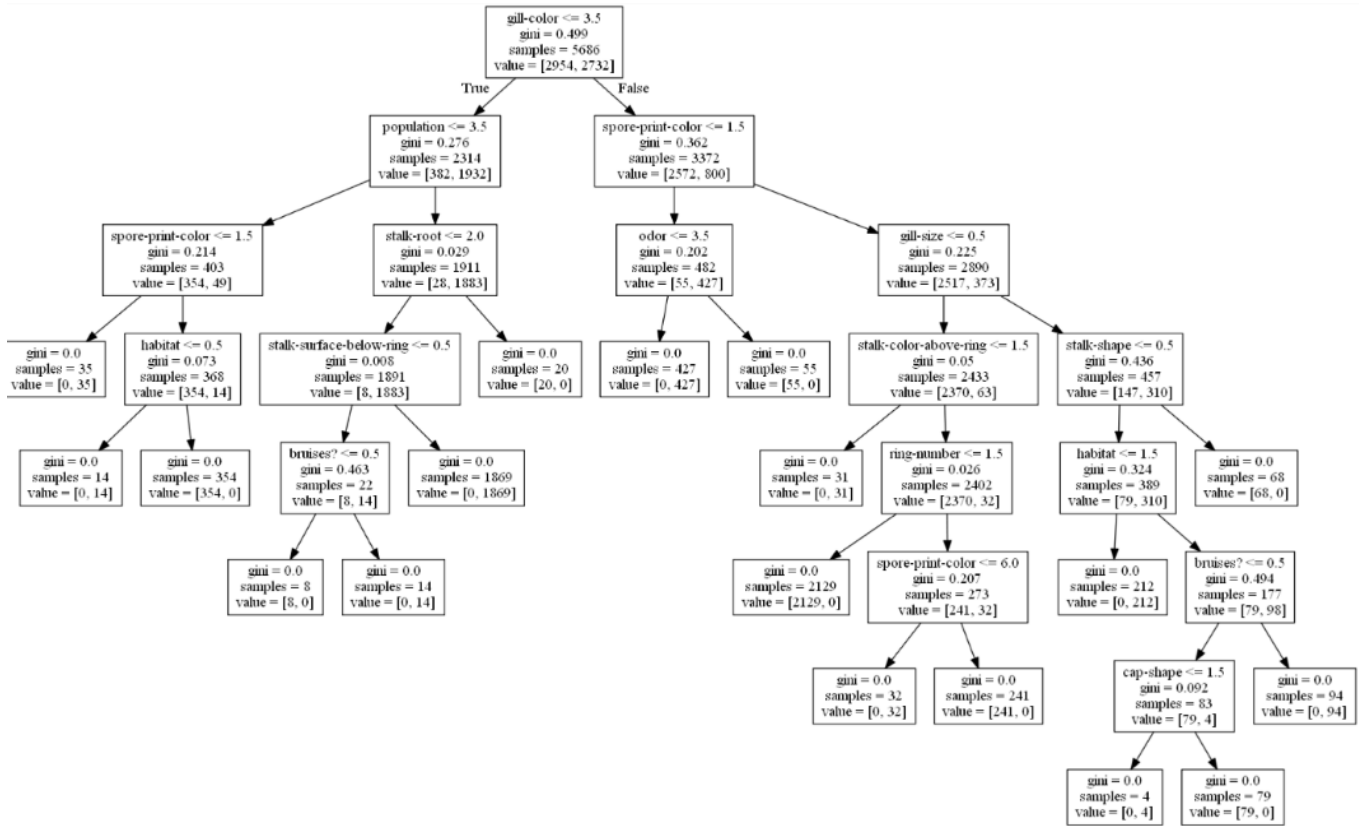
After we have all the results, we will compare the accuracy and FN rate for every case and decide which option suits every expertise level.

## 4. Results

### 4.1. Mushrooms

#### 4.1.1. Expert Data

First, we will run a decision tree on both datasets to know the baseline accuracy and FN rate. We can see that the tree depth is eight, and the gill-color was chosen as the root of the tree.



accuracy using decision tree: 100.0 %  
Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=1)]: Using backend LokyBackend with 4 concurrent workers.

```
[[1383  0]
 [ 4 1294]]
```

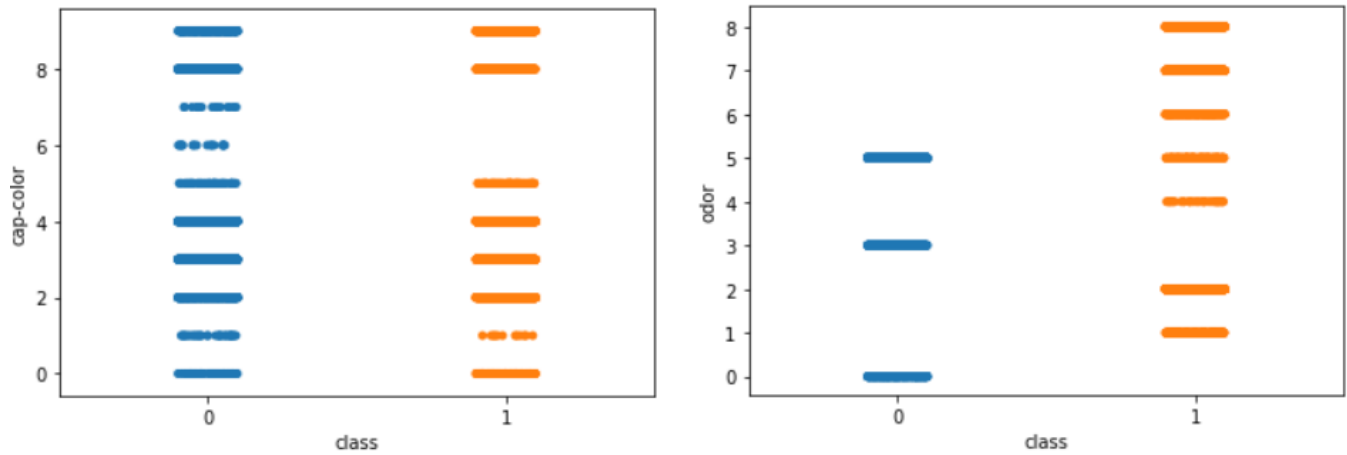
CROSS VALIDATION ACCURACY SCORE: 1.0

```
# == CLASSIFICATION REPORT == #
precision  recall  f1-score  support

0.0       1.00    1.00    1.00    1383
1.0       1.00    1.00    1.00    1298

accuracy
macro avg    1.00    1.00    1.00    2681
weighted avg 1.00    1.00    1.00    2681
```

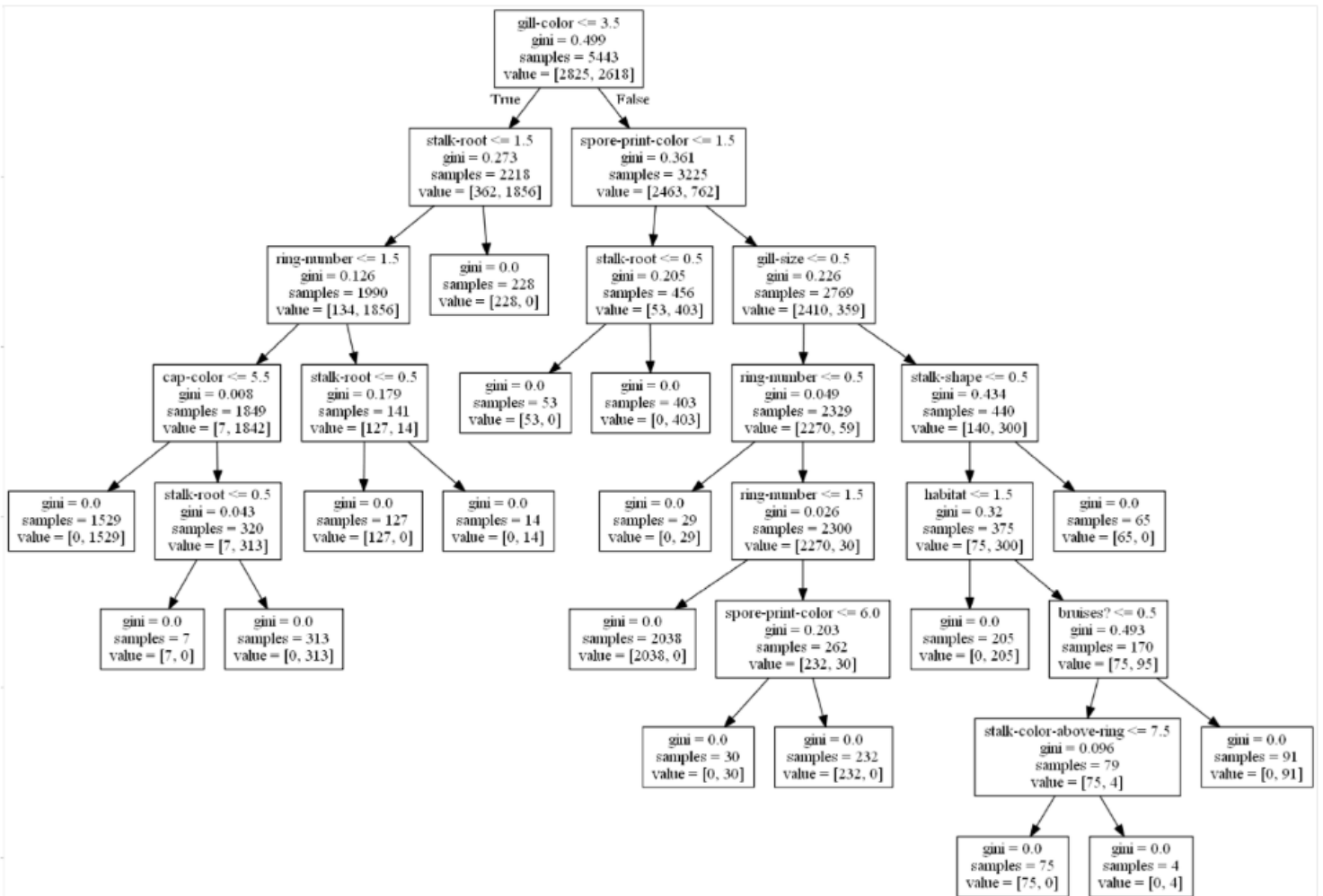
The accuracy of the model is 100%. And the FN rate is 0.



We can see from these charts that odor has significant importance to the classification process since there are three edible odors, and the seven other odors are poisonous. Another conclusion is that there are only two cap-colors which poisonous mushrooms do not show.

#### 4.1.2. Amateur data

Now we will test the model at the amateur level. The tree depth is still eight however the model



accuracy dropped to 99.7% and the FN rate is now 70.

accuracy using decision tree: 99.7 %  
Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers

```
[[1383    0]
 [    0 1298]]
```

CROSS VALIDATION ACCURACY SCORE: 1.0

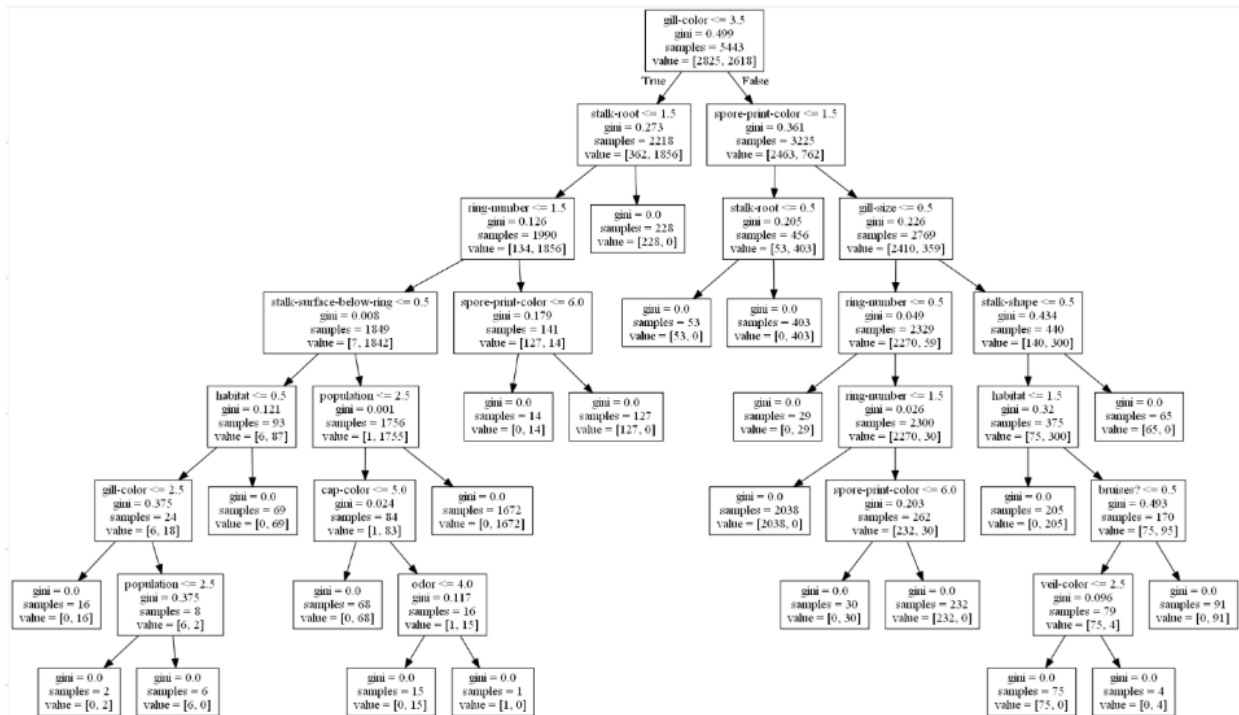
```

# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

   0.0         1.00         1.00         1.00        1383
   1.0         1.00         1.00         1.00        1298

 accuracy          1.00          1.00          1.00        2681
 macro avg         1.00          1.00          1.00        2681
 weighted avg         1.00          1.00          1.00        2681
```

### 4.1.3. Amateur data with 10 percent



accuracy using decision tree: 99.8 %

Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 conc

```
[[1382  1]
 [ 0 1298]]
```

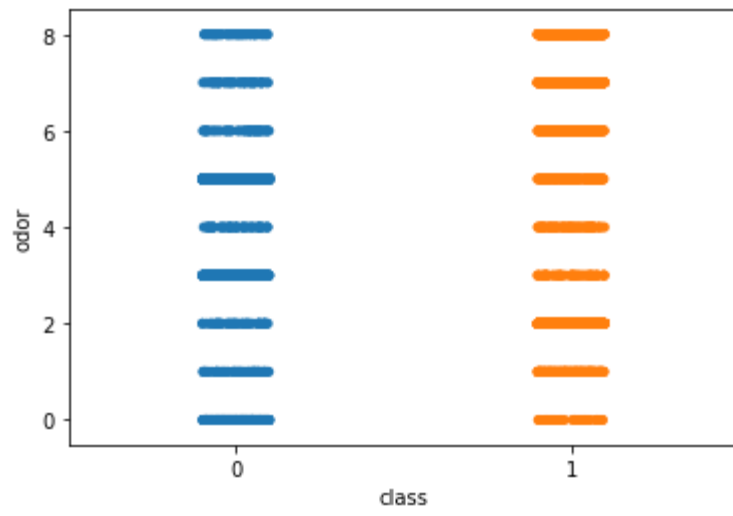
CROSS VALIDATION ACCURACY SCORE: 0.9998162777879845

```
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

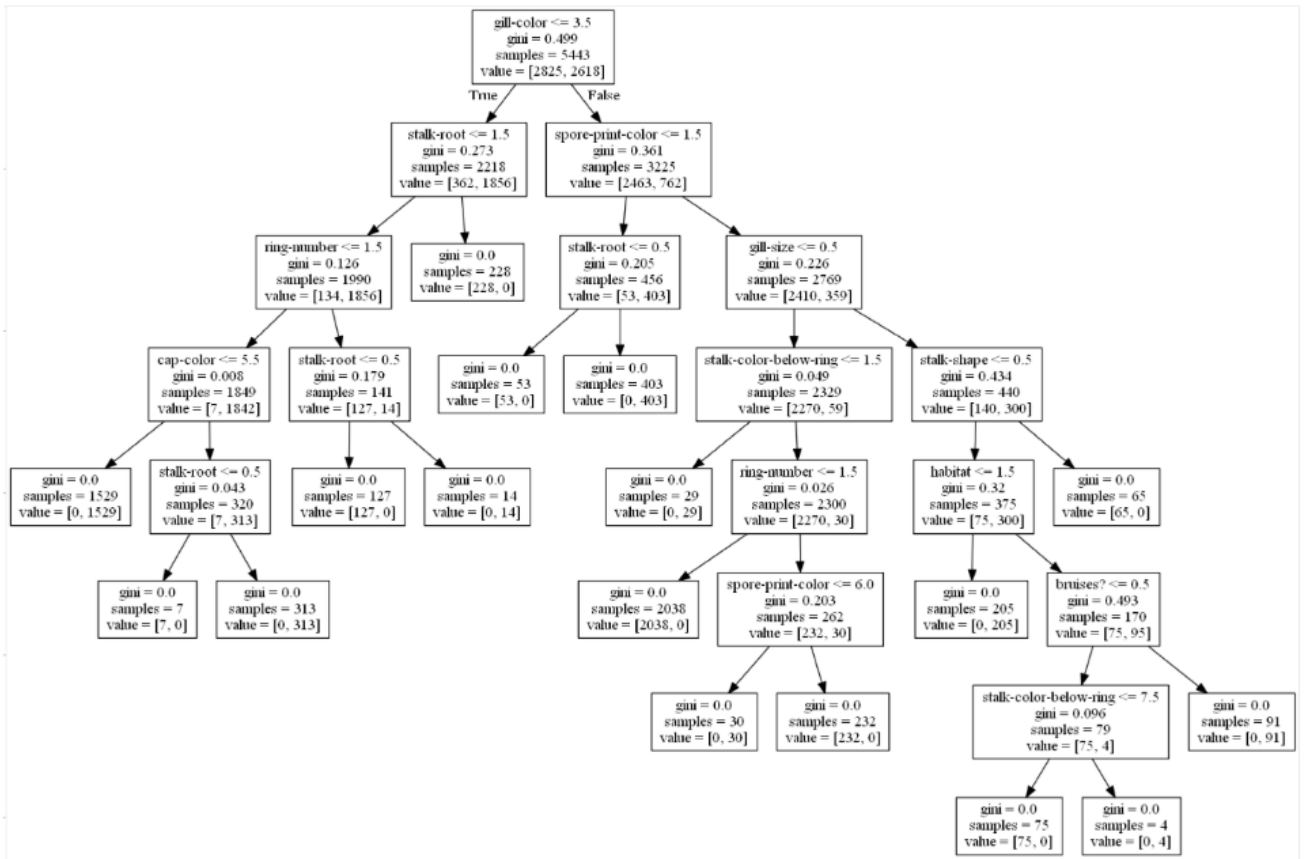
0.0          1.00      1.00      1.00     1383
1.0          1.00      1.00      1.00     1298

accuracy
macro avg      1.00      1.00      1.00     2681
weighted avg   1.00      1.00      1.00     2681
```

We can also see that the odor attribute, which was a handy parameter to distinguish between the classes is no longer such.



#### 4.1.4. Amateur data with 30 percent



accuracy using decision tree: 99.9 %

Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 concurrent

```
[[1383    0]
 [   4 1294]]
```

CROSS VALIDATION ACCURACY SCORE: 1.0

```

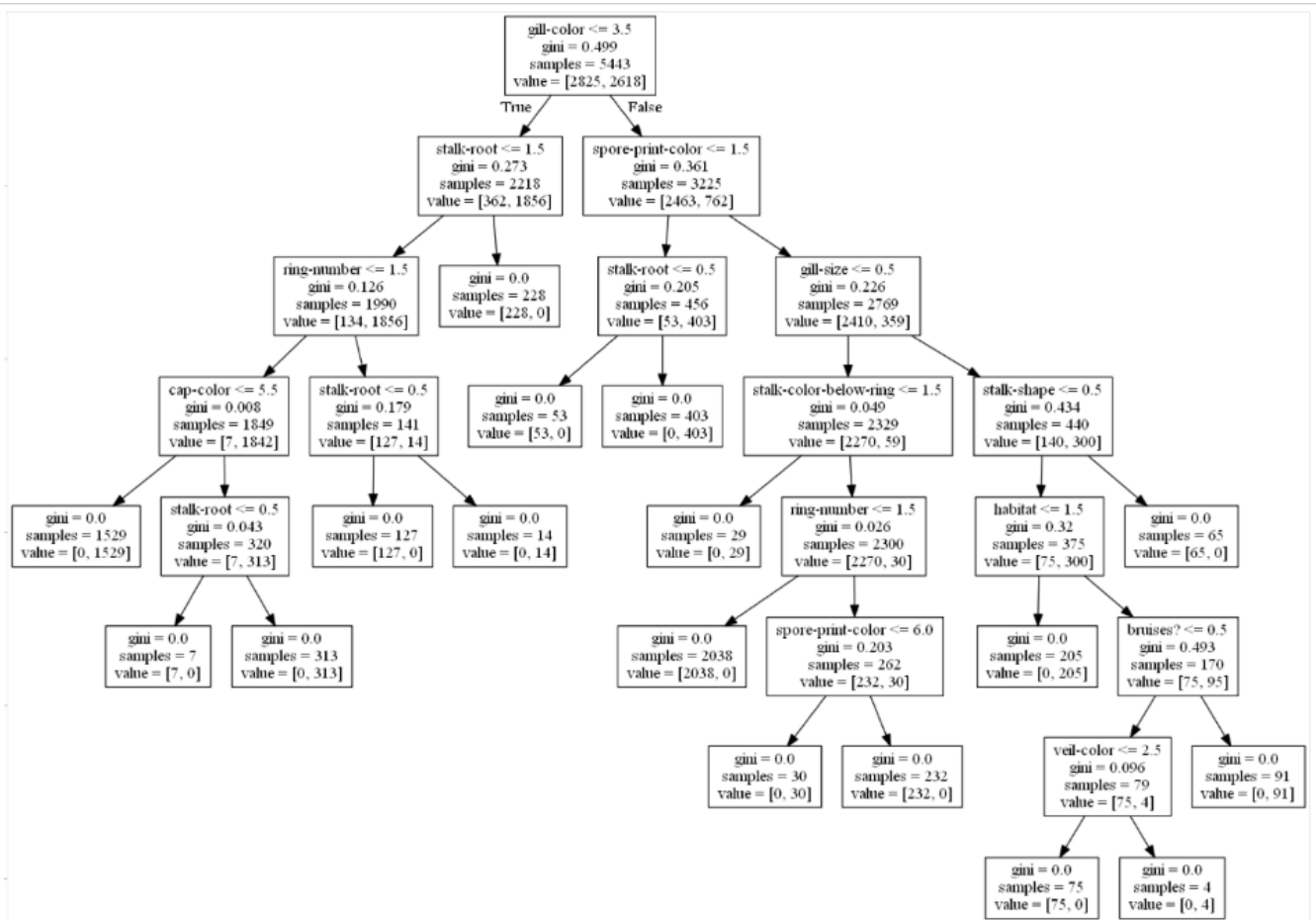
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

   0.0         1.00         1.00         1.00        1383
   1.0         1.00         1.00         1.00        1298

 accuracy          1.00          1.00          1.00        2681
 macro avg          1.00          1.00          1.00        2681
weighted avg          1.00          1.00          1.00        2681
```

#### 4.1.5. Amateur data with 50 percent

We can see that the tree has not changed much from 30 percent.



```
accuracy using decision tree: 100.0 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[[1383  0]
 [ 0 1298]]
CROSS VALIDATION ACCURACY SCORE: 1.0
```

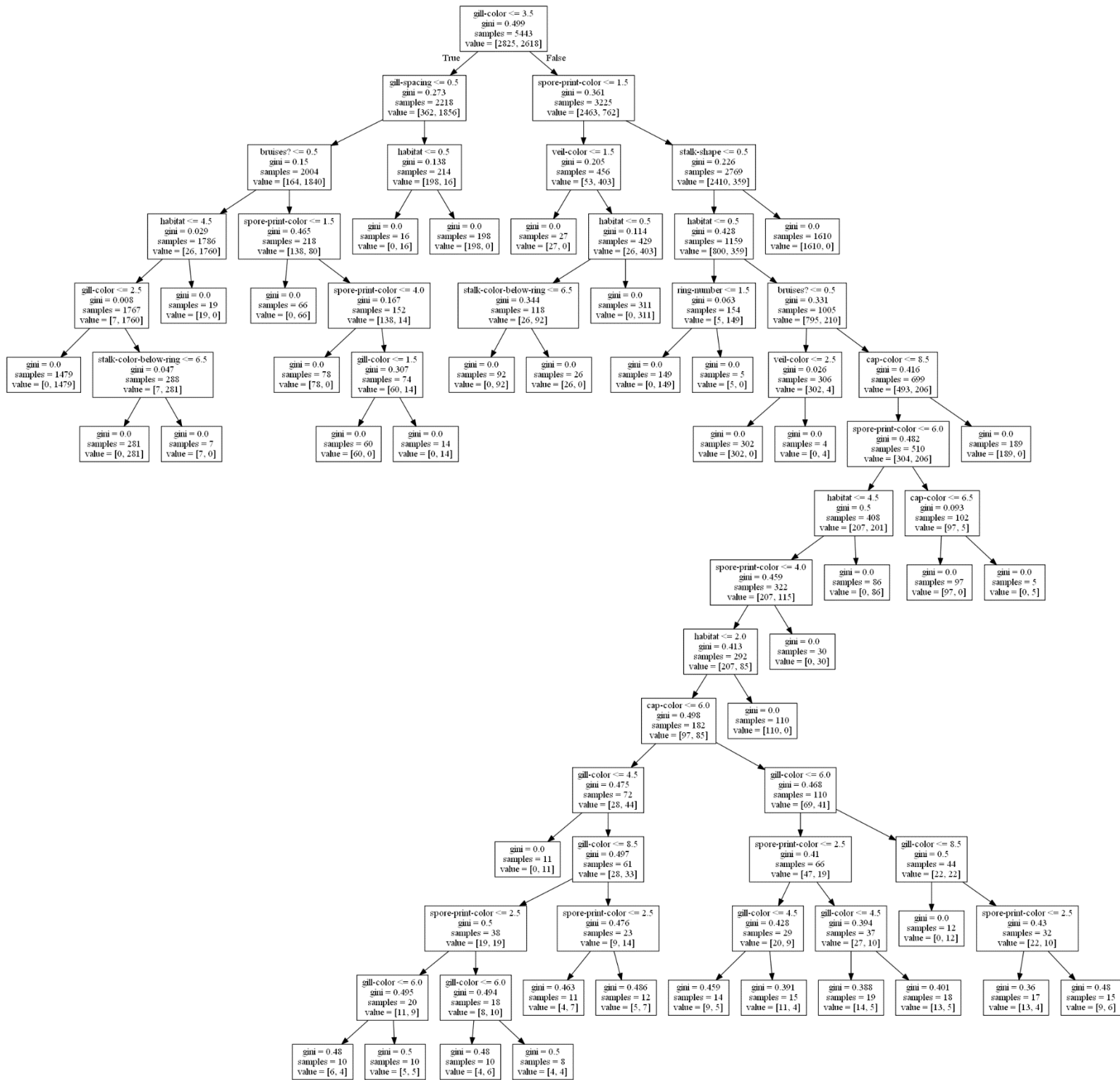
```
# == CLASSIFICATION REPORT == #
precision    recall  f1-score   support

    0.0         1.00         1.00         1.00        1383
    1.0         1.00         1.00         1.00        1298

 accuracy          1.00          1.00          1.00        2681
 macro avg         1.00          1.00          1.00        2681
weighted avg         1.00          1.00          1.00        2681
```



#### 4.1.6. Layman data



accuracy using decision tree: 98.7 %  
Fitting 5 folds for each of 9 candidates, totalling 45 fits

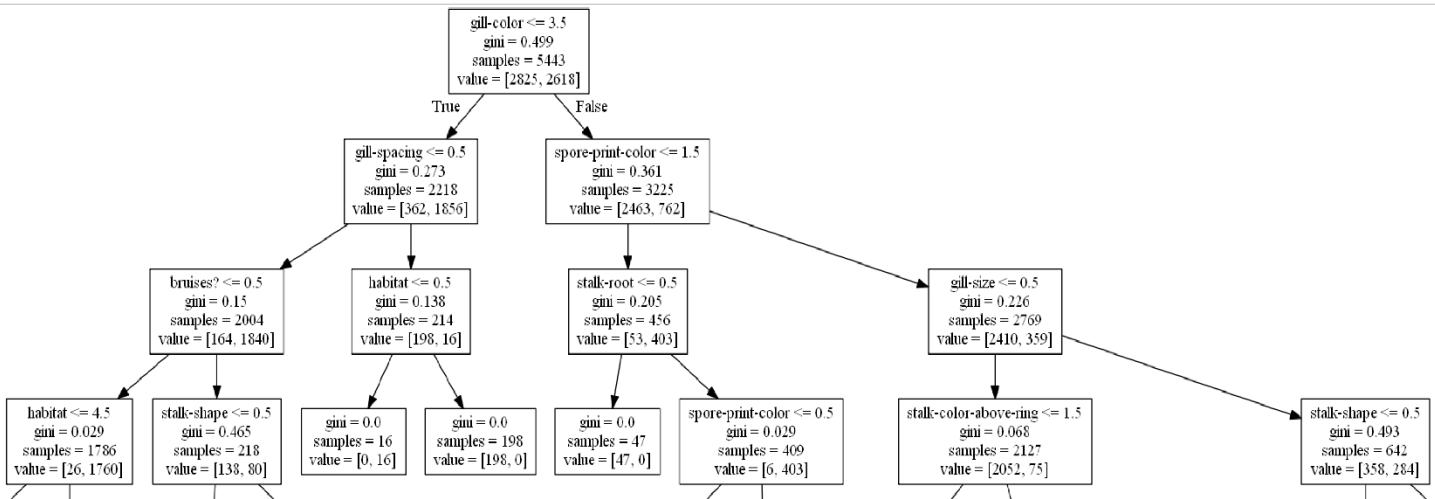
[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers

[[1372 11]  
[ 30 1268]]

CROSS VALIDATION ACCURACY SCORE: 0.9871394451589197

	# == CLASSIFICATION REPORT == #			
	precision	recall	f1-score	support
0.0	0.98	0.99	0.99	1383
1.0	0.99	0.98	0.98	1298
accuracy			0.98	2681
macro avg	0.99	0.98	0.98	2681
weighted avg	0.98	0.98	0.98	2681

#### 4.1.7. Layman data with 10 percent



I've cut the tree to the first four levels since it too small to see.

accuracy using decision tree: 96.4 %

Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 concurrent wo

```
[[1378  5]
 [ 16 1282]]
```

CROSS VALIDATION ACCURACY SCORE: 0.9920999448833364

```

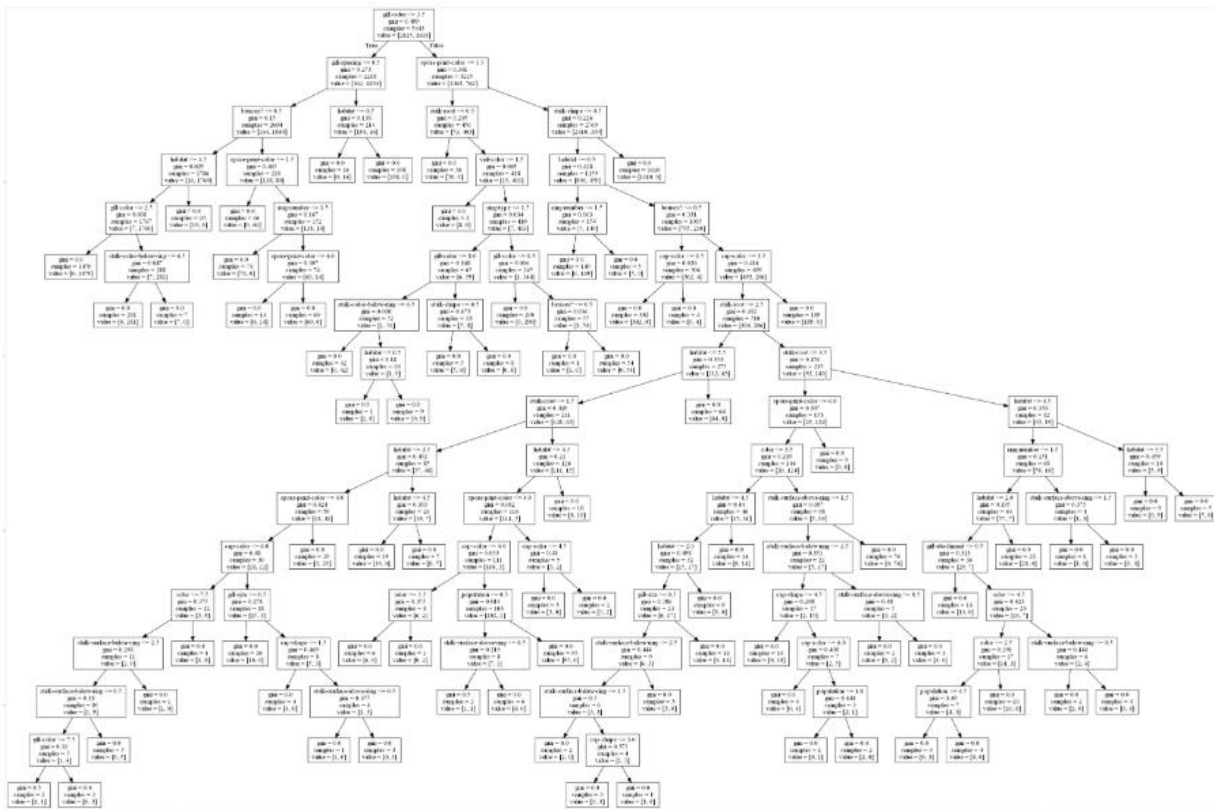
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

    0.0       0.99      1.00      0.99      1383
    1.0       1.00      0.99      0.99      1298

 accuracy          0.99      0.99      0.99      2681
 macro avg         0.99      0.99      0.99      2681
 weighted avg         0.99      0.99      0.99      2681
  
```

#### 4.1.8. Layman data with 30 percent

We can see now that the tree is much more complicated.



```
accuracy using decision tree: 92.9 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits
```

```
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers
```

```
[[1372  11]
 [ 15 1283]]
CROSS VALIDATION ACCURACY SCORE: 0.9867720007348888
```

```

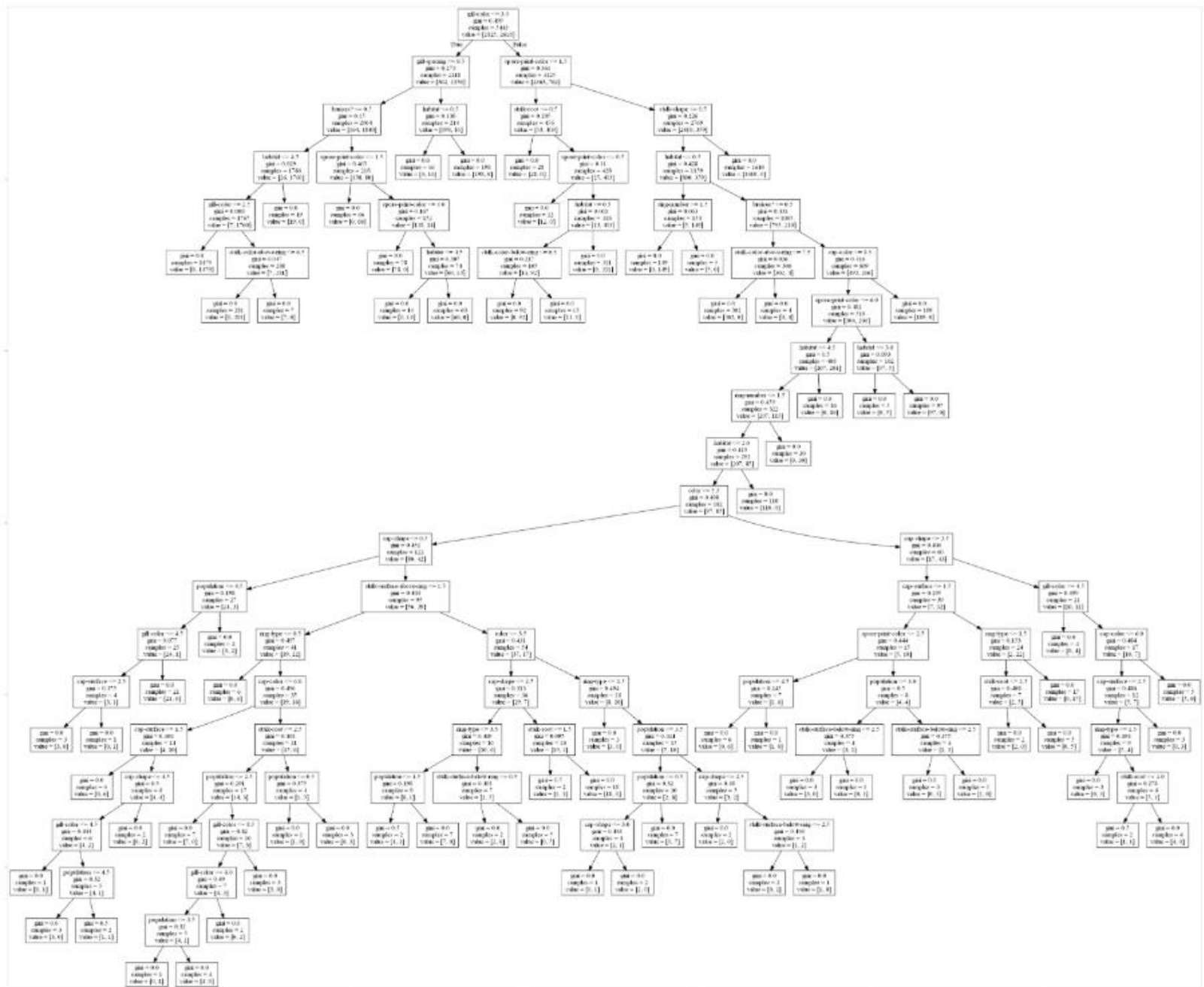
# == CLASSIFICATION REPORT == #
precision    recall  f1-score   support

   0.0         0.99         0.99         0.99       1383
   1.0         0.99         0.99         0.99       1298

 accuracy          0.99          2681
  macro avg         0.99         0.99          2681
 weighted avg         0.99         0.99          2681

```

#### 4.1.9. Layman data with 50 percent



accuracy using decision tree: 90.4 %  
Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 conc

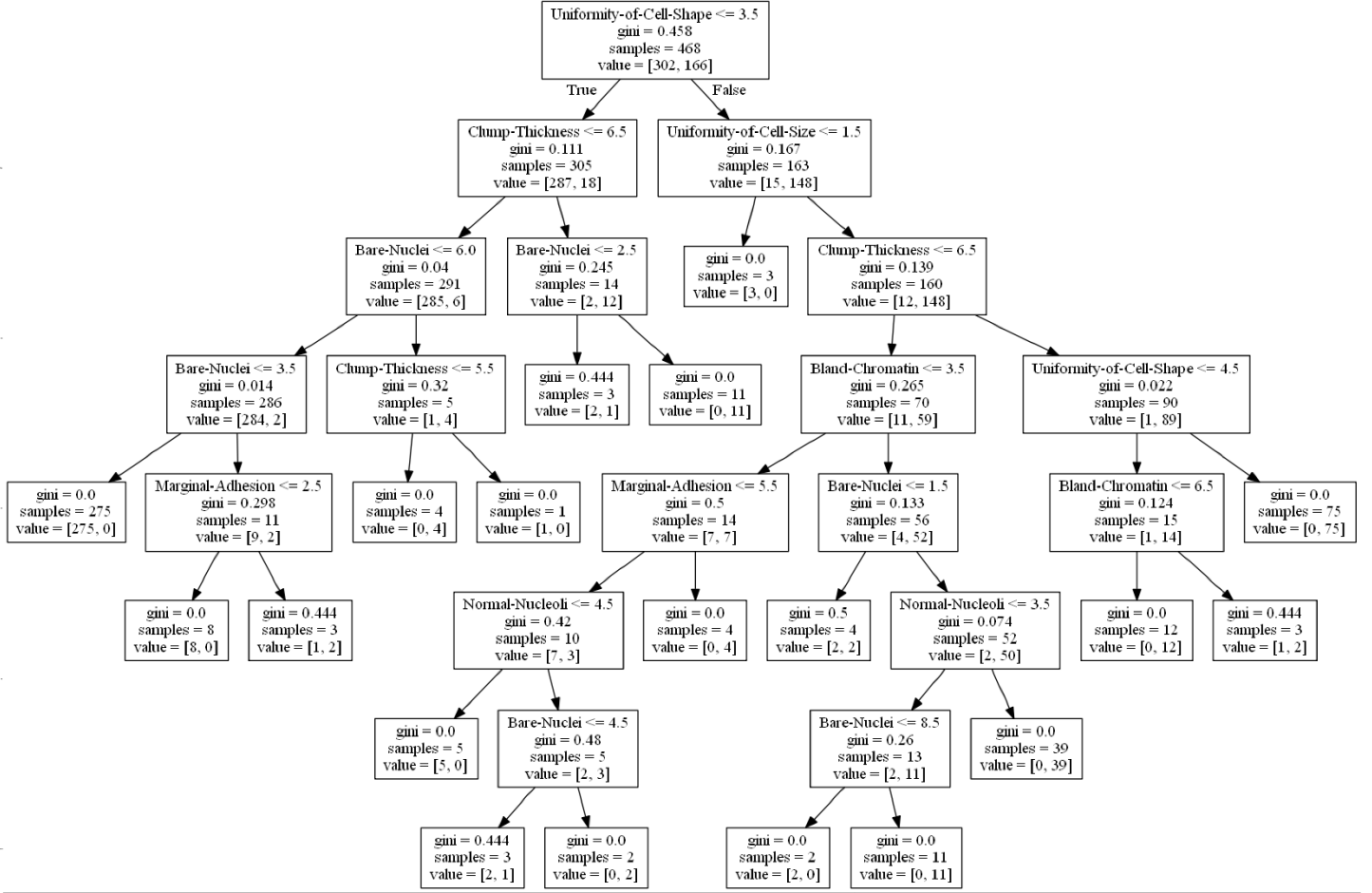
[[1364 19]  
[ 20 1278]]

CROSS VALIDATION ACCURACY SCORE: 0.9860371118868271

	# == CLASSIFICATION REPORT == #			
	precision	recall	f1-score	support
0.0	0.99	0.99	0.99	1383
1.0	0.99	0.98	0.98	1298
accuracy			0.99	2681
macro avg	0.99	0.99	0.99	2681
weighted avg	0.99	0.99	0.99	2681

## 4.2. Cancer Dataset

### 4.2.1. Expert



```

accuracy using decision tree: 97.8 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent worl

[[151  5]
 [ 9 66]]
CROSS VALIDATION ACCURACY SCORE: 0.9337606837606838

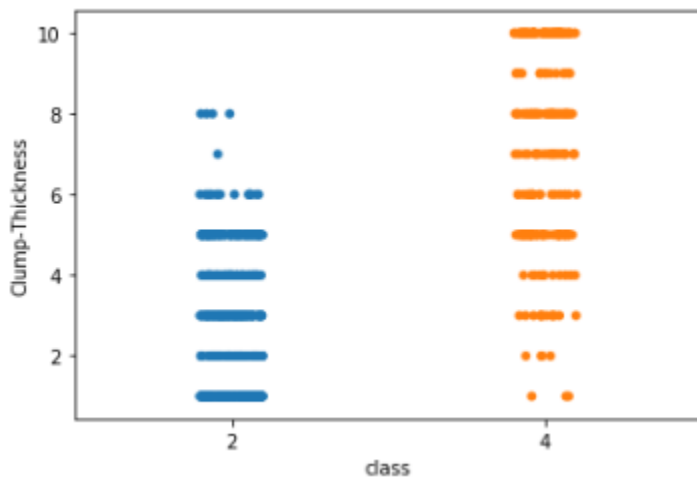
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

2.0          0.94      0.97      0.96       156
4.0          0.93      0.88      0.90        75

accuracy          0.94          0.94          0.94       231
macro avg          0.94          0.92          0.93       231
weighted avg          0.94          0.94          0.94       231

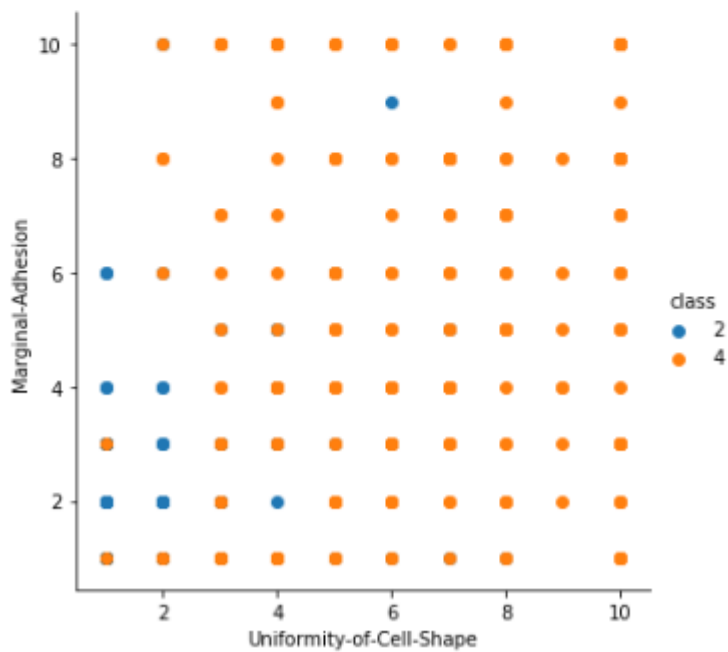
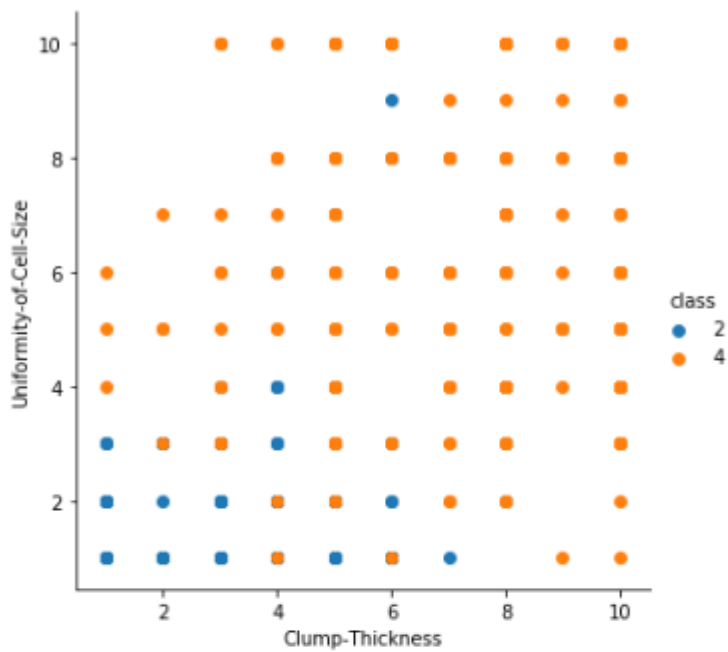
```

In this chart, we can see that high clump thickness is more recognized with malignant clumps.

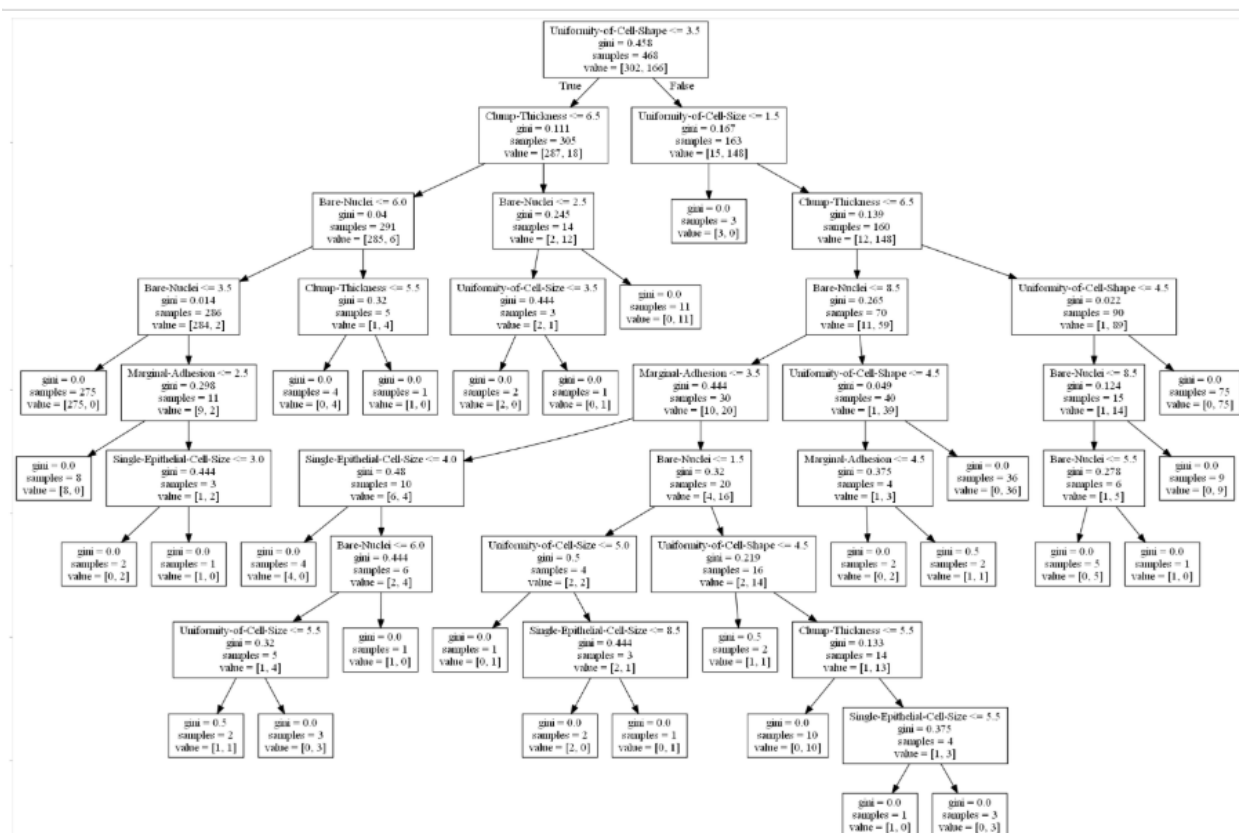




Here we can see that there is some correlation between the uniformity of cell size and the clump-thickness regarding its classification as a Benign (2) or malignant (4).



## 4.2.2. Amateur



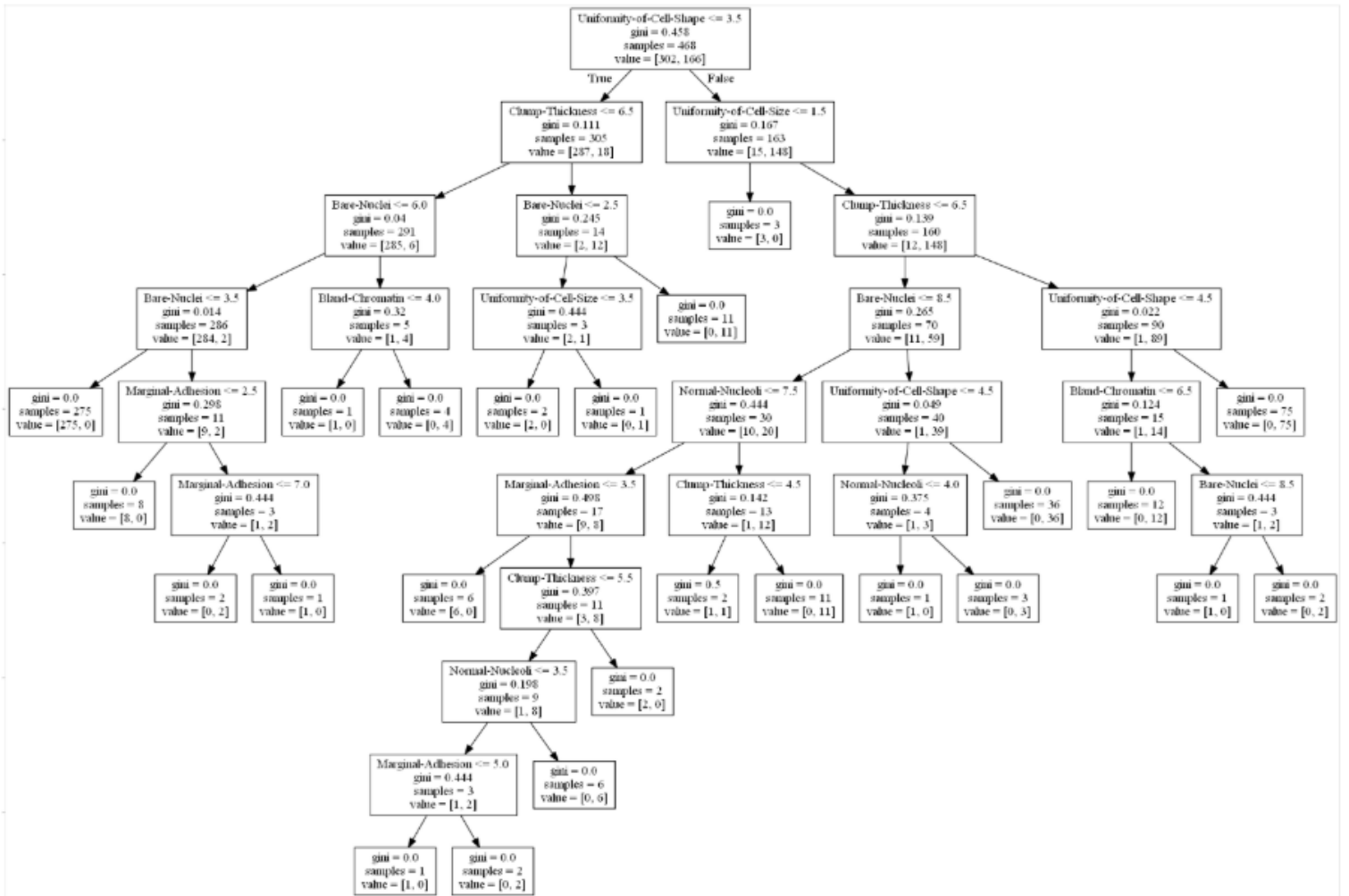
accuracy using decision tree: 95.7 %  
 Fitting 5 folds for each of 9 candidates, totalling 45 fits  
 [[152 4]  
 [ 7 68]]  
 CROSS VALIDATION ACCURACY SCORE: 0.9487179487179487

```
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

   2.0       0.96    0.97    0.97     156
   4.0       0.94    0.91    0.93      75

 accuracy          0.95          231
 macro avg         0.95          231
 weighted avg      0.95          231
```

### 4.2.3. Amateur with 10 percent



---

```

accuracy using decision tree: 97.4 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n_jobs=-1)]: Using backend LokyBackend with 4 concurrent

[[154  2]
 [ 14 61]]
CROSS VALIDATION ACCURACY SCORE: 0.9465811965811965

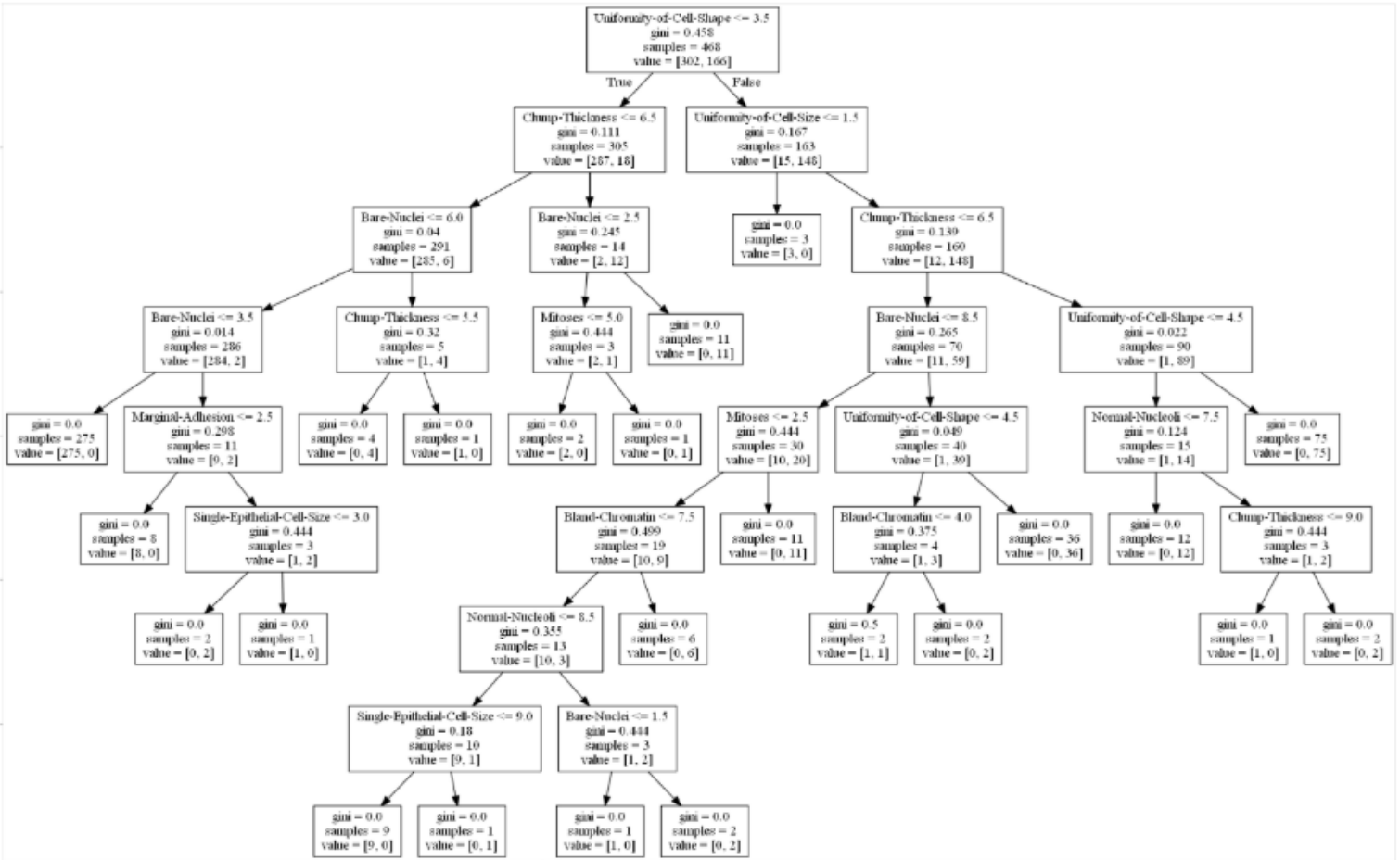
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

   2.0         0.92         0.99         0.95         156
   4.0         0.97         0.81         0.88          75

 accuracy          0.93         231
 macro avg         0.94         0.90         0.92         231
weighted avg         0.93         0.93         0.93         231

```

#### 4.2.4. Amateur with 30 percent



accuracy using decision tree: 96.5 %

Fitting 5 folds for each of 9 candidates, totalling 45 fits

[Parallel(n\_jobs=-1)]: Using backend LokyBackend with 4 concurrent workers.

```
[[152  4]
 [ 10 65]]
```

CROSS VALIDATION ACCURACY SCORE: 0.9529914529914529

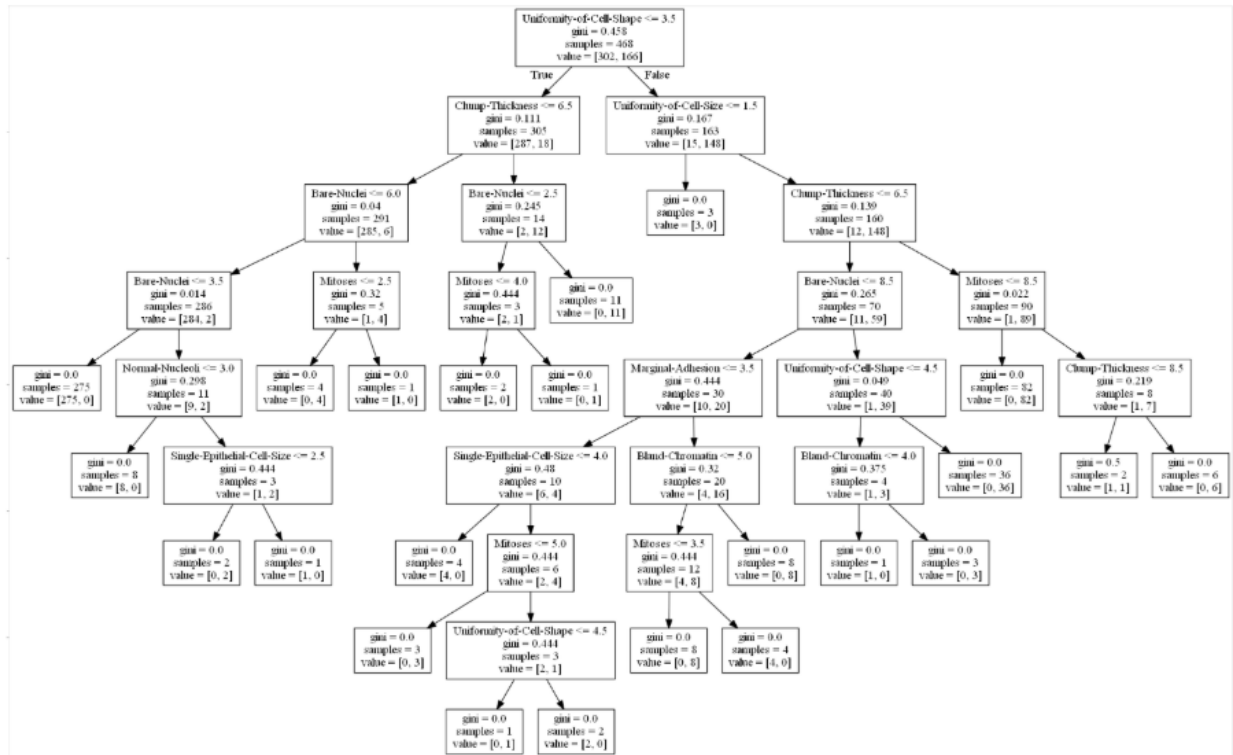
```

# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

   2.0         0.94         0.97         0.96         156
   4.0         0.94         0.87         0.90          75

 accuracy          0.94          0.94          0.94         231
 macro avg         0.94         0.92         0.93         231
 weighted avg         0.94         0.94         0.94         231
```

#### 4.2.5. Amateur with 50 percent



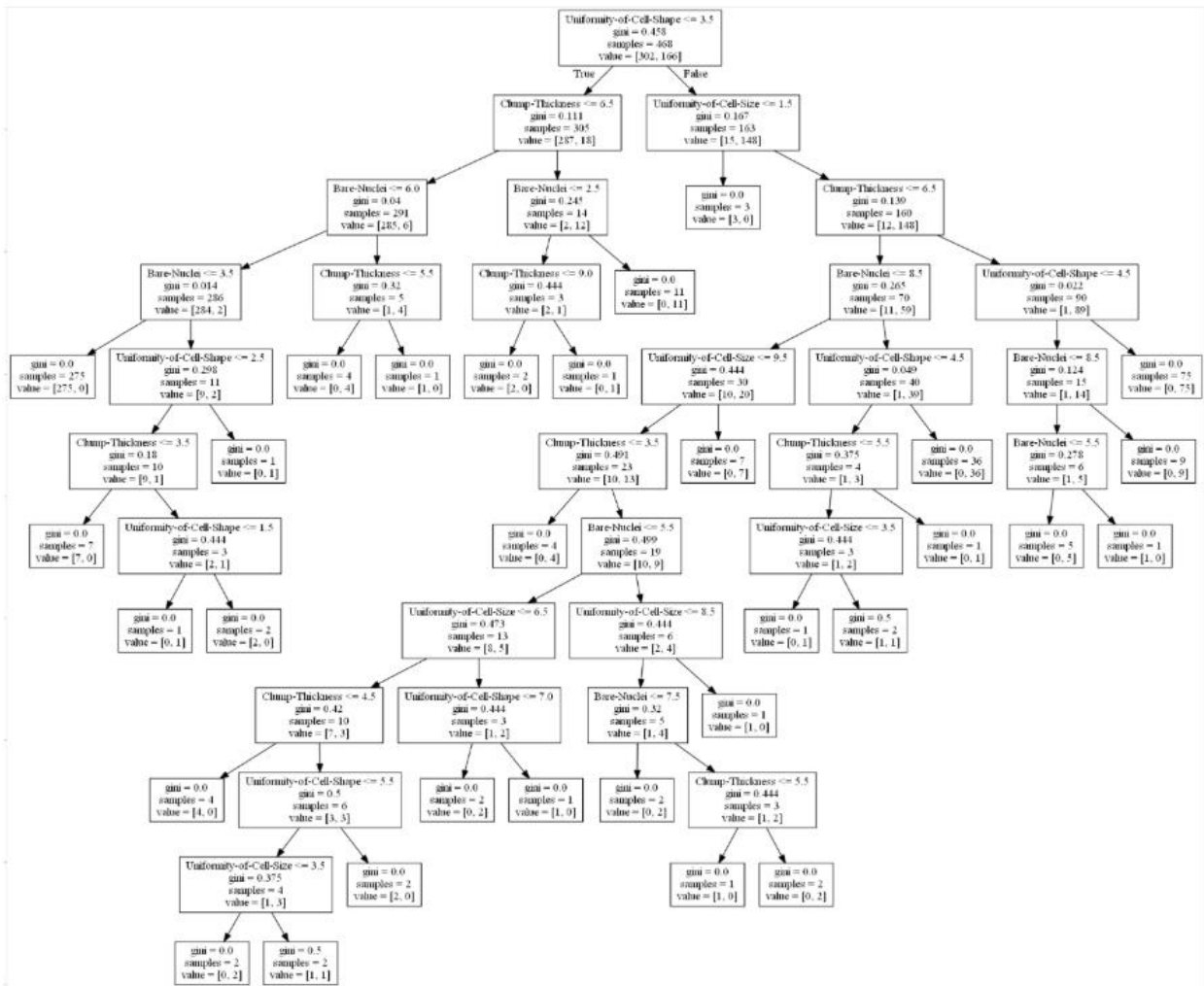
accuracy using decision tree: 96.5 %  
 Fitting 5 folds for each of 9 candidates, totalling 45 fits  
 [[153 3]  
 [ 6 69]]  
 CROSS VALIDATION ACCURACY SCORE: 0.9358974358974359

```
# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

2.0         0.96     0.98     0.97       156
4.0         0.96     0.92     0.94        75

accuracy                0.96       231
macro avg              0.96     0.95     0.96       231
weighted avg           0.96     0.96     0.96       231
```

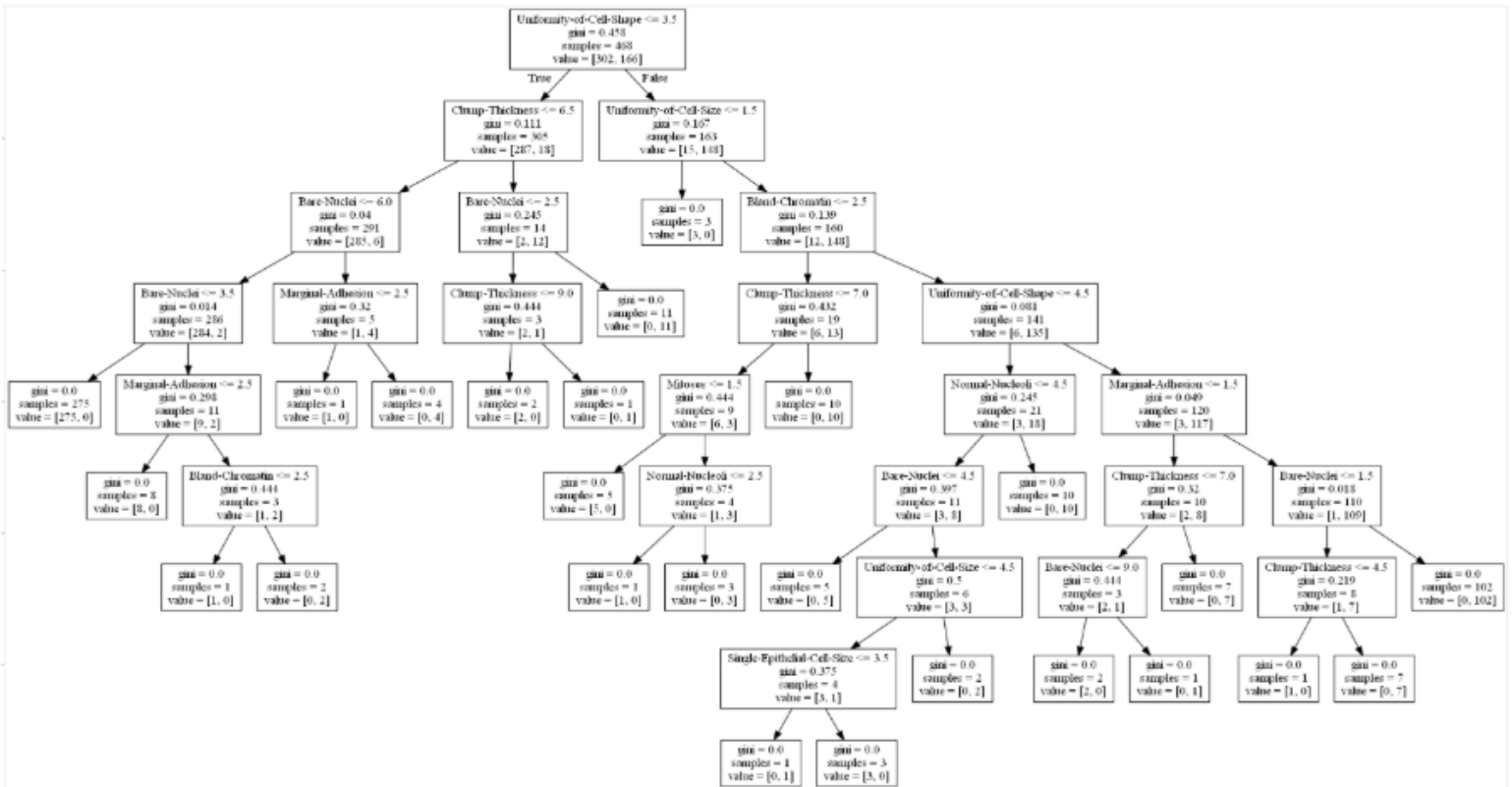
#### 4.2.6. Layman



accuracy using decision tree: 95.2 %  
 Fitting 5 folds for each of 9 candidates, totalling 45 fits  
 [[149 7]  
 [ 4 71]]  
 CROSS VALIDATION ACCURACY SCORE: 0.9508547008547008

# === CLASSIFICATION REPORT === #				
	precision	recall	f1-score	support
2.0	0.97	0.96	0.96	156
4.0	0.91	0.95	0.93	75
accuracy			0.95	231
macro avg	0.94	0.95	0.95	231
weighted avg	0.95	0.95	0.95	231

#### 4.2.7. Layman with 10 percent



```
accuracy using decision tree: 96.5 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[[150  6]
 [ 5 70]]
CROSS VALIDATION ACCURACY SCORE: 0.9465811965811965
```

```

# == CLASSIFICATION REPORT == #
precision    recall  f1-score   support

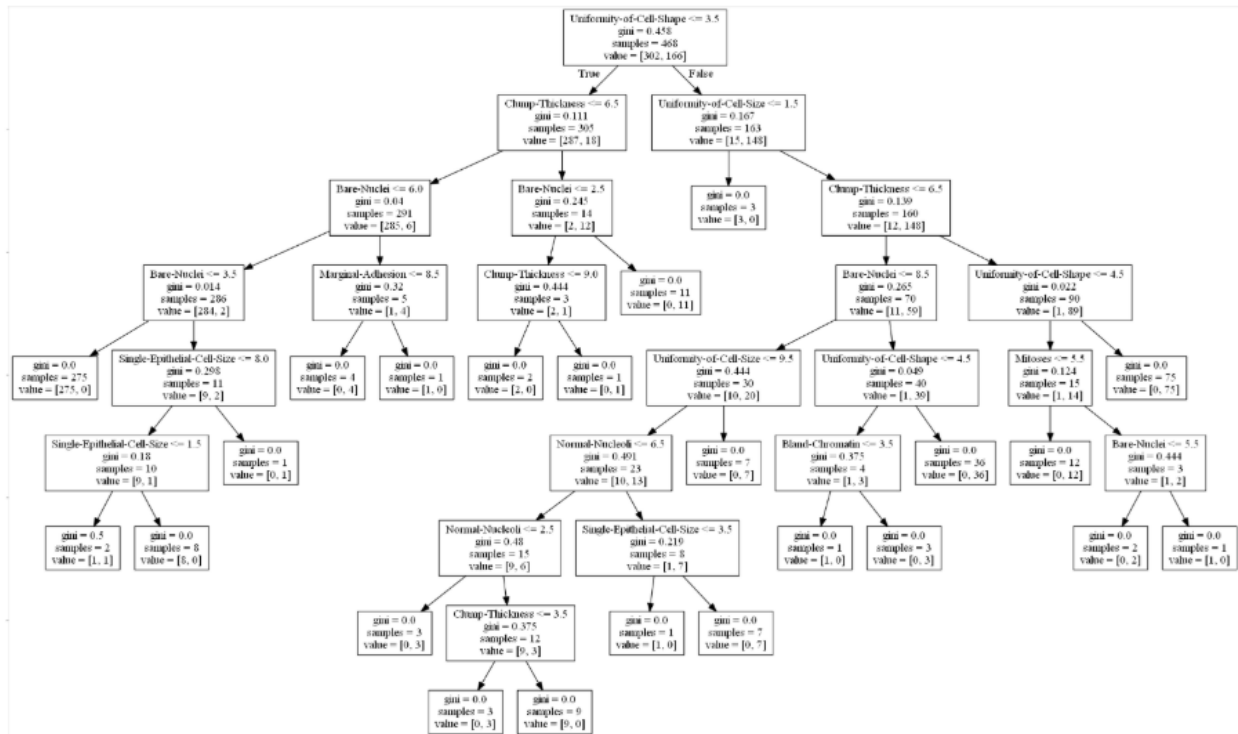
 2.0         0.97     0.96     0.96       156
 4.0         0.92     0.93     0.93        75

 accuracy                    0.95       231
 macro avg         0.94     0.95     0.95       231
 weighted avg      0.95     0.95     0.95       231

```



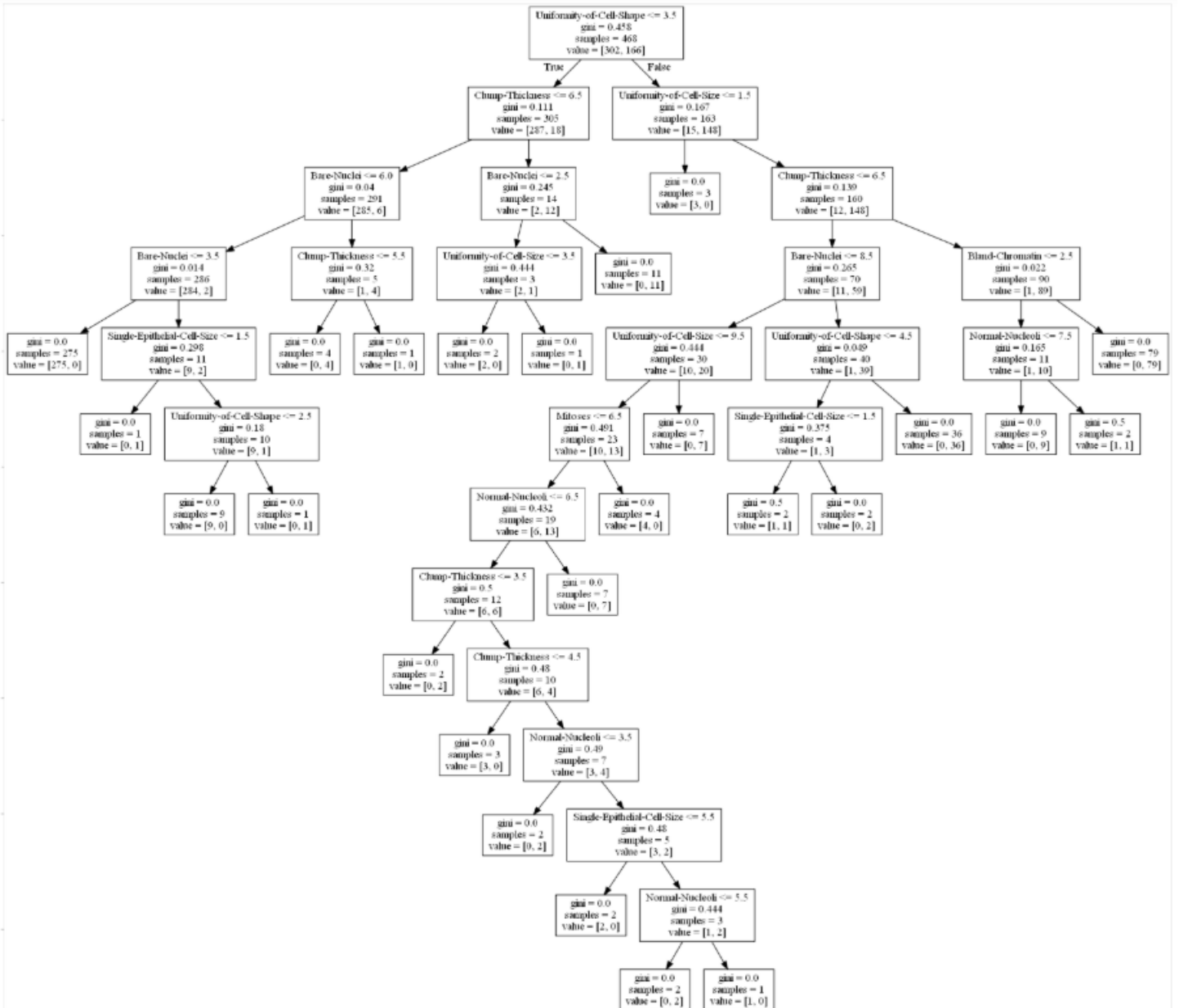
#### 4.2.8. Layman with 30 percent



accuracy using decision tree: 93.5 %  
 Fitting 5 folds for each of 9 candidates, totalling 45 fits  
 [[150 6]  
 [10 65]]  
 CROSS VALIDATION ACCURACY SCORE: 0.9423076923076923

# == CLASSIFICATION REPORT == #				
	precision	recall	f1-score	support
2.0	0.94	0.96	0.95	156
4.0	0.92	0.87	0.89	75
accuracy			0.93	231
macro avg	0.93	0.91	0.92	231
weighted avg	0.93	0.93	0.93	231

#### 4.2.9. Layman with 50 percent



```

accuracy using decision tree: 93.5 %
Fitting 5 folds for each of 9 candidates, totalling 45 fits
[[149  7]
 [ 4 71]]
CROSS VALIDATION ACCURACY SCORE: 0.9423076923076923

```

```

# === CLASSIFICATION REPORT === #
precision    recall  f1-score   support

2.0         0.97     0.96     0.96     156
4.0         0.91     0.95     0.93     75

accuracy          0.95     231
macro avg         0.94     0.95     0.95     231
weighted avg      0.95     0.95     0.95     231

```

## 5. Conclusions

### Conclusion 1:

From looking at the data, we have learned that eliminating parameters with a great info gain can still lead us to good results, even to better results. We have learned that from trying to generate trees without the highest info gain parameter repeatedly. That is because different combinations of parameters can sometimes give more robust models (like the case of the mushrooms). Moreover, not all parameters are independent; sometimes, one variable is affected by another variable, and eliminating one of them results in a weaker model.

### Conclusion 2:

We have also discovered that the shortest tree is not necessarily the best one, and sometimes a longer, less elegant tree is more accurate. We can see that in the cancer dataset results, where an amateur with 30 percent of mistakes had a tree of 9 levels, and accuracy of 96.5%, and an amateur with 10 percent received a tree with 10 levels but an accuracy of 97.4% (see sections 5.2.3 and 5.2.4).

### Conclusion 3:

Table 3 below summarizes the results of both datasets for each level of noise and for each type of decision-maker. For example, the cell located on the fourth row and the third column contains the value of 19. this means that a layman with a noise level of 50% obtained 19 false negative cases.

(the records marked in green are the relatively better results for that expertise level).

Table 3: Results summery

FN Rate	Mushrooms			Cancer		
Case	Amateur(FN)	Layman(FN)	Expert(FN)	Amateur(FN)	Layman(FN)	Expert(FN)
10%	1	5		2	6	
30%	0	11		4	6	
50%	0	19		3	7	
Baseline	0	11	0	4	7	5
Accuracy	Mushrooms			Cancer		
Case	Amateur	Layman	Expert	Amateur	Layman	Expert
10%	99.8%	96.4%		97.4%	96.5%	
30%	99.8%	92.9%		96.5%	93.5%	
50%	99.9%	90.4%		96.5%	93.5%	
Baseline	99.7%	98.7%	100%	95.7%	95.2%	97.8%

In interesting phenomenon occurs when the amateur noise level switched from 10 to 30 and 50 percent. The ID3 algorithm chooses to ignore the noisy variables and build a robust model based on variables with less info gain level. This tree results with minimal FN as the baseline. For example, the mushrooms amateur baseline is 0.

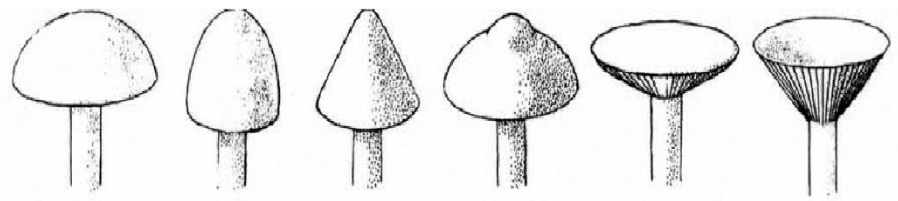
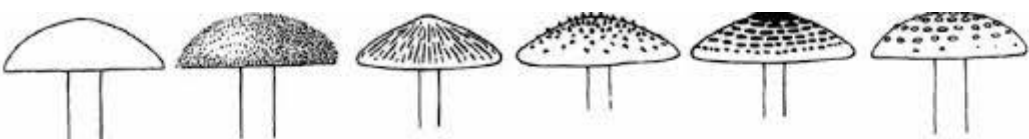

We can see that in both datasets, the amateur got better results when they chose to gamble on their unknown parameters. In all 3 cases, the results were better than the baseline, especially in terms of model accuracy.



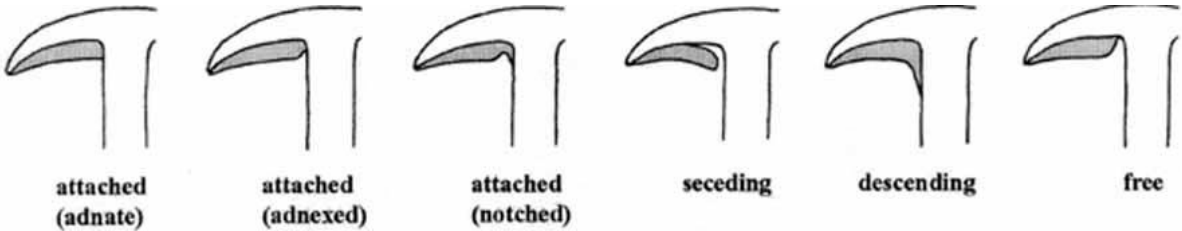
In the layman's case, however, the models of 30 and 50 percent showed results worse than the layman's baseline. So, the layman should rely only on the knowledge in his/her hands unless their error rate is smaller than 10 percent on the amateur's parameters and 30 percent on the experts.

I think the results of the experiment are rather impressive, I believe my assumption before this experiment was that one should either "trust their gut" or base on their knowledge solely. But this experiment has proved that, to some extent, it is better to make an educated guess. In the future, I might be interested in testing this theory in other domains.

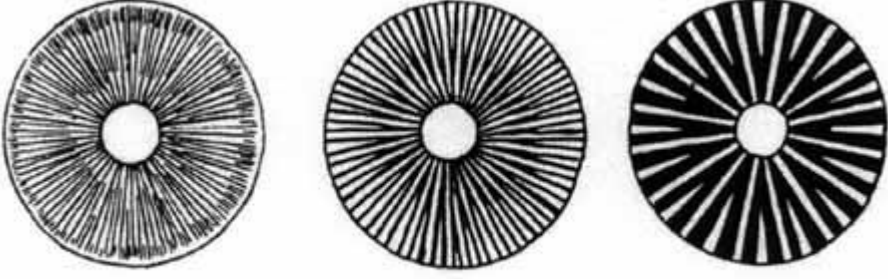


## 6. Appendix 1 – Mushrooms Parameters List



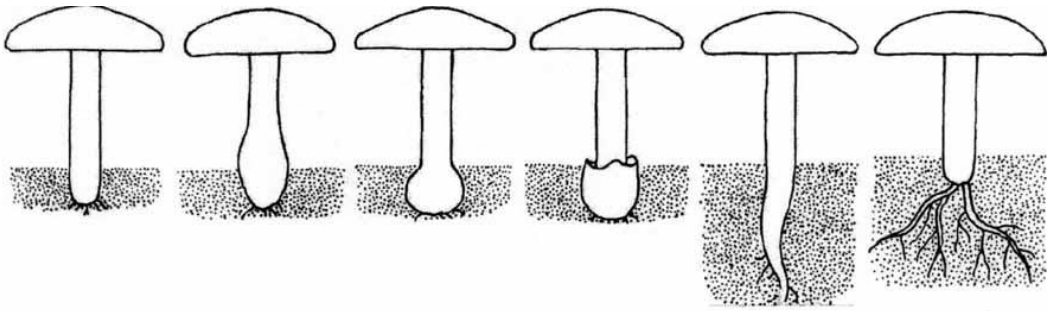
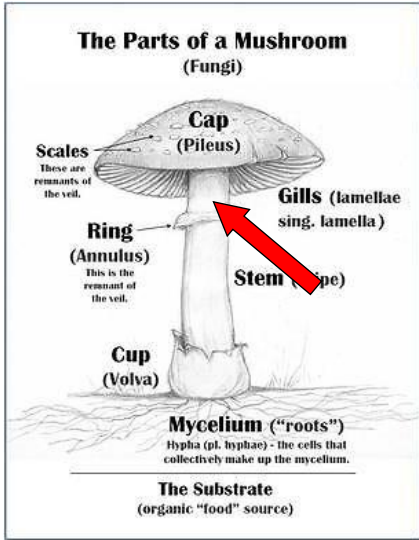
1. cap-shape: bell=b, conical=c, convex=x, flat=f, knobbed=k, sunken=s
2. cap-surface: fibrous=f, grooves=g, scaly=y, smooth=s
3. cap-color: brown=n, buff=b, cinnamon=c, gray=g, green=r, pink=p, purple=u, red=e, white=w, yellow=y
4. bruises: bruises=t, no=f
5. odor: almond=a, anise=l, creosote=c, fishy=y, 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, white=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

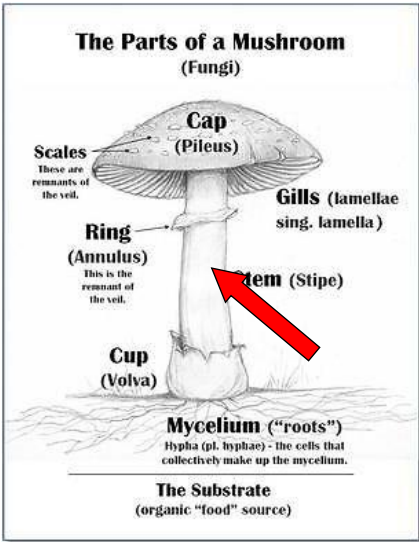
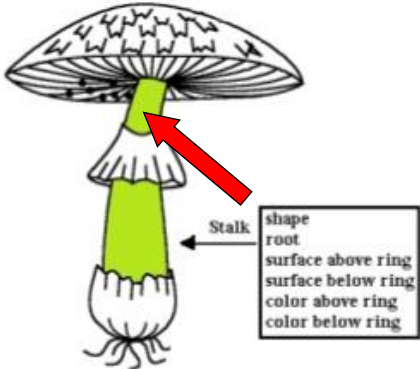
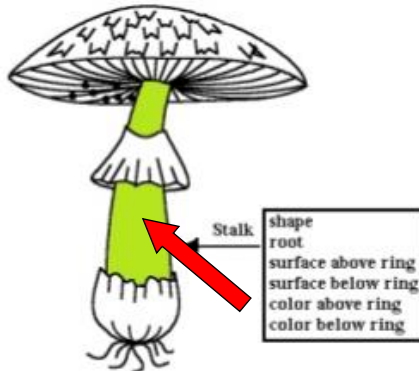
No.	Name	Details
1	cap-shape	 <p>Convex      Bell-Shaped      Conical      Knobbed      Flat      Sunken</p> <p>The Mushroom head shape, not the easiest to recognize since sometimes differences between variations is very small.(for example convex and bell-shaped).</p>
2	cap-surface	 <p>smooth      velvety      hairy or fibrous      raised scales      flat scales      patches</p> <p>The surface of the cap, quite difficult to determine. Differences between types can be very small(for example velvety and hairy)</p>
3	cap-color	 <p>PORCINI      BOLETUS LUTEUS      AGARIC HONEY      RUSSULE      YELLOW BOLETUS</p> <p>BROWN CAP      FLY-AGARIC      MOREL      OYSTER MUSHROOM      BLEWIT</p> <p>CHAMPIGNON      CHANTERELLE      ORANGE CAP      TRUFFLE      TOADSTOOL</p>

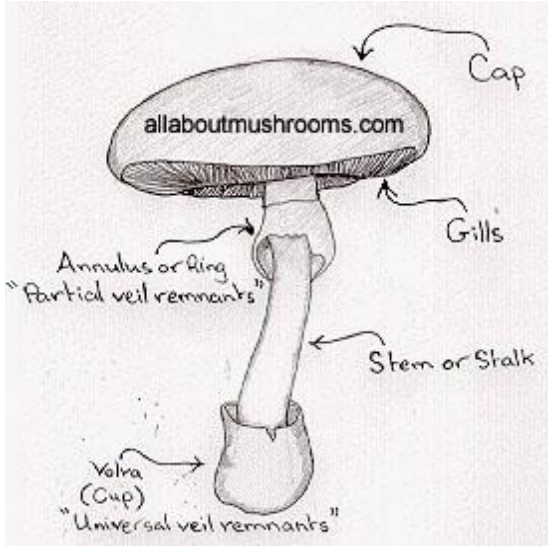

		Cap color – easy to recognize since colors are universal and differ from each other.
4	bruises	 <p>Determines whether the mushroom changes color when cut. Can be relatively difficult to determine.</p>
5	odor	 <p>Mushroom smell. Can be very difficult to determine, sense of smell can be sometimes subjective.</p>
6	gill-attachment	 <p>The way the gill part is connected to the mushroom. Can be slightly difficult to classify, differences between types are very small.</p>

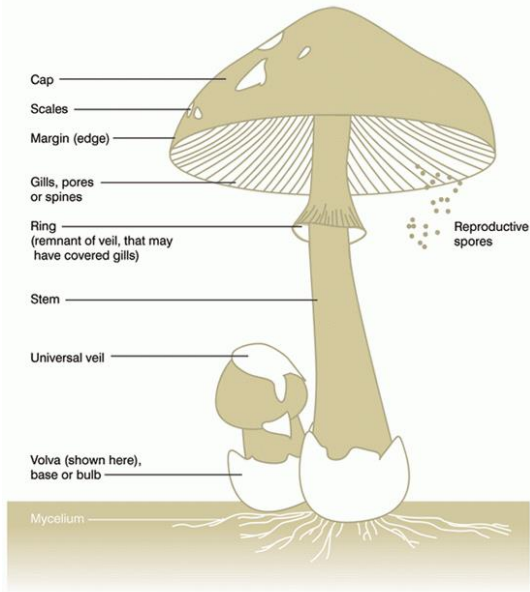
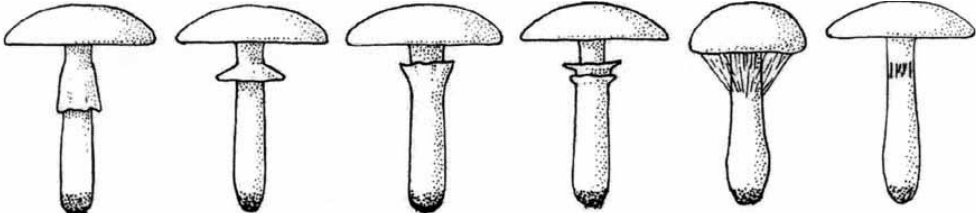



7	gill-spacing	<div data-bbox="370 195 1253 541">  <p>The diagrams show three circular cross-sections of a mushroom head. The first, labeled 'crowded', has many thin, closely packed gills. The second, labeled 'close', has fewer, more widely spaced gills. The third, labeled 'distant', has the fewest and most widely spaced gills.</p> <p><b>crowded</b>                      <b>close</b>                      <b>distant</b></p> </div> <p data-bbox="365 590 1528 657">Relatively easy since spacing is relative and can be measured by anyone, and differences between options are quite noticeable.</p>
8	gill-size	<div data-bbox="383 758 742 1115">  </div> <p data-bbox="764 1016 1502 1083">The gill size – weather they are broad or narrow. Measuring is relative in this case and quite easy to notice.</p>
9	gill-color	<div data-bbox="365 1266 883 1652">  </div> <p data-bbox="906 1482 1437 1509">Color of gill – easy since colors are universal.</p>

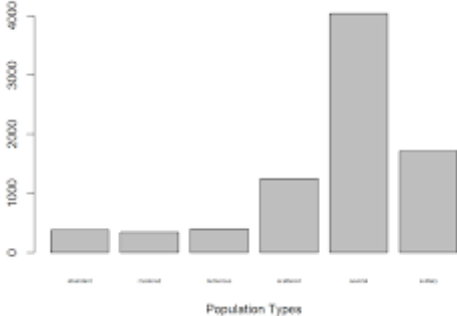

10	stalk-shape	<div data-bbox="365 191 915 558">  </div> <div data-bbox="998 205 1494 558">  </div> <p data-bbox="365 583 1357 617">Stalk shape – only two options – enlarging or tapering, relatively easy to determine.</p>
11	stalk-root	<div data-bbox="380 772 1421 1134">  <div data-bbox="418 1102 1409 1134"> <span>equal</span> <span>club shaped</span> <span>bulbous</span> <span>with cup (volva)</span> <span>rooting</span> <span>with rhizoids</span> </div> </div> <p data-bbox="365 1157 1520 1222">The root of the mushroom – can be a little difficult to see, some options are much like others and differences are small.</p>
12	stalk-surface-above-ring	<p data-bbox="365 1335 574 1369">(Like cap surface)</p> <div data-bbox="1042 1306 1458 1843">  <p data-bbox="1117 1335 1386 1381"><b>The Parts of a Mushroom</b> (Fungi)</p> <p data-bbox="1073 1409 1442 1772"> <b>Cap (Pileus)</b>  <b>Scales</b> These are remnants of the veil.  <b>Ring (Annulus)</b> This is the remnant of the veil.  <b>Cup (Volva)</b>  <b>Gills (lamellae)</b> sing. lamella  <b>Stem (Stipe)</b>  <b>Mycelium ("roots")</b> Hypha (pl. hyphae) - the cells that collectively make up the mycelium.  <b>The Substrate</b> (organic "food" source) </p> </div>

13	stalk- surface- below-ring	(like cap surface)	
14	stalk-color- above-ring	Colors are universal and therefore easy to recognize	
15	stalk-color- below-ring	Colors are universal and therefore easy to recognize	

16	veil-type	<p>Quite difficult to determine, partially because sometimes the veil can fall and one must know the type of mushroom in order to know.</p>	
17	veil-color	<p>Colors are universal and therefore easy to recognize</p>	

18	ring-number	<p>The part which connects the veil to the stalk. Easy to determine since there can only be 1,2 or 0. One can just count.</p> 
19	ring-type	 <p><b>pendant      flaring      sheathing      double      cobwebby      ring zone</b></p> <p>Slightly difficult to determine since options are very similar, and some require a little knowledge in the possible options besides “facing up/down”</p>
20	spore-print-color	 <p>the color of dust which comes out if the gill. Can be spotted easily.</p>



21	population	<div><p>Mushroom Population Distribution</p><table border="1"><thead><tr><th>Population Types</th><th>Count</th></tr></thead><tbody><tr><td>abundant</td><td>400</td></tr><tr><td>common</td><td>400</td></tr><tr><td>rare</td><td>400</td></tr><tr><td>abundant</td><td>1200</td></tr><tr><td>common</td><td>4000</td></tr><tr><td>abundant</td><td>1800</td></tr></tbody></table></div> <p>How common is the mushroom worldwide? Very difficult to tell without being an expert.</p>	Population Types	Count	abundant	400	common	400	rare	400	abundant	1200	common	4000	abundant	1800
Population Types	Count															
abundant	400															
common	400															
rare	400															
abundant	1200															
common	4000															
abundant	1800															
22	habitat	<div><p>The location where the mushroom grows. Very easy, one can look around him/her.</p><div></div><p>just</p></div>														

## 7. Appendix 2 – Cancer Tumor Parameters List

Clump Thickness	(1 – 10)
Uniformity of Cell Size	(1 – 10)
Uniformity of Cell Shape	(1 – 10)
Marginal Adhesion	(1 – 10)
Single Epithelial Cell Size	(1 – 10)
Bare Nuclei	(1 – 10)
Bland Chromatin	(1 – 10)
Normal Nucleoli	(1 – 10)
Mitoses	(1 – 10)
Class	(2 – Benign, 4 – malignant)

Attribute	Details
Clump Thickness	Benign cells tend to be grouped in monolayers, while cancerous cells are often grouped in multilayers.
Uniformity of Cell Size	Cancer cells tend to vary in size and shape. That is why these parameters are valuable in determining whether the cells are cancerous or not.
Uniformity of Cell Shape	Cancer cells tend to vary in size and shape. That is why these parameters are valuable in determining whether the cells are cancerous or not.
Marginal Adhesion	Normal cells tend to stick together. Cancer cells tends to lose this ability. So loss of adhesion is a sign of malignancy.
Single Epithelial Cell Size	Is related to the uniformity mentioned above. Epithelial cells that are significantly enlarged may be a malignant cell.
Bare Nuclei	This is a term used for nuclei that is not surrounded by cytoplasm (the rest of the cell). Those are typically seen in benign tumors.
Bland Chromatin	Describes a uniform "texture" of the nucleus seen in benign cells. In cancer cells the chromatin tends to be more coarse.
Normal Nucleoli	Nucleoli are small structures seen in the nucleus. In normal cells the nucleolus is usually very small if visible at all. In cancer cells the nucleoli become more prominent, and sometimes there are more of them.
Mitoses	Mitosis is the process by which genetic matter gets identically replicated many times over. Since cancer is caused by a damage or mutation to cellular DNA, mitosis plays an active role in spreading cancer in the body by making exact copies of these damaged and mutated cellular genetic materials.

## Bibliography

- Basics of Wild Harvested Mushroom Identification
- field\_book\_of\_common\_gilled\_mushrooms\_1928