'To eat, or not to eat: that is the question.' An application of machine learning techniques to distinguish edible from poisonous mushrooms with 100% accuracy

Jordi Sanjuán Belda

13/12/2020

Introduction

The present work aims to develop a prediction algorithm that allows to guess, from certain physical characteristics, if a mushrooms are edible or not.

To do so, the first thing that will be done is to become familiar with the data in the following section (Data overview: a first descriptive approach). Then, a classification tree and a random forest model will be applied, the results will be discussed and, finally, some conclusions will be drawn.

This work is done as part of the ninth and final course of the Professional Certificate in Data Science at Harvard University. Its goal is to put into practice, in an autonomous way, all the knowledge and skills acquired during the previous eight courses of the program.

Data overview: a first descriptive approach

The database chosen to carry out this work includes a sample of 8,124 mushrooms catalogued as edible or as poisonous. In turn, a series of 22 characteristics (visual, olfactory, shape, touch ...) of each mushroom are collected.

There is no single, clear or even less simple rule to distinguish whether a mushroom is edible or not from a few features, and that is where the machine learning techniques to recognize more complex hidden rules will come in.

The variables present in the database (all of them categorical and not ordinal) and their codification are the following:

class: edible = e, poisonous = p

cap.shape: bell = b, conical = c, convex = x, flat = f, knobbed = k, sunken = s

cap.surface: fibrous = f, grooves = g, scaly = y, smooth = s

cap.color: brown = n, buff = b, cinnamon = c, gray = g, green = r, pink = p, purple = u, red = e, white = w, yellow = y

bruises: bruises = t, no = f

odor: almond = a, anise = l, creosote = c, fishy = y, foul = f, musty = m, none = n, pungent = p, spicy = s

gill.attachment: attached = a, descending = d, free = f, notched = n

gill.spacing: close = c, crowded = w, distant = d

gill.size: broad = b, narrow = n

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

stalk.shape: enlarging = e, tapering = t

stalk.root: bulbous = b, club = c, cup = u, equal = e, rhizomorphs = z, rooted = r, missing = ?

stalk.surface.above.ring: fibrous = f, scaly = y, silky = k, smooth = s

stalk.surface.below.ring: fibrous = f, scaly = y, silky = k, smooth = s

stalk.color.above.ring: brown = n, buff = b, cinnamon = c, gray = g, orange = o, pink = p, red = e, white = w, yellow = y

stalk.color.below.ring: brown = n, buff = b, cinnamon = c, gray = g, orange = o, pink = p, red = e, white = w, yellow = y

veil.type: partial = p, universal = u

veil.color: brown = n, orange = o, white = w, yellow = y

ring.number: none = n, one = o, two = t

ring.type: cobwebby = c, evanescent = e, flaring = f, large = l, none = n, pendant = p, sheathing = s, zone = z

spore.print.color: black = k, brown = n, buff = b, chocolate = h, green = r, orange = o, purple = u, white = w, yellow = y

population: abundant = a, clustered = c, numerous = n, scattered = s, several = v, solitary = y

habitat: grasses = g, leaves = l, meadows = m, paths = p, urban = u, waste = w, woods = d

More information about the data can be obtained in the following links:

https://www.kaggle.com/uciml/mushroom-classification

https://archive.ics.uci.edu/ml/datasets/Mushroom

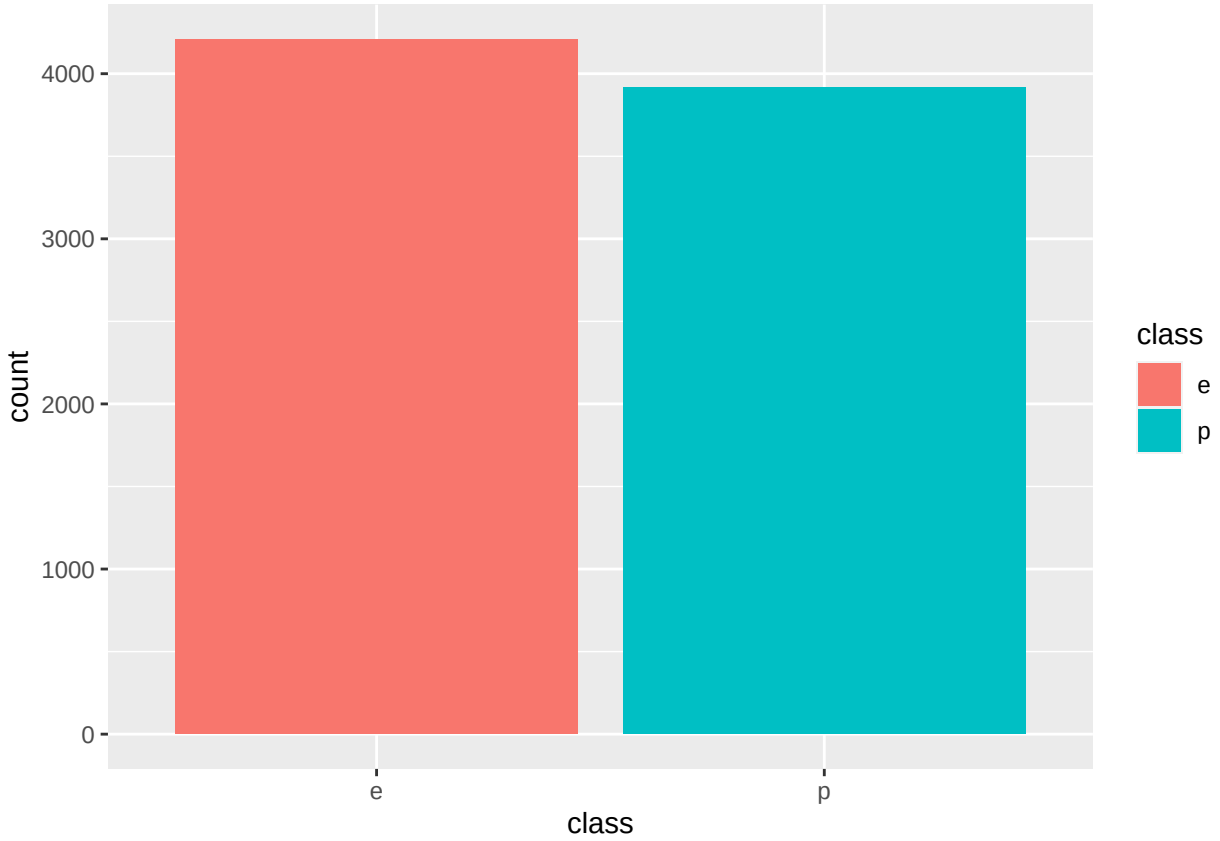
The first step, after converting the variables into factors to be processed correctly, is to make a first general observation of the content of all the variables.

```
##  class    cap.shape cap.surface   cap.color    bruises       odor
##  e:4208   b: 452    f:2320      n      :2284   f:4748   n      :3528
##  p:3916   c:   4    g:   4      g      :1840   t:3376   f      :2160
##           f:3152    s:2556      e      :1500            s      : 576
##           k: 828    y:3244      y      :1072            y      : 576
##           s:  32                w      :1040            a      : 400
##           x:3656                b      : 168            l      : 400
##                                 (Other): 220            (Other): 484
##  gill.attachment gill.spacing gill.size   gill.color   stalk.shape stalk.root
##  a: 210          c:6812       b:5612    b      :1728   e:3516      ?:2480
##  f:7914          w:1312       n:2512    p      :1492   t:4608      b:3776
##                                         w      :1202               c: 556
##                                         n      :1048               e:1120
##                                         g      : 752               r: 192
##                                         h      : 732
##                                         (Other):1170
##  stalk.surface.above.ring stalk.surface.below.ring stalk.color.above.ring
##  f: 552                   f: 600                   w      :4464
##  k:2372                   k:2304                   p      :1872
```

```
##  s:5176                   s:4936                   g      : 576
##  y:  24                   y: 284                   n      : 448
##                                                    b      : 432
##                                                    o      : 192
##                                                    (Other): 140
##  stalk.color.below.ring veil.type veil.color ring.number ring.type
##  w      :4384           p:8124    n:  96     n:  36      e:2776
##  p      :1872                     o:  96     o:7488      f:  48
##  g      : 576                     w:7924     t: 600      l:1296
##  n      : 512                     y:   8                 n:  36
##  b      : 432                                            p:3968
##  o      : 192
##  (Other): 156
##  spore.print.color population habitat
##  w      :2388      a: 384     d:3148
##  n      :1968      c: 340     g:2148
##  k      :1872      n: 400     l: 832
##  h      :1632      s:1248     m: 292
##  r      :  72      v:4040     p:1144
##  b      :  48      y:1712     u: 368
##  (Other): 144                 w: 192
```

The first thing that comes out of this is that the variable *veil.type* has only one value (p for partial, there is no mushroom catalogued with the u for universal). Therefore, it does not provide any information and will be dropped from now on.

Another important fact is to observe what proportion of mushrooms in the sample are edible and what proportion are poisonous. The following plot shows that they are in fact quite balanced. 51.8% of the mushrooms in the database are edible, which means that there will be no problems of low prevalence.

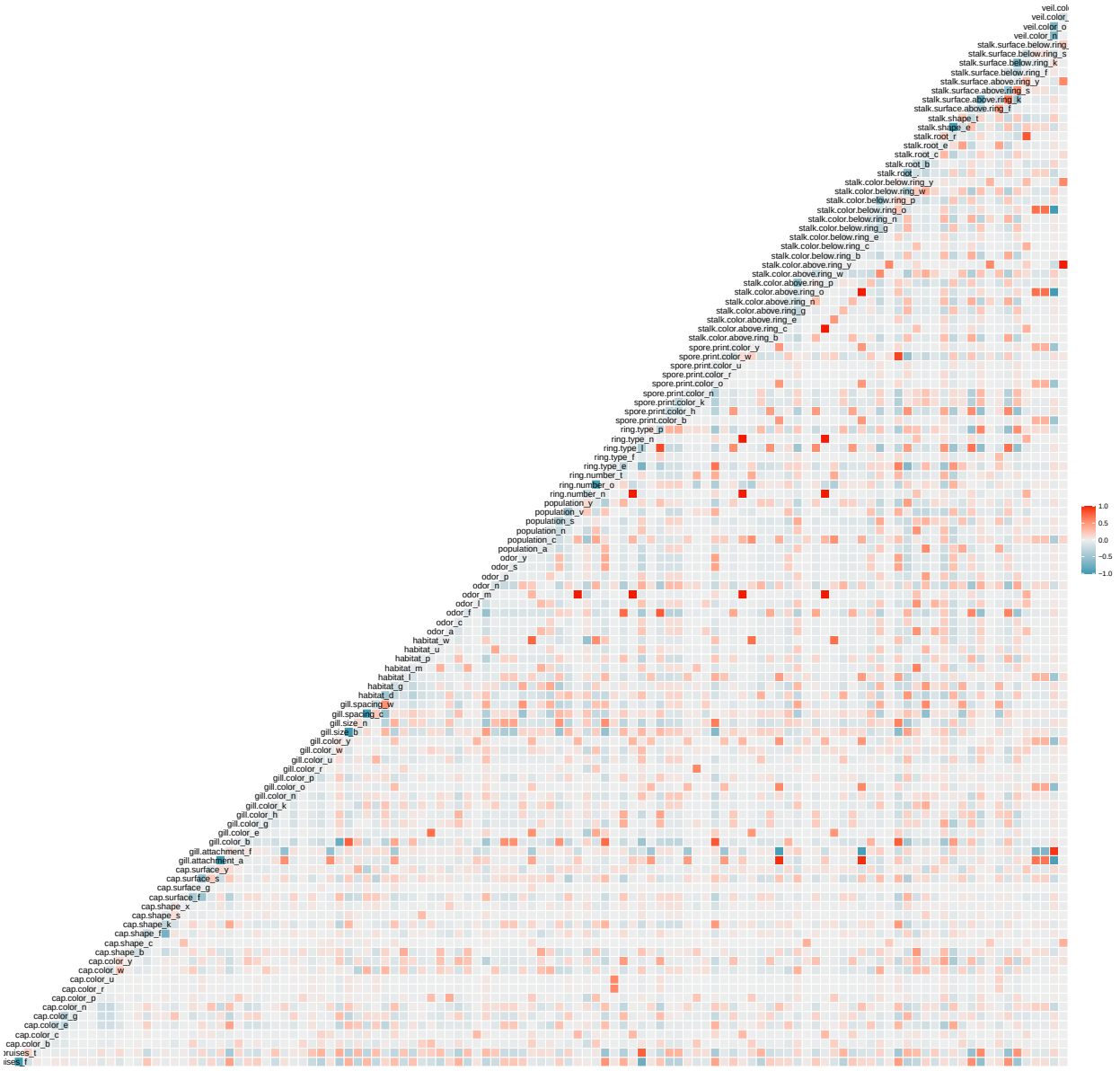


It is also logical to observe graphically the distribution according to the other variables, distinguishing between edible and poisonous mushrooms.


count
class
e
p
cap.shape
b c f k s x
cap.surface
f g s y
cap.color
b c e g n p r u w y
bruises
f t
odor
a c f l m n p s y
gill.attachment
a f
gill.spacing
c w
gill.size
b n
gill.color
b e g h k n o p r u w y
stalk.shape
e t
stalk.root
? b c e r
stalk.surface.above.ring
f k s y
stalk.surface.below.ring
f k s y
stalk.color.above.ring
b c e g n o p w y
stalk.color.below.ring
b c e g n o p w y
veil.color
n o w y
ring.number
n o t
ring.type
e f l n p
spore.print.color
b h k n o r u w y
population
a c n s v y
habitat
d g l m p u w

The reader can appreciate that there are some differences. Edible mushrooms are more common among those with certain traits, and vice versa. For example, most unscented mushrooms ($odor$ = n) are edible, while those with a foul odor are usually not; among mushrooms with bruises, most are edible, while those without bruises are mostly poisonous; if the spore print color is black or brown, the mushroom is usually edible, and conversely if it is chocolate or white; and so on. These characteristics are the ones that will help the models to classify the mushrooms.

Now it will be checked whether there are also correlations between some of these features. Do they usually appear together? Separately? Or is it completely random? This can be analyzed through a correlation matrix, after making some conversions, since the data are categorical, and not numerical.



Indeed, it can be seen that there are elements that relate to others, either in a positive or a negative way.

Is it possible, from all this information, to generate a prediction algorithm to classify mushrooms as edible or poisonous based on their features?

Method

This is a classification problem based on categorical variables, for which it has been considered that classification trees and random forest would be appropriate techniques. To do this, the database is first broken down into a training set with 80% of the observations and a test set with the remaining 20%.

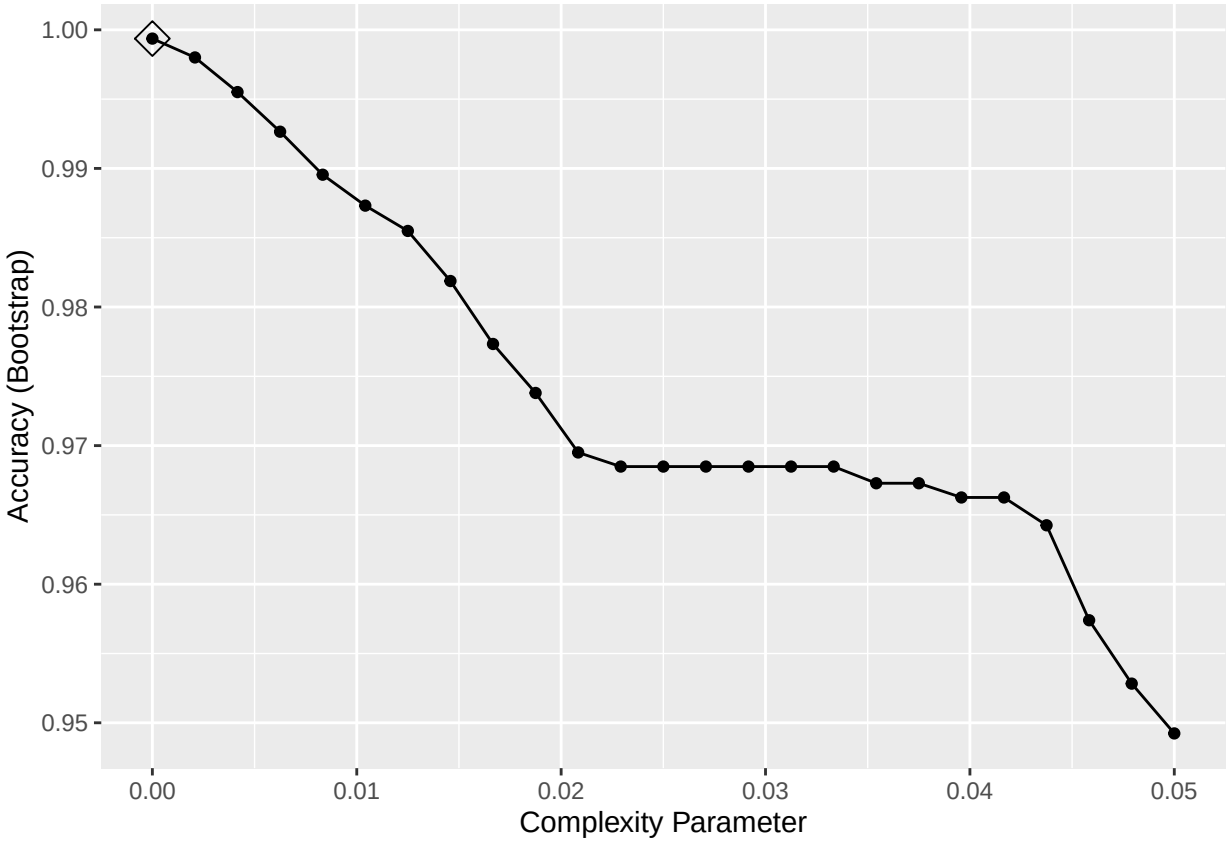
Two different models will be tested: classification tree and random forest.

Classification Tree

The first approach considered is that of the decision tree, which is basically a flow chart or yes or no questions by partitioning the predictors. The predictions at the ends are referred as nodes. Since the outcomes in this case are categorical, we call this a classification tree.

To apply this method, 25 complexity parameters are tested, from 0 to 0.05. The one that gets a better result in terms of precision is 0. With higher values, the accuracy decreases (as can be seen in the next plot), so this is the selected parameter.

```
##   cp
## 1  0
```



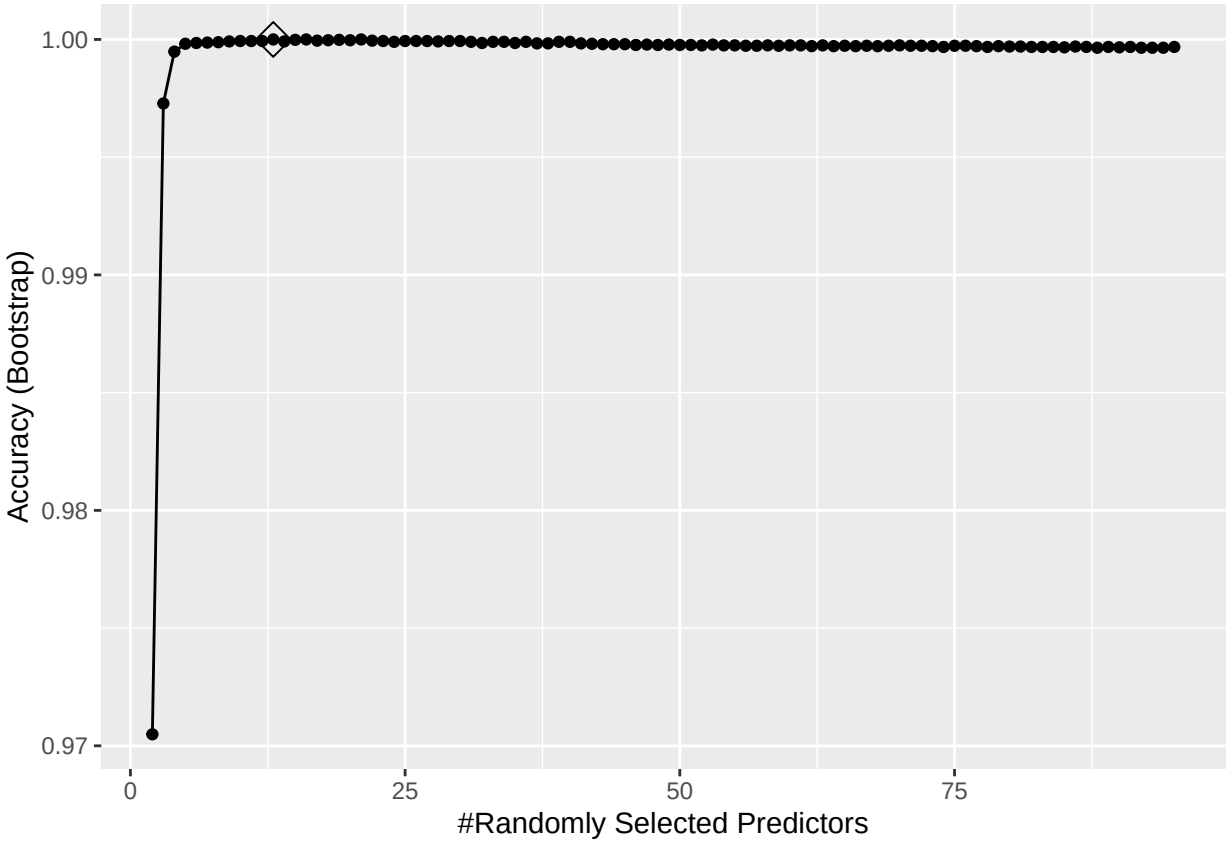
Random Forest

Now, a random forest model is applied. Since there are only two possible outcomes and both have a noticeable presence in the data, there is no point in tuning the minimum node size. This is not the case for the number

of selected predictors. These can be up to 95 (all combinations of the 21 explanatory variables with all their levels), so all options from 2 to 95 are tested. The best adjustment in terms of accuracy occurs with 12 predictors.

```
##    predFixed minNode
## 12        13       2
```

Nevertheless, as can be seen in the following plot, the results are quite similar between 8 and 40 predictors (approximately), close to 100%. From that point on, it declines but very slightly.



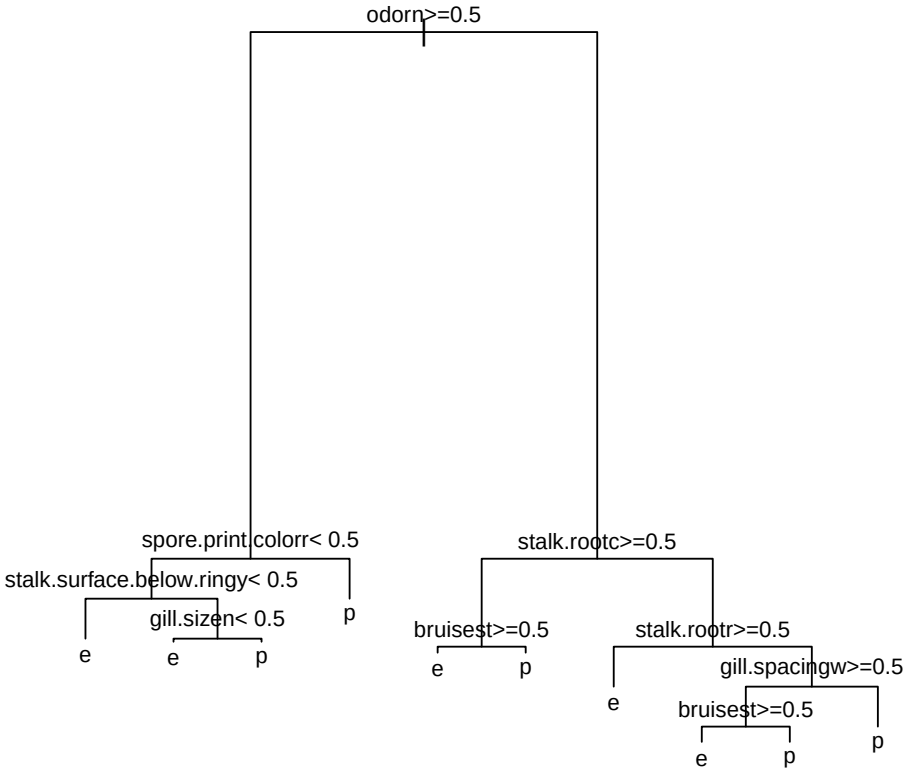
Results

The models have not yet been applied in the test set, so they may be over-adjusted. To check their actual validity they should be tested on a data sample that has not been used for training.

Classification Tree

Before testing it, it is convenient to observe the classification tree that has been generated in the previous section. It is a tree with 10 end nodes. The $>$ or $<$ 0.5 of the plot should be interpreted as a question. If this characteristic (variable name + code letter) is fulfilled, the value is 1; if it is not fulfilled, it is 0. With this information, we answer each question and go to the right if the answer is yes, and to the left if the answer is no. For example, in the first splitting rule, the tree asks if the mushroom has no odor (*odorn* = n value of the *odor* variable, i.e. no odor). If it does not have it (affirmative answer), it follows to the right, if

it does smell (negative answer), it follows to the left. One of the main advantages of decision trees is their easy interpretation.



Now it is time to validate the model with the test set. The complete results of the confusion matrix are shown below.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   e   p
##          e 842   3
##          p   0 781
##
##                Accuracy : 0.9982
##                  95% CI : (0.9946, 0.9996)
```

```
##     No Information Rate : 0.5178
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9963
##
##  Mcnemar's Test P-Value : 0.2482
##
##             Sensitivity : 1.0000
##             Specificity : 0.9962
##          Pos Pred Value : 0.9964
##          Neg Pred Value : 1.0000
##              Prevalence : 0.5178
##          Detection Rate : 0.5178
##    Detection Prevalence : 0.5197
##       Balanced Accuracy : 0.9981
##
##        'Positive' Class : e
##
```

It can be seen that, although the overall accuracy achieved is really high (99.82%), there are 3 poisonous mushrooms that have been classified as edible. This is very dangerous! In this case, specificity is much more important than sensitivity (which does reach 100%), and a small error like these three can be fatal.

The random forest method is designed to correct some problems of over-adjustment and instability of decision trees. That's why this has been the next step: will the random forest model be able to correct these errors?

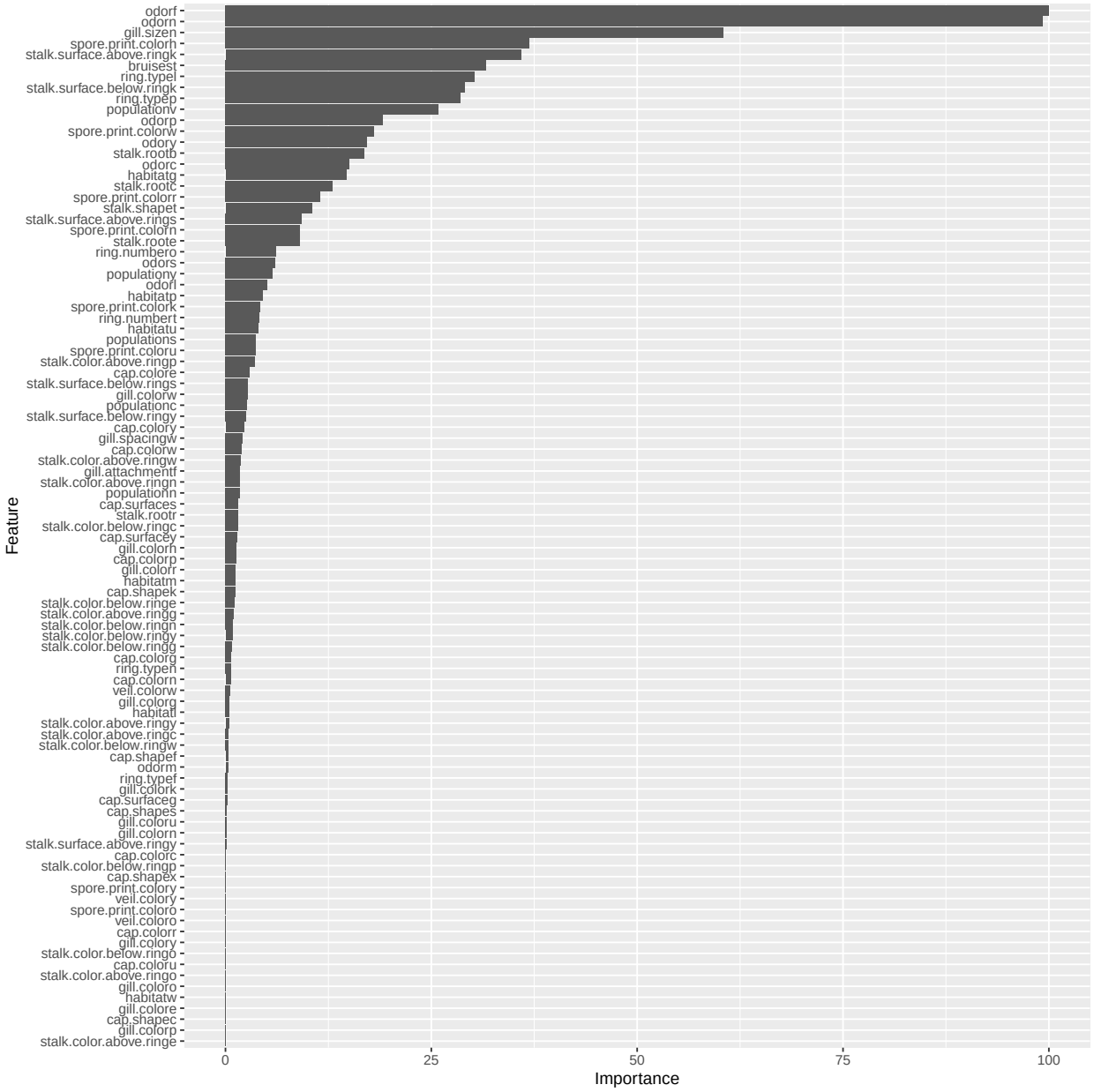
Random Forest

As seen in the previous section, the final random forest model with 12 randomly selected predictors achieves 100% accuracy, although it still needs to be validated in the test set.

Random forest models are not as easily interpreted as decision trees. However, it is possible to know that the most important variables for the classification made by the random forest are the following:

```
## Rborist variable importance
##
##   only 20 most important variables shown (out of 95)
##
##                            Overall
## odorf                      100.000
## odorn                       99.307
## gill.sizen                  60.463
## spore.print.colorh          36.896
## stalk.surface.above.ringk   35.894
## bruisest                    31.635
## ring.typel                  30.276
## stalk.surface.below.ringk   29.059
## ring.typep                  28.543
## populationv                 25.870
## odorp                       19.157
## spore.print.colorw          18.017
## odory                       17.213
## stalk.rootb                 16.837
## odorc                       15.009
```

```
## habitatg                   14.676
## stalk.rootc                13.005
## spore.print.colorr         11.528
## stalk.shapet               10.490
## stalk.surface.above.rings   9.282
```



Finally, it is checked that the model is indeed 100% accurate also in the test set. The complete confusion matrix is now provided and it is observed that, logically, there are no false positives or false negatives, and therefore the sensitivity and specificity are also both 100%.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   e   p
```

```
##          e 842   0
##          p   0 784
##
##                Accuracy : 1
##                  95% CI : (0.9977, 1)
##     No Information Rate : 0.5178
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##             Sensitivity : 1.0000
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 1.0000
##              Prevalence : 0.5178
##          Detection Rate : 0.5178
##    Detection Prevalence : 0.5178
##       Balanced Accuracy : 1.0000
##
##        'Positive' Class : e
##
```

Conclusion

In this work, two classification algorithms have been developed to distinguish edible from poisonous mushrooms based on a series of characteristics. Specifically, a classification tree and a random forest have been applied, the latter achieving 100% accuracy. The final model is, therefore, difficult to improve.

Nevertheless, this does not mean that the classification tree, with an accuracy of 99.82%, should be discarded altogether. Before the decision to eat or not a mushroom, the advantage that the classification tree offers is its easy and intuitive interpretation, so following the different splitting rules it would be possible to reach a conclusion by ourselves. Though, accepting a minimum risk of 0.18%.

However, it is worth making some observations. This is a prototypical problem in which specificity (proportion of negative outcomes, i.e. poisonous mushrooms, correctly identified) is much more important than sensitivity (proportion of positive outcomes, i.e. edible mushrooms, correctly identified). In other words, it is more dangerous to eat a poisonous mushroom than to throw away an edible one.

This has not been a problem because it has been possible to achieve 100% accuracy (and therefore specificity). But if it had not been possible to reach that level of accuracy, it would have been necessary to optimize the model, not from the overall accuracy, but by seeking the highest possible specificity. Or, at least, weigh it more than the sensitivity. Here again, it would have been possible to seek a better fit of the model by exploring other techniques and combining them in the form of an ensemble.

These aspects will be taken into account for future research. But for the moment it is already possible to eat mushrooms without excessive worries. Bon appetit!