STATISTICAL DATA MINING

FINAL GROUP PROJECT

BY:
GAURAV KULKARNI
SADHASHIV SHARMA

UNIVERSITY of SOUTH FLORIDA

PROJECT OBJECTIVE

- This is a mushroom dataset where several of mushroom's features have been provided to us like its odour, whether its bruised or not, habitat and several other features are given. Mushrooms comes in different varieties. Some may be edible while others may not be. Consumption of poisonous mushrooms may lead to serious illness.

- The purpose of this project is to build predictive models which determines which mushrooms are edible and which are not and to know what are the main predictors in deciding the response variables.

DESCRIPTIVE STATISTICS

- Summary()
- By looking at the summary , we can conclude that all the variables are categorical and there is one variable stalk.root which has 2480 missing values.

```
PE        cap.shape cap.surface   cap.color     bruises       odor       gill.attachment
e:4208    b: 452    f:2320       n      :2284   f:4748   n      :3528   a: 210
p:3916    c:   4    g:   4       g      :1840   t:3376   f      :2160   f:7914
          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.spacing gill.size   gill.color   stalk.shape stalk.root stalk.surface.above.ring
c:6812       b:5612    b      :1728   e:3516      ?:2480     f: 552
w:1312       n:2512    p      :1492   t:4608      b:3776     k:2372
                       w      :1202               c: 556     s:5176
                       n      :1048               e:1120     y:  24
                       g      : 752               r: 192
                       h      : 732
                       (Other):1170
stalk.surface.below.ring stalk.color.above.ring stalk.color.below.ring veil.type veil.color
f: 600                   w      :4464           w      :4384           p:8124     n:  96
k:2304                   p      :1872           p      :1872                      o:  96
s:4936                   g      : 576           g      : 576                      w:7924
y: 284                   n      : 448           n      : 512                      y:   8
                         b      : 432           b      : 432
                         o      : 192           o      : 192
                         (Other): 140           (Other): 156
ring.number ring.type spore.print.color population habitat
n:  36      e:2776    w      :2388      a: 384     d:3148
o:7488      f:  48    n      :1968      c: 340     g:2148
t: 600      l:1296    k      :1872      n: 400     l: 832
            n:  36    h      :1632      s:1248     m: 292
            p:3968    r      :  72      v:4040     p:1144
                      b      :  48      y:1712     u: 368
                      (Other): 144                 w: 192
```

- This provides general structure of our dataset. Factor tells that its categorical and levels tells us the number of categories each variable has respectively. If we closely observe, we find that veil.type has only one category and hence we can eliminate this from our final dataset as this will not be proven useful for our response variable.

```
'data.frame':   8124 obs. of  23 variables:
 $ PE                      : Factor w/ 2 levels "e","p": 2 1 1 2
 $ cap.shape               : Factor w/ 6 levels "b","c","f","k",.
 $ cap.surface             : Factor w/ 4 levels "f","g","s","y":
 $ cap.color               : Factor w/ 10 levels "b","c","e","g",
 $ bruises                 : Factor w/ 2 levels "f","t": 2 2 2 2
 $ odor                    : Factor w/ 9 levels "a","c","f","l",.
 $ gill.attachment         : Factor w/ 2 levels "a","f": 2 2 2 2
 $ gill.spacing            : Factor w/ 2 levels "c","w": 1 1 1 1
 $ gill.size               : Factor w/ 2 levels "b","n": 2 1 1 2
 $ gill.color              : Factor w/ 12 levels "b","e","g","h",
 $ stalk.shape             : Factor w/ 2 levels "e","t": 1 1 1 1
 $ stalk.root              : Factor w/ 5 levels "?","b","c","e",.
 $ stalk.surface.above.ring: Factor w/ 4 levels "f","k","s","y":
 $ stalk.surface.below.ring: Factor w/ 4 levels "f","k","s","y":
 $ stalk.color.above.ring  : Factor w/ 9 levels "b","c","e","g",.
 $ stalk.color.below.ring  : Factor w/ 9 levels "b","c","e","g",.
 $ veil.type               : Factor w/ 1 level "p": 1 1 1 1 1 1 1
 $ veil.color              : Factor w/ 4 levels "n","o","w","y":
 $ ring.number             : Factor w/ 3 levels "n","o","t": 2 2
 $ ring.type               : Factor w/ 5 levels "e","f","l","n",.
 $ spore.print.color       : Factor w/ 9 levels "b","h","k","n",.
 $ population              : Factor w/ 6 levels "a","c","n","s",.
 $ habitat                 : Factor w/ 7 levels "d","g","l","m",.
```

VISUALIZATION IN R

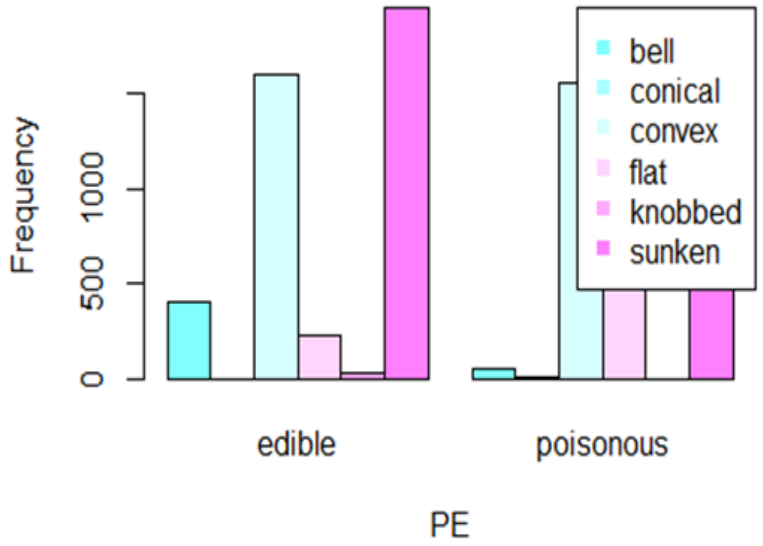
- We have performed Univariate - Bivariate Analysis in R. We plotted Bargraph and separated it via target variable. In short it is kind of both univariate and bivariate analysis.

- #BAR GRAPH FOR MUSHROOM'S CAP SHAPE

```
jointcapShape=CrossTable(capShape,PE,prop.chisq = FALSE)
joint_counts1=jointcapShape$t
barplot(joint_counts,beside=TRUE,col=cm.colors(6),ylab='Frequency',xlab='PE')
legend('topright',c('bell','conical','convex','flat','knobbed','sunken'),pch=15,col=cm.colors(6))
```

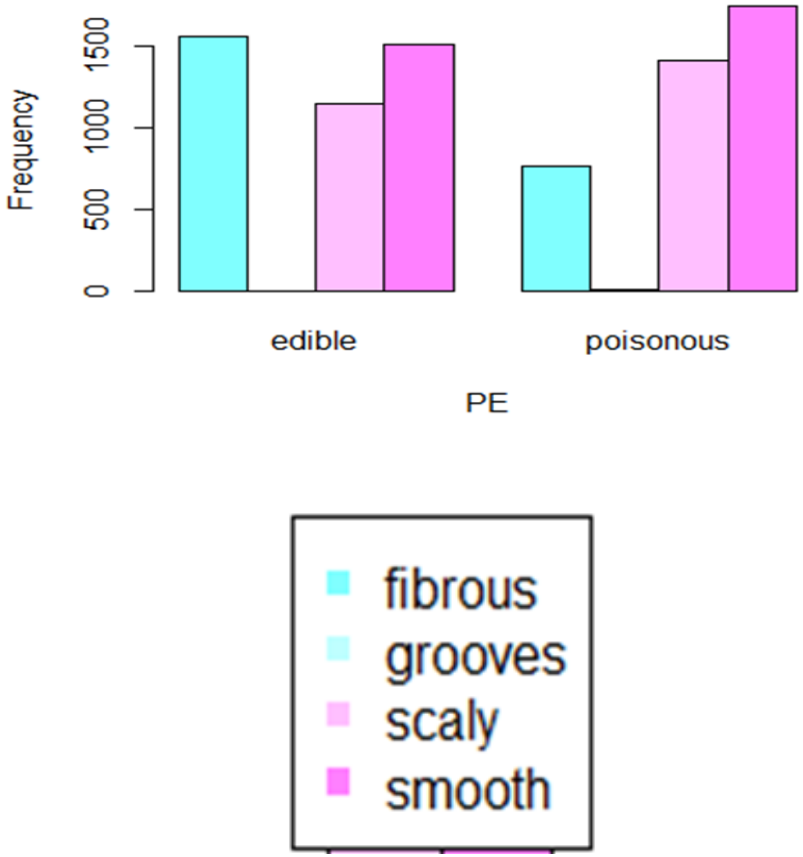
- #BAR GRAPH FOR MUSHROOM'S CAP COLOR

```
jointcapColor=CrossTable(capColor,PE,prop.chisq = FALSE)
joint_counts1=jointcapColor$t
barplot(joint_counts,beside=TRUE,col=cm.colors(6),ylab='Frequency',xlab='PE')
legend('topright',c('brown','buff','cinnamon','gray','green','pink','purple','red','white','yellow'),pch=1
5,col=cm.colors(10))
```



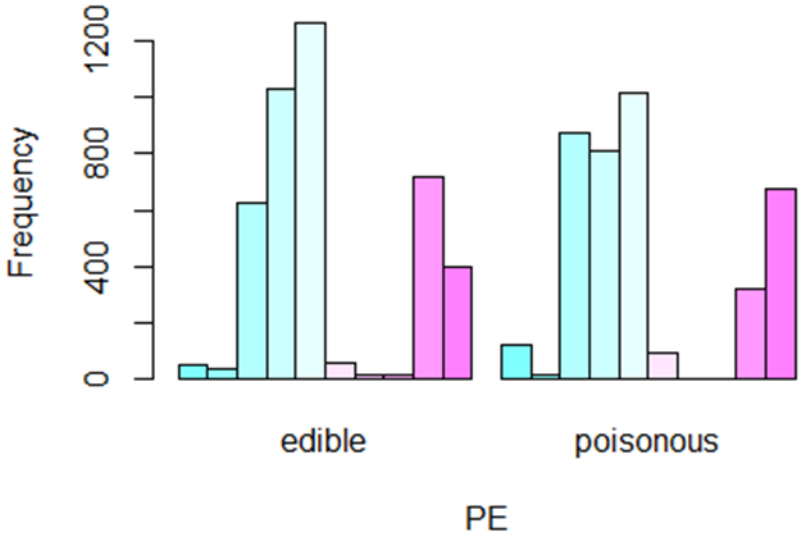
Total Observations in Table: 8124

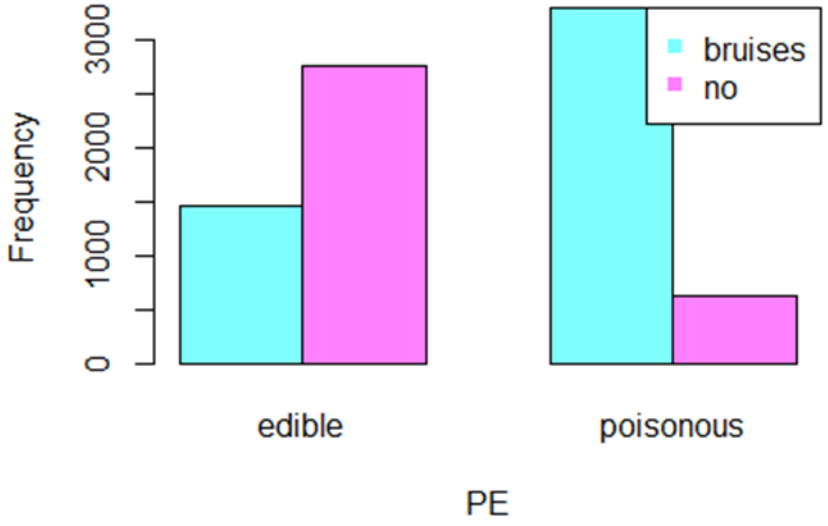
capShape	PE edible	poisonous	Row Total
b	404	48	452
	0.894	0.106	0.056
	0.096	0.012	
	0.050	0.006	
c	0	4	4
	0.000	1.000	0.000
	0.000	0.001	
	0.000	0.000	
f	1596	1556	3152
	0.506	0.494	0.388
	0.379	0.397	
	0.196	0.192	
k	228	600	828
	0.275	0.725	0.102
	0.054	0.153	
	0.028	0.074	
s	32	0	32
	1.000	0.000	0.004
	0.008	0.000	
	0.004	0.000	
x	1948	1708	3656
	0.533	0.467	0.450
	0.463	0.436	
	0.240	0.210	
Column Total	4208	3916	8124
	0.518	0.482	



Total Observations in Table: 8124

capSurface	PE edible	poisonous	Row Total
f	1560	760	2320
	0.672	0.328	0.286
	0.371	0.194	
	0.192	0.094	
g	0	4	4
	0.000	1.000	0.000
	0.000	0.001	
	0.000	0.000	
s	1144	1412	2556
	0.448	0.552	0.315
	0.272	0.361	
	0.141	0.174	
y	1504	1740	3244
	0.464	0.536	0.399
	0.357	0.444	
	0.185	0.214	
Column Total	4208	3916	8124
	0.518	0.482	


Frequency
0
400
800
1200
edible
poisonous
PE


Frequency
0
1000
2000
3000
bruises
no
edible
poisonous
PE

CONCLUSIONS

- Bell cup shape mushrooms are very less poisonous ,whereas conical shape are neither edible nor poisonous.
- Groove surface are neither edible nor poisonous or we don’t have enough evidence to prove that.
- If the mushrooms have bruises, then they have high probability of being poisonous.
- In this way, bar graphs are highly useful in concluding such factors.

FREQUENCIES AND RELATIVE FREQUENCIES

- The relative frequencies can be also found out if we want to know the frequency and percentage of each categories. cbind function has been used to bind together both frequency and relative frequency
- Relative Frequency is really useful if we want to know the percentage and number of missing values.
- We have 51% of edible mushrooms in our dataset and 49% of poisonous mushrooms as seen below.
- From the second figure, we can concluded that 30% of the values are missing in the variable Stalk Root

```
> PE<-factor(mushroomData$PE)
> freqPE=table(PE)
> relfreqPE=table(PE)/8124
> cbind(freqPE,relfreqPE)
  freqPE relfreqPE
e   4208 0.5179714
p   3916 0.4820286
```

```
> stalkRoot<-factor(mushroomData$stalk.root)
> freqStalkroot=table(stalkRoot)
> relfreqStalkroot=table(stalkRoot)/8124
> cbind(freqStalkroot,relfreqStalkroot)
  freqStalkroot relfreqStalkroot
?          2480       0.30526834
b          3776       0.46479567
c           556       0.06843919
e          1120       0.13786312
r           192       0.02363368
```

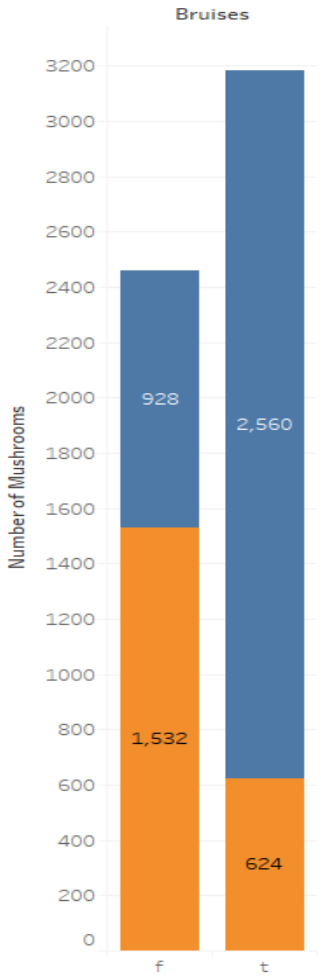
DATA CLEANING AND PREPARATION

- The 30% of missing values in variable stalk root has been taken care by following approaches:
- Dataset1-Deletion of the rows completely which contain missing values
- Dataset2-Imputing it with Mode which is ‘b’ in this case since it is the mode.
- Observation: One crucial observation which was found is that the stalk root details is completely missing in the case where mushroom’s cap color is either r or u.
- We have built Naïve Bayes, RPART and Random Forest predictive models based on above 2 different datasets. Based upon the accuracies and confusion matrices, we have concluded which model fits well for Mushroom dataset.

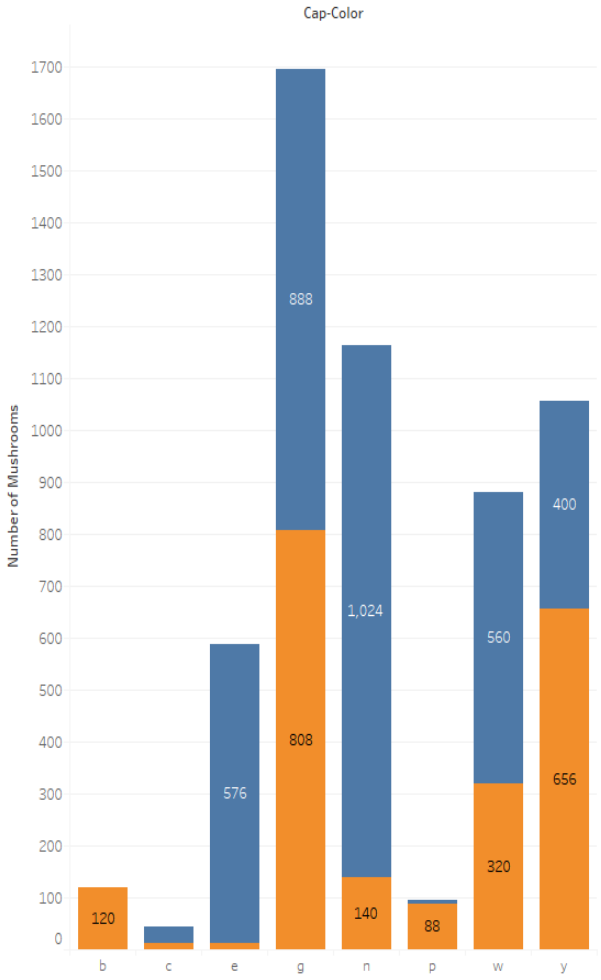
VISUALIZATION IN TABLEAU

- Visualizations here shows number of poisonous and edible mushrooms in each category.
- Our Bar Graphs show that Blue as EDIBLE and Orange as POISONOUS.

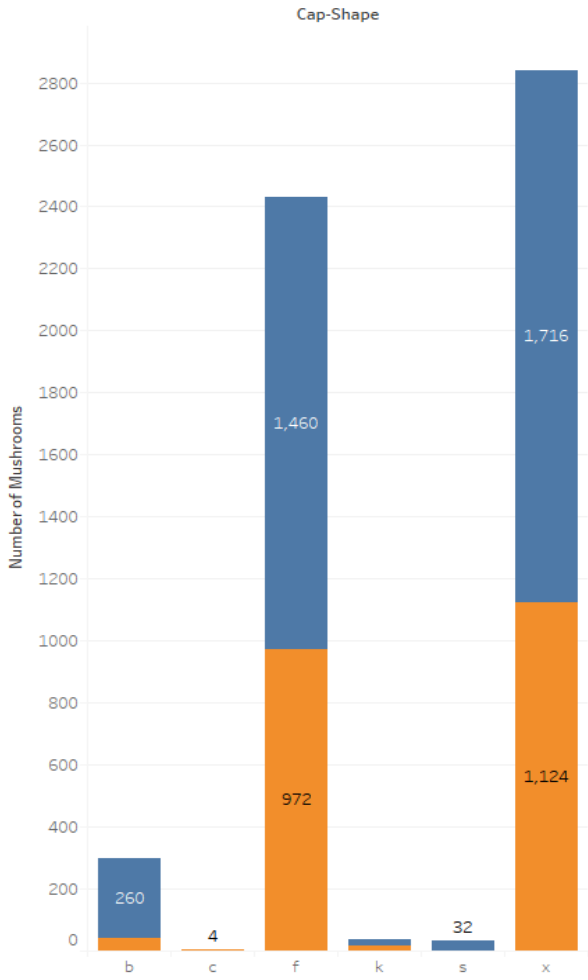
Bruises



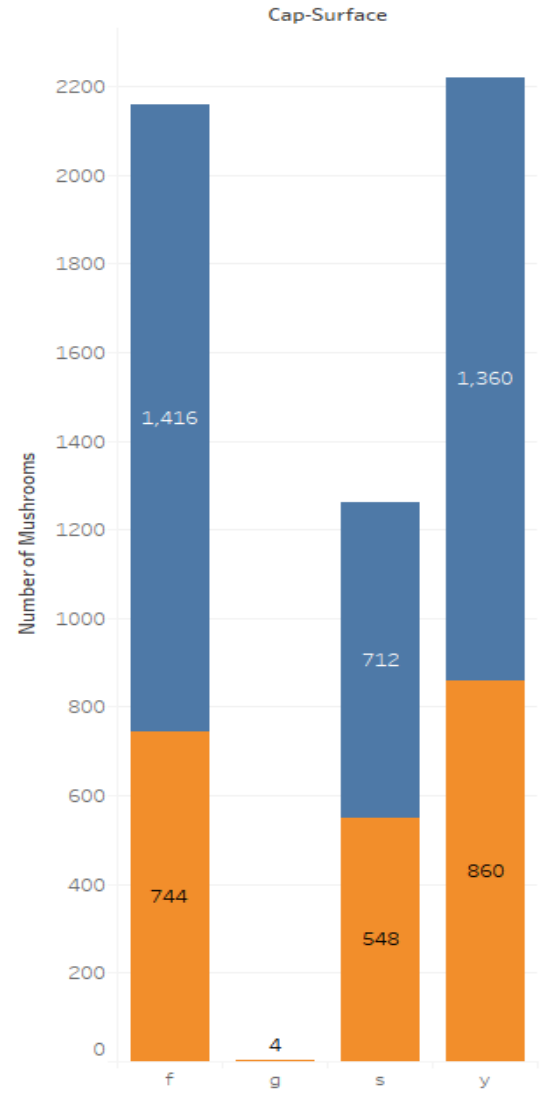
Cap Color



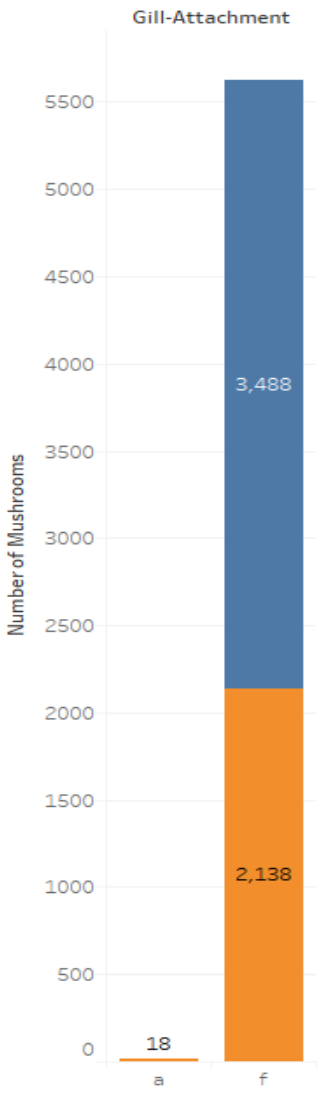
Cap Shape



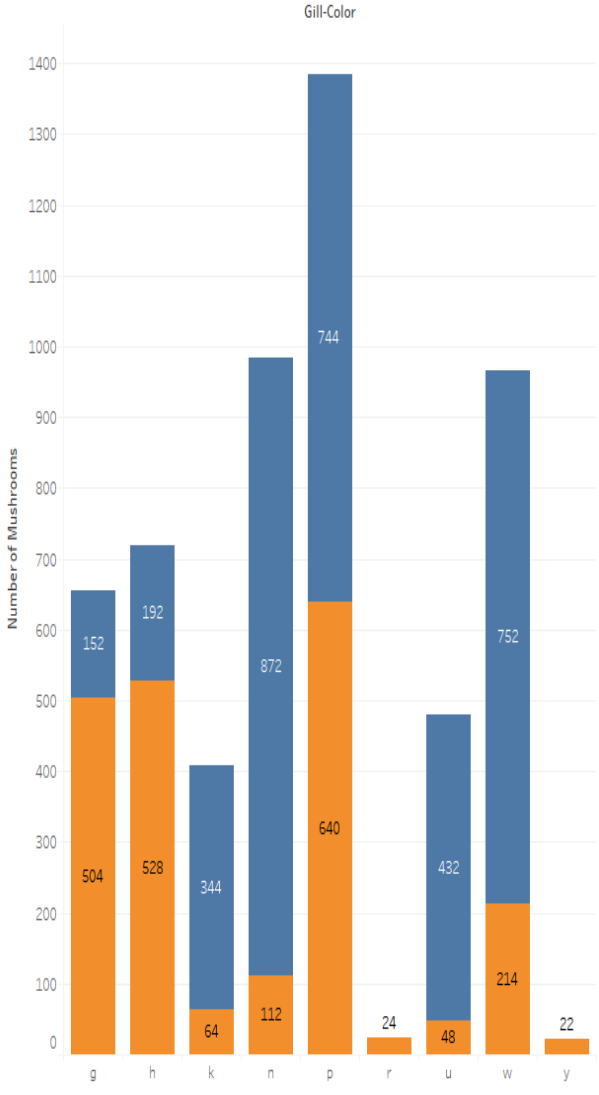
Cap Surface



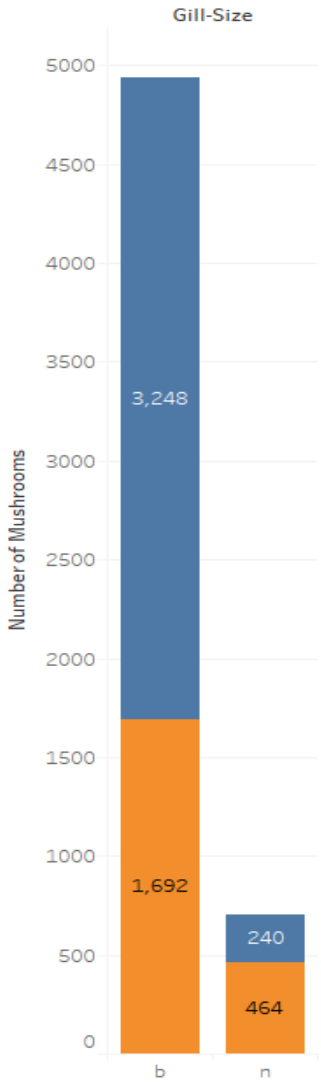
Gill Attachment



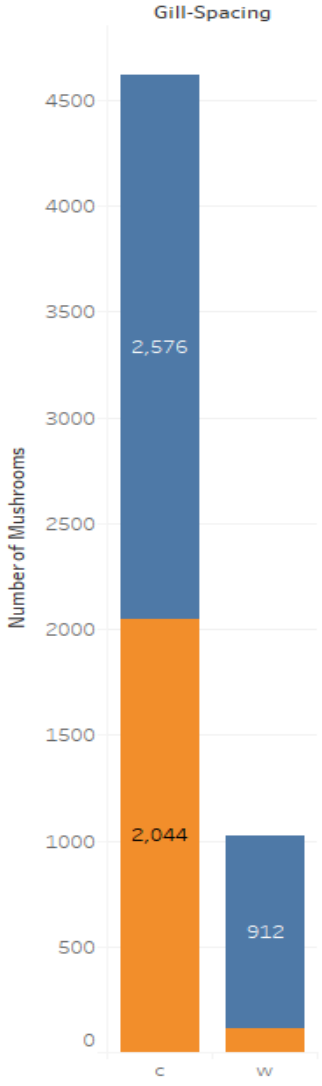
Gill Color



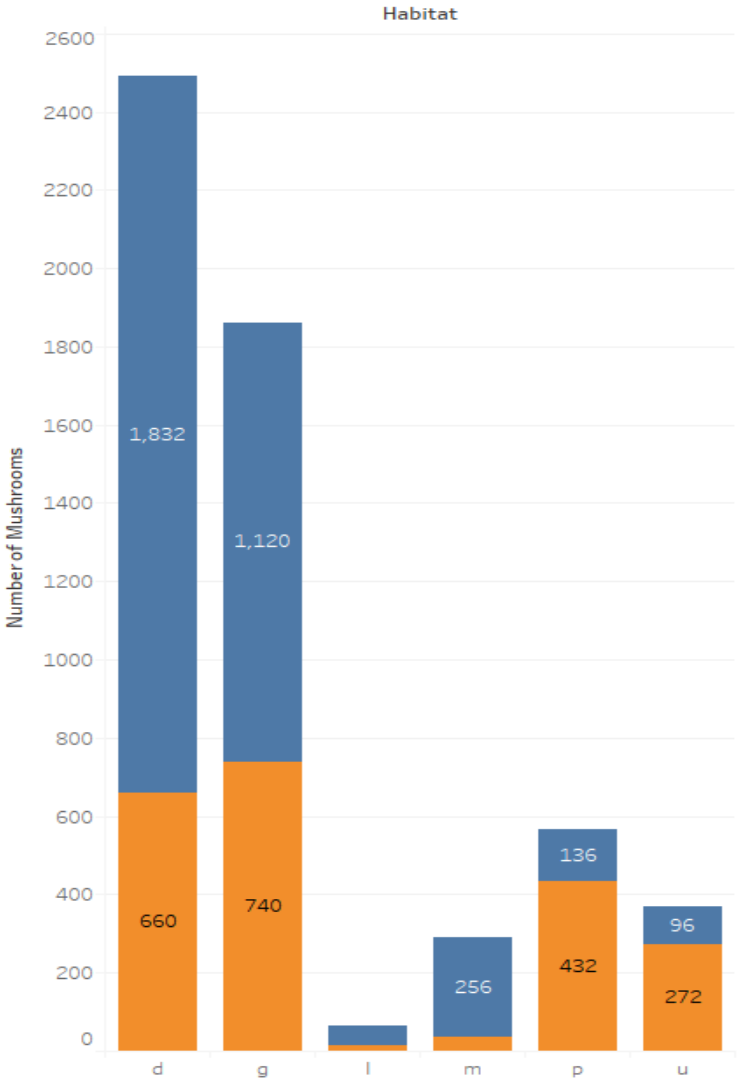
Gill Size



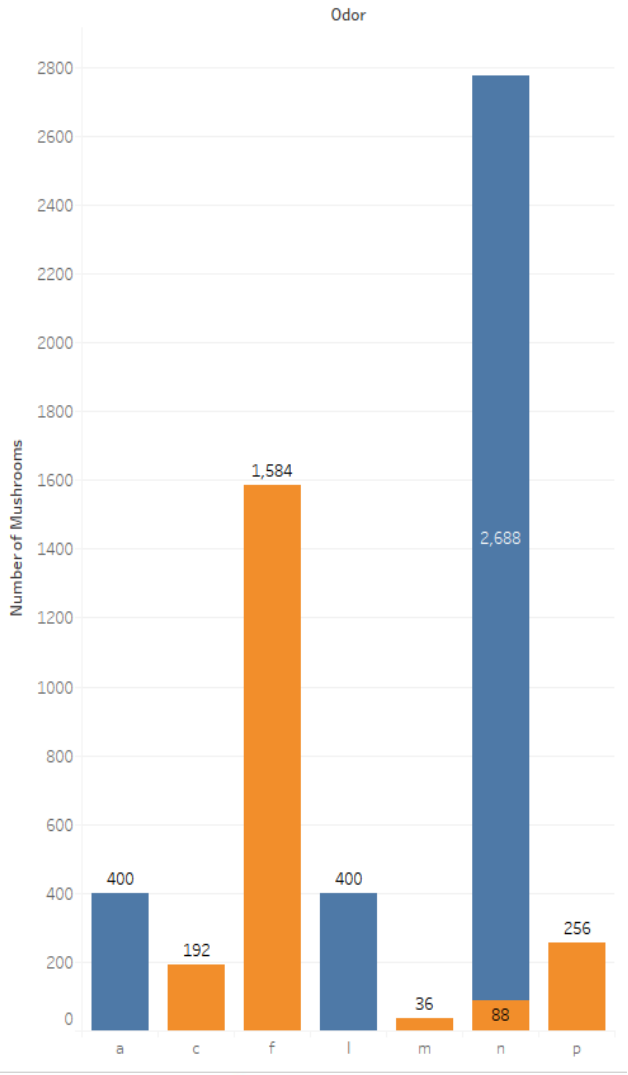
Gill Spacing



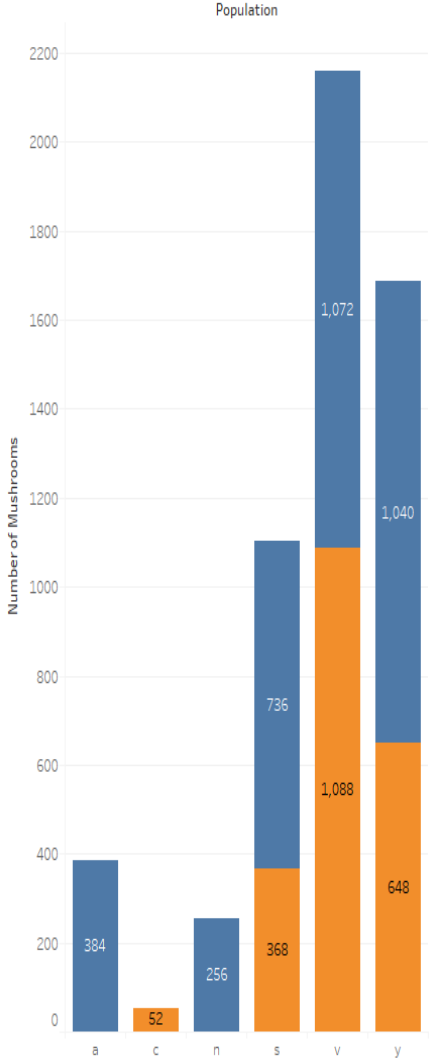
Habitat



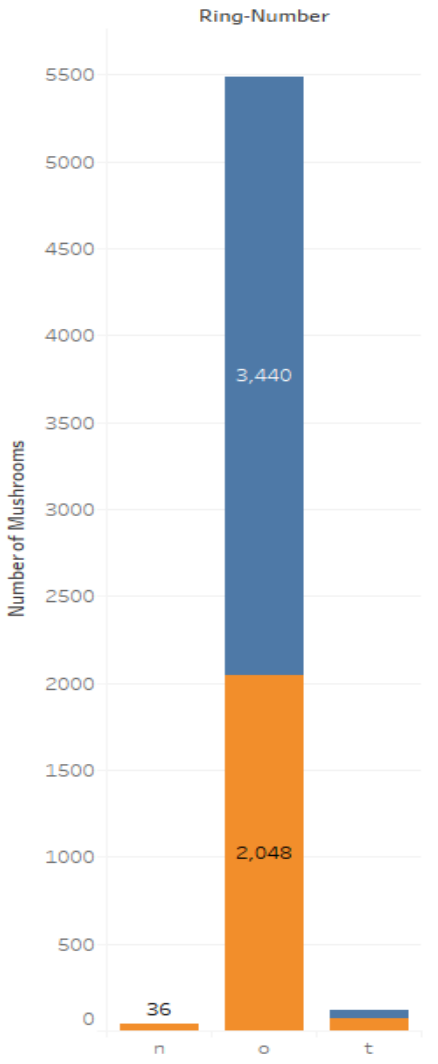
Odor



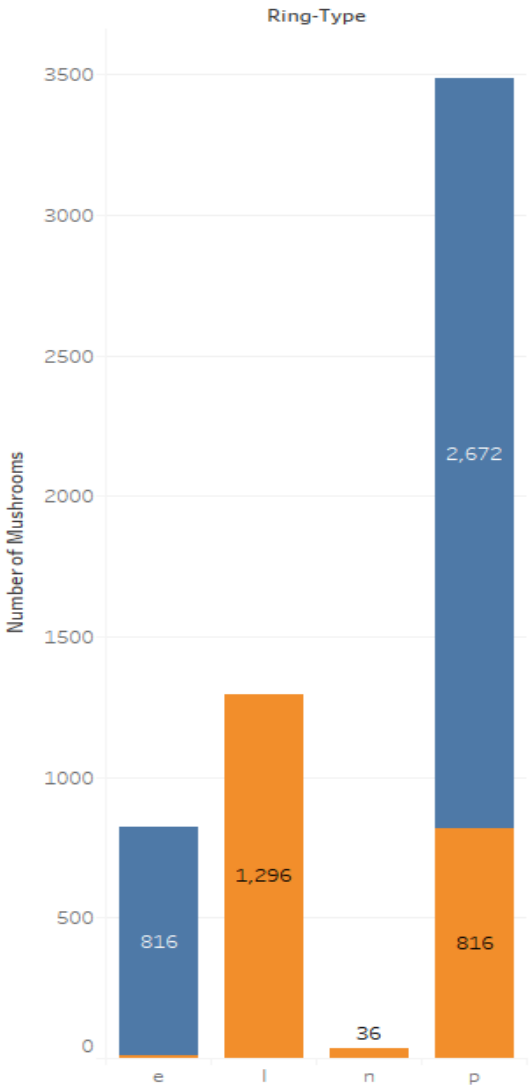
Population



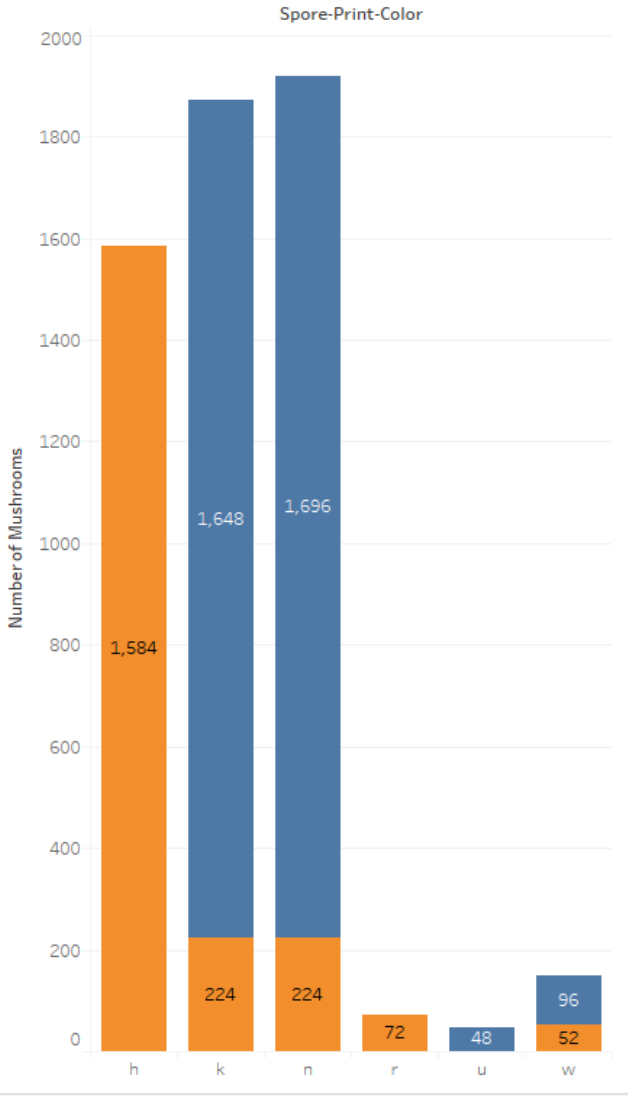
Ring Number



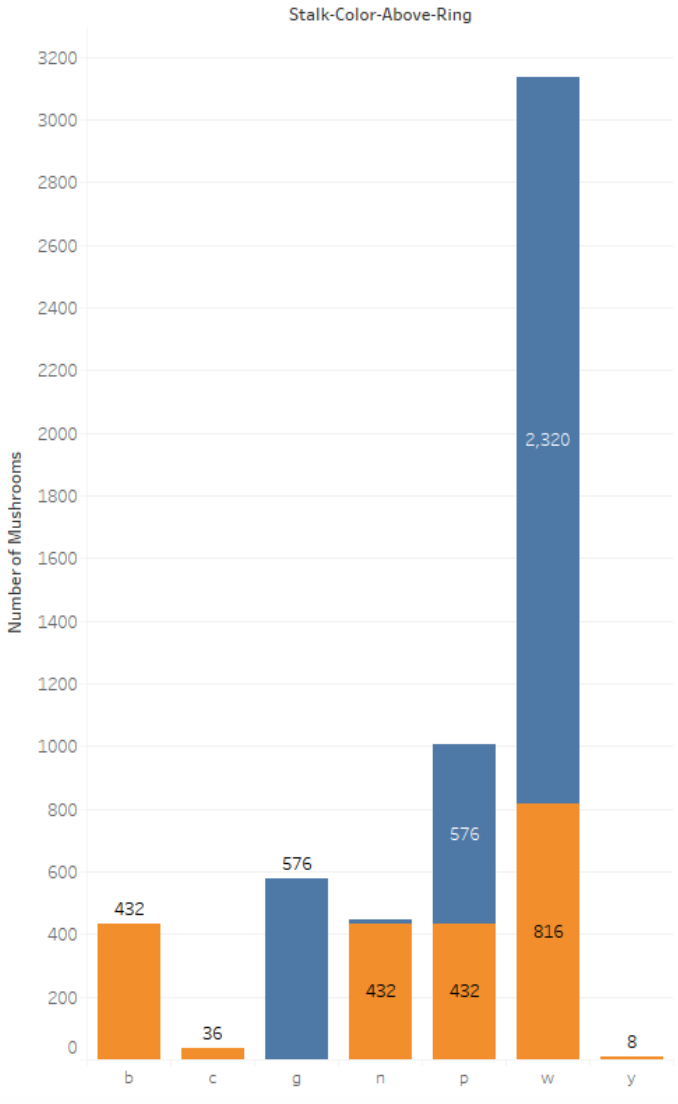
Ring Type



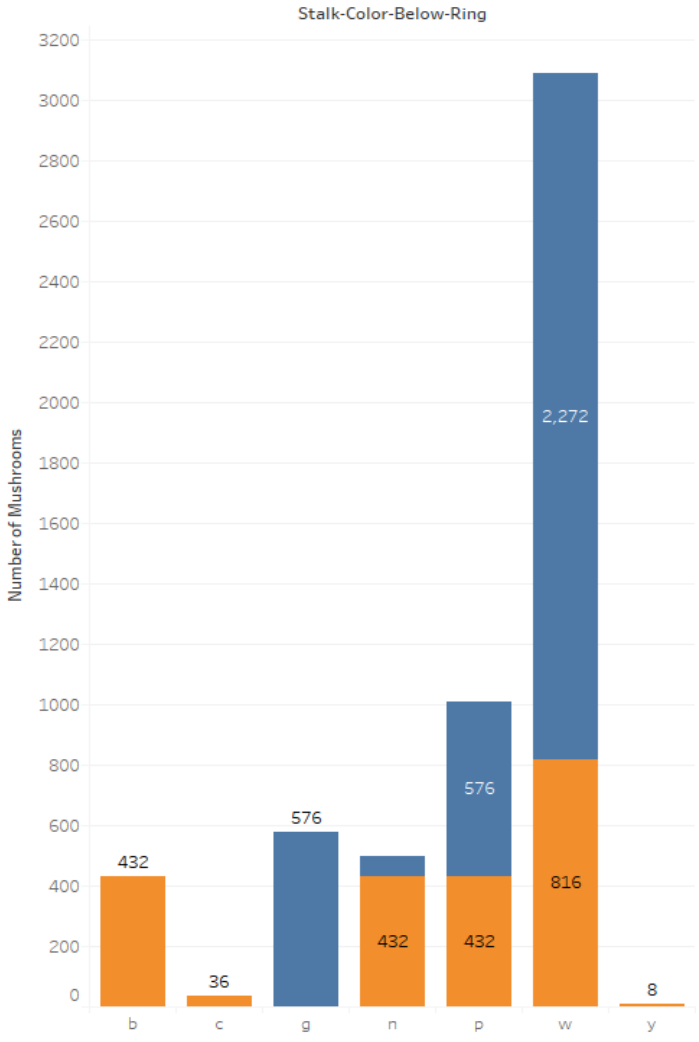
Spore Print Color



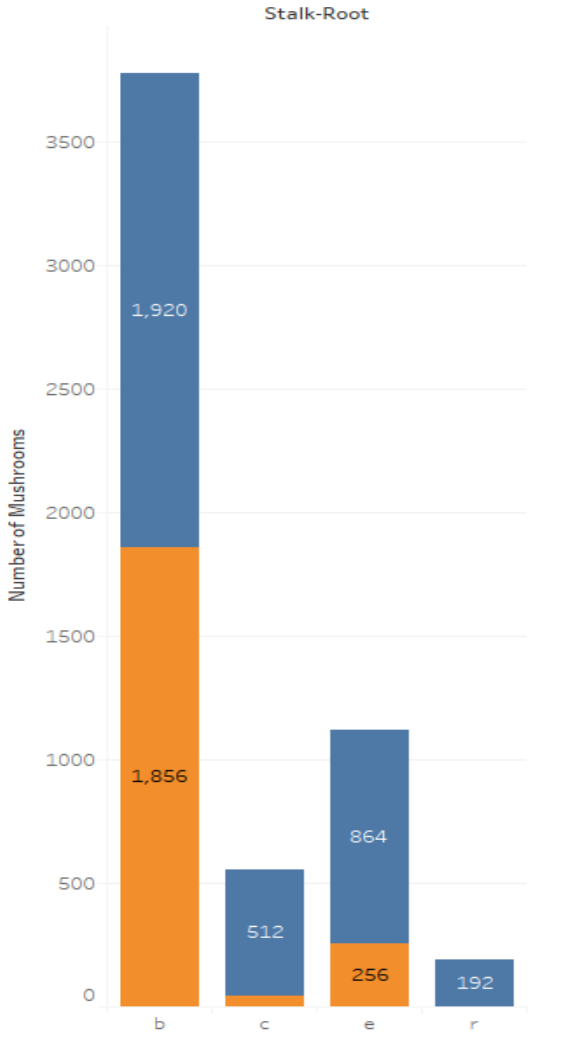
Stalk Color Above Ring



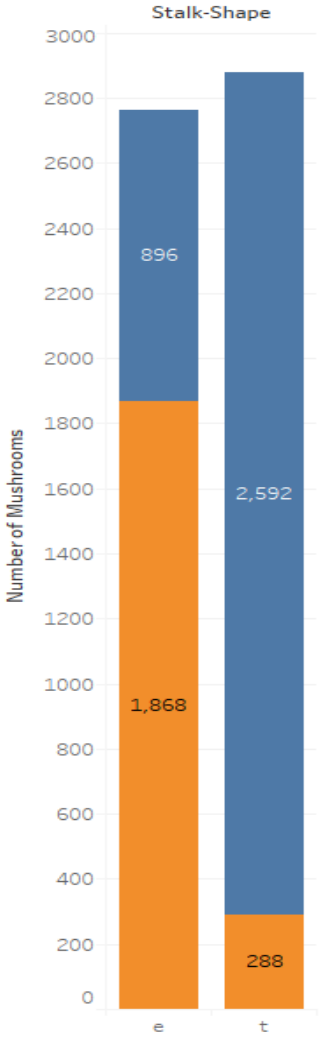
Stalk Color Below Ring



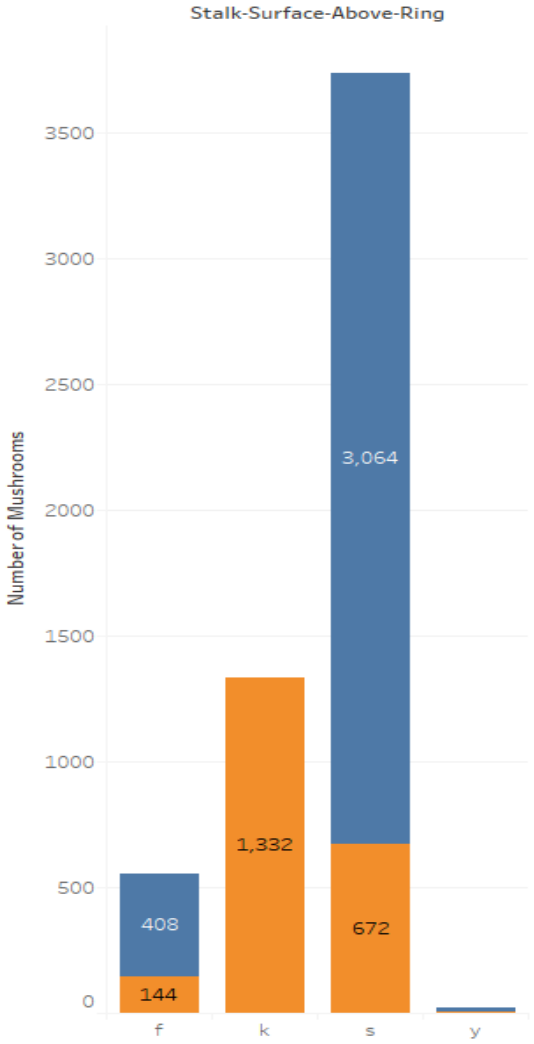
Stalk Root



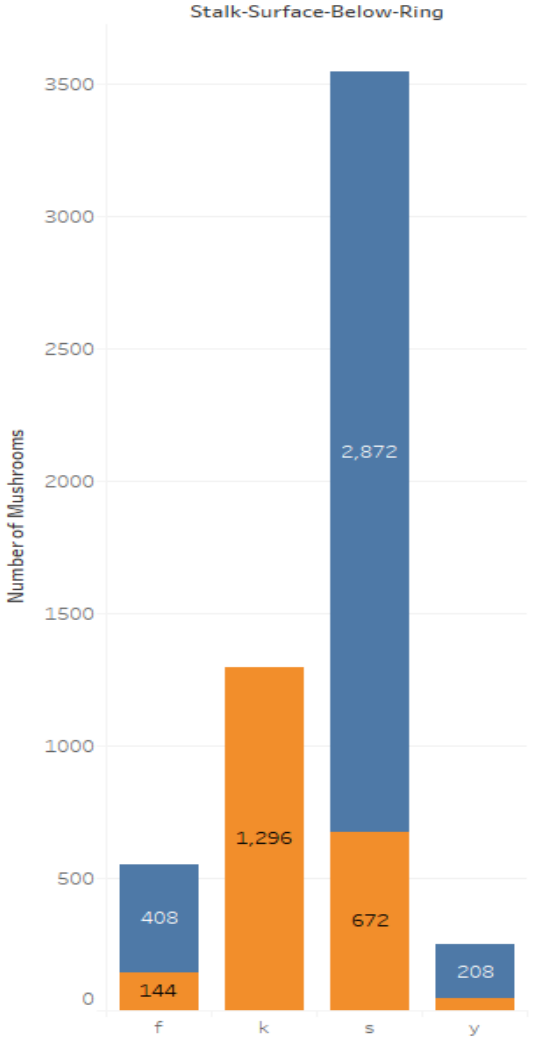
Stalk Shape



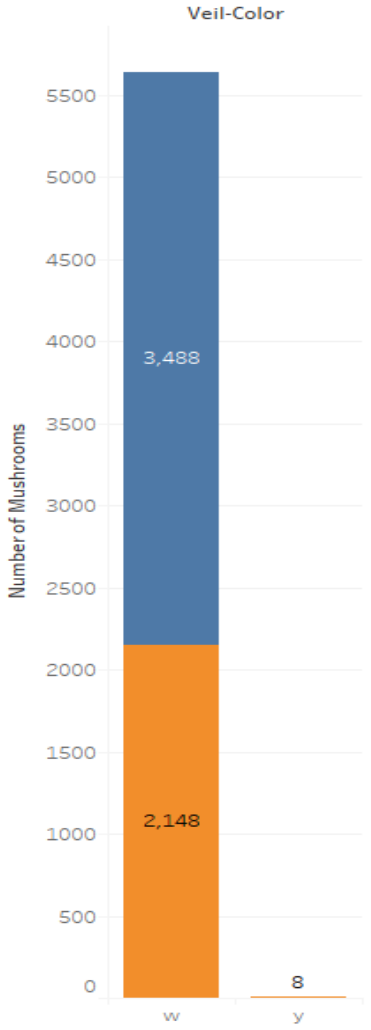
Stalk Surface Above Ring



Stalk Surface Below Ring



Viel Color



PREDICTIVE MODELLING

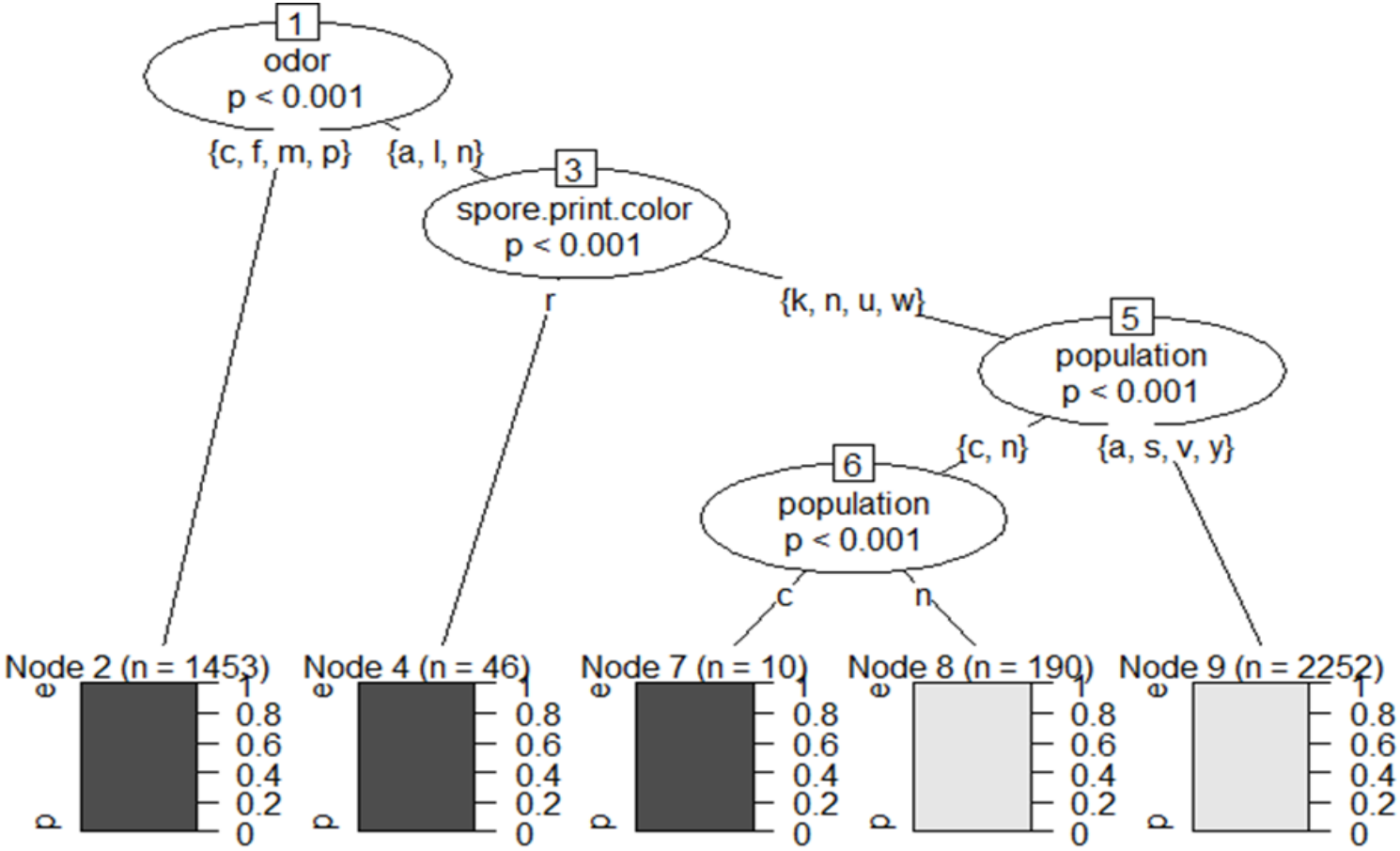
- We ran three models viz. Naïve Bayes, Recursive Partitioning followed by Random Forest. We tried various split ratio in our training and testing set like 60-40,70-30 but the best was given by 60-40 ratios. So, the final results which are shown here is of 60-40 split.
- Dataset1- Deleted all dataset containing missing rows and applied decision tree and random forest to see the accuracy of the same:

- Decision Tree:

Below is the decision tree code followed by the actual tree which was created in R and the most important predictor is odor followed by color and population.

```
cleaningdata<-read.csv (file.choose())
#Dropping column veil type since it wouldn't
be much useful in modelling
cleaningdata$veil.type<-NULL
#Replacing ? with NA
cleaningdata[cleaningdata$stalk.root=='?',"stal
k.root"] <- NA
#Omitting the rows with NA
cleaningdata<-na.omit(cleaningdata)
cleaningdata<-
sample.split(Y=cleaningdata$PE,SplitRatio = .6)
train<-cleaningdata[cleaningdata1,]
test<-cleaningdata[!cleaningdata1,]
#building tree_model decision tree
library(rpart)
tree_model<-
rpart(PE~.,data=train,method='class')
plot(tree_model)
text(tree_model,pretty=0)
```

```
#Summar of decsion tree and how it is being
splitted
summary(tree_model)
#Predicition in the testing dataset
predictionwithClass<-
predict(tree_model,test,type='class')
t<-
table(predictions=predictionwithClass,actual=t
est$PE)
#Accuracy matrix to see the accuracy of the
model
sum(diag(t))/sum(t)
#Calculation of prediction with probability
predictwithprob<-
predict(tree_model,test,type='prob')
#Calculating area under curve
auc<-auc(test$PE,predictwithprob[,2])
#Plotting ROC curve
plot(roc(test$PE,predictwithprob[,2]))
```


1
odor
p < 0.001
{c, f, m, p}
{a, l, n}
3
spore.print.color
p < 0.001
r
{k, n, u, w}
5
population
p < 0.001
{c, n}
{a, s, v, y}
6
population
p < 0.001
c
n
Node 2 (n = 1453)
Node 4 (n = 46)
Node 7 (n = 10)
Node 8 (n = 190)
Node 9 (n = 2252)
e
p
1
0.8
0.6
0.4
0.2
0

CONFUSION MATRIX

```
> t
           actual
predictions    e    p
          e 1046    6
          p    0  641
```

- **Accuracy Matrix:**

```
> predict<-predict(tree_model,test,type='class')
> t<-table(predictions=predict,actual=test$PE)
> sum(diag(t))/sum(t)
[1] 0.9964559953
```

```
> library(pROC)
> predictwithprob<-predict(tree_model,test,type='prob')
> auc<-auc(test$PE,predictwithprob[,2])
> auc
Area under the curve: 0.9953632
```

CONCLUSION OF DECISION TREE

- The most important predictor is odor followed by color and population.
- The accuracy came out to be pretty well 99.64 to be exact.
- The area under the curve came out to be 99.5% and ROC i.e. plot of specificity against sensitivity came out to be exceptionally well.

NAÏVE BAYES MODEL

- For this model, we have removed all the missing values in column stalk.root
- Also, we removed column veil.type since it had only one category which would not help much in building a good model.
- We trained our Naïve Bayes model on 70% data and tested it on remaining 30% data.

CODE

```
library ("klaR")
library ("caret")
library ("e1071")

mushroom = read.csv(file.choose())

mushroom$PE <- as.factor(mushroom$PE) #If not done, get error : Invalid prediction for "rpart" object
head(mushroom)

set.seed(1234)
Shuffledmushroom <-mushroom[sample(nrow(mushroom)),]
#Sample Indexes
indexes = sample(1:5644, .7*5644)
# Split data
training = Shuffledmushroom[indexes,]
testing = Shuffledmushroom[-indexes,]
head(training)
head(testing)

#train the model
model <- NaiveBayes(PE ~ ., data=training)
```

```
#test the model
predictions <- predict(model, testing)
warnings() #ignore the warnings
confusionMatrix(testing$PE, predictions$class)

#Create 10 equally size folds
folds <- cut(seq(1,nrow(Shuffledmushroom)),breaks=10,labels=FALSE)
head(folds)
tail(folds)
BayesoutputData = 0

# Cross validation
#Perform 10 fold cross validation
for(i in 1:10){
  Sampleindexes <- which(folds==i,arr.ind=TRUE)
  train <- Shuffledmushroom[Sampleindexes, ]
  test <- Shuffledmushroom[-Sampleindexes, ]

  classifier = NaiveBayes(PE ~ ., data=train)
  pred = predict(classifier, test)
  misClassifyError = mean(pred$class != test$PE)
  misClassifyError
  Accuracy = 1-misClassifyError
  Accuracy
  BayesoutputData[i] = Accuracy
}
head(BayesoutputData,10)
summary(BayesoutputData)
#confusionMatrix(test$PE, pred$class)
```

RESULTS AND INTERPRETATION

```
> head(mushroom)
  PE cap.shape cap.surface cap.color bruises odor gill.attachment gill.spacing gill.size gill.color stalk.shape stalk.root
1  p         x           s         n       t    p               f            c         n          k           e          e
2  e         x           s         y       t    a               f            c         b          k           e          c
3  e         b           s         w       t    l               f            c         b          n           e          c
4  p         x           y         w       t    p               f            c         n          n           e          e
5  e         x           s         g       f    n               f            w         b          k           t          e
6  e         x           y         y       t    a               f            c         b          n           e          c
  stalk.surface.above.ring stalk.surface.below.ring stalk.color.above.ring stalk.color.below.ring veil.color ring.number
1                        s                        s                      w                      w          w           o
2                        s                        s                      w                      w          w           o
3                        s                        s                      w                      w          w           o
4                        s                        s                      w                      w          w           o
5                        s                        s                      w                      w          w           o
6                        s                        s                      w                      w          w           o
  ring.type spore.print.color population habitat
1         p                 k          s       u
2         p                 n          n       g
3         p                 n          n       m
4         p                 k          s       u
5         e                 n          a       g
6         p                 k          n       g
```

CONFUSION MATRIX WITHOUT CV

```
> confusionMatrix(testing$PE, predictions$class)
Confusion Matrix and Statistics

          Reference
Prediction   e   p
         e 997   7
         p  76 614

               Accuracy : 0.951
                 95% CI : (0.9396, 0.9608)
    No Information Rate : 0.6334
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8969
 Mcnemar's Test P-Value : 8.395e-14

            Sensitivity : 0.9292
            Specificity : 0.9887
         Pos Pred Value : 0.9930
         Neg Pred Value : 0.8899
             Prevalence : 0.6334
         Detection Rate : 0.5885
   Detection Prevalence : 0.5927
      Balanced Accuracy : 0.9589

       'Positive' Class : e
```

MEAN ACCURACY WITH CV

head(BayesoutputData,10)

summary(BayesoutputData)

```
> head(BayesoutputData,10)
 [1] 0.9462493 0.9314961 0.9631890 0.9261666 0.9692913 0.9714567 0.9775546 0.9413386 0.9403543 0.9373892
> summary(BayesoutputData)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.9262  0.9381  0.9438  0.9504  0.9678  0.9776
```

- Thus, after cross validation, accuracy increases slightly.

RANDOM FOREST

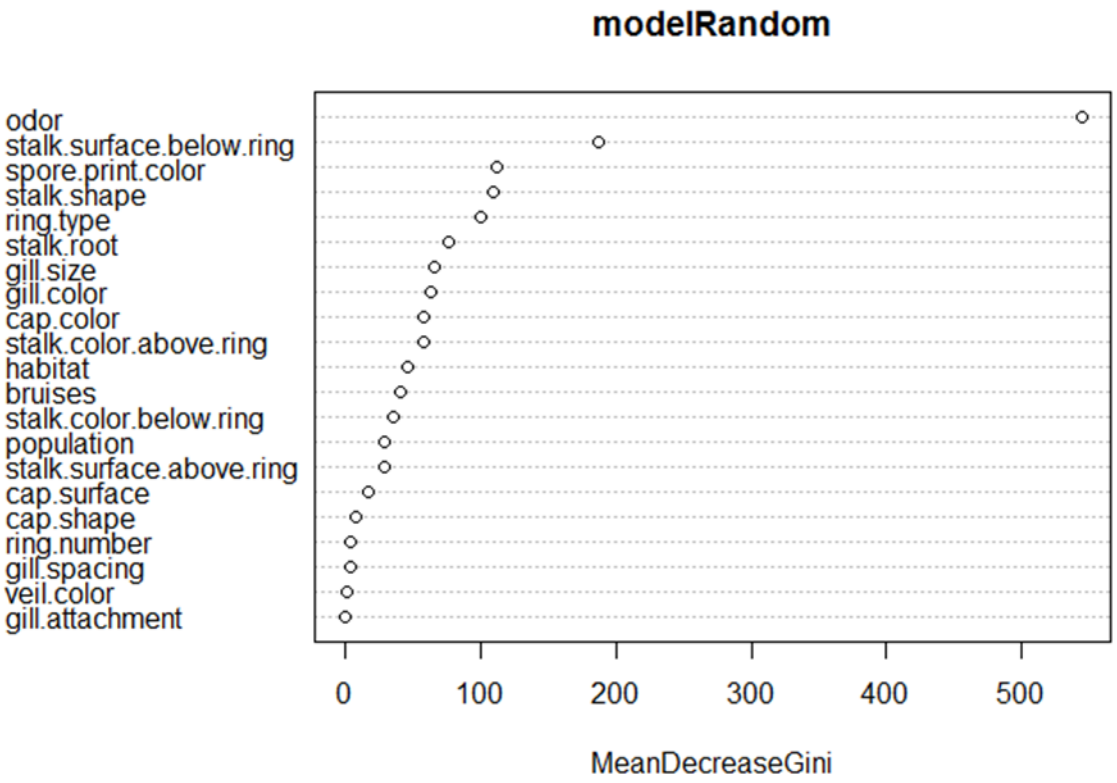
- Random forest is a better version of Decision tree.

```
> data1<-read.csv("mushroom1.csv")
> library(caTools)
> data2<-sample.split(Y=data1$PE,SplitRatio = .6)
> train<-data1[data2,]
> test<-data1[!data2,]
> modelRandom<-randomForest(PE~.,data=train,mtry=3,ntree=20)
> modelRandom

Call:
 randomForest(formula = PE ~ ., data = train, mtry = 3, ntree = 20)
               Type of random forest: classification
                     Number of trees: 20
No. of variables tried at each split: 3

        OOB estimate of  error rate: 0.03%
Confusion matrix:
     e    p     class.error
e 2091    1 0.0004780114723
p    0 1294 0.0000000000000
```

```
> importance(modelRandom)
                           MeanDecreaseGini
cap.shape                         8.364318469
cap.surface                      16.708052757
cap.color                        58.668512135
bruises                          40.443886594
odor                            543.882884182
gill.attachment                   0.000000000
gill.spacing                      3.798283583
gill.size                        65.782190109
gill.color                       63.331573834
stalk.shape                     109.502193728
stalk.root                       76.117312550
stalk.surface.above.ring         28.573440031
stalk.surface.below.ring        187.063829483
stalk.color.above.ring           57.844550779
stalk.color.below.ring           35.784195291
veil.color                        1.694285893
ring.number                       4.349322074
ring.type                        99.900828528
spore.print.color               112.407439118
population                       28.602365964
habitat                          45.961356800
```




```
            actual
predictions    e    p
          e 1395    0
          p    0  862
> sum(diag(t)/sum(t))
[1] 1
```

CONCLUSION OF RANDOM FOREST:

- It performed excellent with 100% accuracy.
- Importance of variable has been plotted once model is build and we could concluded that our most important predictor is odor followed by stalk surface below ring and color. The similar observation was also predicted in decision tree as well.
- Area under curve came out to be 1 since it predicted everything accurately.
- The misclassification rate came out to be 0.03%.

ANOTHER CLEANED DATASET

- We manually Imputed the dataset with Mode in this case and build predictive models.
- For RPART, we decided to train the model on 60% data and tested it on remaining 40% data.

CONCLUSION

- The most important predictor is again odor followed by color and population.
- The accuracy came out to be 99.5 to be exact.
- The area under the curve came out to be 99.5% and ROC came out to be exceptionally well.

```
> predict1<-predict(tree_model1,test1,type='class')
> t<-table(predictions=predict1,actual=test1$PE)
> t
           actual
predictions    e    p
          e 1262   12
          p    0 1163
> sum(diag(t))/sum(t)
[1] 0.995075913
```

RANDOM FOREST ON IMPUTED DATASET

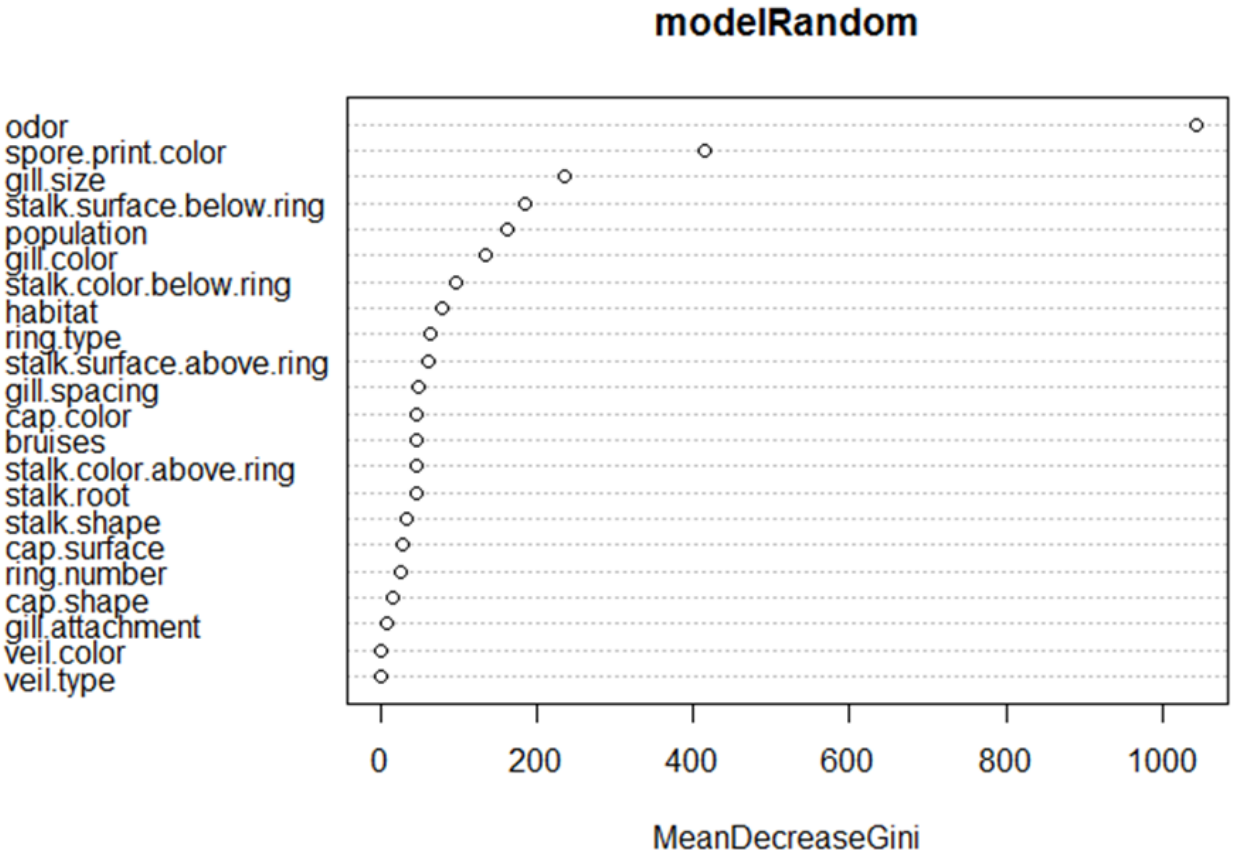
- **Conclusions**:

- It performed excellent with 100% accuracy.
- Importance of variable has been plotted once model is build and we concluded that our most important predictor is odor followed by stalk surface below ring and color. The similar observation was also predicted in decision tree as well.
- Misclassification rate is 0.02%.

```
Call:
 randomForest(formula = PE ~ ., data = train, mtry = 3, ntree = 20)
               Type of random forest: classification
                     Number of trees: 20
No. of variables tried at each split: 3

        OOB estimate of  error rate: 0.02%
Confusion matrix:
     e    p     class.error
e 2946    0 0.0000000000000
p    1 2740 0.0003648303539
```

```
> importance(modelRandom)
                           MeanDecreaseGini
cap.shape                        16.5839330018
cap.surface                      26.7275290399
cap.color                        46.7160759581
bruises                          46.2888438583
odor                           1042.8887842887
gill.attachment                   6.7261285945
gill.spacing                     47.5290863649
gill.size                       235.9131997576
gill.color                      134.8955710452
stalk.shape                      32.9055903510
stalk.root                       45.9118609549
stalk.surface.above.ring         61.4465611776
stalk.surface.below.ring        185.1084862340
stalk.color.above.ring           46.0130947636
stalk.color.below.ring           97.3247082305
veil.type                         0.0000000000
veil.color                        0.3007319735
ring.number                      24.2619080702
ring.type                        63.2961532857
spore.print.color               414.6776527742
population                      161.7569019674
habitat                          79.0348770738
>
```


modelRandom
odor
spore.print.color
gill.size
stalk.surface.below.ring
population
gill.color
stalk.color.below.ring
habitat
ring.type
stalk.surface.above.ring
gill.spacing
cap.color
bruises
stalk.color.above.ring
stalk.root
stalk.shape
cap.surface
ring.number
cap.shape
gill.attachment
veil.color
veil.type
0
200
400
600
800
1000
MeanDecreaseGini

PROJECT CONCLUSION

- Random Forest performed best with 100 % accuracy in all the three datasets.
- Naïve Bayes model without cross validation gives 95.1% accuracy however after cross validation it drops a bit to 93.74%.
- Decision tree also performed well and there was not much difference in any of the dataset. The difference was of just 0.1 %.
- Odor and color plays a great role in determining whether mushrooms are poisonous or not.

LIMITATIONS AND IMPROVEMENTS

- There is enough scope of improvement in the model since the dataset was very small.
- The RPart and Naïve Bayes models' accuracy should have been more i.e. closer to 100% since the failure in predicting the poisonous mushrooms may result fatal.
- We can segment and combine categories into groups for variables having larger categories