> #### Rushith Kalakonda, 14311016.

> ### Training data analysis

>

> mushroom.df <- read.csv("mushroom\_train.csv")

>

> #Variable 'veil.type' can be eliminated as it has only one category, 'p' throughout.

> mushroom.df$veil.type <- NULL

>

> #Identifying the missing values

> mushroom.df[mushroom.df == "?"] <- NA

> library(mice)

> md.pattern(mushroom.df)

class cap.shape cap.surface cap.color bruises odor gill.attachment gill.spacing

3959 1 1 1 1 1 1 1 1

1727 1 1 1 1 1 1 1 1

0 0 0 0 0 0 0 0

gill.size gill.color stalk.shape radius stalk.surface.above.ring

3959 1 1 1 1 1

1727 1 1 1 1 1

0 0 0 0 0

stalk.surface.below.ring stalk.color.above.ring stalk.color.below.ring veil.color

3959 1 1 1 1

1727 1 1 1 1

0 0 0 0

weight ring.number ring.type spore.print.color population habitat stalk.root

3959 1 1 1 1 1 1 1 0

1727 1 1 1 1 1 1 0 1

0 0 0 0 0 0 1727 1727

>

> #Converting continuous variables 'weight' and 'radius' to factor variables.

> mushroom.df$weight <- cut(mushroom.df$weight, c(0,10,20,30,40,50,60,70,80,90,102))

> mushroom.df$radius <- cut(mushroom.df$radius, c(0,100,200,300,400,500,600,700,800,900,2300,3500))

>

> #Predicting the missing values using kNN-Imputation method

> library(VIM)

> corrected.df <- kNN(mushroom.df, variable = c("stalk.root","weight"), k = 32, imp\_var = FALSE)

> corrected.df$stalk.root <- factor(corrected.df$stalk.root)

>

> ##1-R classifier

> library(OneR)

> model1 <- OneR(class ~ ., data = corrected.df)

> model1

Call:

OneR.formula(formula = class ~ ., data = corrected.df)

Rules:

If odor = a then class = e

If odor = c then class = p

If odor = f then class = p

If odor = l then class = e

If odor = m then class = p

If odor = n then class = e

If odor = p then class = p

If odor = s then class = p

If odor = y then class = p

Accuracy:

5594 of 5686 instances classified correctly (98.38%)

> summary(model1)

Call:

OneR.formula(formula = class ~ ., data = corrected.df)

Rules:

If odor = a then class = e

If odor = c then class = p

If odor = f then class = p

If odor = l then class = e

If odor = m then class = p

If odor = n then class = e

If odor = p then class = p

If odor = s then class = p

If odor = y then class = p

Accuracy:

5594 of 5686 instances classified correctly (98.38%)

Contingency table:

odor

class a c f l m n p s y Sum

e \* 270 0 0 \* 280 0 \* 2395 0 0 0 2945

p 0 \* 127 \* 1501 0 \* 24 92 \* 188 \* 411 \* 398 2741

Sum 270 127 1501 280 24 2487 188 411 398 5686

---

Maximum in each column: '\*'

Pearson's Chi-squared test:

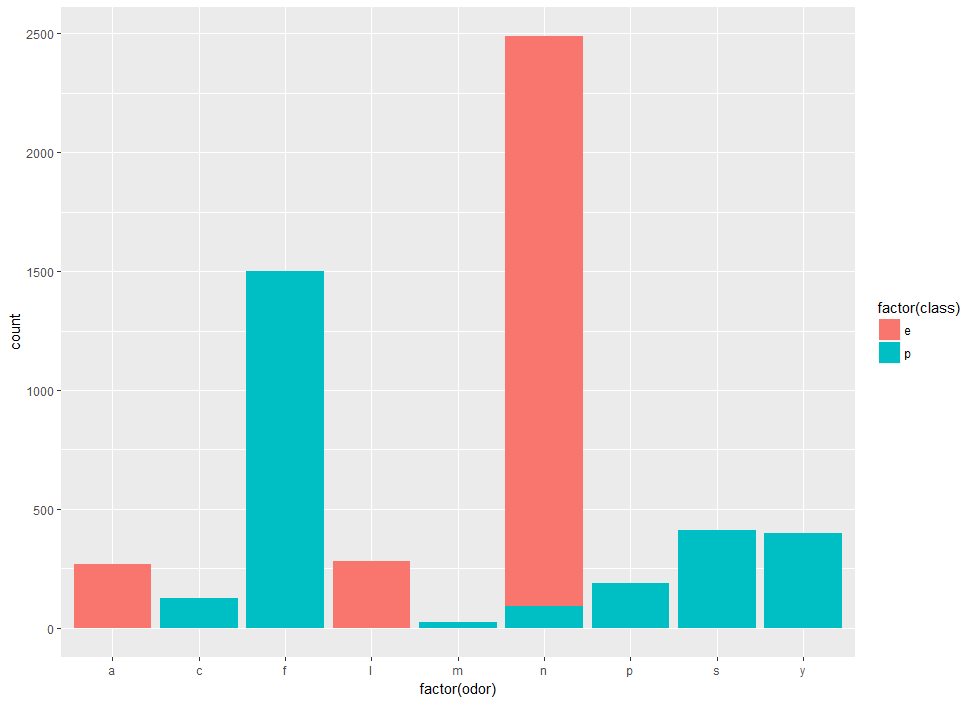
X-squared = 5331.2, df = 8, p-value < 2.2e-16

> #Plotting 'class' against 'odor'

> library(ggplot2)

> qplot(factor(odor),data = corrected.df,geom = "bar",fill = factor(class)

)



>

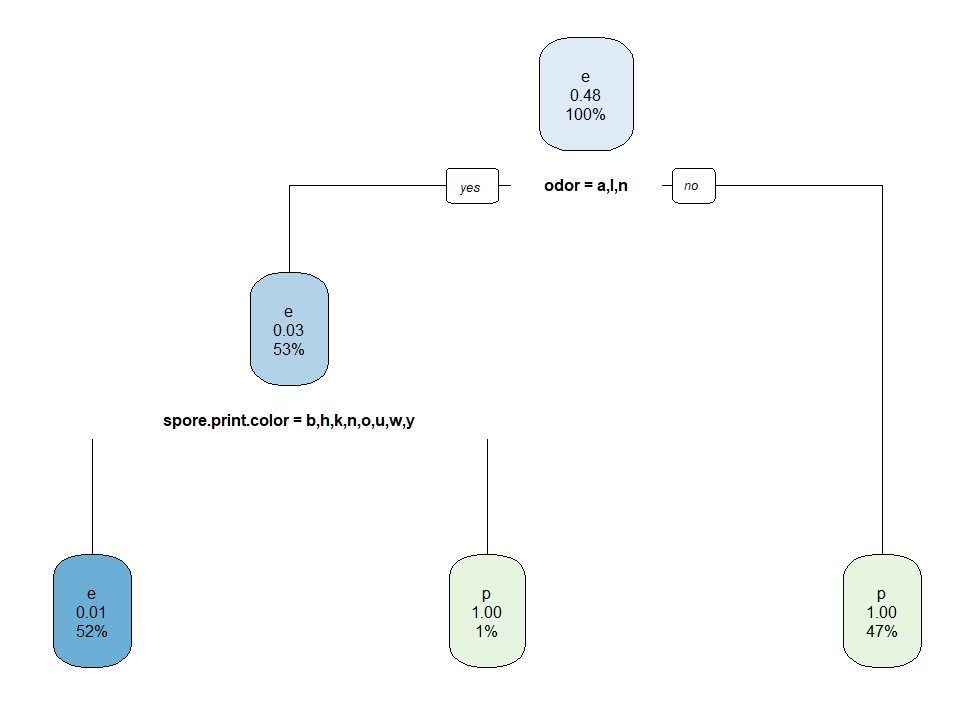
> ##Recursive Partitioning

> library(rpart)

> library(rpart.plot)

> model2 <- rpart(class ~., data = corrected.df)

> rpart.plot(model2)



> summary(model2)

Call:

rpart(formula = class ~ ., data = corrected.df)

n= 5686

CP nsplit rel error xerror xstd

1 0.96643561 0 1.00000000 1.00000000 0.013746265

2 0.01897118 1 0.03356439 0.03356439 0.003470905

3 0.01000000 2 0.01459321 0.01459321 0.002299259

Variable importance

odor spore.print.color gill.color

25 19 15

stalk.surface.above.ring stalk.surface.below.ring ring.type

14 14 13

Node number 1: 5686 observations, complexity param=0.9664356

predicted class=e expected loss=0.4820612 P(node) =1

class counts: 2945 2741

probabilities: 0.518 0.482

left son=2 (3037 obs) right son=3 (2649 obs)

Primary splits:

odor splits as LRRLRLRRR, improve=2660.9140, (0 missing)

spore.print.color splits as LRLLLRLRL, improve=1520.7840, (0 missing)

gill.color splits as RLRRLLLLRLLL, improve=1049.8420, (0 missing)

stalk.surface.above.ring splits as LRLL, improve= 980.6724, (0 missing)

stalk.surface.below.ring splits as LRLL, improve= 927.0266, (0 missing)

Surrogate splits:

spore.print.color splits as LRLLLLLRL, agree=0.858, adj=0.696, (0 split)

gill.color splits as RLRRLLLLLLLL, agree=0.808, adj=0.589, (0 split)

stalk.surface.above.ring splits as LRLL, agree=0.782, adj=0.533, (0 split)

stalk.surface.below.ring splits as LRLL, agree=0.781, adj=0.531, (0 split)

ring.type splits as RLRRL, agree=0.781, adj=0.530, (0 split)

Node number 2: 3037 observations, complexity param=0.01897118

predicted class=e expected loss=0.03029305 P(node) =0.5341189

class counts: 2945 92

probabilities: 0.970 0.030

left son=4 (2985 obs) right son=5 (52 obs)

Primary splits:

spore.print.color splits as LLLLLRLLL, improve=99.49811, (0 missing)

stalk.color.below.ring splits as --LLLLLLR, improve=39.76892, (0 missing)

gill.color splits as -LLLLLLLRLLL, improve=32.15124, (0 missing)

cap.color splits as RLLLLRLLLL, improve=17.93490, (0 missing)

stalk.color.above.ring splits as --LLLLLLR, improve=15.08504, (0 missing)

Surrogate splits:

gill.color splits as -LLLLLLLRLLL, agree=0.988, adj=0.327, (0 split)

Node number 3: 2649 observations

predicted class=p expected loss=0 P(node) =0.4658811

class counts: 0 2649

probabilities: 0.000 1.000

Node number 4: 2985 observations

predicted class=e expected loss=0.01340034 P(node) =0.5249736

class counts: 2945 40

probabilities: 0.987 0.013

Node number 5: 52 observations

predicted class=p expected loss=0 P(node) =0.009145269

class counts: 0 52

probabilities: 0.000 1.000

>

> ##RIPPER algorithm

> library(RWeka)

> model3\_trial <- JRip(class ~., data = corrected.df)

> model3\_trial

JRIP rules:

===========

(odor = f) => class=p (1501.0/0.0)

(gill.size = n) and (gill.color = b) => class=p (809.0/0.0)

(gill.size = n) and (odor = p) => class=p (188.0/0.0)

(odor = c) => class=p (127.0/0.0)

(spore.print.color = r) => class=p (52.0/0.0)

(stalk.surface.below.ring = y) and (stalk.surface.above.ring = k) => class=p (48.0/0.0)

(stalk.color.above.ring = y) => class=p (8.0/0.0)

(habitat = l) and (cap.color = w) => class=p (8.0/0.0)

=> class=e (2945.0/0.0)

Number of Rules : 9

>

> #Avoiding overfitting of RIPPER algorithm using results from 'model3\_trial'

> model3 <- JRip(class ~ odor + gill.color + spore.print.color, data = corrected.df)

> model3

JRIP rules:

===========

(odor = f) => class=p (1501.0/0.0)

(gill.color = b) => class=p (809.0/0.0)

(odor = p) => class=p (188.0/0.0)

(odor = c) => class=p (127.0/0.0)

(spore.print.color = r) => class=p (52.0/0.0)

(spore.print.color = w) and (odor = m) => class=p (24.0/0.0)

=> class=e (2985.0/40.0)

Number of Rules : 7

> summary(model3)

=== Summary ===

Correctly Classified Instances 5646 99.2965 %

Incorrectly Classified Instances 40 0.7035 %

Kappa statistic 0.9859

Mean absolute error 0.0139

Root mean squared error 0.0833

Relative absolute error 2.7798 %

Root relative squared error 16.6727 %

Total Number of Instances 5686

=== Confusion Matrix ===

a b <-- classified as

2945 0 | a = e

40 2701 | b = p

>

>

> ###Test Data and prediction

>

> #Proceeding similarly for the test data

> test.df <- read.csv("mushroom\_test.csv")

> newtest.df <- test.df

>

> newtest.df$veil.type <- NULL

>

> newtest.df[newtest.df == "?"] <- NA

>

> newtest.df$weight <- cut(test.df$weight, c(0,10,20,30,40,50,60,70,80,90,102))

> newtest.df$radius <- cut(test.df$radius, c(0,100,200,300,400,500,600,700,800,900,2300,3500))

>

> newtest.df <- kNN(newtest.df, variable = "stalk.root", k = 25, imp\_var = FALSE)

> newtest.df$stalk.root <- factor(newtest.df$stalk.root)

>

> ##Prediction using model1

> prediction1 <- predict(model1, newdata = newtest.df)

> prediction1\_table <- cbind(prediction1, test.df)

>

> ##Prediction using model2

> prediction2 <- round(predict(model2, newdata = newtest.df))

> prediction2\_table <- cbind(prediction2, test.df)

>

> ##Prediction using model3

> prediction3 <- predict(model3, newdata = newtest.df)

> prediction3\_table <- cbind(prediction3, test.df)

> View(prediction3\_table)

> #model3 is the most optimal for this classification