รายงาน

การบ้าน 3 วิชา CS358

โดย

นางสาวกัลยาณี ดุมเกษม 5809610420

รายงานนี้เป็นส่วนหนึ่งของวิชา คพ.358

สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี

มหาวิทยาลัยธรรมศาสตร์ ปีการศึกษา 2/2560

GitHub: https://github.com/boombibi/CS358-R-Project-

รายงานผลการดำเนินการในการทำ Decision Tree Model ดังนี้

1. **การเตรียมชุดข้อมูล (Data acquisition)**

   วิธีการในการเตรียมชุดข้อมูล

   เริ่มจากการอ่าน ไฟล์ .csv

```
> mushroomData <- read.csv("C:/Users/tsb/Desktop/R Project/mushroom-class
ification/mushrooms.csv");
```

   ดู Structure ของข้อมูล

```
'data.frame':   8124 obs. of  23 variables:
 $ class                    : Factor w/ 2 levels "e","p": 2 1 1 2 1 1 1 1 2 1 ...
 $ cap.shape                : Factor w/ 6 levels "b","c","f","k",..: 6 6 1 6 6 6 1 1 6 1 ...
 $ cap.surface              : Factor w/ 4 levels "f","g","s","y": 3 3 3 4 3 4 3 4 4 3 ...
 $ cap.color                : Factor w/ 10 levels "b","c","e","g",..: 5 10 9 9 4 10 9 9 9 10 ...
 $ bruises                  : Factor w/ 2 levels "f","t": 2 2 2 2 1 2 2 2 2 2 ...
 $ odor                     : Factor w/ 9 levels "a","c","f","l",..: 7 1 4 7 6 1 1 4 7 1 ...
 $ gill.attachment          : Factor w/ 2 levels "a","f": 2 2 2 2 2 2 2 2 2 2 ...
 $ gill.spacing             : Factor w/ 2 levels "c","w": 1 1 1 1 2 1 1 1 1 1 ...
 $ gill.size                : Factor w/ 2 levels "b","n": 2 1 1 2 1 1 1 1 2 1 ...
 $ gill.color               : Factor w/ 12 levels "b","e","g","h",..: 5 5 6 6 5 6 3 6 8 3 ...
 $ stalk.shape              : Factor w/ 2 levels "e","t": 1 1 1 1 2 1 1 1 1 1 ...
 $ stalk.root               : Factor w/ 5 levels "?","b","c","e",..: 4 3 3 4 4 3 3 3 4 3 ...
 $ stalk.surface.above.ring: Factor w/ 4 levels "f","k","s","y": 3 3 3 3 3 3 3 3 3 3 ...
 $ stalk.surface.below.ring: Factor w/ 4 levels "f","k","s","y": 3 3 3 3 3 3 3 3 3 3 ...
 $ stalk.color.above.ring   : Factor w/ 9 levels "b","c","e","g",..: 8 8 8 8 8 8 8 8 8 8 ...
 $ stalk.color.below.ring   : Factor w/ 9 levels "b","c","e","g",..: 8 8 8 8 8 8 8 8 8 8 ...
 $ veil.type                : Factor w/ 1 level "p": 1 1 1 1 1 1 1 1 1 1 ...
 $ veil.color               : Factor w/ 4 levels "n","o","w","y": 3 3 3 3 3 3 3 3 3 3 ...
 $ ring.number              : Factor w/ 3 levels "n","o","t": 2 2 2 2 2 2 2 2 2 2 ...
 $ ring.type                : Factor w/ 5 levels "e","f","l","n",..: 5 5 5 5 1 5 5 5 5 5 ...
 $ spore.print.color        : Factor w/ 9 levels "b","h","k","n",..: 3 4 4 3 4 3 3 4 3 3 ...
 $ population               : Factor w/ 6 levels "a","c","n","s",..: 4 3 3 4 1 3 3 4 5 4 ...
 $ habitat                  : Factor w/ 7 levels "d","g","l","m",..: 6 2 4 6 2 2 4 4 2 4 ...
>
```

   ทำการเปลี่ยนชื่อ Columns ให้อ่านง่ายขึ้น

```
> colnames(mushroomData) <- c("edibility", "cap_shape", "cap_surface",
+                             "cap_color", "bruises", "odor",
+                             "gill_attachement", "gill_spacing", "gill_size",
+                             "gill_color", "stalk_shape", "stalk_root",
+                             "stalk_surface_above_ring", "stalk_surface_below_ring", "stalk_color_above_ring",
+                             "stalk_color_below_ring", "veil_type", "veil_color",
+                             "ring_number", "ring_type", "spore_print_color",
+                             "population", "habitat")
```

ทำการ Assign ค่าตัวแปรใน แต่ละ Attribute ใหม่

```
> levels(mushroomData$edibility) <- c("edible", "poisonous")
> levels(mushroomData$cap_shape) <- c("bell", "conical", "flat", "knobbed", "sunken", "convex")
> levels(mushroomData$cap_color) <- c("buff", "cinnamon", "red", "gray", "brown", "pink",
+                                     "green", "purple", "white", "yellow")
> levels(mushroomData$cap_surface) <- c("fibrous", "grooves", "scaly", "smooth")
> levels(mushroomData$bruises) <- c("no", "yes")
> levels(mushroomData$odor) <- c("almond", "creosote", "foul", "anise", "musty", "none", "pungent", "spicy", "fishy")
> levels(mushroomData$gill_attachement) <- c("attached", "free")
> levels(mushroomData$gill_spacing) <- c("close", "crowded")
> levels(mushroomData$gill_size) <- c("broad", "narrow")
> levels(mushroomData$gill_color) <- c("buff", "red", "gray", "chocolate", "black", "brown", "orange",
+                                      "pink", "green", "purple", "white", "yellow")
> levels(mushroomData$stalk_shape) <- c("enlarging", "tapering")
> levels(mushroomData$stalk_root) <- c("missing", "bulbous", "club", "equal", "rooted")
> levels(mushroomData$stalk_surface_above_ring) <- c("fibrous", "silky", "smooth", "scaly")
> levels(mushroomData$stalk_surface_below_ring) <- c("fibrous", "silky", "smooth", "scaly")
> levels(mushroomData$stalk_color_above_ring) <- c("buff", "cinnamon", "red", "gray", "brown", "pink",
+                                                  "green", "purple", "white", "yellow")
> levels(mushroomData$stalk_color_below_ring) <- c("buff", "cinnamon", "red", "gray", "brown", "pink",
+                                                  "green", "purple", "white", "yellow")
> levels(mushroomData$veil_type) <- "partial"
> levels(mushroomData$veil_color) <- c("brown", "orange", "white", "yellow")
> levels(mushroomData$ring_number) <- c("none", "one", "two")
> levels(mushroomData$ring_type) <- c("evanescent", "flaring", "large", "none", "pendant")
> levels(mushroomData$spore_print_color) <- c("buff", "chocolate", "black", "brown", "orange",
+                                             "green", "purple", "white", "yellow")
> levels(mushroomData$population) <- c("abundant", "clustered", "numerous", "scattered", "several", "solitary")
> levels(mushroomData$habitat) <- c("wood", "grasses", "leaves", "meadows", "paths", "urban", "waste")
```

2. การแบ่งข้อมูลเพื่อ Train และ Test แบบจำลอง (Data partitioning)

ในการแบ่งข้อมูลเพื่อ Train และ Test จะแบ่งโดยวิธีการสุ่ม และแบ่งข้อมูลสำหรับ Train 70% และสำหรับ Test 30% โดยใช้ฟังก์ชัน ดังต่อไปนี้

```
partitionData <- function( data, fractionOfDataForTrainingData = 0.7 )
{
  numberOfRows <- nrow(data)
  randomRows   <- runif(numberOfRows)
  index         <- randomRows <= fractionOfDataForTrainingData
  trainingData <- data[ index, ]
  testingData  <- data[ !index, ]
  dataSetSplit <- list( trainingData = trainingData, testingData = testingData )
}
```

```
set.seed(1420)

> PartitionedData <- partitionData(mushroomData)
```

เมื่อ Partition แล้วจะได้ข้อมูลมา 2 ชุด คือชุดที่ใช้ Train และ Test

PartitionedData	list [2]	List of length 2
trainingData	list [5707 x 23] (S3: data.frame)	A data.frame with 5707 rows and 23 columns
testingData	list [2417 x 23] (S3: data.frame)	A data.frame with 2417 rows and 23 columns

และเพื่อทดสอบว่า การแบ่งข้อมูลถูกต้อง

```
> round(prop.table(table(mushroomData$edibility)), 2)

   edible poisonous
     0.52      0.48
> round(prop.table(table(trainingData$edibility)), 2)

   edible poisonous
     0.52      0.48
> round(prop.table(table(testingData$edibility)), 2)

   edible poisonous
     0.52      0.48
```

จะเห็นได้ว่าความน่าจะเป็นของ edible และ poisonous ในชุดข้อมูล ทั้ง 3 มีค่าเท่ากัน แสดงว่าการแบ่งข้อมูลถูกต้องแล้

3. **การเลือก Attribute เพื่อสร้างแบบจำลอง (Attribute selection)**

การเริ่มต้นจะต้องหาค่า Information gain ของแต่ละ Attribute เพื่อใช้ในการเลือก Attribute นั้น โดยการเรียกใช้ฟังก์ชัน จากที่เรียนในชั่วโมงบรรยาย

```
> #### Information Gain ####
> InformationGain(table(trainingData[,c('cap_shape','edibility')]))
[1] 0.04440139
> InformationGain(table(trainingData[,c('cap_surface','edibility')]))
[1] 0.03023997
> InformationGain(table(trainingData[,c('cap_color','edibility')]))
[1] 0.03591036
> InformationGain(table(trainingData[,c('bruises','edibility')]))
[1] 0.1891671
> InformationGain(table(trainingData[,c('odor','edibility')]))
[1] 0.9122506
> InformationGain(table(trainingData[,c('gill_attachement','edibility')]))
[1] 0.01342796
> InformationGain(table(trainingData[,c('gill_spacing','edibility')]))
[1] 0.1064521
> InformationGain(table(trainingData[,c('gill_size','edibility')]))
[1] 0.2349405
> InformationGain(table(trainingData[,c('gill_color','edibility')]))
[1] 0.4183871
> InformationGain(table(trainingData[,c('stalk_shape','edibility')]))
[1] 0.008815155
> InformationGain(table(trainingData[,c('stalk_root','edibility')]))
[1] 0.1347293
> InformationGain(table(trainingData[,c('stalk_surface_above_ring','edibility')]))
[1] 0.2823962
> InformationGain(table(trainingData[,c('stalk_surface_below_ring','edibility')]))
[1] 0.2707338
> InformationGain(table(trainingData[,c('stalk_color_above_ring','edibility')]))
[1] 0.2476053
```

```
> InformationGain(table(trainingData[,c('stalk_color_below_ring','edibility')]))
[1] 0.2380757
> InformationGain(table(trainingData[,c('veil_type','edibility')]))
[1] 0
> InformationGain(table(trainingData[,c('veil_color','edibility')]))
[1] 0.02252624
> InformationGain(table(trainingData[,c('ring_number','edibility')]))
[1] 0.03940197
> InformationGain(table(trainingData[,c('ring_type','edibility')]))
[1] 0.3129148
> InformationGain(table(trainingData[,c('spore_print_color','edibility')]))
[1] 0.4757143
> InformationGain(table(trainingData[,c('population','edibility')]))
[1] 0.2026837
> InformationGain(table(trainingData[,c('habitat','edibility')]))
[1] 0.158629
```

เลือก Attribute ที่มีค่า Information Gain จากมากที่สุดดังนี้

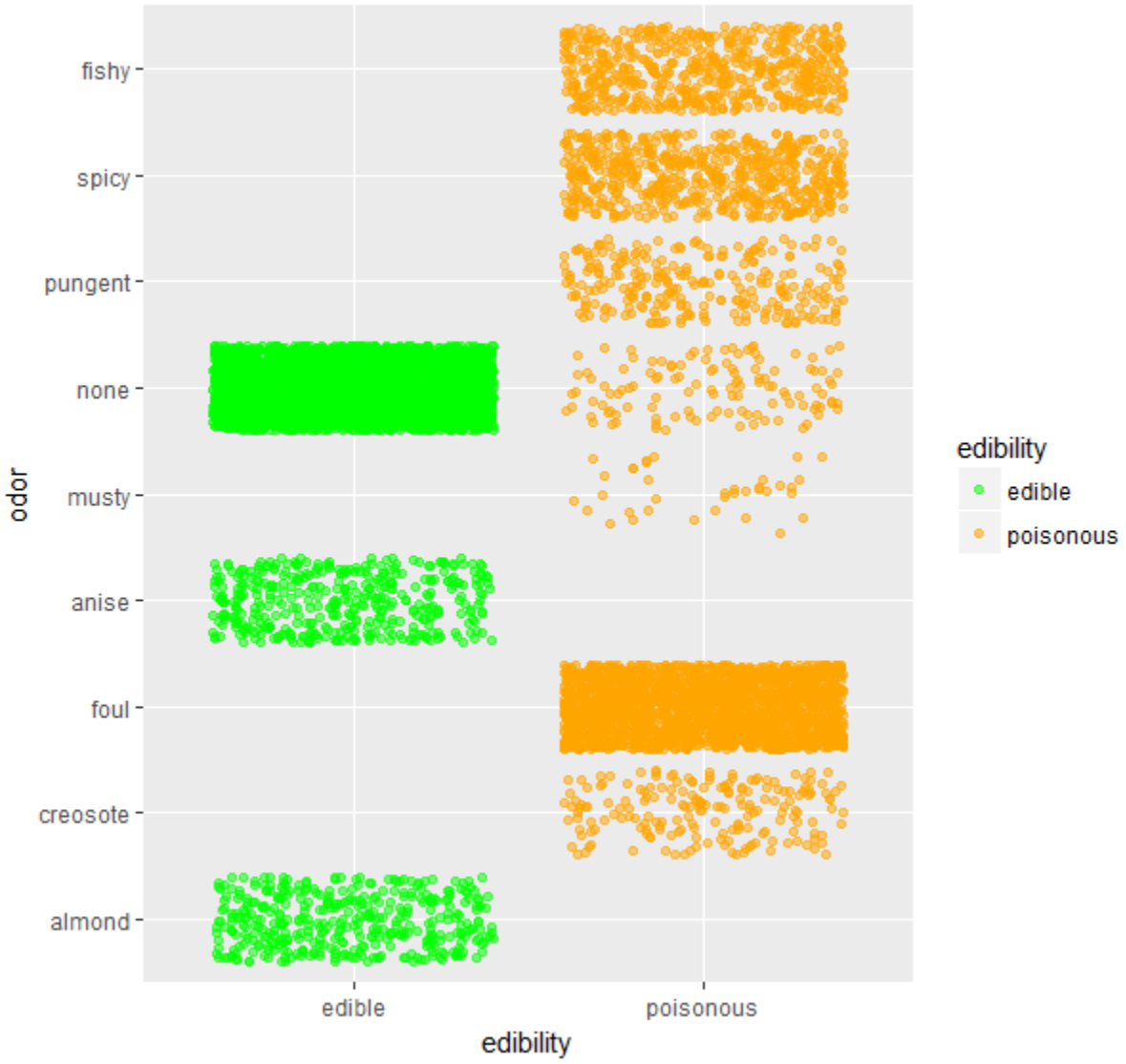
Odor, Spore_print_color

4. การแสดงภาพเกี่ยวกับ Attribute ที่เลือก (Attribute visualization)

Attribute ที่มี Information Gain สูงที่สุด คือ Odor

แสดงภาพที่เกี่ยวกับ Attribute Odor โดยใช้ฟังก์ชันต่อไปนี้

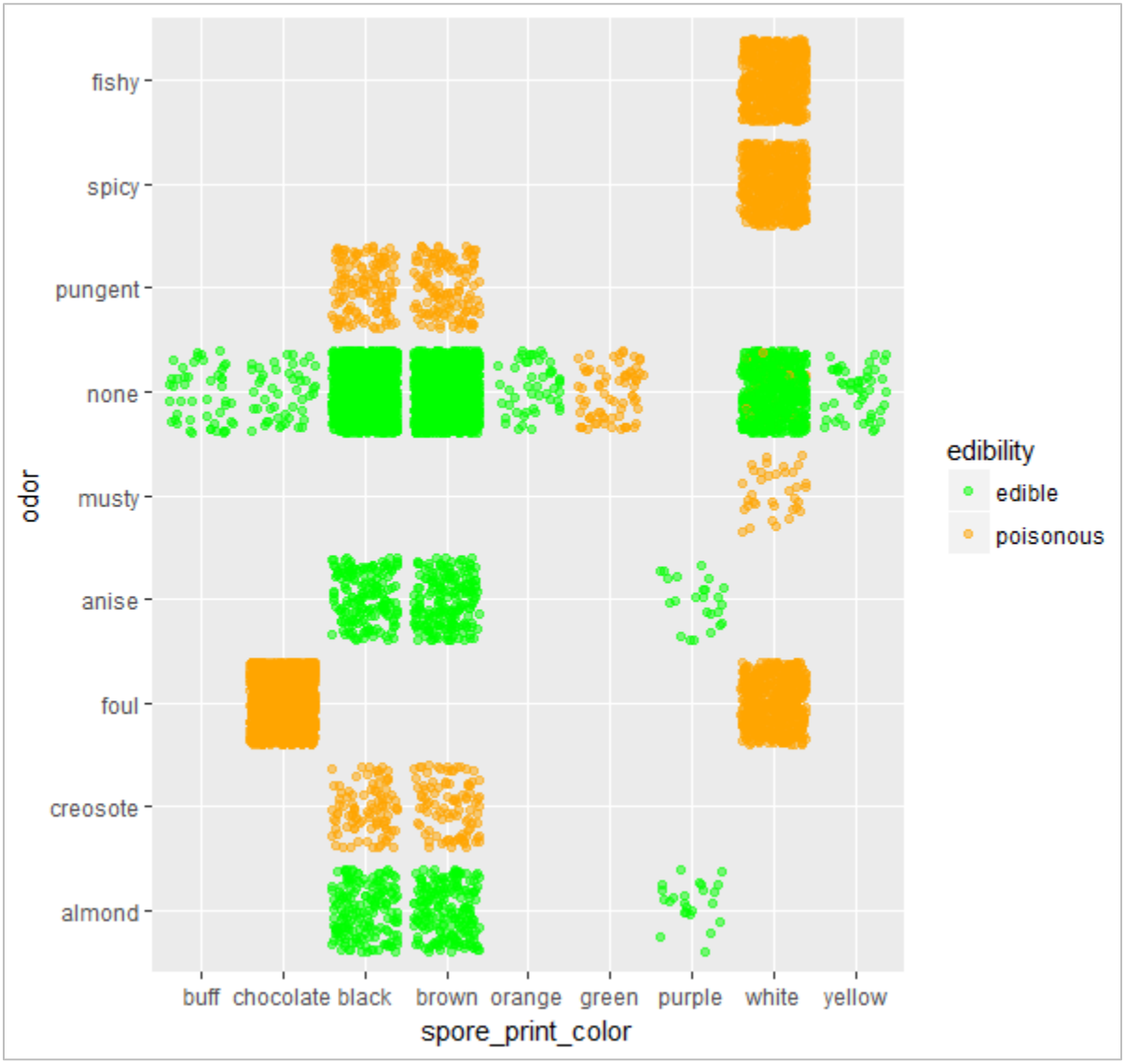
```
> ggplot(mushroomData, aes(x = edibility, y = odor, col = edibility)) +
+     geom_jitter(alpha = 0.5) +
+     scale_color_manual(breaks = c("edible", "poisonous"),
+                        values = c("green", "orange"))
```



จะเห็นชัดเจนว่า edible และ poisonous ไม่มีการปะปนกัน แยกกันอย่างเห็นได้ชัด แต่จะมีลักษณะ none ที่ส่วนใหญ่จะเป็น edible ส่วนน้อยจะเป็น poisonous

แสดงภาพที่เกี่ยวกับ Attribute Odor และ Spore_print_color โดยใช้ฟังชันต่อไปนี้

```
ggplot(mushroomData, aes(x = spore_print_color, y = odor, col = edibility)) +
    geom_jitter(alpha = 0.5) +
    scale_color_manual(breaks = c("edible", "poisonous"),
                       values = c("green", "orange"))
```



จากภาพจะเห็นว่า

- การจับคู่ของ 2 ลักษณะ มีลักษณะ odor – none และ spore_print_color – white มีการผสมกันเล็กน้อย ส่วนใหญ่จะเป็น edible
- คู่ของลักษณะอื่นจะมี edible และ poisonous แยกกันชัดเจน

5. Classification ด้วย Decision Tree (Classification with Decision Tree)

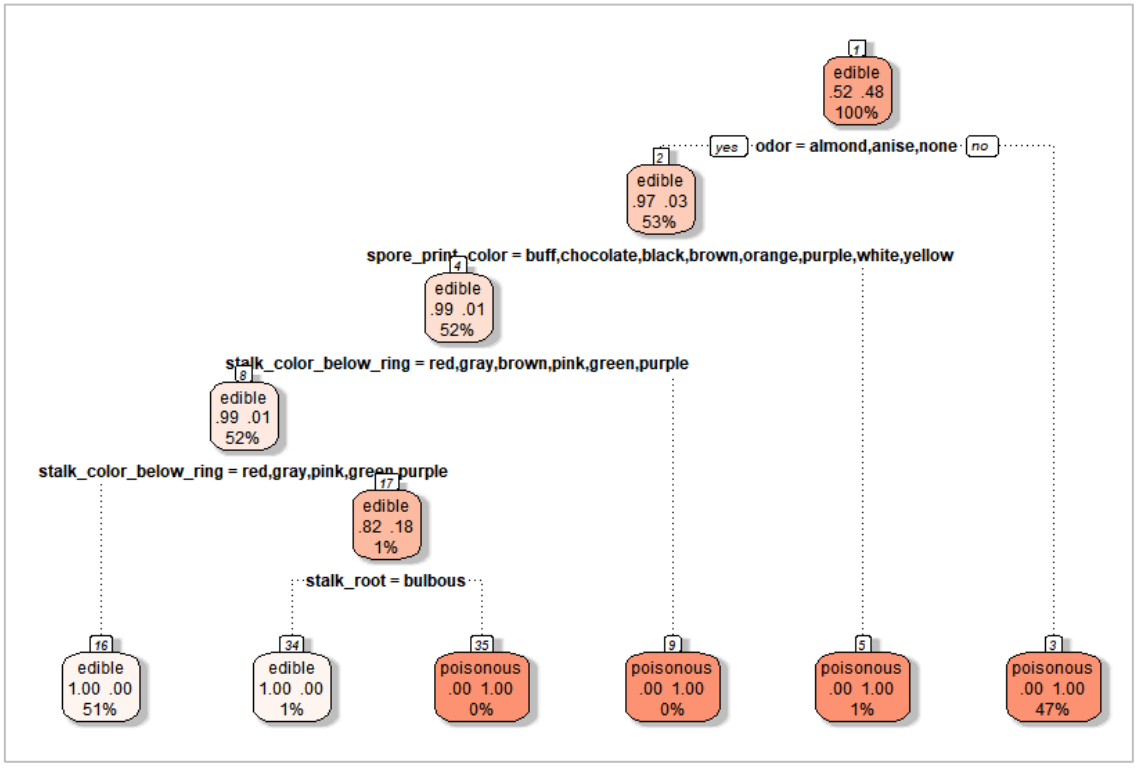
สร้าง Tree Model โดยใช้วิธีการเรียกไลบราลสำเร็จรูป โดยใช้ rPart Package โดยมีขั้นตอนการสร้างดังนี้

1. สร้าง tree_model ใช้ Library rpart โดยใช้ method = “class ” เพราะ data เป็น Factor และใช้ ฟังก์ชัน printcp เพื่อดู Fitted rpart

```
> tree_model <- rpart(edibility ~ ., data = trainingData,method = "class", cp = 0.0001)
> printcp(tree_model)

Classification tree:
rpart(formula = edibility ~ ., data = trainingData, method = "class",
    cp = 1e-04)

Variables actually used in tree construction:
[1] odor                  spore_print_color     stalk_color_below_ring stalk_root

Root node error: 2751/5707 = 0.48204

n= 5707

         CP nsplit rel error    xerror       xstd
1 0.9720102      0 1.0000000 1.0000000 0.01372154
2 0.0178117      1 0.0279898 0.0279898 0.00316814
3 0.0043621      2 0.0101781 0.0101781 0.00191876
4 0.0018175      3 0.0058161 0.0058161 0.00145198
5 0.0001000      5 0.0021810 0.0021810 0.00088993
```

2. ใช้ฟังก์ชัน rapart.plot เพื่อ plot Tree model ที่สร้างในก่อนหน้านี้

```
> rpart.plot(tree_model, extra = 105, box.palette = "Red",
+            branch.lty = 3, shadow.col = "gray", nn = TRUE)
```



3. ใช้ฟังก์ชัน เพื่อคำณวน cross-tabulation ของการ Predict Training Data แสดงผลในรูปแบบสถิติที่เกี่ยวข้อง

เนื่องจากใน factor มี 2 level คือ edible และ poisonous จึงให้ positive = “edible”

```
> caret::confusionMatrix(data=predict(tree_model, type = "class"),
+                        reference = trainingData$edibility,
+                        positive="edible")
Confusion Matrix and Statistics

          Reference
Prediction  edible poisonous
  edible      2956         6
  poisonous      0      2745

               Accuracy : 0.9989
                 95% CI : (0.9977, 0.9996)
    No Information Rate : 0.518
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.9979
 Mcnemar's Test P-Value : 0.04123

            Sensitivity : 1.0000
            Specificity : 0.9978
         Pos Pred Value : 0.9980
         Neg Pred Value : 1.0000
             Prevalence : 0.5180
         Detection Rate : 0.5180
   Detection Prevalence : 0.5190
      Balanced Accuracy : 0.9989

       'Positive' Class : edible
```

อธิบายผลที่ได้ ดังนี้

- ความแม่นยำ (Accuracy) : 0.9989
- ระดับความมั่นใจ 95 % ช่วงความเชื่อมั่น 0.9977 ถึง 0.9996
- ค่าสัมประสิทธิ์ตัวชี้วัดทางสถิติระหว่างผู้ให้ความเห็นสองฝ่าย (Kappa) : 0.9979
- สัดส่วนของผลบวกที่เป็นจริงสำหรับภาวะนั้นๆ (Sensitivity) : 1.0000
- สัดส่วนของผลลบที่เป็นจริงสำหรับภาวะนั้นๆ (Specificity) : 0.9978

4. และใช้ฟังก์ชัน เพื่อคำณวน cross-tabulation ของการ Predict Testing Data แสดงผลในรูปแบบสถิติที่เกี่ยวข้อง

```
> tree_test <- predict(tree_model, newdata = testingData)
> caret::confusionMatrix(data = predict(tree_model, newdata = testingData, type = "class"),
+                        reference = testingData$edibility,
+                        positive = "edible")
Confusion Matrix and Statistics

           Reference
Prediction  edible poisonous
  edible      1252         2
  poisonous      0      1163

               Accuracy : 0.9992
                 95% CI : (0.997, 0.9999)
    No Information Rate : 0.518
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.9983
 Mcnemar's Test P-Value : 0.4795

            Sensitivity : 1.0000
            Specificity : 0.9983
         Pos Pred Value : 0.9984
         Neg Pred Value : 1.0000
             Prevalence : 0.5180
         Detection Rate : 0.5180
   Detection Prevalence : 0.5188
      Balanced Accuracy : 0.9991

       'Positive' Class : edible
```

อธิบายผลที่ได้ ดังนี้

- ความแม่นยำ (Accuracy) : 0.9992
- ระดับความมั่นใจ 95 % ช่วงความเชื่อมั่น 0.997 ถึง 0.9999
- ค่าสัมประสิทธิ์ตัวชี้วัดทางสถิติระหว่างผู้ให้ความเห็นสองฝ่าย (Kappa) : 0.9979
- สัดส่วนของผลบวกที่เป็นจริงสำหรับภาวะนั้นๆ (Sensitivity) : 1.0000
- สัดส่วนของผลลบที่เป็นจริงสำหรับภาวะนั้นๆ (Specificity) : 0.9978

6. สรุปองค์ความรู้ที่ได้จากการใช้แบบจำลองในการแก้ปัญหา

1. ใน Data Set Mushroom มีมีเห็ดประเภท Edible มากกว่า เห็ดชนิด Poisonous
2. จะสามารถทำนายได้ว่า เห็ด ที่กินได้มีลักษณะ ต่อไปนี้
    - กลิ่น Odor : almond, anise, none
    - สีลายพิมพ์สปอร์ Spore Print Color : buff, chocolate, black, brown, purple, white, yellow
    - สีของลำต้นใต้ดอก Stalk color below ring : red, gray, brown, pink, pink, green, purple
    - ก้านของราก Stalk root : bulbous