Project One

2024-11-15

```r
options(repos = c(CRAN = "https://cran.rstudio.com"))


library(mlbench)
```

```
## Warning: package 'mlbench' was built under R version 4.4.2
```

```r
data(BreastCancer)
```

```r
data <- BreastCancer
head(data)
```

```
##        Id Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size
## 1 1000025            5         1          1             1            2
## 2 1002945            5         4          4             5            7
## 3 1015425            3         1          1             1            2
## 4 1016277            6         8          8             1            3
## 5 1017023            4         1          1             3            2
## 6 1017122            8        10         10             8            7
##   Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses     Class
## 1           1           3               1       1    benign
## 2          10           3               2       1    benign
## 3           2           3               1       1    benign
## 4           4           3               7       1    benign
## 5           1           3               1       1    benign
## 6          10           9               7       1 malignant
```

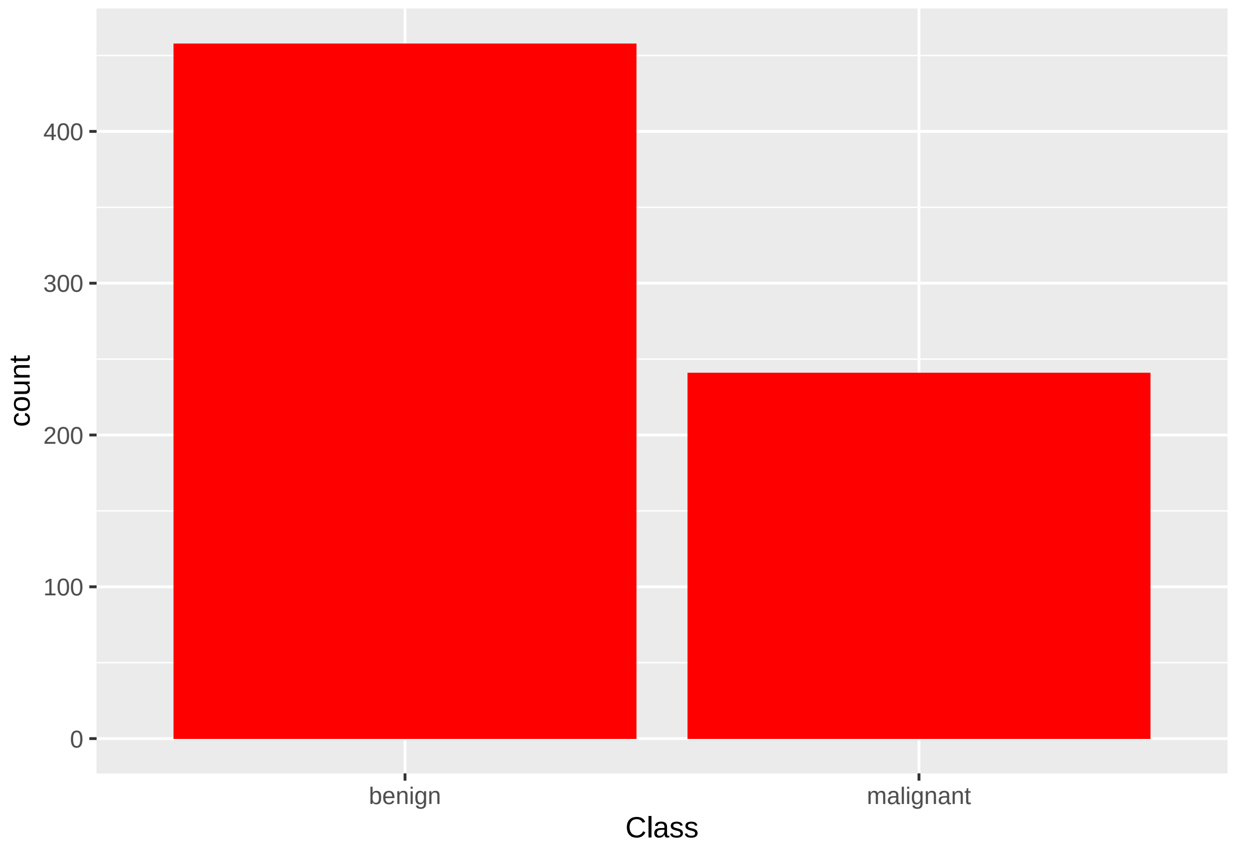
```r
str(data)
```

```
## 'data.frame':    699 obs. of  11 variables:
##  $ Id             : chr  "1000025" "1002945" "1015425" "1016277" ...
##  $ Cl.thickness   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 5 5 3 6 4 8 1 2 2 4 ...
##  $ Cell.size      : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 1 1 2 ...
##  $ Cell.shape     : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 2 1 1 ...
##  $ Marg.adhesion  : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 5 1 1 3 8 1 1 1 1 ...
##  $ Epith.c.size   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 2 7 2 3 2 7 2 2 2 2 ...
##  $ Bare.nuclei    : Factor w/ 10 levels "1","2","3","4",..: 1 10 2 4 1 10 10 1 1 1 ...
##  $ Bl.cromatin    : Factor w/ 10 levels "1","2","3","4",..: 3 3 3 3 3 9 3 3 1 2 ...
##  $ Normal.nucleoli: Factor w/ 10 levels "1","2","3","4",..: 1 2 1 7 1 7 1 1 1 1 ...
##  $ Mitoses        : Factor w/ 9 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 5 1 ...
##  $ Class          : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...
```

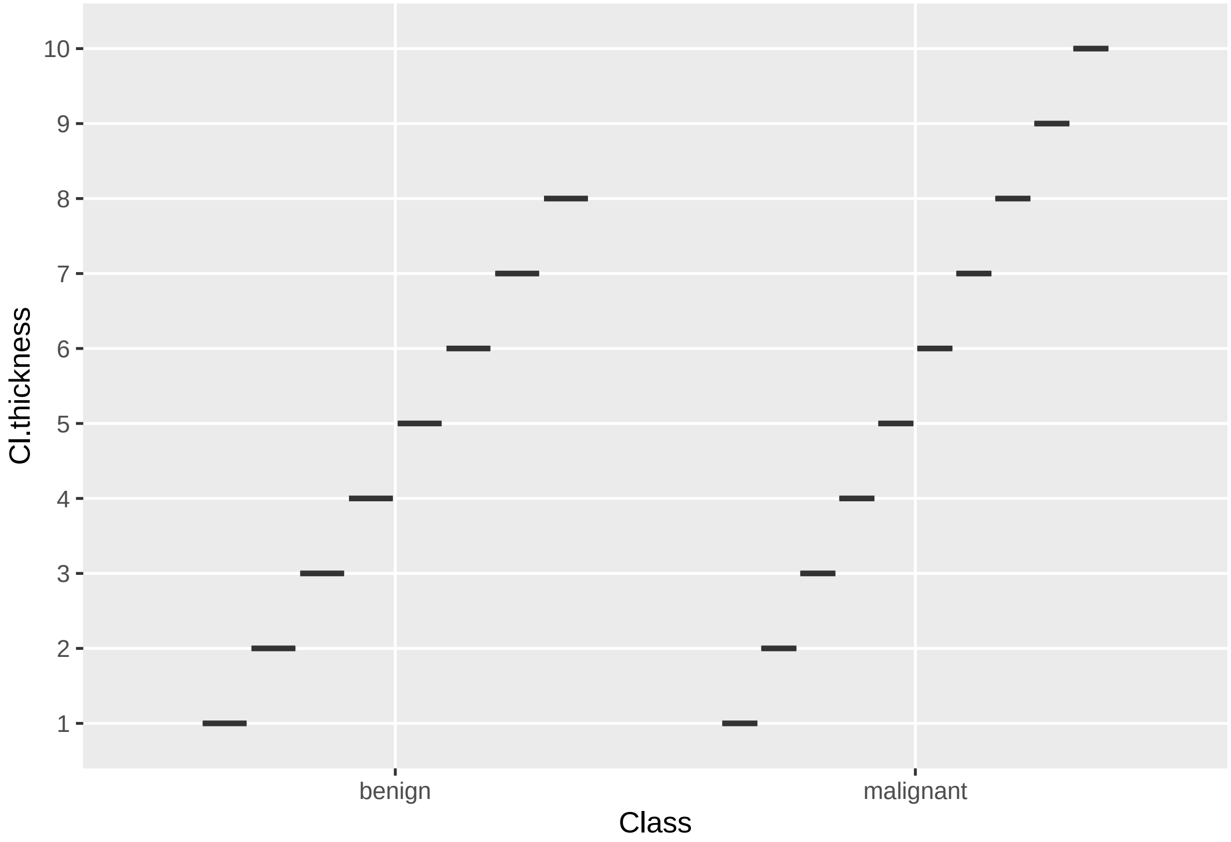
```r
summary(data)
```

```
##        Id             Cl.thickness   Cell.size     Cell.shape  Marg.adhesion
##  Length:699          1      :145   1      :384   1      :353   1      :407  
##  Class :character   5      :130   10     : 67   2      : 59   2      : 58  
##  Mode  :character   3      :108   3      : 52   10     : 58   3      : 58  
##                     4      : 80   2      : 45   3      : 56   10     : 55  
##                     10     : 69   4      : 40   4      : 44   4      : 33  
##                     2      : 50   5      : 30   5      : 34   8      : 25  
##                     (Other):117   (Other): 81   (Other): 95   (Other): 63  
##   Epith.c.size  Bare.nuclei   Bl.cromatin  Normal.nucleoli    Mitoses   
##  2      :386   1      :402   2      :166   1      :443     1      :579  
##  3      : 72   10     :132   3      :165   10     : 61     2      : 35  
##  4      : 48   2      : 30   1      :152   3      : 44     3      : 33  
##  1      : 47   5      : 30   7      : 73   2      : 36     10     : 14  
##  6      : 41   3      : 28   4      : 40   8      : 24     4      : 12  
##  5      : 39   (Other): 61   5      : 34   6      : 22     7      :  9  
##  (Other): 66   NA's   : 16   (Other): 69   (Other): 69     (Other): 17  
##        Class    
##  benign   :458  
##  malignant:241  
##                 
##                 
##                 
##                 
##                 
```

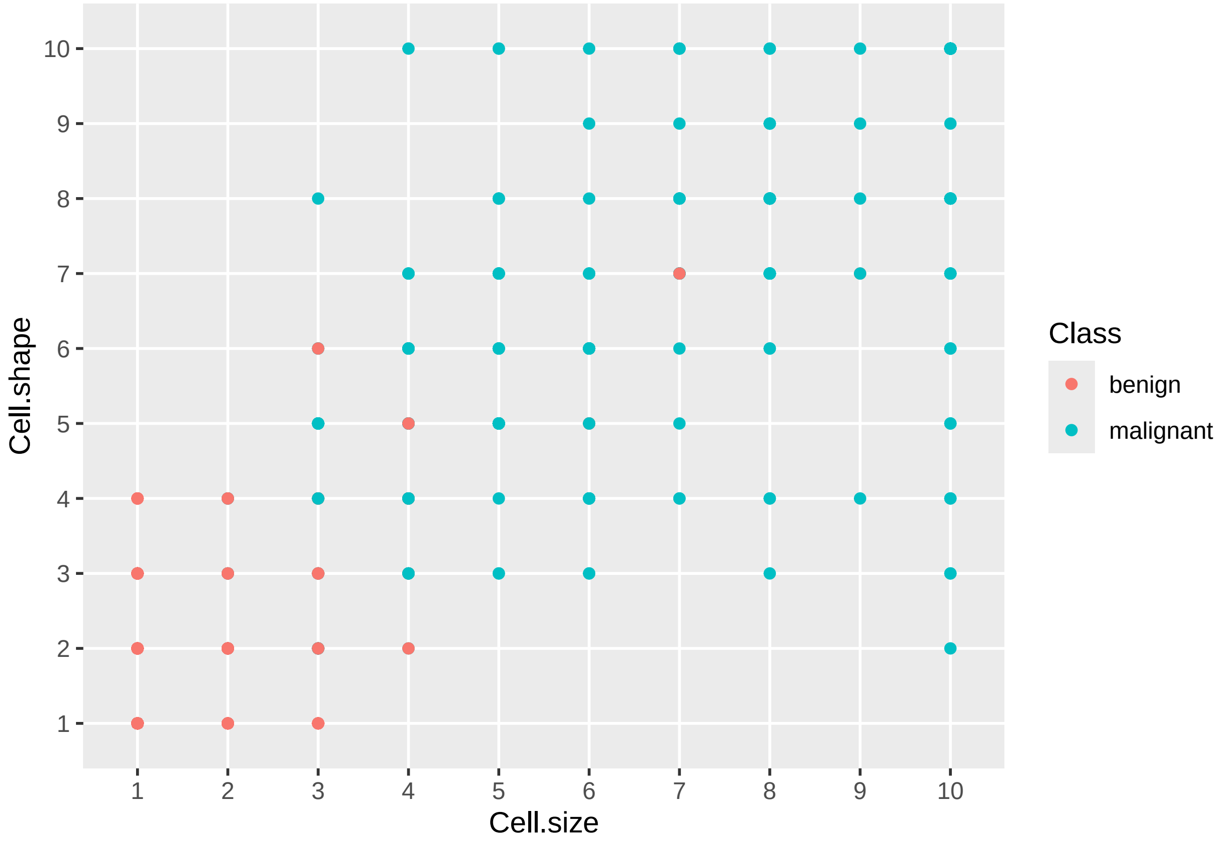
```r
library(ggplot2)
plot <- ggplot(data, aes(x=Class)) +geom_bar(fill = "red")
plot
```



```
plot2 <- ggplot(data, aes(x = Class, y = Cl.thickness)) + geom_boxplot(fill = "blue")
plot2
```



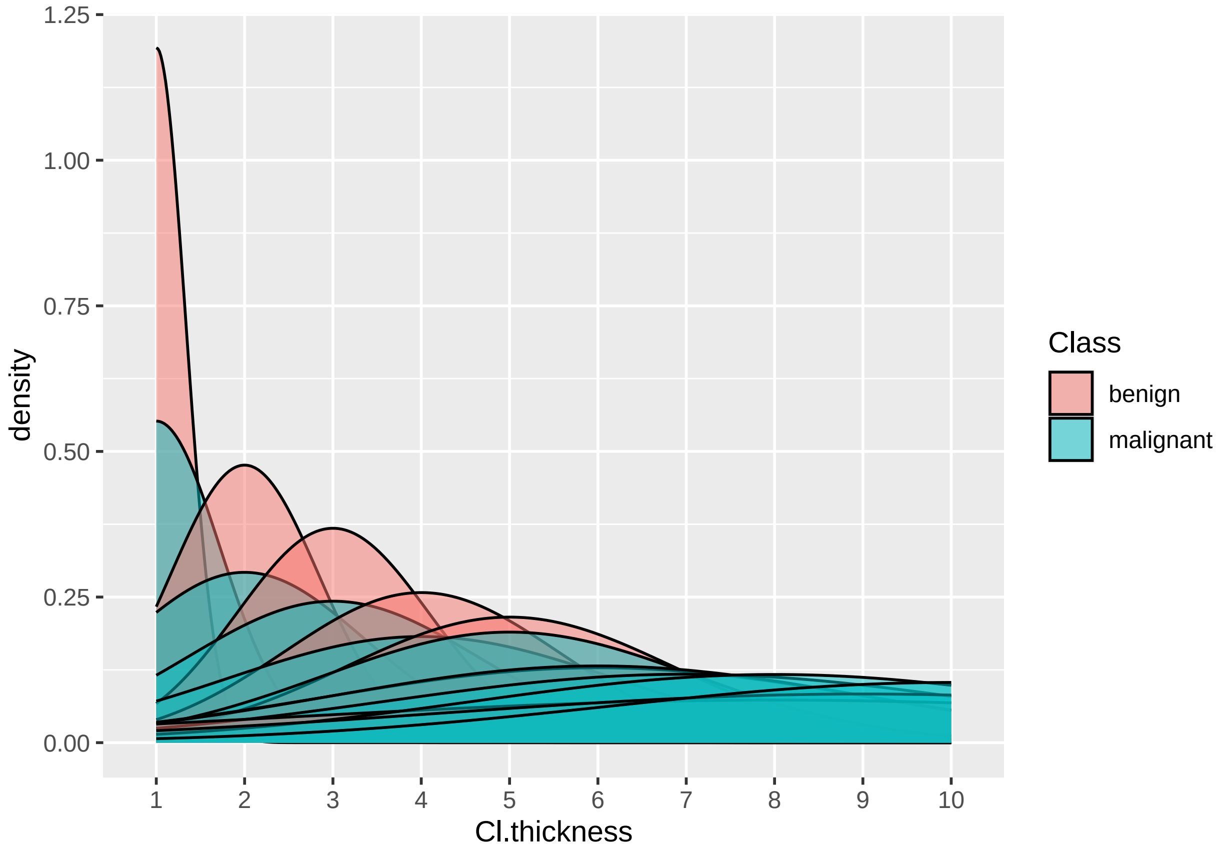
```
ggplot(BreastCancer, aes(x = Cell.size, y = Cell.shape, color = Class)) +
  geom_point()
```



```r
ggplot(BreastCancer, aes(x = Cl.thickness, fill = Class)) +
  geom_density(alpha = 0.5)
```

```
## Warning: Groups with fewer than two data points have been dropped.
```

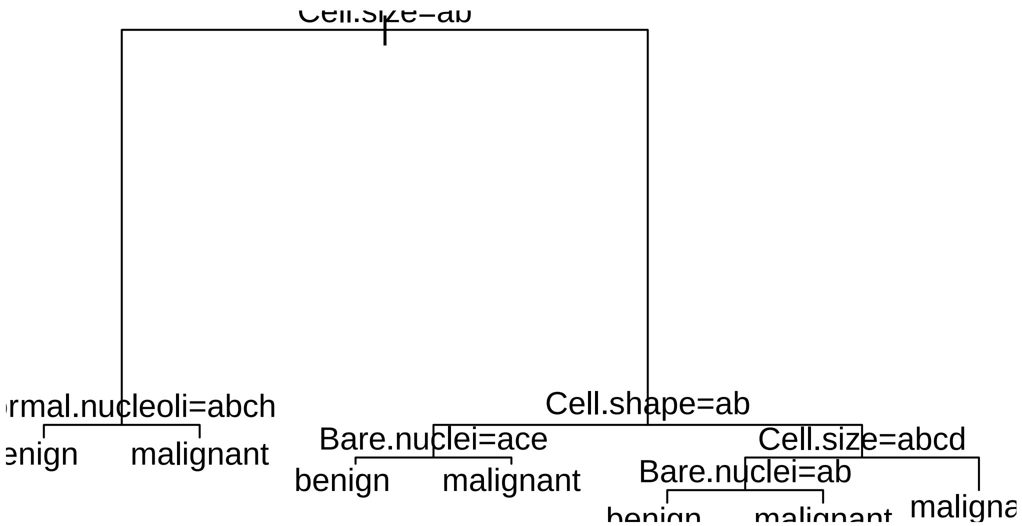
```
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf
```



```
library(rpart)
```

```
## Warning: package 'rpart' was built under R version 4.4.2
```

```
model_rpart <- rpart(Class ~ . - Id, data = data, method = "class")
plot(model_rpart)
text(model_rpart)
```



```r
library(caret)
```

```
## Loading required package: lattice
```

```r
predictions_rpart <- predict(model_rpart, BreastCancer, type = "class")
confusionMatrix(predictions_rpart, BreastCancer$Class)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  benign malignant
##   benign       442         9
##   malignant     16       232
##
##                Accuracy : 0.9642
##                  95% CI : (0.9477, 0.9767)
##     No Information Rate : 0.6552
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9214
##
##  Mcnemar's Test P-Value : 0.2301
##
##             Sensitivity : 0.9651
##             Specificity : 0.9627
##          Pos Pred Value : 0.9800
##          Neg Pred Value : 0.9355
##              Prevalence : 0.6552
##          Detection Rate : 0.6323
##    Detection Prevalence : 0.6452
##       Balanced Accuracy : 0.9639
##
##        ’Positive’ Class : benign
##
```

```
library(e1071)
model_nb <- naiveBayes(Class ~ . - Id, data = BreastCancer)
predictions_nb <- predict(model_nb, BreastCancer)
confusionMatrix(predictions_nb, BreastCancer$Class)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  benign malignant
##   benign       443         3
##   malignant     15       238
##
##                Accuracy : 0.9742
##                  95% CI : (0.9596, 0.9847)
##     No Information Rate : 0.6552
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9437
##
##  Mcnemar’s Test P-Value : 0.009522
##
##             Sensitivity : 0.9672
##             Specificity : 0.9876
##          Pos Pred Value : 0.9933
##          Neg Pred Value : 0.9407
##              Prevalence : 0.6552
##          Detection Rate : 0.6338
##    Detection Prevalence : 0.6381
##       Balanced Accuracy : 0.9774
##
##        ’Positive’ Class : benign
##
```

```
library(caret)
library(rpart)
library(e1071)
set.seed(123)
data <- BreastCancer
predictors <- data[, -which(names(data) == "Class")]
target <- data$Class
data <- data.frame(predictors, Class = target)
train_control <- trainControl(method = "cv", number = 5)
```

```
dataset <- na.omit(data)

rpart_model <- train(Class ~ ., data = dataset, method = "rpart",
                     trControl = train_control)
print(rpart_model)
```

```
## CART
##
## 683 samples
##  10 predictor
##   2 classes: 'benign', 'malignant'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 546, 546, 547, 547, 546
## Resampling results across tuning parameters:
##
##   cp          Accuracy   Kappa
##   0.02092050  0.9385144  0.8645561
##   0.05439331  0.9267497  0.8383238
##   0.79079498  0.8552168  0.6349432
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.0209205.
```

```
nb_model <- train(Class ~ ., data = dataset, method = "naive_bayes",
                  trControl = train_control)
print(nb_model)
```

```
## Naive Bayes
##
## 683 samples
##  10 predictor
##   2 classes: 'benign', 'malignant'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 546, 547, 546, 547, 546
## Resampling results across tuning parameters:
##
##   usekernel  Accuracy   Kappa
##   FALSE      0.3499249  0
##    TRUE      0.6500751  0
##
## Tuning parameter 'laplace' was held constant at a value of 0
## Tuning
##  parameter 'adjust' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were laplace = 0, usekernel = TRUE
##  and adjust = 1.
```

```r
results <- resamples(list(rpart = rpart_model, naive_bayes = nb_model))
summary(results)
```

```
##
## Call:
## summary.resamples(object = results)
##
## Models: rpart, naive_bayes
## Number of resamples: 5
##
## Accuracy
##                  Min.  1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## rpart       0.9197080 0.919708 0.9264706 0.9385144 0.9558824 0.9708029    0
## naive_bayes 0.6470588 0.649635 0.6496350 0.6500751 0.6496350 0.6544118    0
##
## Kappa
##                  Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## rpart       0.8210426 0.8277517 0.8341059 0.8645561 0.9034091 0.9364711    0
## naive_bayes 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000    0
```