SVM Classification

Jeffrey Li, Valari Graham

2022-10-19

Dataset

This data set was downloaded from the UCI Machine Learning Repository. The link below can be used to access the data. Occupancy Dataset

```
df <- read.csv("data.csv")
df <- df[, c(2:7)]
df$Occupancy <- factor(df$Occupancy)
```

Split into Traing/Test Data

The data was split into training and testing data for exploration and further analysis.

```
set.seed(1234)
i <- sample(1:nrow(df), 0.80*nrow(df), replace = FALSE)
train <- df[i,]
test <- df[-i,]
```

Exploration on Training Data

Below is an exploration of the training data using functions and creating visual plots.

```
str(train)
```

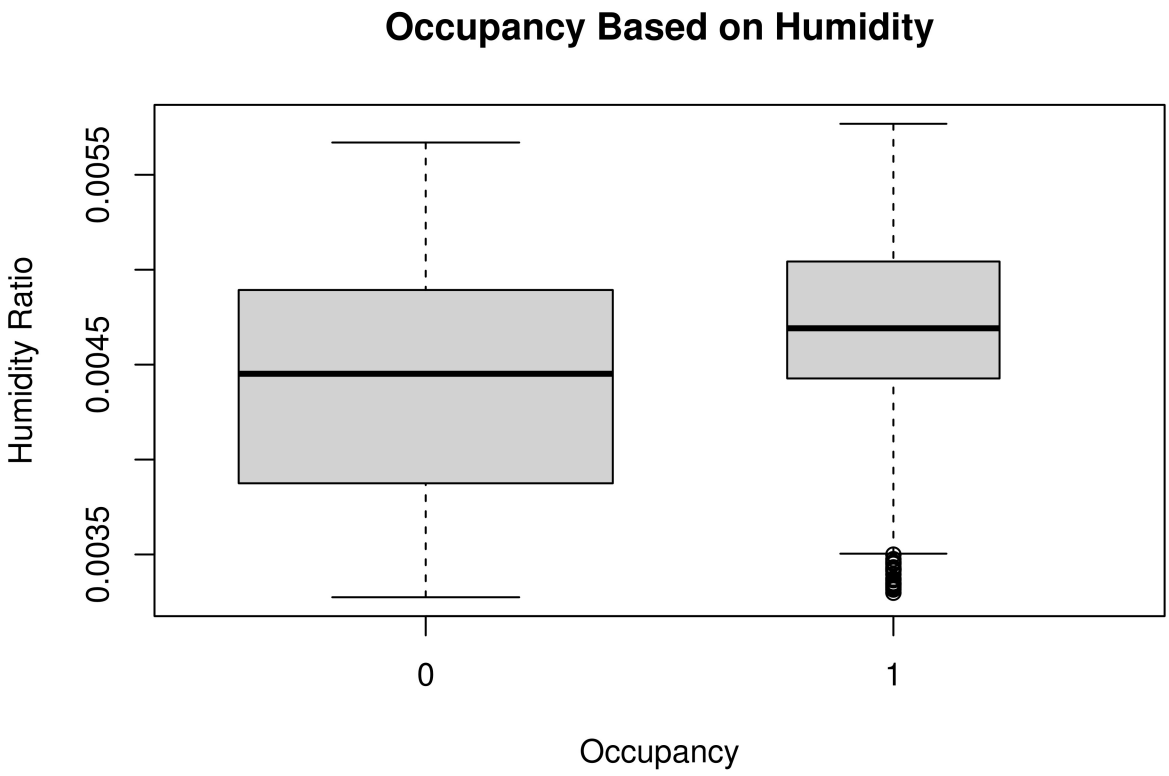
```
## 'data.frame':    9933 obs. of  6 variables:
##  $ Temperature  : num  20.9 21.4 22.6 21.8 21 ...
##  $ Humidity     : num  24.7 27.8 24.9 28.1 25.4 ...
##  $ Light        : num  0 0 732 0 14 0 0 454 433 0 ...
##  $ CO2          : num  572 566 588 594 522 ...
##  $ HumidityRatio: num  0.00377 0.00438 0.00423 0.00453 0.0039 ...
##  $ Occupancy    : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 2 2 1 ...
```

```
head(train)
```

```
##      Temperature Humidity Light      CO2 HumidityRatio Occupancy
## 7452    20.89000   24.695     0 571.5000   0.003768244         0
## 8016    21.39000   27.790     0 566.0000   0.004376975         0
## 7162    22.63333   24.890   732 587.6667   0.004228062         1
## 8086    21.79000   28.050     0 594.5000   0.004528488         0
## 7269    21.00000   25.390    14 522.0000   0.003901421         0
## 9196    20.60000   26.890     0 498.0000   0.004032247         0
```

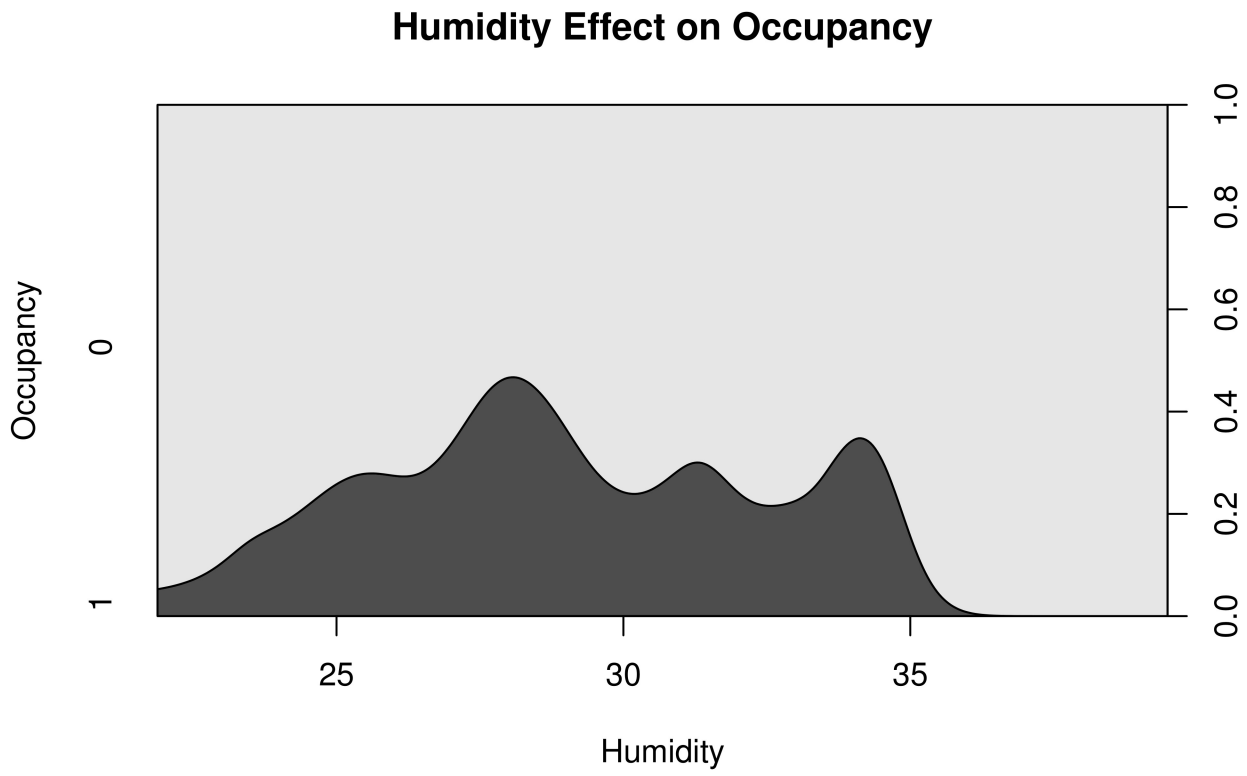
Data visualization - Box Plot

```
plot(HumidityRatio~Occupancy, data = train, main = "Occupancy Based on Humidity", xlab = 'Occupancy', y
```



Data visualization - Conditional Density Plot

```
cdplot(train$Occupancy~train$Humidity, main = 'Humidity Effect on Occupancy', xlab = 'Humidity', ylab =
```



Linear SVM

Try linear kernel with a cost of 1.

```
library(e1071)
svm1 <- svm(Occupancy~., data = train, kernel = "linear", cost = 1, scale = TRUE)
summary(svm1)
```

```
##
## Call:
## svm(formula = Occupancy ~ ., data = train, kernel = "linear", cost = 1,
##     scale = TRUE)
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  1
##
## Number of Support Vectors:  353
##
##  ( 176 177 )
##
##
```

```
## Number of Classes:  2
##
## Levels:
##  0 1
```

Evaluate Linear SVM

Show results of linear SVM using confusion matrix and accuracy of the model.

```
pred <- predict(svm1, newdata = test)
table(pred, test$Occupancy)
```

```
##
## pred    0    1
##    0 1871    1
##    1   13  599
```

```
mean(pred == test$Occupancy)
```

```
## [1] 0.9943639
```

Polynomial SVM

Try polynomial kernel with a cost of 10

```
svm2 <- svm(Occupancy~., data = train, kernel = 'polynomial', cost = 10, scale = TRUE)
summary(svm2)
```

```
##
## Call:
## svm(formula = Occupancy ~ ., data = train, kernel = "polynomial",
##     cost = 10, scale = TRUE)
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  polynomial
##        cost:  10
##      degree:  3
##      coef.0:  0
##
## Number of Support Vectors:  490
##
##  ( 244 246 )
##
##
## Number of Classes:  2
##
## Levels:
##  0 1
```

Evaluate SVM

Show results of polynomial SVM using confusion matrix and accuracy of the model.

```
pred2 <- predict(svm2, newdata = test)
table(pred2, test$Occupancy)
```

```
##
## pred2    0    1
##     0 1871    2
##     1   13  598
```

```
mean(pred2 == test$Occupancy)
```

```
## [1] 0.9939614
```

Radial SVM

Try radial kernel with a cost of 100

```
svm3 <- svm(Occupancy~., data = train, kernel = "radial", cost = 100, gamma = 0.5, scale = TRUE)
summary(svm3)
```

```
##
## Call:
## svm(formula = Occupancy ~ ., data = train, kernel = "radial", cost = 100,
##     gamma = 0.5, scale = TRUE)
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  radial
##        cost:  100
##
## Number of Support Vectors:  274
##
##  ( 131 143 )
##
##
## Number of Classes:  2
##
## Levels:
##  0 1
```

Evaluate Radial SVM

Show results of radial SVM using confusion matrix and accuracy of the model.

```r
pred3 <- predict(svm3, newdata = test)
table(pred3, test$Occupancy)
```

```
##
## pred3    0    1
##     0 1871    6
##     1   13  594
```

```r
mean(pred3 == test$Occupancy)
```

```
## [1] 0.992351
```

Compare the Results from the Kernels

The results are as follows, the linear model had an accuracy of 98.5%, the polynomial model had an accuracy of 98.3%, and the radial model had an accuracy of 98.6%. Overall, the radial kernel performed slightly better than the other models. The models performed very well on the test data.