House Occupancy Prediction

Muhammad Hamzah

Contents


```
set.seed(151)
library("knitr")
library("pander")
library("readr")
library("magrittr")
library("car")
library("MASS")
library("klaR")
library("tree")
library("rpart")
library("rpart.plot")
```

```
occupancy_train <- readr::read_csv("occupancy_train.csv")
occupancy_test <- readr::read_csv("occupancy_test.csv")
```

Introduction

Occupancy detection has emerged as a crucial component in modern building management, directly impacting energy efficiency and safety protocols. Studies estimate that occupancy-based control systems can reduce energy usage in buildings by up to 30%. This project aims to harness machine learning techniques to predict room occupancy using environmental factors such as temperature, humidity, CO2 levels, and time of day. Utilizing the dataset from "Accurate occupancy detection of an office room from light, temperature, humidity, and CO2 measurements using statistical learning models" by Luis M. Candanedo and Véronique Feldheim, we will train models using 'occupancy_train.csv' and evaluate their performance with 'occupancy_test.csv'.

The objective is to develop a reliable classifier for real-time occupancy detection, a critical tool for enhancing building energy efficiency and ensuring rapid response in emergencies.

Exploratory Data Analysis

Background and Variables

The 'occupancy_train' dataset, central to this project, is a comprehensive collection of environmental readings aimed at detecting room occupancy.

The dataset encompasses the following predictor variables:

- **Temperature:**The ambient temperature of the room.(In Degree Celsius)
- **Humidity:** The relative humidity present in the room.(As Percentage)
- **CO2:** Indicates the concentration of carbon dioxide in the room.(Represented in parts per million **(ppm)**)
- **Hour:** Hour of the day when the measurement was taken(range from 0 to 23)

The response variable, which we will aim to predict, is:

- **Occupancy:** This is a binary variable, where '0' indicates that the room is not occupied, and '1' signifies that the room is occupied.

Summary of the Response Label in the Training Dataset

```
table(occupancy_train$Occupancy)
```

```
##
##    0    1
## 4497 1203
```

```
prop.table(table(occupancy_train$Occupancy))
```

```
##
##         0         1
## 0.7889474 0.2110526
```

- There are a total of 5,700 observation in the training set
- There are 4,497 unoccupied rooms, which comprise 78.89% of the Training Dataset
- There are 1,203 occupied rooms, which comprise 21.1% of the Training Dataset

EDA on relationships between Occupancy and Quantitative Variables

In order to better understand whether the quantitative variables will be useful in predicting the Occupancy Status of the room, we will be using boxplots, which will help us visualize the relationship between the response(Occupancy) and the predictors

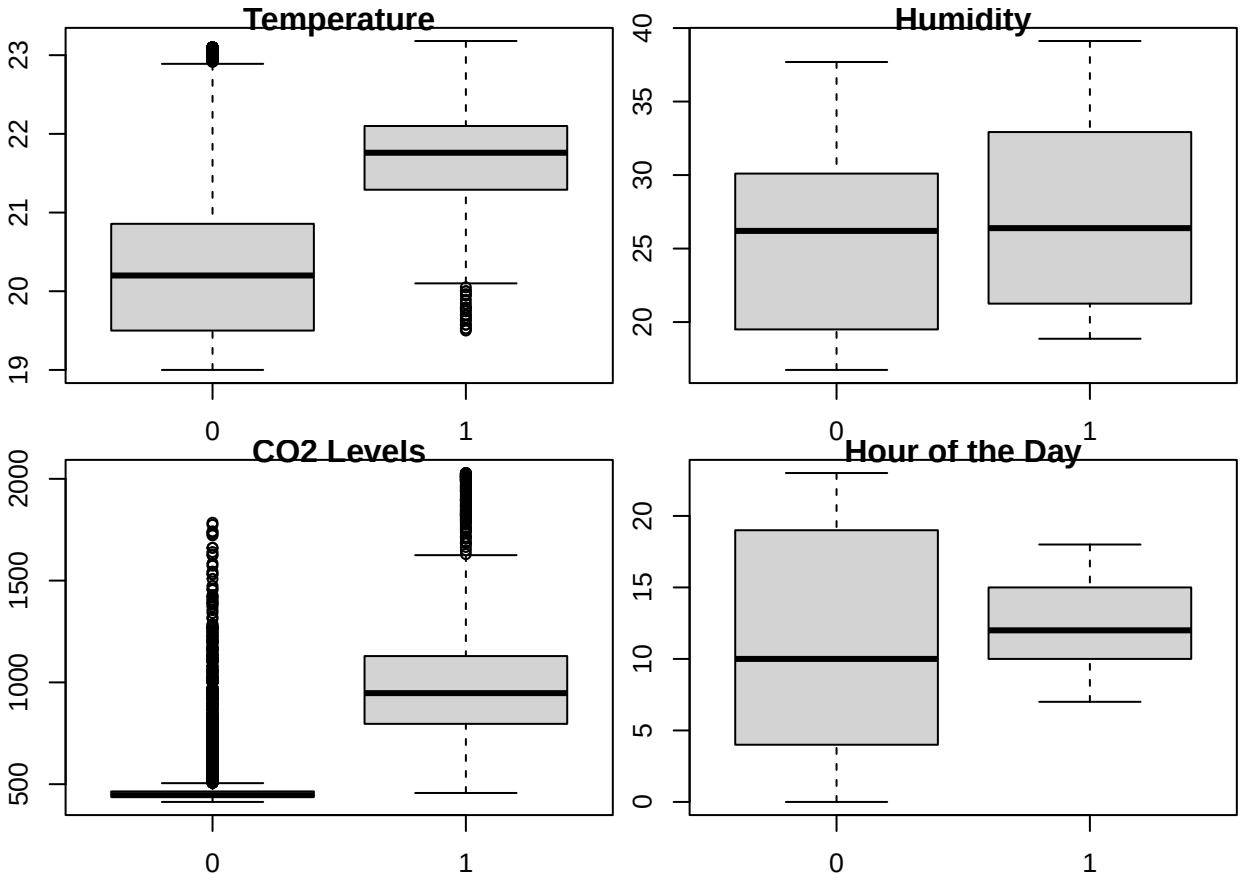
```
# Set the plotting parameters
par(mfrow = c(2, 2), mai = c(0.3, 0.3, 0.1, 0.1))


# Boxplot for Temperature vs Occupancy
boxplot(Temperature ~ as.factor(Occupancy),
        main = "Temperature",
        data = occupancy_train,
        xlab = "Occupancy",
        ylab = "Temperature (°C)")
```

```r
# Boxplot for Humidity vs Occupancy
boxplot(Humidity ~ as.factor(Occupancy),
        main = "Humidity",
        data = occupancy_train,
        xlab = "Occupancy",
        ylab = "Humidity (%)")

# Boxplot for CO2 vs Occupancy
boxplot(CO2 ~ as.factor(Occupancy),
        main = "CO2 Levels",
        data = occupancy_train,
        xlab = "Occupancy",
        ylab = "CO2 (ppm)")

# Boxplot for Hour vs Occupancy
boxplot(Hour ~ as.factor(Occupancy),
        main = "Hour of the Day",
        data = occupancy_train,
        xlab = "Occupancy",
        ylab = "Hour")
```



We note here that if the boxplots are showing differences in whether the place is shown as occupied or not, gives us certain idea that there is some relationship and the variable might be useful in the classifier, however, this is not a definite way of saying it to be a statistically significant relationship

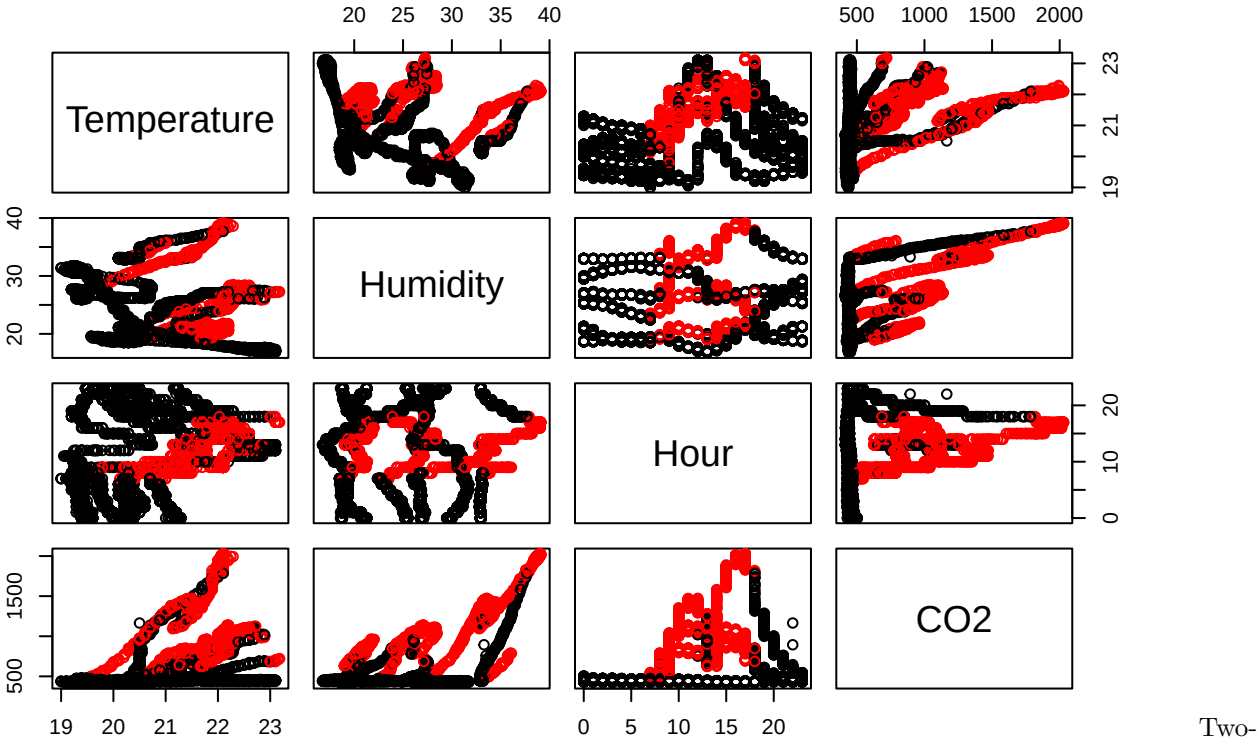
With that in mind, we note that Occupied rooms have higher **median** temperatures and humidity levels.

Moreover, CO2 levels are markedly higher in occupied rooms, with a substantial number of outliers indicating peak concentrations, making it a potentially strong predictor for occupancy.

However, The hour of the day, while covering a full range for both states, does not display a significant median difference, implying a less direct relationship with occupancy. These findings offer a basis to expect that temperature, humidity, and CO2 levels could be influential in classifying room occupancy

Some visual EDA on classification pairs

```
pairs(occupancy_train[ , c(1,2,3,4)],
col=ifelse(occupancy_train$Occupancy=="1","red","black"))
```



Two-dimensional pair plots give us insight into how combinations of variables may help differentiate between occupied (red circles) and unoccupied (black circles) rooms. While some variable pairs show overlap between the two categories, several combinations exhibit patterns that could be promising for classification:

- **Temperature and CO2:** Visible separation between occupied and unoccupied rooms, with higher CO2 levels often coinciding with higher temperatures, which could be indicative of occupancy.
- **Humidity and CO2:** Higher humidity levels combined with higher CO2 concentrations appear to be associated with occupied rooms, suggesting this pair may be useful for classification.
- **Hour and CO2:** There is a mild separation, especially during certain hours where higher CO2 levels may signify occupancy.
- **Temperature and Humidity:** Although there is some overlap, there are regions in the plot where occupied rooms tend to cluster, suggesting a relationship between these variables when the room is in use.

The plot also reveals some outliers, particularly in CO2 levels, which may need further investigation to understand their impact on the classification models.

It should be noted, though, that our analysis has been limited to single or pairs of variables, suggesting that the actual relationship in a higher-dimensional space may be more complex.

Modeling

We now turn to building and assessing our classifiers for predicting whether the room is occupied or not

Our four classifiers are:

- Linear Discriminant Analysis **(LDA)**
- Quadratic Discriminant Analysis **(QDA)**
- Classification trees
- Binary logistic regression

To ensure that our models are not overfitting to our sample, we randomly split our observations into training and test sets. All four models were built using the same training observations and assessed on the same set of test observations

Linear Discriminant Analysis (LDA)

For LDA models we only use Quantitative variables(in our case all of them are quantitative)

```
occupancy.lda <-  lda( factor(Occupancy) ~ Temperature + Humidity + CO2 + Hour,
                       data = occupancy_train)
```

Next, we will investigate the performance of the LDA classifier on our test data as follows:

```
occupancy.lda.pred <- predict(occupancy.lda,
                              as.data.frame(occupancy_test))
```

```
table(occupancy.lda.pred$class, occupancy_test$Occupancy)
```

```
##
##        0    1
##   0 1844  111
##   1   73  415
```

- On the test data, LDA gave an overall error rate of (73+111)/2443 = 0.075 which is quite low.
- The error rate for the rooms that were occupied is (111/586) is 0.21
- The error rate for rooms that were not occupied is (73/1917) is 0.038

Hence, the error rates indicate that while the classifier performs well overall, it is better at predicting not occupied (0) statuses compared to occupied (1) statuses.

Quadratic Discriminant Analysis (QDA)

For QDA models we only use Quantitative variables(in our case all of them are quantitative)

```
occupancy.qda <- qda(factor(Occupancy) ~ Temperature + Humidity + CO2 + Hour,
                     data=occupancy_train)
```

Next, we will investigate the performance of the QDA classifier on our test data as follows:

```
occupancy.qda.pred <- predict(occupancy.qda,
                        as.data.frame(occupancy_test))
```

```
table(occupancy.qda.pred$class, occupancy_test$Occupancy)
```

```
##
##        0    1
##   0 1832   81
##   1   85  445
```

Using QDA, we could anticipate a somewhat improved performance compared to LDA, as QDA possesses greater flexibility in identifying nonlinear and curved decision boundaries.

- Indeed, our results in the QDA show a slight decrease in overall error rate: (85+81)/2443 = 0.0679.
- The error rate for the rooms that were occupied is (81/526) is 0.15
- The error rate for rooms that were not occupied is (85/1917) is 0.044, which is slighlty higher than the LDA

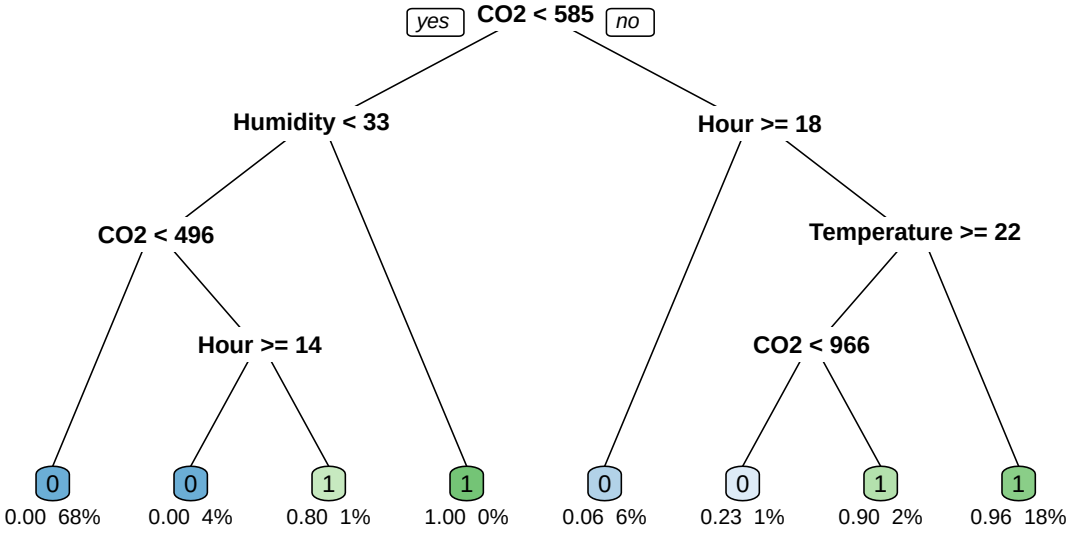
Hence, as before in the case of LDA the error rates indicate that while the classifier performs well overall, it is better at predicting not occupied (0) statuses compared to occupied (1) statuses.

Classification Trees

We fit a classification tree on the training data and plot it, as follows:

```
occupancy.tree <- rpart(factor(Occupancy) ~  Temperature + Humidity + CO2 + Hour,
                        data=occupancy_train,
                        method="class")
```

```
rpart.plot(occupancy.tree,
            type = 0,
            clip.right.labs = FALSE,
            branch = 0.1,
            under = TRUE)
```



Tthe "most important" variables that the tree determines for classification is is indicated from top down on the tree, and in this case the tree deteremined CO2 to be the "most imporatant one"

```
occupancy.tree.pred <- predict(occupancy.tree,
                               as.data.frame(occupancy_test),
                               type="class")
```

```
table(occupancy.tree.pred, occupancy_test$Occupancy)
```

```
##
## occupancy.tree.pred    0    1
##                   0 1883   15
##                   1   34  511
```

- Our results in the Classification Tree show a decrease in overall error rate: (34+15)/2443 = 0.02.
- The error rate for the rooms that were occupied is (15/526) is almost 0.028

- The error rate for rooms that were not occupied is (34/1917) is 0.017

The error rate in this case is very low as compared to QDA and LDA, and similarly for the occupied and unoccupied rooms. As in the Case of LDA and QDA, it perfmroms better in predicting rooms which are not occupied

However, considering the very minimal error rate, there could be a slight concern of overfitting, and hence being unstable in other cases outside of the test data. One way which we can remedy is this by pruning the tree or of more data can be found then using random forests.

Binary Logistic Regression

Finally, we consider binary logistic regression to model whether the room is occupied or not. Similarly to the classification trees, here too we use all the variables

We train a logistic classifier using the training data and then examine the resulting confusion matrix from the test data in the following manner:

First, we will fit a binary logistic regression model,

```
occupancy.logit <- glm(factor(Occupancy) ~  Temperature + Humidity + CO2 + Hour,
                  data = occupancy_train,
                  family = binomial(link = "logit"))
```

Now we will apply it on the test data

```
occupancy.logit.prob <- predict(occupancy.logit,
                                as.data.frame(occupancy_test),
                                type = "response")
```

Given that the logistic model, when applied to the test data, produces probabilities for the classification of occupied versus not occupied, we will transform these logistic probabilities into classification predictions. This will be done by setting a threshold for the probability; if the probability is greater than 0.5, it will be classified as occupied, otherwise, it will be classified as not occupied.

To correctly link the probability direction with the respective Occupancy status, we must determine the default order of "Occupancy". This can be accomplished by applying the "levels" function to the factored response variable, as shown below:

```
levels(factor(occupancy_test$Occupancy))
```

```
## [1] "0" "1"
```

Next, we derive the test classifications from the logistic model using a threshold probability of 0.5, in the following way:

```
occupancy.logit.pred <-ifelse(occupancy.logit.prob > 0.5,"1","0")
```

We then assess the performance of the logistic classifier on our test data by examining the confusion matrix, as illustrated below:

```
table(occupancy.logit.pred, occupancy_test$Occupancy)
```

```
##
## occupancy.logit.pred    0    1
##                    0 1849   96
##                    1   68  430
```

- The logistic model as a classifier (using threshold probability of 0.5) performs poorer than Classification Tree with overall error as 0.067
- The error rate for the rooms that were occupied is (96/526) is almost 0.18

- The error rate for rooms that were not occupied is (34/1917) is 0.035

Overall we can see that the binary logistic model performs poorer as compared to the Classification Tree in all possible ways.However, like all the other models, it too performs better in predicting the non-occupied rooms compared to the occupied room.

Final Recommendation

- Of the four classifiers we tested, the classification tree performed the best.
- QDA and Binary Logistic Regression had almost similar performances however, the QDA was slightly better at predicting which rooms are not occupied
- LDA performed the poorest from all the models, with it being significantly poorer compared to the Classification tree

Hence, the final recommendation is to use the Classification Tree, however, since it did have a very low error rate, there is a risk of it actually overfitting. If this is indeed an issue, then either QDA or Binary Logistic classified seem to be a good secondary recommendation as they can be potentially more stable.

Discussion

Overall, our models did well at classifying whether or not a room is occupied.However, as a caution we do note that the Classification Tree could have the risk of overfitting, considering its very low overall error rate.In order to remedy this issue of overfitting we could potentially use a random forest, if enough data can be found.

Other areas for future research that could be of greater interest to the industry would be to build models to predict the multinomial quality variable. If a combination of other measurements, along with other factors like (e.g. population density) could be found that corresponds to higher or lower occupancy rates, it could help city planners as well as emergency services to better understand and formulate strategies for evacuation as well as developmental plans.