Predicting Room Occupancy

Jonathan Lai jklai

Due Wed, November 30, at 11:59PM

Contents

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

Introduction

Rooms do not have labels that show the occupancy status (except for airplane toilets). But during emergencies, rooms in buildings must be evacuated, and the rooms don't indicate whether rooms are occupied. So first-respondents and people trying to help cannot identify the occupancy of a room. To increase the efficiency of the evacuation process, identifying a room's occupancy status is imperative. A study by Luis M. Candanedo and Véronique Feldheim in the Energy Buildings journal collected data on a few different predictors that can be measured remotely inside the room. By analyzing those predictors, we can produce a classifier model that hopefully generates accurate predictions of whether a room is occupied or empty. In the future, if a classifier model is able to accurately predict the occupancy of a room during emergencies, evacuation would be done swiftly and potentially save more lives by collecting data on only a few measurements.

Exploratory Data Analysis

Background

5700 observations were collected from the training data set. There are 5 variables in total: 4 potential predictors/explanatory variables and 1 response variable which is the occupancy of the room.

Variable Descriptions:

- `Occupancy` refers to the occupancy status of a room (1 for occupied, 0 for vacant or not occupied)
- `Temperature` refers to the room temperature (in degrees Celsius)
- `Humidity` refers to the room relative humidity (in percent)
- `CO2` refers to the room's carbon dioxide level (in ppm)
- `Hour` refers to the hour of the day (ranges from 0 to 23; 0 as 12:00 am and 23 as 11:00 pm)

To understand and get a general idea of the data that is being explored, below are the first and last few samples in the dataset:

```
head(occ_tr)
```

```
## # A tibble: 6 x 5
##   Temperature Humidity  Hour   CO2 Occupancy
##         <dbl>    <dbl> <dbl> <dbl>     <dbl>
## 1        21.4     25.7    22  486          0
## 2        20.8     19.6    19  547          0
## 3        19.3     31.2     9  431.         0
## 4        19.2     31.2     7  431          0
## 5        20.3     32.9     3  452          0
## 6        20.4     18.6     5  433          0
```

```
tail(occ_tr)
```

```
## # A tibble: 6 x 5
##   Temperature Humidity  Hour   CO2 Occupancy
##         <dbl>    <dbl> <dbl> <dbl>     <dbl>
## 1        21.1     25.5     0  446          0
## 2        19.4     31.1     3  440.         0
## 3        19.7     19.4     6  444          0
## 4        20.5     33.9    21  726          0
## 5        21.5     20.7    10  839          1
## 6        19.7     19.4     7  448.         0
```

The first data point describes that a room with a room temperature of 21.39 ºC, 25.70% relative humidity, and a carbon dioxide level of 486 ppm at 10:00 pm is not occupied.

The second last data point describes that a room with a room temperature of 21.5 ºC, 20.725% relative humidity, and a carbon dioxide level of 839 ppm at 10:00 am is occupied.

Univariate Analysis on Occupancy

Below is the proportion that rooms are occupied (Y = 1) in the training dataset.

```
table(factor(occ_tr$Occupancy))
```

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

```
1203/5700
```

```
## [1] 0.2110526
```

```
4497/5700
```

```
## [1] 0.7889474
```

1203 rooms out of the 5700 data observations collected were occupied, which is 21.11% of the rooms.

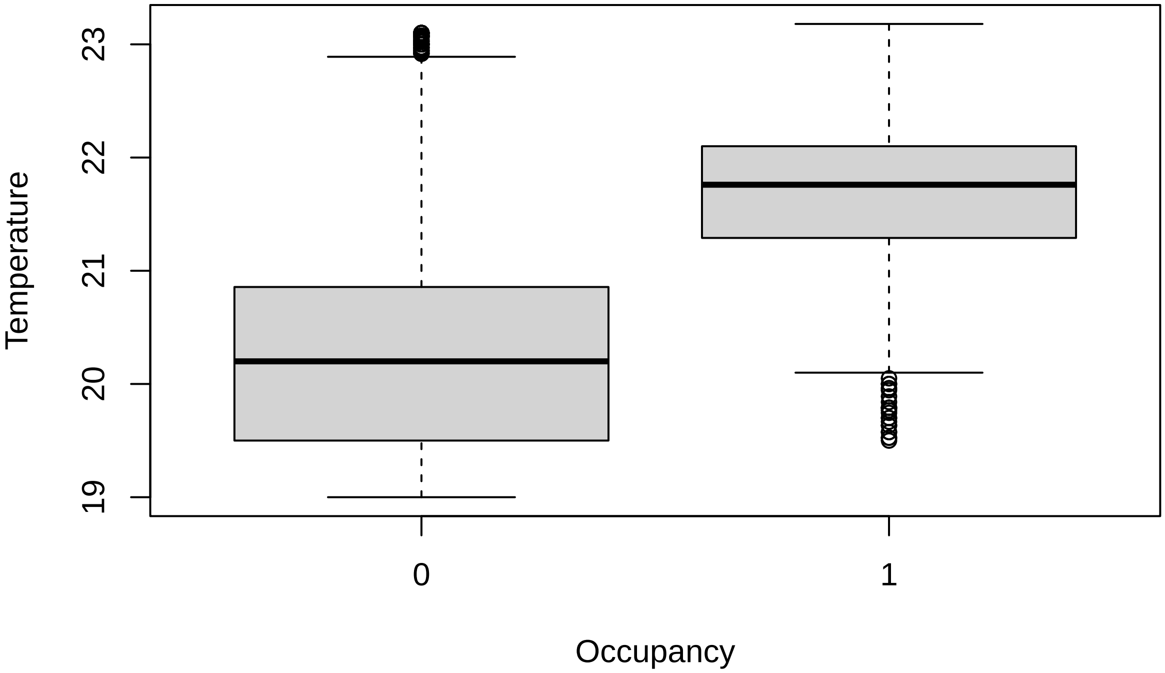
4497 rooms out of the 5700 data observations collected were not occupied, which is 78.89% of the rooms.

Bivariate Analysis

After analyzing the response variable `Occupancy` and discovering the proportions of rooms that are occupied and rooms that aren't occupied, we can explore the relationship between the categorical response variable `Occupancy` with the other predictors/explanatory variables.

Variable Temperature

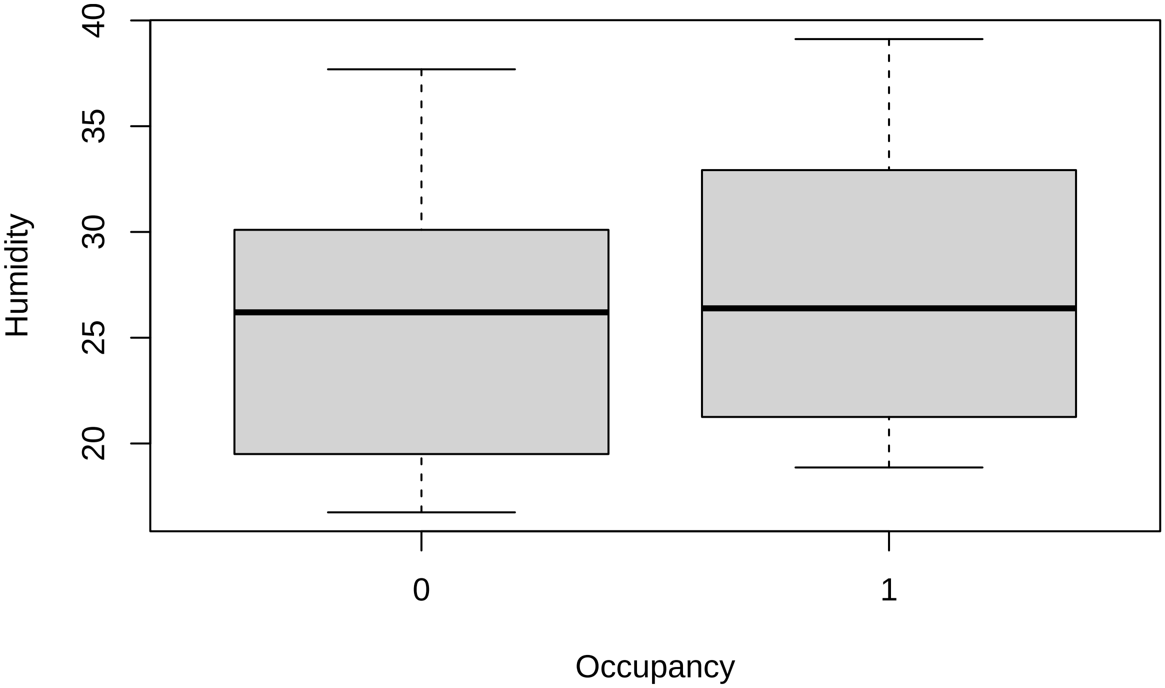
```
boxplot(Temperature ~ Occupancy, data = occ_tr)
```



The variable `Temperature` or room temperature appears to be associated with the `Occupancy` of the room. The median room temperature of the rooms that are not occupied is around 20.4 ºC, which is lower than that of the rooms that are occupied at 21.6 ºC. Though this difference in medians doesn't seem great, the lower quadrant (Q1) room temperature of rooms that are occupied is at 21.3ºC, which is higher than the upper quadrant (Q3) room temperature of rooms that are not occupied at around 20.8ºC. There are also many outlying room temperature data in both rooms that are occupied and those that aren't occupied.

Variable Humidity

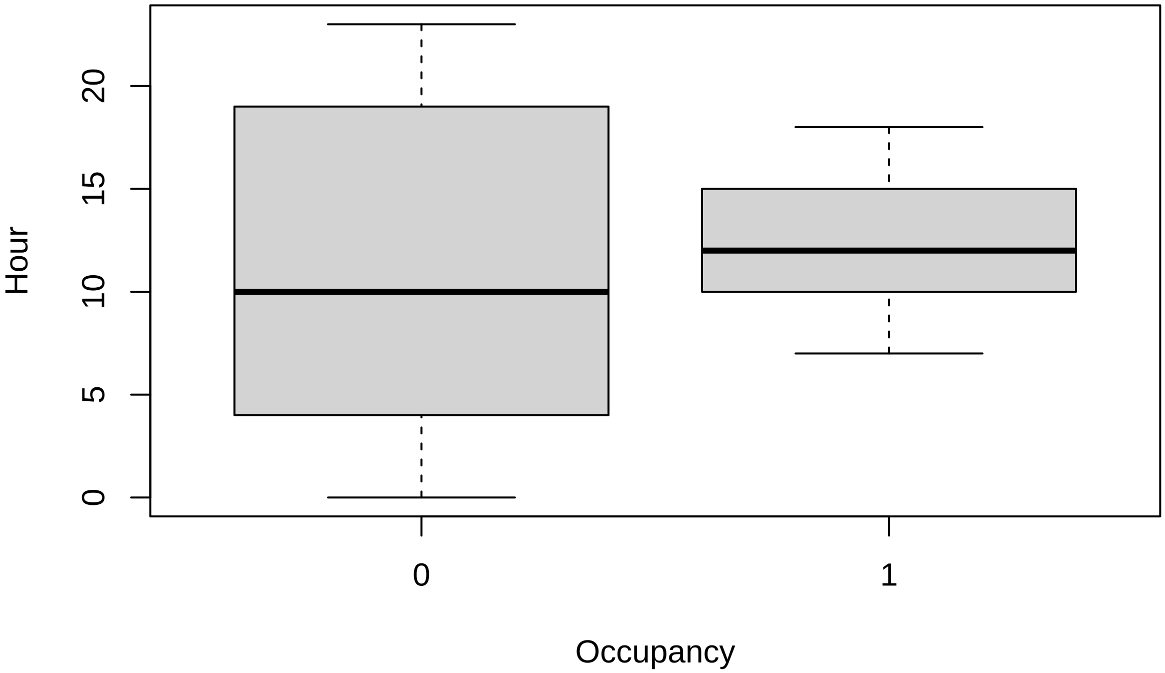
```
boxplot(Humidity ~ Occupancy, data = occ_tr)
```



The variable `Humidity` or room relative humidity appears to not be associated with the `Occupancy` of the room. The median room relative humidity for both rooms that are occupied and rooms that aren't occupied are at around 26%. The difference in medians doesn't seem great, the lower quadrant (Q1) room relative humidity of rooms that are occupied is at around 22%, which overlaps with the upper quadrant (Q3) room relative humidity of rooms that are not occupied at around 30%. There are no outlying room relative humidity data in both rooms that are occupied and those that aren't occupied.

Variable Hour

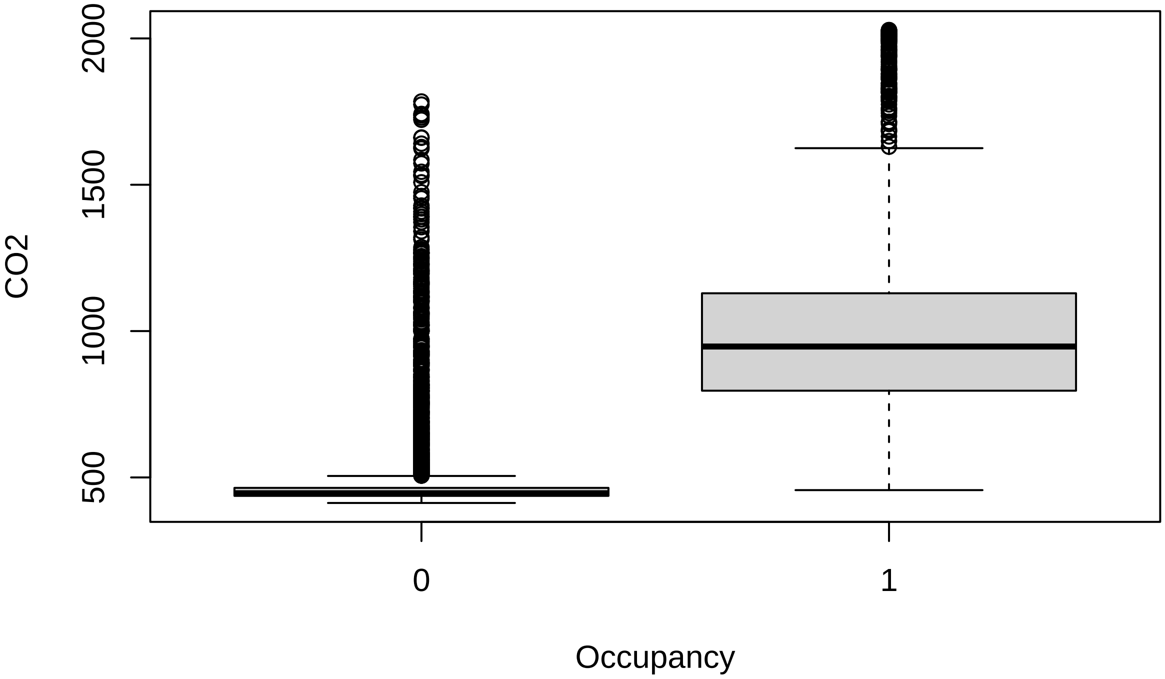
```
boxplot(Hour ~ Occupancy, data = occ_tr)
```



The variable `Hour` or the time of day appears to not be associated with the `Occupancy` of the room. The median time of day for rooms that are occupied is at around 12pm, which is later than the median time of day for rooms that aren't occupied at around 10 am. The difference in medians doesn't seem great, the lower quadrant (Q1) time of day for rooms that are occupied is at 10 am, which overlaps with the upper quadrant (Q3) time of day for rooms that are not occupied at around 6 pm. There are no outlying room relative humidity data in both rooms that are occupied and those that aren't occupied.

Variable CO2

```
boxplot(CO2 ~ Occupancy, data = occ_tr)
```

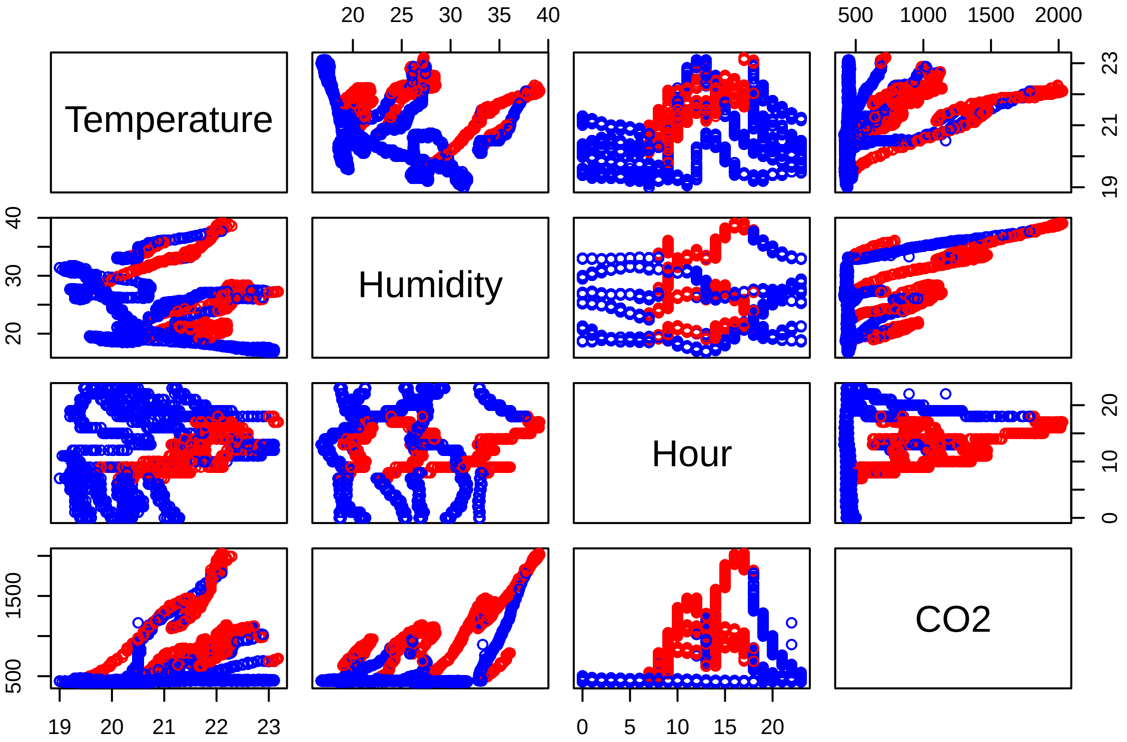


The variable `CO2` or room's carbon dioxide level appears to be associated with the `Occupancy` of the room. The median CO2 level for rooms that are not occupied is around 450 ppm, which is lower than that of the rooms that are occupied at 900 ppm. This difference in medians seem great; the lower quadrant (Q1) CO2 level of rooms that are occupied is at around 750 ppm, which is higher than the upper quadrant (Q3) CO2 levels of rooms that are not occupied at around 470 ppm. There are also many outlying CO2 level data in both rooms that are occupied and those that aren't occupied.

Pairs Plot Analysis

We want to understand which predictors are the most appropriate for predicting the `Occupancy` of the rooms. By producing a pairs plot, we can determine the pair of explanatory variables that most accurately classifies the data and distinguish the `Occupancy`. Below is the pairs plot:

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



The pairs plot generates all the possible combinations between all the predictors: `Temperature`, `Humidity`, `Hour`, `CO2`. In the pairs plot, the red circles represent occupied rooms and the blue circles represent rooms that are not occupied. If there is a separation between the different color circles, then the variables may be useful for classification. Specifically, the variable `Temperature` and `Humidity` appear to not be useful for classification because there doesn't seem to be a separation of data points when the variable `Temperature` or the variable `Humidity` is paired up with another explanatory variables. In this pairs plot, the variables `Hour` and `CO2` seem to be the most useful for classification because the separation of different-colored circles is the most obvious.

However, we need to be mindful that the pairs plot does not provide convincing and sufficient evidence to remove variables in our classifier model as the pairs plot only involves two explanatory variables; our classifier model tries to account for as many predictors as it can.

Modeling

We will explore which model will be the most appropriate for this data to classify whether rooms are occupied or not. The model is developed from the training data, and verified with the test data. The models involved are the linear discriminant analysis (LDA), the quadratic discriminant analysis (QDA), the classification tree, and the binary logistic regression model.

Discriminant Analysis

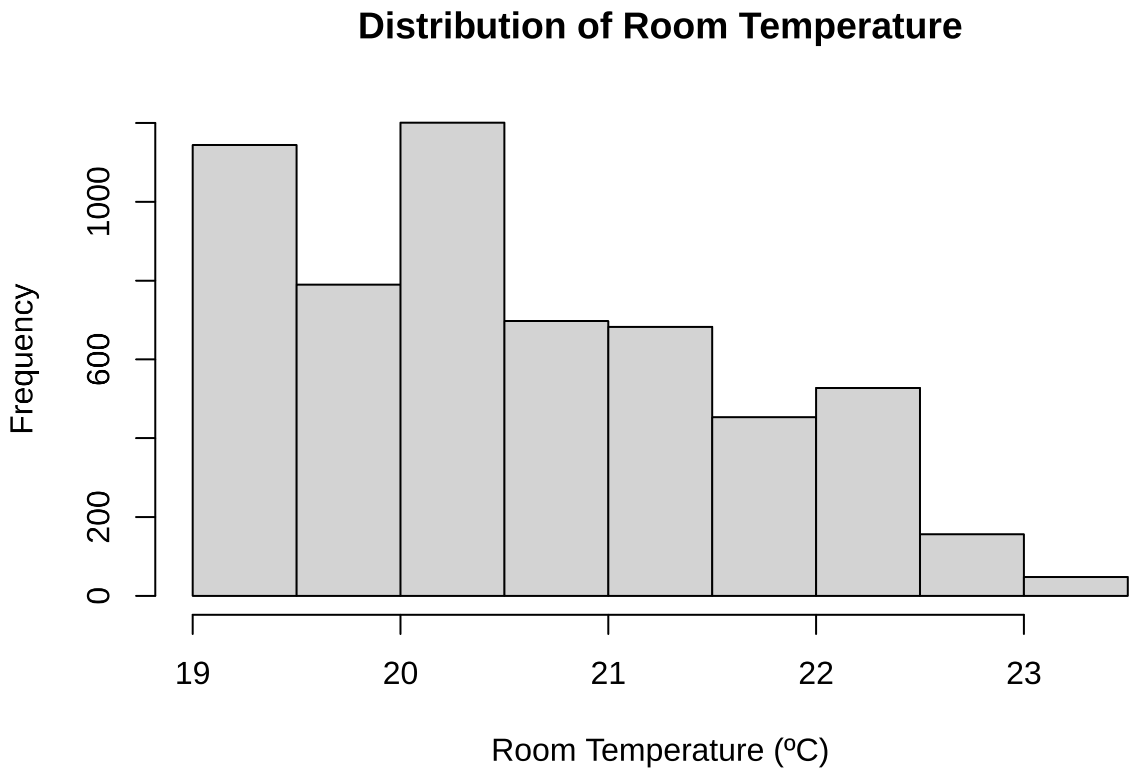
For discriminant analysis models, they can only use quantitative explanatory variables, which must also be Gaussian, meaning all the potential predictors must be normally distributed. Before creating the LDA and QDA, we must perform exploratory data analysis on each explanatory variable.

EDA on Explanatory Variables and Transformations

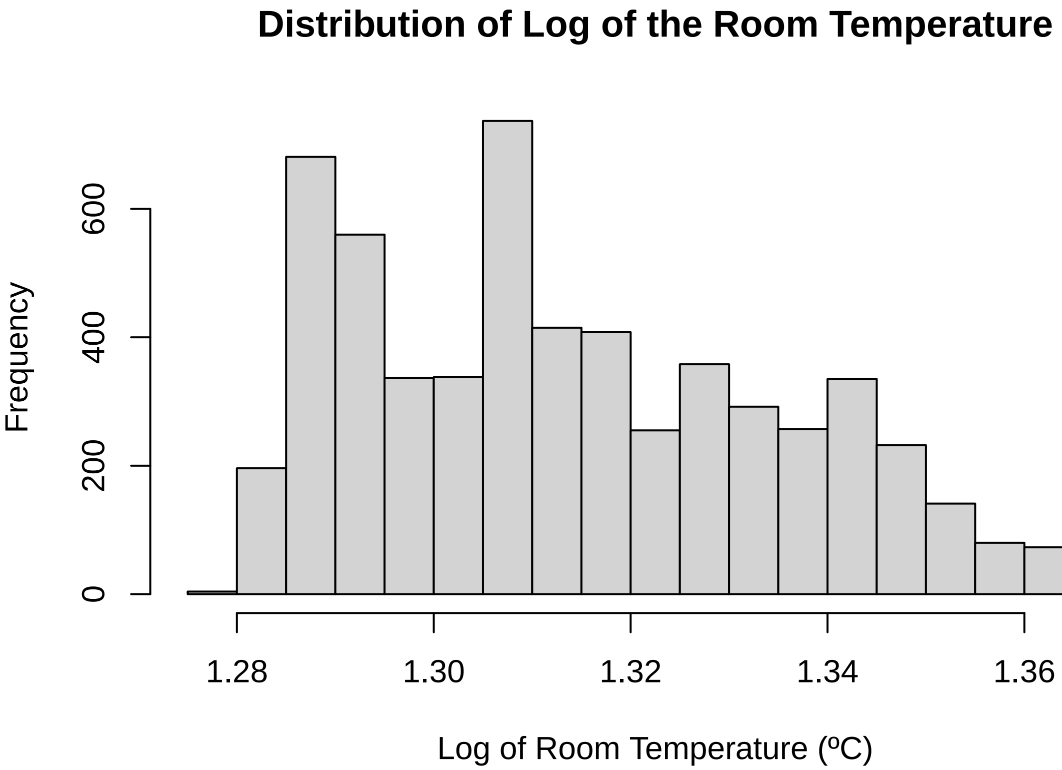
Below are the histograms of each explanatory variable, and the histograms of some of the transformed explanatory variables.

Temperature Transformation

```
hist(occ_tr$Temperature, main = "Distribution of Room Temperature", xlab = "Room Temperature (ºC)")
```



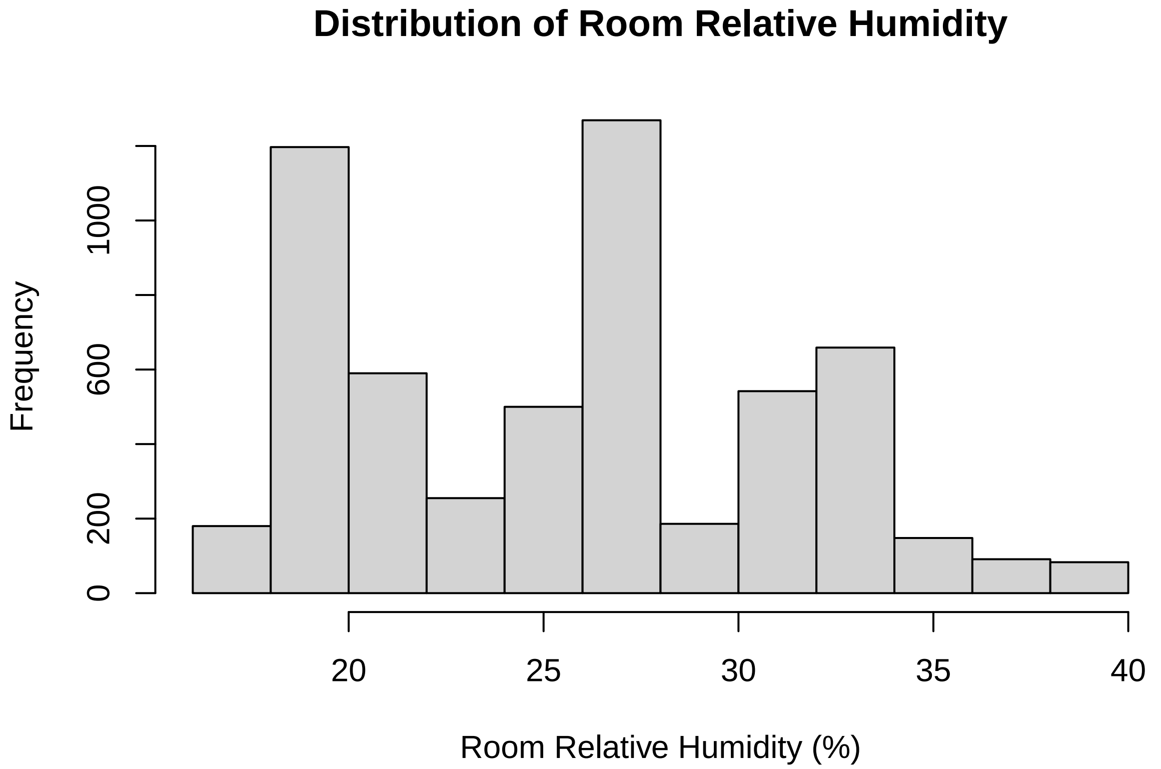
```
occ_tr$log10Temperature <- log10(occ_tr$Temperature)

hist(occ_tr$log10Temperature, main = "Distribution of Log of the Room Temperature", xlab = "Log of Room
```



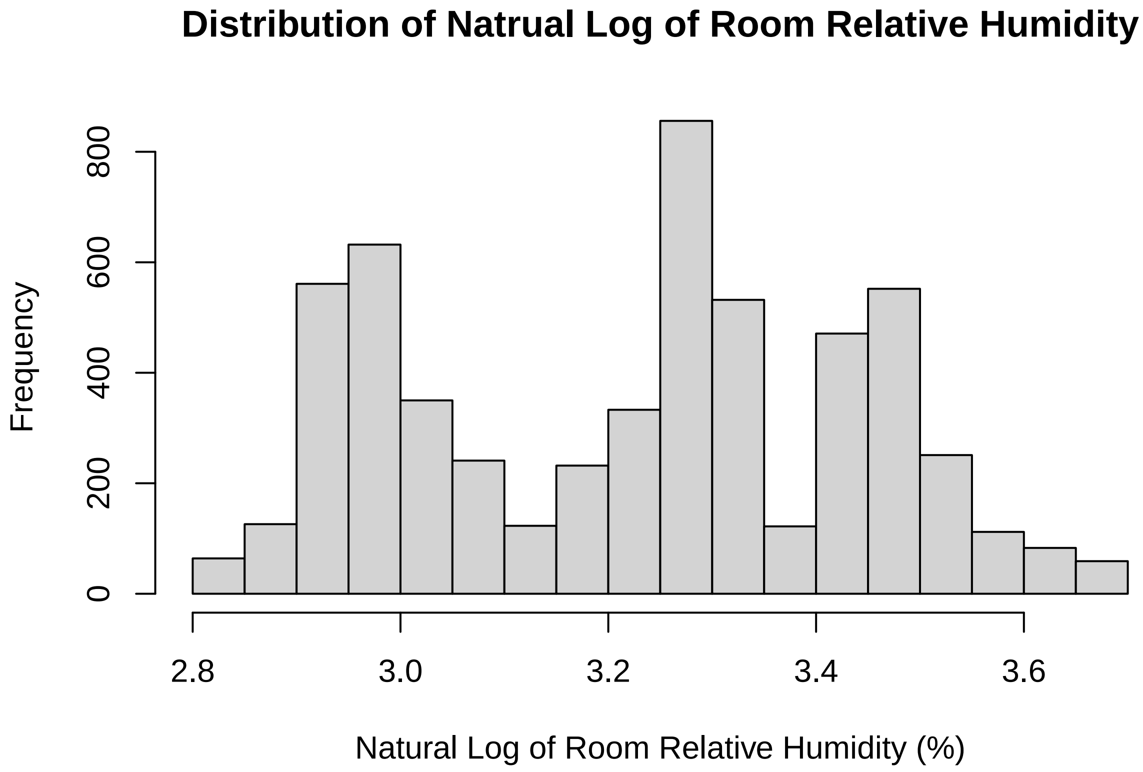
The original distribution for the room temperature appears to be skewed to the right. To modify the variable so that it is normally distributed, we can use logarithmic functions. We use the logarithmic function with a base 10 for the transformation because the distribution seems to be more normal transforming with a natural logarithmic function. We do recognize that the transformed version of the distribution is still skewed to the right, but the distribution of the log of the room temperature with a base 10 appears to be less skewed (or more normally distributed) than the distribution of the original room temperature variable.

Humidity Transformation

```
hist(occ_tr$Humidity, main = "Distribution of Room Relative Humidity", xlab = "Room Relative Humidity (%
```



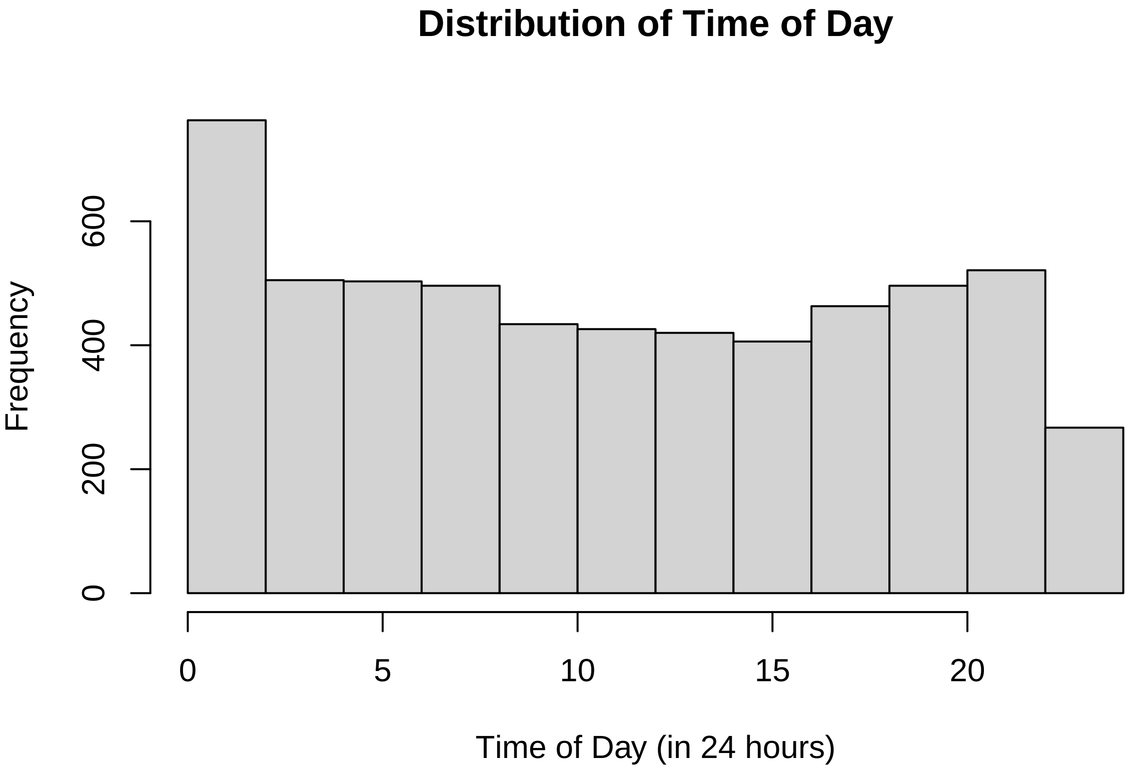
```
occ_tr$logHumidity <- log(occ_tr$Humidity)

hist(occ_tr$logHumidity, main = "Distribution of Natrual Log of Room Relative Humidity", xlab = "Natural
```



The original distribution for the room relative humidity appears to be skewed to the right. To modify the variable so that it is normally distributed, we can use logarithmic functions. We used the natural logarithmic function for the transformation, but the distribution after transforming with a logarithmic function of base 10 is also quite similar to the distribution after transforming with the natural logarithmic function. We do recognize that the transformed version of the distribution is not perfectly normal, but the distribution of the natural log of the room relative humidity appears to be less skewed (or more normally distributed) than the distribution of the original room relative humidity variable.

Hour Transformation

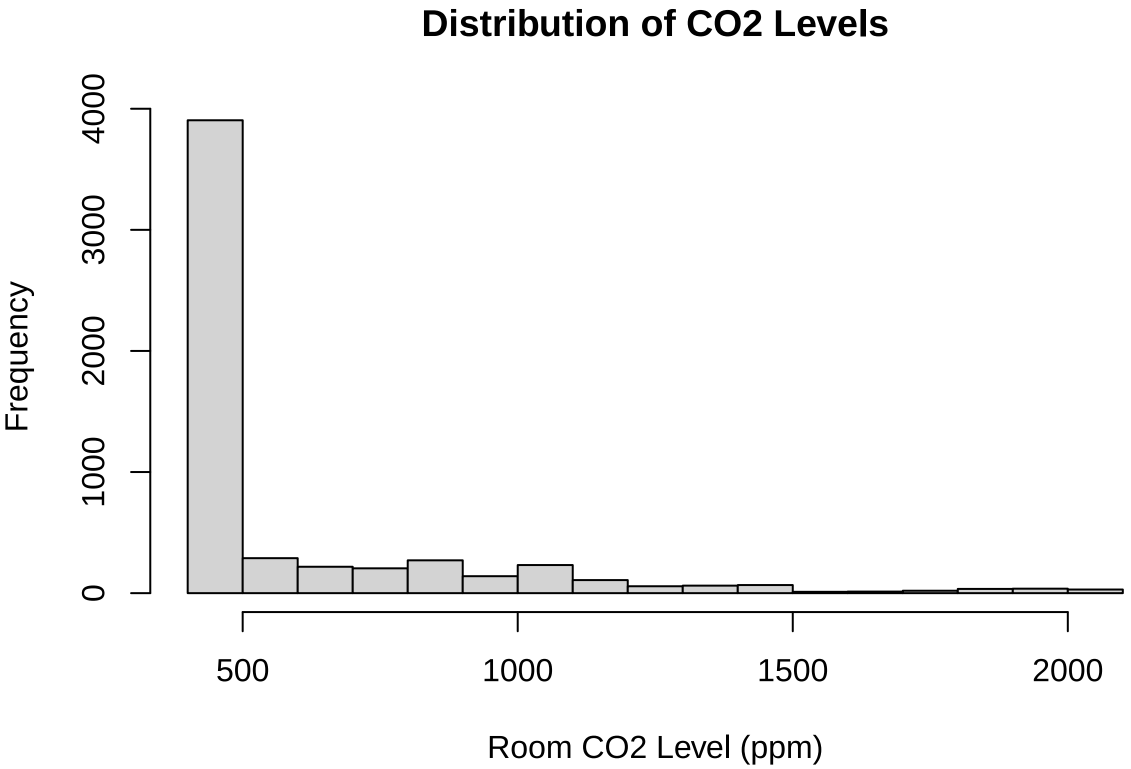
```
hist(occ_tr$Hour, main = "Distribution of Time of Day", xlab = "Time of Day (in 24 hours)")
```



The distribution appears to be slightly skewed to the right and mostly uniform. Transforming the variable `Hour` by using logarithmic functions and root functions would produce distributions that are skewed to the right. So not transforming the variable `Hour` may be the most appropriate decision.

CO2 Transformation

```
hist(occ_tr$CO2, main = "Distribution of CO2 Levels", xlab = "Room CO2 Level (ppm)")
```



The distribution appears to be very skewed to the right. Transforming the variable `CO2` by using logarithmic functions and root functions would produce distributions that are still skewed to the right. So not transforming the variable `CO2` would keep the data simpler.

LDA

The LDA is built on the training data. There are two versions of the LDA because we have to account for the transformed variables due to the Gaussian requirement for discriminant analysis.

```
occ_LDA1 <- lda(factor(Occupancy) ~ log10Temperature + logHumidity + Hour + CO2, data = occ_tr)
```

The LDA model is then compared to the testing data.

```
occ_test$log10Temperature <- log10(occ_test$Temperature)
occ_test$logHumidity <- log(occ_test$Humidity)

occ_LDA1.pre <- predict(occ_LDA1, as.data.frame(occ_test))
```

LDA error table:

```
table(occ_LDA1.pre$class, occ_test$Occupancy)
```

```
##
##         0    1
##    0 1847  123
##    1   70  403
```

This version of the LDA does not have transformed variables.

```
occ_LDA2 <- lda(factor(Occupancy) ~ Temperature + Humidity + Hour + CO2, data = occ_tr)
```

The LDA model is then compared to the testing data.

```
occ_LDA2.pre <- predict(occ_LDA2, as.data.frame(occ_test))
```

LDA error table:

```
table(occ_LDA2.pre$class, occ_test$Occupancy)
```

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

LDA Error Rates

LDA with Transformations:

```
(70+123)/2443 #overall
```

```
## [1] 0.07900123
```

```
70/(1847+70) #for truly not occupied
```

```
## [1] 0.03651539
```

```
123/(403+123) #for truly occupied
```

```
## [1] 0.2338403
```

LDA without Transformations:

```
(73+111)/2443 #overall
```

```
## [1] 0.07531723
```

```
73/(1844+73) #for truly not occupied
```

```
## [1] 0.03808033
```

```
111/(415+111) #for truly occupied
```

```
## [1] 0.2110266
```

Overall: The LDA without transformations has a lower overall error rate at 0.075 than that of the LDA with transformations at 0.079.

For truly not occupied: The LDA without transformations has a higher error rate for rooms that are truly not occupied at 0.038 than that of the LDA with transformations at 0.037.

For truly occupied: The LDA without transformations has a lower error rate for the rooms that are truly occupied at 0.021 than that of the LDA with transformations at 0.023.

Although it does appear that LDA with no transformation produces better results based on its lower error rates, we must use the LDA with the transformations because of the requirement that variables in the LDA should be Gaussian/normally distributed. The error rates for the rooms that are truly occupied for both LDA models are also quite high; the error rates are greater than 20%, meaning the LDA model will classify a occupied room incorrectly once every 5 rooms, on average.

QDA

The QDA is built on the training data. There are two versions of the QDA because we have to account for the transformed variables due to the Gaussian requirement for discriminant analysis.

```
occ_QDA1 <- qda(factor(Occupancy) ~ log10Temperature + logHumidity + Hour + CO2, data = occ_tr)
```

The QDA model is then compared to the testing data.

```
occ_test$log10Temperature <- log10(occ_test$Temperature)
occ_test$logHumidity <- log(occ_test$Humidity)

occ_QDA1.pre <- predict(occ_QDA1, as.data.frame(occ_test))
```

QDA error table:

```
table(occ_QDA1.pre$class, occ_test$Occupancy)
```

```
##
##        0    1
##   0 1821   83
##   1   96  443
```

This version of the QDA does not have transformed variables.

```
occ_QDA2 <- qda(factor(Occupancy) ~ Temperature + Humidity + Hour + CO2, data = occ_tr)
```

The QDA model is then compared to the testing data.

```
occ_QDA2.pre <- predict(occ_QDA2, as.data.frame(occ_test))
```

QDA error table:

```
table(occ_QDA2.pre$class, occ_test$Occupancy)
```

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

QDA Error Rates

QDA with Transformations:

```r
(96+83)/2443 #overall
```

```
## [1] 0.07327057
```

```r
96/(1821+96) #for truly not occupied
```

```
## [1] 0.05007825
```

```r
83/(443+83) #for truly occupied
```

```
## [1] 0.1577947
```

QDA without Transformations:

```r
(85+81)/2443 #overall
```

```
## [1] 0.06794924
```

```r
85/(1832+85) #for truly not occupied
```

```
## [1] 0.04434011
```

```r
81/(445+81) #for truly occupied
```

```
## [1] 0.1539924
```

Overall: The QDA without transformations has a lower overall error rate at 0.068 than that of the QDA with transformations at 0.073.

For truly not occupied: The QDA without transformations has a lower error rate for rooms that are truly not occupied at 0.044 than that of the QDA with transformations at 0.050.

For truly occupied: The QDA without transformations has a lower error rate for the rooms that are truly occupied at 0.0153 than that of the QDA with transformations at 0.0158.

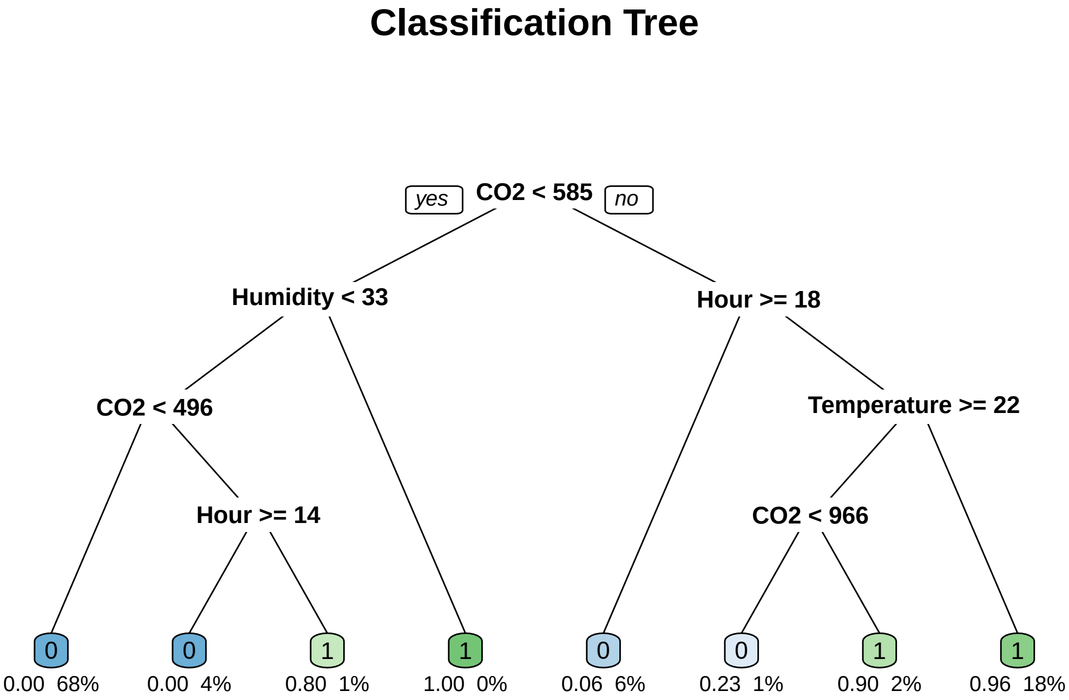
Although it does appear that the QDA with no transformation produces better results based on its lower error rates, we must use the QDA with the transformations because of the requirement that variables in the QDA should be Gaussian/normally distributed. Additionally, the error rates for the rooms that are truly occupied for both QDA models are also quite high; the error rates are greater than 15%, meaning the QDA model will classify a occupied room incorrectly once every 6.67 rooms, on average.

Classification Tree

Classification Tree models can use both quantitative and categorical variables. Below is the classification tree model:

```
occ_tree <- rpart(Occupancy ~ Temperature + Humidity + Hour + CO2, data = occ_tr, method = "class")
```

```
rpart.plot(occ_tree, type = 0, clip.right.labs = FALSE, branch = 0.1, under = TRUE, main = "Classificat
```



```
occ_tree.pre <- predict(occ_tree, as.data.frame(occ_test), type = "class")
```

```
table(occ_tree.pre, occ_test$Occupancy) #normal tree
```

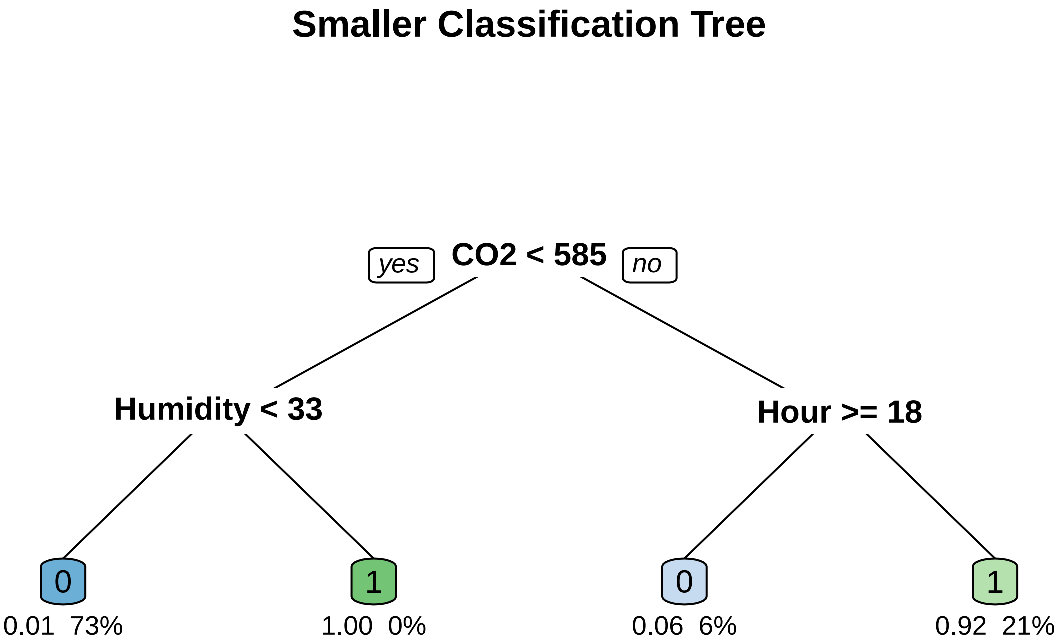
```
##
## occ_tree.pre    0    1
##            0 1883   15
##            1   34  511
```

This classification tree creates a very specific model because the variable `CO2` is repeated 3 times and the variable `Hour` is repeated twice. The leaf node of the classification tree has "0"s and "1"s labeled; if the probability of the room being occupied is greater than 0.5, then the leaf will be 1; if the probability of the room being occupied is less than 0.5, then the leaf will indicate 0. The percentages at the bottom next to the probability number is the percent of data that is in the training data set. As an example, the right-most leaf indicates 0.96 and 18%, meaning that the observations that goes down that pathway has a probability 0.96 of being occupied, and 18% of all the observations in the training data goes down that route towards that leaf.

Pruning (Smaller Classification Tree)

```
occ_tree.small <- rpart(Occupancy ~ Temperature + Humidity + Hour + CO2, data = occ_tr, method = "class
```

```
rpart.plot(occ_tree.small, type = 0, clip.right.labs = FALSE, branch = 0.1, under = TRUE, main = "Small
```



```
occ_tree.small.pre <- predict(occ_tree.small, as.data.frame(occ_test), type = "class")
```

```
table(occ_tree.small.pre, occ_test$Occupancy) #small tree
```

```
##
## occ_tree.small.pre    0    1
##                  0 1869   30
##                  1   48  496
```

This classification tree creates a less specific model because the variable `Temperature` is not used. The leaf node of the classification tree has "0"s and "1"s labeled; if the probability of the room being occupied is greater than 0.5, then the leaf will be 1; if the probability of the room being occupied is less than 0.5, then the leaf will indicate 0. The percentages at the bottom next to the probability number is the percent of data that is in the training data set. As an example, the right-most leaf indicates 0.92 and 21%, meaning that the observations that goes down that pathway has a probability 0.92 of being occupied, and 21% of all the observations in the training data goes down that route towards that leaf.

Classification Trees Error Rates Comparison

Normal Tree Error Rates:

```
(34+15)/2443 #overall
```

```
## [1] 0.02005731
```

```
34/(1883+34) #for truly not occupied
```

```
## [1] 0.01773605
```

```
15/(511+15) #for truly occupied
```

```
## [1] 0.02851711
```

Smaller Tree Error Rates:

```
(48+30)/2443 #overall
```

```
## [1] 0.03192796
```

```
48/(1869+48) #for truly not occupied
```

```
## [1] 0.02503912
```

```
30/(496+30) #for truly occupied
```

```
## [1] 0.05703422
```

Overall: The normal classification tree has a lower overall error rate at 0.02 than that of the smaller tree at 0.032.

For truly not occupied: The normal classification tree has a lower error rate for the rooms that are truly not occupied at 0.018 than that of the smaller tree at 0.025.

For truly occupied: The normal classification tree has a lower error rate for the rooms that are truly occupied at 0.029 than that of the smaller tree at 0.057.

Binary Logistic Regression

```
occ_log <- glm(factor(Occupancy) ~ Temperature + Humidity + Hour + CO2, data = occ_tr, family = binomial
```

```
occ_log.prob <- predict(occ_log, as.data.frame(occ_test), type = "response")
```

```
occ_log.pre <- (occ_log.prob > 0.5)
```

```
table(occ_log.pre, occ_test$Occupancy)
```

```
##
## occ_log.pre    0    1
##       FALSE 1849   96
##       TRUE    68  430
```

Binary Logistic Regression Error Rates

```
(68+96)/2443 #overall
```

```
## [1] 0.06713058
```

```
68/(1849+68) #for truly not occupied
```

```
## [1] 0.03547209
```

```
96/(430+96) #for truly occupied
```

```
## [1] 0.1825095
```

Although the overall error rate for the binary logistic regression model is quite small at 6.71%, the error rate for the room to be truly occupied is quite high at 18.25%. This means that the model will classify incorrectly for a occupied room every 5.5 rooms, on average.

Recommended Model

```
occ_tree.pre2 <- predict(occ_tree, as.data.frame(occ_tr), type = "class")
```

```
table(occ_tree.pre2, occ_tr$Occupancy) #normal tree
```

```
##
## occ_tree.pre2    0    1
##             0 4430   45
##             1   67 1158
```

```
(67+45)/5700 #overall
```

```
## [1] 0.01964912
```

```
67/(4430+67) #for truly not occupied
```

```
## [1] 0.01489882
```

```
45/(1158+45) #for truly occupied
```

```
## [1] 0.03740648
```

```
(34+15)/2443 #overall
```

```
## [1] 0.02005731
```

```
34/(1883+34) #for truly not occupied
```

```
## [1] 0.01773605
```

```
15/(511+15) #for truly occupied
```

```
## [1] 0.02851711
```

To classify whether a room is occupied or not, the normal (un-pruned) classification tree is the most appropriate model to predict the occupancy of a room. This is because it has the lowest error rates. The overall error rate is 2%, which is lower than all the other overall error rates in the other models; the error rate for rooms that are truly not occupied is 1.78%, which is lower than all the other error rates for rooms that are truly not occupied in the other models; the error rate for rooms that are truly occupied is 2.85%, which is lower than all the other error rates for rooms that are truly occupied in the other models.

Furthermore, the pruned version of the classification tree did not produce lower error rates. Pruned classification trees are supposed to reduce a over-fitting problem, and over-fitted models usually gives high error rates. But since the normal classification tree for this data wasn't over-fitted, the smaller classification tree didn't help. The smaller classification tree actually produced higher error rates because it did not account for the `Temperature` explanatory variable, so the chance of the smaller classification tree making an error is higher.

Discussion & Conclusion

The normal classification tree model is the most appropriate model for this data. Despite the explanatory variables not being normal distributed, and despite the pairs plot not having a clear separation between occupied rooms and unoccupied rooms, the error rates for all of the models that were explored were quite low. All of the models produced error rates that were less than 10% except for the error rates of the rooms that are truly occupied. So this means that most of these models will at least have some success in classifying the occupancy of the room.

Furthermore, we have tested all the models with out-of-sample data, meaning the error rates from the model wasn't derived from the data that was used to develop the model. And since all the error rates, derived from the testing data, seem to be relatively low, it is very likely that a overfitting problem does not exist for our models.

It is also important to note that because variables `Hour` and `CO2` are not Gaussian, the LDA and QDA models' assumptions are not met. And it is also important to note that even though classification trees and binary logarithmic regression models has an advantage over discriminant analysis models because discriminant analysis models cannot use categorical variables, the advantage does not exist for our data because there are no categorical variables in our dataset.

However, assuming our classification tree has a overfitting problem, to further improve our classifier model, we may use random forest technique to get the majority result from multiple classification trees based on multiple samples (if available). We would need to collect data from a lot of rooms to create multiple classification trees that determine the `Occupancy`. It may also seem like tracking data on more variables can decrease the error rates because we can predict and classify `Occupancy` more accurately. But this is only true on the training data. This means that if the model that includes the new variables is tested on the original set of data, the error rates would be very low, but if the model that includes the new variable is tested on a new set of data, the error rates would be higher because the model cannot generalize on other data. But assuming our current classification tree doesn't have a overfitting problem, we can possibly add more variables like oxygen level, electricity usage, or whether there is a radio frequency from mobile phones. So the best classifier model to accurately predict the `Occupancy` is to find a balance between not overfitting and not simplify the model too much.