Final Project EDA

Andrea Cui

26 July, 2023

```
firedata <- read.csv("forestfires.csv")
firedata <- as.data.frame(firedata)

# Load the necessary packages
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)

# Load the dataset
data <- read.csv("forestfires.csv")

# View the first few rows of the dataset
head(data)
```

```
##   X Y month day FFMC  DMC    DC  ISI temp RH wind rain area
## 1 7 5   mar fri 86.2 26.2  94.3  5.1  8.2 51  6.7  0.0    0
## 2 7 4   oct tue 90.6 35.4 669.1  6.7 18.0 33  0.9  0.0    0
## 3 7 4   oct sat 90.6 43.7 686.9  6.7 14.6 33  1.3  0.0    0
## 4 8 6   mar fri 91.7 33.3  77.5  9.0  8.3 97  4.0  0.2    0
## 5 8 6   mar sun 89.3 51.3 102.2  9.6 11.4 99  1.8  0.0    0
## 6 8 6   aug sun 92.3 85.3 488.0 14.7 22.2 29  5.4  0.0    0
```

```
# Get a summary of the dataset
summary(data)
```

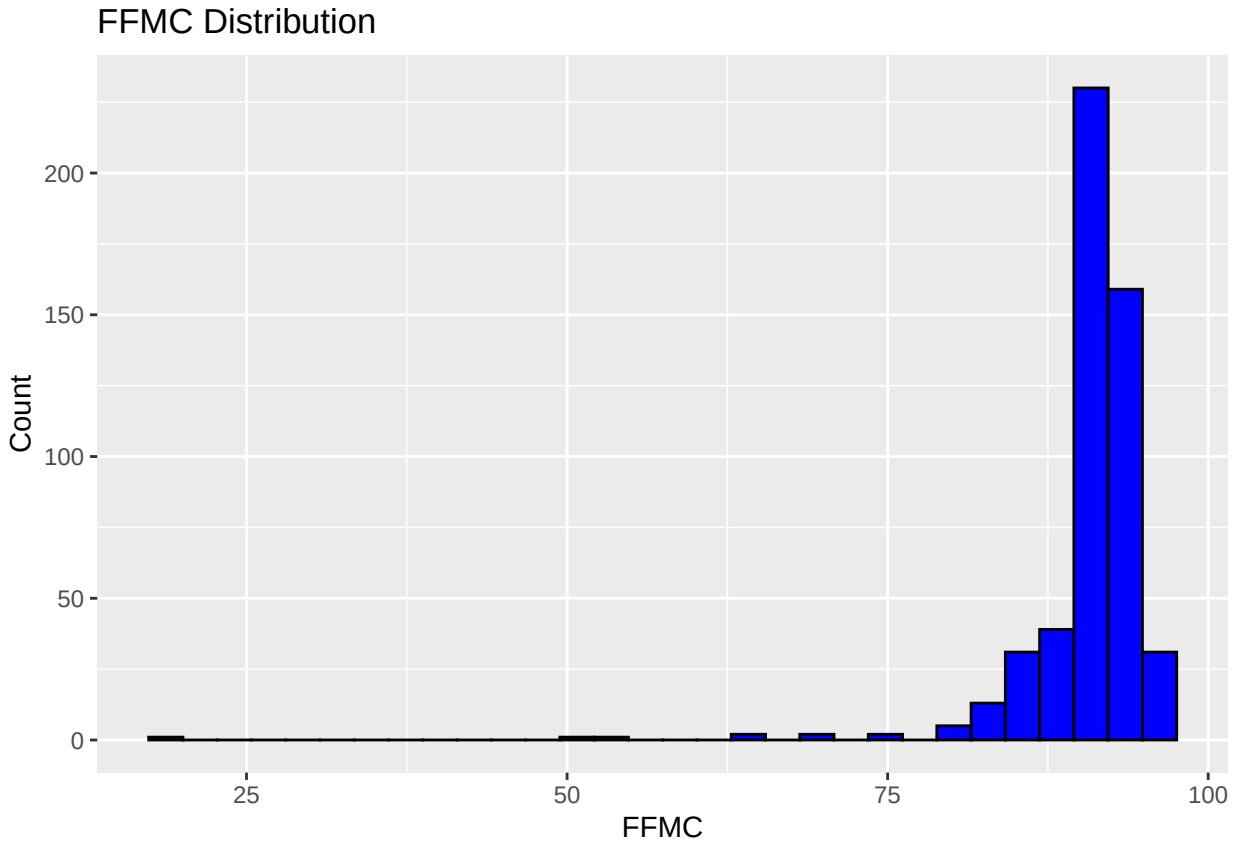
```
##        X               Y          month               day
##  Min.   :1.000   Min.   :2.0   Length:517         Length:517
##  1st Qu.:3.000   1st Qu.:4.0   Class :character   Class :character
```

```
##  Median :4.000   Median :4.0   Mode  :character   Mode  :character
##  Mean   :4.669   Mean   :4.3
##  3rd Qu.:7.000   3rd Qu.:5.0
##  Max.   :9.000   Max.   :9.0
##       FFMC            DMC              DC             ISI
##  Min.   :18.70   Min.   :  1.1   Min.   :  7.9   Min.   : 0.000
##  1st Qu.:90.20   1st Qu.: 68.6   1st Qu.:437.7   1st Qu.: 6.500
##  Median :91.60   Median :108.3   Median :664.2   Median : 8.400
##  Mean   :90.64   Mean   :110.9   Mean   :547.9   Mean   : 9.022
##  3rd Qu.:92.90   3rd Qu.:142.4   3rd Qu.:713.9   3rd Qu.:10.800
##  Max.   :96.20   Max.   :291.3   Max.   :860.6   Max.   :56.100
##       temp             RH              wind            rain
##  Min.   : 2.20   Min.   : 15.00   Min.   :0.400   Min.   :0.00000
##  1st Qu.:15.50   1st Qu.: 33.00   1st Qu.:2.700   1st Qu.:0.00000
##  Median :19.30   Median : 42.00   Median :4.000   Median :0.00000
##  Mean   :18.89   Mean   : 44.29   Mean   :4.018   Mean   :0.02166
##  3rd Qu.:22.80   3rd Qu.: 53.00   3rd Qu.:4.900   3rd Qu.:0.00000
##  Max.   :33.30   Max.   :100.00   Max.   :9.400   Max.   :6.40000
##       area
##  Min.   :   0.00
##  1st Qu.:   0.00
##  Median :   0.52
##  Mean   :  12.85
##  3rd Qu.:   6.57
##  Max.   :1090.84
```

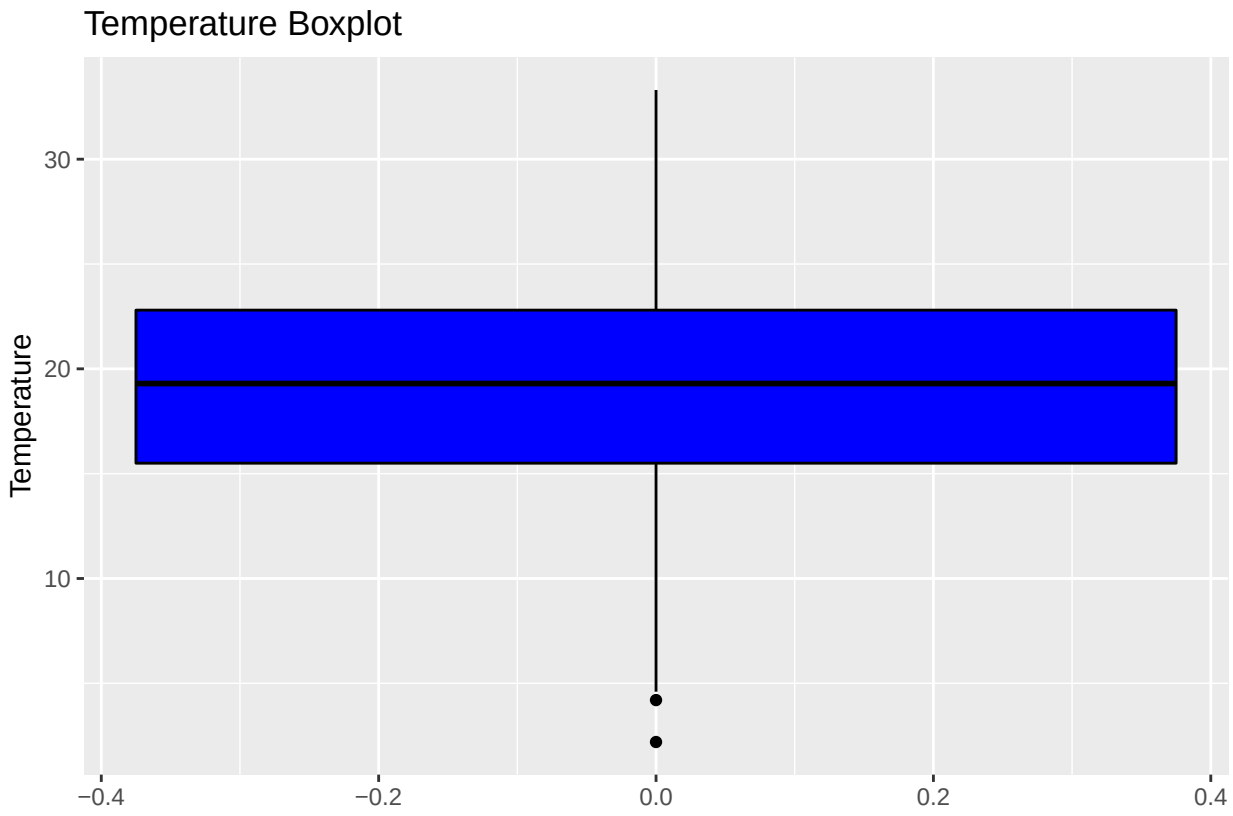
Data Visualizations:

```
# Histogram for the FFMC variable
ggplot(data, aes(x=FFMC)) +
  geom_histogram(fill='blue', color='black') +
  labs(title="FFMC Distribution", x="FFMC", y="Count")
```

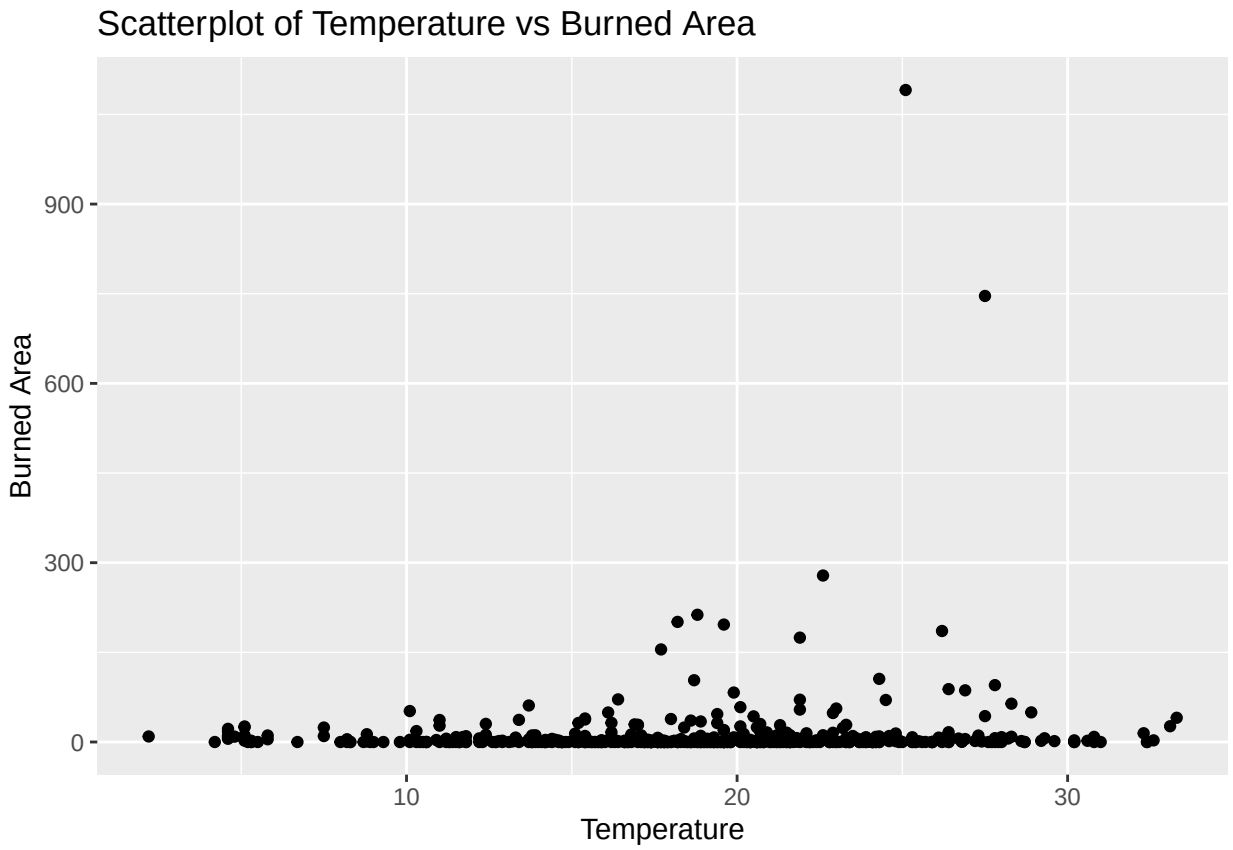
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Boxplot for temperature
ggplot(data, aes(y=temp)) +
  geom_boxplot(fill='blue', color='black') +
  labs(title="Temperature Boxplot", x="", y="Temperature")
```

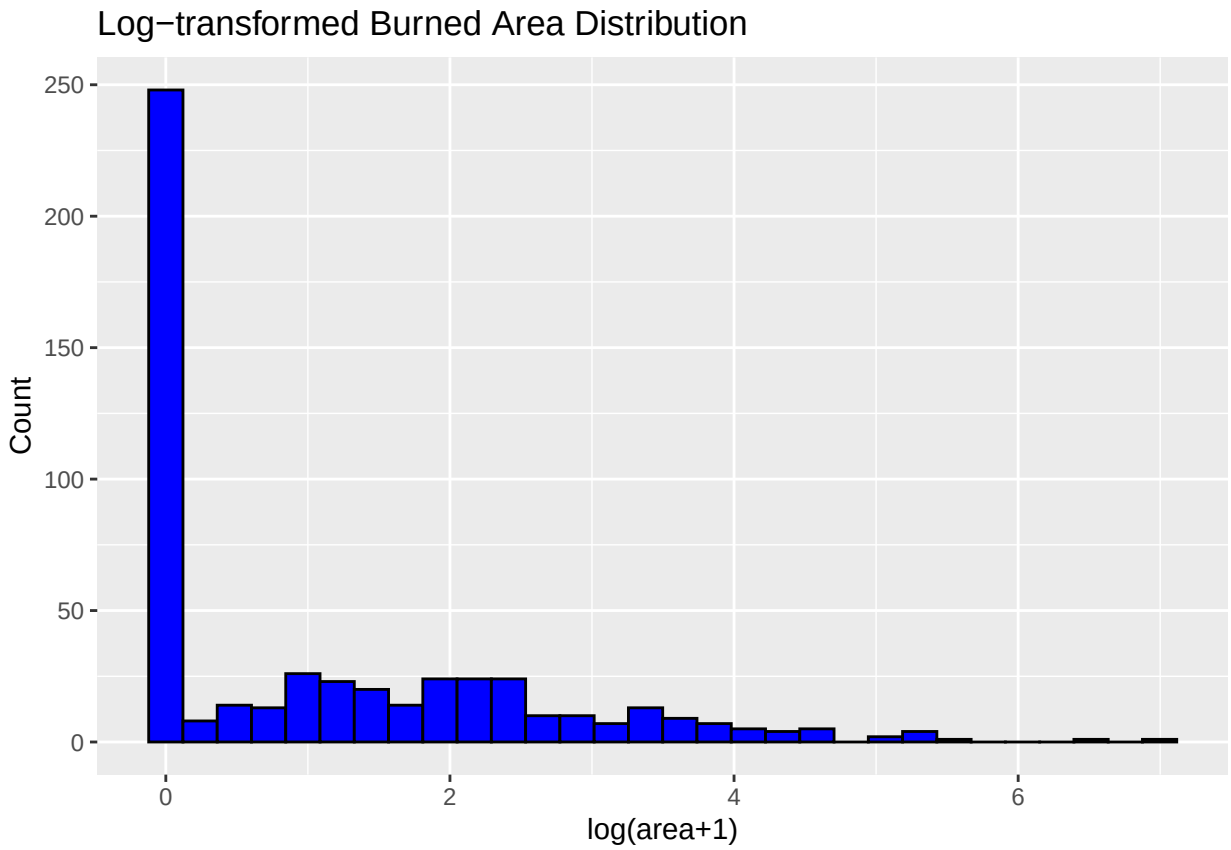



```
ggplot(data, aes(x=temp, y=area)) +
  geom_point() +
  labs(title="Scatterplot of Temperature vs Burned Area", x="Temperature", y="Burned Area")
```




```
# Histogram for the log-transformed area
ggplot(data, aes(x=log(area + 1))) +
  geom_histogram(fill='blue', color='black') +
  labs(title="Log-transformed Burned Area Distribution", x="log(area+1)", y="Count")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



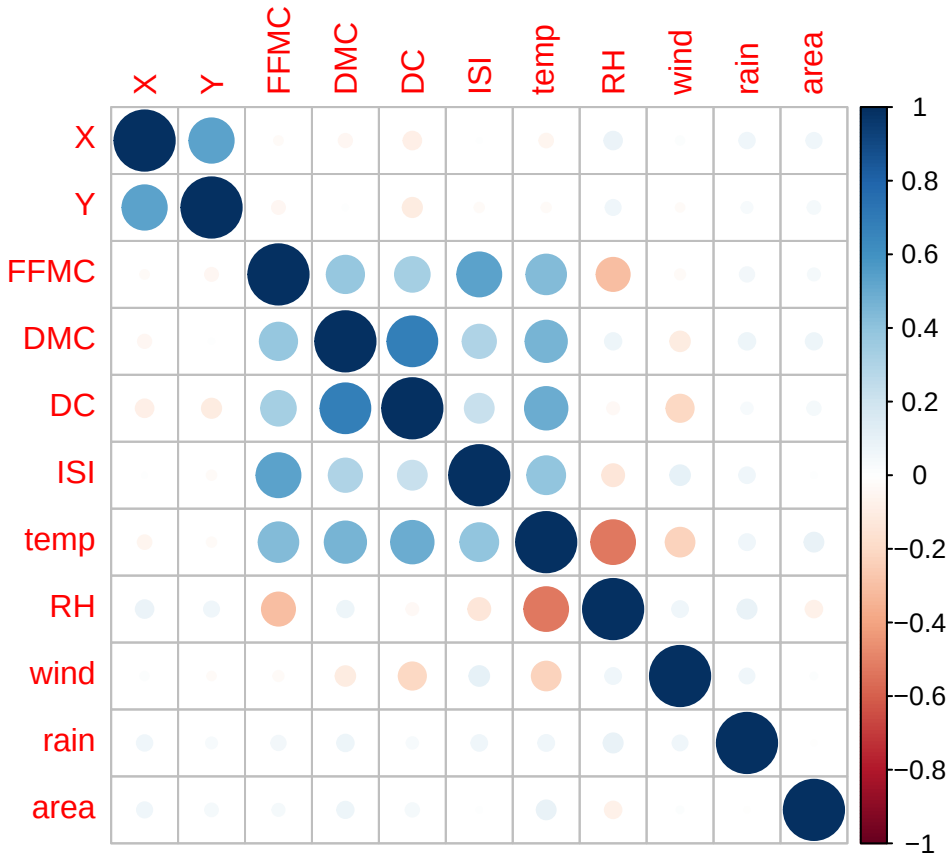

```
# In the code above, I added 1 to the area before taking the logarithm to avoid
# undefined values since log(0) is not defined.
```

```
# Load required package
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
# Compute the correlation matrix
correlationMatrix <- cor(data[,sapply(data, is.numeric)])

# Generate the correlation plot
corrplot(correlationMatrix, method = "circle")
```

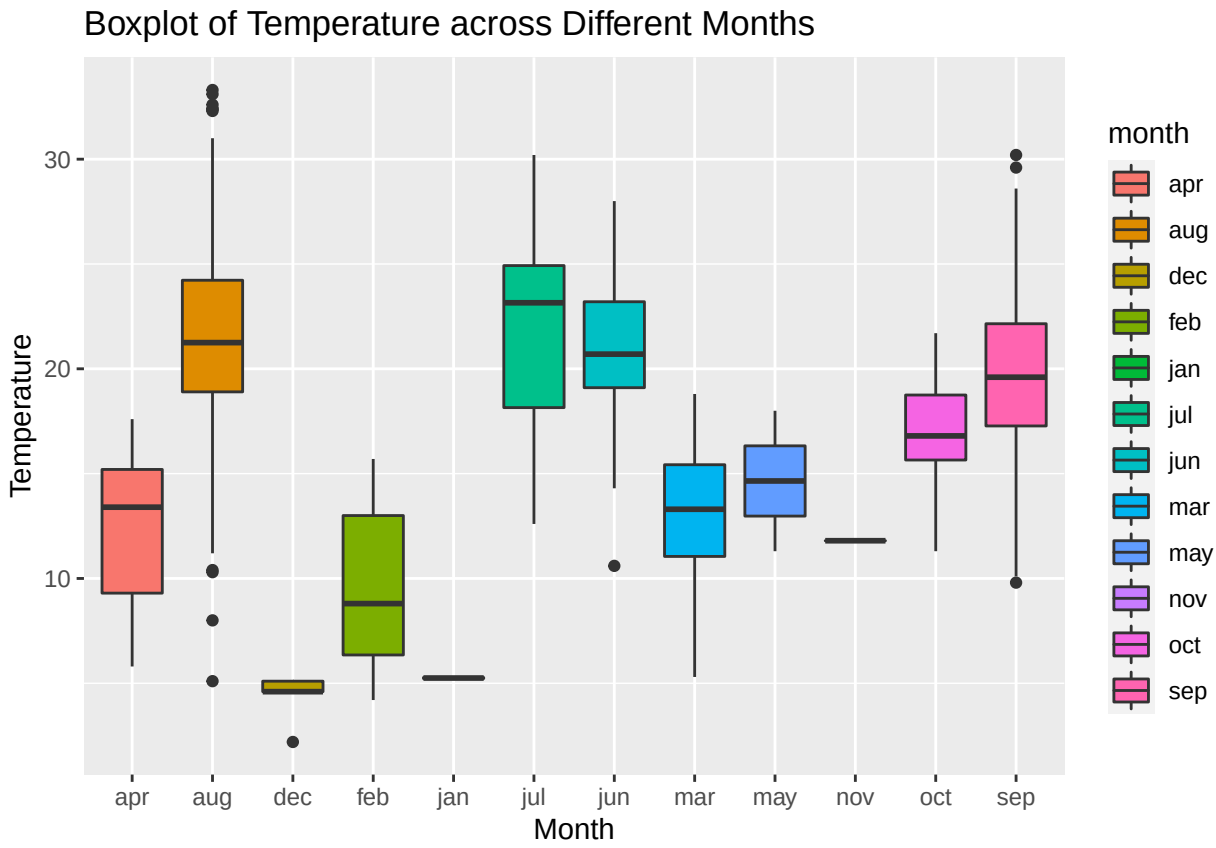



```
# Check for missing values
sapply(data, function(x) sum(is.na(x)))
```

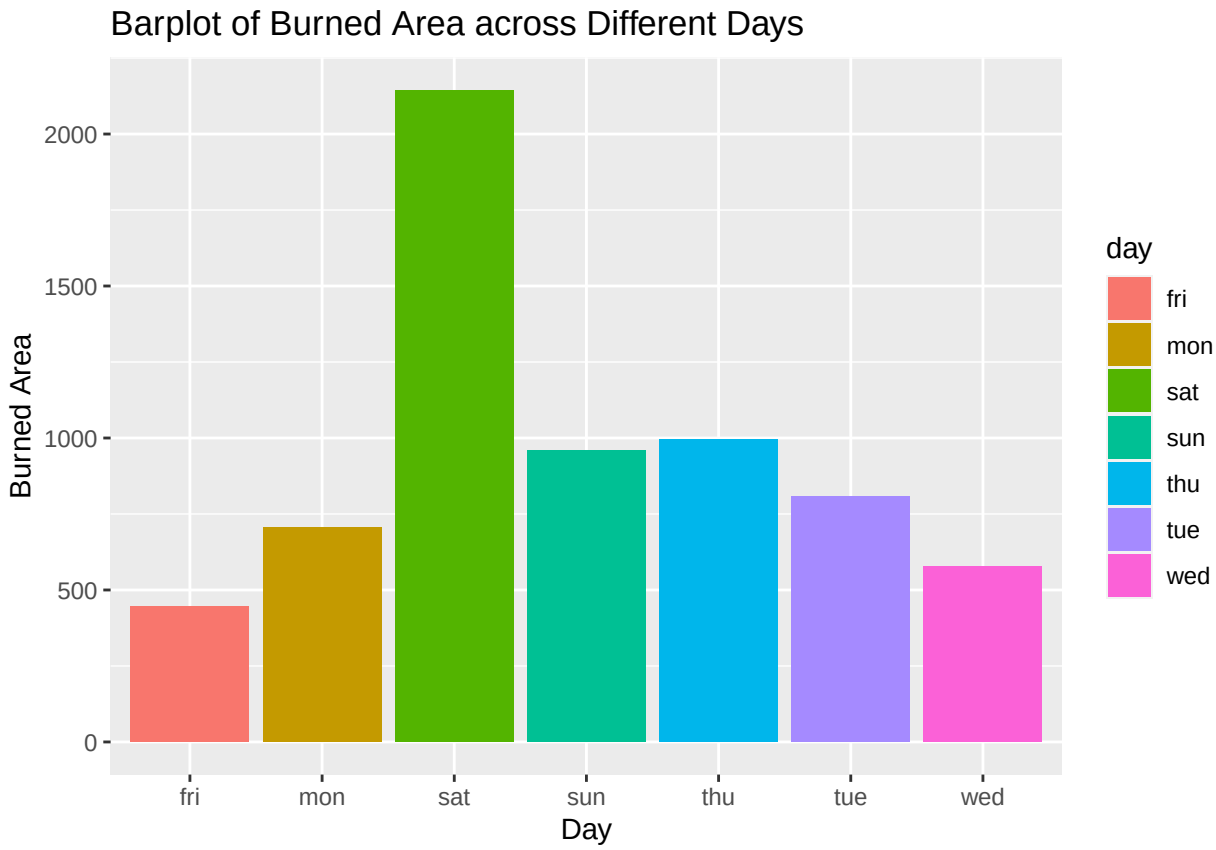
```
##     X     Y month   day  FFMC   DMC    DC   ISI  temp    RH  wind  rain  area
##     0     0     0     0     0     0     0     0     0     0     0     0     0
```

Comparisons Between Variables:

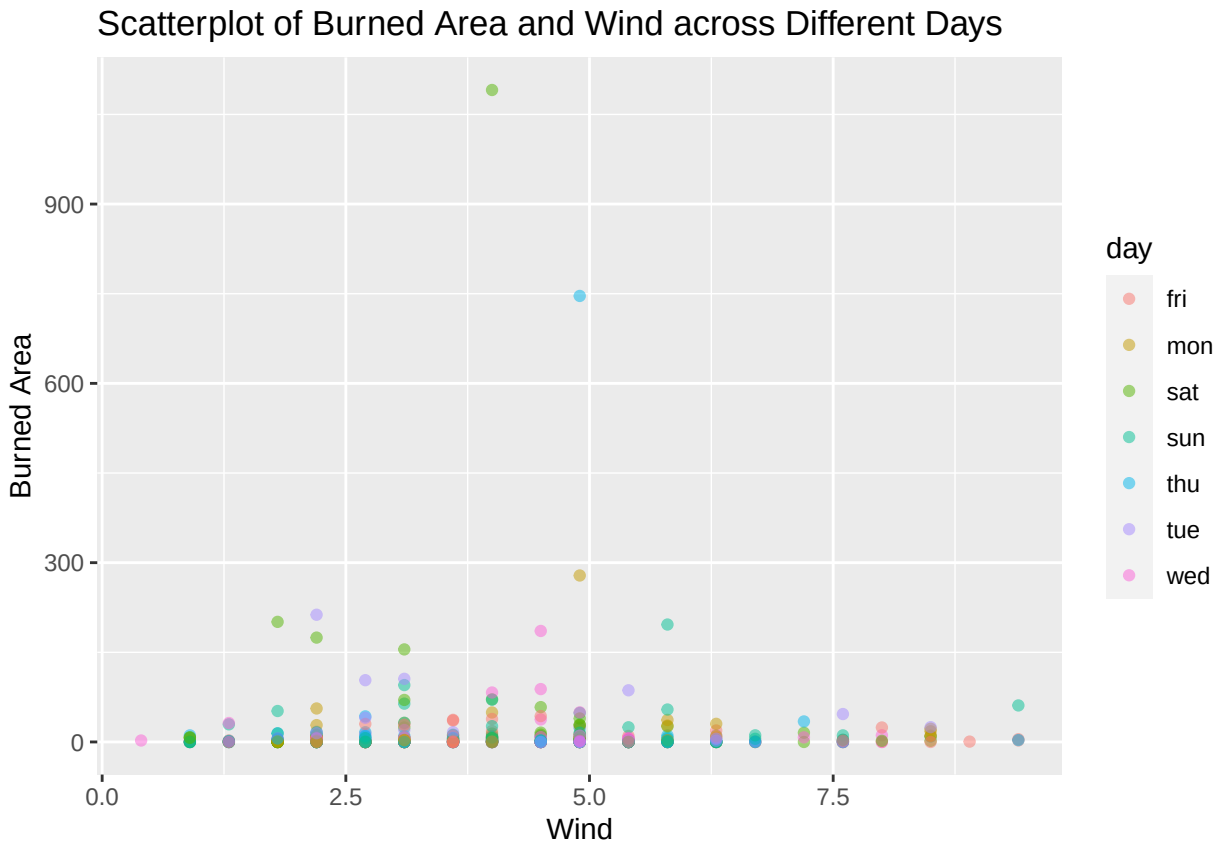
```
# Boxplot of Temperature across Different Months
ggplot(data, aes(x=month, y=temp, fill=month)) +
  geom_boxplot() +
  labs(title="Boxplot of Temperature across Different Months", x="Month", y="Temperature")
```




```r
# Barplot of Burned Area across Different Days
ggplot(data, aes(x=day, y=area, fill=day)) +
  geom_bar(stat="identity") +
  labs(title="Barplot of Burned Area across Different Days", x="Day", y="Burned Area")
```




```
# Scatterplot of Burned Area and Wind with respect to different days of the week
ggplot(data, aes(x=wind, y=area, color=day)) +
  geom_point(alpha=0.5) +
  labs(title="Scatterplot of Burned Area and Wind across Different Days", x="Wind", y="Burned Area")
```



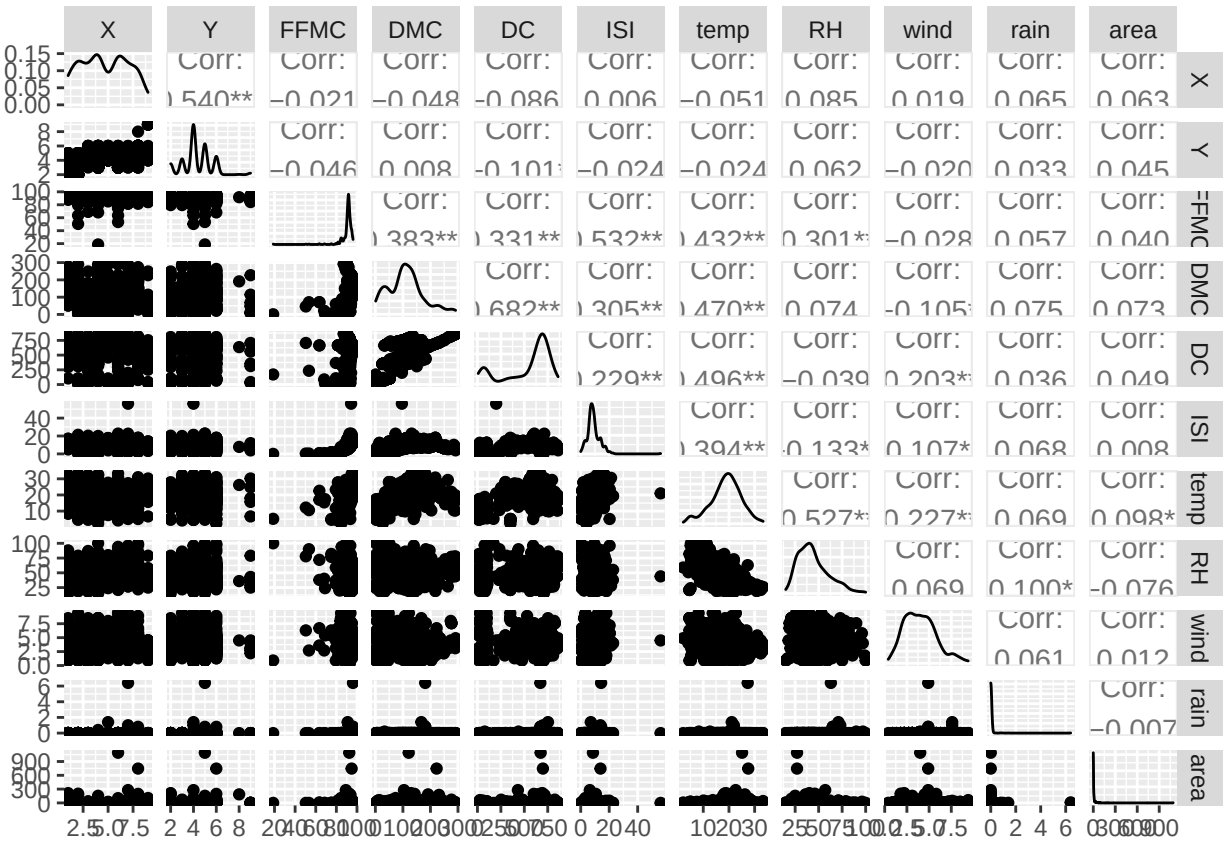
pair plots:

```
# Load required package
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

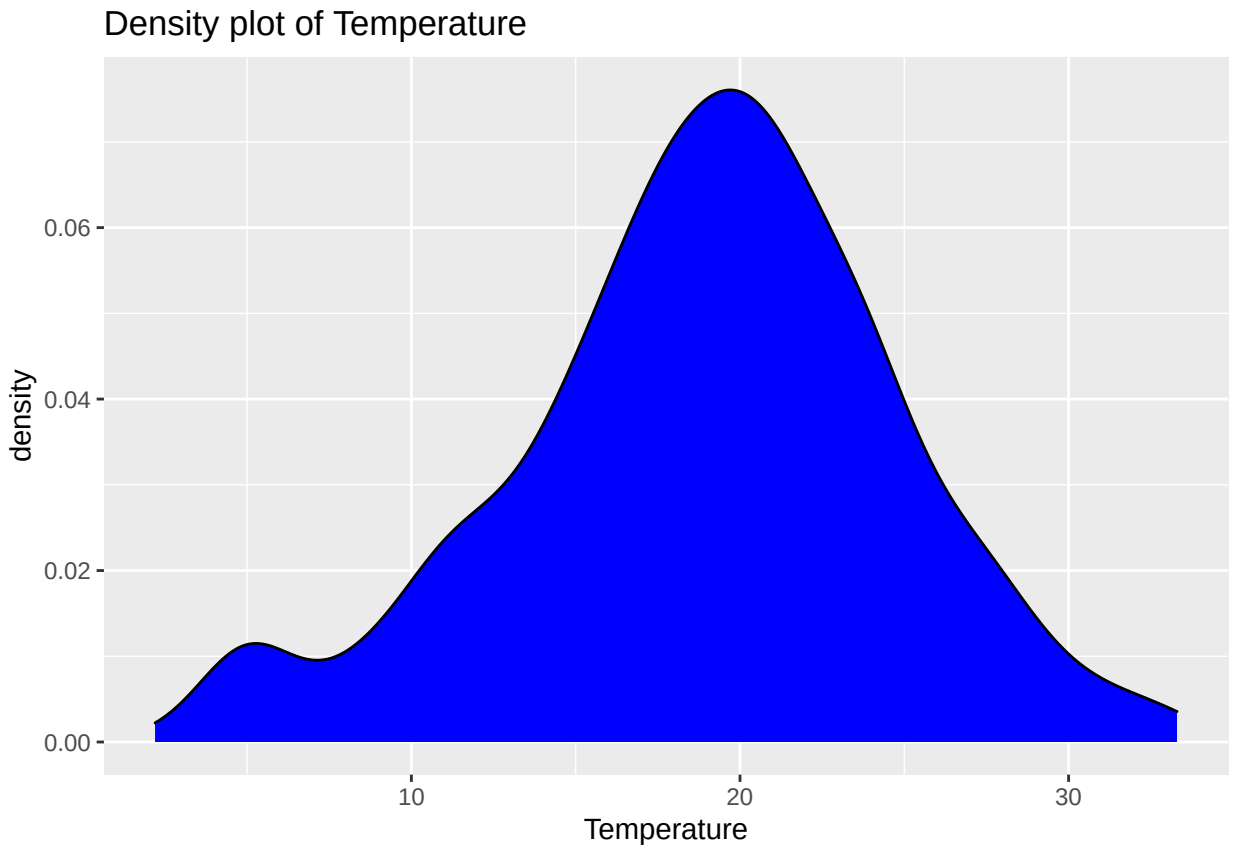
```
# Select numerical variables to avoid clutter
data_num <- data[, sapply(data, is.numeric)]

# Generate pairs plot
ggpairs(data_num)
```

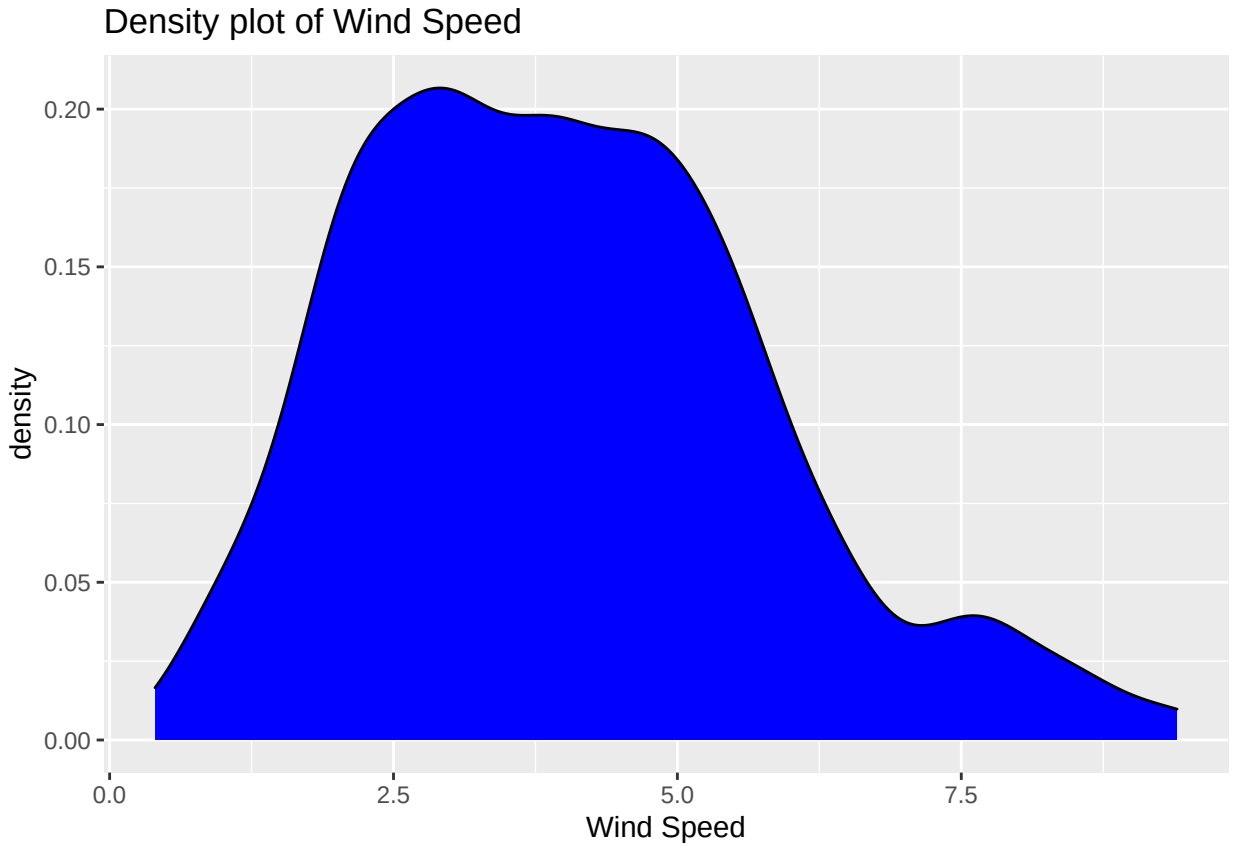


Density Plots:

```
# Density plot of Temperature
ggplot(data, aes(x=temp)) +
  geom_density(fill='blue') +
  labs(title="Density plot of Temperature", x="Temperature")
```

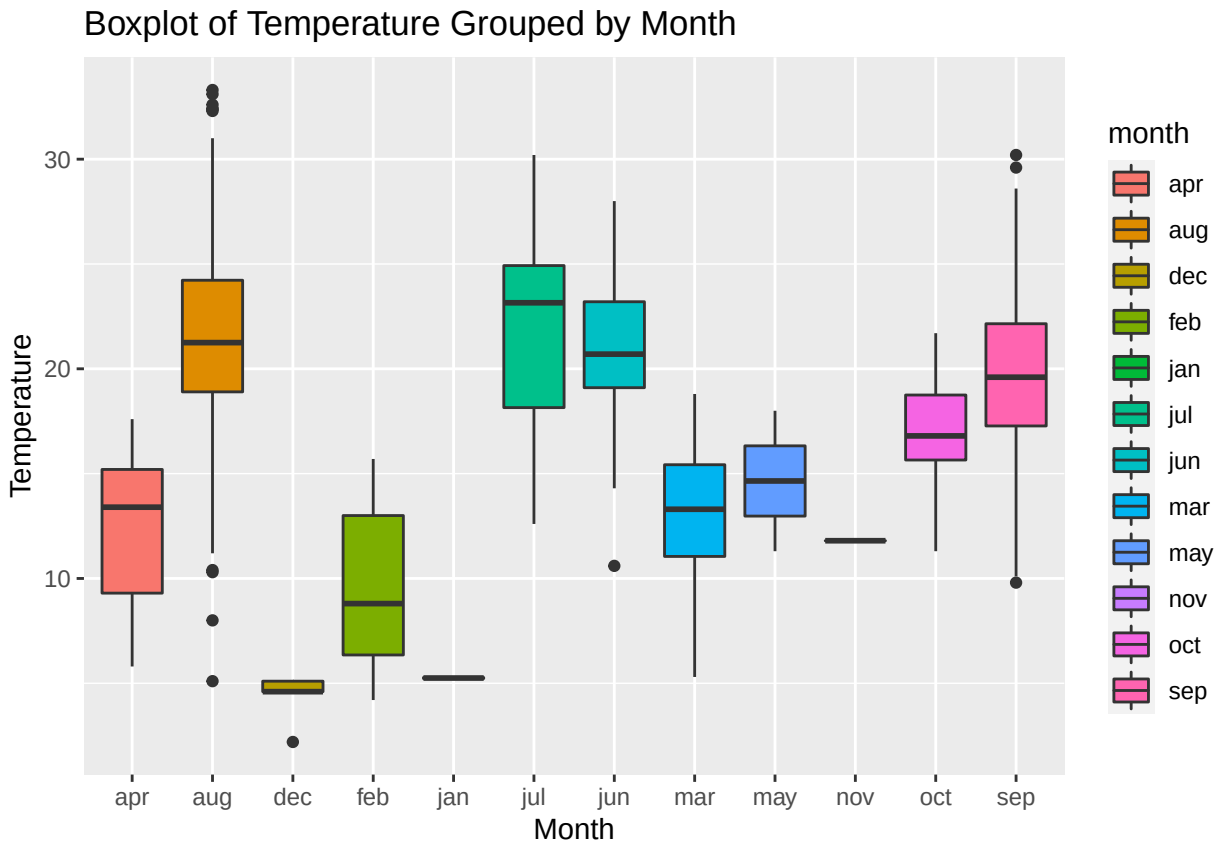



```
# Density plot of Wind Speed
ggplot(data, aes(x=wind)) +
  geom_density(fill='blue') +
  labs(title="Density plot of Wind Speed", x="Wind Speed")
```

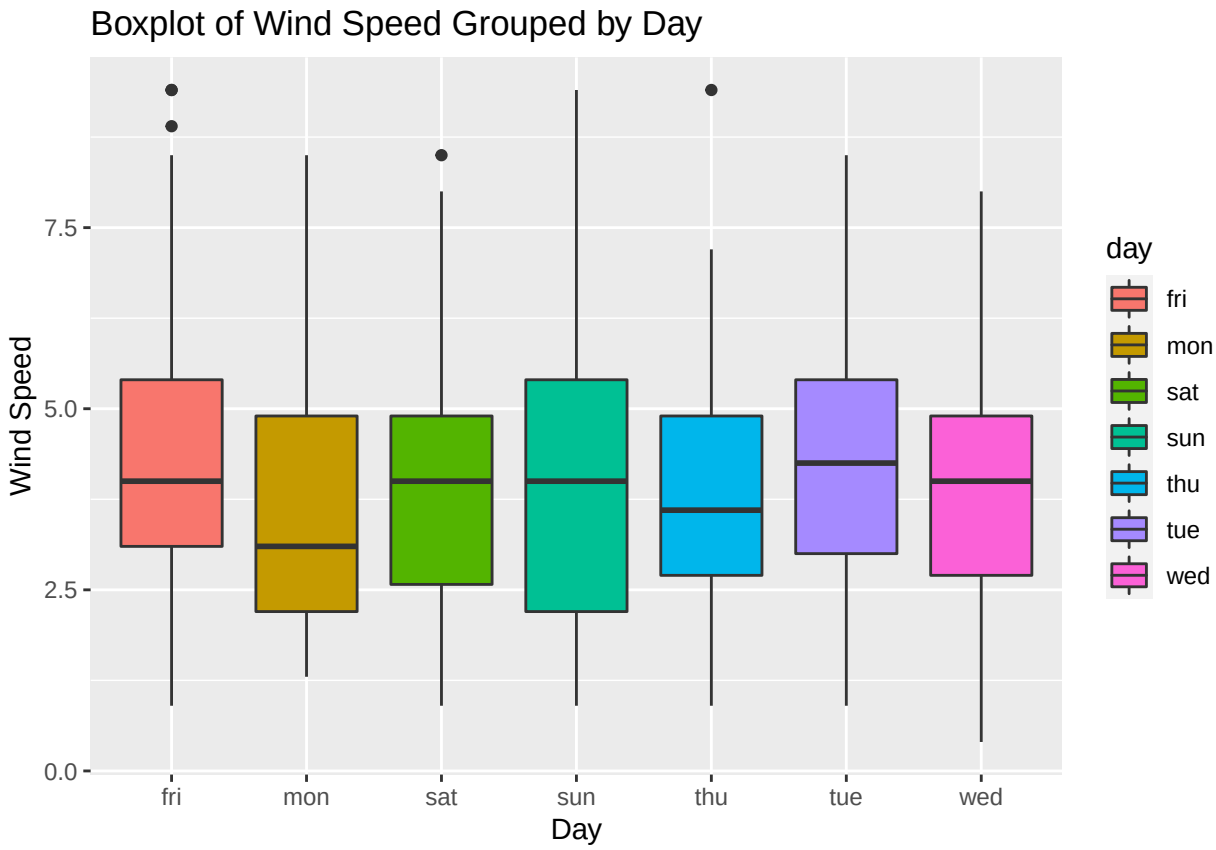


Interactions Between Categorical and Continuous Variables:

```
# Boxplot of temperature grouped by months
ggplot(data, aes(x=month, y=temp, fill=month)) +
  geom_boxplot() +
  labs(title="Boxplot of Temperature Grouped by Month", x="Month", y="Temperature")
```

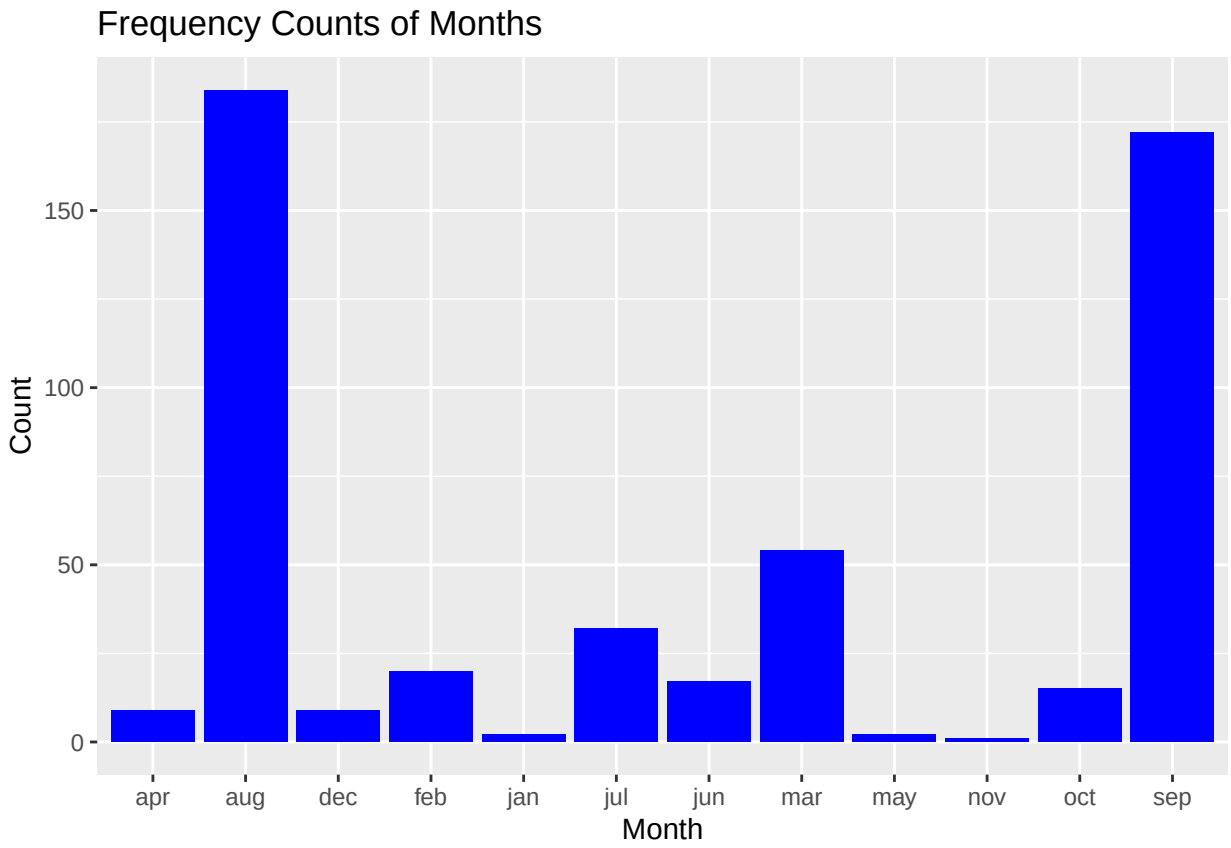



```
# Boxplot of wind speed grouped by day
ggplot(data, aes(x=day, y=wind, fill=day)) +
  geom_boxplot() +
  labs(title="Boxplot of Wind Speed Grouped by Day", x="Day", y="Wind Speed")
```

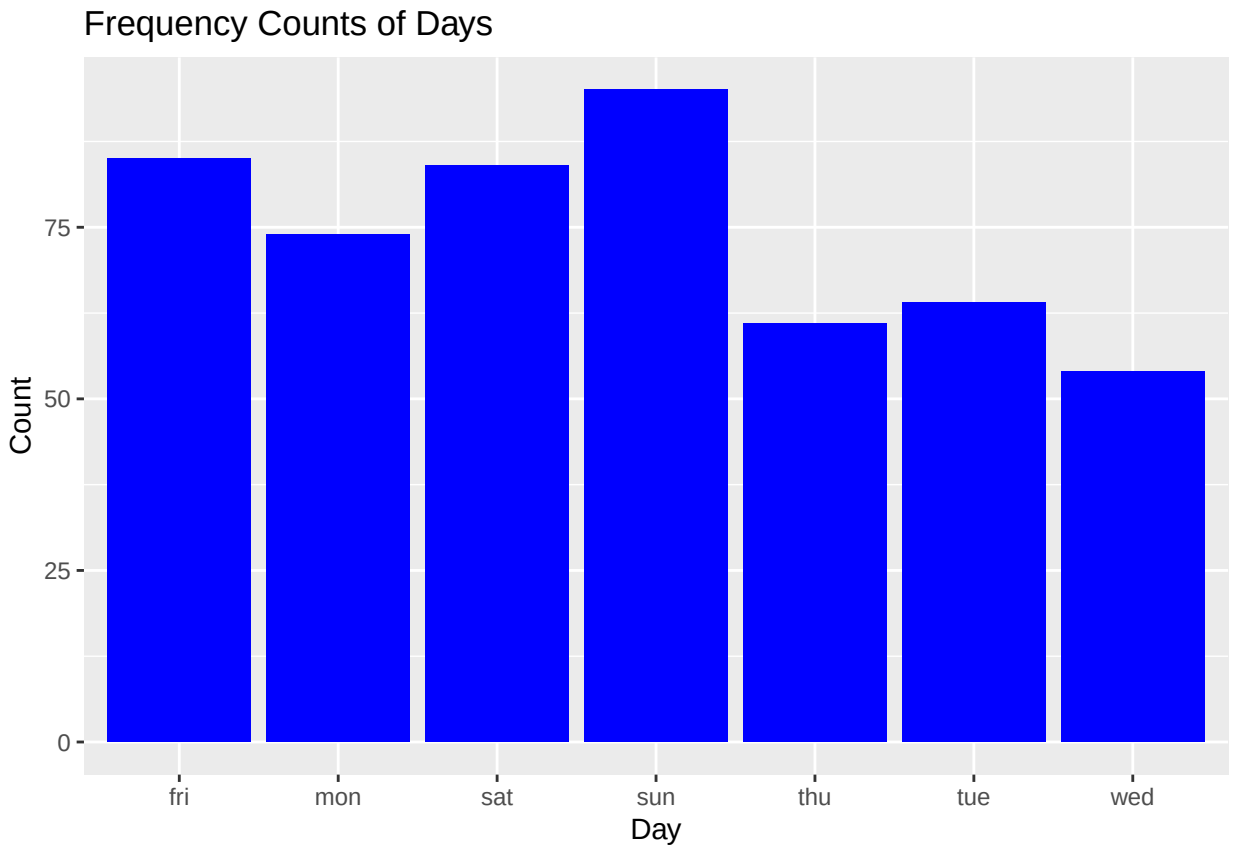


Frequency Counts of Categorical Variables:

```
# Frequency counts of months
ggplot(data, aes(x=month)) +
  geom_bar(fill='blue') +
  labs(title="Frequency Counts of Months", x="Month", y="Count")
```




```
# Frequency counts of days
ggplot(data, aes(x=day)) +
  geom_bar(fill='blue') +
  labs(title="Frequency Counts of Days", x="Day", y="Count")
```


Frequency Counts of Days
Count
75
50
25
0
fri
mon
sat
sun
thu
tue
wed
Day