Dataquest Guided Project: Analyzing Forest Fire Data

Cindy Zhang

8/14/2020

Contents

Introduction

This is my solution to Dataquest's Guided Project from the Data Visualization in R course, which visually examines the occurrence of forest fires in a Portugal park using the `ggplot2` package.

More details such as the RMD and csv files can be found in the repository in GitHub. Details about the variables can be found here.

Findings

Step 1: Examining When Forest Fires Occur

As always, first steps included loading the data and taking a look at the object types in the set:

```
forest_fires <- data.frame(read.csv("forestfires.csv"))
dim(forest_fires)
```

```
## [1] 517  13
```

```
colnames(forest_fires)
```

```
##  [1] "X"     "Y"     "month" "day"   "FFMC"  "DMC"   "DC"    "ISI"   "temp"
## [10] "RH"    "wind"  "rain"  "area"
```

```
for (i in colnames(forest_fires)) {
  print(class(forest_fires[[i]]))
}
```

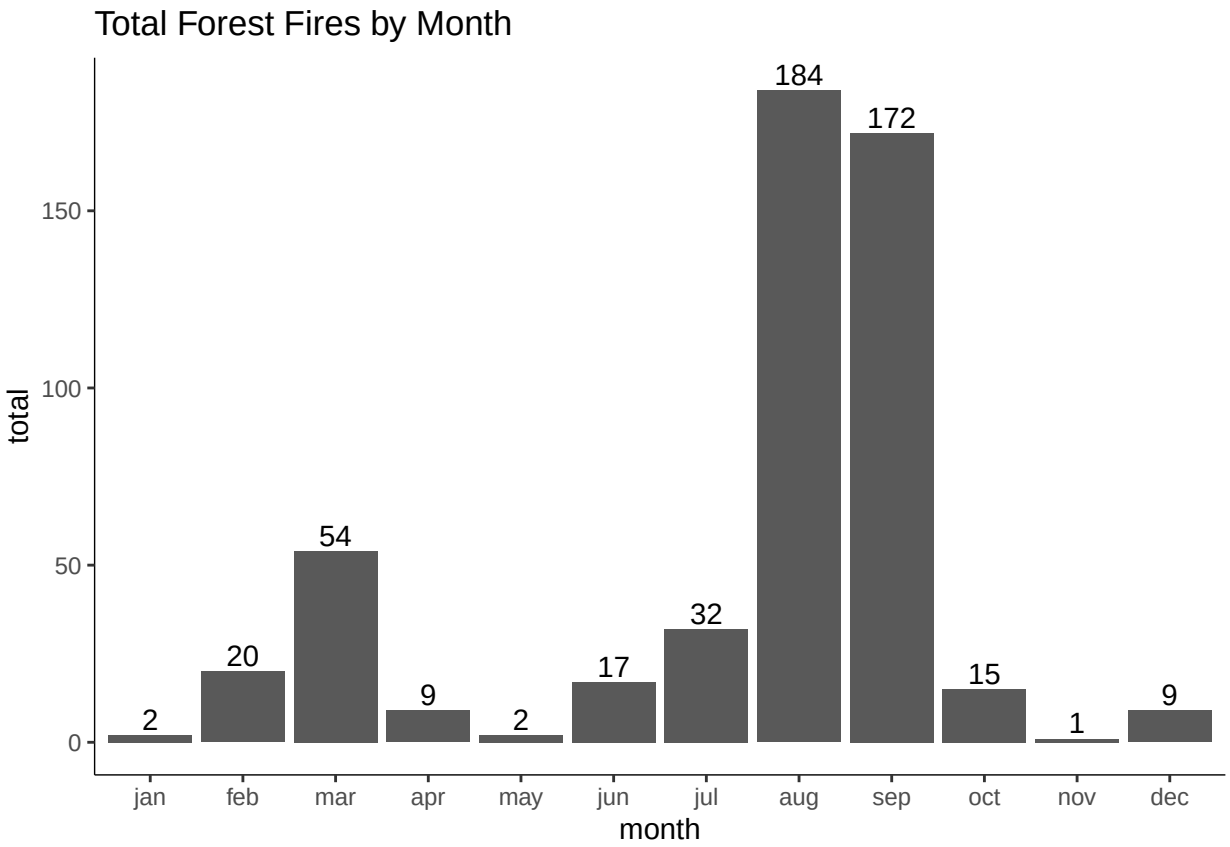
```
## [1] "integer"
## [1] "integer"
## [1] "character"
## [1] "character"
## [1] "numeric"
## [1] "numeric"
## [1] "numeric"
## [1] "numeric"
## [1] "numeric"
## [1] "integer"
## [1] "numeric"
## [1] "numeric"
## [1] "numeric"
```

After loading the data, I examined when the fires were most likely to occur on a monthly and daily basis.To do so, I first split the data into groups by month and day. I wanted to present the data as a bar graph in the correct calendar order so I added factor levels to each new dataframe:

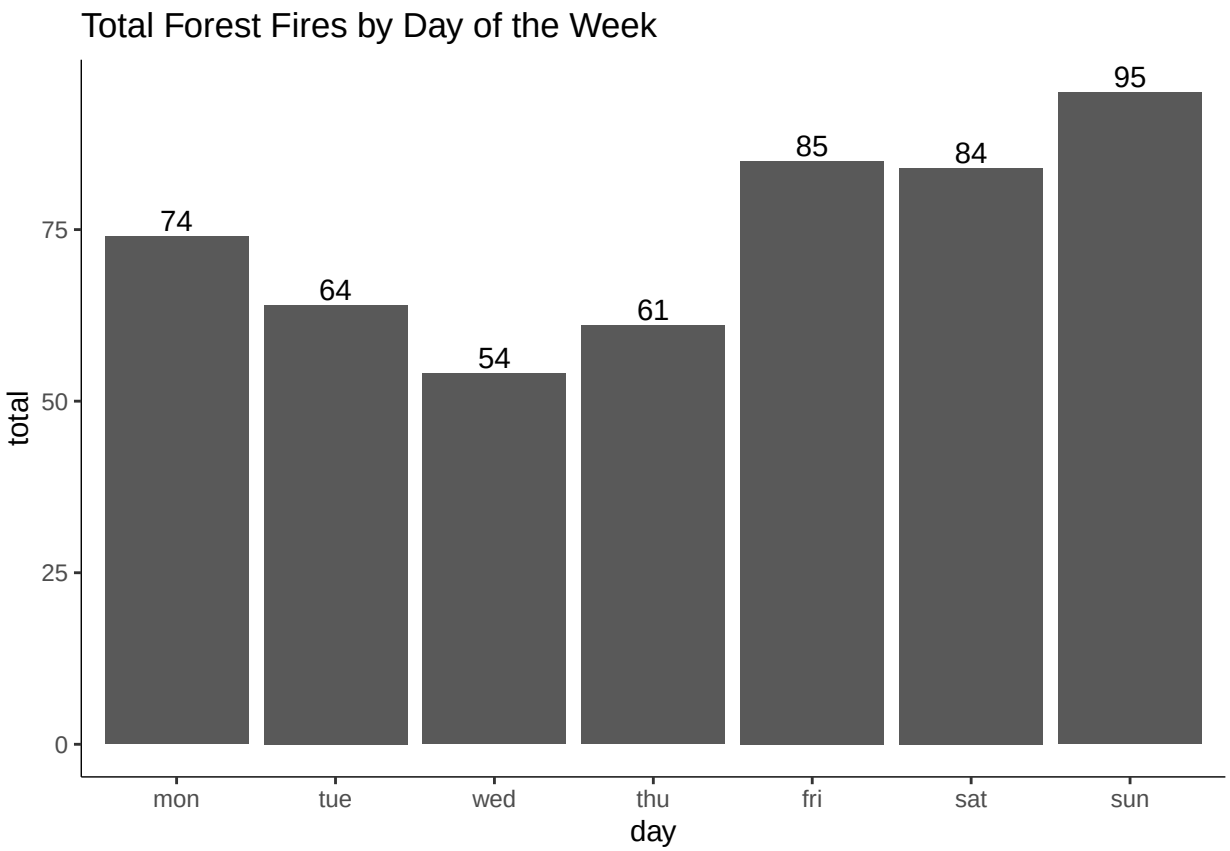
```
# Month dataframe
forest_fires_month <- forest_fires %>%
  mutate(month = factor(month, levels = c("jan", "feb", "mar", "apr", "may", "jun", "jul", "aug", "sep"
  group_by(month) %>%
  summarize(total=n())

#Weekday dataframe
forest_fires_day <- forest_fires %>%
  mutate(day = factor(day, levels = c("mon", "tue", "wed", "thu", "fri", "sat", "sun"))) %>%
  group_by(day) %>%
  summarize(total=n())
```

I then plotted each data frame into a bar graph:

```
# month graph
ggplot(data=forest_fires_month) +
  aes(x = month, y = total) +
  geom_bar(stat="identity") +
  geom_text(aes(label=total), position=position_dodge(width=0.9), vjust=-0.25) +
  labs(title= "Total Forest Fires by Month") +
  theme(panel.background=element_rect(fill="white"), axis.line = element_line(size=0.25, colour = "blac
```




```
#day graph
ggplot(data=forest_fires_day) +
  aes(x = day, y = total) +
  geom_bar(stat="identity") +
  geom_text(aes(label=total), position=position_dodge(width=0.9), vjust=-0.25) +
  labs(title = "Total Forest Fires by Day of the Week") +
  theme(panel.background=element_rect(fill="white"), axis.line = element_line(size=0.25, colour = "blac
```



Forest fires are more prevalent in during the months of August and September and on the weekends. Forest fire prevalence during hotter months of the year seems self-explanatory, but why are they more prevalent on the weekends?

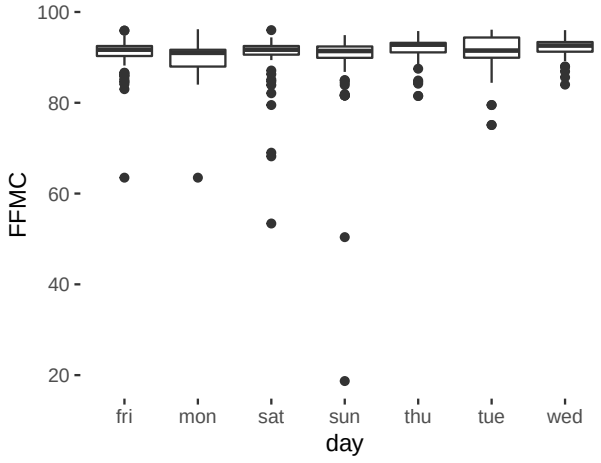
Step 2: Investigating Why Forest Fires Occur When They Do

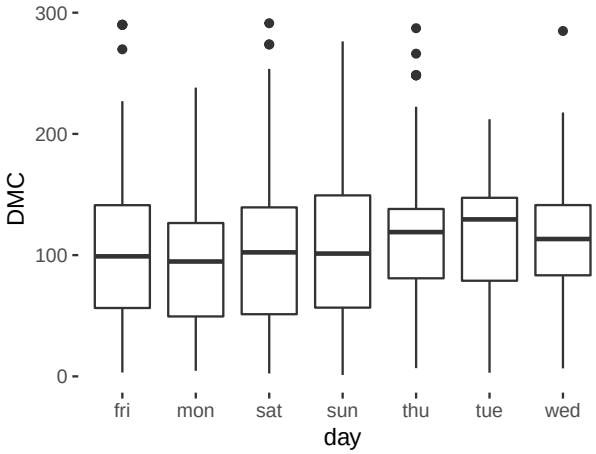
To explore causes of the temporal patterns of forest fires, I examined how the other variables in the dataset relate to forest fires by month and by day of the week. To make plotting each variable against the months and days of the year, I wrote a function that prints a boxplot for any variable affiliated with the `forest_fires` dataset. To use the `map2` function, I created a character vector for months, one for weekdays, and one with the names of the rest of the variables.

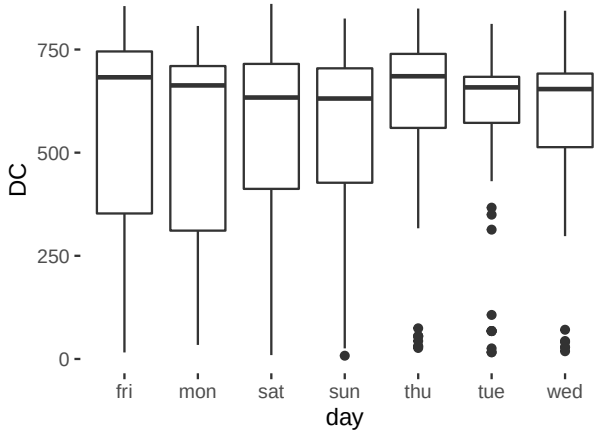
```
create_box <- function(time,y){
  ggplot(data=forest_fires) +
    aes_string(x=time, y=y) +
    geom_boxplot()+
    theme(panel.background=element_rect(fill="white"))
}

x_month <- names(forest_fires)[3]
x_day <- names(forest_fires)[4]
y_var <- names(forest_fires)[5:13]
```

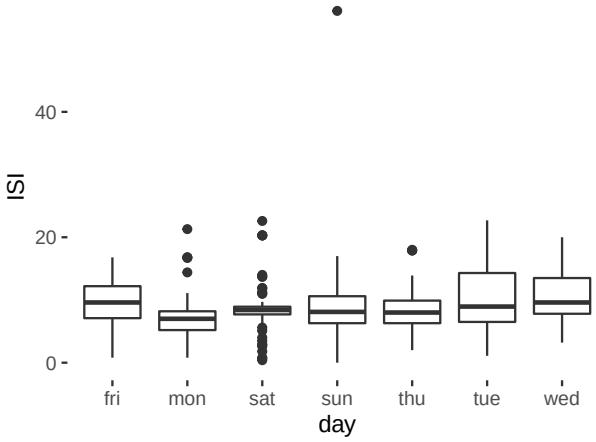
I then mapped the variables to the `create_box` function to produce the following results:

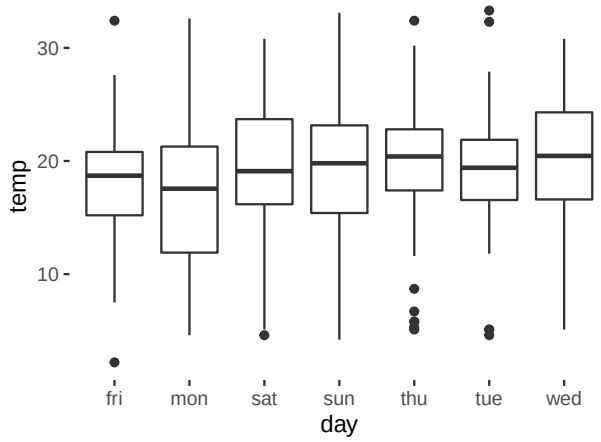
```
map2(x_day, y_var, create_box)
```

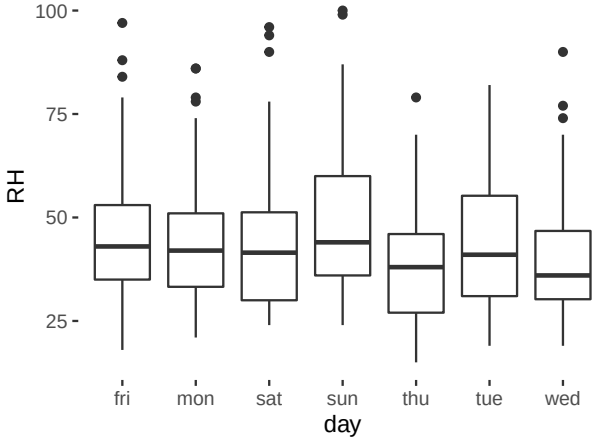


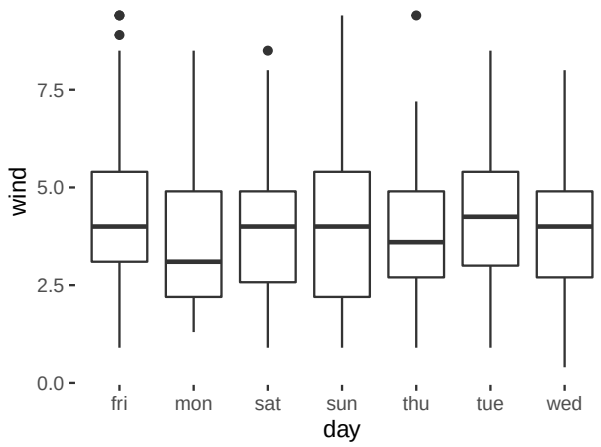


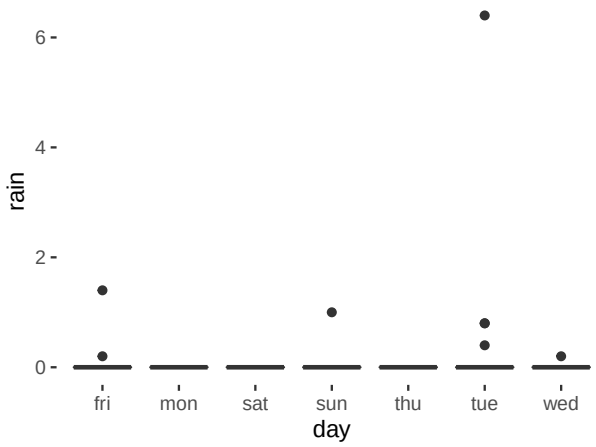


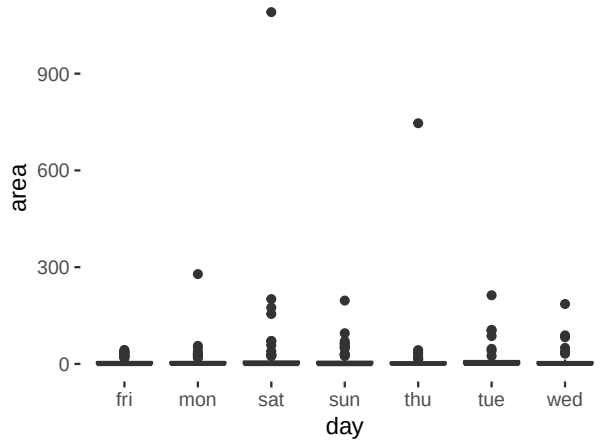






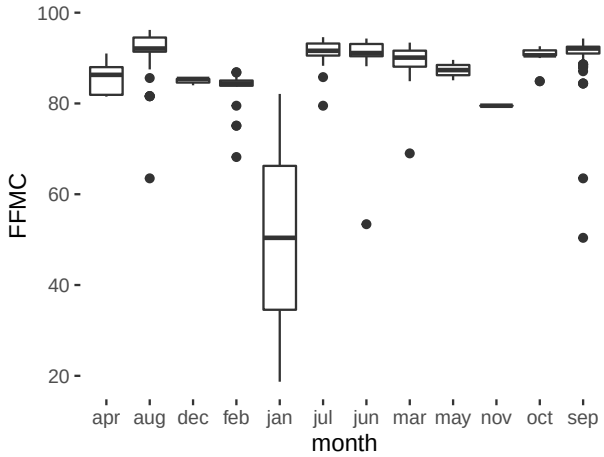


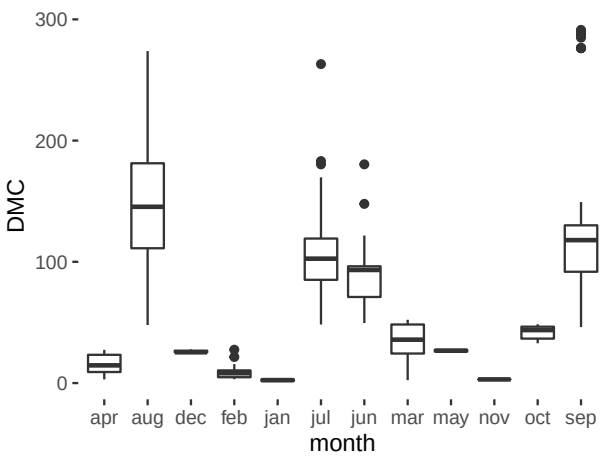


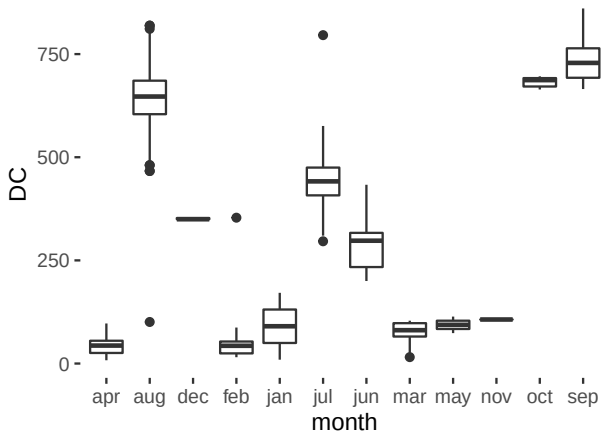


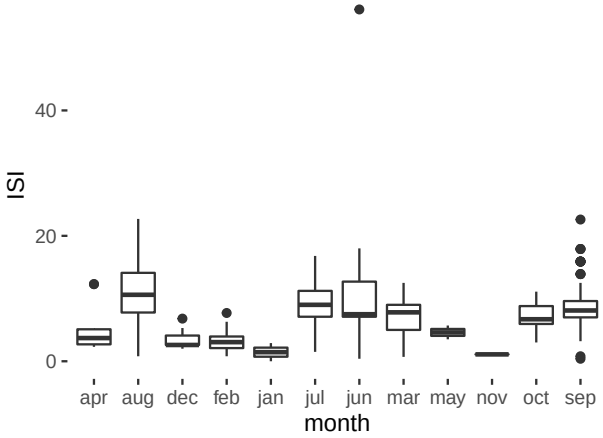
The medians and boxplot size (range) for each variable seem to be consistent across the days of the week. There are high and low outlier points that vary from day to day, but there do not seem to be any patterns that suggest the variables differ by day of the week. One possible cause of this trend of forest fire prevalence on the weekends is more people entering the park outside of regular work and school hours.

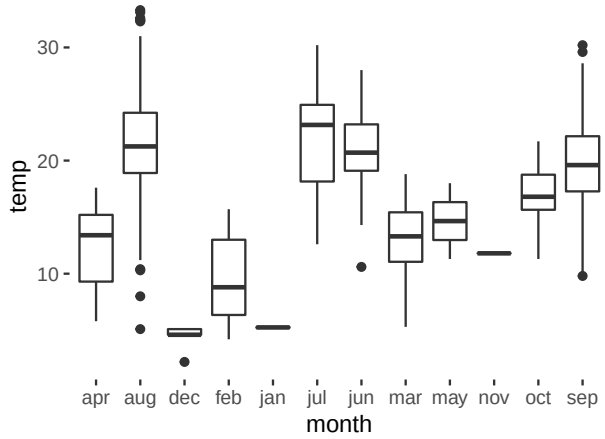
```
map2(x_month, y_var, create_box)
```

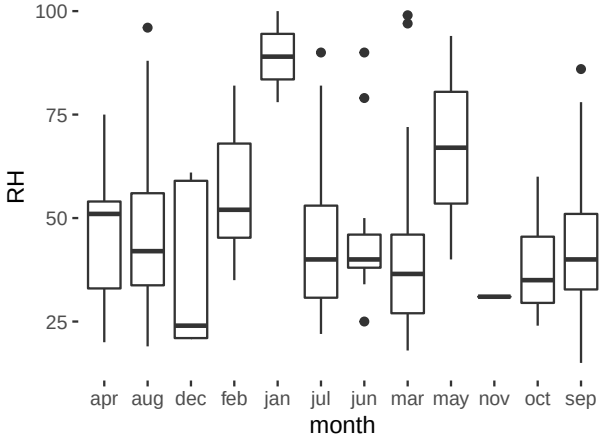


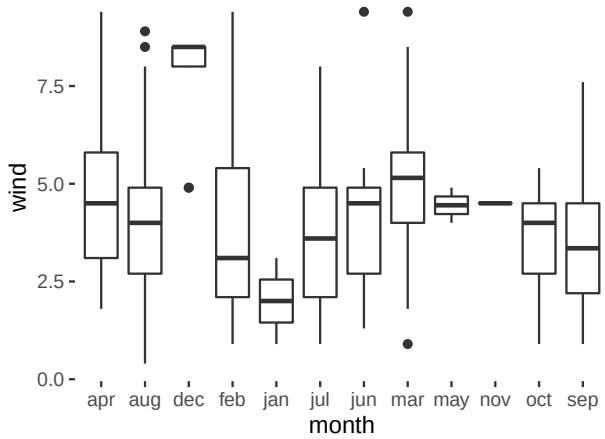


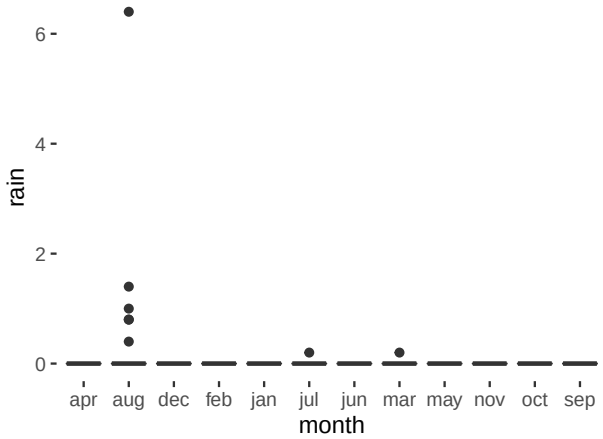

DC
750
500
250
0
apr aug dec feb jan jul jun mar may nov oct sep
month

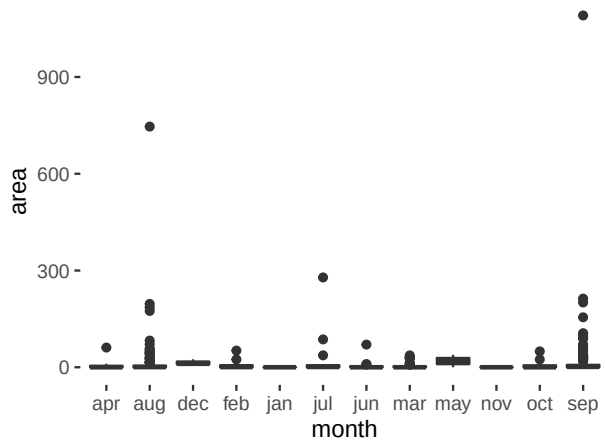

ISI
40
20
0
apr aug dec feb jan jul jun mar may nov oct sep
month


temp
30
20
10
apr aug dec feb jan jul jun mar may nov oct sep
month


RH
100
75
50
25
apr aug dec feb jan jul jun mar may nov oct sep
month


wind
7.5
5.0
2.5
0.0
apr aug dec feb jan jul jun mar may nov oct sep
month


rain
6
4
2
0
apr aug dec feb jan jul jun mar may nov oct sep
month



In contrast to the weekday boxplots, the month boxplots display clear differences for each variable among months. The `temp` and `DC` (drought code) variables in particular are highest during the late summer months, which corresponds with more forest fires during those months.

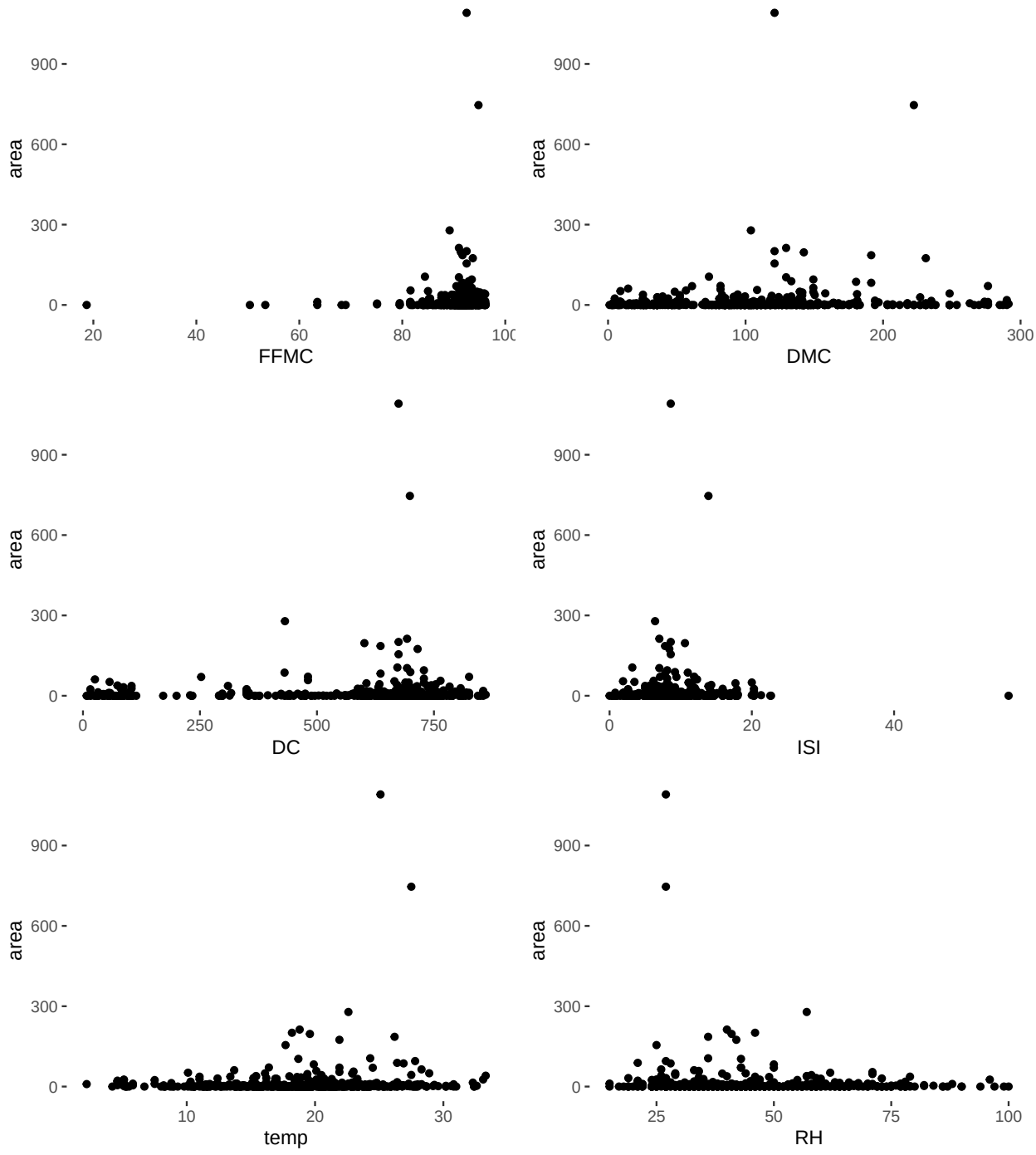
Step 3: Determining Which Variables Are Related To Forest Fire Severity

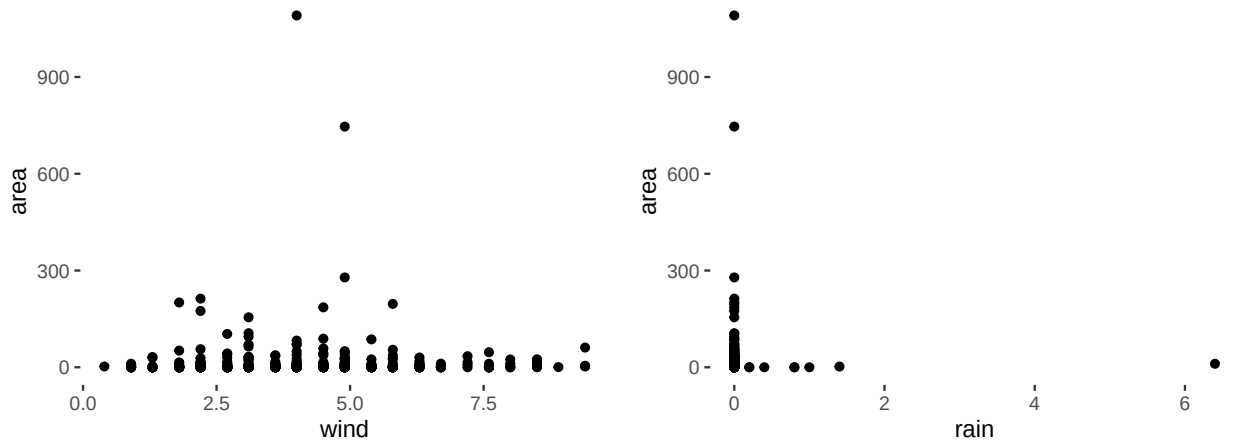
To figure out which variables are related to forest fire severity, I plotted several variables against the `area` variable, which measures the number of hectares of forest that burned down per forest fire. The larger the `area` of a forest fire, the worse the fire. As in Step 2, I wrote a function that prints a scatterplot for any variable affiliated with the `forest_fires` dataset.To use the `map2` function, I created a character vector containing the column title of the `area` variable and the column titles for the other variables:

```
create_scatter <- function(area, y) {
  ggplot(data=forest_fires)+
    aes_string(x=area, y=y)+
    geom_point()+
    theme(panel.background=element_rect(fill="white"))
}

area <- names(forest_fires)[13]
x_var <- names(forest_fires)[5:12]
```

I then mapped the variables to the `create_scatter` function to produce the following results:

```
map2(x_var, area, create_scatter)
```


area
900
600
300
0
20
40
60
80
FFMC
0
100
200
300
DMC
0
250
500
750
DC
0
20
40
ISI
10
20
30
temp
25
50
75
100
RH



Unfortunately, these scatterplots don't display notable trends or relationships as most points are clustered around the bottom.

Step 4: Deciding on Subsets of Data for Scatter Plots

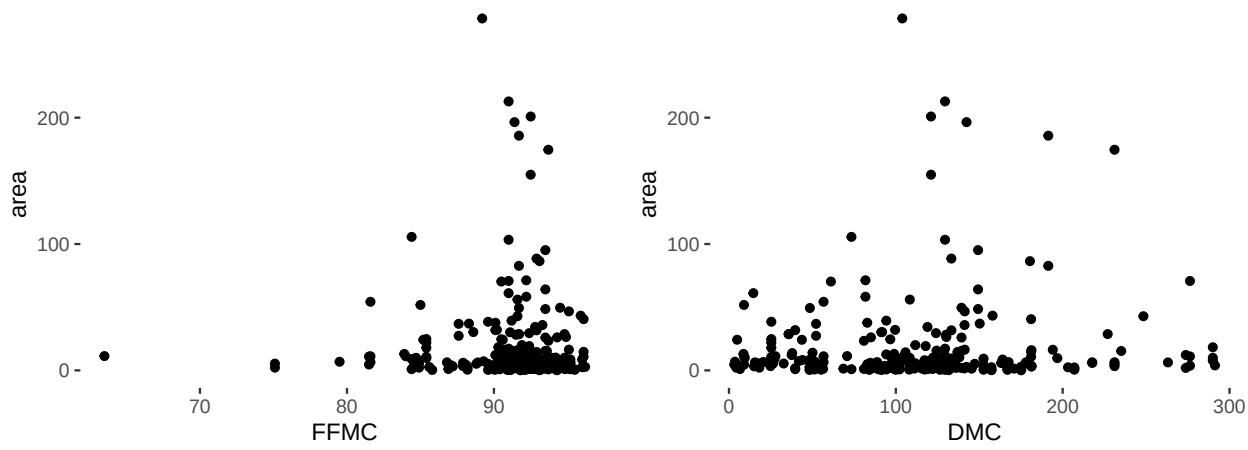
To more clearly visualize relationships between variables, I experimented with filtering the original `forest_fires` dataframe and plotting the subsets. I extracted a subsets that excludes data with area over 350 hectares adn data with area of 0 hectares.

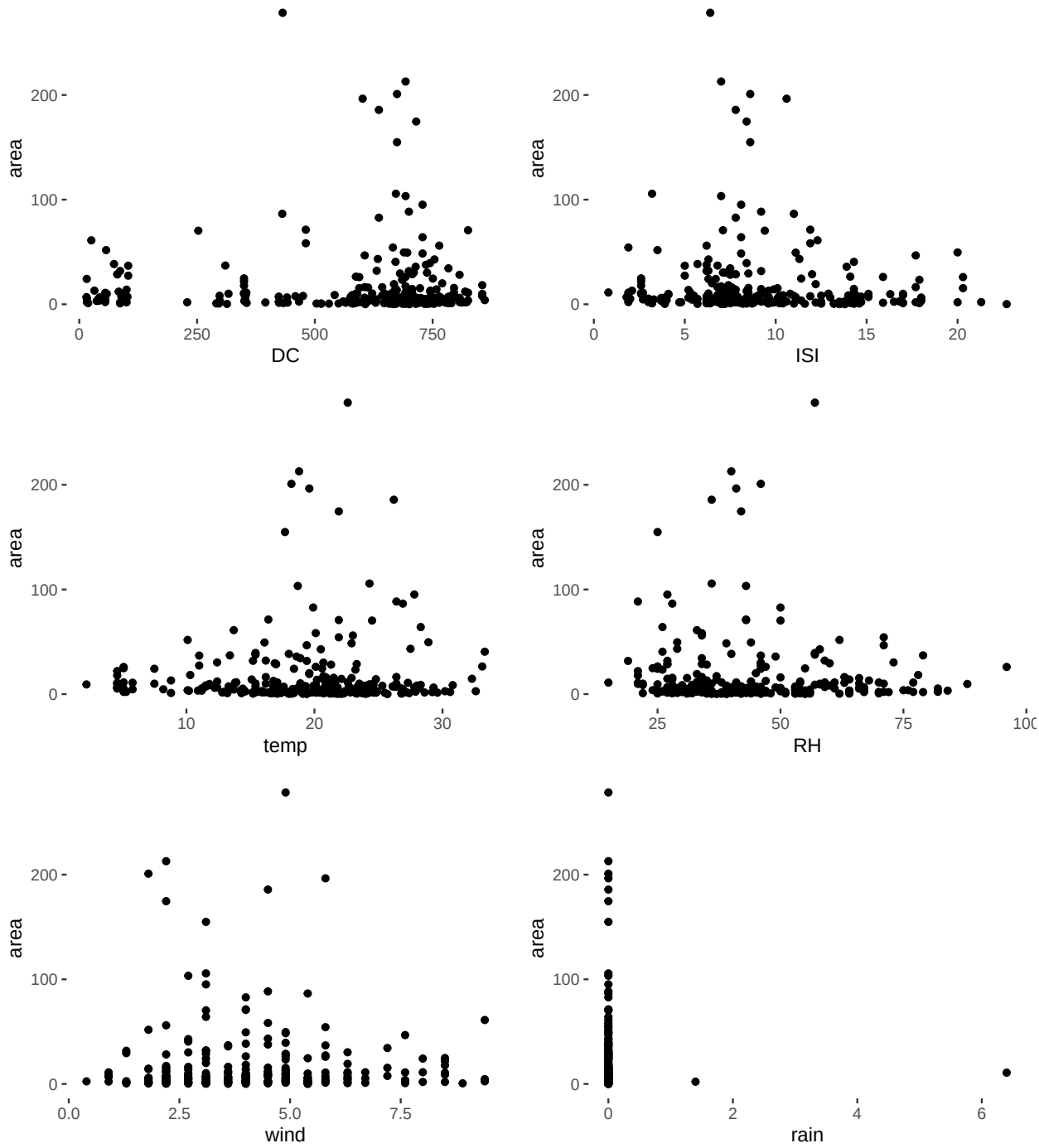
```
forest_fires_range <- subset(forest_fires, area < 350 & area > 0)
```

I rewrote the function used in Step 3 by replacing the original `forest_fires` data with the subsets and re-mapped the `area` and `x_var` vectors to the new function.

```
create_scatter_subset <- function(area, y) {
  ggplot(data=forest_fires_range)+
    aes_string(x=area, y=y)+
    geom_point()+
    theme(panel.background=element_rect(fill="white"))
}
```

```
map2(x_var, area, create_scatter_subset)
```





The scatterplots of the subsetted data show a clearer picture of how variables are related to forest fire severity. A higher FFMC (Fine Fuel Moisture Code), higher DC (drought code), low humidity, and low levels of rain are related to more severe forest fires (although that is not always the case for all fires).

Conclusion

Plotting the `forest_fires` data set in a variety of ways displays how certain weather elements can have an effect on forest fire severity. However, as I've mentioned in past projects, the more accurate way of

determining relationships between variables is regression analysis.

Thanks to everyone who's been reading so far and stay tuned for more projects!