Algerian_Forest_Fire_Final_Project

Peter Conant

2025-03-24

Introduction

Living in this day, I am constantly reminded that global warming is creating a climate crisis. Growing up in the central valley of California fires and smoke are on the for front of my mind. It has always been a dream of mine to help prevent forest fires and protect our habitats with the skills I learn in this program. This is a great opportunity to apply what I am learning to a topic that is important to me. I will be approaching this project as though I am a environmental researching working with the California forest protection agency using Algerian data to try to understand the conditions of where and when fires will occur. I want to use the skills and strategies I have learned to find a better way to predict and prevent forest fires.

Source

My project will be centered around the 'Algerian Forest Fires' found on UCI's Machine Learning Repository. The original article this data was collected for was 'Predicting Forest Fire in Algeria Using Data Mining Techniques: Case Study of the Decision Tree Algorithm' and completed a very similar task to the one I hope to complete. However they are using decision tree algorithms to make their predictions where as I hope to only use the data analyses tools we learn in this class.

The data set is a collection of meteorological observations (observational study) across during June 2012 to September 2012 in Algerian. It regroups data sets that were collected in two regions: the Sidi Bel-abbas region and the Bejaia region. Each region had a total of 122 instances, each row represents a day observed, totaling to 244 instances.

Temperature, relative humidity, wind speed, and rain are base data needed to calculate the codes: Fine FuelMoister Code, Duff Moister Code and Drought Code, which in turn are used to calculate the indexes: Initial Spread Index, Buildup Index, Fire Weather Index (FWI), a universal system for determining fire risk. I consider date, location, temperature, wind, relative humidity, fire, and rain to be raw data. In this project I will be using raw data to make my own predictions and comparing my results with the FWI.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.4.2
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

Data Quality

We will begin by importing our data processing it.

```
fires <- read.csv('Algerian_forest_fires_dataset.csv')

fires <- as_tibble(fires)

#summary
print(head(fires), width = Inf)
```

```
## # A tibble: 6 x 14
##   day   month year  Temperature RH    Ws    Rain  FFMC  DMC   DC    ISI   BUI
##   <chr> <chr> <chr> <chr>       <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 01    06    2012  29          57    18    0     65.7  3.4   7.6   1.3   3.4
## 2 02    06    2012  29          61    13    1.3   64.4  4.1   7.6   1     3.9
## 3 03    06    2012  26          82    22    13.1  47.1  2.5   7.1   0.3   2.7
## 4 04    06    2012  25          89    13    2.5   28.6  1.3   6.9   0     1.7
## 5 05    06    2012  27          77    16    0     64.8  3     14.2  1.2   3.9
## 6 06    06    2012  31          67    14    0     82.6  5.8   22.2  3.1   7
##   FWI   Classes
##   <chr> <chr>
## 1 0.5   "not fire   "
## 2 0.4   "not fire   "
## 3 0.1   "not fire   "
## 4 0     "not fire   "
## 5 0.5   "not fire   "
## 6 2.5   "fire   "
```

```
summary(fires)
```

```
##      day                month              year           Temperature
##  Length:247         Length:247         Length:247         Length:247
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##       RH                 Ws                 Rain               FFMC
##  Length:247         Length:247         Length:247         Length:247
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##      DMC                DC                 ISI                BUI
##  Length:247         Length:247         Length:247         Length:247
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##      FWI              Classes
##  Length:247         Length:247
##  Class :character   Class :character
##  Mode  :character   Mode  :character
```

First I will check for missing and duplicate values to remove them.

```
fires_clean <- fires

#NA values
fires_clean <- fires_clean %>% mutate(across(everything(), ~na_if(., "")))

colSums(is.na(fires_clean))
```

```
##         day       month        year Temperature          RH          Ws
##           1           2           2           2           2           2
##        Rain        FFMC         DMC          DC         ISI         BUI
##           2           2           2           2           2           2
##         FWI     Classes
##           2           3
```

```
rows_with_na <- fires_clean[!complete.cases(fires_clean), ]

# Print rows with NA values
print(rows_with_na)
```

```
## # A tibble: 3 x 14
##   day    month year  Temperature RH    Ws    Rain  FFMC  DMC   DC    ISI   BUI
##   <chr>  <chr> <chr> <chr>       <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 <NA>   <NA>  <NA>  <NA>        <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 2 Sidi-~ <NA>  <NA>  <NA>        <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>  <NA>
## 3 14     07    2012  37          37    18    0.2   88.9  12.9  14.6~ 12.5  10.4
## # i 2 more variables: FWI <chr>, Classes <chr>
```

Three rows with NA values. The top two rows separate the regional data. The third has a NA Classes (fire) value. All will be removed

```
fires_clean <- na.omit(fires_clean)

colSums(is.na(fires_clean))
```

```
##         day       month        year Temperature          RH          Ws
##           0           0           0           0           0           0
##        Rain        FFMC         DMC          DC         ISI         BUI
##           0           0           0           0           0           0
##         FWI     Classes
##           0           0
```

```
#Remove second column name row
fires_clean <- fires_clean[-123, ]

#duplicates
table(duplicated(fires_clean))
```

```
##
## FALSE
##   243
```

NA values removed and no duplicates found. The data is clean.

The values of the Class column representing if there was a fire on that day should be changed to a binary 0 representing no fire and 1 for fire. This will make future calculations easier.

```
fires_formatted <- fires_clean
```

```
fires_formatted$Classes <- trimws(fires_formatted$Classes)

fires_formatted$Classes<- replace(fires_formatted$Classes, fires_formatted$Classes == 'not fire', 0)
fires_formatted$Classes<- replace(fires_formatted$Classes, fires_formatted$Classes == 'fire', 1)
```

I will turn all numeric character values into integers.

```
fires_formatted <- fires_formatted %>%
  mutate(across(where(is.character), as.numeric))
```

I can make a Date column out of the day month and year provided.

```
fires_formatted$Date <- as.Date(paste(fires_formatted$year, fires_formatted$month, fires_formatted$day,
```

The data is split between two regions but there is no column to represent this. The first 122 rows represent the Bejaia region and rows 123 to 244 are from Sidi-Bel Abbes Region. I may make this column binary later when running calculations but I will leave it for now for readability.

```
fires_formatted$Region <- NA

fires_formatted$Region[1:122] <- "Bejaia"

fires_formatted$Region[123:nrow(fires_formatted)] <- "Sidi-Bel Abbes"
```

I will also change the name of the columns to make it easier to read.

```
colnames(fires_formatted) <- c('Day','Month','Year','Temperature','Relative Humidity','Wind Speed','Rai
```

```
print(head(fires_formatted), n = Inf, width = Inf)
```

```
## # A tibble: 6 x 16
##     Day Month  Year Temperature `Relative Humidity` `Wind Speed`  Rain
##   <dbl> <dbl> <dbl>       <dbl>               <dbl>        <dbl> <dbl>
## 1     1     6  2012          29                  57           18   0
## 2     2     6  2012          29                  61           13   1.3
## 3     3     6  2012          26                  82           22  13.1
## 4     4     6  2012          25                  89           13   2.5
## 5     5     6  2012          27                  77           16   0
## 6     6     6  2012          31                  67           14   0
##   `Fine Fuel Moister Code` `Duff Moister Code` `Drought Code`
##                      <dbl>               <dbl>          <dbl>
## 1                     65.7                 3.4            7.6
## 2                     64.4                 4.1            7.6
## 3                     47.1                 2.5            7.1
## 4                     28.6                 1.3            6.9
## 5                     64.8                 3             14.2
## 6                     82.6                 5.8           22.2
##   `Initial Spread Index` `Buildup Index` `Fire Weather Index`  Fire Date
##                    <dbl>           <dbl>                <dbl> <dbl> <date>
## 1                    1.3             3.4                  0.5     0 2012-06-01
## 2                    1               3.9                  0.4     0 2012-06-02
## 3                    0.3             2.7                  0.1     0 2012-06-03
## 4                    0               1.7                  0       0 2012-06-04
## 5                    1.2             3.9                  0.5     0 2012-06-05
## 6                    3.1             7                    2.5     1 2012-06-06
##   Region
##   <chr>
```

```
## 1 Bejaia
## 2 Bejaia
## 3 Bejaia
## 4 Bejaia
## 5 Bejaia
## 6 Bejaia
```

Exploratory Data Analysis

Lets begin by computing summary statistics.

```
#Variance
numeric_variances <- fires_formatted %>%
  summarise(across(where(is.numeric), var, na.rm = TRUE))
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(where(is.numeric), var, na.rm = TRUE)`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```
print(numeric_variances, n = Inf, width = Inf)
```

```
## # A tibble: 1 x 14
##     Day Month  Year Temperature `Relative Humidity` `Wind Speed`  Rain
##   <dbl> <dbl> <dbl>       <dbl>               <dbl>        <dbl> <dbl>
## 1  78.2  1.24     0        13.2                220.         7.90  4.01
##   `Fine Fuel Moister Code` `Duff Moister Code` `Drought Code`
##                      <dbl>               <dbl>          <dbl>
## 1                     206.                154.          2272.
##   `Initial Spread Index` `Buildup Index` `Fire Weather Index`  Fire
##                    <dbl>           <dbl>                <dbl> <dbl>
## 1                   17.3            202.                 55.4 0.247
```

```
getmode <- function(v) {
  v <-v[!is.na(v)]

  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v,uniqv)))]
}

numeric_mode <- fires_formatted %>%
  summarise(across(where(is.numeric), getmode))

print(numeric_mode, n = Inf, width = Inf)
```

```
## # A tibble: 1 x 14
##     Day Month  Year Temperature `Relative Humidity` `Wind Speed`  Rain
##   <dbl> <dbl> <dbl>       <dbl>               <dbl>        <dbl> <dbl>
## 1     1     8  2012          35                  55           14     0
```

```
##   `Fine Fuel Moister Code` `Duff Moister Code` `Drought Code`
##                      <dbl>                <dbl>          <dbl>
## 1                     88.9                  7.9              8
##   `Initial Spread Index` `Buildup Index` `Fire Weather Index`  Fire
##                    <dbl>           <dbl>                <dbl> <dbl>
## 1                    1.1               3                  0.4     1
```

```
#Select one categorical variable, compute these statistics on a numeric variable by grouping on a categ
```

```
fires_formatted %>% summarise(mean_temp = mean(Temperature, na.rm = TRUE), median_temp = median(Tempera
```

```
## # A tibble: 1 x 6
##   mean_temp median_temp sd_temp range_temp mode_temp var_temp
##       <dbl>       <dbl>   <dbl> <chr>          <dbl>    <dbl>
## 1      32.2          32    3.63 22 - 42           35     13.2
```

```
fires_formatted %>% summarise(mean_rain = mean(Rain, na.rm = TRUE), median_rain = median(Rain, na.rm = 
```

```
## # A tibble: 1 x 6
##   mean_rain median_rain sd_rain range_rain mode_rain var_wind
##       <dbl>       <dbl>   <dbl> <chr>          <dbl>    <dbl>
## 1     0.763           0    2.00 0 - 16.8           0     4.01
```

```
fires_formatted %>% summarise(mean_humidity = mean(`Relative Humidity`, na.rm = TRUE), median_humidity =
```

```
## # A tibble: 1 x 6
##   mean_humidity median_humidity sd_humidity range_humidity mode_humidity
##           <dbl>           <dbl>       <dbl> <chr>                  <dbl>
## 1          62.0              63        14.8 21 - 90                   55
## # i 1 more variable: var_humidity <dbl>
```

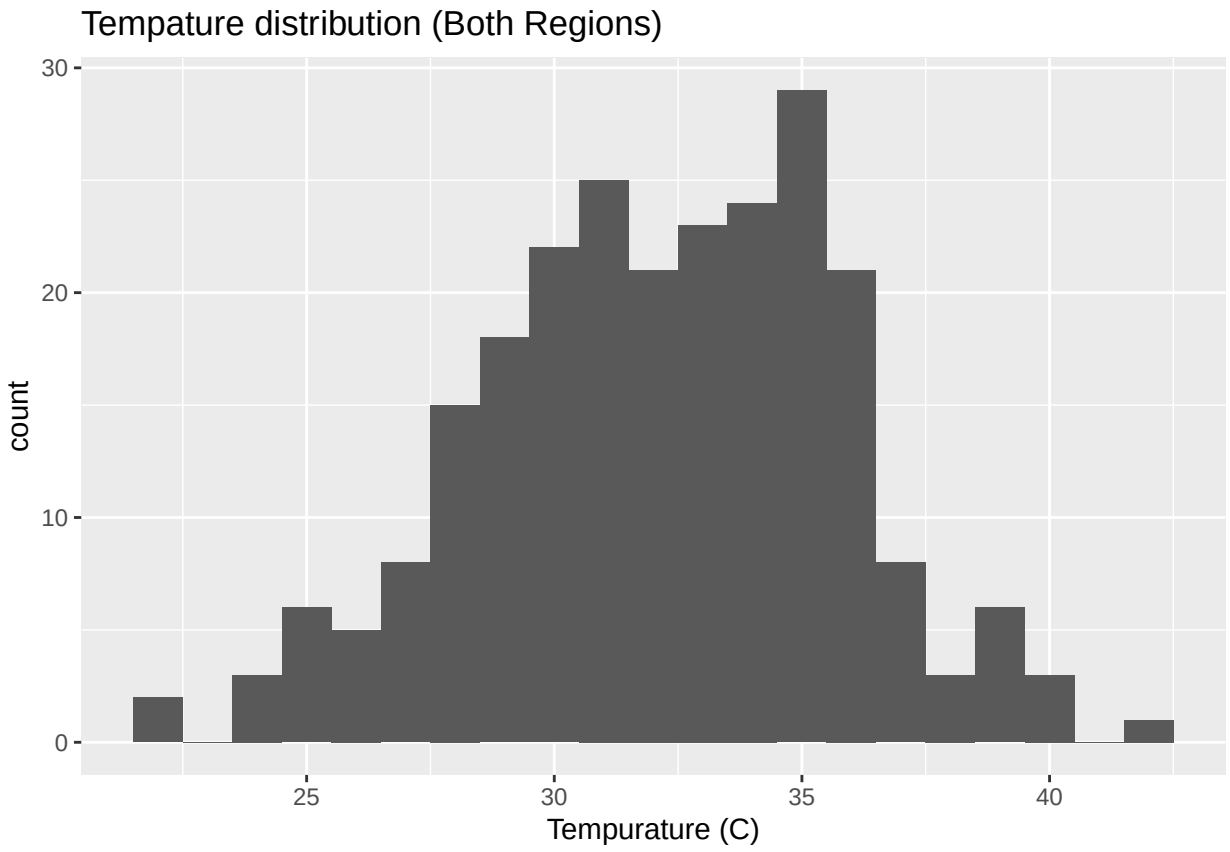
```
fires_formatted %>% summarise(mean_wind = mean(`Wind Speed`, na.rm = TRUE), median_wind = median(`Wind S
```

```
## # A tibble: 1 x 6
##   mean_wind median_wind sd_wind range_wind mode_wind var_wind
##       <dbl>       <dbl>   <dbl> <chr>          <dbl>    <dbl>
## 1      15.5          15    2.81 6 - 29            14     7.90
```

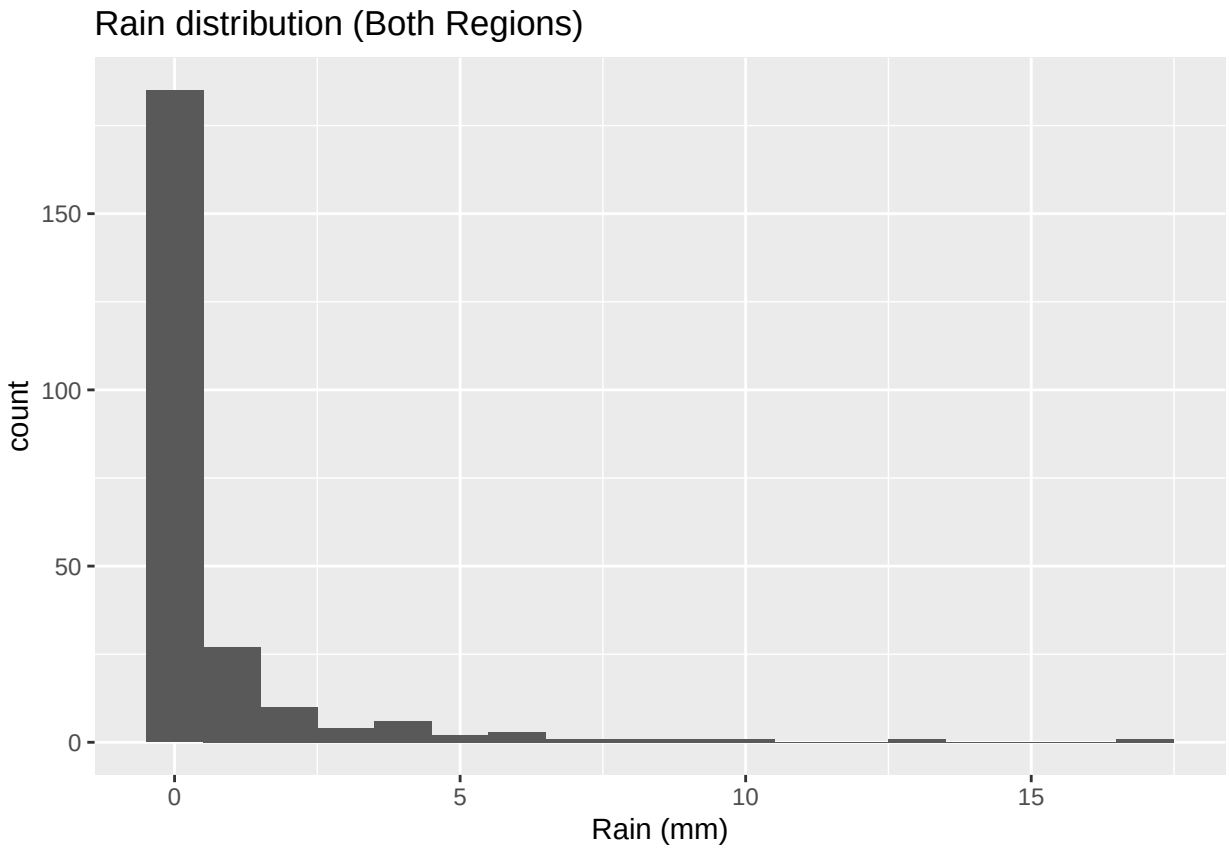
It seems we have a mean temperature of 32 Celsius, an mean rain of .76 mm a day and a mean humidity of 62. Wind speeds can range from 6 - 29 km/h but typically stays between 12 and 18 km/h.

Lets see the distribution of this data.

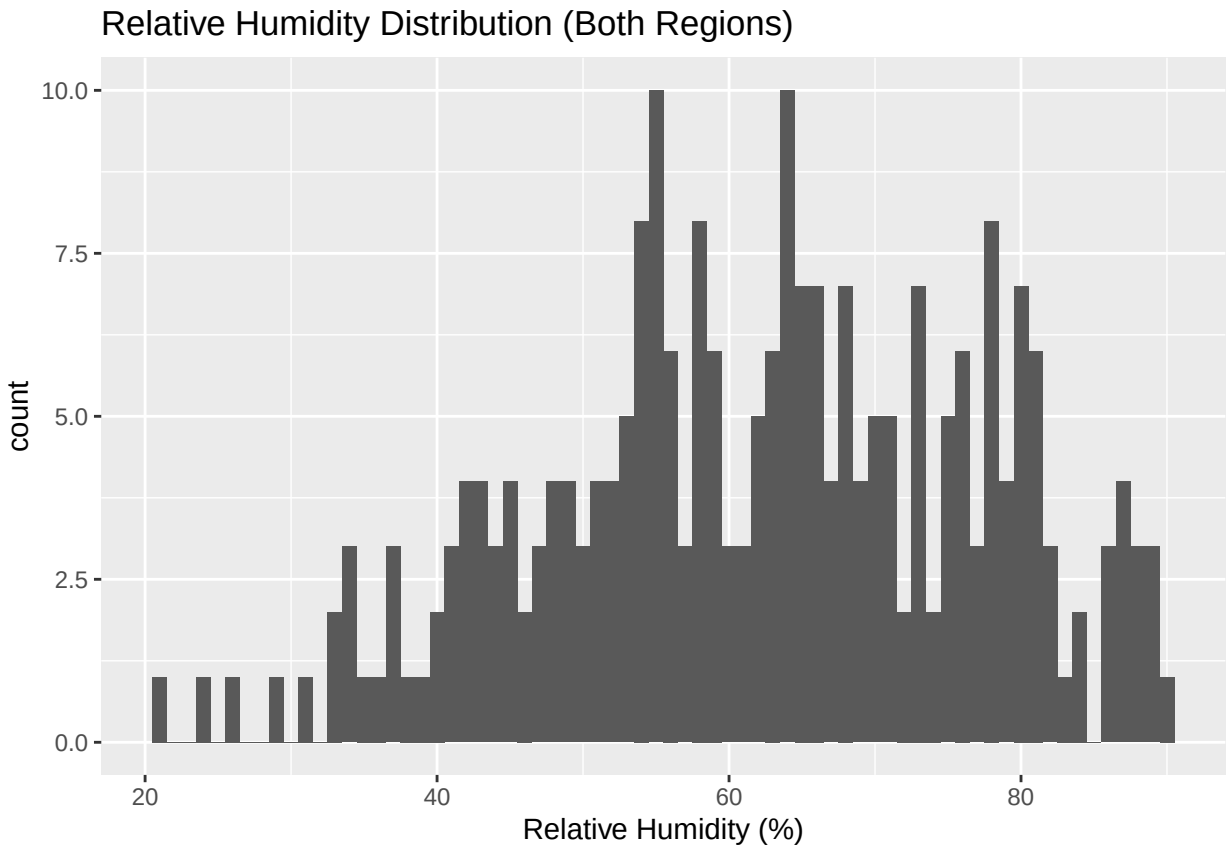
```
ggplot(data = fires_formatted) + geom_histogram(mapping = aes(x = Temperature), binwidth = 1)+ labs(x =
```



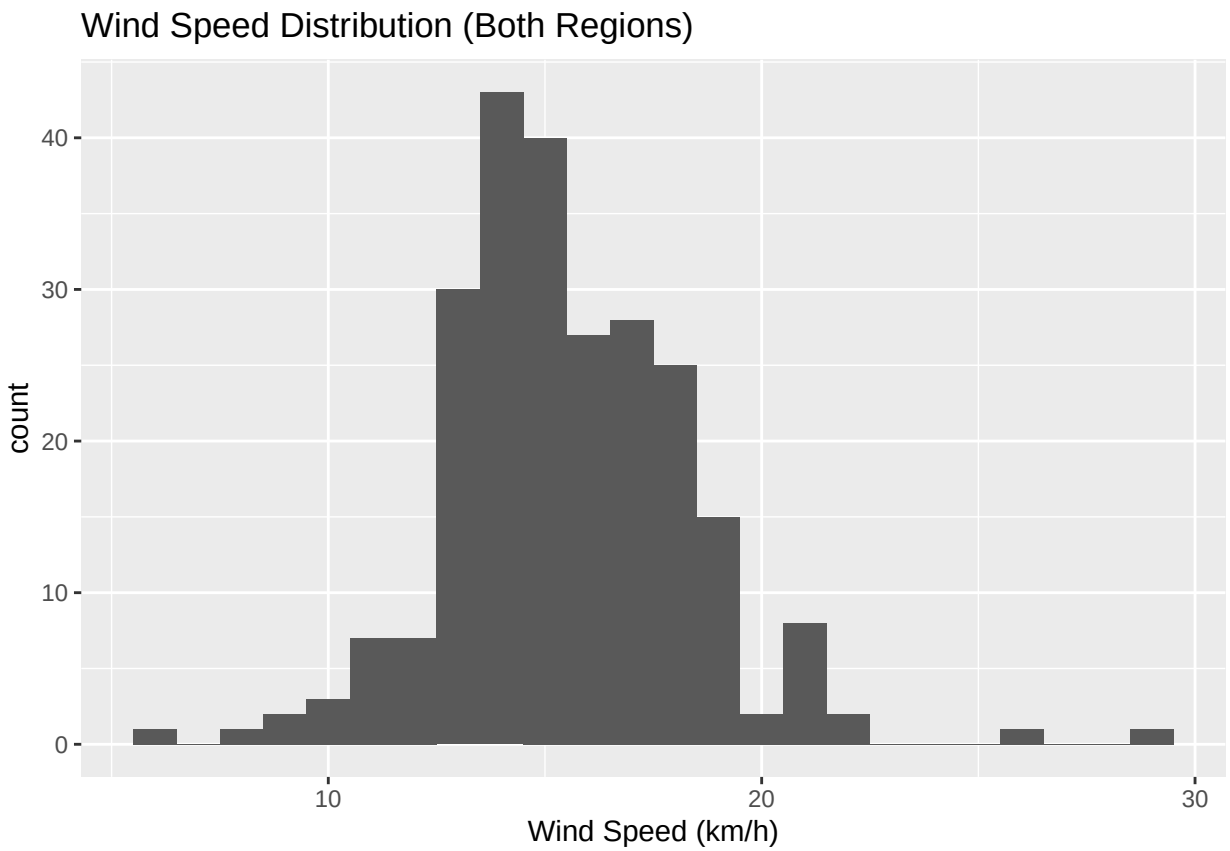

```
ggplot(data = fires_formatted) + geom_histogram(mapping = aes(x = Rain), binwidth = 1)+ labs(x = "Rain
```




```
ggplot(data = fires_formatted) + geom_histogram(mapping = aes(x = `Relative Humidity`), binwidth = 1) +
```

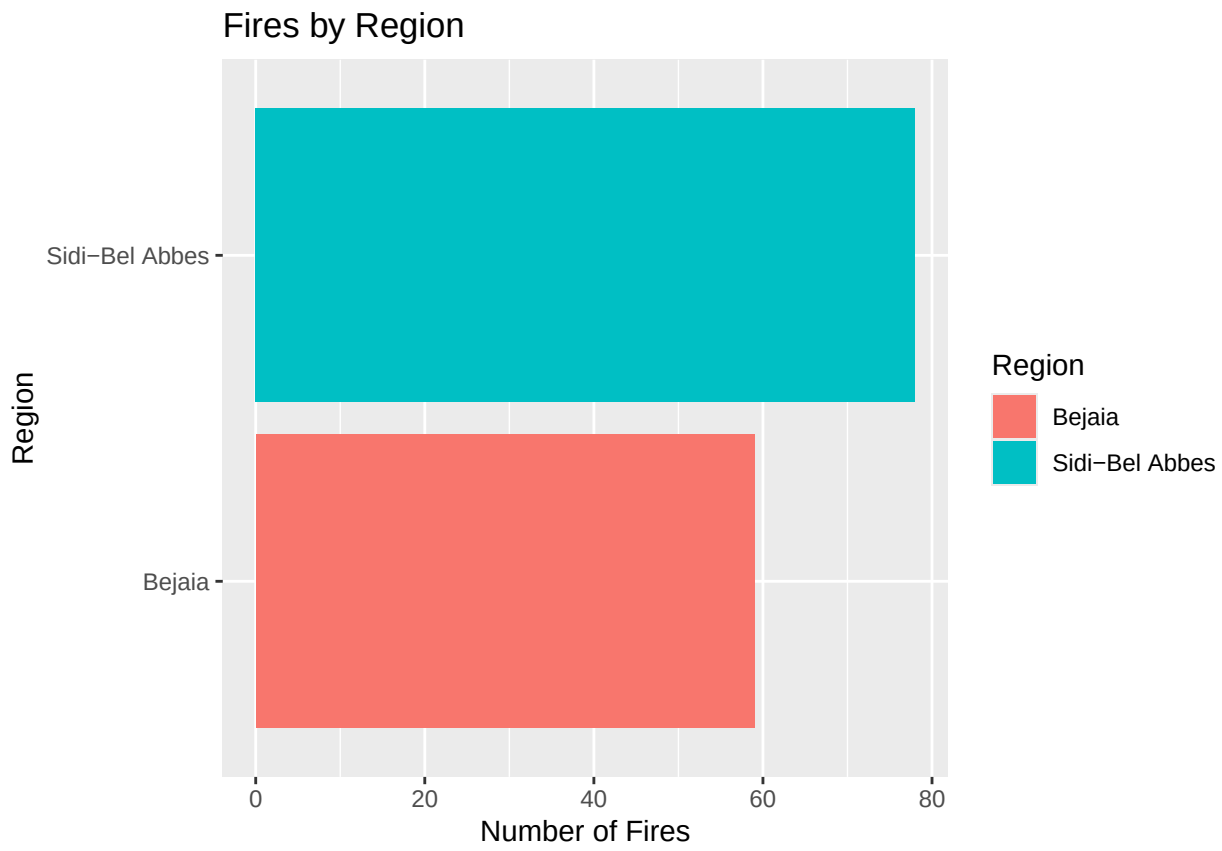



```
ggplot(data = fires_formatted) + geom_histogram(mapping = aes(x = `Wind Speed`), binwidth = 1) + labs(x
```



We should get an idea of where and when fires happen. To get a sense for the total number distribution of fires of the time frame we'll do some visualizations.

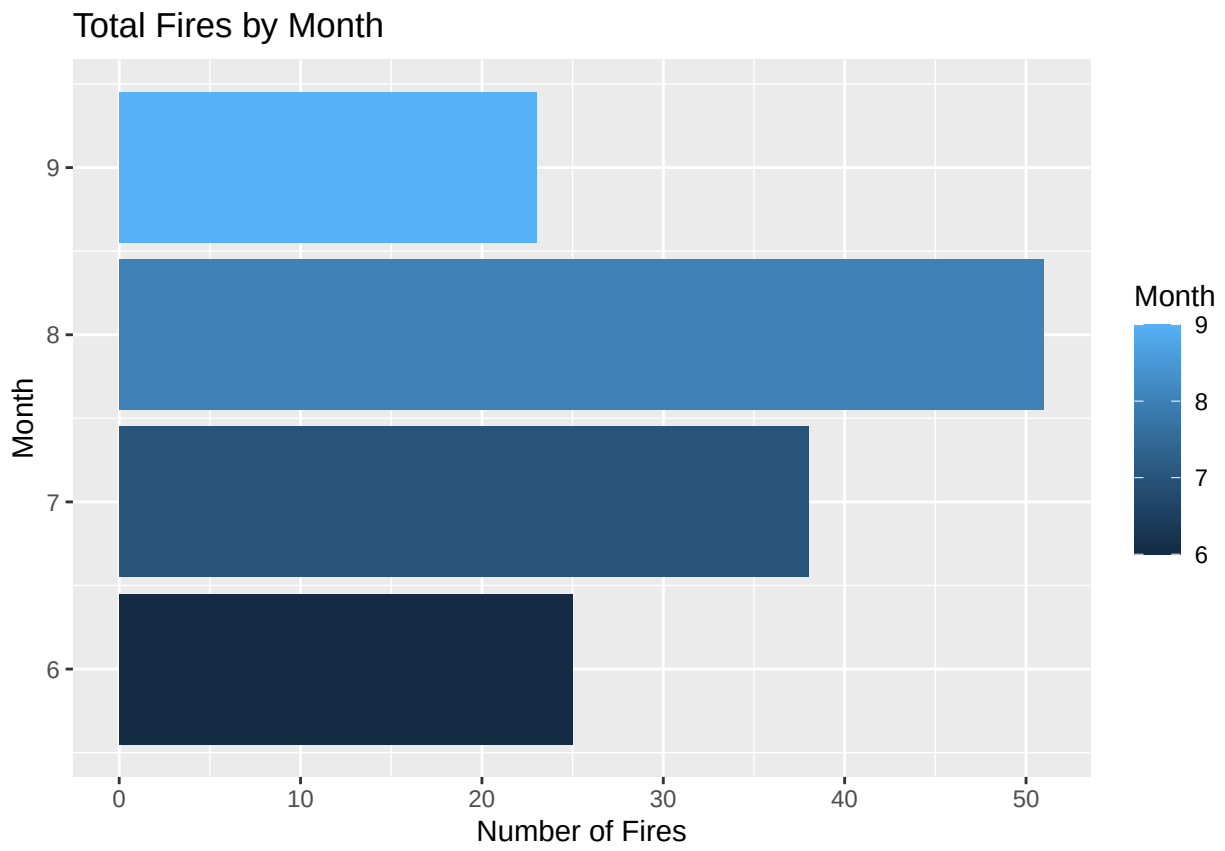
```
fires_formatted %>% group_by(Region) %>% summarize(fires_active = sum(Fire)) %>% ggplot(mapping = aes(x
```



There are more fire in Sidi-Bel Abbes Bejaia.

Lets break that data down into something a little more interesting.

```
fires_formatted %>% group_by(Month) %>% summarize(fires_active = sum(Fire)) %>% ggplot(mapping = aes(x =
```

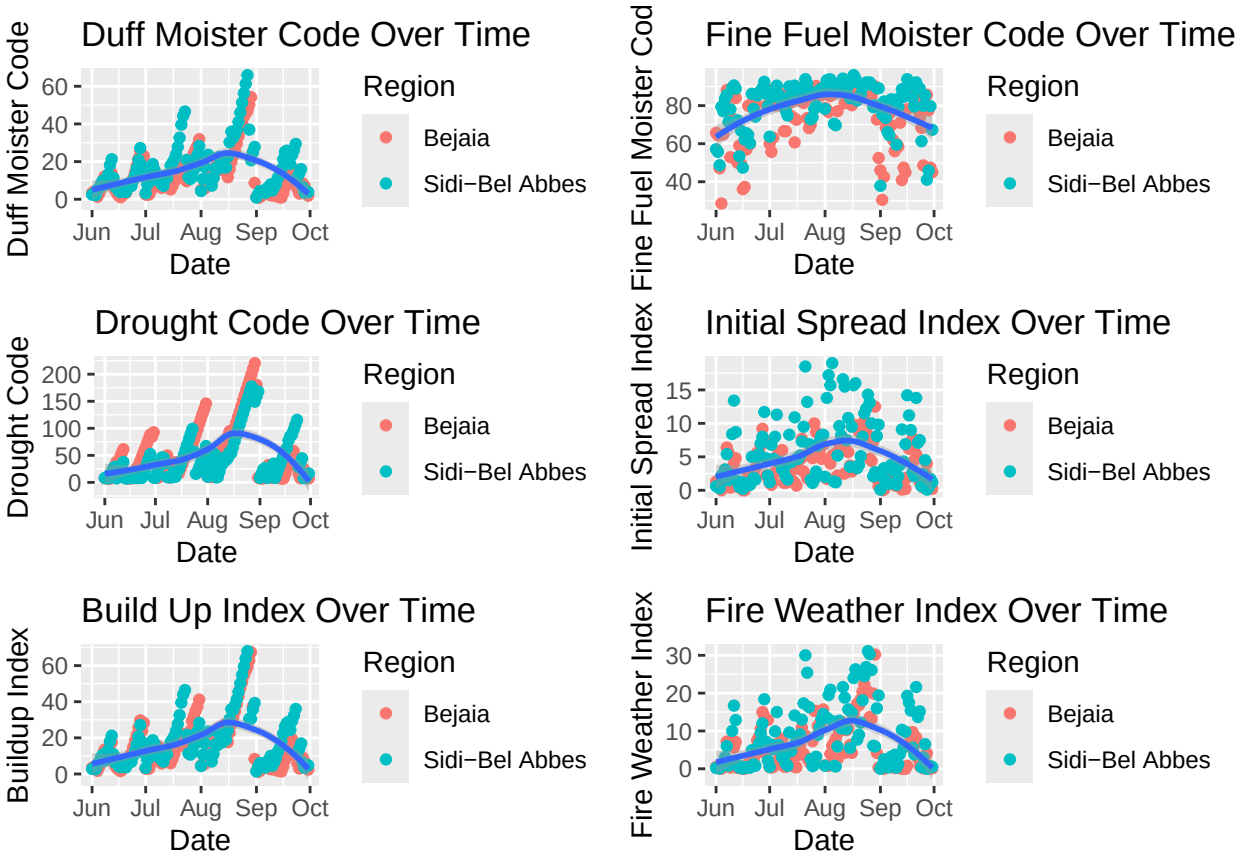


Fires peak in hottest month of the year, August the hottest days of the year. In the future we will likely be able to ignore date an focus on weather.

Now I will do a little visualization of the indexes and codes to see see if they provide any useful insight. We will use color to indicate if a fire happened that day or not. I hope to find some correlation or tends among the data that I may explore later.

```
plot1 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Duff Moister Code`, c
plot2 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Fine Fuel Moister Cod
plot3 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Drought Code`, color =
plot4 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Initial Spread Index`
plot5 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Buildup Index`, color
plot6 <- ggplot(data = fires_formatted) + geom_point(mapping = aes(x = Date, y = `Fire Weather Index`, c
grid.arrange(plot1, plot2, plot3, plot4, plot5, plot6, ncol=2)
```

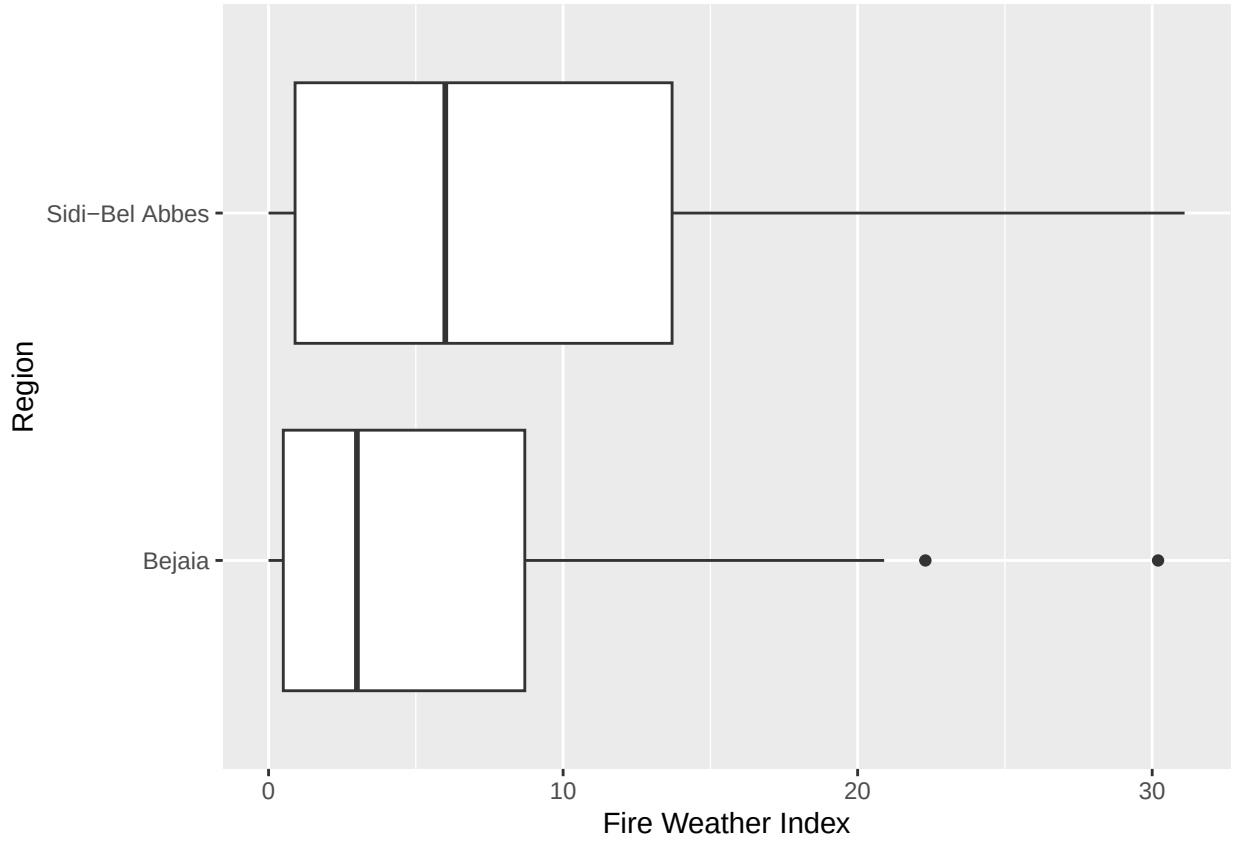
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Most of these charts share the pattern of rising values with a peak in mid August. This follows typical weather patterns of the northern hemisphere with dry with July August and September being the dryest hottest months. The one acceptation to the shape of our plots is the Fine Fuel Moister Code which peaks in early August. Many of the graphs seem to rise steadily and fall when rain occurs. Nothing from these graphs will help solve my questions or raises any more relevant questions.

Lets see if either region have any outlines in the FWI.

```
ggplot(data =  fires_formatted, mapping = aes(x = Region, y = `Fire Weather Index`)) + geom_boxplot() +
```



Average FWI are around 6 and 3 respectively with typically lower ranges being shared at about 1. Upper range and outter quartiles are very different between theses two regions. There are two outliers in the chart, they belong to the Bejaia Region. However they are withing upper bound of the Sida-Bel Abbes region.

Hypotheses Testing

I want to test two things using hypothesis testing. First I want to see if weather conditions when fires start are the same between the two regions. This will tell me if I can treat the two regions as the same or if i should seperate them in further tests. Second I want to see if USA guidelines apply to the regions I am applying my methods. The will assure me that the FWI applies to my region as well.

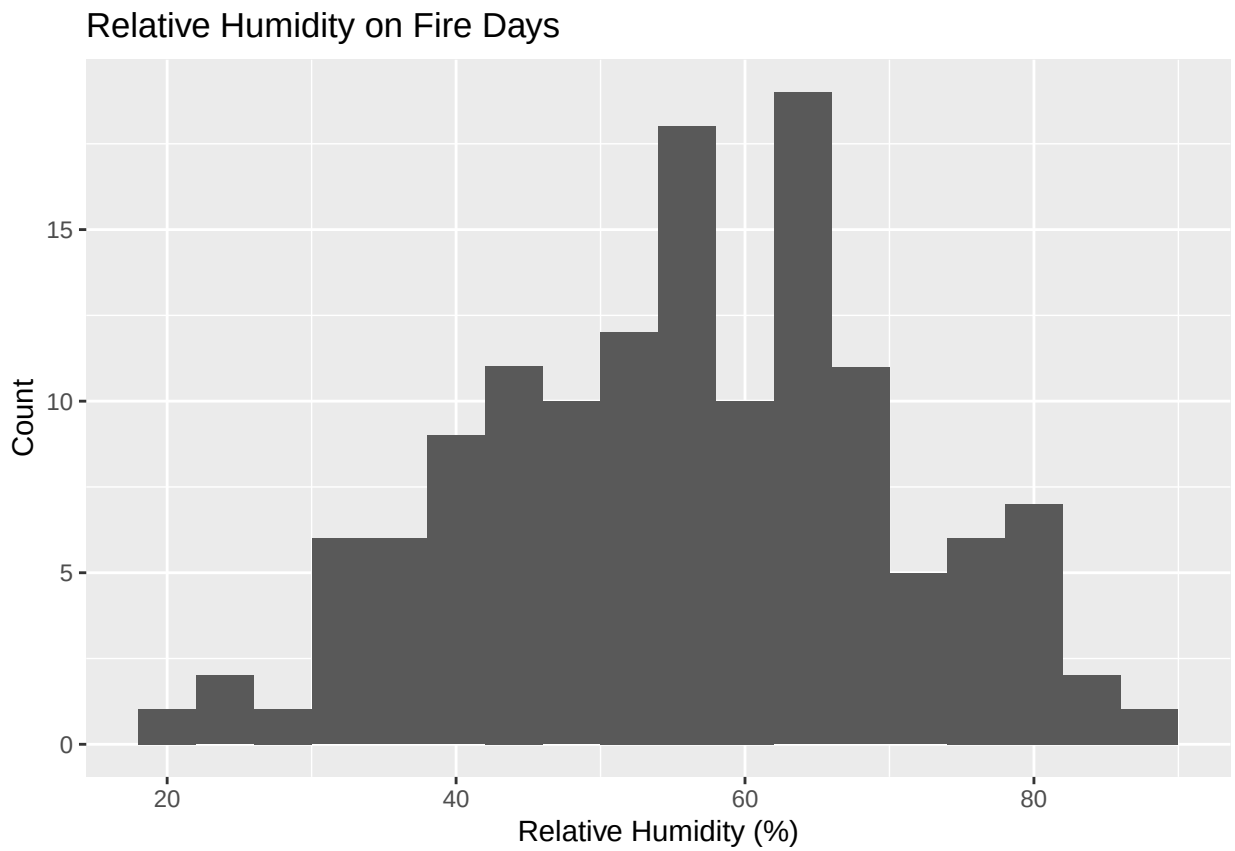
According to the US National Weather Service, the first indicators to the Fire Watch of possible forest fires are a Relative humidity of 25% or less and a temperature of 75F or greater. These indicators suggest that fire conditions do not vary for location or climate. I would like to see if these indicators apply to my regions. I would also like to test if the mean of these key indicators are the same between the two regions on my data set.

First some visualization to ensure that this data has a normal distribution.

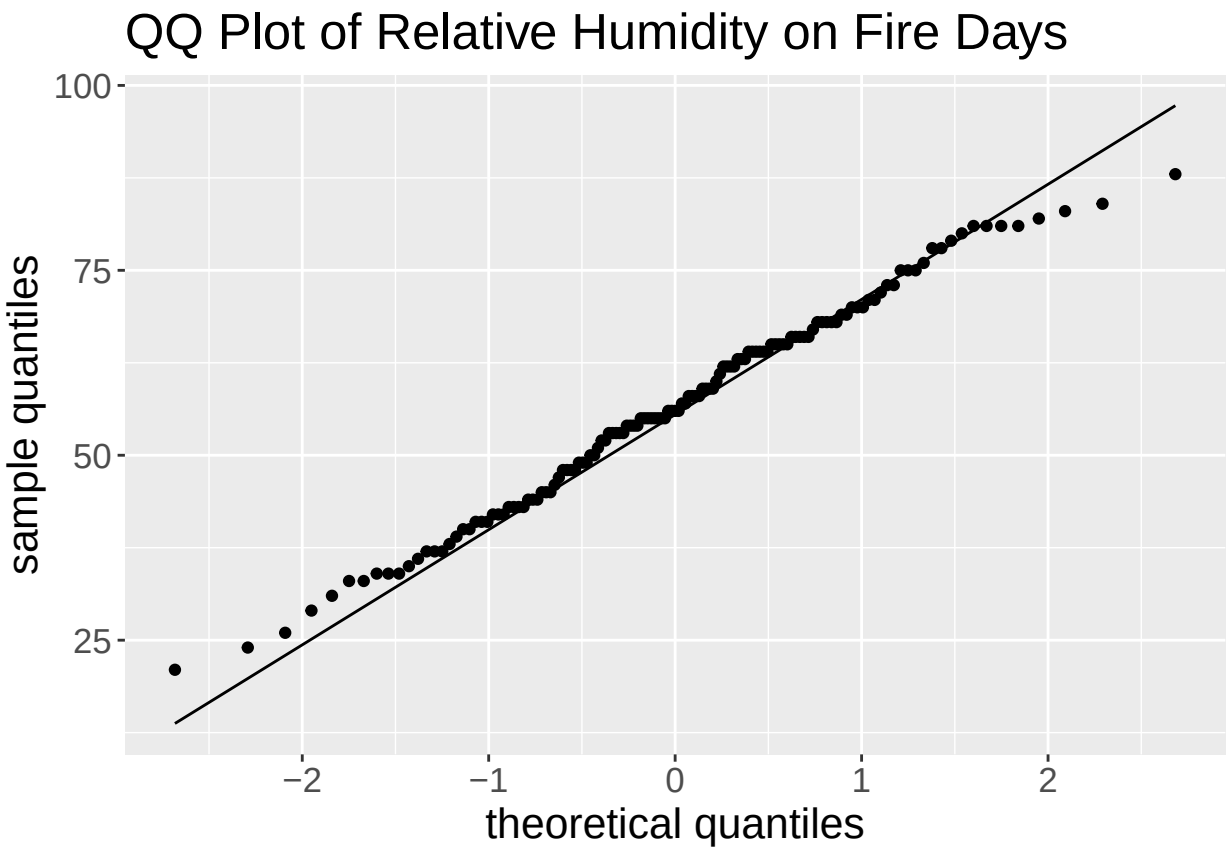
Visualize Relative Humidity

```
#Relative Humidity
fires_temp = fires_formatted[which(fires_formatted$Fire == 1), ]
ggplot(data = fires_temp) + geom_histogram(mapping = aes(x = `Relative Humidity`, groupby = Region), bi
```

```
## Warning in geom_histogram(mapping = aes(x = `Relative Humidity`, groupby =
## Region), : Ignoring unknown aesthetics: groupby
```



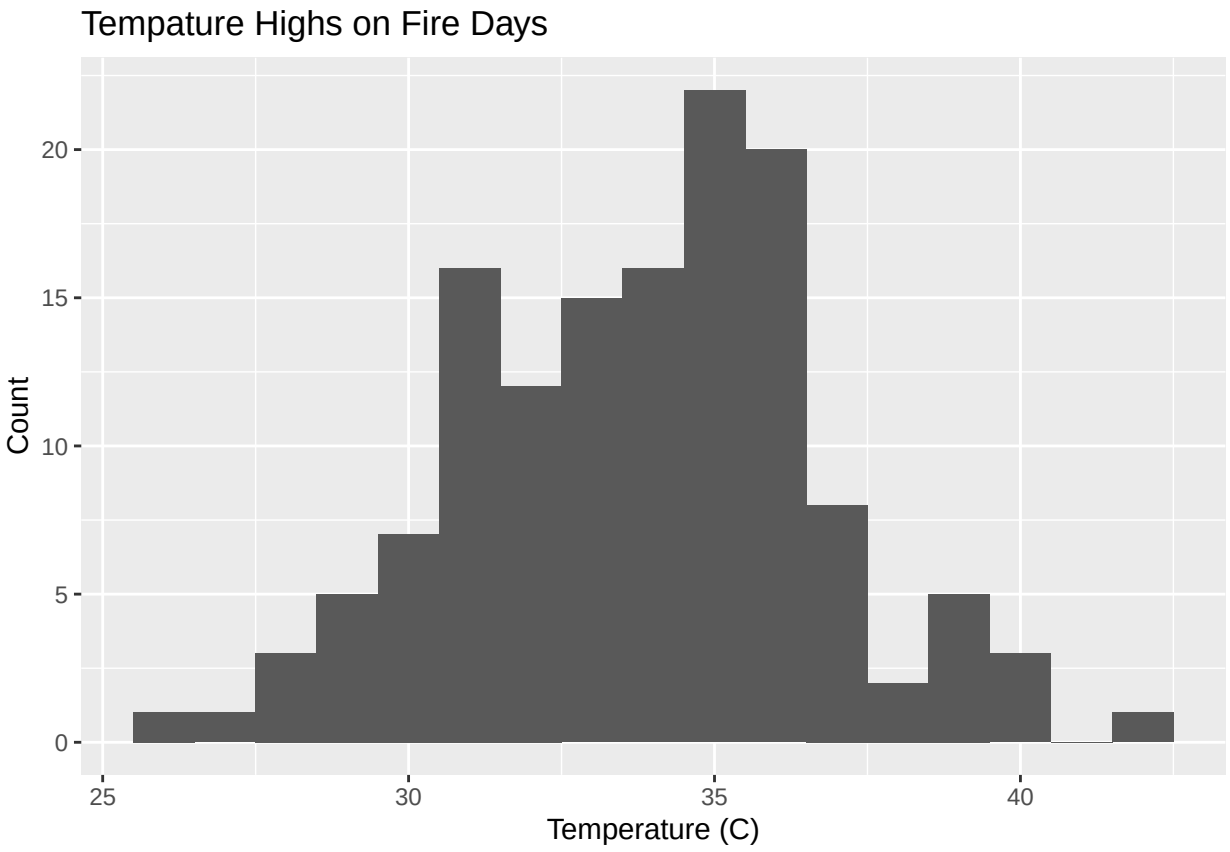

```
fires_rh <- fires_temp$`Relative Humidity` %>% as.data.frame()
fires_rh %>% ggplot(aes(sample = fires_temp$`Relative Humidity`)) + stat_qq(distribution = stats::qnorm
```



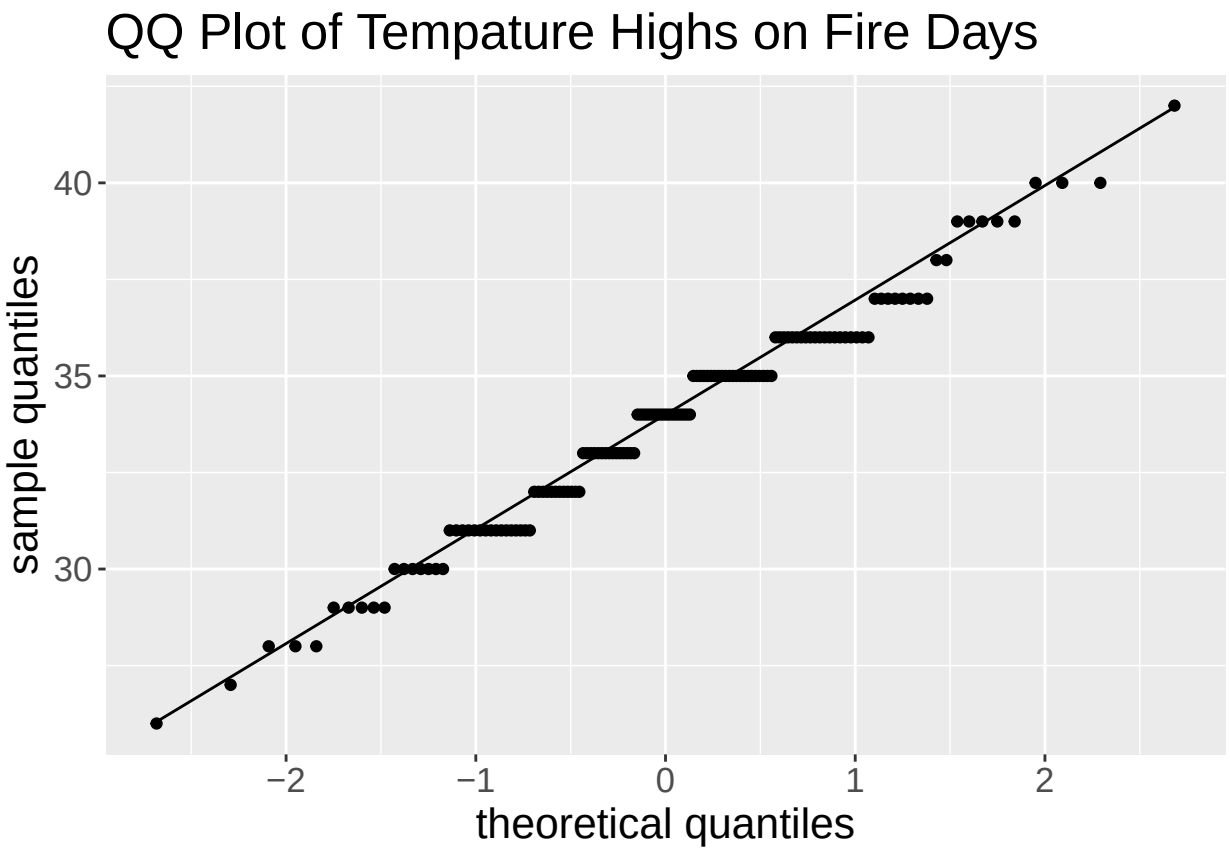
Visualize Temperature

```
#Temperature
ggplot(data = fires_temp) + geom_histogram(mapping = aes(x = `Temperature`, groupby = Region), binwidth
```

```
## Warning in geom_histogram(mapping = aes(x = Temperature, groupby = Region), :
## Ignoring unknown aesthetics: groupby
```




```
fires_temperature <- fires_temp$`Temperature` %>% as.data.frame()
fires_temperature %>% ggplot(aes(sample = fires_temp$`Temperature`)) + stat_qq(distribution = stats::qn
```



Both of the data sets have a normal distribution.

Hypothesis 1: Relative Humidity between Regions

Null hypothesis: The mean relative humidity for all fires is the same in both regions(uA = uB). Alternate hypothesis, the regions have different humidity levels during fires (uA != uB).

I will be using two sample two sided t-testing.

```
Bejaia = subset(fires_formatted, Region == "Bejaia")
Bejaia_fires_RH = Bejaia$`Relative Humidity`[Bejaia$Fire == 1]

SidiBel = subset(fires_formatted, Region == "Sidi-Bel Abbes")
SidiBel_fires_RH = SidiBel$`Relative Humidity`[SidiBel$Fire == 1]

var.test(SidiBel_fires_RH, Bejaia_fires_RH)
```

```
##
##  F test to compare two variances
##
## data:  SidiBel_fires_RH and Bejaia_fires_RH
## F = 2.7721, num df = 77, denom df = 58, p-value = 7.892e-05
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  1.690554 4.469880
## sample estimates:
## ratio of variances
##            2.77213
```

```r
t.test(SidiBel_fires_RH, Bejaia_fires_RH, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  SidiBel_fires_RH and Bejaia_fires_RH
## t = -6.1364, df = 129, p-value = 9.632e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -16.78863  -8.60207
## sample estimates:
## mean of x mean of y
##  50.94872  63.64407
```

We compare the variance of the two samples with var-test and determine the varaince between the two samples is not equal. We use this information in our t.test.

In this two sided two sample t-test the test statistic is -6.136, the reference distribution is 129 and the p-value is 0.00007. the p-value is so low that we reject the null hypothesis for the alternative: The Regions have different mean Relative Humidity when fires start.

Hypothesis 2: Mean Tempature compared to US standards

Now I would like to see if the conditions for temperature suggested by the National Weather (75F, 23.89C) service apply to our regions

Null hypothesis: The US National Weather Service guidelines apply to our region and fires start above 23.89C in Bejaia and Sidi-Bel Abbes (uA >= 23.89). Alternate hypothesis, the US NWS guidlines do not apply to our region, fires tend to start at temperatures lower than 23.89(uA <23.89).

```r
fires_both_regions = fires_formatted$`Temperature`[fires_formatted$Fire == 1]

t.test(fires_both_regions, mu = 23.89, alternative= "l")
```

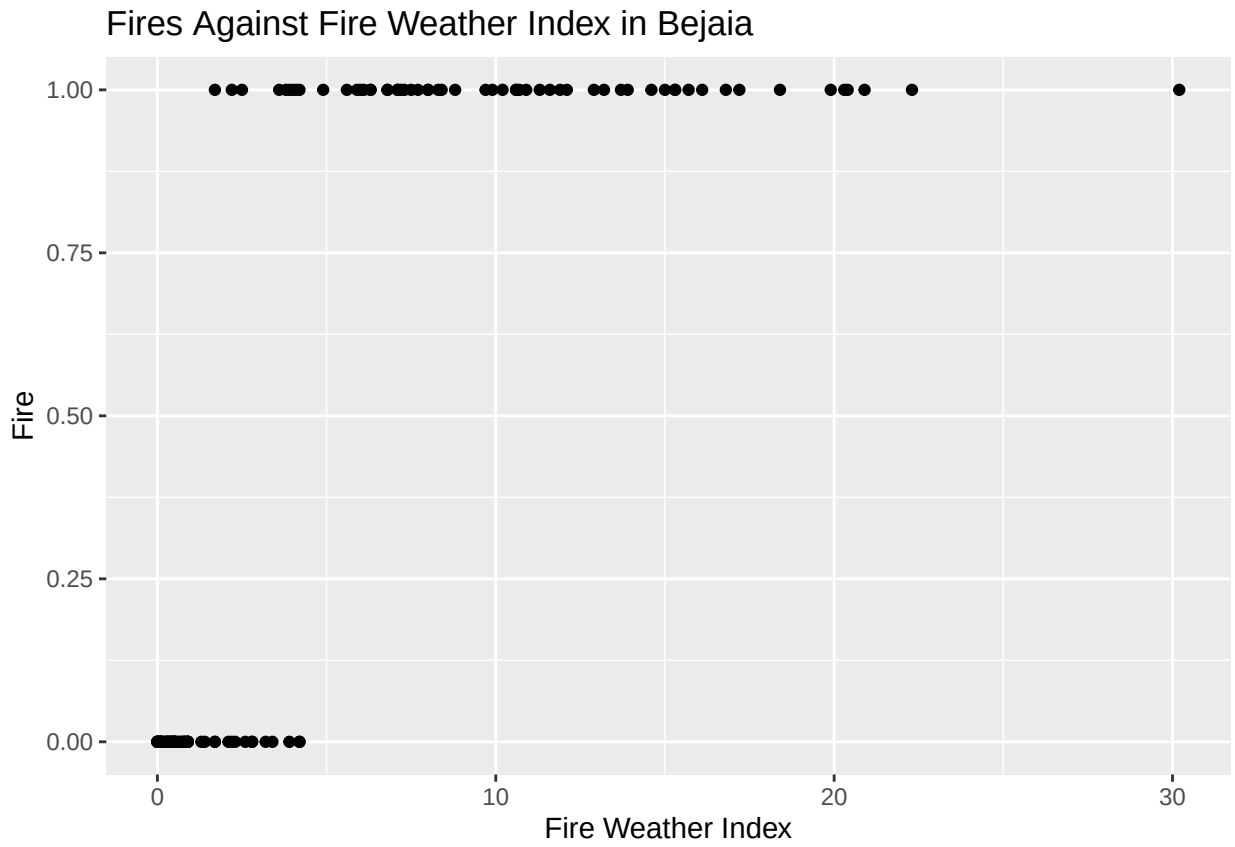
```
##
##  One Sample t-test
##
## data:  fires_both_regions
## t = 39.66, df = 136, p-value = 1
## alternative hypothesis: true mean is less than 23.89
## 95 percent confidence interval:
##      -Inf 34.20926
## sample estimates:
## mean of x
##  33.79562
```

From this one sided test and one sample t-test, the test statistic is 39.66. The reference distribution is 136. Our p-value is very high so we can accept the null hypothesis as true: the US National Weather Service guidelines apply to our region, fires start above 23.89C in our regions.

Logistical Regression

In this section we want to find out if raw data can predict the event of a fire better than the Fire Weather Index. To complete this test we will use regression to predict Fires with raw data and the FWI and compare the r-sqaured values from training. For this study we will begin by testing on the Bejaia region, the more wooded and humid of the two regions.

```
Bejaia %>% ggplot() + geom_point(mapping = aes(x = `Fire Weather Index`, y = Fire)) + ggtitle("Fires Ag
```



In the graph "Fires against Fire Weather Index" we can see values below 2 and above 5 clearly correlate with fires, while values within that index are less predictable. This should make for good results in our regression model.

Lets measure how predictable Fires are with the FWI using r-squared.

```
Bejaia$`Fire Weather Index_scaled` <- scale(Bejaia$`Fire Weather Index`)
fitfwi.logit <- lm(Fire ~ `Fire Weather Index_scaled`, data = Bejaia, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitfwi.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Fire Weather Index_scaled`, data = Bejaia,
##     family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9635 -0.1949 -0.1544  0.2521  0.7495
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  0.48361    0.02966    16.3   <2e-16 ***
```

```
## `Fire Weather Index_scaled`  0.38124    0.02978    12.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3276 on 120 degrees of freedom
## Multiple R-squared:  0.5772, Adjusted R-squared:  0.5737
## F-statistic: 163.8 on 1 and 120 DF,  p-value: < 2.2e-16
```

The FWI scores 57.72% in r-squared.

Now we will conduct a similar analyses on the raw data individually.

```
Bejaia$`Temperature_scaled` <- scale(Bejaia$`Temperature`)
fittemp.logit <- lm(Fire ~ `Temperature_scaled`, data = Bejaia, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fittemp.logit)
```

```
##
## Call:
## lm(formula = Fire ~ Temperature_scaled, data = Bejaia, family = binomial(link = "logit"))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84620 -0.39481  0.06758  0.37950  0.90611
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.48361    0.03956  12.223  < 2e-16 ***
## Temperature_scaled  0.24980    0.03973   6.288 5.42e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.437 on 120 degrees of freedom
## Multiple R-squared:  0.2478, Adjusted R-squared:  0.2415
## F-statistic: 39.54 on 1 and 120 DF,  p-value: 5.416e-09
```

```
Bejaia$`Relative Humidity_scaled` <- scale(Bejaia$`Relative Humidity`)
fitrh.logit <- lm(Fire ~ `Relative Humidity_scaled`, data = Bejaia, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitrh.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Relative Humidity_scaled`, data = Bejaia,
##     family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8736 -0.3474 -0.1267  0.4489  0.7884
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)                 0.48361    0.04225  11.447  < 2e-16 ***
## `Relative Humidity_scaled` -0.18934    0.04242  -4.463 1.83e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4666 on 120 degrees of freedom
## Multiple R-squared:  0.1424, Adjusted R-squared:  0.1352
## F-statistic: 19.92 on 1 and 120 DF,  p-value: 1.833e-05
```

```
Bejaia$`Rain_scaled` <- scale(Bejaia$`Rain`)
fitrain.logit <- lm(Fire ~ `Rain_scaled`, data = Bejaia, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitrain.logit)
```

```
##
## Call:
## lm(formula = Fire ~ Rain_scaled, data = Bejaia, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5423 -0.5127  0.2656  0.4577  0.6278
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.48361    0.04299  11.249  < 2e-16 ***
## Rain_scaled -0.16780    0.04317  -3.887 0.000167 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4749 on 120 degrees of freedom
## Multiple R-squared:  0.1118, Adjusted R-squared:  0.1044
## F-statistic: 15.11 on 1 and 120 DF,  p-value: 0.0001669
```

```
Bejaia$`Wind Speed_scaled` <- scale(Bejaia$`Wind Speed`)
fitwind.logit <- lm(Fire ~ `Wind Speed_scaled`, data = Bejaia, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitwind.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Wind Speed_scaled`, data = Bejaia, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5905 -0.4836 -0.3553  0.4950  0.6233
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.48361    0.04528   10.68   <2e-16 ***
## `Wind Speed_scaled` -0.06092    0.04547   -1.34    0.183
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5002 on 120 degrees of freedom
## Multiple R-squared:  0.01474,    Adjusted R-squared:  0.006529
## F-statistic: 1.795 on 1 and 120 DF,  p-value: 0.1828
```

All of these raw date values score 25% or lower, much lower than the score of the FWI. Now lets combine all of the data points into a single regression model. I hope that by combining these into a regression model that we may improve fire prediction.

```
fitmulti.logit <- lm(Fire ~ `Relative Humidity_scaled` + `Wind Speed_scaled` + `Rain_scaled` + `Tempera
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitmulti.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Relative Humidity_scaled` + `Wind Speed_scaled` +
##     Rain_scaled + Temperature_scaled, data = Bejaia, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8665 -0.3910  0.1415  0.3892  0.8282
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 0.48361    0.03964  12.201  < 2e-16 ***
## `Relative Humidity_scaled` -0.04525    0.05346  -0.846  0.39903
## `Wind Speed_scaled`         0.01678    0.04296   0.391  0.69681
## Rain_scaled                -0.06529    0.04749  -1.375  0.17179
## Temperature_scaled          0.19196    0.05731   3.350  0.00109 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4378 on 117 degrees of freedom
## Multiple R-squared:  0.2639, Adjusted R-squared:  0.2387
## F-statistic: 10.49 on 4 and 117 DF,  p-value: 2.7e-07
```

The accuracy of this model is slightly higher than the individual raw data regression models, however the model with the highest accuracy remains the FWI model, showing us that the FWI is the best way to predict a fire.

I am curious to see if FWI is consistent in it's effectiveness across regions and climates. Lets run the FWI model on Sidi-Bel Abbes data and compare the r-squared scores.

```
SidiBel$`Fire Weather Index_scaled` <- scale(SidiBel$`Fire Weather Index`)
fitSBFWI.logit <- lm(Fire ~ `Fire Weather Index_scaled`, data = SidiBel, family = binomial(link = "logi
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitSBFWI.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Fire Weather Index_scaled`, data = SidiBel,
```

```
##     family = binomial(link = "logit"))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55397 -0.32649 -0.03078  0.31533  0.59704
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  0.64463    0.03200   20.14   <2e-16 ***
## `Fire Weather Index_scaled`  0.32878    0.03214   10.23   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.352 on 119 degrees of freedom
## Multiple R-squared:  0.468,  Adjusted R-squared:  0.4635
## F-statistic: 104.7 on 1 and 119 DF,  p-value: < 2.2e-16
```

The FWI accuracy for Sidi-Bel Abbes is similar Bejaia suggesting FWI has a higher accuracy than raw data regardless of climate.

Lets solidify this theory by running the multi-raw-data models on the Sidi-Bel Abbes regional data.

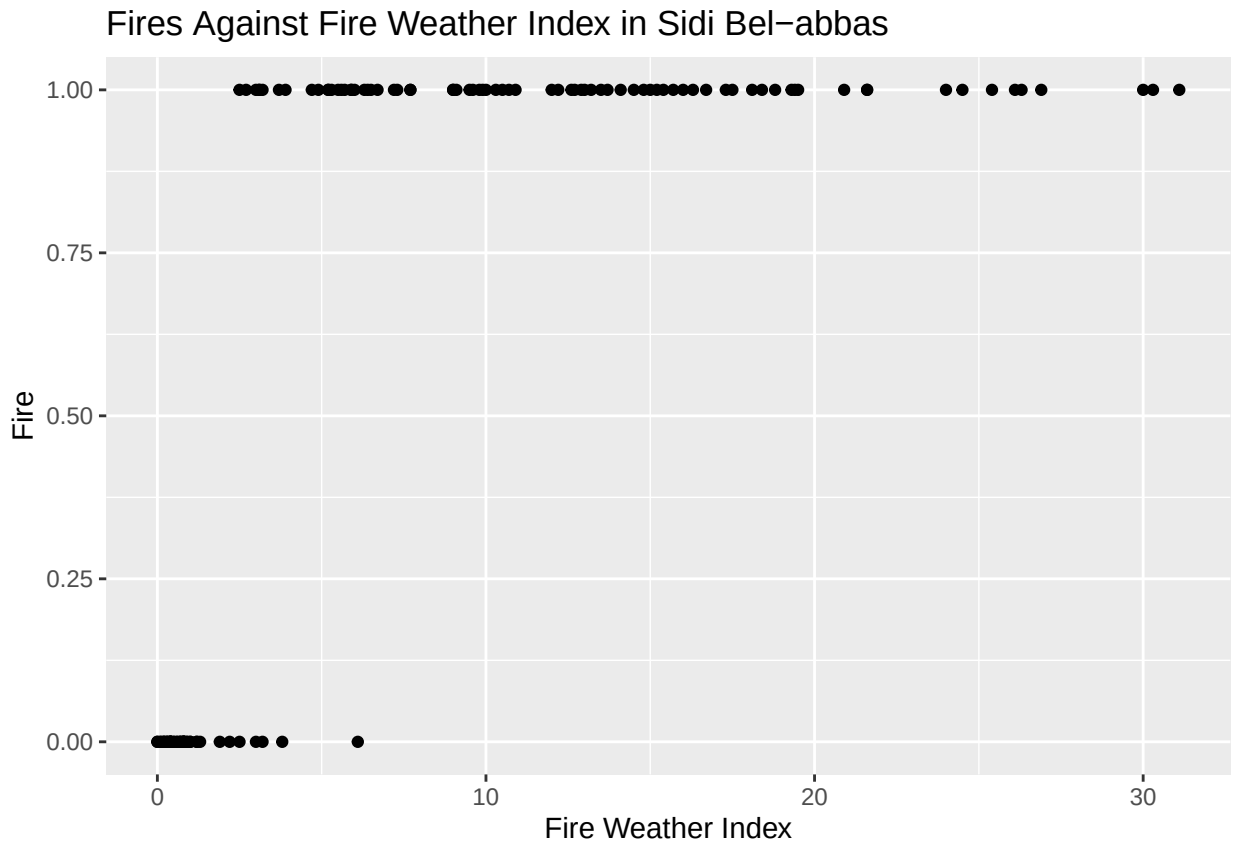
```
SidiBel$`Relative Humidity_scaled` <- scale(SidiBel$`Relative Humidity`)
SidiBel$`Rain_scaled` <- scale(SidiBel$`Rain`)
SidiBel$`Temperature_scaled` <- scale(SidiBel$`Temperature`)
SidiBel$`Wind Speed_scaled` <- scale(SidiBel$`Wind Speed`)
fitSBmulti.logit <- lm(Fire ~ `Relative Humidity_scaled` + `Wind Speed_scaled` + `Rain_scaled` + `Tempe
summary(fitSBmulti.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Relative Humidity_scaled` + `Wind Speed_scaled` +
##     Rain_scaled + Temperature_scaled, data = SidiBel)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.96929 -0.26059  0.07398  0.22205  0.80285
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.64463    0.03282  19.643  < 2e-16 ***
## `Relative Humidity_scaled` -0.08631    0.04113  -2.098   0.0380 *
## `Wind Speed_scaled`          0.05955    0.03480   1.711   0.0897 .
## Rain_scaled                 -0.18415    0.03391  -5.430 3.14e-07 ***
## Temperature_scaled           0.18105    0.04222   4.288 3.75e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.361 on 116 degrees of freedom
## Multiple R-squared:  0.4546, Adjusted R-squared:  0.4358
## F-statistic: 24.18 on 4 and 116 DF,  p-value: 1.461e-14
```

Surprisingly the raw data multi-model preforms nearly as well as FWI. This suggests that regression is a viable option but there are some conditions.

I will plot the fires of this region and calculate the r-squared value for the raw data in this region for more information.

```
SidiBel %>% ggplot() + geom_point(mapping = aes(x = `Fire Weather Index`, y = Fire)) + ggtitle("Fires Ag
```



This plot is very similar to our previous plot, it tells us the FWI is working consistantly in both regions, but does not explain why the multi model regression preforms better in Sidi Bel-abbas.

```
fitSBtemp.logit <- lm(Fire ~ `Temperature_scaled`, data = SidiBel, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitSBtemp.logit)
```

```
##
## Call:
## lm(formula = Fire ~ Temperature_scaled, data = SidiBel, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.0275 -0.3750  0.1030  0.2988  0.7555
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.64463    0.03803  16.951  < 2e-16 ***
## Temperature_scaled  0.23971    0.03819   6.277 5.81e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4183 on 119 degrees of freedom
```

```
## Multiple R-squared:  0.2488, Adjusted R-squared:  0.2424
## F-statistic:  39.4 on 1 and 119 DF,  p-value: 5.809e-09
```

```
fitSBrh.logit <- lm(Fire ~ `Relative Humidity_scaled`, data = SidiBel, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitSBrh.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Relative Humidity_scaled`, data = SidiBel,
##     family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8344 -0.4430  0.1386  0.3411  0.7865
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 0.64463    0.03939  16.366  < 2e-16 ***
## `Relative Humidity_scaled` -0.21171    0.03955  -5.353 4.28e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4333 on 119 degrees of freedom
## Multiple R-squared:  0.194,  Adjusted R-squared:  0.1873
## F-statistic: 28.65 on 1 and 119 DF,  p-value: 4.277e-07
```

```
fitSBrain.logit <- lm(Fire ~ `Rain_scaled`, data = SidiBel, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

```
summary(fitSBrain.logit)
```

```
##
## Call:
## lm(formula = Fire ~ Rain_scaled, data = SidiBel, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7490 -0.4433  0.2510  0.2510  1.1681
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.64463    0.03862  16.692  < 2e-16 ***
## Rain_scaled -0.22810    0.03878  -5.882 3.81e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4248 on 119 degrees of freedom
## Multiple R-squared:  0.2252, Adjusted R-squared:  0.2187
## F-statistic:  34.6 on 1 and 119 DF,  p-value: 3.81e-08
```

```
fitSBwind.logit <- lm(Fire ~ `Wind Speed_scaled`, data = SidiBel, family = binomial(link = "logit"))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'family' will be disregarded
```

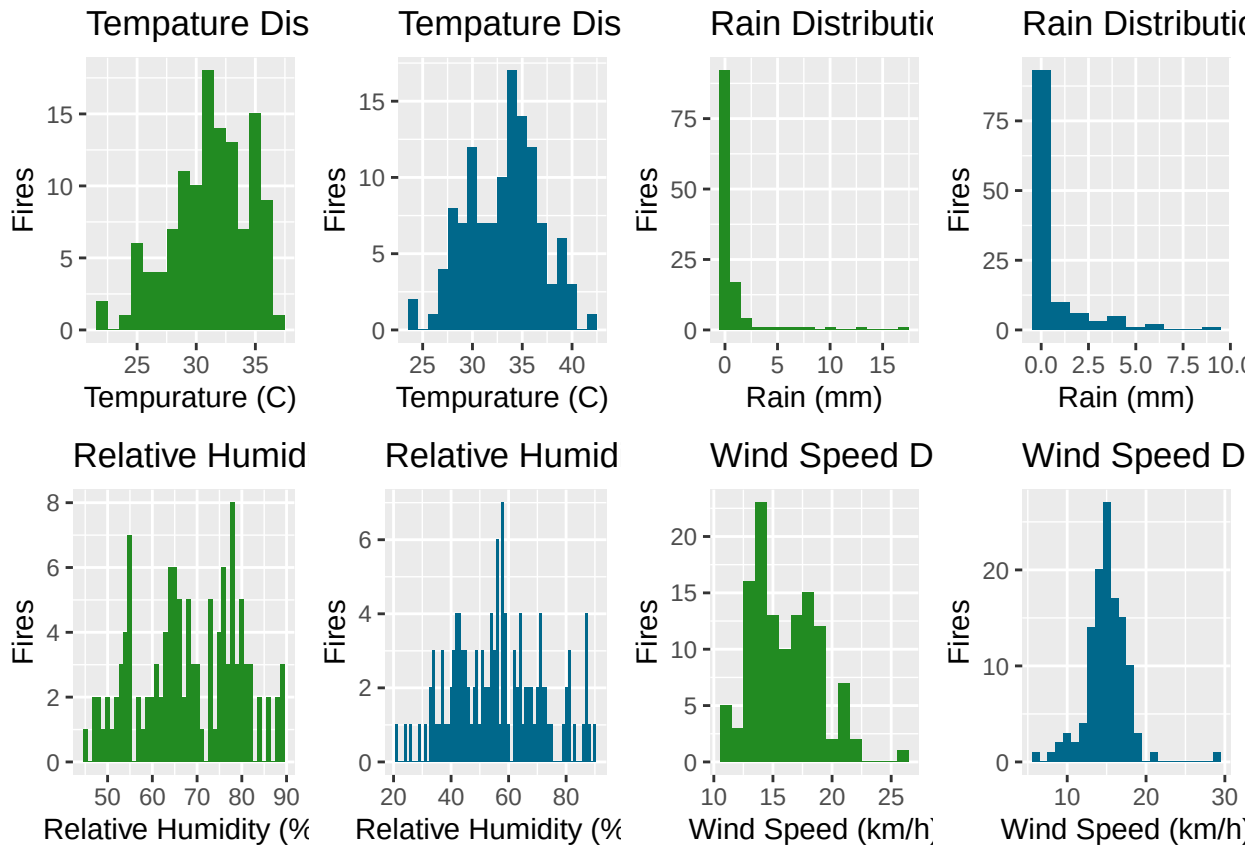
```
summary(fitSBwind.logit)
```

```
##
## Call:
## lm(formula = Fire ~ `Wind Speed_scaled`, data = SidiBel, family = binomial(link = "logit"))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7623 -0.6364  0.3384  0.3552  0.4140
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.64463    0.04383  14.708   <2e-16 ***
## `Wind Speed_scaled`  0.02259    0.04401   0.513    0.609
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4821 on 119 degrees of freedom
## Multiple R-squared:  0.002208,   Adjusted R-squared:  -0.006176
## F-statistic: 0.2634 on 1 and 119 DF,  p-value: 0.6088
```

It seems that the r-squared values of Sidi Bel-abbas raw data are higher individually as well as collectively compared to Bejaia, meaning the region's fires are more predictable with raw data individually and colletivley.

Lets compare the distribution of these data in the two regions to see if there is any difference in distribution that could cause this.

```
plot1 <- ggplot(data = Bejaia) + geom_histogram(mapping = aes(x = Temperature), binwidth = 1, fill = "f
plot2 <- ggplot(data = SidiBel) + geom_histogram(mapping = aes(x = Temperature), binwidth = 1, fill = "

plot3 <- ggplot(data = Bejaia) + geom_histogram(mapping = aes(x = Rain), binwidth = 1, fill = "forestgr
plot4 <- ggplot(data = SidiBel) + geom_histogram(mapping = aes(x = Rain), binwidth = 1, fill = "deepsky

plot5 <- ggplot(data = Bejaia) + geom_histogram(mapping = aes(x = `Relative Humidity`), binwidth = 1, f
plot6 <- ggplot(data = SidiBel) + geom_histogram(mapping = aes(x = `Relative Humidity`), binwidth = 1,

plot7 <- ggplot(data = Bejaia) + geom_histogram(mapping = aes(x = `Wind Speed`), binwidth = 1, fill = "
plot8 <- ggplot(data = SidiBel) + geom_histogram(mapping = aes(x = `Wind Speed`), binwidth = 1, fill =

options(repr.plot.width = 30, repr.plot.height = 10)

grob <- grid.arrange(plot1, plot2, plot3, plot4, plot5, plot6, plot7, plot8, ncol=4, widths = c(10,10,1
```



The only difference is in Wind Speed which has a very small r-square score individually and likely doesn't contribute much to the multi-model regression.

```
ggsave("plots.png", grob, width = 15, height = 8, dpi = 300)
```

Conclusion

In this study I set out to find a new a better way to predict forest fires. Unfortunately I was unable to draw a clear conclusion from this study. While the FWI outperformed my raw data method in Bejaia, the two predictors has similar accuracy in Sidi Bel-abbas suggesting the feasibility of a regression approach may be dependent on the climate. This conclusion is supported by our hypothesis testing that proved fires start under different conditions in different climates. Overall FWI preformed the best in all regions and maintains the best metric for predicting fires in both climates.

However my study was limited and more research needs be done on this topic. My study was limited to a data set consisting of two regions and only used regression for prediction. Following research should include more data from more climate and test the application of Machine Learning methods such as Random Forest Decision Tree and Neural Networks.

Citations

Data set: Abid, . (2019). Algerian Forest Fires [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C5KW4N.

Previous Work: Abid, Faroudja & Izeboudjen, Nouma. (2020). Predicting Forest Fire in Algeria Using Data Mining Techniques: Case Study of the Decision Tree Algorithm. 10.1007/978-3-030-36674-2_37.