615-HW4

Gary Wang

2024-09-25

```
library(data.table)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year

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

```
library(ggplot2)
library(dplyr)
```

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

## The following objects are masked from 'package:data.table':
##
##     between, first, last

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

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

b

```r
buoy_data <- fread("buoy_data.csv")

# convert placeholder values (99, 999, and 9999) to NA for relevant columns

# Define a function to replace them with NA
replace_na <- function(x) {
  x[x %in% c(99, 999, 9999)] <- NA
  return(x)
}

# apply the function to the relevant columns
cols_to_replace <- c("WDIR", "WSPD", "GST", "WVHT", "DPD", "APD", "ATMP",
                     "WTMP", "DEWP", "VIS", "PRES")

buoy_data[, (cols_to_replace) := lapply(.SD, replace_na), .SDcols = cols_to_replace]

head(buoy_data)
```

```
##       YY    MM    DD    hh    WD  WSPD   GST  WVHT   DPD   APD   MWD    BAR
##    <int> <int> <int> <int> <int> <num> <num> <num> <num> <num> <int>  <num>
## 1:    85     1     1     0    60     4     5    NA    NA    NA   999 1030.3
## 2:    85     1     1     1    80     4     5    NA    NA    NA   999 1030.0
## 3:    85     1     1     2   100     4     5    NA    NA    NA   999 1030.1
## 4:    85     1     1     3   100     4     5    NA    NA    NA   999 1029.4
## 5:    85     1     1     4   110     4     5    NA    NA    NA   999 1028.6
## 6:    85     1     1     5    90     4     5    NA    NA    NA   999 1027.8
##     ATMP  WTMP  DEWP   VIS    mm  YYYY  TIDE   #YY  WDIR  PRES
##    <num> <num> <num> <num> <int> <int> <int> <int> <int> <num>
## 1:   4.7   6.7    NA    NA     0    NA    NA    NA    NA    NA
## 2:   5.1   6.7    NA    NA     0    NA    NA    NA    NA    NA
## 3:   5.6   6.6    NA    NA     0    NA    NA    NA    NA    NA
## 4:   5.8   6.7    NA    NA     0    NA    NA    NA    NA    NA
## 5:   5.8   6.7    NA    NA     0    NA    NA    NA    NA    NA
## 6:   5.3   6.7    NA    NA     0    NA    NA    NA    NA    NA
```

```r
summary(buoy_data)
```

```
##        YY               MM               DD              hh
##  Min.   :85.0     Min.   : 1.000   Min.   : 1.00   Min.   : 0.0
##  1st Qu.:88.0     1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 5.0
##  Median :92.0     Median : 7.000   Median :16.00   Median :11.0
##  Mean   :91.5     Mean   : 6.593   Mean   :15.73   Mean   :11.5
##  3rd Qu.:95.0     3rd Qu.:10.000   3rd Qu.:23.00   3rd Qu.:17.0
##  Max.   :98.0     Max.   :12.000   Max.   :31.00   Max.   :23.0
##  NA’s   :346151
##        WD              WSPD            GST             WVHT
##  Min.   :  0.0    Min.   : 0.0    Min.   : 0.00   Min.   :0.00
##  1st Qu.:134.0    1st Qu.: 3.5    1st Qu.: 4.20   1st Qu.:0.41
##  Median :222.0    Median : 5.3    Median : 6.50   Median :0.66
##  Mean   :264.2    Mean   : 5.9    Mean   : 7.29   Mean   :0.87
##  3rd Qu.:297.0    3rd Qu.: 7.9    3rd Qu.: 9.70   3rd Qu.:1.06
##  Max.   :999.0    Max.   :25.7    Max.   :32.40   Max.   :9.10
##  NA’s   :280220   NA’s   :33183   NA’s   :33485   NA’s   :144269
##       DPD              APD              MWD             BAR
##  Min.   : 0.00    Min.   : 0.00    Min.   :  0.0   Min.   : 964.6
##  1st Qu.: 4.55    1st Qu.: 3.85    1st Qu.:232.0   1st Qu.:1010.3
##  Median : 7.69    Median : 4.70    Median :999.0   Median :1015.8
##  Mean   : 7.39    Mean   : 4.96    Mean   :739.7   Mean   :1066.8
##  3rd Qu.:10.00    3rd Qu.: 5.85    3rd Qu.:999.0   3rd Qu.:1021.2
##  Max.   :25.00    Max.   :12.10    Max.   :999.0   Max.   :9999.0
##  NA’s   :147961   NA’s   :144269                   NA’s   :280220
##       ATMP             WTMP            DEWP             VIS
##  Min.   :-19.70   Min.   :-1.80   Min.   :-24.9    Min.   : 0.0
##  1st Qu.:  3.90   1st Qu.: 5.80   1st Qu.: -0.2    1st Qu.: 8.1
##  Median :  9.70   Median :10.50   Median :  7.1    Median : 9.4
##  Mean   :  9.86   Mean   :11.04   Mean   :  6.6    Mean   :12.5
##  3rd Qu.: 16.70   3rd Qu.:16.20   3rd Qu.: 14.7    3rd Qu.:11.6
##  Max.   : 32.10   Max.   :27.80   Max.   : 26.1    Max.   :36.0
##  NA’s   :102761   NA’s   :13186   NA’s   :253613   NA’s   :443062
##        mm             YYYY             TIDE             #YY
##  Min.   : 0.00   Min.   :1999     Min.   :99       Min.   :2007
##  1st Qu.: 0.00   1st Qu.:2001     1st Qu.:99       1st Qu.:2015
##  Median :10.00   Median :2003     Median :99       Median :2021
##  Mean   :20.33   Mean   :2003     Mean   :99       Mean   :2018
##  3rd Qu.:50.00   3rd Qu.:2005     3rd Qu.:99       3rd Qu.:2022
##  Max.   :50.00   Max.   :2006     Max.   :99       Max.   :2023
##                  NA’s   :396370   NA’s   :129610   NA’s   :182081
##       WDIR             PRES
##  Min.   :  0.0    Min.   : 970
##  1st Qu.:131.0    1st Qu.:1011
##  Median :205.0    Median :1016
##  Mean   :197.3    Mean   :1016
##  3rd Qu.:280.0    3rd Qu.:1021
##  Max.   :360.0    Max.   :1046
##  NA’s   :210734   NA’s   :187776
```

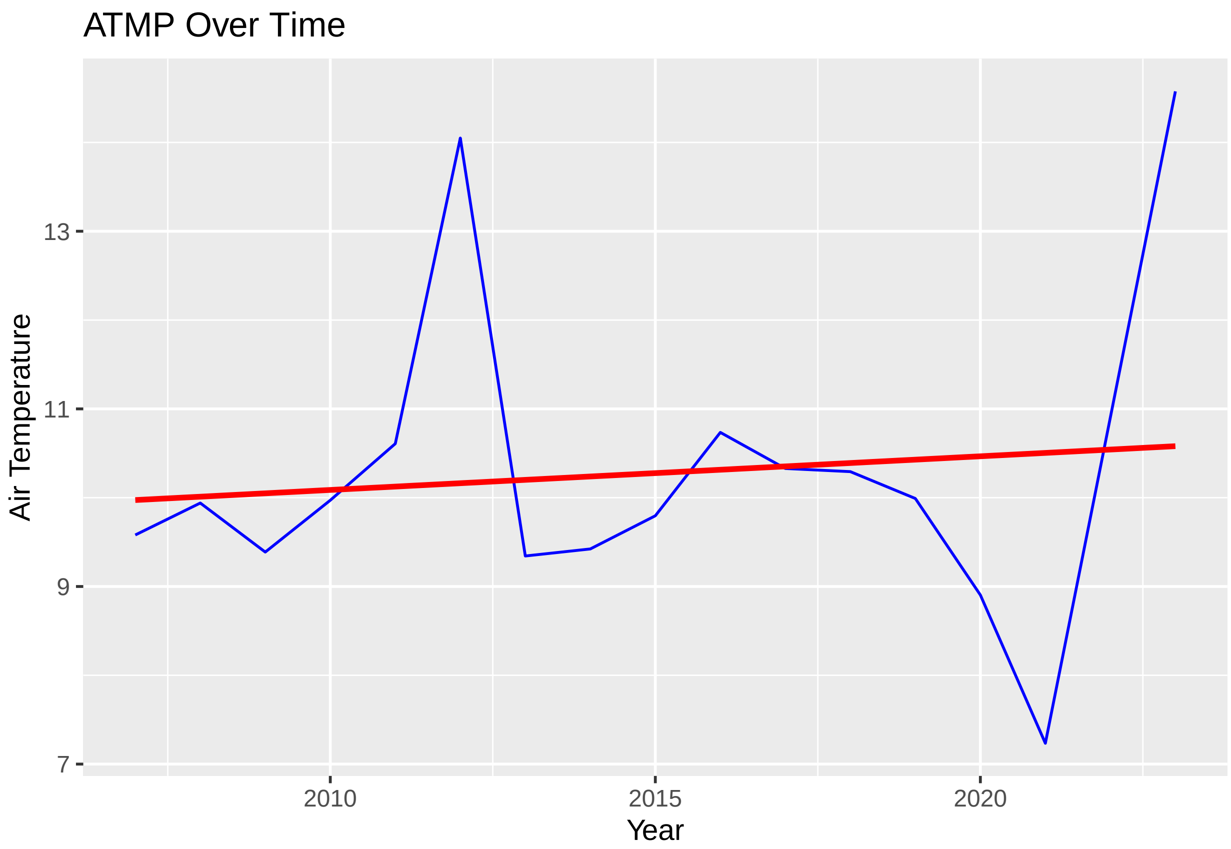
```
# Converting missing/null data to NA is not always a good idea. Because the placeholder values, such as
# The NA values appear to be distributed in a structured way as they clustered around certain variables
```

c

```
# Read in the cleaned dataset
buoy_data <- fread("cleaned_buoy_data.csv")

# filter out rows with NA for key variables
climate_data <- buoy_data %>%
  filter(!is.na(ATMP) & !is.na(WTMP) & !is.na(PRES))

# aggregate yearly averages for key variables
annual_data <- climate_data %>%
  group_by(Year) %>%
  summarize(mean_ATMP = mean(ATMP, na.rm = TRUE),
            mean_WTMP = mean(WTMP, na.rm = TRUE),
            mean_PRES = mean(PRES, na.rm = TRUE))
```

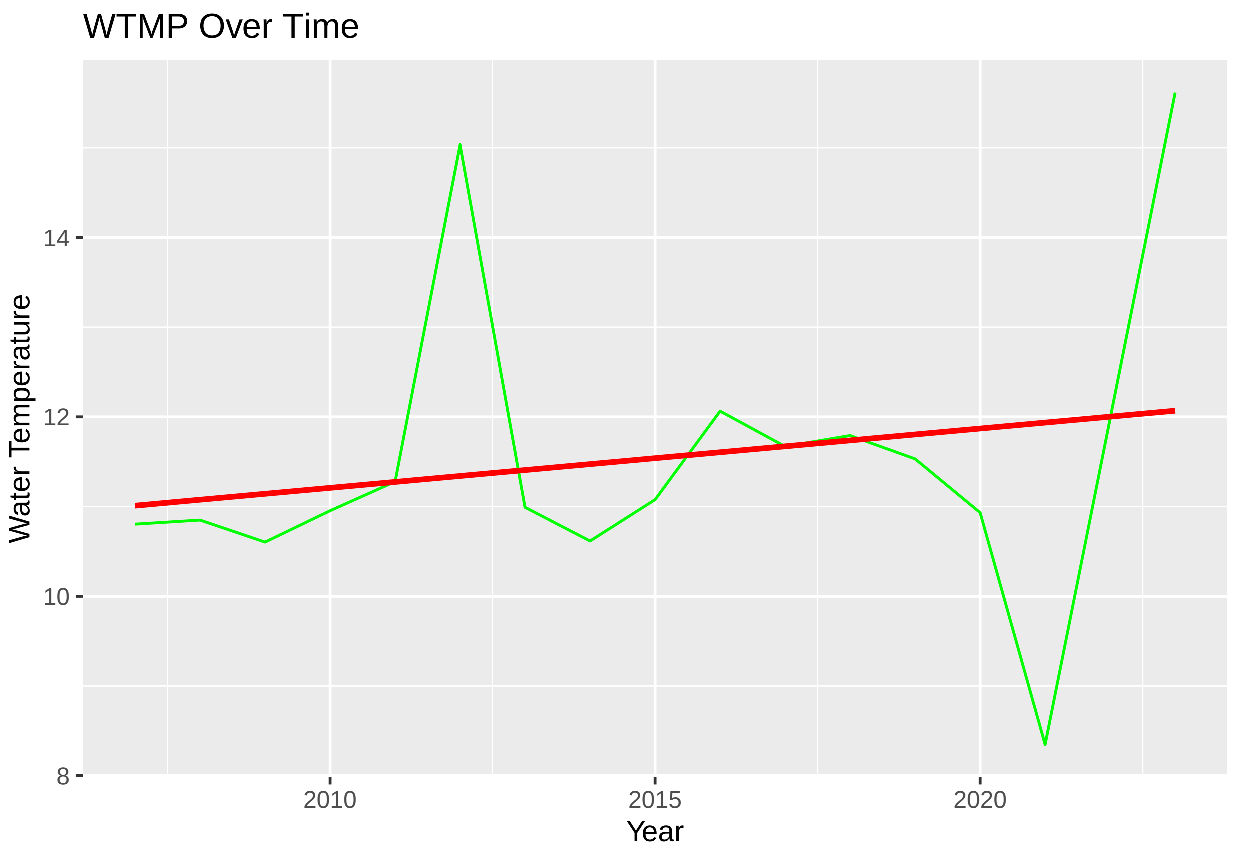
```
# visualize the trends in air temperature, water temperature, and pressure over time

# ATMP
ggplot(annual_data, aes(x = Year, y = mean_ATMP)) +
  geom_line(color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "ATMP Over Time",
       x = "Year", y = "Air Temperature")
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



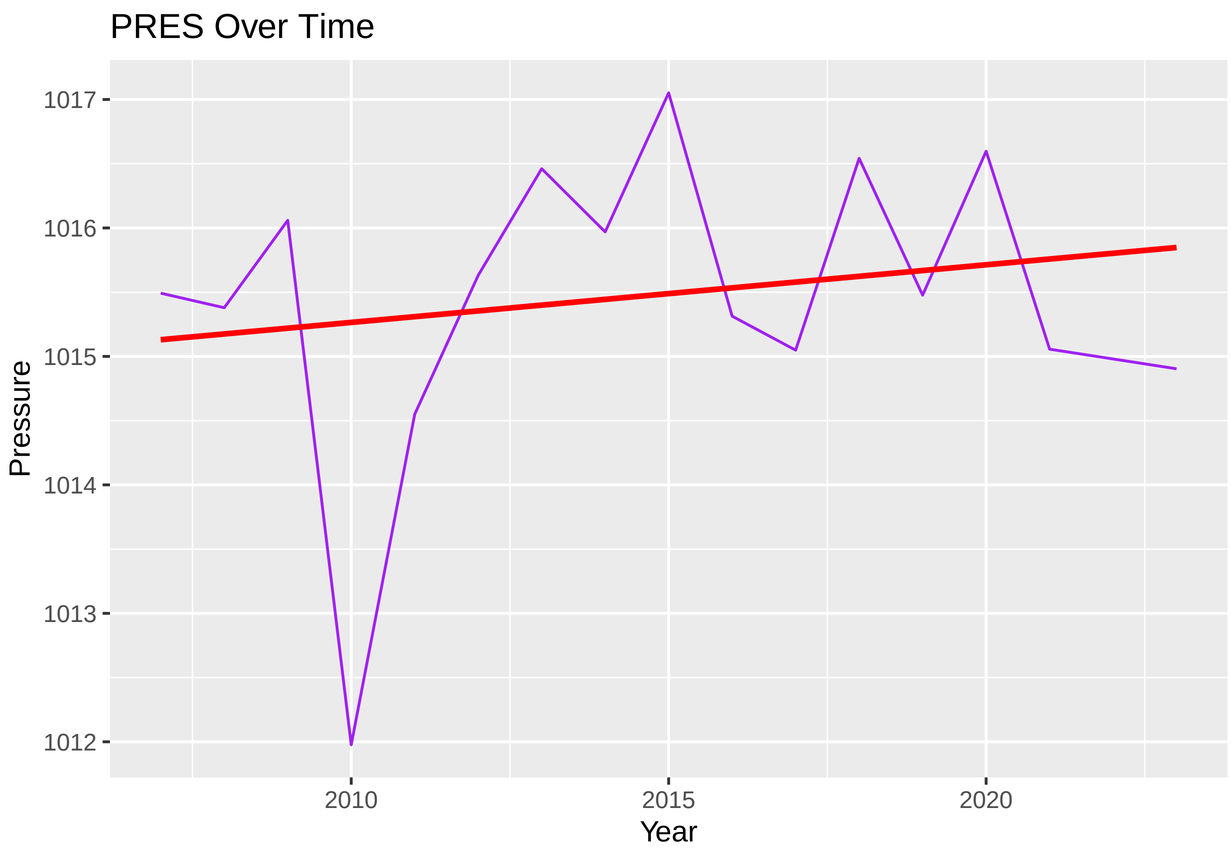
```
# WTMP
ggplot(annual_data, aes(x = Year, y = mean_WTMP)) +
  geom_line(color = "green") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "WTMP Over Time",
       x = "Year", y = "Water Temperature")
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



```
# PRES
ggplot(annual_data, aes(x = Year, y = mean_PRES)) +
  geom_line(color = "purple") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "PRES Over Time",
       x = "Year", y = "Pressure")
```

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

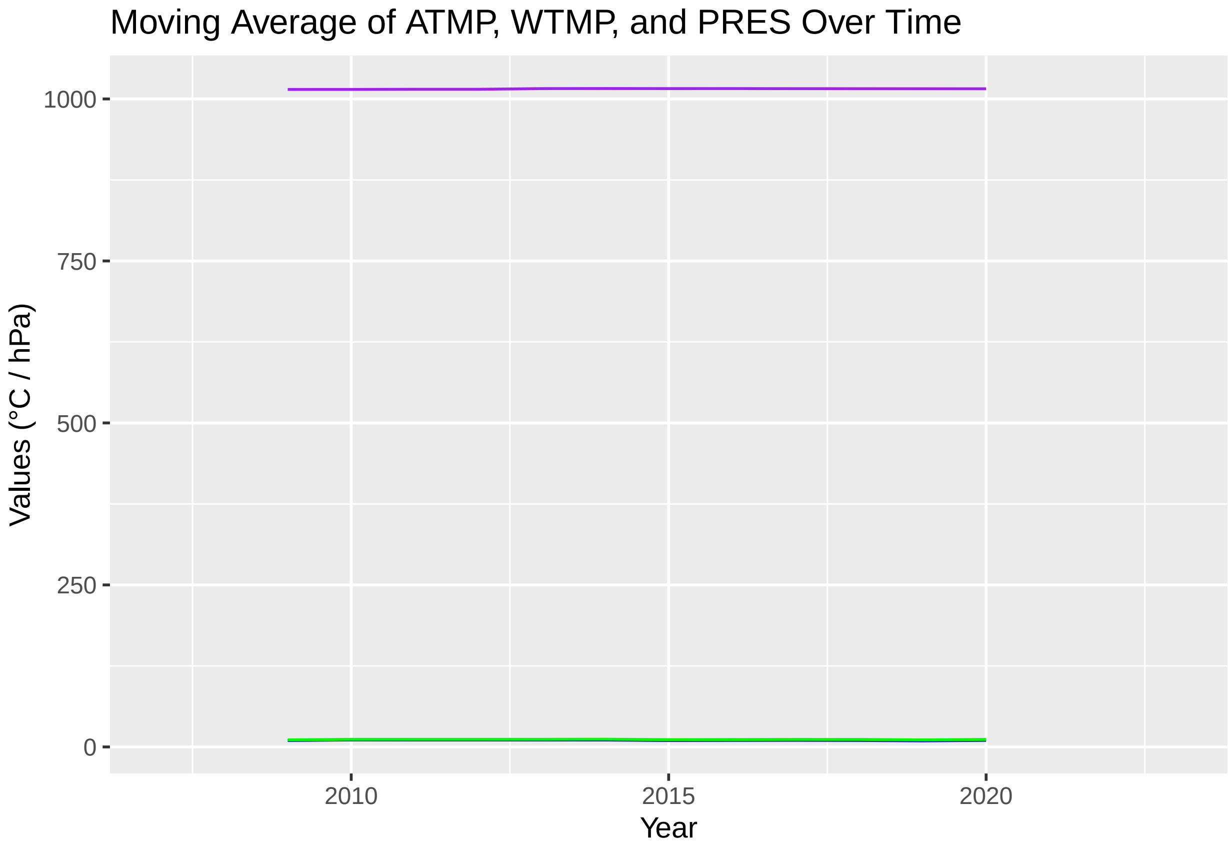


```
# calculate the correlation between air and water temperatures
correlation <- cor(annual_data$mean_ATMP, annual_data$mean_WTMP, use = "complete.obs")
print(correlation)
```

```
## [1] 0.983591
```

```
# moving average to smooth the data
annual_data <- annual_data %>%
  mutate(ATMP_MA = zoo::rollmean(mean_ATMP, k = 5, fill = NA),
         WTMP_MA = zoo::rollmean(mean_WTMP, k = 5, fill = NA),
         PRES_MA = zoo::rollmean(mean_PRES, k = 5, fill = NA))

# plot the moving averages
ggplot(annual_data, aes(x = Year)) +
  geom_line(aes(y = ATMP_MA), color = "blue") +
  geom_line(aes(y = WTMP_MA), color = "green") +
  geom_line(aes(y = PRES_MA), color = "purple") +
  labs(title = "Moving Average of ATMP, WTMP, and PRES Over Time",
       x = "Year", y = "Values (°C / hPa)")
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (‘geom_line()‘).
## Removed 4 rows containing missing values or values outside the scale range
## (‘geom_line()‘).
## Removed 4 rows containing missing values or values outside the scale range
## (‘geom_line()‘).
```



```
# linear regression model for temperature trends
lm_ATMP <- lm(mean_ATMP ~ Year, data = annual_data)
summary(lm_ATMP)
```

```
##
## Call:
## lm(formula = mean_ATMP ~ Year, data = annual_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2699 -0.7000 -0.2546  0.0874  3.9956
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -66.11549  195.76599  -0.338    0.741
## Year          0.03791    0.09718   0.390    0.702
##
## Residual standard error: 1.833 on 14 degrees of freedom
## Multiple R-squared:  0.01076,    Adjusted R-squared:  -0.05991
## F-statistic: 0.1522 on 1 and 14 DF,  p-value: 0.7023
```

```r
lm_WTMP <- lm(mean_WTMP ~ Year, data = annual_data)
summary(lm_WTMP)
```

```
##
## Call:
## lm(formula = mean_WTMP ~ Year, data = annual_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5893 -0.4812 -0.2408  0.0162  3.6980
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -121.72718  184.67271  -0.659    0.520
## Year           0.06614    0.09167   0.721    0.482
##
## Residual standard error: 1.729 on 14 degrees of freedom
## Multiple R-squared:  0.03585,    Adjusted R-squared:  -0.03302
## F-statistic: 0.5205 on 1 and 14 DF,  p-value: 0.4825
```

```r
# check for significant trends
summary(lm_ATMP)
```

```
##
## Call:
## lm(formula = mean_ATMP ~ Year, data = annual_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2699 -0.7000 -0.2546  0.0874  3.9956
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -66.11549  195.76599  -0.338    0.741
## Year          0.03791    0.09718   0.390    0.702
##
## Residual standard error: 1.833 on 14 degrees of freedom
## Multiple R-squared:  0.01076,    Adjusted R-squared:  -0.05991
## F-statistic: 0.1522 on 1 and 14 DF,  p-value: 0.7023
```

```r
summary(lm_WTMP)
```

```
##
## Call:
## lm(formula = mean_WTMP ~ Year, data = annual_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5893 -0.4812 -0.2408  0.0162  3.6980
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -121.72718  184.67271  -0.659    0.520
## Year           0.06614    0.09167   0.721    0.482
##
## Residual standard error: 1.729 on 14 degrees of freedom
## Multiple R-squared:  0.03585,    Adjusted R-squared:  -0.03302
## F-statistic: 0.5205 on 1 and 14 DF,  p-value: 0.4825
```

d

```
rainfall_data <- fread("Rainfall.csv")
head(rainfall_data)
```

```
##        STATION                            STATION_NAME           DATE  HPCP
##         <char>                                  <char>         <char> <num>
## 1: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 01:00  0.00
## 2: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 09:00  0.01
## 3: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 10:00  0.01
## 4: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 11:00  0.01
## 5: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 12:00  0.01
## 6: COOP:190770 BOSTON LOGAN INTERNATIONAL AIRPORT MA US 19850101 13:00  0.01
##    Measurement Flag Quality Flag
##              <char>       <lgcl>
## 1:                g           NA
## 2:                            NA
## 3:                            NA
## 4:                            NA
## 5:                            NA
## 6:                            NA
```

```
summary(rainfall_data)
```

```
##    STATION           STATION_NAME           DATE                HPCP
##  Length:31714       Length:31714       Length:31714       Min.   :0.00000
##  Class :character   Class :character   Class :character   1st Qu.:0.00000
##  Mode  :character   Mode  :character   Mode  :character   Median :0.01000
##                                                           Mean   :0.03875
##                                                           3rd Qu.:0.04000
##                                                           Max.   :2.03000
##  Measurement Flag   Quality Flag
##  Length:31714       Mode:logical
##  Class :character   NA’s:31714
##  Mode  :character
##
##
##
```

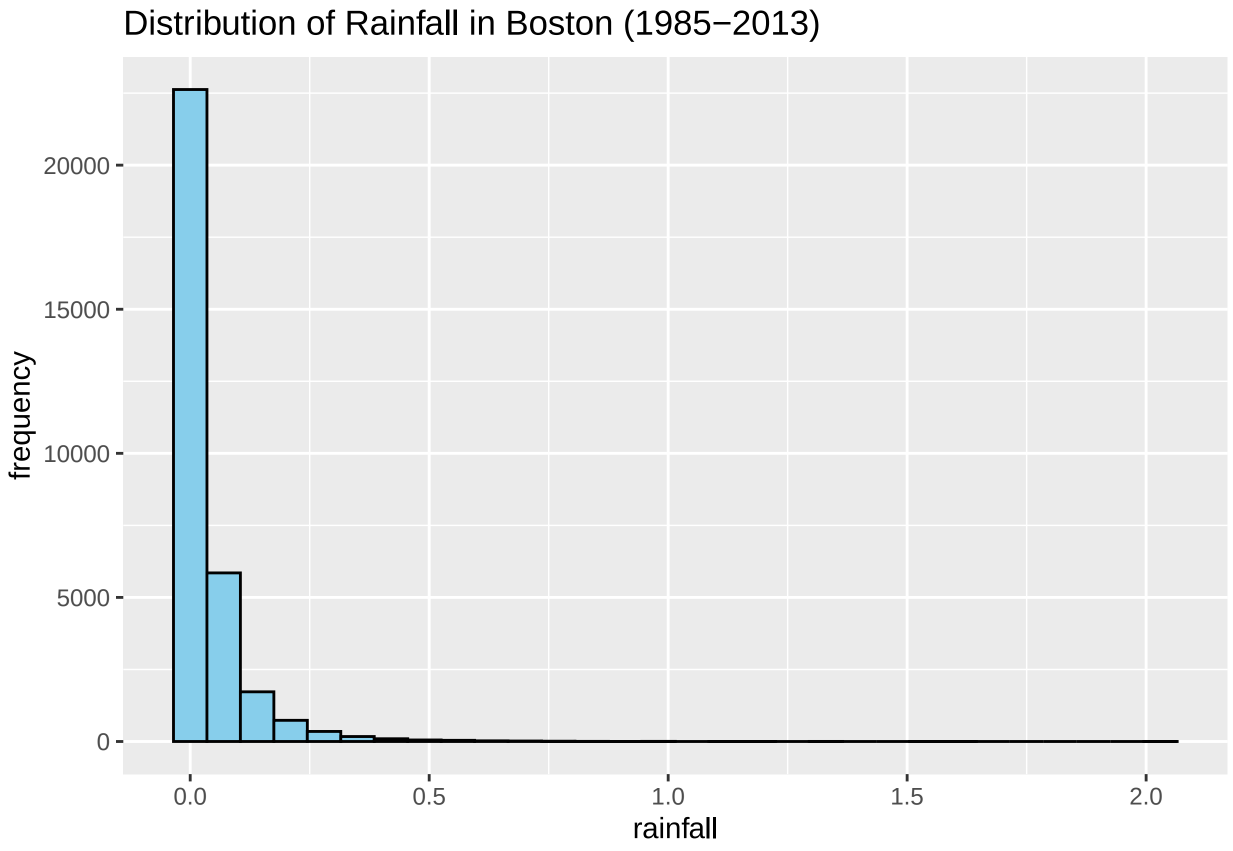
```
# check for missing values
colSums(is.na(rainfall_data))
```

```
##          STATION     STATION_NAME             DATE             HPCP
##                0                0                0                0
## Measurement Flag     Quality Flag
##                0            31714
```

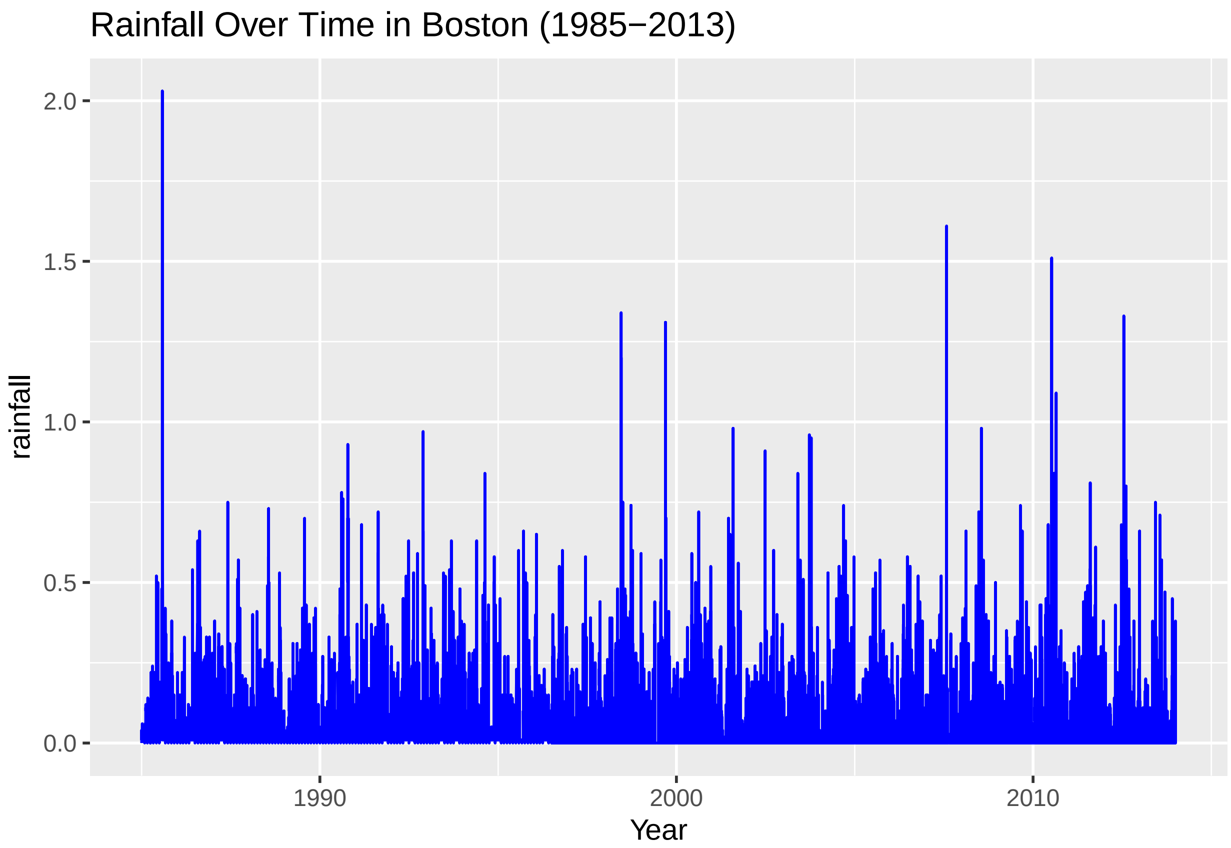
```r
# convert date to date-time format
rainfall_data$Date <- as.POSIXct(rainfall_data$DATE, format = "%Y%m%d %H:%M", tz = "UTC")

# calculate summary statistics for rainfall
rainfall_stats <- rainfall_data %>%
  summarise(
    mean_rainfall = mean(HPCP, na.rm = TRUE),
    median_rainfall = median(HPCP, na.rm = TRUE),
    max_rainfall = max(HPCP, na.rm = TRUE),
    min_rainfall = min(HPCP, na.rm = TRUE)
  )

print(rainfall_stats)
```

```
##   mean_rainfall median_rainfall max_rainfall min_rainfall
## 1     0.0387485            0.01         2.03            0
```

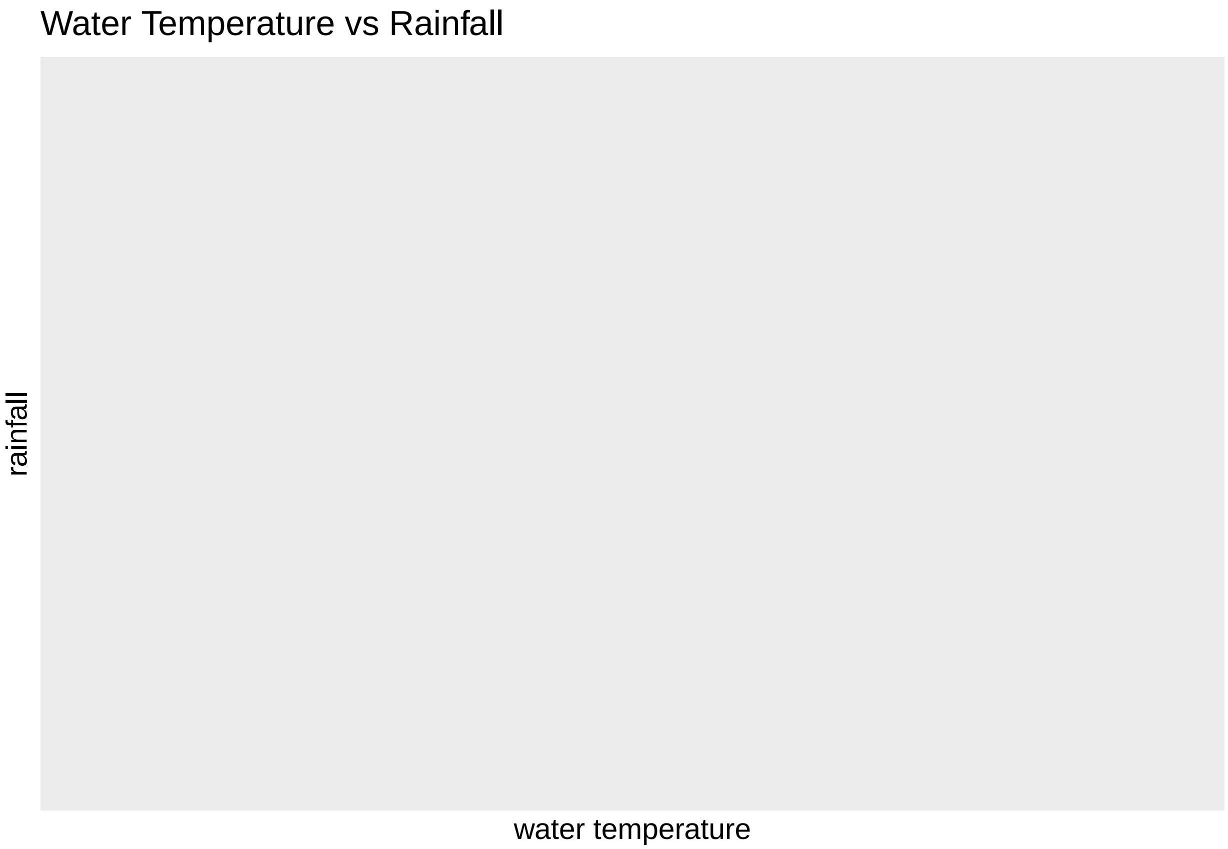
```r
# plot rainfall distribution
ggplot(rainfall_data, aes(x = HPCP)) +
  geom_histogram(bins = 30, fill = "skyblue", color = "black") +
  labs(title = "Distribution of Rainfall in Boston (1985-2013)",
       x = "rainfall", y = "frequency")
```



```r
# plot time series of rainfall
ggplot(rainfall_data, aes(x = Date, y = HPCP)) +
  geom_line(color = "blue") +
  labs(title = "Rainfall Over Time in Boston (1985-2013)",
       x = "Year", y = "rainfall")
```



```r
# merge datasets by date
rainfall_buoy <- merge(rainfall_data, buoy_data, by.x = "Date", by.y = "Year")

# explore relationships between rainfall (HPCP) and buoy readings
ggplot(rainfall_buoy, aes(x = WTMP, y = HPCP)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Water Temperature vs Rainfall",
         x = "water temperature", y = "rainfall")
```



```
# In my analysis of Boston's rainfall data from 1985 to 2013, I found that rainfall is heavily skewed t
```