615 HW4

Haoran Cui

2024-09-26

##a

```
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(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
```

```
file_root <- "https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h"
tail <- ".txt.gz&dir=data/historical/stdmet/"

years <- 1985:2023

buoy_data_list <- list()

for (year in years) {
  path <- paste0(file_root, year, tail)
  header <- scan(path, what = 'character', nlines = 1)
  skip_lines <- ifelse(year >= 2007, 2, 1)
  buoy <- fread(path, header = FALSE, skip = skip_lines)
  num_cols <- ncol(buoy)
```

```
  if (length(header) > num_cols) {
    header <- header[1:num_cols]
  } else if (length(header) < num_cols) {
    header <- c(header, paste0("V", (length(header) + 1):num_cols))
  }

  colnames(buoy) <- header

  if ("YY" %in% colnames(buoy) & "MM" %in% colnames(buoy) & "DD" %in% colnames(buoy) & "hh" %in% colnam
    buoy$Date <- ymd_hms(paste(buoy$YY, buoy$MM, buoy$DD, buoy$hh, buoy$mm))
  }

  buoy_data_list[[as.character(year)]] <- buoy
}
```

```
## Warning in fread(path, header = FALSE, skip = skip_lines): Stopped early on
## line 5114. Expected 16 fields but found 17. Consider fill=TRUE and
## comment.char=. First discarded non-empty line: <<2000 08 01 00 78 4.3 5.1 0.58
## 8.33 5.36 999 1022.9 17.3 17.5 15.0 99.0 99.00>>
```

```
buoy_data_list <- rbindlist(buoy_data_list, fill = TRUE)

# Merge the YY, YYYY, and #YY columns into one (if any of them exist)
buoy_data_list <- buoy_data_list %>%
  mutate(Year = coalesce(as.numeric(YY), as.numeric(YYYY), as.numeric(`#YY`))) %>%
  select(-YY, -YYYY, -`#YY`) %>%
  select(Year, everything())

# Adjust year format for years between 85 and 98
buoy_data_list <- buoy_data_list %>%
  mutate(Year = ifelse(Year < 100, ifelse(Year >= 85, 1900 + Year, 2000 + Year), Year))

# Merge WD and WDIR columns into one
buoy_data_list <- buoy_data_list %>%
  mutate(Wind_Direction = coalesce(WD, WDIR)) %>%
  select(-WD, -WDIR)

# Merge BAR and PRES columns into one
buoy_data_list <- buoy_data_list %>%
  mutate(Pressure = coalesce(BAR, PRES)) %>%
  select(-BAR, -PRES)

# Create a proper date column if possible
if (all(c("Year", "MM", "DD", "hh") %in% colnames(buoy_data_list))) {
  buoy_data_list[, date := make_datetime(Year, MM, DD, hh)]
} else if (all(c("Year", "MM", "DD", "hh") %in% colnames(buoy_data_list))) {
  buoy_data_list[, date := make_datetime(Year, MM, DD, hh)]
}

buoy_data_list <- buoy_data_list %>% select(date, everything())

str(buoy_data_list)
```

```
## Classes 'data.table' and 'data.frame':   462301 obs. of  19 variables:
```

```
##  $ date          : POSIXct, format: "1985-01-01 00:00:00" "1985-01-01 01:00:00" ...
##  $ Year          : num  1985 1985 1985 1985 1985 ...
##  $ MM            : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ DD            : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ hh            : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ WSPD          : num  4 4 4 4 4 4 4 4 6 7 ...
##  $ GST           : num  5 5 5 5 5 5 6 5 6 8 ...
##  $ WVHT          : num  99 99 99 99 99 99 99 99 99 99 ...
##  $ DPD           : num  99 99 99 99 99 99 99 99 99 99 ...
##  $ APD           : num  99 99 99 99 99 99 99 99 99 99 ...
##  $ MWD           : int  999 999 999 999 999 999 999 999 999 999 ...
##  $ ATMP          : num  4.7 5.1 5.6 5.8 5.8 5.3 5.5 5.8 5.9 6.2 ...
##  $ WTMP          : num  6.7 6.7 6.6 6.7 6.7 6.7 6.7 6.7 6.7 6.7 ...
##  $ DEWP          : num  999 999 999 999 999 999 999 999 999 999 ...
##  $ VIS           : num  99 99 99 99 99 99 99 99 99 99 ...
##  $ TIDE          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ mm            : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ Wind_Direction: int  60 80 100 100 110 90 60 30 40 40 ...
##  $ Pressure      : num  1030 1030 1030 1029 1029 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

##b

```
library(tidyverse)
```

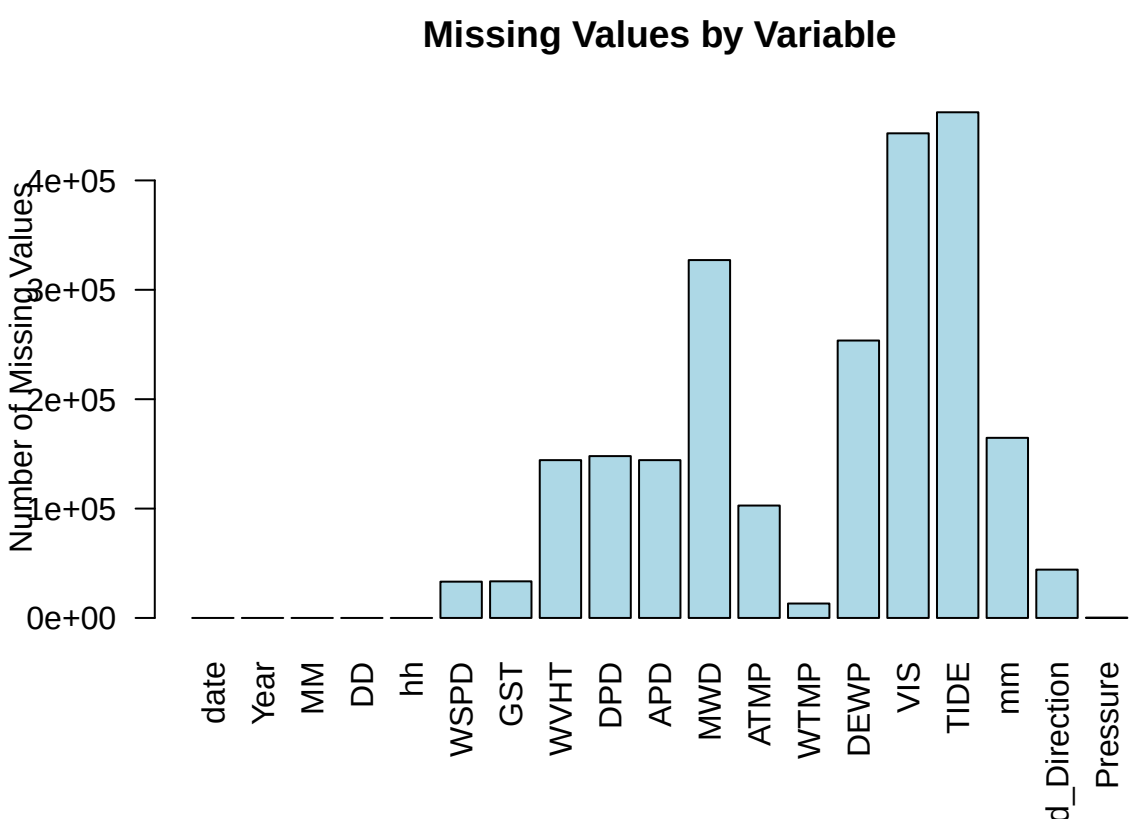
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats 1.0.0     v stringr 1.5.0
## v ggplot2 3.4.4     v tibble  3.2.1
## v purrr   1.0.2     v tidyr   1.3.0
## v readr   2.1.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::between()     masks data.table::between()
## x dplyr::filter()      masks stats::filter()
## x dplyr::first()       masks data.table::first()
## x lubridate::hour()    masks data.table::hour()
## x lubridate::isoweek() masks data.table::isoweek()
## x dplyr::lag()         masks stats::lag()
## x dplyr::last()        masks data.table::last()
## x lubridate::mday()    masks data.table::mday()
## x lubridate::minute()  masks data.table::minute()
## x lubridate::month()   masks data.table::month()
## x lubridate::quarter() masks data.table::quarter()
## x lubridate::second()  masks data.table::second()
## x purrr::transpose()   masks data.table::transpose()
## x lubridate::wday()    masks data.table::wday()
## x lubridate::week()    masks data.table::week()
## x lubridate::yday()    masks data.table::yday()
## x lubridate::year()    masks data.table::year()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
# Convert 999 values to NA for WDIR and other relevant columns
buoy_data_list <- buoy_data_list %>%
  mutate(across(everything(), ~ replace(.x, .x == 999, NA))) %>%
  mutate(across(everything(), ~ replace(.x, .x == 99, NA)))

# Create a summary of missing values in each column
```

```
missing_summary <- sapply(buoy_data_list, function(x) sum(is.na(x)))
print(missing_summary)
```

```
##           date           Year             MM             DD             hh
##              0              0              0              0              0
##           WSPD            GST           WVHT            DPD            APD
##          33183          33485         144269         147961         144269
##            MWD           ATMP           WTMP           DEWP            VIS
##         327167         102761          13186         253613         443062
##           TIDE             mm Wind_Direction       Pressure
##         462301         164650          44175            261
```

```
# Basic visualization of missing values
barplot(missing_summary, main = "Missing Values by Variable", ylab = "Number of Missing Values",
        names.arg = names(missing_summary), las = 2, col = "lightblue")
```




```
# Extract year from date if not already present
buoy_data_list[, Year := year(date)]

# Check missing data by year
missing_by_year <- buoy_data_list %>%
  group_by(Year) %>%
  summarise(across(everything(), ~ sum(is.na(.), na.rm = TRUE)))

# Print missing data by year summary
print(missing_by_year)
```

```
## # A tibble: 39 x 19
##     Year  date    MM    DD    hh  WSPD   GST  WVHT   DPD   APD   MWD  ATMP  WTMP
```

```
##    <dbl> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int> <int>
##  1  1985     0     0     0     0     5    30  8719  8719  8719  8719     4    11
##  2  1986     0     0     0     0     5    20  3080  3080  3080  8168     9    10
##  3  1987     0     0     0     0  1575  1583    88    88    88  7602    13    10
##  4  1988     0     0     0     0  4627  4633    53    53    53  8071    11     8
##  5  1989     0     0     0     0     8    47   134   135   134  7933    76    26
##  6  1990     0     0     0     0   818   825    49    50    49  8703     9    10
##  7  1991     0     0     0     0     2     4    15    20    15  8730     7     2
##  8  1992     0     0     0     0     3    20    48    48    48  8736     5    12
##  9  1993     0     0     0     0     4    38   125   125   125  6677    12    19
## 10  1994     0     0     0     0  2275  2282   141   141   141   281     4  2281
## # i 29 more rows
## # i 6 more variables: DEWP <int>, VIS <int>, TIDE <int>, mm <int>,
## #   Wind_Direction <int>, Pressure <int>
```

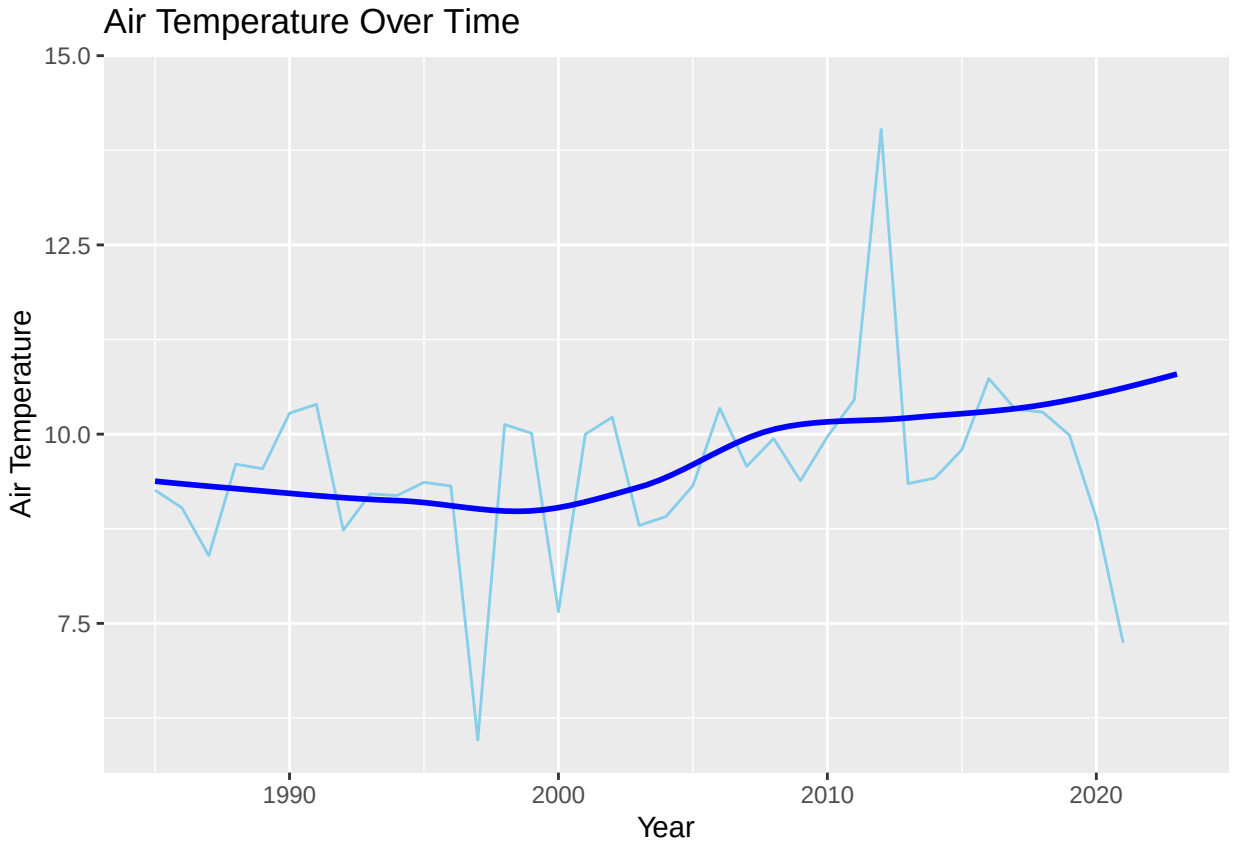
No, it is not always appropriate to convert missing data to NA automatically. If 999 represents a placeholder for missing or invalid data, it should be converted to NA. While 999 might represent a specific or extreme value rather than a placeholder for missing data. Yes, some patterns are spotted in the way/dates that these are distributed. From 1985 to 2005, there were many missing observations, showing that data collection methods were less reliable compared to 2005 to 2023. The fewer missing observations from 2005 to 2023 suggest that data collection improved or more was invested in buoy technology.

##c

```
# Ensure necessary libraries are loaded
library(ggplot2)
library(dplyr)

summary_data <- buoy_data_list %>%
  group_by(Year) %>%
  summarise(avg_atmp = mean(ATMP, na.rm = TRUE), groups = 'drop')

ggplot(summary_data, aes(x = Year, y = avg_atmp)) +
  geom_line (color = "skyblue") +
  geom_smooth(method = "loess", se = FALSE, color = "blue" ) +
  labs(title = "Air Temperature Over Time",
x = "Year", y = "Air Temperature")
```

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

```
## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).
```



Trend: There is a slight upward trend in air temperature, particularly after the year 2000. This could potentially indicate warming, which may be tied to climate change. **Fluctuations:** The data shows a number of fluctuations, especially in the 1990s, followed by a more consistent trend around the mid-2000s. **Recent Data:** The tail end of the data (near 2020) shows a notable decrease, which could be due to various factors, including localized weather events or possible missing/incomplete data. There is evidence of a warming trend over the study period, especially in recent decades.Despite short-term fluctuations, the overall long-term trend points to rising temperatures.The recent dip should be explored further, but overall, the warming trend dominates the period from 1985 to 2020.

```
model = lm(avg_atmp ~ Year, data = summary_data)
summary(model)
```

```
##
## Call:
## lm(formula = avg_atmp ~ Year, data = summary_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4397 -0.5040  0.0139  0.5168  4.0793
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -74.66171   41.04015  -1.819   0.0772 .
## Year          0.04209    0.02048   2.055   0.0472 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.39 on 36 degrees of freedom
```

```
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.105,  Adjusted R-squared:  0.08013
## F-statistic: 4.223 on 1 and 36 DF,  p-value: 0.04719
```

Year Coefficient: 0.04209 means that for each additional year, the average air temperature increases by approximately 0.04209 units.

This positive coefficient suggests a gradual warming trend over time. Each year is associated with a rise in the average air temperature by about 0.042 degrees.

Because the p-value for this coefficient is 0.0472, which is less than 0.05, this increase is statistically significant at the 95% confidence level. Thus, we can confidently say that the average air temperature has been rising over the years based on this model.

##d ###1)

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

rainfall <- read_csv("Rainfall.csv")
```

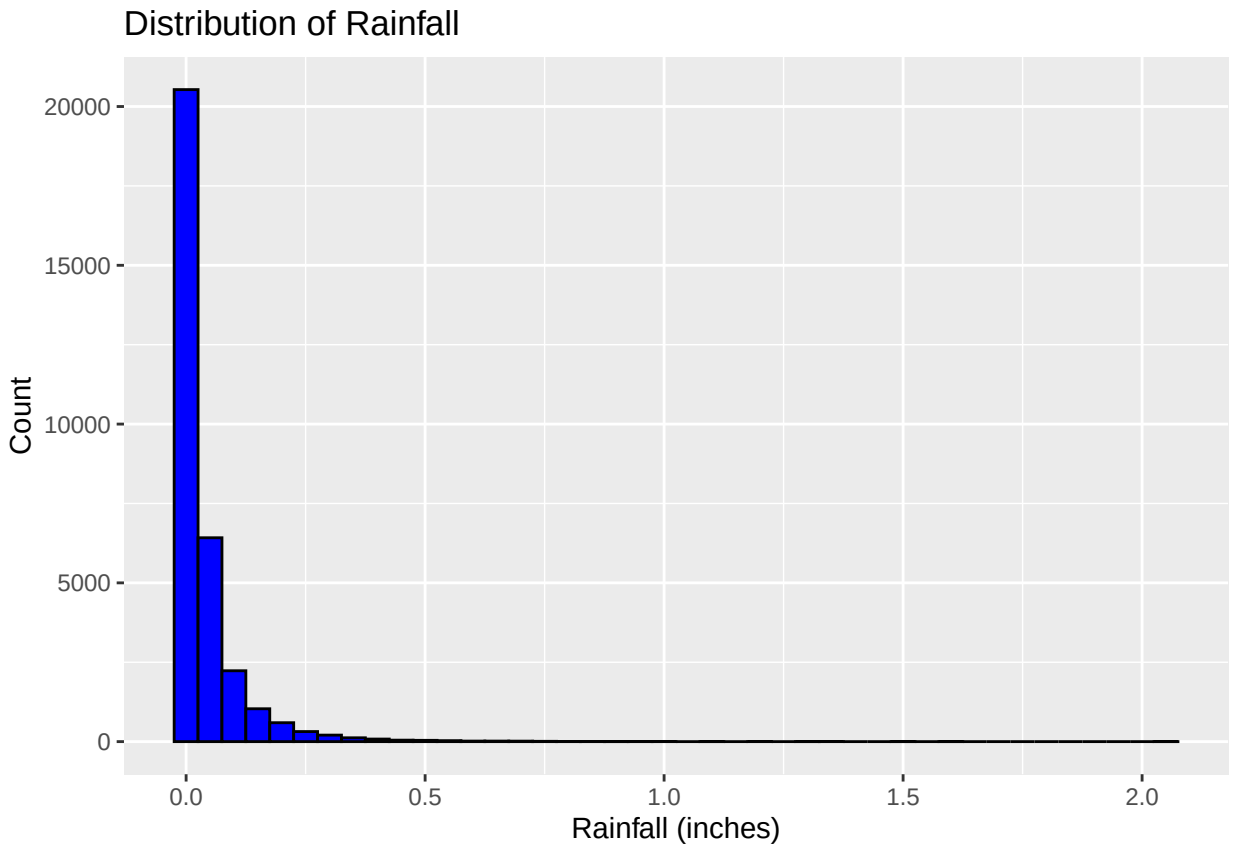
```
## Rows: 31714 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): STATION, STATION_NAME, Measurement Flag
## dbl  (1): HPCP
## lgl  (1): Quality Flag
## dttm (1): DATE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
rainfall <- rainfall %>%
  rename(date = DATE)

combined_data <- left_join(rainfall, buoy_data_list, by = "date")


# Summary statistics for rainfall and key buoy variables
summary_combined <- combined_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),
    Mean_Temperature = mean(WTMP, na.rm = TRUE),
    Mean_Pressure = mean(Pressure, na.rm = TRUE)
  )
summary_combined
```

```
## # A tibble: 1 x 6
##   Mean_Rainfall Median_Rainfall Max_Rainfall Min_Rainfall Mean_Temperature
##           <dbl>           <dbl>        <dbl>        <dbl>            <dbl>
## 1        0.0387            0.01         2.03            0             9.10
## # i 1 more variable: Mean_Pressure <dbl>
```

###2)

```r
# Histogram for Rainfall
ggplot(combined_data, aes(x = HPCP)) +
  geom_histogram(binwidth = 0.05, fill = "blue", color = "black") +
  labs(title = "Distribution of Rainfall", x = "Rainfall (inches)", y = "Count")
```

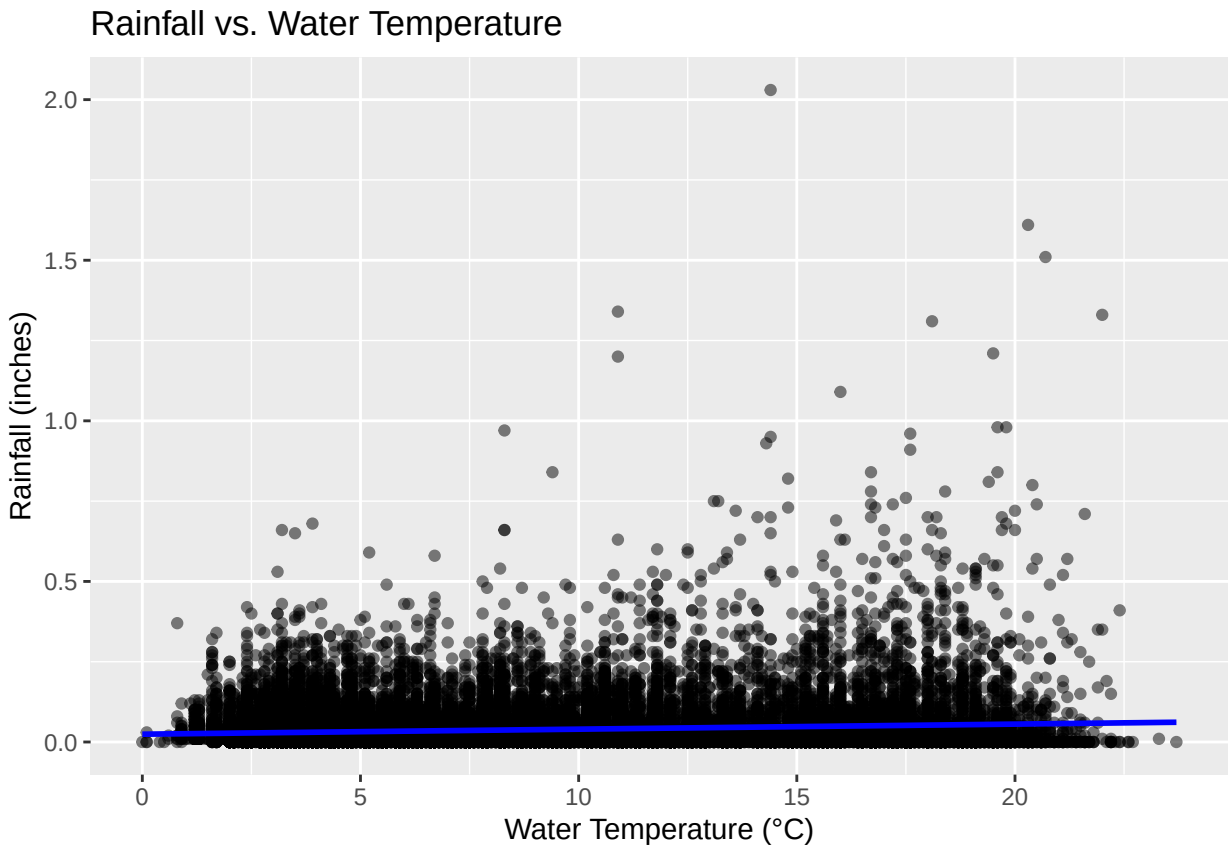



```r
ggplot(combined_data, aes(x = WTMP, y = HPCP)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", color = "blue") +
  labs(title = "Rainfall vs. Water Temperature", x = "Water Temperature (°C)", y = "Rainfall (inches)")
```

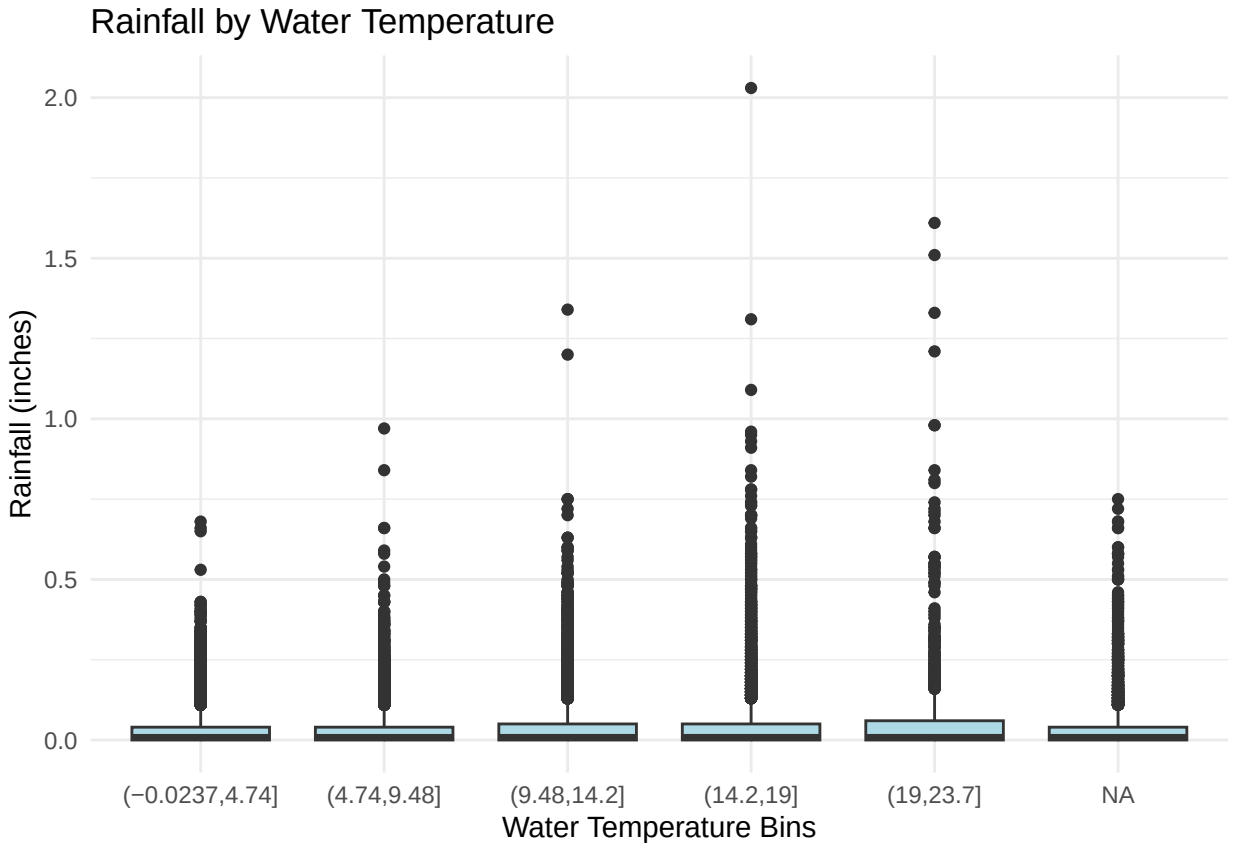
```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3075 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 3075 rows containing missing values (`geom_point()`).
```




```r
ggplot(combined_data, aes(x = cut(WTMP, breaks = 5), y = HPCP)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Rainfall by Water Temperature", x = "Water Temperature Bins", y = "Rainfall (inches)") +
  theme_minimal()
```



1. Distribution of Rainfall (Histogram): This is a common pattern for rainfall data, where the majority of time periods may experience little to no rain, while a few periods record heavy rainfall. The distribution is right-skewed, indicating that larger rainfall events are rare. **2. Rainfall vs. Water Temperature (Scatter Plot):** The relationship between water temperature and rainfall appears to be weak based on the scatter plot and regression line. However, the overall spread and density of points suggest that other factors may also be influencing rainfall events. **3. Rainfall by Water Temperature (Boxplot):** The boxplot indicates that large rainfall events (outliers) are possible at all temperature ranges, though they are more frequent at higher water temperatures. However, the overall median rainfall remains low in each bin, indicating that temperature alone may not be a strong predictor of rainfall magnitude.

###3)

```
cleaned_data <- combined_data %>%
  filter(!is.na(HPCP), !is.na(WTMP),!is.na(Pressure))

simple_model <- lm(HPCP ~ WTMP + Pressure, data = cleaned_data)

summary(simple_model)
```

```
##
## Call:
## lm(formula = HPCP ~ WTMP + Pressure, data = cleaned_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.06151 -0.03532 -0.02422  0.00576  1.98308
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  2.487e-02  4.999e-03   4.976 6.54e-07 ***
## WTMP         1.569e-03  8.513e-05  18.431  < 2e-16 ***
## Pressure    -5.439e-07  4.849e-06  -0.112    0.911
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07584 on 28633 degrees of freedom
## Multiple R-squared:  0.01173,    Adjusted R-squared:  0.01166
## F-statistic: 169.9 on 2 and 28633 DF,  p-value: < 2.2e-16
```

Despite the significance of the predictors, the R^2 value is low, indicating that rainfall prediction is difficult. Many factors, including non-linear relationships, local effects, and sudden weather events, may not be fully explained by a simple model. However, even with complex models, predicting rain accurately involves dealing with chaotic systems. As seen in our simple linear model, we may observe some patterns, but the accuracy is far from perfect. This highlights how challenging it can be to predict the weather, even with modern technology and data, leading to more understanding of why forecasts can sometimes be inaccurate.

Credit to Yiming Chen and Ruijian (Maggie) Lin.