# Impact of Water Quality on Traffic Incidents

**Name: Pranjal Adhikari**

## Introduction

The datasets chosen for this project includes water quality sampling data and traffic incidents report. These datasets are interesting to me as with water, it is very important to know the quality of water you are drinking. Contaminated water is a problem that can cause serious health issues and other problems in the body, and is an issue throughout the world. Thus, it is vital to know the quality of the water to validate whether it is safe to drink. With vehicles, they are a daily part of everyone's life, especially in the US. In Austin, TX with the increasing population, more vehicles are on the road as well. Thus, assessing the type of risks (such as accident, vehicle fire, etc.) while on the roads is important for awareness and safe driving.

The water data includes water samples collected in different parts of Austin, TX such as in natural creeks, aquifers, and lakes to assess the water quality conditions. Each row in the dataset represents a water sample collected in a specific location at a specific day and time. The data includes both categorical data, such as the location where the sample was collected, and numerical data, such as the exact number of bacteria/pathogens within a sample.

The traffic incidents data includes some type of traffic incident reported within the Austin, TX city as well. Each row is a specific incident with the exact date and time the incident occurred. Additionally, the dataset includes what type of incident occurred (accident, stalled vehicle, etc.). Both categorical and numerical data is found in the dataset, such as the address and the exact latitude/longitude of the incident.

The water and traffic data both include the exact date of when the sample was collected or when the incident occurred. Thus, the two datasets can be joined by date.

With the analysis of these two dataset, the question that will be explored is: Does water quality have any impact on the number of traffic incidents seen in Austin, TX? A potential trend that could be seen between the dataset is if the water quality deteriorates, the number of traffic incidents increases and vice versa.

Both datasets were acquired from the City of Austin open data portal. The water quality data can be found here, and the traffic incidents data can be found here.

```r
# loading necessary libraries into environment
library(tidyverse) # tidyverse
library(readxl) # read data in Excel
```

The libraries used in this project include `tidyverse` and `readxl`. The former allows for the usage of `tidyr` functions and `ggplot` for data visualization. The latter allows for the data in the Excel files that will be used in this project to be read within the R environment.

```r
# loading water data into environment
WaterData <- read_excel("WaterData.xlsx")

# loading traffic incidents data into environment
TrafficData <- read_excel("TrafficData.xlsx")
```

To load the datasets that will be used in this project, they first have to loaded into the environment. Since they are stored in an Excel file, the `read_excel` function is utilized then saved into the environment using `<-` as `WaterData` and `TrafficData`, respectively.

The `WaterData` dataset has data starting on year 1947 up to end of August 2023, while `TrafficData` has data starting on 2018 up to the end of 2022. The `WaterData` will be filtered between the same time periods as the `TrafficData` to later join the two datasets by date. Thus, the range of years that will be used in this project for both datasets is from 2018 to 2022.

```r
# separating date into four columns: year, month, day, and TIME
WaterDataNew <- WaterData |>
  separate(SAMPLE_DATE, into = c("year", "month", "daytime"), sep = "-") |>
  separate(daytime, into = c("day", "TIME"), sep = 2) |> select(-TIME)

# set year, month, and day values as numeric
WaterDataNew$year <- as.numeric(WaterDataNew$year)
WaterDataNew$month <- as.numeric(WaterDataNew$month)
WaterDataNew$day <- as.numeric(WaterDataNew$day)
```

Both data sets have the month, date, and year in one column. For easier usage, the three time periods will be separated into three different columns of year, month, day, and time. The code chunk above depicts that as the 'SAMPLE_DATE' column is separated into four total columns. The time is unneeded in this project, thus is removed from the dataset. After using the `separate` function to separate the one date column into multiple columns, the `as.numeric` function is used to set the new column's values as data type numeric, instead of the default character.

```r
# selecting the 5 year range
WaterDataNew <- WaterDataNew |> filter(year > 2017 & year < 2023)
```

The five year range as mentioned above is filtered to load into a new dataset using the `filter` function. The new `WaterDataNew` dataset will be used in the next code chunks below.

```r
# separating date into four columns: year, month, day, and time
TrafficDataNew <- TrafficData |>
  separate(`Published Date`, into = c("month", "day", "yeartime"), sep = "/") |>
  separate(yeartime, into = c("year", "TIME"), sep = " ") |>
  select(-`Status Date`, -TIME) # removing unneeded date column

# set year, month, and day values as numeric
TrafficDataNew$year <- as.numeric(TrafficDataNew$year)
TrafficDataNew$month <- as.numeric(TrafficDataNew$month)
TrafficDataNew$day <- as.numeric(TrafficDataNew$day)
```

As done with the `WaterData` dataset, the date column of the `TrafficData` is separated into four total columns. The `TrafficData` dataset has two columns of date, the 'Published Date' column and 'Status Date'. In this project, 'Published Date' is used to separate the date into the four total columns of year, month, day, and time. Thus, the 'Status Date' column is removed because it is unneeded along with the newly created time column, similar to the `WaterData` dataset.

```r
# filtering the 5 year range
TrafficDataNew <- TrafficDataNew |> filter(year > 2017 & year < 2023)
```

Mentioned above, the The traffic dataset has incomplete data for the year 2017, thus we will take data starting from year 2018.

### Wrangling

```r
# number of distinct watersheds water sample was taken from in the 5 year range
WaterDataNew |> summarize(distinct_watersheds = n_distinct(WATERSHED))
```

```
## # A tibble: 1 x 1
##   distinct_watersheds
##                 <int>
## 1                  33
```

**Summary statistics #1 (categorical):** Within the water data, there are multiple watersheds where the water sample is collected from. The chunk above computes exactly how many distinct watersheds are present in the dataset. It is found that there are 33 distinct watersheds.

```r
# filter Barton Creek watershed and bacteria/pathogens measurement
WaterDataWithBacteria <- WaterDataNew |>
  filter(WATERSHED == "Barton Creek", PARAM_TYPE == "Bacteria/Pathogens")

WaterDataWithBacteria
```

```
## # A tibble: 242 x 25
##    DATA_REF_NO SAMPLE_REF_NO  year month   day TIME_NULL WATERSHED
##          <dbl>         <dbl> <dbl> <dbl> <dbl> <chr>     <chr>
##  1     2620816        524096  2018     1    10 N         Barton Creek
##  2     2627194        525293  2018     2    14 N         Barton Creek
##  3     2627060        526438  2018     3    21 N         Barton Creek
##  4     2630246        526419  2018     3    21 N         Barton Creek
##  5     2626972        526395  2018     3    21 N         Barton Creek
##  6     2627008        526272  2018     3    21 N         Barton Creek
##  7     2679727        556034  2018     4    18 N         Barton Creek
##  8     2681868        526471  2018     4    18 N         Barton Creek
##  9     2678152        526469  2018     4    18 N         Barton Creek
## 10     2640919        528189  2018     5    23 N         Barton Creek
## # i 232 more rows
## # i 18 more variables: SAMPLE_SITE_NO <dbl>, SITE_NAME <chr>,
## #   LAT_DD_WGS84 <dbl>, LON_DD_WGS84 <dbl>, SITE_TYPE <chr>, PROJECT <chr>,
## #   SAMPLE_ID <chr>, DEPTH_IN_METERS <dbl>, MEDIUM <chr>, PARAM_TYPE <chr>,
## #   PARAMETER <chr>, QUALIFIER <chr>, RESULT <dbl>, UNIT <chr>, METHOD <chr>,
## #   FILTER <chr>, QC_TYPE <chr>, QC_FLAG <chr>
```

The water dataset has multiple ways of measuring the water quality in multiple watersheds. In this project, we will be taking a look at the Barton Creek watershed, and the bacteria/pathogens found in the water sample. The result of the water sample will be in units of MPN/100ML. In the code chunk above, a new dataset is created called `WaterDataWithBacteria` which will be used with the data described above. Since in some dates there were no water samples collected with the `PARAM_TYPE` as bacteria/pathogens, some dates will be missing in the new dataset.

```r
# lowest and highest value of bacteria/pathogens in the dataset in MPN/100ML
WaterDataWithBacteria |>
  summarise(minimum = min(RESULT), maximum = max(RESULT))
```

```
## # A tibble: 1 x 2
##   minimum maximum
##     <dbl>   <dbl>
## 1       1   2420.
```

**Summary statistics #2 (numerical):** Within the water data, there is a large range of the bacteria/pathogens found in water samples collected. The chunk above retrieves the lowest and highest concentration in MPN/100ML found in the dataset. The lowest is 1 MPN/100ML while the highest is 2419.6 MPN/100ML.

```
# creating new column for number of incidents per day
TrafficDataWithIncidents <- TrafficDataNew |>
  group_by(year, month, day) |> count() |>
  mutate(num_of_incidents = n) |> select(-n)

head(TrafficDataWithIncidents)
```

```
## # A tibble: 6 x 4
## # Groups:   year, month, day [6]
##    year month   day num_of_incidents
##   <dbl> <dbl> <dbl>            <int>
## 1  2018     1     1               93
## 2  2018     1     2              272
## 3  2018     1     3              202
## 4  2018     1     4              178
## 5  2018     1     5              209
## 6  2018     1     6              141
```

```
tail(TrafficDataWithIncidents)
```

```
## # A tibble: 6 x 4
## # Groups:   year, month, day [6]
##    year month   day num_of_incidents
##   <dbl> <dbl> <dbl>            <int>
## 1  2022    12    26               76
## 2  2022    12    27              118
## 3  2022    12    28              115
## 4  2022    12    29              129
## 5  2022    12    30              124
## 6  2022    12    31              148
```

There are multiple traffic incidents per day, thus the total number of incidents per day will have to be calculated. To do this, the `mutate` function is used to create a new column within the dataset to display the number of incidents per day. This is then added to a new dataset called `TrafficDataWithIncidents` with the exact day, month, and year along with the number of incidents for readability. The first date in the dataset (1/1/2018) had 93 incidents in the day, while the last date (12/31/2018) had 148 incidents.

```
# arrange by lowest number of incidents in a day to the highest
TrafficDataWithIncidents |>
  group_by(year) |>
  arrange(num_of_incidents)
```

```
## # A tibble: 1,823 x 4
## # Groups:   year [5]
##     year month   day num_of_incidents
##    <dbl> <dbl> <dbl>            <int>
##  1  2019     8     7                6
##  2  2018    11    23               29
##  3  2021    10    12               35
##  4  2018    11     4               39
##  5  2019     8     8               42
##  6  2020     4     5               44
##  7  2020     4    18               44
##  8  2020    12     8               45
##  9  2019     8     5               48
## 10  2018     3     3               49
## # i 1,813 more rows
```

```
# arrange by highest number of incidents in a day to the lowest
TrafficDataWithIncidents |>
  group_by(year) |>
  arrange(desc(num_of_incidents))
```

```
## # A tibble: 1,823 x 4
## # Groups:   year [5]
##     year month   day num_of_incidents
##    <dbl> <dbl> <dbl>            <int>
##  1  2021     2    12              440
##  2  2018     5     4              394
##  3  2019     5     4              382
##  4  2021     2    14              371
##  5  2021     5    29              355
##  6  2019     6    10              350
##  7  2019    10    25              345
##  8  2018     4    13              323
##  9  2019     4    13              320
## 10  2022     2    24              319
## # i 1,813 more rows
```

The chunk above arranges the `TrafficDataWithIncidents` dataset from the lowest number of incidents to highest, and vice versa, by date. It is found that the highest number of incidents was on 2/12/2021, while the lowest number of incidents was on 8/7/2019.

```
# highest and lowest number of traffic incidents per year in the dataset
MinMaxIncidentsPerYear <-
  TrafficDataWithIncidents |>
  group_by(year) |>
  summarize(minimum_incidents = min(num_of_incidents),
            maximum_incidents = max(num_of_incidents))

MinMaxIncidentsPerYear
```

```
## # A tibble: 5 x 3
##    year minimum_incidents maximum_incidents
##   <dbl>             <int>             <int>
## 1  2018                29                394
## 2  2019                 6                382
## 3  2020                44                245
## 4  2021                35                440
## 5  2022                61                319
```

**Summary statistics #3 (numerical):** Within each year in the traffic dataset, there is a date with the fewest and highest number of incidents. The chunk above displays that data for each year. That data is then saved into a new dataset titled `MinMaxIncidentsPerYear`. It is found that 2021 had the most incident in a specific date at 440 incidents, while 2019 had the the least incidents in a specific date with 6 incidents.

## Tidying

```
# pivot to display min and max num of incidents per year in column format
MinMaxIncidentsPerYear |> pivot_longer(cols = starts_with("m"),
              names_to = "variable",
              values_to = "num_of_incidents")
```

```
## # A tibble: 10 x 3
##     year variable          num_of_incidents
##    <dbl> <chr>                        <int>
##  1  2018 minimum_incidents               29
##  2  2018 maximum_incidents              394
##  3  2019 minimum_incidents                6
##  4  2019 maximum_incidents              382
##  5  2020 minimum_incidents               44
##  6  2020 maximum_incidents              245
##  7  2021 minimum_incidents               35
##  8  2021 maximum_incidents              440
##  9  2022 minimum_incidents               61
## 10  2022 maximum_incidents              319
```

The code chunk above uses the `pivot_longer` function to display the data in a column format. The columns `minimum_incidents` and `maximum_incidents` in the original are pivoted under the `variable` column and their respective values are displayed in the `num_of_incidents` column.

```
# determining average value of bacteria/pathogens in water samples per date
WaterDataWithAvgLevel <- WaterDataWithBacteria |>
  group_by(year, month, day, PARAM_TYPE, UNIT) |>
  summarize(avg_value = mean(RESULT))

WaterDataWithAvgLevel
```

```
## # A tibble: 71 x 6
## # Groups:   year, month, day, PARAM_TYPE [71]
##     year month   day PARAM_TYPE         UNIT      avg_value
##    <dbl> <dbl> <dbl> <chr>              <chr>         <dbl>
##  1  2018     1    10 Bacteria/Pathogens MPN/100ML     11
##  2  2018     2    14 Bacteria/Pathogens MPN/100ML      5.21
##  3  2018     3    21 Bacteria/Pathogens MPN/100ML      2.02
##  4  2018     4    18 Bacteria/Pathogens MPN/100ML     14.5
##  5  2018     5    23 Bacteria/Pathogens MPN/100ML     50.4
##  6  2018     6    27 Bacteria/Pathogens MPN/100ML      9.6
##  7  2018     7    13 Bacteria/Pathogens MPN/100ML    319.
##  8  2018     7    16 Bacteria/Pathogens MPN/100ML     30.5
##  9  2018     7    19 Bacteria/Pathogens MPN/100ML     12.6
## 10  2018     8    15 Bacteria/Pathogens MPN/100ML     14.5
## # i 61 more rows
```

In the `WaterDataWithBacteria` dataset, since there are multiple water samples collected within the same date, the average of the results for each of the days will be taken for simplicity purposes. The `UNIT` will still remain the same as MPN/100ML.

## Joining/Merging Datasets

```r
# number of observations in water dataset
wnum <- WaterDataWithAvgLevel |> nrow()
wnum
```

```
## [1] 71
```

```r
# number of observations in traffic dataset
tnum <- TrafficDataWithIncidents |> nrow()
tnum
```

```
## [1] 1823
```

```r
# observations that were dropped when joined
num_dropped <- tnum - wnum
num_dropped
```

```
## [1] 1752
```

```r
# joining water data with traffic incidents data by date
WaterTraffic <- WaterDataWithAvgLevel |>
  left_join(TrafficDataWithIncidents, by = c('year', 'month', 'day'))

WaterTraffic
```

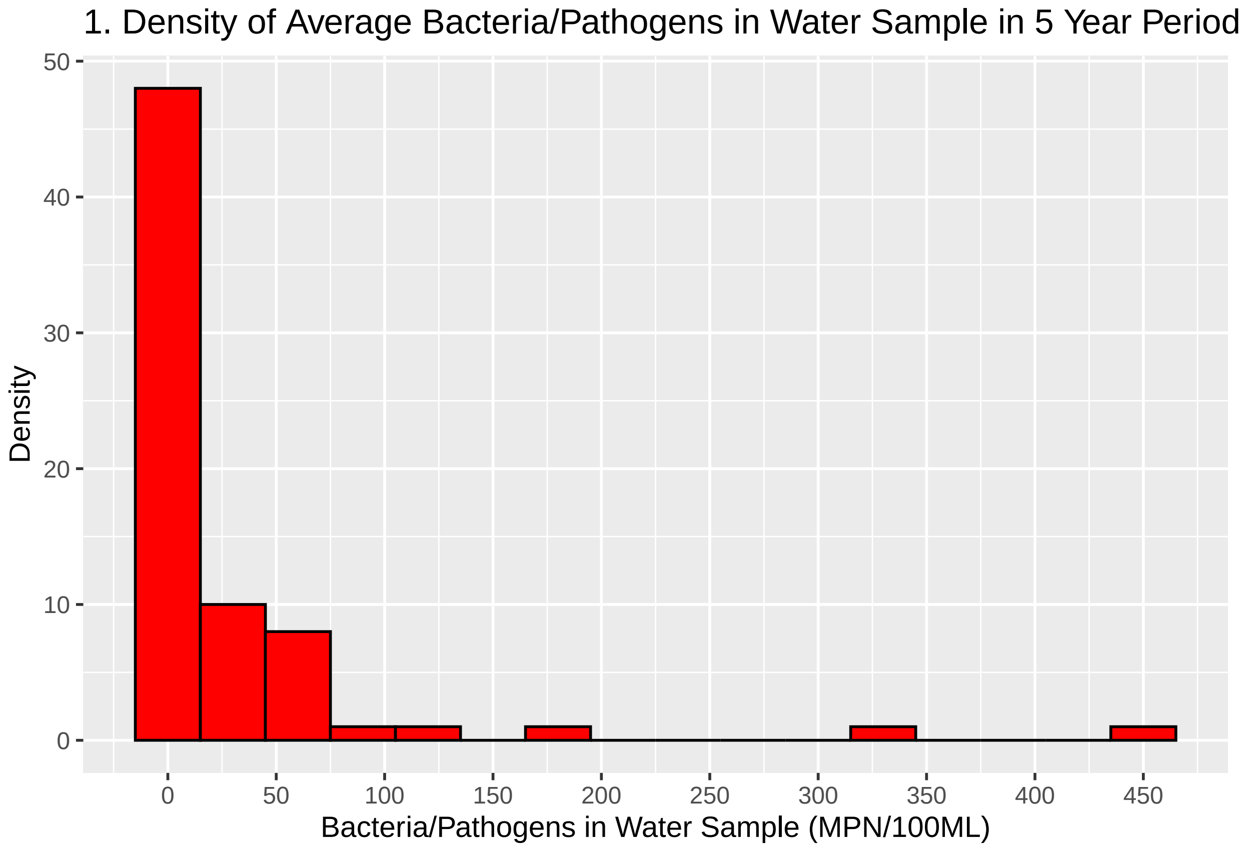
```
## # A tibble: 71 x 7
## # Groups:   year, month, day, PARAM_TYPE [71]
##      year month   day PARAM_TYPE         UNIT      avg_value num_of_incidents
##     <dbl> <dbl> <dbl> <chr>              <chr>         <dbl>            <int>
##  1  2018     1    10 Bacteria/Pathogens MPN/100ML     11                 183
##  2  2018     2    14 Bacteria/Pathogens MPN/100ML      5.21              226
##  3  2018     3    21 Bacteria/Pathogens MPN/100ML      2.02              223
##  4  2018     4    18 Bacteria/Pathogens MPN/100ML     14.5               215
##  5  2018     5    23 Bacteria/Pathogens MPN/100ML     50.4               212
##  6  2018     6    27 Bacteria/Pathogens MPN/100ML      9.6               221
##  7  2018     7    13 Bacteria/Pathogens MPN/100ML    319.                214
##  8  2018     7    16 Bacteria/Pathogens MPN/100ML     30.5                   179
##  9  2018     7    19 Bacteria/Pathogens MPN/100ML     12.6                   237
## 10  2018     8    15 Bacteria/Pathogens MPN/100ML     14.5                   194
## # i 61 more rows
```

The code chunk above joins the two datasets by year, month, and day to where the number of incidents is added to the specific date of the collected water data. IDs that appear in water dataset but not the traffic dataset includes everything (such as the site name, type of method to collect the sample, etc.) except for the year, month, and day. Those three variables are the only IDs that are in common between the two datasets. The IDs that are left out after joining include the exact longitude/latitude values of the water quality, the medium, and other columns present in the water date not found in the traffic incidents data.

The total observations in the water dataset include 71, while the traffic dataset has 1823 observations. The total number of observations that were dropped after joining the two datasets is 1752. A potential issue is some dates may not get added as some dates are missing from the water data, however whichever dates match between the two datasets are added.

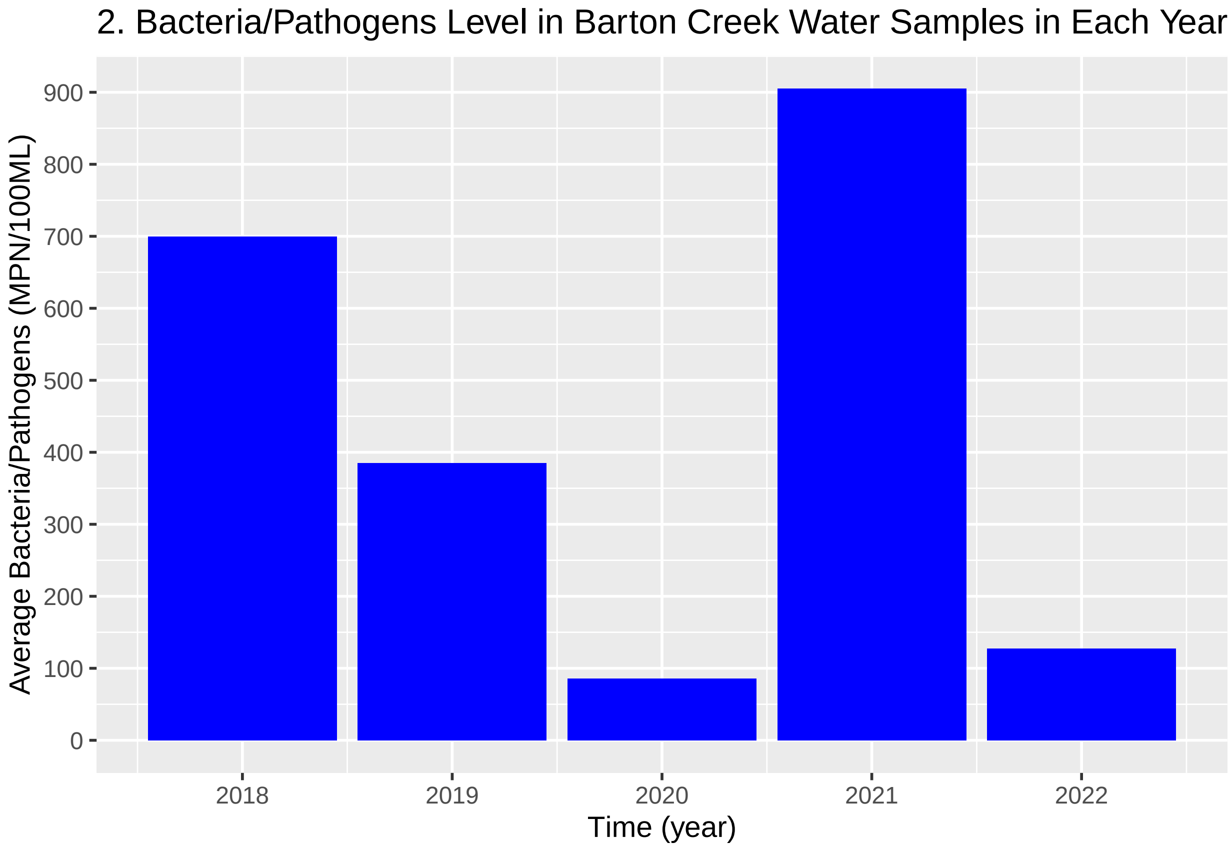
## Visualizing

```r
# density of bacteria/pathogens in water samples over 5 year period
WaterDataWithAvgLevel |>
  ggplot(aes(x = avg_value)) +
  geom_histogram(color = "black", fill = "red", binwidth = 30) +
  labs(x = "Bacteria/Pathogens in Water Sample (MPN/100ML)",
       y = "Density",
       title = "1. Density of Average Bacteria/Pathogens in Water Sample in 5 Year Period") +
  scale_x_continuous(breaks = seq(0, 500, 50)) +
  scale_y_continuous(breaks = seq(0, 50, 10)) +
  theme_grey()
```



In the graph above, the concentration of bacteria/pathogens level in the water samples collected between 2018 to 2022 is shown. In the 5 year period from 2018 to 2022, the level is relatively low as the graph is skewed right, with most values being under 75 MPN/100ML. Though there are some outliers, it can be assumed that the bacteria/pathogen levels in the Barton Creek watershed is low for the 5 year period.

```r
# plot of bacteria/pathogens in water samples overtime
WaterTraffic |> group_by(year) |>
  ggplot(aes(x = year, y = avg_value)) +
  geom_bar(stat = "identity", fill = "blue") +
  labs(x = "Time (year)", y = "Average Bacteria/Pathogens (MPN/100ML)",
       title = "2. Bacteria/Pathogens Level in Barton Creek Water Samples in Each Year") +
  scale_x_continuous(breaks = seq(2018, 2022, 1)) +
  scale_y_continuous(breaks = seq(0, 1000, 100)) +
  theme_grey()
```



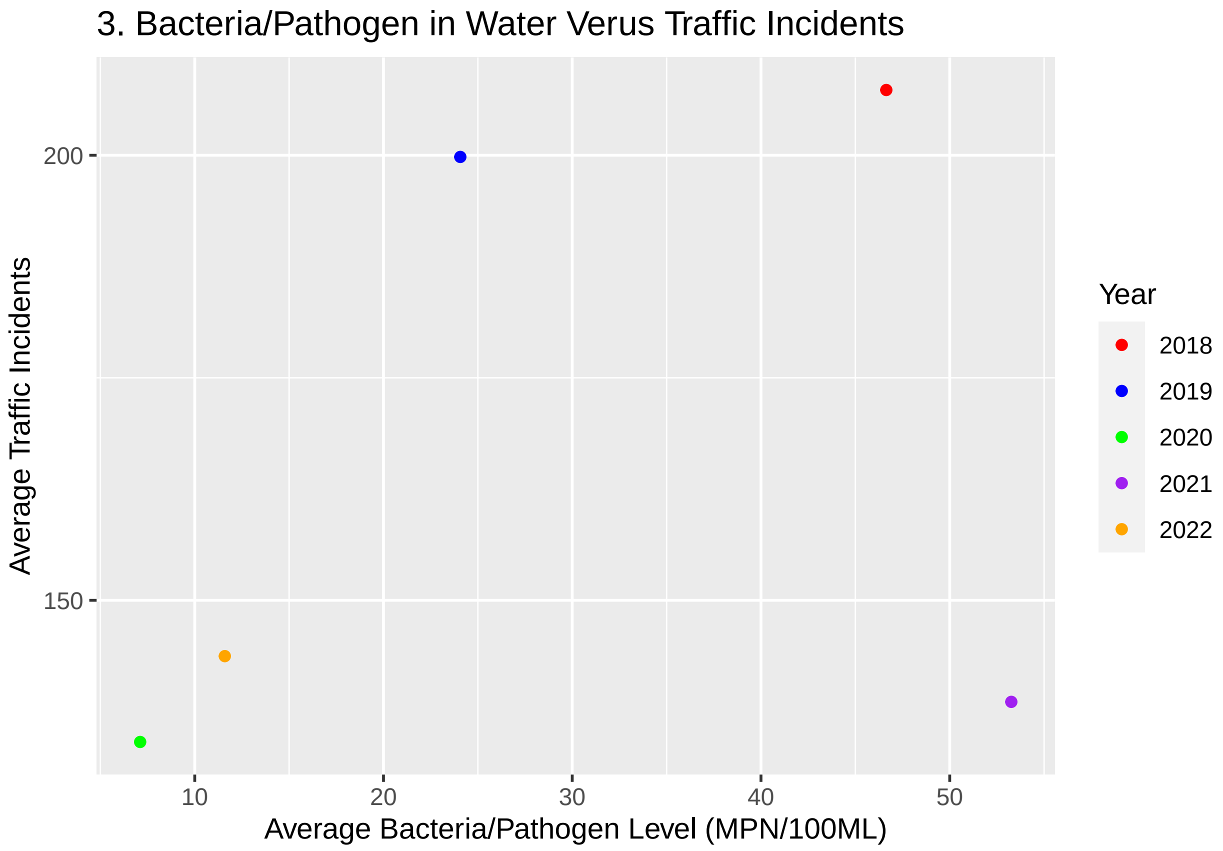
In the graph above, tt can be seen the total sum of the level of bacteria/pathogens in water samples throughout the 5 years. The years 2020 and 2022 had the lowest value of bacteria/pathogen in the water with a level of ~100 MPN/100ML. The year 2021 had the largest sum of bacteria/pathogen levels in the water samples in the 5 year range at ~900 MPN/100ML. All in all, the sum of the levels of bacteria/pathogens in the water samples collected in the five year range for the Barton Creek watershed slowly decreased until the year 2021. Then it suddenly increased then came back down in the year 2022.

```
# creating new column with average water quality value per year
WaterYearAvgVal <- WaterTraffic |> group_by(year) |>
  summarize(year_avg_water_value = mean(avg_value)) |>
  mutate(year_avg_water_value)

# creating new column with average number of incidents per year
TrafficYearAvgVal <- WaterTraffic |> group_by(year) |>
  summarize(year_avg_incidents = mean(num_of_incidents)) |>
  mutate(year_avg_incidents)

# merging average number of incidents to average water quality value
WaterTrafficYearAvg <- WaterYearAvgVal |>
  left_join(TrafficYearAvgVal, by = "year")

# plot of water quality versus number of incidents per year
WaterTrafficYearAvg |>
  ggplot(aes(x = year_avg_water_value, y = year_avg_incidents, color = factor(year))) +
  geom_point() +
  labs(x = "Average Bacteria/Pathogen Level (MPN/100ML)",
       y = "Average Traffic Incidents",
       title = "3. Bacteria/Pathogen in Water Verus Traffic Incidents",
       color = "Year") +
  scale_x_continuous(breaks = seq(0, 60, 10)) +
  scale_y_continuous(breaks = seq(0, 210, 50)) +
  scale_color_manual(values = c("2018" = "red", "2019" = "blue", "2020" = "green", "2021" = "purple", "
```



```
  #theme_minimal()
```

The graph above depicts the relationship between the average bacteria/pathogen level and the average traffic incidents for each year from 2018 to 2022. It can be seen that the relationship between the two variables is positive and direct, where if one increases, so does the other. The outlier in this specific graph can be seen in the year 2021, where the ratio between average bacteria/pathogen level and average traffic incident is much higher and does not follow the other years. Nonetheless, it can be concluded from the graph above that in the specific 5 year period between 2018 to 2022, as the average bacteria/pathogen level increases, so does the average number of traffic incidents in the year.

## Discussion

Regarding the research question in this project, it can be concluded that there is an impact of water quality on traffic incidents within the city of Austin, TX. From visualization 3, it can be seen that if the water quality is low (meaning high bacteria/pathogen level), there will be an increase in the number of traffic incidents. Additionally, from the statistics seen when the water data was joined with traffic data, the number of incidents is higher when there is higher level of bacteria/pathogens per date.

One challenging part of this project was determining how to deal with the observations within each data set that had multiple entries for the same date. From the process, I learned how to utilize functions to determine the average values for the water quality and number of traffic incidents. I would like to give my gratitude to the TA Soumyabrata Bose for helping me with this process.