UlaanBaatar2

2025-04-23

Data Wrangling

```
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(here)
```

```
## here() starts at /home/guest/EDA_Spring2025_kbk
```

```
library(lubridate)

#read all files.
Ulaanbaatar_2015 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2015_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2016 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2016_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2017 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2017_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2018 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2018_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2019 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2019_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2020 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2020_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2021 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2021_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2022 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2022_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2023 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2023_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2024 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2024_YTD.csv"), stringsAsFactors = T
Ulaanbaatar_2025 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2025_YTD.csv"), stringsAsFactors = T

#merge files into one file.
Ulaanbaatar_PM2.5 <- bind_rows(Ulaanbaatar_2015,Ulaanbaatar_2016,Ulaanbaatar_2017,Ulaanbaatar_2018,Ulaa

#remove yearly datasets from environment if wanted.
#rm(Ulaanbaatar_2015,Ulaanbaatar_2016,Ulaanbaatar_2017,Ulaanbaatar_2018,Ulaanbaatar_2019,Ulaanbaatar_20

#clean -999 AQI values, in order to prevent failure in mean calculations.
```

```r
Ulaanbaatar_Clean <- Ulaanbaatar_PM2.5 %>% filter(AQI != -999)

#create monthly data by taking mean of every month.
Ulaanbaatar_Monthly <- Ulaanbaatar_Clean %>%
  group_by(Year,Month) %>%
  summarise(mean_AQI = mean(AQI, na.rm = TRUE)) %>%
  mutate(Year_Month = sprintf("%d-%02d", Year, Month)) %>%
  select(Year_Month,mean_AQI)
```

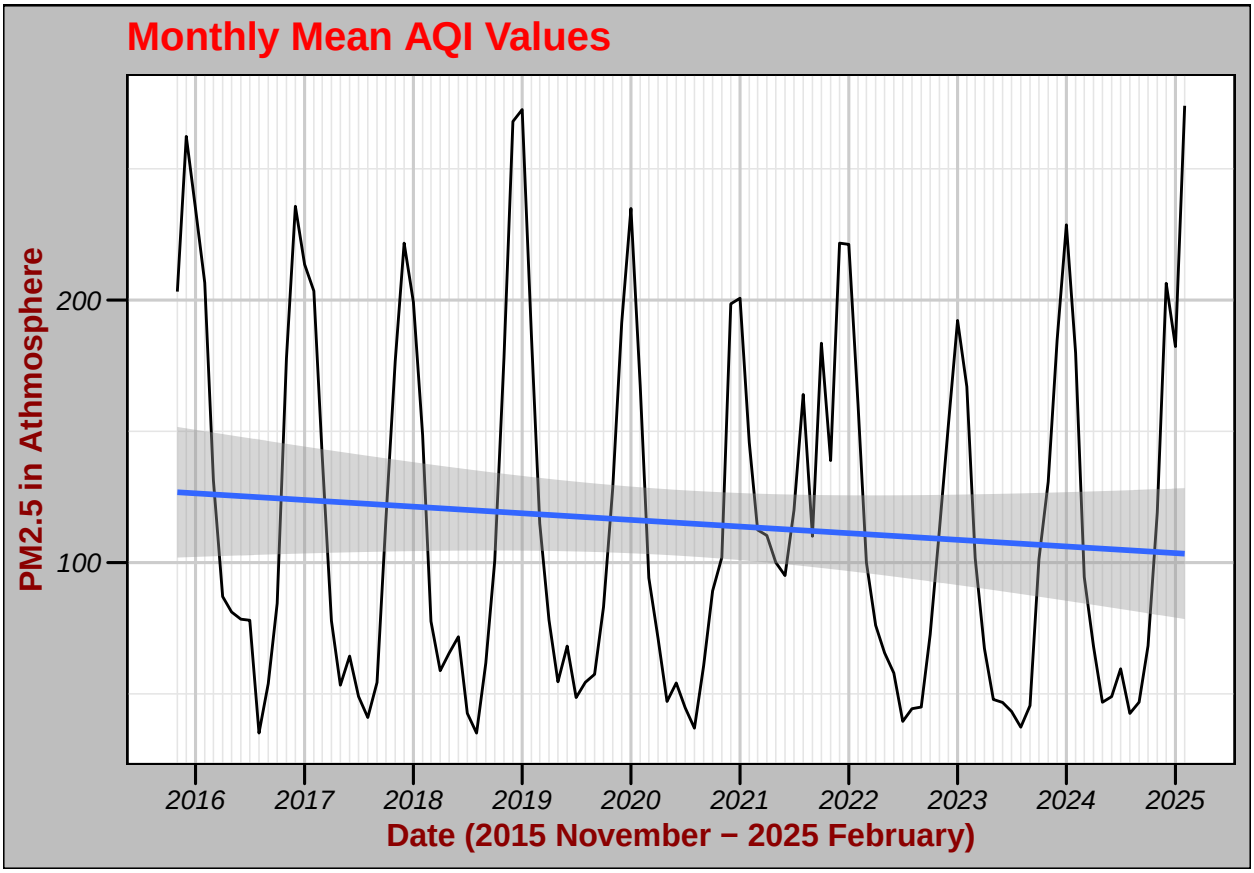
```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
## Adding missing grouping variables: `Year`
```

```r
#create date column
Ulaanbaatar_Monthly <- Ulaanbaatar_Monthly %>%
  mutate(
    Date = as.Date(paste0(Year_Month, "-01")),
    Year = year(Date),
    Month = month(Date)
  ) %>%
  select(Year, Month, mean_AQI, Date)

#detect starting and ending months.
month_range <- seq(
  from = min(Ulaanbaatar_Monthly$Date),
  to   = max(Ulaanbaatar_Monthly$Date),
  by = "month"
)

#remove Date column in order to prevent two same columns after left_join. realized after left_join and
Ulaanbaatar_Monthly <- Ulaanbaatar_Monthly %>% select(-Date)

#create missing months.
Ulaanbaatar_Monthly_Full <- data.frame(
  Date = month_range
) %>%
  mutate(
    Year = year(Date),
    Month = month(Date)
  ) %>%
  left_join(Ulaanbaatar_Monthly, by = c("Year", "Month"))

library(zoo)
```

```
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library(ggthemes)

#fill missing months by linear interpolation
Ulaanbaatar_Monthly_Full$mean_AQI <- na.approx(Ulaanbaatar_Monthly_Full$mean_AQI, na.rm = FALSE)

theme_Ulaanbaatar <- theme_base() +
 theme(
 plot.background = element_rect(colour = 'black', fill = 'grey'),
 #background is grey and frame is black
 plot.title = element_text(size = 15, colour = 'red'),
 #title of the plot is red and size of 15
 axis.title = element_text(size = 12, face = "bold", colour = "darkred"),
 #axis labels are dark red, bold and size of 12
 axis.text = element_text(size = 10, face = "italic"),
 #indicators of the axis are italic and size of 10
 legend.position = 'bottom', #legend will be at the bottom of the plot
 panel.grid.minor = element_line(color = "grey90"),  #show minor grids very slightly
 panel.grid.major = element_line(color = "grey80"))  #major grid for years


#show a line plot of monthly mean values.
ggplot(Ulaanbaatar_Monthly_Full, aes(x = Date, y = mean_AQI)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(
    title = "Monthly Mean AQI Values",
    x = "Date (2015 November - 2025 February)",
    y = "PM2.5 in Athmosphere"
  ) +
  theme_Ulaanbaatar +
  scale_x_date(
    date_breaks = "1 year",    #show every year, without this it shows only even years
    date_labels = "%Y",        #show years only
    minor_breaks = seq(min(Ulaanbaatar_Monthly_Full$Date), max(Ulaanbaatar_Monthly_Full$Date), by = "1 m
  ) +
  theme_Ulaanbaatar
```

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




```
#create daily data by taking mean of every day.
Ulaanbaatar_Daily <- Ulaanbaatar_Clean %>%
  group_by(Year,Month,Day) %>%
  summarise(mean_AQI = mean(AQI, na.rm = TRUE)) %>%
  mutate(Date = sprintf("%d-%02d-%02d", Year, Month, Day))
```

```
## `summarise()` has grouped output by 'Year', 'Month'. You can override using the
## `.groups` argument.
```

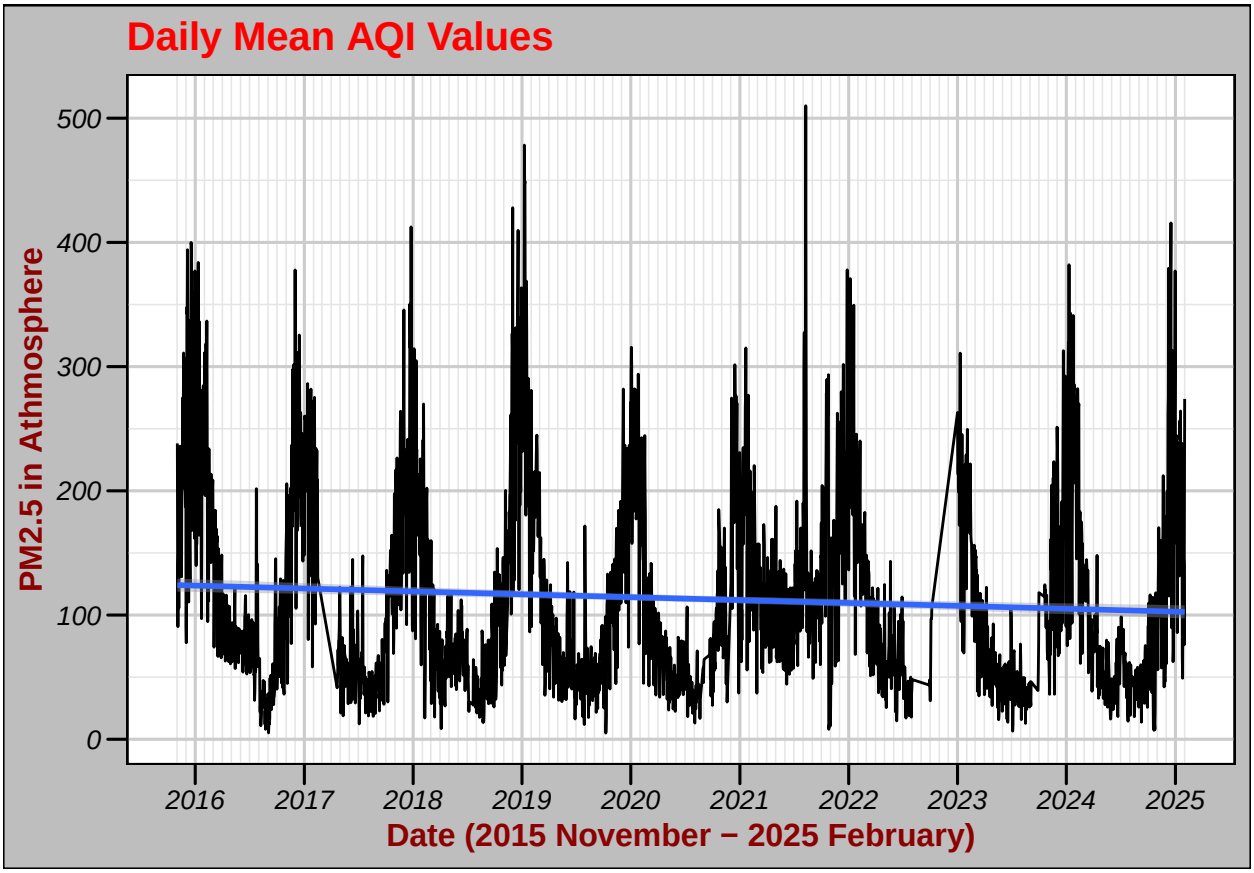
```
#change class of Date to date. and check.
Ulaanbaatar_Daily$Date <- as.Date(Ulaanbaatar_Daily$Date, format = "%Y-%m-%d")
class(Ulaanbaatar_Daily$Date)
```

```
## [1] "Date"
```

```
#detect starting and ending days.
day_range <- seq(
  from = min(Ulaanbaatar_Daily$Date),
  to   = max(Ulaanbaatar_Daily$Date),
  by = "day"
)

#remove Date column in order to prevent two same columns after left_join. realized after left_join and
Ulaanbaatar_Daily <- Ulaanbaatar_Daily %>% select(-Date)
```

```r
#create missing days.
Ulaanbaatar_Daily_Full <- data.frame(
  Date = day_range
) %>%
  mutate(
    Year = year(Date),
    Month = month(Date),
    Day = day(Date)
  ) %>%
  left_join(Ulaanbaatar_Daily, by = c("Year", "Month", "Day"))

#fill missing days by linear interpolation
Ulaanbaatar_Daily_Full$mean_AQI <- na.approx(Ulaanbaatar_Daily_Full$mean_AQI, na.rm = FALSE)

#show a line plot of daily mean values.
ggplot(Ulaanbaatar_Daily_Full, aes(x = Date, y = mean_AQI)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(
    title = "Daily Mean AQI Values",
    x = "Date (2015 November - 2025 February)",
    y = "PM2.5 in Athmosphere"
  ) +
  theme_Ulaanbaatar +
  scale_x_date(
    date_breaks = "1 year",    #show every year, without this it shows only even years
    date_labels = "%Y",        #show years only
    minor_breaks = seq(min(Ulaanbaatar_Monthly_Full$Date), max(Ulaanbaatar_Monthly_Full$Date), by = "1 m
  ) +
  theme_Ulaanbaatar
```

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

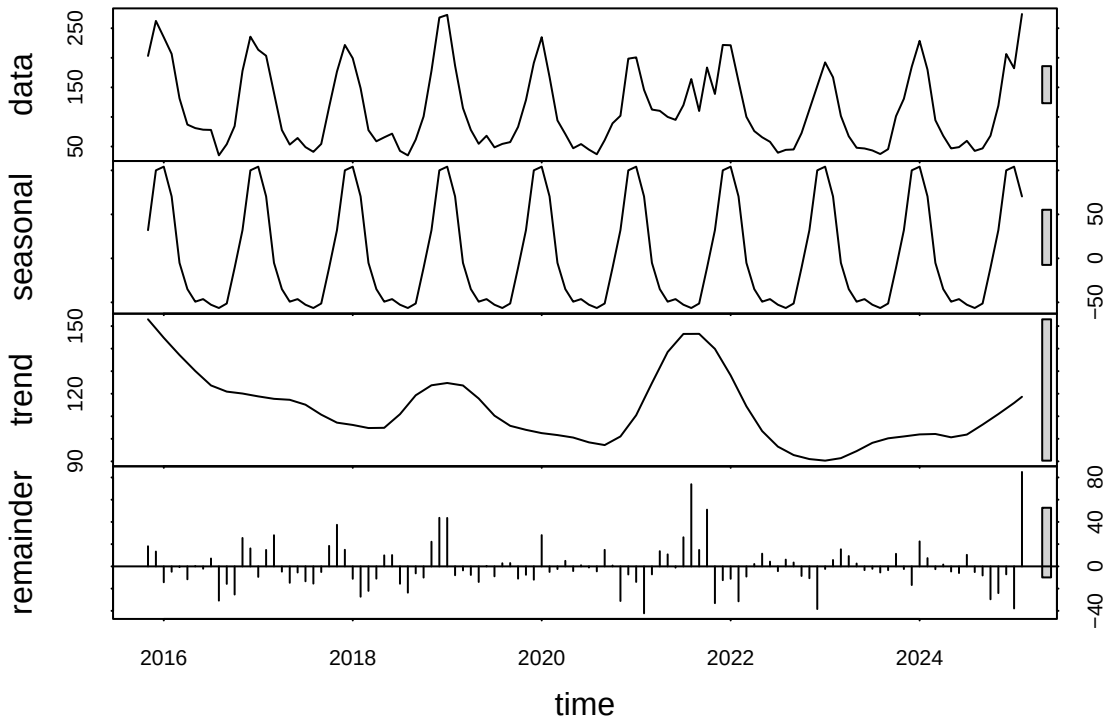


Time Series Analysis

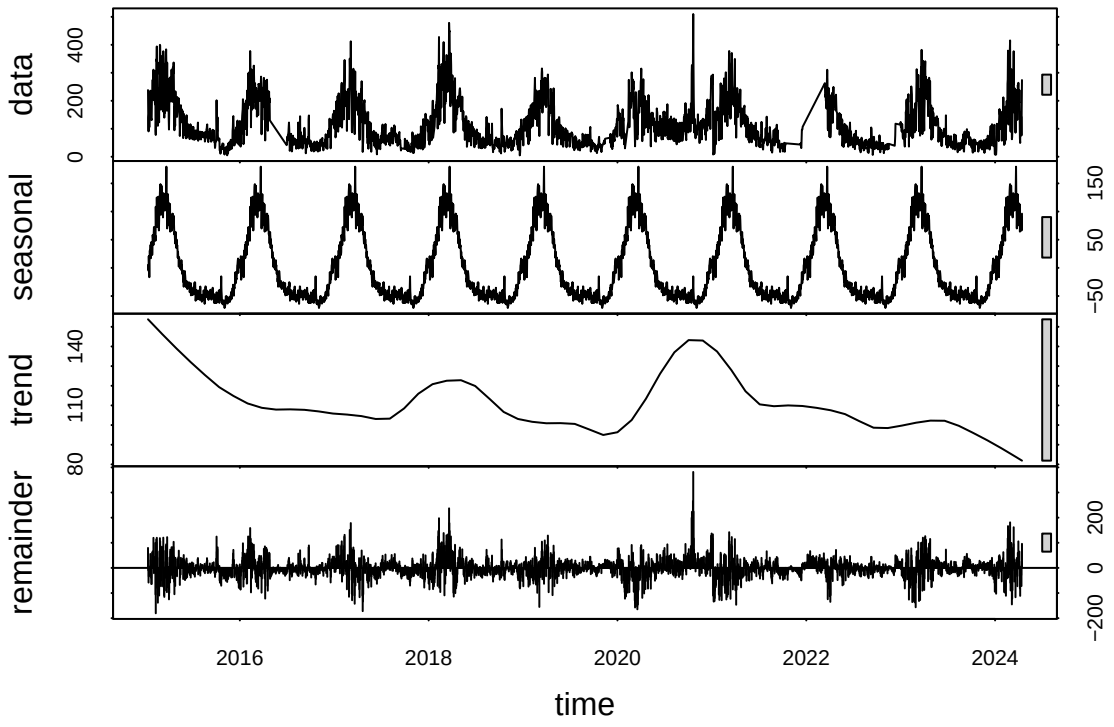
```
library(Kendall)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
#create a monthly ts object including monthly averages from 2015-11 with 12 months cycle.
Ulaanbaatar_Monthly.ts <- ts(Ulaanbaatar_Monthly_Full$mean_AQI,
                             start= c(2015,11),
                             frequency=12)

#decompose monthly time series.
Ulaanbaatar_Monthly.decomposed <- stl(Ulaanbaatar_Monthly.ts, s.window = "periodic")
plot(Ulaanbaatar_Monthly.decomposed)
```



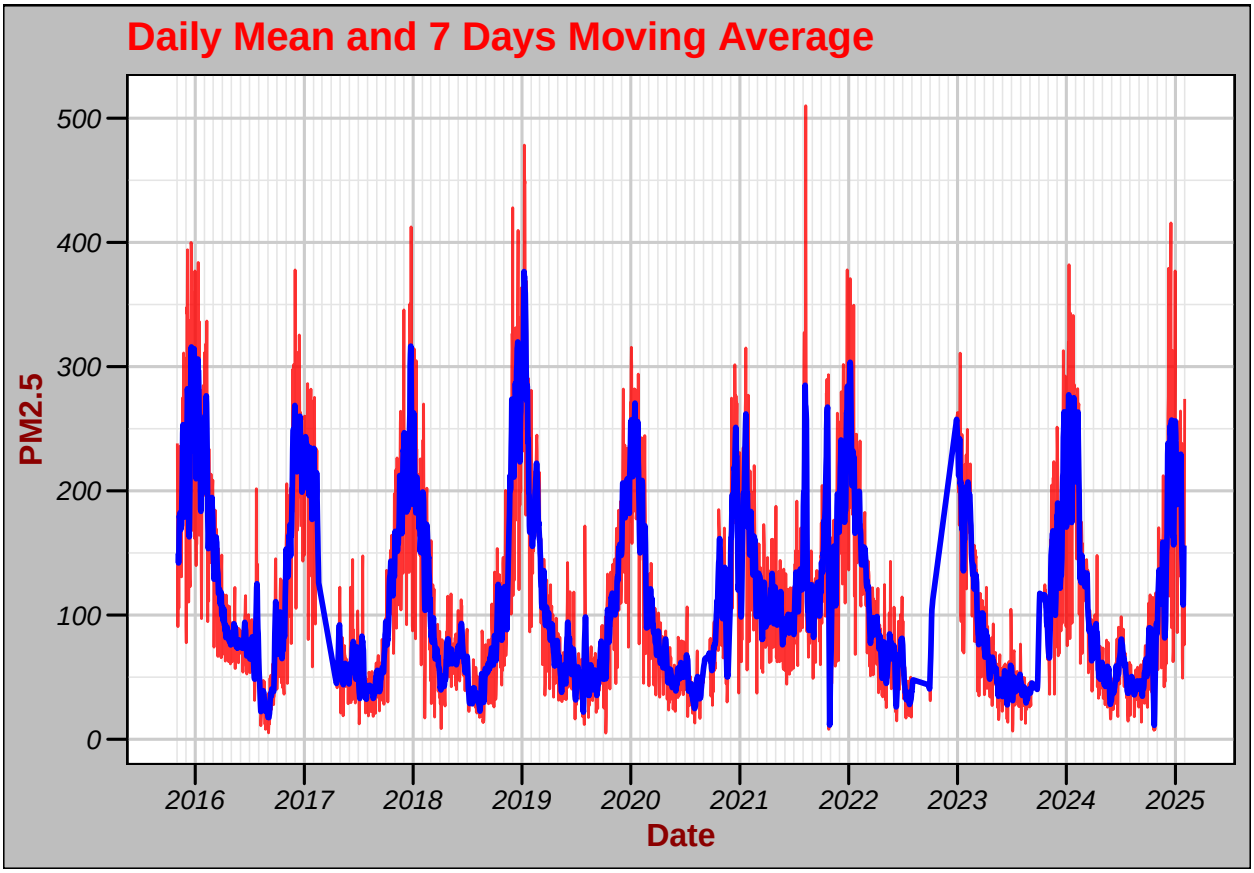

```
#create a daily ts object including daily averages from 2015-11-2 with 365 days cycle.
Ulaanbaatar_Daily.ts <- ts(Ulaanbaatar_Daily_Full$mean_AQI,
                           start= c(2015,11,2),
                           frequency=365)

#decompose daily time series.
Ulaanbaatar_Daily.decomposed <- stl(Ulaanbaatar_Daily.ts,s.window="periodic")
plot(Ulaanbaatar_Daily.decomposed)
```




```
#moving averages for 7 days (daily).
Ulaanbaatar_Daily_Full <- Ulaanbaatar_Daily_Full %>%
  mutate(Moving_Avg = rollmean(mean_AQI, k = 7, fill = NA, align = "center"))

#moving average wrt daily mean AQI value
ggplot(Ulaanbaatar_Daily_Full, aes(x = Date)) +
  geom_line(aes(y = mean_AQI), color = "red", alpha = 0.8) +
  geom_line(aes(y = Moving_Avg), color = "blue", size = 1) +
  labs(title = "Daily Mean and 7 Days Moving Average", y = "PM2.5") +
  theme_Ulaanbaatar +
  scale_x_date(
    date_breaks = "1 year",    #show every year, without this it shows only even years
    date_labels = "%Y",        #show years only
    minor_breaks = seq(min(Ulaanbaatar_Monthly_Full$Date), max(Ulaanbaatar_Monthly_Full$Date), by = "1 m
  )
```

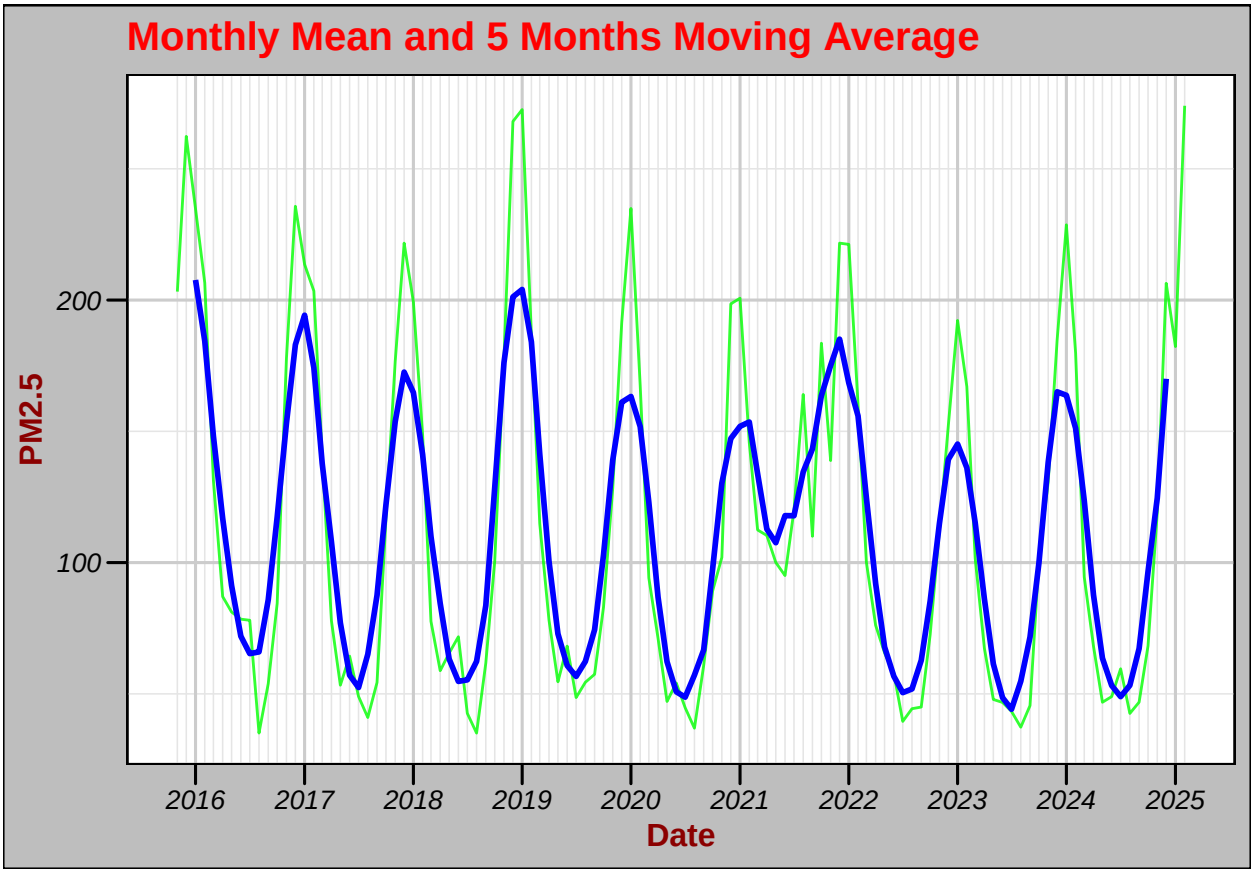
```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

```
## Warning: Removed 6 rows containing missing values or values outside the scale range
## ('geom_line()').
```




```
#moving averages for 5 months (monthly).
Ulaanbaatar_Monthly_Full <- Ulaanbaatar_Monthly_Full %>%
  mutate(Moving_Avg = rollmean(mean_AQI, k = 5, fill = NA, align = "center"))

#moving average wrt daily mean AQI value
ggplot(Ulaanbaatar_Monthly_Full, aes(x = Date)) +
  geom_line(aes(y = mean_AQI), color = "green", alpha = 0.8) +
  geom_line(aes(y = Moving_Avg), color = "blue", size = 1) +
  labs(title = "Monthly Mean and 5 Months Moving Average", y = "PM2.5") +
  theme_Ulaanbaatar +
  scale_x_date(
    date_breaks = "1 year",    #show every year. without this, it shows only even years
    date_labels = "%Y",        #show years only
    minor_breaks = seq(min(Ulaanbaatar_Monthly_Full$Date), max(Ulaanbaatar_Monthly_Full$Date), by = "1 m
  )
```

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




```
#trend analysis (monthly averages)
Monthly_Trend <- Kendall::SeasonalMannKendall(Ulaanbaatar_Monthly.ts)
Monthly_Trend
```

```
## tau = -0.265, 2-sided pvalue =0.00042021
```

```
#Based on seasonal Mann-Kendall test, there is a decreasing trend in AQI values where tau=-0.265 and it

#extract the components and turn them into data frames
Ulaanbaatar_Monthly_components <- as.data.frame(Ulaanbaatar_Monthly.decomposed$time.series[,1:3])

#subtract the seasonal component from the original time series
Ulaanbaatar_Monthly.ts_noseason <- Ulaanbaatar_Monthly.ts - Ulaanbaatar_Monthly_components$seasonal

#run the Mann-Kendall test on the deseasonalized time series
Monthly_noseason <- MannKendall(Ulaanbaatar_Monthly.ts_noseason)
Monthly_noseason
```

```
## tau = -0.252, 2-sided pvalue =8.49e-05
```

```
#with extracting seasonal effects we can say that there is a decreasing trend in AQI values where tau=-

#trend analysis (daily averages)
Daily_Trend <- Kendall::SeasonalMannKendall(Ulaanbaatar_Daily.ts)
Daily_Trend
```

```
## tau = -0.0728, 2-sided pvalue =1.0472e-07
```

```
#Based on seasonal Mann-Kendall test, there is a slightly decreasing trend in AQI values where tau=-0.0

#extract the components and turn them into data frames
Ulaanbaatar_Daily_components <- as.data.frame(Ulaanbaatar_Daily.decomposed$time.series[,1:3])

#subtract the seasonal component from the original time series
Ulaanbaatar_Daily.ts_noseason <- Ulaanbaatar_Daily.ts - Ulaanbaatar_Daily_components$seasonal

#run the Mann-Kendall test on the deseasonalized time series
Daily_noseason <- MannKendall(Ulaanbaatar_Daily.ts_noseason)
Daily_noseason
```

```
## tau = -0.0756, 2-sided pvalue =4.3596e-11
```

```
#with extracting seasonal effects we can say that there is a slightly decreasing trend in AQI values wh
```

Regression Analysis

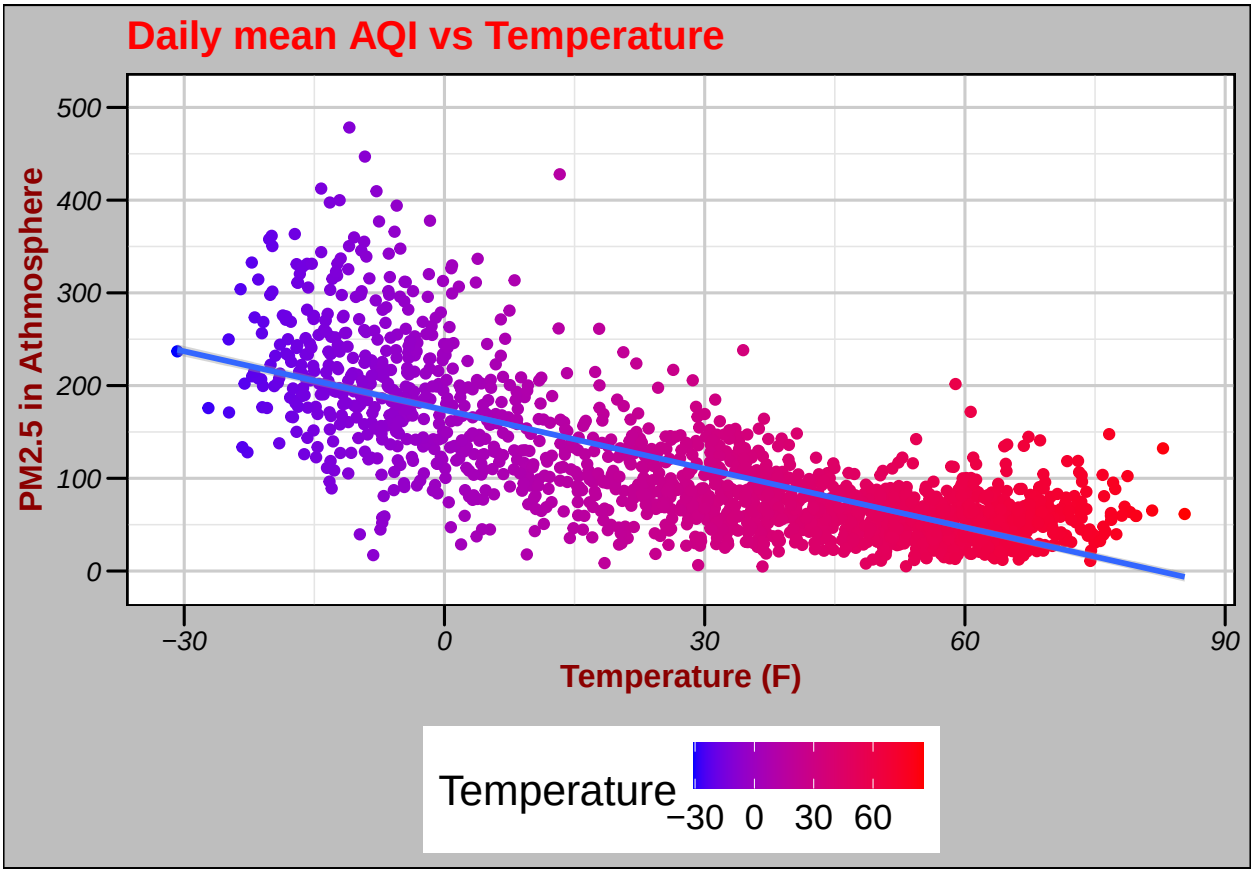
```
#read 2015-2020 weather data. could find only this free in csv.
Weather_2015_2020 <- read.csv(here("Final/Data_Raw/weather_dot_com_2015_2020.csv"), stringsAsFactors = 

#convert to daily.
Weather_Daily <- Weather_2015_2020 %>%
  mutate(Date = as.Date(date)) %>%
  group_by(Date) %>%
  summarise(Temperature = mean(temp, na.rm = TRUE))

#merge daily mean AQI values with daily temperatures.
Ulaanbaatar_AQI_Temp <- Ulaanbaatar_Daily_Full %>%
  select(Date, mean_AQI) %>%
  left_join(Weather_Daily, by = "Date")

#plot mean AQI values with respect to daily temperatures.
ggplot(Ulaanbaatar_AQI_Temp, aes(x = Temperature, y = mean_AQI, colour = Temperature)) +
  geom_point() +
  scale_color_gradient(
    low = "blue",   #blue for low temperatures
    high = "red"    #red for high temperatures
  ) +
  geom_smooth(method = "lm") +
  labs(title = "Daily mean AQI vs Temperature",
       x = "Temperature (F)",
       y = "PM2.5 in Athmosphere"
       ) +
  theme_Ulaanbaatar
```

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

```
## Warning: Removed 1607 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: The following aesthetics were dropped during statistical transformation:
## colour.
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

```
## Warning: Removed 1607 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
#introducing raw coal ban on 15 May 2019.
Ulaanbaatar_AQI_Temp <- Ulaanbaatar_AQI_Temp %>%
  mutate(
    Coal_Ban = as.integer(Date >= ymd("2019-05-15"))
  )

#regression analysis between AQI values with temperature and coal ban.
Reg_Temp_CoalBan <- lm(mean_AQI ~ Temperature + Coal_Ban, data = Ulaanbaatar_AQI_Temp, na.action = na.o
summary(Reg_Temp_CoalBan)
```

```
##
```

```
## Call:
## lm(formula = mean_AQI ~ Temperature + Coal_Ban, data = Ulaanbaatar_AQI_Temp,
##     na.action = na.omit)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -177.416  -31.226   -2.533   23.757  278.066
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 177.49460    1.90145  93.347  < 2e-16 ***
## Temperature  -2.07673    0.04279 -48.533  < 2e-16 ***
## Coal_Ban    -15.46768    2.56406  -6.032 1.96e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.69 on 1770 degrees of freedom
##   (1607 observations deleted due to missingness)
## Multiple R-squared:  0.586,  Adjusted R-squared:  0.5856
## F-statistic:  1253 on 2 and 1770 DF,  p-value: < 2.2e-16
```

```
#According to the analysis, PM2.5 levels are correlated with both temperature and coal ban.

#regression analysis between AQI values and coal ban. temperature data ends in 2020 August.
Reg_CoalBan <- lm(mean_AQI ~ Coal_Ban, data = Ulaanbaatar_AQI_Temp)
summary(Reg_CoalBan)
```

```
##
## Call:
## lm(formula = mean_AQI ~ Coal_Ban, data = Ulaanbaatar_AQI_Temp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -117.02  -60.39  -24.45   43.51  402.16
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  122.158      2.169  56.329  < 2e-16 ***
## Coal_Ban     -14.321      2.758  -5.193  2.2e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 77.89 on 3378 degrees of freedom
## Multiple R-squared:  0.007919,   Adjusted R-squared:  0.007625
## F-statistic: 26.96 on 1 and 3378 DF,  p-value: 2.197e-07
```

Correlation Analysis

```
#starting heath data.
#births under 2500g in Ulaanbaatar
Birth_Under_2500 <- read.csv(here("Final/Data_Raw/BIRTH WEIGTH LOWER THAN 2500 GRAMS.csv"), stringsAsFa
```

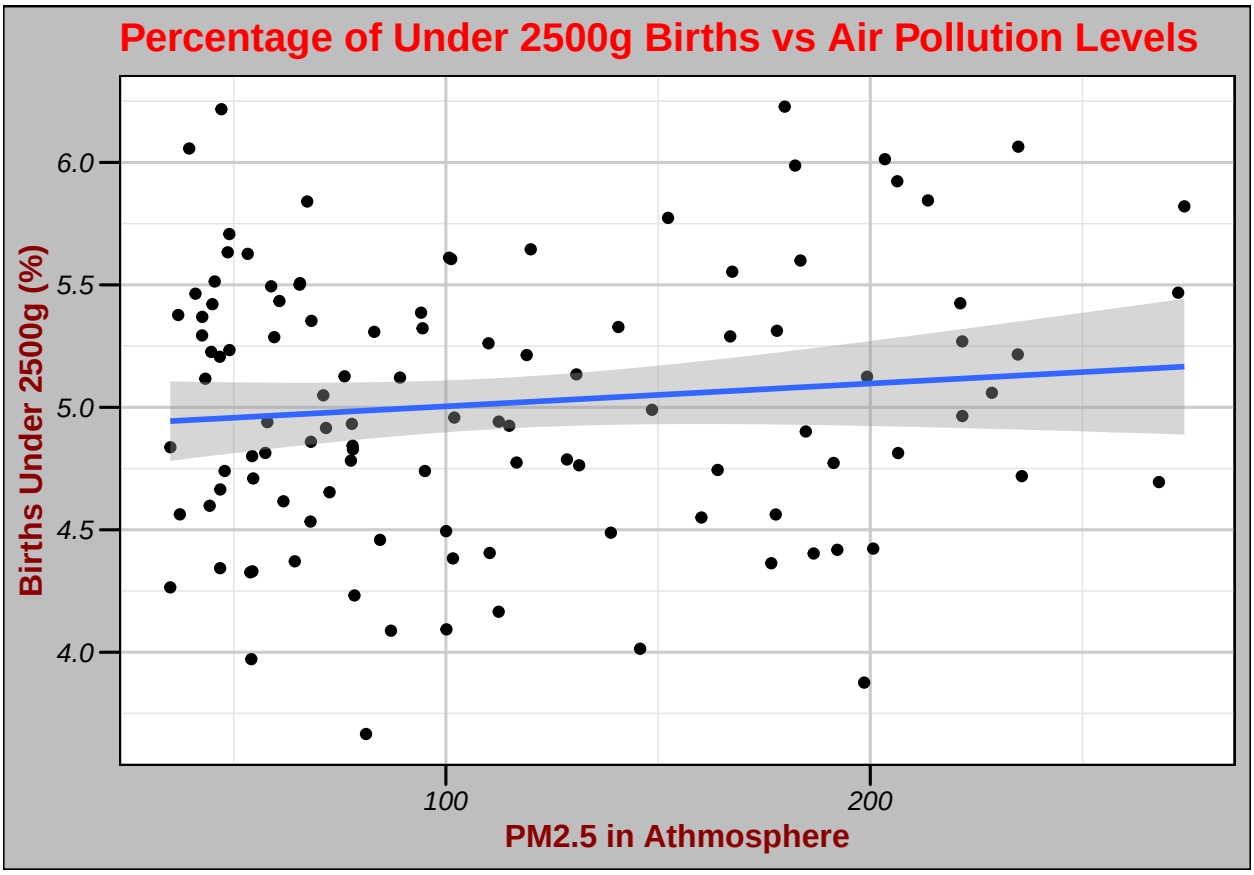
```r
#change column names and format of month (2016.01 to 2016-01)
colnames(Birth_Under_2500) <- colnames(Birth_Under_2500) %>%
  str_replace("^X", "") %>%       # delete x from colnames. read csv added x to every column, don't kno
  str_replace_all("\\.", "-")     # Change the format to 2016-01

#data is horizontal. change to vertical
Birth_Under_2500 <- Birth_Under_2500 %>%
  pivot_longer(
    cols = -1,  # first column includes aimag name Ulaanbaatar. don't take it.
    names_to = "Year_Month",
    values_to = "Birth.Weight.Under.2500"
  ) %>%
  select(Year_Month, Birth.Weight.Under.2500)

#create Date column in Birth Weight data in 2016-01-01 format
Birth_Under_2500 <- Birth_Under_2500 %>%
  mutate(Date = ym(`Year_Month`)) %>%
  select(Date, Birth.Weight.Under.2500)

#merge monthly dataframe with birth weight data with respect to Date columns
Ulaanbaatar_Monthly_Full <- Ulaanbaatar_Monthly_Full %>%
  left_join(Birth_Under_2500, by = "Date")

#read second csv. live births in Ulaanbaatar
Live_Births <- read.csv(here("Final/Data_Raw/LIVE BIRTHS.csv"))

#same procedure as before.
colnames(Live_Births) <- colnames(Live_Births) %>%
  str_replace("^X", "") %>%       # delete x from colnames. read csv added x to every column, don't kno
  str_replace_all("\\.", "-")     # Change the format to 2016-01

#data is horizontal. change to vertical
Live_Births <- Live_Births %>%
  pivot_longer(
    cols = -1,  # first column includes aimag name Ulaanbaatar. don't take it.
    names_to = "Year_Month",
    values_to = "Live.Births"
  ) %>%
  select(Year_Month, Live.Births)

#create Date column in Birth Weight data in 2016-01-01 format
Live_Births <- Live_Births %>%
  mutate(Date = ym(`Year_Month`)) %>%
  select(Date, Live.Births)

#merge monthly dataframe with live births data with respect to Date columns
Ulaanbaatar_Monthly_Full <- Ulaanbaatar_Monthly_Full %>%
  left_join(Live_Births, by = "Date")

#want to calculate percentage of birth weight under 2500 in all births. Live birth column include "," a
Ulaanbaatar_Monthly_Full$Live.Births <- gsub(",", "", Ulaanbaatar_Monthly_Full$Live.Births)

#change class of live births column to numeric to make mathematical calculation
```

```r
Ulaanbaatar_Monthly_Full$Live.Births <- as.numeric(Ulaanbaatar_Monthly_Full$Live.Births)

#create new column that is percentage of under 2500g births in total
Ulaanbaatar_Monthly_Full <- Ulaanbaatar_Monthly_Full %>%
  mutate(Under.2500.Rate = Birth.Weight.Under.2500 / Live.Births *100)

#plot under 2500g births by mean_AQI
ggplot(Ulaanbaatar_Monthly_Full, aes(x=mean_AQI, y=Under.2500.Rate)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Percentage of Under 2500g Births vs Air Pollution Levels",
       x = "PM2.5 in Athmosphere",
       y = "Births Under 2500g (%)") +
  theme_Ulaanbaatar
```

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

```
## Warning: Removed 2 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



```r
#correlation between births under 2500g and air pollution
Reg_2500g_AQI <- lm(Under.2500.Rate ~ mean_AQI, data = Ulaanbaatar_Monthly_Full)
summary(Reg_2500g_AQI)
```

```
##
## Call:
## lm(formula = Under.2500.Rate ~ mean_AQI, data = Ulaanbaatar_Monthly_Full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3204 -0.3756 -0.0552  0.4110  1.2625
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.9109299  0.1050036  46.769   <2e-16 ***
## mean_AQI    0.0009314  0.0008052   1.157     0.25
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5502 on 108 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.01224,    Adjusted R-squared:  0.003091
## F-statistic: 1.338 on 1 and 108 DF,  p-value: 0.2499
```

```r
#according to the regression analysis, births under 2500g ratio in all births is not correlated with me
```

xyz