Final_Start

2025-04-16

CSV Dosyaları Okuma

```
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)
library(agricolae)
library(zoo)
```

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

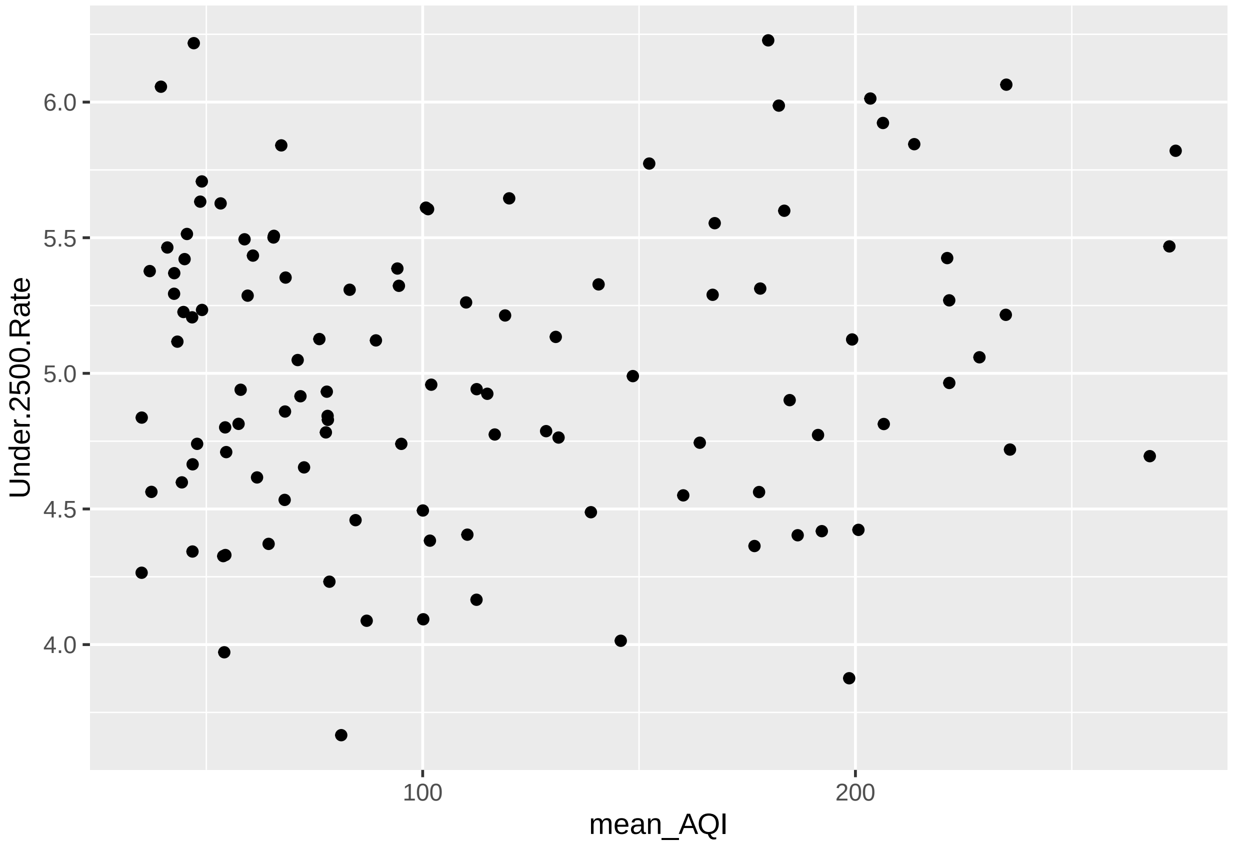
```
#read all files
Ulanbator_2015 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2015_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2016 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2016_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2017 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2017_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2018 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2018_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2019 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2019_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2020 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2020_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2021 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2021_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2022 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2022_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2023 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2023_YTD.csv"), stringsAsFactors = TRU
```

```r
Ulanbator_2024 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2024_YTD.csv"), stringsAsFactors = TRU
Ulanbator_2025 <- read.csv(here("Final/Data_Raw/Ulaanbaatar_PM2.5_2025_YTD.csv"), stringsAsFactors = TRU

#merge files into one file
Ulanbator_PM2.5 <- bind_rows(Ulanbator_2015,Ulanbator_2016,Ulanbator_2017,Ulanbator_2018,Ulanbator_2019

#remove yearly data from environment if wanted
#rm(Ulanbator_2015,Ulanbator_2016,Ulanbator_2017,Ulanbator_2018,Ulanbator_2019,Ulanbator_2020,Ulanbator

#clean -999 AQI values, in order to prevent failure in mean calculations
Ulanbator_clean <- Ulanbator_PM2.5 %>% filter(AQI != -999)

#create monthly data by taking mean of every month. because various health data are monthly
Ulanbator_monthly <- Ulanbator_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 new dataset to include every month from 2015-11 to 2025-02
Ulanbator_monthly_full <- Ulanbator_monthly %>%
  mutate(Year = as.integer(substr(Year_Month, 1, 4)),
         Month = as.integer(substr(Year_Month, 6,7))) %>%
  select(Year,Month,mean_AQI)


#create Date column in this full months dataset
Ulanbator_monthly_full <- Ulanbator_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
date_range <- seq(
  from = min(Ulanbator_monthly_full$Date),
  to   = max(Ulanbator_monthly_full$Date),
  by = "month"
)

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

#Main dataframe is created here. including Date, and mean AQI values. AQI is na for missing months.
Ulanbator <- data.frame(
  Date = date_range
) %>%
  mutate(
    Year = year(Date),
    Month = month(Date)
  ) %>%
  left_join(Ulanbator_monthly_full, by = c("Year", "Month"))


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


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

#change column names
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 main dataframe with birth weight data with respect to Date columns
Ulanbator <- Ulanbator %>%
  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 with main dataframe
Ulanbator <- Ulanbator %>%
  left_join(Live_Births, by = "Date")


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

#change class of live births column to numeric to make mathematical calculation
Ulanbator$Live.Births <- as.numeric(Ulanbator$Live.Births)

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


#plot under 2500g births by mean_AQI
ggplot(Ulanbator, aes(x=mean_AQI, y=Under.2500.Rate)) +
  geom_point()
```

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

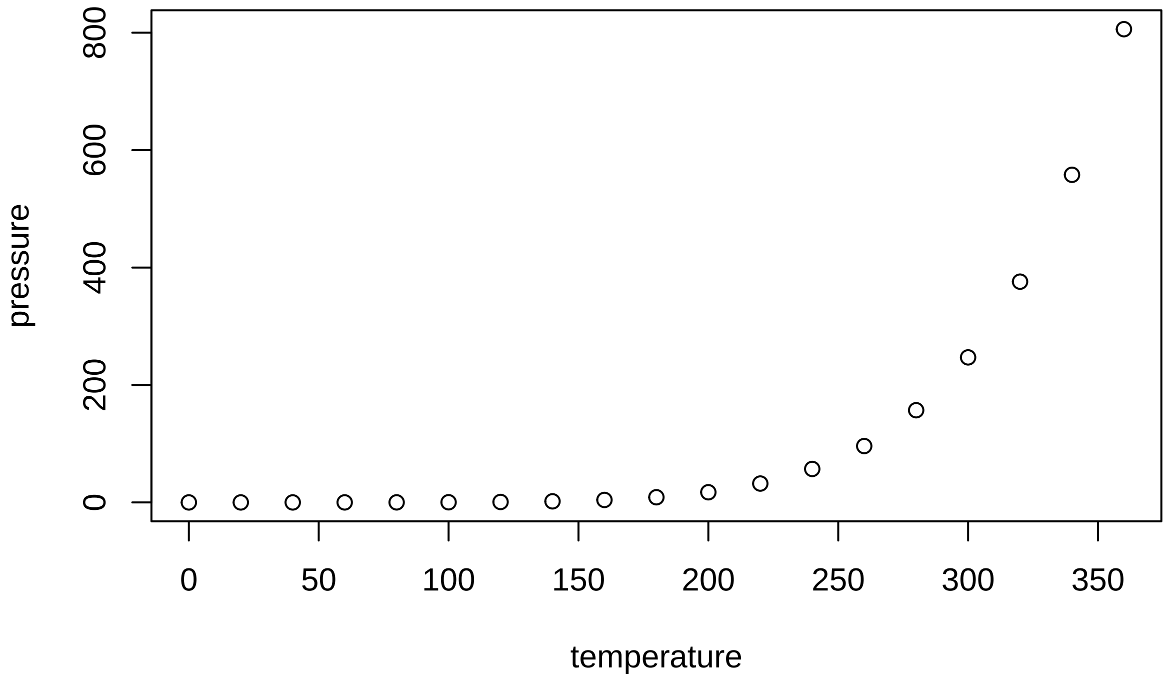


```
#regression example between under 2500g births and mean_AQI
regresyon_deneme <- lm(data = Ulanbator, Under.2500.Rate ~ mean_AQI) #linear regression of temperature

#no relation
summary(regresyon_deneme)
```

```
##
## Call:
## lm(formula = Under.2500.Rate ~ mean_AQI, data = Ulanbator)
##
## 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
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.