# Boston Buoy Data Analysis

Congyao Duan

# 1.Instrution

## My understanding of the Question

Over the past 50 years, the average global temperature has increased at the fastest rate in recorded history. Global warming occurs when carbon dioxide (CO2) and other air pollutants and greenhouse gases collect in the atmosphere and absorb sunlight and solar radiation that have bounced off the earth's surface. Normally, this radiation would escape into space—but these pollutants, which can last for years to centuries in the atmosphere, trap the heat and cause the planet to get hotter. The perpose of our project is to find out if there is evidence of global warming in the data collected by a single weather buoy in the NOAA National Data Buoy Center.

## My Aproach

First, I use R to clean and select the raw data.I chose ATMP and WTMP for analysis. Because these two sets of temperature data best reflect the related trends of global warming.I took the 12 o 'clock data every day, it can reduce the number of observations in my dataset without reducing the amount of information in the data about the climate change question. Secondly, I use R to analysis the data. I do some exploratory research to get familiar with this data， but there is no obvious trend. Then I use linear regression to analyze the data.

## How I orgnize the work

I use R to clean and select the raw data.Then use R to analysis the data and draw a conclution.

# 2.Data Cleaning and Organizing

There are a lot of outliers in this data, and a lot of variables that are not needed for this project. So do the data cleaning and processing before starting the project.In this part, I cleaned up the outliers and removed the variables that were not needed,leave only useful date and temperature data (ATMP&WTMP).I chose ATMP and WTMP for analysis. Because these two sets of temperature data best reflect the related trends of global warming.I took the 12 o 'clock data every day, it can reduce the number of observations in my dataset without reducing the amount of information in the data about the climate change question.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## √ ggplot2 3.3.2     √ purrr   0.3.4
## √ tibble  3.0.3     √ dplyr   1.0.2
## √ tidyr   1.1.2     √ stringr 1.4.0
## √ readr   1.3.1     √ forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(stringr)
library(rstanarm)
```

```
## Loading required package: Rcpp
```

```
## This is rstanarm version 2.21.1
```

```
## - See https://mc-stan.org/rstanarm/articles/priors for changes to default prior
s!
```

```
## - Default priors may change, so it's safest to specify priors, even if equivalen
t to the defaults.
```

```
## - For execution on a local, multicore CPU with excess RAM we recommend calling
```

```
##   options(mc.cores = parallel::detectCores())
```

```r
url_1 <- "http://www.ndbc.noaa.gov/view_text_file.php?filename=mlrf1h"
url_2 <- ".txt.gz&dir=data/historical/stdmet/"
years <- c(1999:2018)
urls <- str_c(url_1, years, url_2, sep = "")
filenames <- str_c("mr", years, sep = "")


# Year 1999 - 2006
for(i in 1:8){
  suppressMessages(
    # Fill any missing values with NA:
    assign(filenames[i], read.table(urls[i], header = TRUE, fill = TRUE))
  )

}
# Year 2007 - 2018
for(i in 9:20){
  suppressMessages(
    # Fill any missing values with NA and use the same column names as year 2006
    assign(filenames[i], read.table(urls[i], header = FALSE,
                                    fill = TRUE, col.names = colnames(mr2006))),
  )

}
```

```r
#use loops to get rid of the excess columns
mr1999$TIDE <- NA
n <- length(urls)
for (i in 1:n){
  file <- get(filenames[i])
  colnames(file)[1] <-"YYYY"
  if(ncol(file) == 18){
    file <- subset(file, select = -mm )
  }
  if(i == 1){
    MR <- file
  }else{
    MR <- rbind.data.frame(MR, file)
  }

}
```

```r
#Get rid of the excess columns
i<-2005
repeat {
  assign(paste("mr",as.character(i),sep=""),
         get(paste("mr",as.character(i),sep=""))[,-5])
  i=i+1
  if(i>2018)
  {break}
}
```

```r
#screen out data at 12 o'clock
i<-1999
repeat {
  assign(paste("mr",as.character(i),sep=""),get(paste("mr",as.character(i),sep=""))
[which(get(paste("mr",as.character(i),sep=""))$hh == 12), ])
  i=i+1
  if(i>2018)
  {break}
}
```

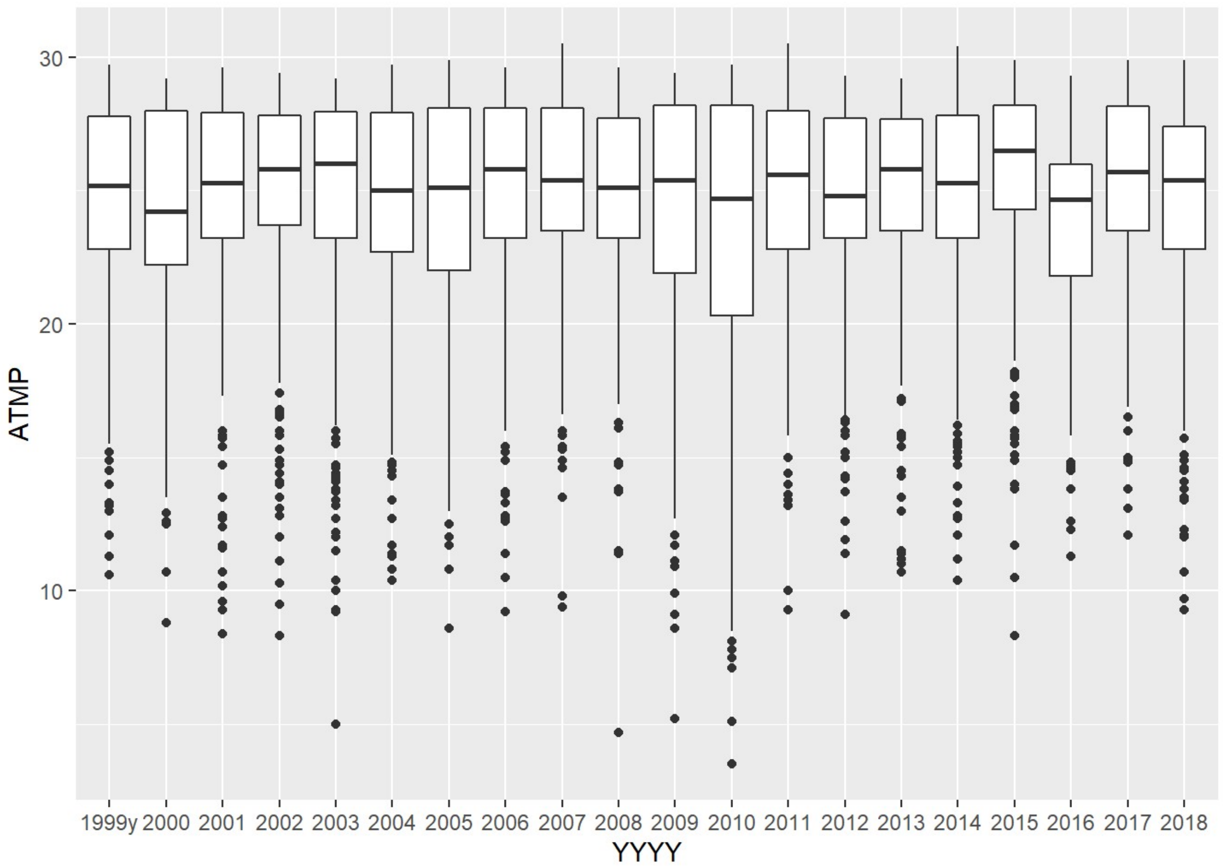
```r
#Find out outliers
mr1999$TIDE <- NA
n <- 20
for (i in 1:n){
  file <- get(filenames[i])
  colnames(file)[1] <-"YYYY"
  if(i == 1){
    MRC <- file
  }else{
    MRC <- rbind.data.frame(MRC, file)
  }

}

#Get rid of the useless columns
MRC<-MRC[c(1,2,3,13,14)]
```

```r
#clean up abnormal data
MRC$ATMP[which(MRC$ATMP>=100)]=NA
MRC$WTMP[which(MRC$WTMP>=100)]=NA
MRC=na.omit(MRC)
```

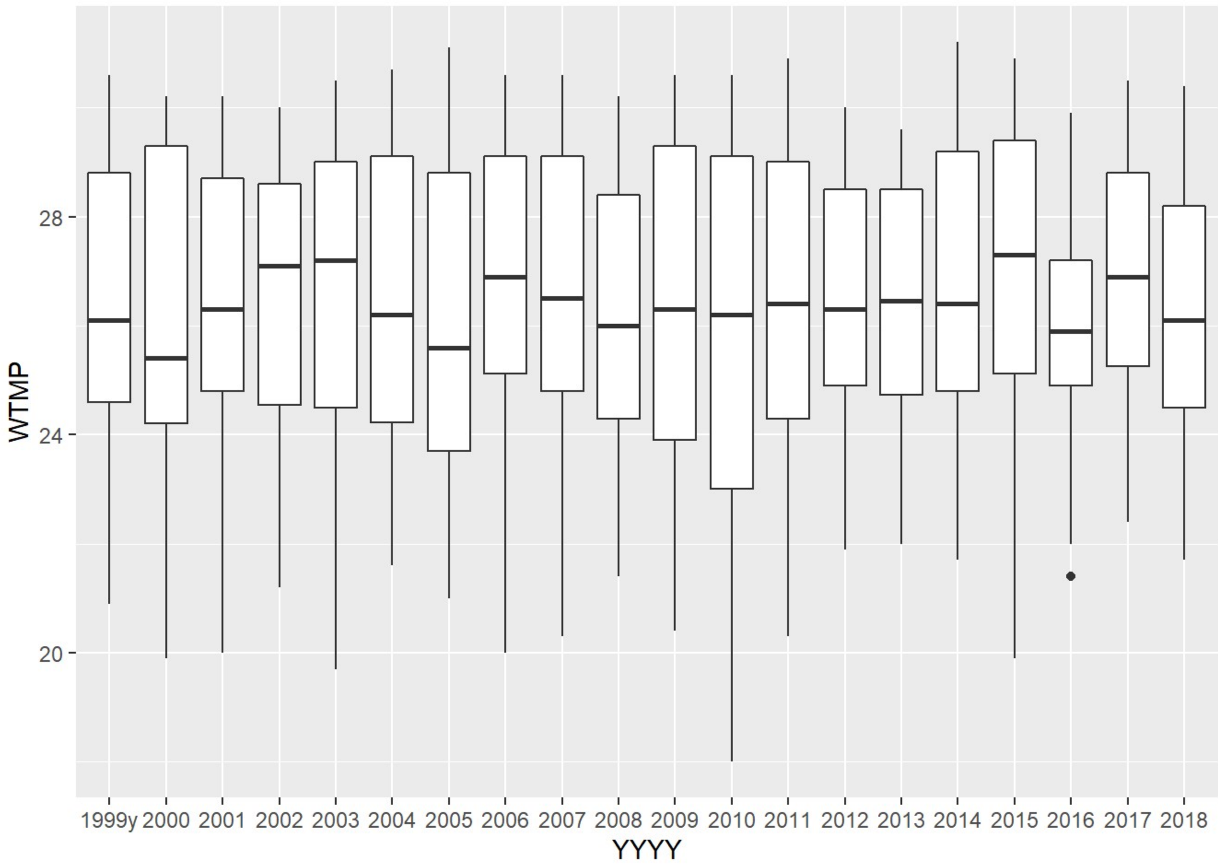
# 3.Exploratory Data Analysis(EDA)

In order to see the distruction of the data clearly,I draw boxplots and violin plots to observe first.Boxplot can not only reflect the distribution characteristics of the original data, but also compare the distribution characteristics of multiple groups of data. It can find the maximum, minimum, median and two quartiles of the data. A boxplot can also be used to find outliers and thus process outliers in data.Violin plots are used to show the distribution and probability density of multiple sets of data. This chart combines the features of box and density charts and is mainly used to show the distribution shape of the data. Similar to the box diagram, but better displayed at the density level.

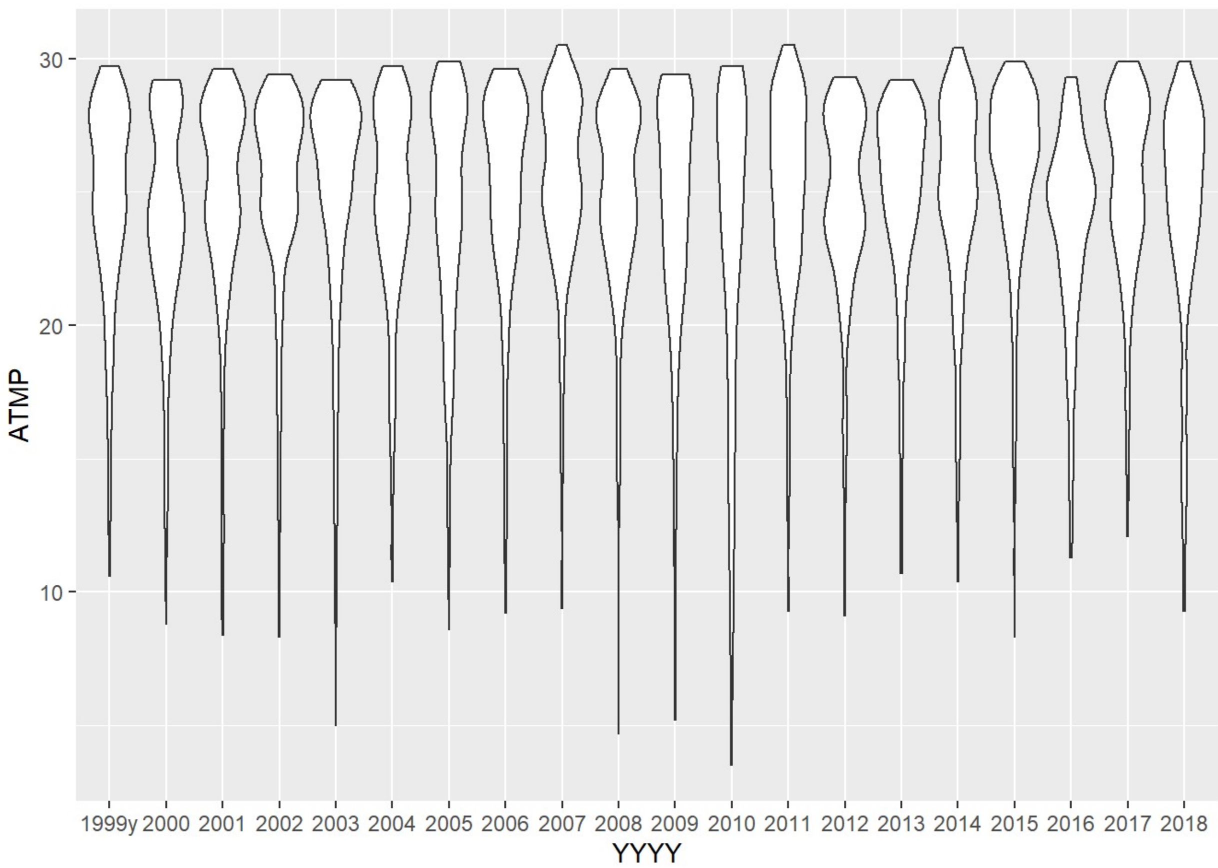
```
#Draw boxplots
MRC$YYYY[which(MRC$YYYY==1999)]="1999y"
ggplot(data=MRC,aes(x=YYYY,y=ATMP))+geom_boxplot()
```



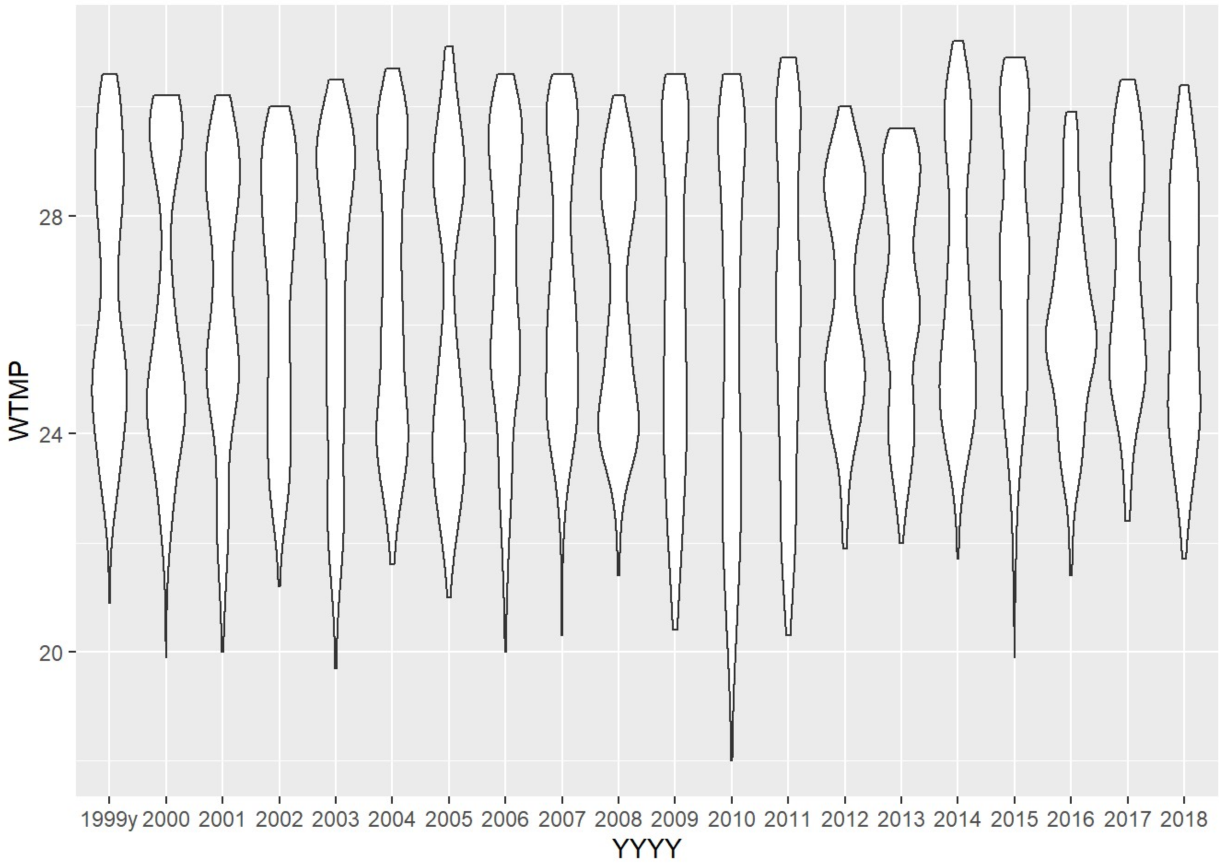

```
ggplot(data=MRC,aes(x=YYYY,y=WTMP))+geom_boxplot()
```




```
#Draw violin plots
ggplot(data=MRC,aes(x=YYYY,y=ATMP))+geom_violin()
```



```
ggplot(data=MRC,aes(x=YYYY,y=WTMP))+geom_violin()
```




By observing the boxplot and violin plot, it was found that no obvious temperature trend could be observed. There was no significant trend in the median, maximum and minimum values. The data fluctuated, and the graph and the distribution of the data did not reveal the relationship between temperature and year.In order to find the relationship between temperature and time, we need to do further research.

# 4.Method

Linear regression can determine the quantitative relationship of interdependence between two or more variables. The density of linear regression can draw a trend line, which represents the long-term trend of time series data. It tells us whether a particular set of data has increased or decreased over a period of time. Although we can visually observe the position of the data points in the coordinate system to roughly draw the trend line, it is more appropriate to use linear regression to calculate the position and slope of the trend line.

```
#draw scatter diagram and regression lines of the average temperature
datamean=group_by(MRC,YYYY)%>%summarize_each(funs(mean))
```

```
## Warning: `summarise_each_()` is deprecated as of dplyr 0.7.0.
## Please use `across()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
## Warning: `funs()` is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
x=c(1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010,2011,2012,2013,201
4,2015,2016,2017,2018)
y<-datamean$ATMP
z<-datamean$WTMP
fit_1<-stan_glm(y~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

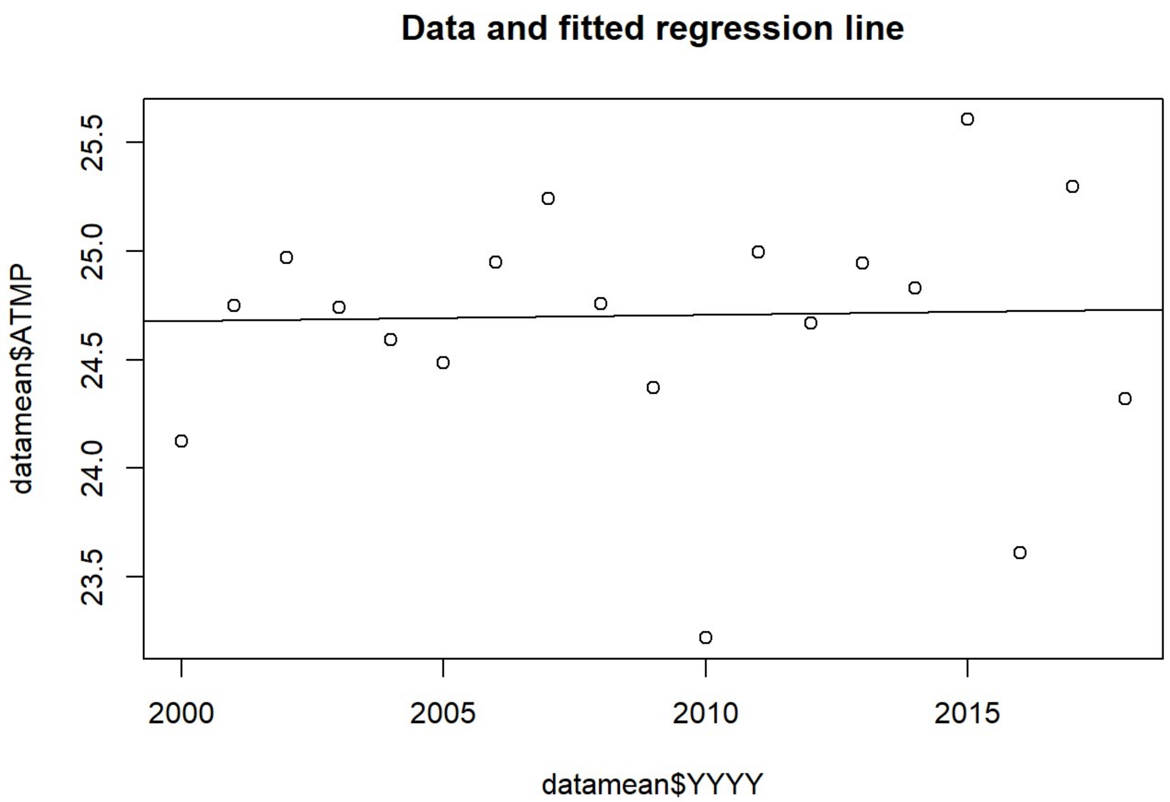
```
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1: 
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1: 
## Chain 1: 
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1: 
## Chain 1:  Elapsed Time: 0.055 seconds (Warm-up)
## Chain 1:                0.049 seconds (Sampling)
## Chain 1:                0.104 seconds (Total)
## Chain 1: 
## 
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2: 
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2: 
## Chain 2: 
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2: 
## Chain 2:  Elapsed Time: 0.072 seconds (Warm-up)
## Chain 2:                0.062 seconds (Sampling)
## Chain 2:                0.134 seconds (Total)
## Chain 2: 
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.062 seconds (Warm-up)
## Chain 3:                0.038 seconds (Sampling)
## Chain 3:                0.1 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.065 seconds (Warm-up)
## Chain 4:                0.055 seconds (Sampling)
## Chain 4:                0.12 seconds (Total)
## Chain 4:
```

```
plot(datamean$YYYY, datamean$ATMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

```
abline(fit_1)
```




```
print(fit_1)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      y ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) 18.6   44.3  
## x            0.0    0.0  
## 
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.6    0.1   
## 
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_1)
```

```
##  (Intercept)           x 
## 18.648593323  0.003014432
```

```
fit_2<-stan_glm(z~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

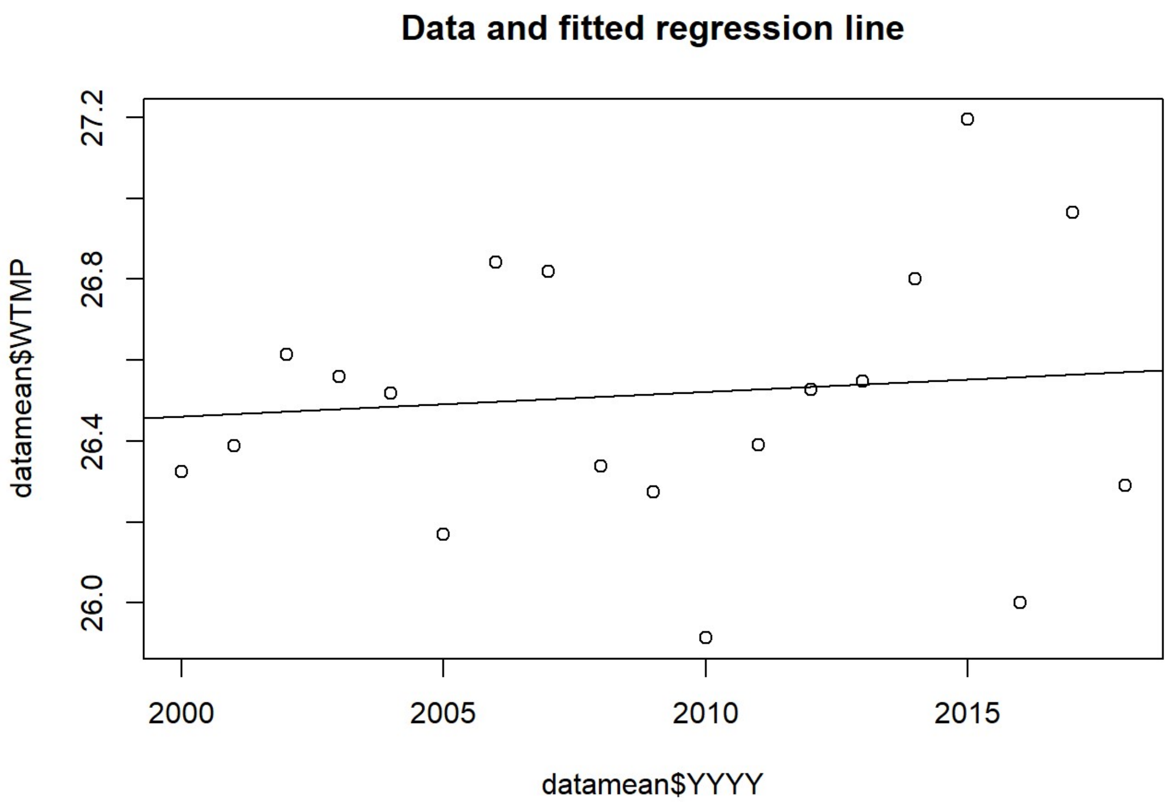
```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.064 seconds (Warm-up)
## Chain 1:                0.055 seconds (Sampling)
## Chain 1:                0.119 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.043 seconds (Warm-up)
## Chain 2:                0.052 seconds (Sampling)
## Chain 2:                0.095 seconds (Total)
## Chain 2:
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.061 seconds (Warm-up)
## Chain 3:                0.058 seconds (Sampling)
## Chain 3:                0.119 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.04 seconds (Warm-up)
## Chain 4:                0.069 seconds (Sampling)
## Chain 4:                0.109 seconds (Total)
## Chain 4:
```

```
plot(datamean$YYYY, datamean$WTMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

```
abline(fit_2)
```




```
print(fit_2)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      z ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) 14.4   25.2  
## x            0.0    0.0  
## 
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.3    0.1   
## 
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_2)
```

```
## (Intercept)           x 
## 14.39179307  0.00603502
```

First, we calculate the average temperature of each year and draw a scatter plot of the average. And make linear regression model.The regression equation with one variable is obtained:

```
            ATMP(mean)=18.181+0.0032*year+error
            WTMP(mean)=14.475+0.00597*year+error
```

Then we draw the regression line in the scatter plot.

```
#Select the highest temperature of the year(12 o'clock)
#draw scatter diagram and regression lines of the max temperature
datamax=group_by(MRC,YYYY)%>%summarize_each(funs(max))
p<-datamax$ATMP
plot(datamax$YYYY, datamax$ATMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

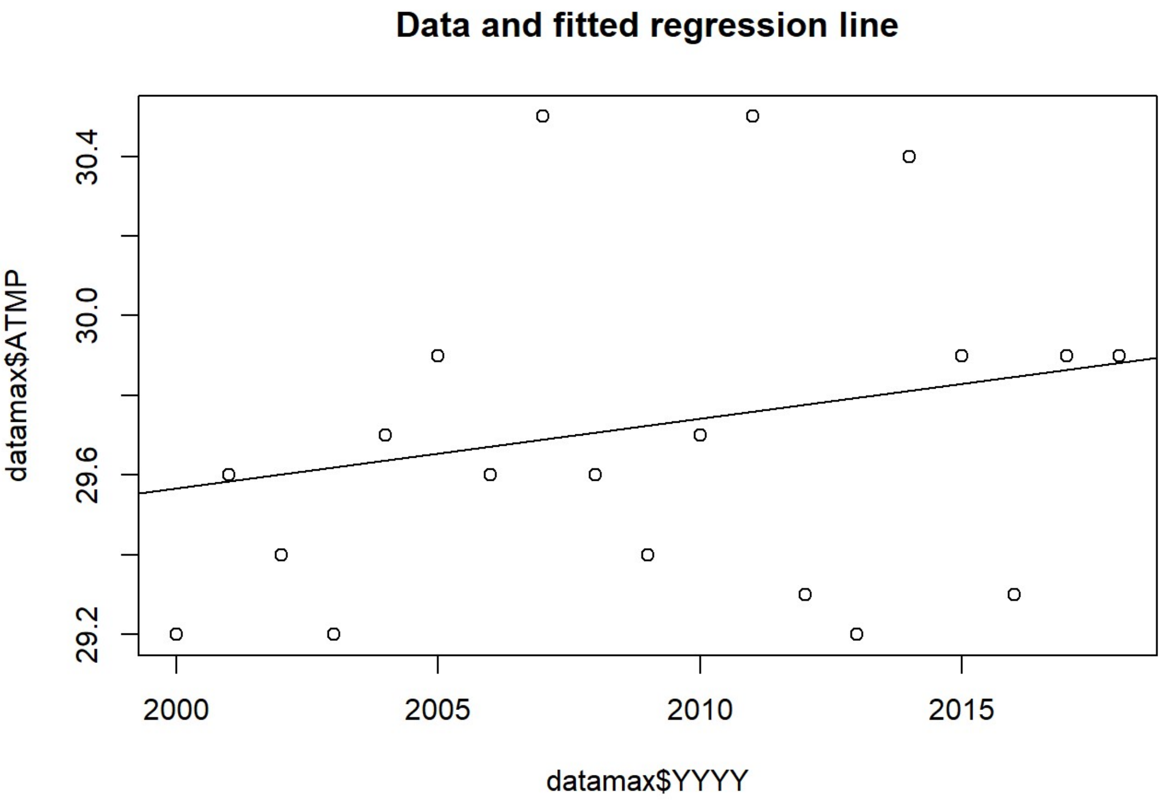
```
fit_3<-stan_glm(p~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.071 seconds (Warm-up)
## Chain 1:                0.064 seconds (Sampling)
## Chain 1:                0.135 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.059 seconds (Warm-up)
## Chain 2:                0.041 seconds (Sampling)
## Chain 2:                0.1 seconds (Total)
## Chain 2:
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.048 seconds (Warm-up)
## Chain 3:                0.058 seconds (Sampling)
## Chain 3:                0.106 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.053 seconds (Warm-up)
## Chain 4:                0.071 seconds (Sampling)
## Chain 4:                0.124 seconds (Total)
## Chain 4:
```

```
abline(fit_3)
```




```
print(fit_3)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      p ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) -5.5   30.7
## x            0.0    0.0
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.4    0.1
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_3)
```

```
## (Intercept)           x
## -5.45309029  0.01751017
```

```
q<-datamax$WTMP
plot(datamax$YYYY, datamax$WTMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

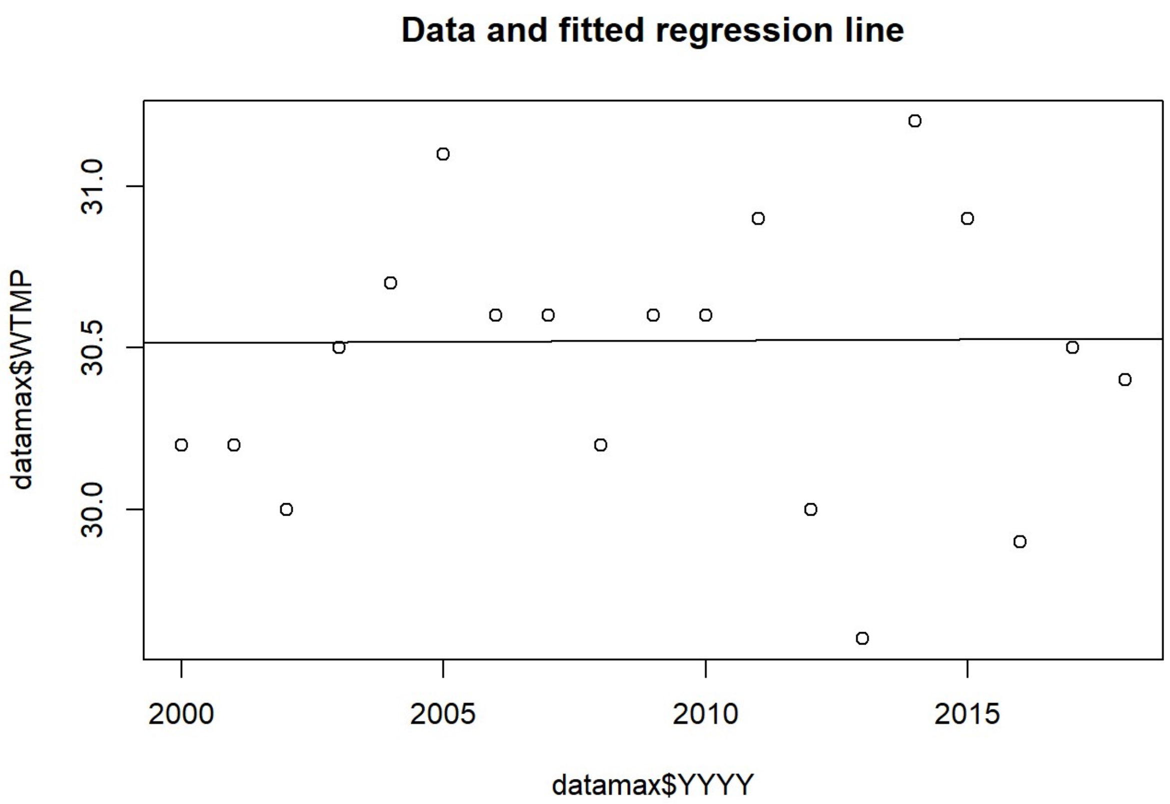
```
fit_4<-stan_glm(q~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.063 seconds (Warm-up)
## Chain 1:                0.036 seconds (Sampling)
## Chain 1:                0.099 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.066 seconds (Warm-up)
## Chain 2:                0.062 seconds (Sampling)
## Chain 2:                0.128 seconds (Total)
## Chain 2:
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.053 seconds (Warm-up)
## Chain 3:                0.05 seconds (Sampling)
## Chain 3:                0.103 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.051 seconds (Warm-up)
## Chain 4:                0.047 seconds (Sampling)
## Chain 4:                0.098 seconds (Total)
## Chain 4:
```

```
abline(fit_4)
```




```
print(fit_4)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      q ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) 28.9   33.3  
## x            0.0    0.0  
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.4    0.1   
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_4)
```

```
##  (Intercept)           x
## 2.892727e+01 7.935994e-04
```

Second , we calculate the highest temperature of each year and draw a scatter plot of the max. And make linear regression model.The regression equation with one variable is obtained：

```
             ATMP(max)=-6.975+0.01828*year+error
             WTMP(max)=28.8025+0.00087*year+error
```

Then we draw the regression line in the scatter plot.

```
#Select the lowest temperature of the year(12 o'clock)
#draw scatter diagram and regression lines of the min temperature
datamin=group_by(MRC,YYYY)%>%summarize_each(funs(min))
r<-datamin$ATMP
plot(datamin$YYYY, datamin$ATMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

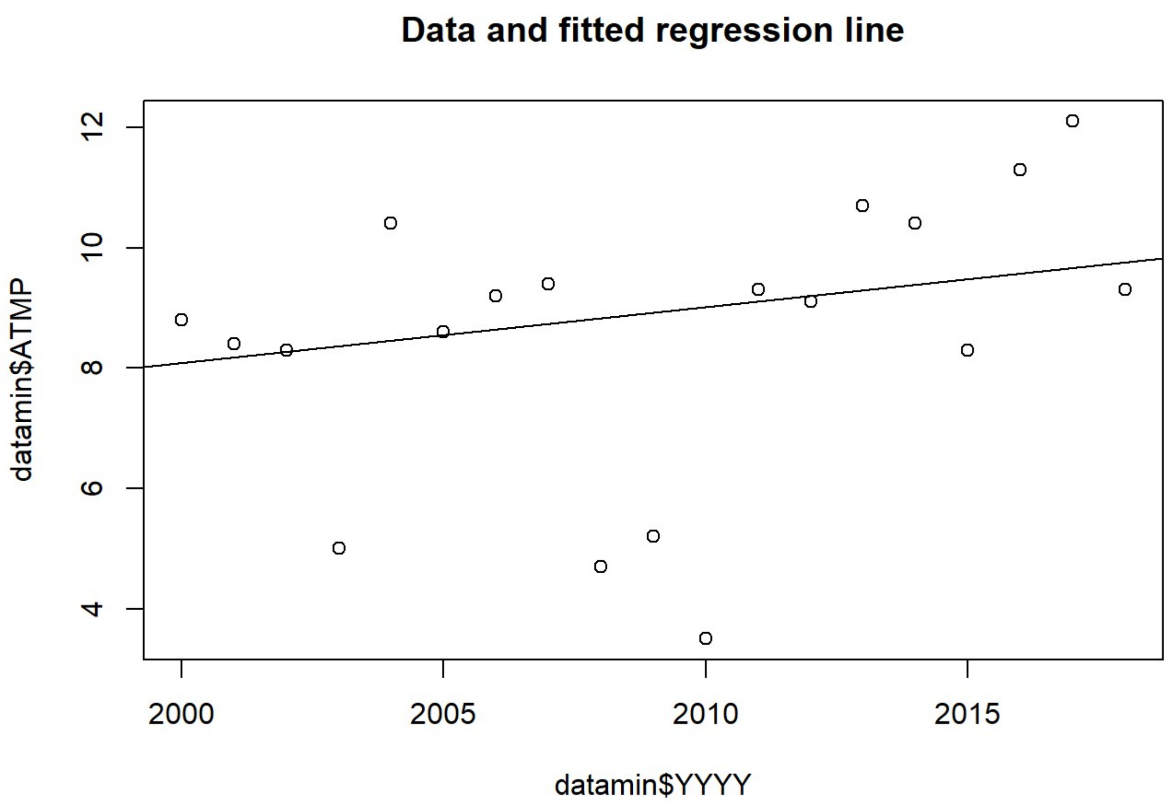
```
fit_5<-stan_glm(r~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.055 seconds (Warm-up)
## Chain 1:                0.057 seconds (Sampling)
## Chain 1:                0.112 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.053 seconds (Warm-up)
## Chain 2:                0.053 seconds (Sampling)
## Chain 2:                0.106 seconds (Total)
## Chain 2:
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.052 seconds (Warm-up)
## Chain 3:                0.066 seconds (Sampling)
## Chain 3:                0.118 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.055 seconds (Warm-up)
## Chain 4:                0.053 seconds (Sampling)
## Chain 4:                0.108 seconds (Total)
## Chain 4:
```

```
abline(fit_5)
```




```
print(fit_5)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      r ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) -176.7  185.2
## x              0.1    0.1
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 2.4    0.4
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_5)
```

```
##   (Intercept)             x
## -176.71343782    0.09240129
```

```
s<-datamin$WTMP
plot(datamin$YYYY, datamin$WTMP, main="Data and fitted regression line")
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): 强制改变过程中产生了NA
```

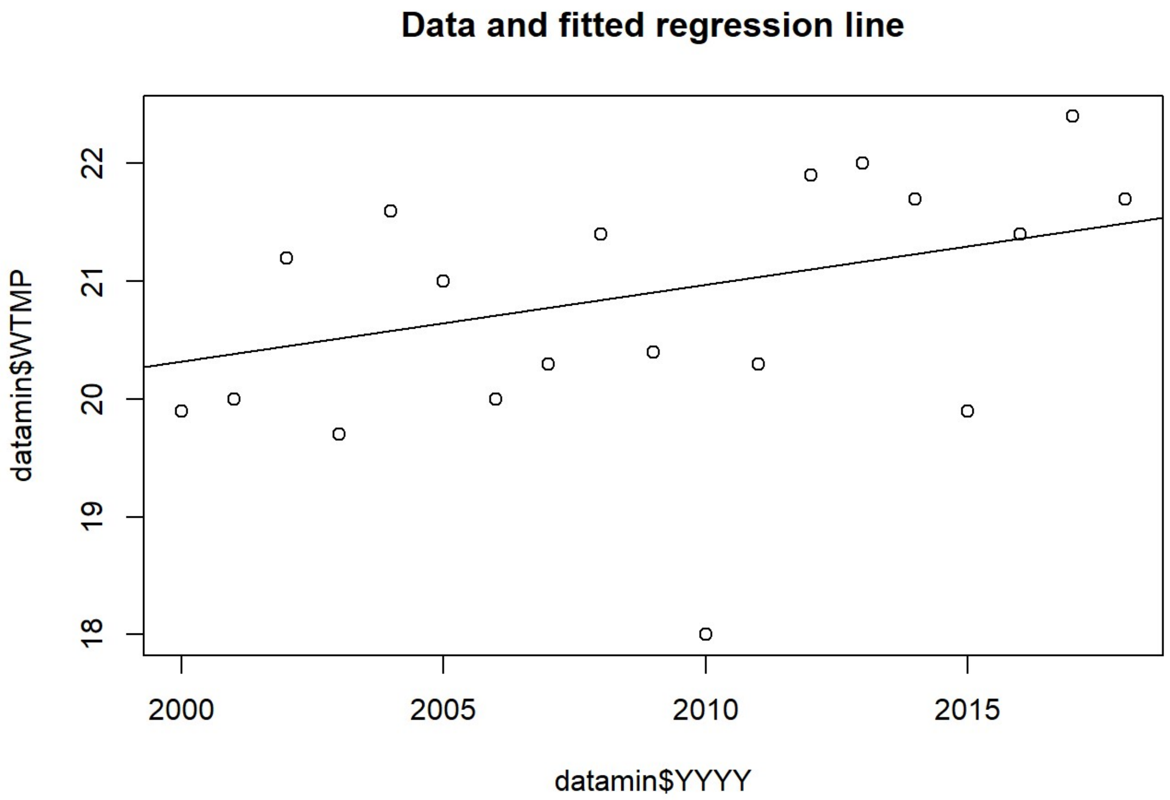
```
fit_6<-stan_glm(s~x)
```

```
## Warning: Omitting the 'data' argument is not recommended and may not be allowed
## in future versions of rstanarm. Some post-estimation functions (in particular
## 'update', 'loo', 'kfold') are not guaranteed to work properly unless 'data' is
## specified as a data frame.
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 0.05 seconds (Warm-up)
## Chain 1:                0.054 seconds (Sampling)
## Chain 1:                0.104 seconds (Total)
## Chain 1:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 2).
## Chain 2:
## Chain 2: Gradient evaluation took 0 seconds
## Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 2: Adjust your expectations accordingly!
## Chain 2:
## Chain 2:
## Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 2:
## Chain 2:  Elapsed Time: 0.061 seconds (Warm-up)
## Chain 2:                0.06 seconds (Sampling)
## Chain 2:                0.121 seconds (Total)
## Chain 2:
```

```
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 3).
## Chain 3:
## Chain 3: Gradient evaluation took 0 seconds
## Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 3: Adjust your expectations accordingly!
## Chain 3:
## Chain 3:
## Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 3:
## Chain 3:  Elapsed Time: 0.04 seconds (Warm-up)
## Chain 3:                0.068 seconds (Sampling)
## Chain 3:                0.108 seconds (Total)
## Chain 3:
##
## SAMPLING FOR MODEL 'continuous' NOW (CHAIN 4).
## Chain 4:
## Chain 4: Gradient evaluation took 0 seconds
## Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0 se
conds.
## Chain 4: Adjust your expectations accordingly!
## Chain 4:
## Chain 4:
## Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
## Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
## Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
## Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
## Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
## Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
## Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
## Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
## Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
## Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
## Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
## Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
## Chain 4:
## Chain 4:  Elapsed Time: 0.055 seconds (Warm-up)
## Chain 4:                0.065 seconds (Sampling)
## Chain 4:                0.12 seconds (Total)
## Chain 4:
```

```
abline(fit_6)
```




```
print(fit_6)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      s ~ x
##  observations: 20
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept) -110.4   79.9
## x              0.1    0.0
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 1.0    0.2
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
coef(fit_6)
```

```
##   (Intercept)            x
## -110.36609977   0.06534153
```

At last , we calculate the lowest temperature of each year and draw a scatter plot of the min. And make linear regression model.The regression equation with one variable is obtained：

```
        ATMP(min)=-177.5036+0.0927*year+error
        WTMP(min)=-108.5465+0.0643*year+error
```

Then we draw the regression line in the scatter plot.

# 5.Discussion of the Problem(My conclution)

Through the above studies, we have the boxplot/violin plots.And found their linear regression relationship:

```
        ATMP(mean)=18.181+0.0032*year+error
        WTMP(mean)=14.475+0.00597*year+error
        ATMP(max)=-6.975+0.01828*year+error
        WTMP(max)=28.8025+0.00087*year+error
        ATMP(min)=-177.5036+0.0927*year+error
        WTMP(min)=-108.5465+0.0643*year+error
```

We can find out that the temperature has a tendency to increase over time, but it’s not obvious.I don’t think that this is a result that tells you that temperature is related to the realization of the year, or that the temperature is going to increase over time in just 20 years. I think a time series model would be more appropriate for these data (but I haven’t studied it).Or maybe we should take a longer time frame. And the study can 10 years as a unit of time rather than a year,because temperature change is a very slow process.In this way, the relationship between temperature and time will be more significant, which is convenient for us to study and get a clear conclusion.

# 6.Reference

package:https://cran.r-project.org/web/packages/citation/index.html (https://cran.r-project.org/web/packages/citation/index.html) links：https://www.cnblogs.com/sylvanas2012/p/4328861.html (https://www.cnblogs.com/sylvanas2012/p/4328861.html) https://www.ndbc.noaa.gov/station_page.php?station=44013 (https://www.ndbc.noaa.gov/station_page.php?station=44013) http://www.mamicode.com/info-detail-1662071.html (http://www.mamicode.com/info-detail-1662071.html) https://bbs.pinggu.org/forum.php?mod=viewthread&tid=6612706 (https://bbs.pinggu.org/forum.php?mod=viewthread&tid=6612706)