Buoy_report

GC

2020/9/26

###Boston Buoy Data Analysis Report ##Research Objective: Is there evidence of global warming in the data collected by a single weather buoy in the NOAA National Data Buoy Center? The Boston Buoy Data Analysis project requires you to complete a project in its entirety from acquiring data to presenting results. As you work, keep the goals and vision for the completed work in perspective.

##My understanding of the project: To find the evidence of global warming, we should pay attention to the temperature trend over the last two decades. More specifically, in the Buoy data analysis project, air temperature(ATMP) and water temperature(WTMP) data could display the trend of warming.

##My approach: Using R to complete the whole project. First of all, I imported data ranging from 1999 to 2018 from website. Besides, to form an available data set, I used some tidying functions. A new "NA"column was added to data of 1999, which solely had 16 variables at that year. In addtion, the 18th column of data from 2005 to 2018 was deleted and I got data frame "MR". After that, I keeped varibles about tempreture and date and transformed those date varibales to POSIX type. Finally, I obtained the data frame "ND" and did relevent analysis.

###How I organized my work: I organize my work as following steps(containing the codes and plots):

##import data from NOAA

```
library(tidyverse)
```

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

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

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

```
library(stringr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
url_1 <- "http://www.ndbc.noaa.gov/view_text_file.php?filename=mlrf1h"
url_1
```

```
## [1] "http://www.ndbc.noaa.gov/view_text_file.php?filename=mlrf1h"
```

```r
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 = "")
###  Read the data from the website
```

```r
# Year 1999 - 2006
for(i in 1:8){
  suppressMessages(
    assign(filenames[i], read.table(urls[i], header = TRUE, fill = TRUE))
  )
}
```

```r
# Year 2007 - 2018
for(i in 9:20){
 suppressMessages(
 assign(filenames[i], read.table(urls[i], header = FALSE,
  fill = TRUE, col.names = colnames(mr2006))),
 )
}
```

#add a new “NA” column to 1999, delete the extra column “mm” and bind data from 1999 to 2018 to a dataframe “MR”

```r
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)
  }
}
```

##replace outliers in “ATMP” and “WTMP” with “NA” ##delete variables I don’t choose and form a new data frame

```r
library(lubridate)
as_tibble(MR)
```

```
## # A tibble: 162,695 x 17
##     YYYY    MM    DD    hh    WD  WSPD   GST  WVHT   DPD   APD   MWD   BAR  ATMP
##    <int> <int> <int> <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl> <int> <dbl> <dbl>
##  1  1999     1     1     0    83   7.2   7.8    99    99    99   999 1019.  23.3
##  2  1999     1     1     1    78   6.4   7.4    99    99    99   999 1019.  23.4
##  3  1999     1     1     2    68   7.2   7.9    99    99    99   999 1019.  23.4
##  4  1999     1     1     3    77   8.2   9.2    99    99    99   999 1019.  23.5
##  5  1999     1     1     4    75   8.4   9.2    99    99    99   999 1019.  23.6
##  6  1999     1     1     5    64   7.9   8.9    99    99    99   999 1019.  23.5
##  7  1999     1     1     6    84   7.7   8.6    99    99    99   999 1019   23.6
##  8  1999     1     1     7    81   7.9   9.3    99    99    99   999 1018.  23.5
```

```
## 9  1999     1     1     8    84    7.9    8.7    99    99    99   999 1018.  23.5
## 10  1999     1     1     9    74    8.3    9      99    99    99   999 1018.  23.6
## # ... with 162,685 more rows, and 4 more variables: WTMP <dbl>, DEWP <dbl>,
## #   VIS <dbl>, TIDE <dbl>
```

```r
MR$ATMP[MR$ATMP==999]<-NA
MR$WTMP[MR$WTMP==999]<-NA
Mydata<-names(MR)%in% c("WD","WSPD","GST","WVHT","DPD","APD","MWD","BAR","DEWP","VIS","TIDE")
NewData1<-MR[!Mydata]
View(NewData1)
```

##use lubridate to transform the date-time data into posix numbers

```r
# changing data/time data
NewData<-NewData1%>%mutate(DATETIME=make_datetime(YYYY,MM,DD,hh))
ND1<-NewData[,5:7]
colnames(ND1)
```

```
## [1] "ATMP"     "WTMP"     "DATETIME"
```

```r
ND<-ND1[, c(3, 1, 2)]
View(ND)
tibble(ND)
```

```
## # A tibble: 162,695 x 3
##    DATETIME             ATMP  WTMP
##    <dttm>              <dbl> <dbl>
##  1 1999-01-01 00:00:00  23.3  25.7
##  2 1999-01-01 01:00:00  23.4  25.6
##  3 1999-01-01 02:00:00  23.4  25.6
##  4 1999-01-01 03:00:00  23.5  25.4
##  5 1999-01-01 04:00:00  23.6  25.3
##  6 1999-01-01 05:00:00  23.5  25.2
##  7 1999-01-01 06:00:00  23.6  25.3
##  8 1999-01-01 07:00:00  23.5  25.2
##  9 1999-01-01 08:00:00  23.5  25.2
## 10 1999-01-01 09:00:00  23.6  25.2
## # ... with 162,685 more rows
```

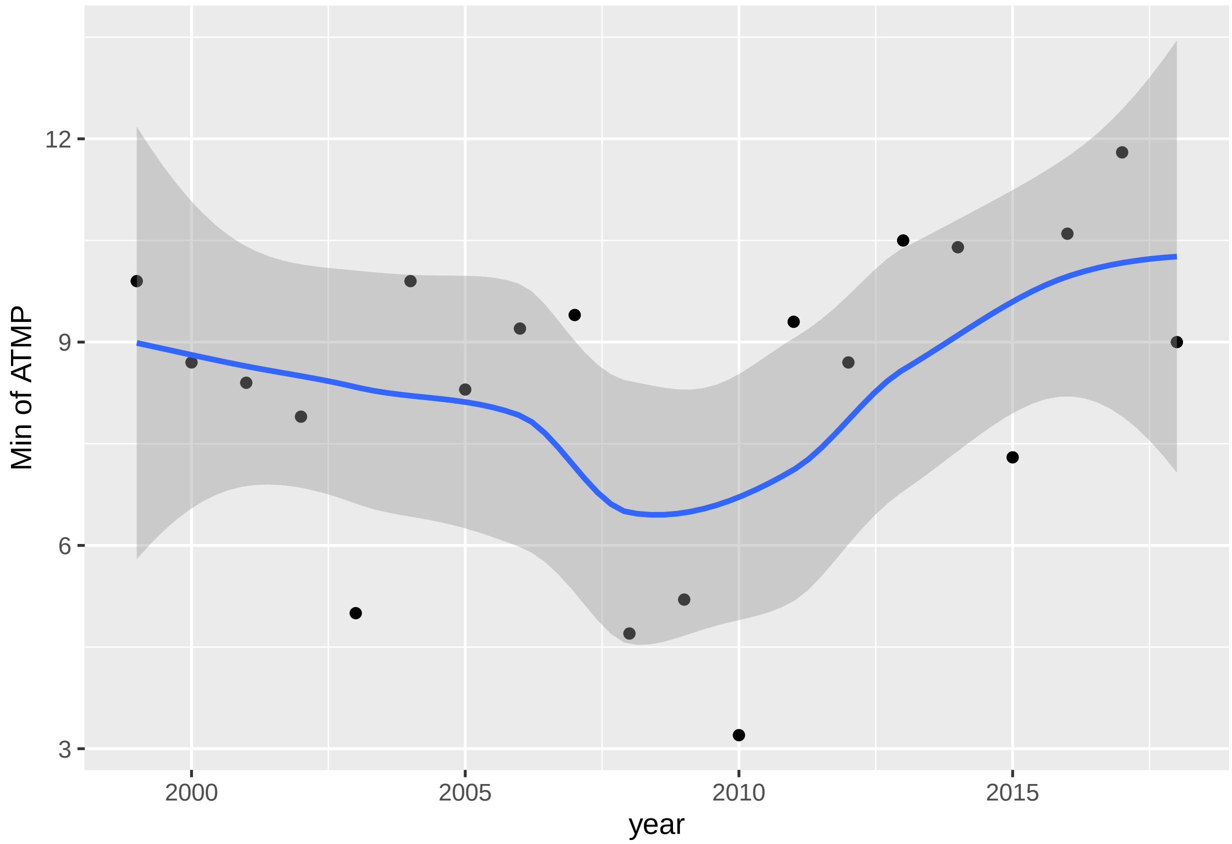
##delete rows containing "NA" ##find the minimum air temperature in a year and plot annual temperature characteristics

```r
library(dplyr)
ND$year <- year(ND$DATETIME)
ND1 <- na.omit(ND)
ND2 <- ND1%>%group_by(year)%>%summarise(min_temp=min(ATMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

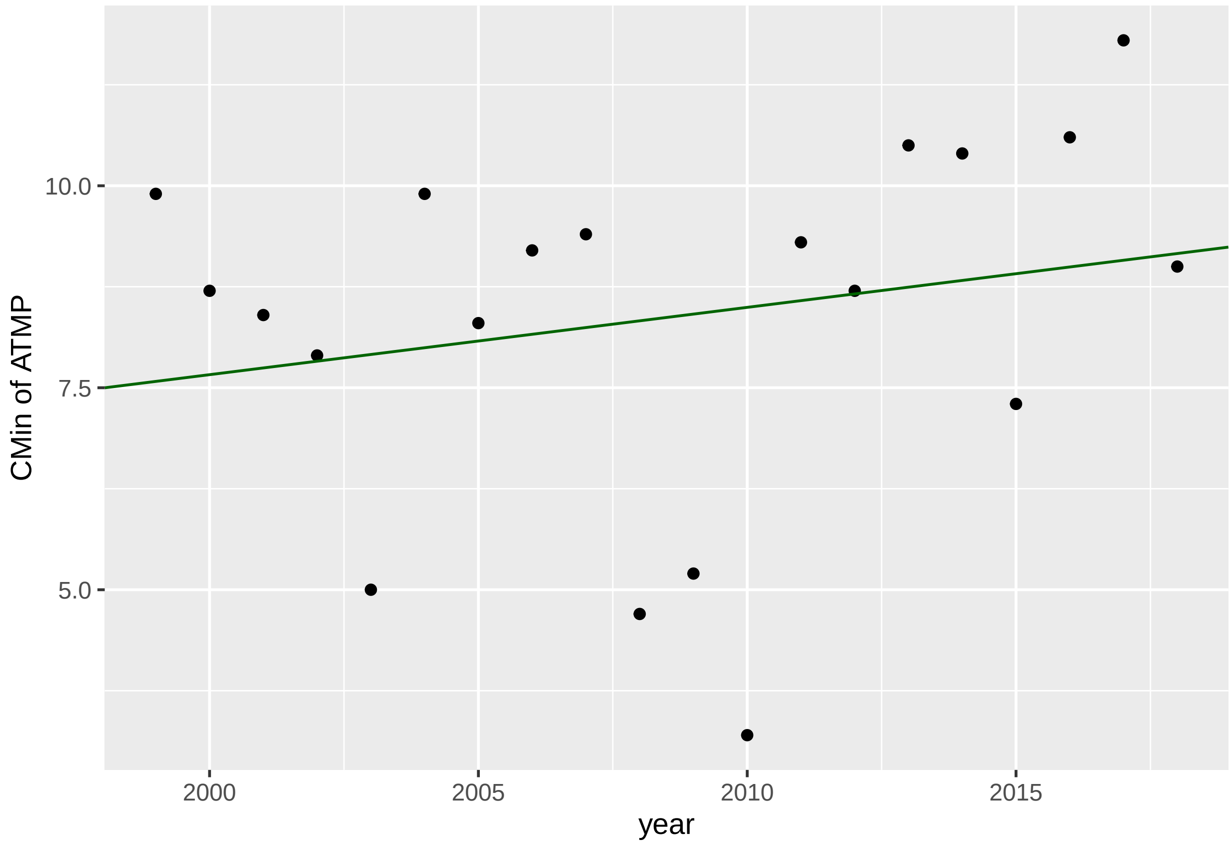
```r
y_min<-ND2$min_temp
x<-c(1999:2018)
data_min<-data.frame(x,y_min)
ggplot(data=data_min,mapping=aes(x,y_min))+
geom_point()+geom_smooth()+labs(x="year",y="Min of ATMP")
```

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



##find the minimum air temperature in a year ##do lm regression and check fit

```
data_min<-data.frame(x,y_min)
fit<-lm(y_min~x,data=data_min)
ggplot(data=data_min)+geom_point(mapping=aes(x=x,y=y_min))+ xlab("year")+ylab("CMin of ATMP")+geom_ablin
```




```r
summary(fit)
```

```
##
## Call:
## lm(formula = y_min ~ x, data = data_min)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2950 -0.5239  0.6883  1.5802  2.7219
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -158.95466  175.76500  -0.904    0.378
## x              0.08331    0.08751   0.952    0.354
##
## Residual standard error: 2.257 on 18 degrees of freedom
## Multiple R-squared:  0.04794,    Adjusted R-squared:  -0.004957
## F-statistic: 0.9063 on 1 and 18 DF,  p-value: 0.3537
```

```r
#R^2 is small, this regression doesn't fit well.
```

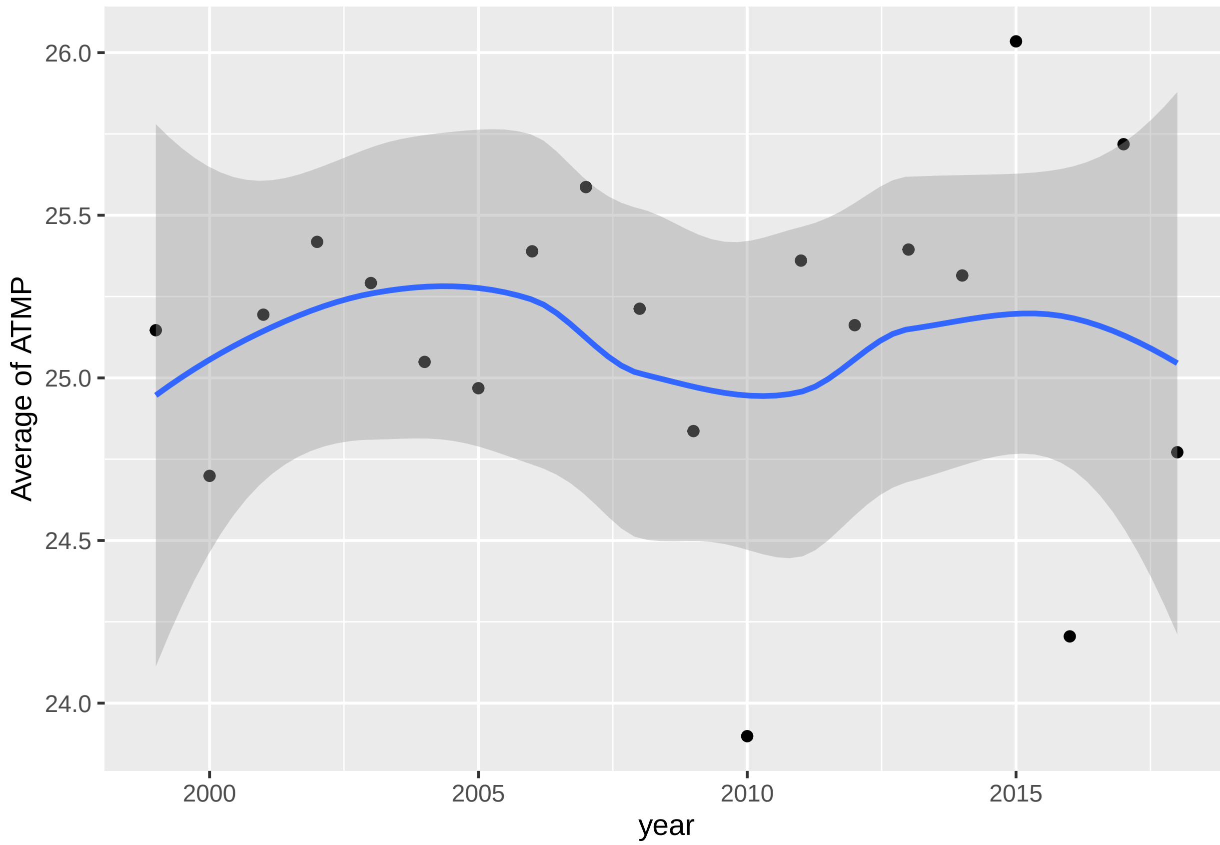
##find the average air temperature in a year and plot annual temperature characteristics

```r
ND3 <- ND1%>%group_by(year)%>%summarise(mean_temp=mean(ATMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
y_mean<-ND3$mean_temp
x<-c(1999:2018)
data_mean<-data.frame(x,y_mean)
ggplot(data=data_mean,mapping=aes(x,y_mean))+
geom_point()+geom_smooth()+labs(x="year",y="Average of ATMP")
```

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



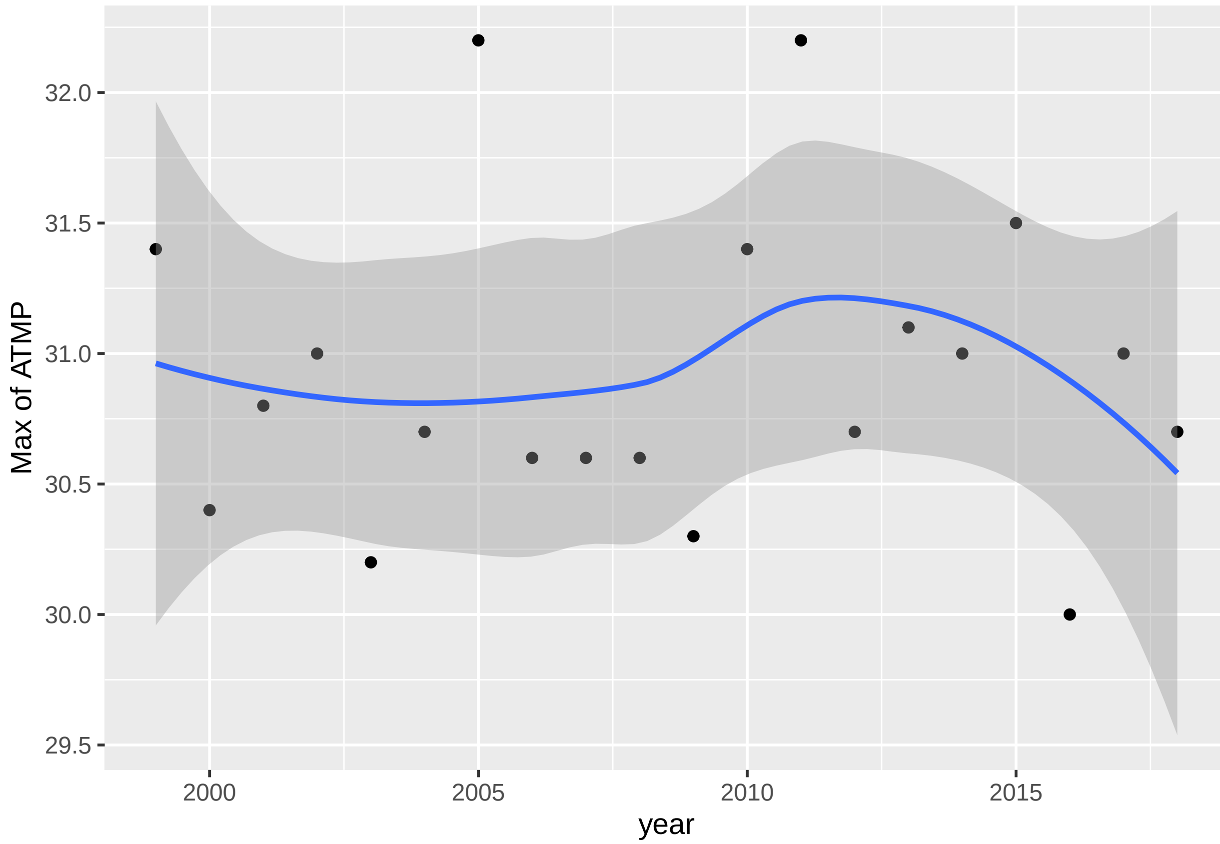
##find the maximum air temperature in a year and plot annual temperature characteristics

```r
ND4 <- ND1%>%group_by(year)%>%summarise(max_temp=max(ATMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
y_max<-ND4$max_temp
x<-c(1999:2018)
data_max<-data.frame(x,y_max)
ggplot(data=data_max,mapping=aes(x,y_max))+
geom_point()+geom_smooth()+labs(x="year",y="Max of ATMP")
```

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



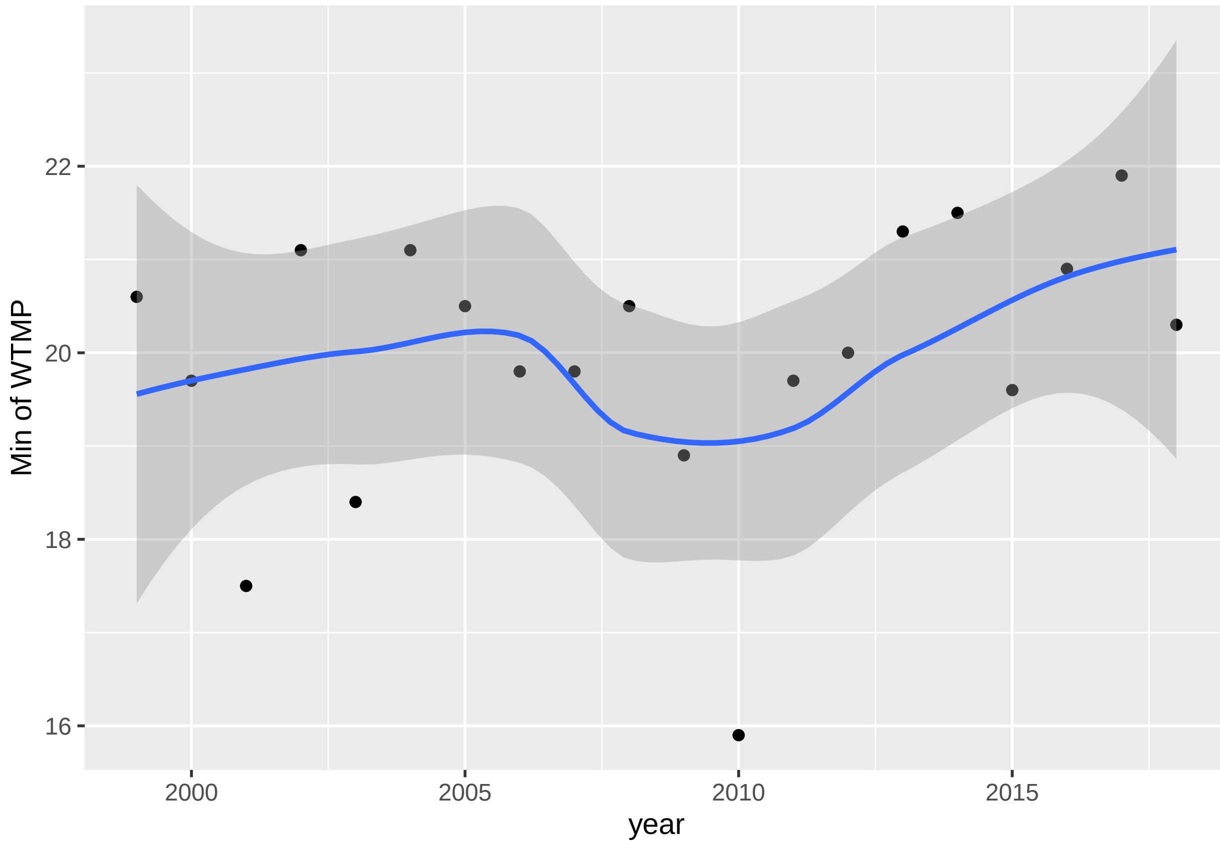
##find the minimum water temperature in a year and plot annual temperature characteristics

```
library(dplyr)
ND$year <- year(ND$DATETIME)
ND1 <- na.omit(ND)
ND5 <- ND1%>%group_by(year)%>%summarise(min_wtemp=min(WTMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
z_min<-ND5$min_wtemp
x<-c(1999:2018)
data_wmin<-data.frame(x,z_min)
ggplot(data=data_wmin,mapping=aes(x,z_min))+
geom_point()+geom_smooth()+labs(x="year",y="Min of WTMP")
```

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



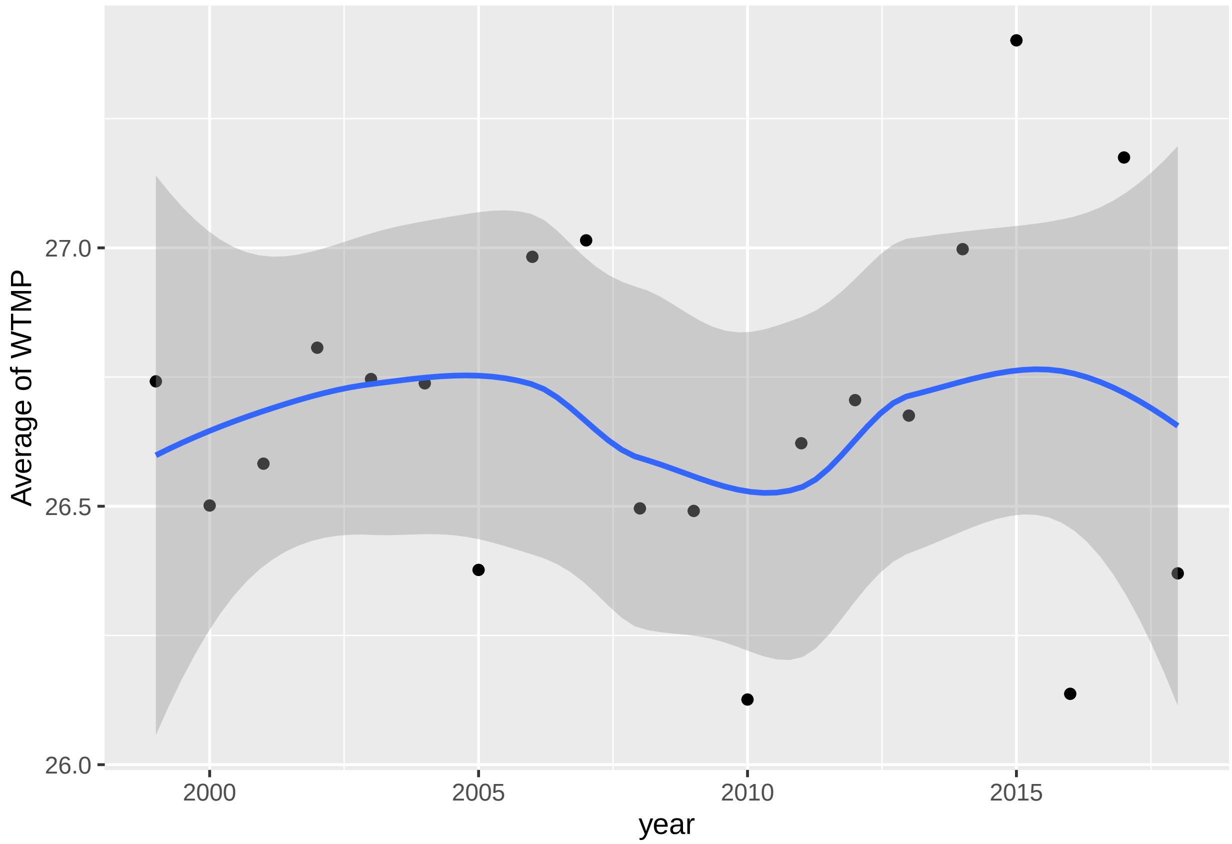
##find the average water temperature in a year and plot annual temperature characteristics

```
ND6 <- ND1%>%group_by(year)%>%summarise(mean_wtemp=mean(WTMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
z_mean<-ND6$mean_wtemp
x<-c(1999:2018)
data_wmean<-data.frame(x,z_mean)
ggplot(data=data_wmean,mapping=aes(x,z_mean))+
geom_point()+geom_smooth()+labs(x="year",y="Average of WTMP")
```

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



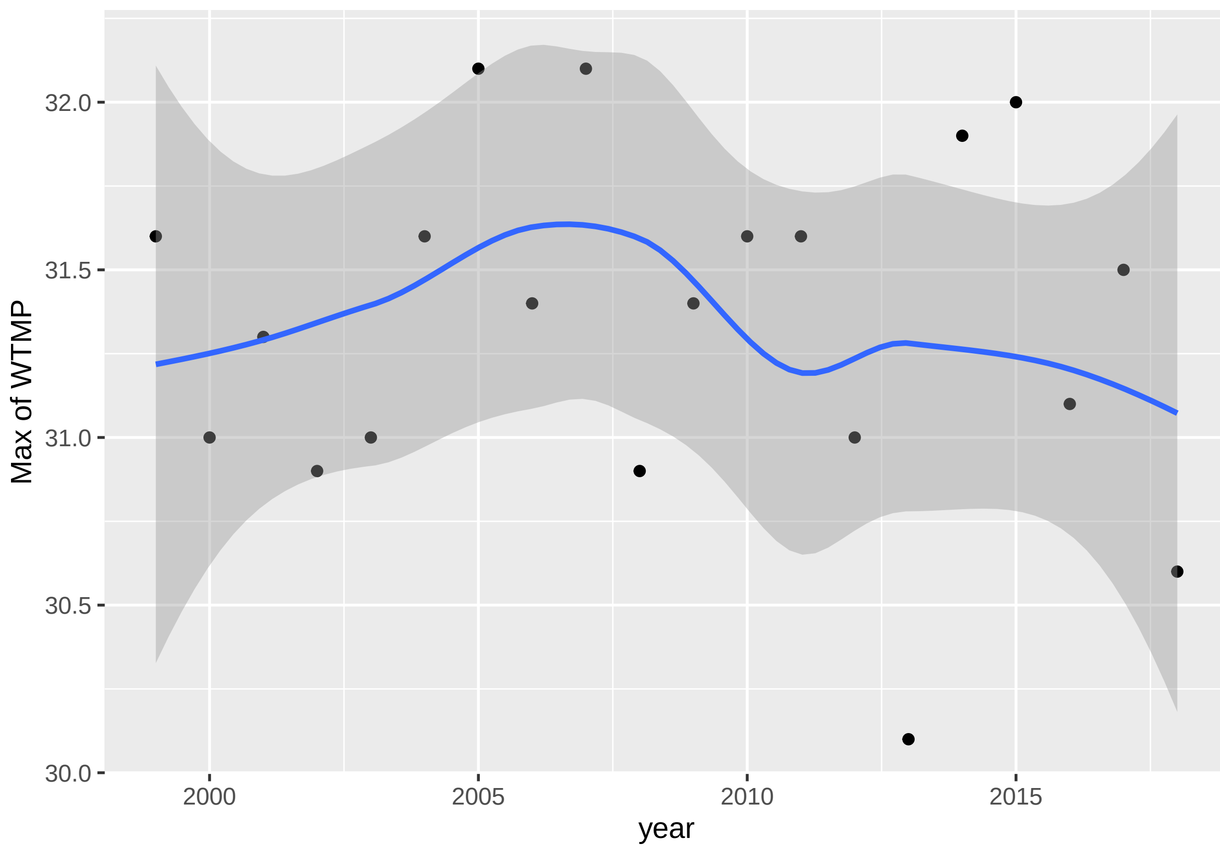
##find the Maximum water temperature in a year and plot annual temperature characteristics

```
ND7 <- ND1%>%group_by(year)%>%summarise(max_wtemp=max(WTMP))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
z_max<-ND7$max_wtemp
x<-c(1999:2018)
data_wmax<-data.frame(x,z_max)
ggplot(data=data_wmax,mapping=aes(x,z_max))+
geom_point()+geom_smooth()+labs(x="year",y="Max of WTMP")
```

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



##Conclusion: According to the 7 figures above, most of the regression line are regular wavy lines. The only linear regression (min of air tempreture ~ year) doesn't fit well due to the small R^2. To some extent, "minimum of air tempreture" could slightly show the evidence of global warming since the regression line increases from 2010 to 2018. However, there is insufficient evidence from figures to show the trend of global warmming.

##Reference Hadley Wickham (2019). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.4.0. https://CRAN.R-project.org/package=stringr

Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL http://www.jstatsoft.org/v40/i03/.

Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2020). dplyr: A Grammar of Data Manipulation. R package version 1.0.2. https://CRAN.R-project.org/package=dplyr

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.

National Oceanic and Atmospheric Adiministration's National Data Buoy Center(2020) ,Station 44013(LLNR 420)- BOSTON 16 NM East of Boston, MA Boston Approach Lighted Buoy BF NOAA 44013. Available at:https://www.ndbc.noaa.gov/station_history.php?station=44013 (Accessed: 26 Sep 2020)