

빅데이터를이용한통계그래픽스

TEAM PROJECT

[Final Report]

4조

김송희 양보연 이하은 정유진 주선미 최희원

2017.12.22. 제출

CONTENTS

I . INTRODUCTION

II. PREPARATION FOR ANALYSIS

III. DATA ANALYSIS

VARIABLES

OBSERVATIONS

POLLUTANTS

DATE

LOCATION

IV. HYPOTHESIS

V . CONCLUSION

I . INTRODUCTION

[Dataset 설명]

4조가 선정한 Pollution 자료는 EPA(미국 환경 보호국)에서 추출된 자료로 Kaggle에서 찾아서¹ 분석 자료로 사용하였다. 2000년부터 2016년까지 미국 내 주·군·시 별 네 가지 주요 오염물질(NO₂, SO₂, CO, O₃)을 매일 측정하여 csv파일에 배치한 자료이며 다음과 같다.

- 자료 크기: 약 390MB
- 관측치 개수: 1746661
- 변수 개수: 29 (범주형(9) + 연속형(21))

[자료 선정 및 역할 분배]

1. 자료 선정 과정: 각 팀원이 관측치 50000개 이상, 변수 20개 이상인 자료를 하나씩 찾아 투표를 통해 선정했다.

2. 역할 분배

: 오염물질별 - 정유진, 주선미 / 날짜별 - 김송희, 최희원 / 지역별 - 양보연, 이하은

특히, 오염물질의 경우 단위가 parts per billion 또는 parts per million으로 달라서 이를 통일시키기 위한 방법을 많이 고민했지만, 이후 데이터셋을 나누면서 자연스럽게 고려하지 않아도 되었다. .

[분석 순서]

1. Dataset 확인
2. 변수 정리 - 변수 이름 정리, 변수 추가, 관측치 정리, 이상치 처리
3. 데이터 분석 - 오염물질 자체, 날짜별, 지역별 분석 + EDA과정
4. 가설 검정 - 데이터 분석을 통해 확인한 내용을 바탕으로 다섯 가지 가설을 세우고 검정

¹ <https://www.kaggle.com/sogun3/uspollution>(접속일 2017.12.17)

II. PREPARATION FOR ANALYSIS

Kaggle 에서 받은 데이터의 경우 분석에 용이하지 않아서 자료 변형을 시도했다. 다음 순서대로 자료 변형해보았다. 이후에 나오는 모든 분석과 데이터셋은 대부분 파트를 토대로 한다.

[변수 정리]

먼저, 변수 이름의 공백을 모두 제거하고 '_'로 연결했다. 즉, 코드를 작성 시 공백으로 발생하는 모호함을 줄이고 편리함을 추구하기 위해 변수 이름을 한 단어로 만들었다. 그리고 row_number 를 나타내는 첫 번째 변수는 제외했다.

```
Pollution <- read_csv("pollution_us_2000_2016.csv", col_names = T)

## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:
## cols(
##   .default = col_double(),
##   X1 = col_integer(),
##   `State Code` = col_integer(),
##   `County Code` = col_integer(),
##   `Site Num` = col_integer(),
##   Address = col_character(),
##   State = col_character(),
##   County = col_character(),
##   City = col_character(),
##   `Date Local` = col_date(format = ""),
##   `NO2 Units` = col_character(),
##   `NO2 1st Max Hour` = col_integer(),
##   `NO2 AQI` = col_integer(),
##   `O3 Units` = col_character(),
##   `O3 1st Max Hour` = col_integer(),
##   `O3 AQI` = col_integer(),
##   `SO2 Units` = col_character(),
##   `SO2 1st Max Hour` = col_integer(),
##   `CO Units` = col_character(),
##   `CO 1st Max Hour` = col_integer()
## )

## See spec(...) for full column specifications.

#Main data 의 원자료 불러오기

#변수 이름 정리
Pollution <- Pollution %>% select(-X1) %>% rename(State_Code = 'State Code', County_Code = '
County Code', Site_Num = 'Site Num', Date_Local = 'Date Local', NO2_Units = 'NO2 Units', NO2
_Mean = 'NO2 Mean', NO2_1st_Max_Value = 'NO2 1st Max Value', NO2_1st_Max_Hour = 'NO2 1st Ma
x Hour', NO2_AQI = 'NO2 AQI', O3_Units = 'O3 Units', O3_Mean = 'O3 Mean', O3_1st_Max_Value
= 'O3 1st Max Value', O3_1st_Max_Hour = 'O3 1st Max Hour', O3_AQI = 'O3 AQI', SO2_Units = 'S
O2 Units', SO2_Mean = 'SO2 Mean', SO2_1st_Max_Value = 'SO2 1st Max Value', SO2_1st_Max_Hour
= 'SO2 1st Max Hour', SO2_AQI = 'SO2 AQI', CO_Units = 'CO Units', CO_Mean = 'CO Mean', CO_1
st_Max_Value = 'CO 1st Max Value', CO_1st_Max_Hour = 'CO 1st Max Hour', CO_AQI = 'CO AQI')

#select(), 첫 번째 컬럼은 행 순서로 별다른 의미가 없어서 제외.

#rename(), 변수명에 공백이 있는 경우 변수 호출 시 문제가 발생하지 않도록 '_'로 변경
```

[변수 추가]

Date_local 변수는 '2010-01-01'와 같은 date 형 변수를 나타낸다. 이 변수를 연, 월, 일로 나눈 Year, Month, Day 를 추가하여 분석 시 용이하게 사용했다.

```
Pollution <- Pollution %>% separate(Date_Local, into = c("Year", "Month", "Day"), sep = "-",
remove=FALSE, convert=TRUE)
```

#separate(), Date_Local 변수를 나눈 Year, Month, Day 변수 추가. Date_Local 는 자료에 그대로 두고 세 변수 모두 int 형으로 변경.

또한, 변수 이름과 내용을 확인하며 AQI 에 대해 자세히 알아보았다. 공기 품질 지수를 나타내는 AQI(Air quality index)는 저희 조의 데이터셋 출처와 같은 미국 환경 보호국(EPA)에서 정한 기준으로, 이 지수에 따라 여러 범주²로 나눌 수 있다는 사실을 알게되었다.

Air Quality Index Levels of Health Concern	Numerical Value	Meaning
Good	0 to 50	Air quality is considered satisfactory, and air pollution poses little or no risk.
Moderate	51 to 100	Air quality is acceptable; however, for some pollutants there may be a moderate health concern for a very small number of people who are unusually sensitive to air pollution.
Unhealthy for Sensitive Groups	101 to 150	Members of sensitive groups may experience health effects. The general public is not likely to be affected.
Unhealthy	151 to 200	Everyone may begin to experience health effects; members of sensitive groups may experience more serious health effects.
Very Unhealthy	201 to 300	Health warnings of emergency conditions. The entire population is more likely to be affected.
Hazardous	301 to 500	Health alert: everyone may experience more serious health effects.

따라서, 이 지표를 기준으로 나눈 변수 또한 추가했다. 이는 데이터셋을 나눈 후에 추가했기 때문에, 이후 나오는 [관측치 정리]에서 보고자 한다.

[관측치 정리]

Kaggle 에서 받은 데이터셋은 중복 관측이 존재했다. 다음을 보자.

Date_Local	NO2_Mean	Max_Value	O3_Mean	Max_Value	SO2_Mean	Max_Value	CO_Mean	Max_Value
2000-01-01	19.04167	49	0.0225	0.04	3	9	1.14583	4.2
2000-01-01	19.04167	49	0.0225	0.04	3	9	0.87895	2.2
2000-01-01	19.04167	49	0.0225	0.04	2.975	6.6	1.14583	4.2
2000-01-01	19.04167	49	0.0225	0.04	2.975	6.6	0.87895	2.2
2000-01-02	22.95833	36	0.01338	0.032	1.958333	3	0.85	1.6
2000-01-02	22.95833	36	0.01338	0.032	1.958333	3	1.06667	2.3
2000-01-02	22.95833	36	0.01338	0.032	1.9375	2.6	0.85	1.6
2000-01-02	22.95833	36	0.01338	0.032	1.9375	2.6	1.06667	2.3

² <https://forum.airnowtech.org/t/aqi-calculations-overview-ozone-pm2-5-and-pm10/168> (접속일 2017.12.21.)

위는 4 조 데이터셋의 일부이다. 하루 안에서 NO2 는 4 번씩, O3 는 4 번씩, SO2 는 2 번씩, CO 는 2 번씩의 중복이 발생하고 있다. 따라서 다른 시간에 측정된 것인지 먼저 의심했다. 이 과정에서 지만, 변수를 보았을 때, mean 은 하루 평균, max_value 는 하루 중 가장 높은 수치, max_value_hour 는 max_value 가 관측된 시간으로 한 레코드는 하루의 관측을 정리한 것이다. 이 중복을 도식화해보았다. 그러나, SO2 와 CO 의 경우 max_value 와 max_hour 의 다른 수치가 하루에 두 번씩 있는 셈이다. 이 이유는 정확히 알 수 없지만, 이런 경우는 데이터를 따로 삭제하지 않기로 합의했다.

NO2	O3	SO2	CO

위 표와 같은 방식으로 데이터가 엮여있고, 이는 네 오염물질 별 관측을 합치면서 생긴 문제로 판단했다. 이 kaggle 데이터의 출처인 EPA 에 들어가서 확인한 결과 연도별로, 오염물질별로 데이터가 모두 나뉘어있었고, 이를 한 데이터셋에 합치는 과정에서 생긴 문제로 추측했다. 또한, 더 확인해본 결과 중복이 4 번 이상이 일어난 경우도 꽤 있었다. 이 문제를 처리하기 위해 팀 내에서 굉장히 많은 상의를 거쳤고, 중복된 내용을 정리할 때 다른 중요한 내용이 삭제되지 않도록 데이터셋을 오염물질별로 나누고 각각 중복된 내용을 정리하기로 결정했다. 다음의 코드를 이용했다.

#오염물질 별로 자료 나눈 data set

```
Pollution_NO2 <- Pollution %>% select(State_Code, County_Code, Site_Num, Address, State, County, City, Date_Local, Year, Month, Day, NO2_Units, NO2_Mean, NO2_1st_Max_Value, NO2_1st_Max_Hour, NO2_AQI) distinct()
```

```
Pollution_SO2 <- Pollution %>% select(State_Code, County_Code, Site_Num, Address, State, County, City, Date_Local, Year, Month, Day, SO2_Units, SO2_Mean, SO2_1st_Max_Value, SO2_1st_Max_Hour, SO2_AQI) distinct()
```

```
Pollution_O3 <- Pollution %>% select(State_Code, County_Code, Site_Num, Address, State, County, City, Date_Local, Year, Month, Day, O3_Units, O3_Mean, O3_1st_Max_Value, O3_1st_Max_Hour, O3_AQI) distinct()
```

```
Pollution_CO <- Pollution %>% select(State_Code, County_Code, Site_Num, Address, State, County, City, Date_Local, Year, Month, Day, CO_Units, CO_Mean, CO_1st_Max_Value, CO_1st_Max_Hour, CO_AQI) distinct()
```

정리한 결과, 불가피하게도 관측 기간은 모두 같지만 각 오염물질 별 레코드 수가 달라졌다. 원래 데이터셋 Pollution 의 레코드는 1746661 였는데, Pollution_NO2 는 413759, Pollution_O3 는 416130, Pollution_SO2 는 836741, Pollution_CO 는 843691 로 줄어들었다. NO2 의 경우 가장 많은 중복이 있었고, 거의 4 분의 1 가량으로 레코드 수가 줄어들었다.

다음으로, 5p 에서 언급했던 AQI 범주를 나타내는 변수를 추가했다.

#AQI 범주 넣기

```
Pollution_NO2 <- Pollution_NO2 %>% mutate(NO2_AQI_Class = NO2_AQI %/% 50, NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "0", "Good"), NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "1", "Moderate"), NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "2", "Unhealthy for Sensitive Groups"), NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "3", "Unhealthy"), NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "4", "Very Unhealthy"), NO2_AQI_Class =
stringr::str_replace(NO2_AQI_Class, "5", "Hazardous"))

Pollution_SO2 <- Pollution_SO2 %>% mutate(SO2_AQI_Class = SO2_AQI %/% 50, SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "0", "Good"), SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "1", "Moderate"), SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "2", "Unhealthy for Sensitive Groups"), SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "3", "Unhealthy"), SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "4", "Very Unhealthy"), SO2_AQI_Class =
stringr::str_replace(SO2_AQI_Class, "5", "Hazardous"))

Pollution_O3 <- Pollution_O3 %>% mutate(O3_AQI_Class = O3_AQI %/% 50, O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "0", "Good"), O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "1", "Moderate"), O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "2", "Unhealthy for Sensitive Groups"), O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "3", "Unhealthy"), O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "4", "Very Unhealthy"), O3_AQI_Class =
stringr::str_replace(O3_AQI_Class, "5", "Hazardous"))

Pollution_CO <- Pollution_CO %>% mutate(CO_AQI_Class = CO_AQI %/% 50, CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "0", "Good"), CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "1", "Moderate"), CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "2", "Unhealthy for Sensitive Groups"), CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "3", "Unhealthy"), CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "4", "Very Unhealthy"), CO_AQI_Class =
stringr::str_replace(CO_AQI_Class, "5", "Hazardous"))
```

[이상치 처리]

str(), summary() 함수를 통해 기본적인 데이터의 구조와 각 변수의 기본적인 통계량을 알아보았다. 한 번에 확인하기 위해 Pollution 데이터 셋을 이용했다.

```
summary(Pollution)
```

```
## State_Code County_Code Site_Num Address
## Min. : 1.00 Min. : 1.00 Min. : 1 Length:1746661
## 1st Qu.: 6.00 1st Qu.: 17.00 1st Qu.: 9 Class :character
## Median :17.00 Median : 59.00 Median : 60 Mode :character
## Mean :22.31 Mean : 71.69 Mean :1118
## 3rd Qu.:40.00 3rd Qu.: 97.00 3rd Qu.:1039
## Max. :80.00 Max. :650.00 Max. :9997
##
## State County City
## Length:1746661 Length:1746661 Length:1746661
## Class :character Class :character Class :character
## Mode :character Mode :character Mode :character
##
```

```

##   Date_Local      Year      Month      Day
##   Min. :2000-01-01   Min. :2000   Min. : 1.00   Min. : 1.00
##   1st Qu.:2004-11-23   1st Qu.:2004   1st Qu.: 4.00   1st Qu.: 8.00
##   Median :2009-02-03   Median :2009   Median : 7.00   Median :16.00
##   Mean :2008-10-13   Mean :2008   Mean : 6.52   Mean :15.75
##   3rd Qu.:2012-11-06   3rd Qu.:2012   3rd Qu.: 9.00   3rd Qu.:23.00
##   Max. :2016-05-31   Max. :2016   Max. :12.00   Max. :31.00
##
##   NO2_Units      NO2_Mean      NO2_1st_Max_Value NO2_1st_Max_Hour
##   Length:1746661   Min. : -2.00   Min. : -2.00   Min. : 0.00
##   Class :character 1st Qu.: 5.75   1st Qu.:13.00   1st Qu.: 5.00
##   Mode :character  Median :10.74   Median :24.00   Median : 9.00
##                      Mean :12.82   Mean :25.41   Mean :11.73
##                      3rd Qu.:17.71   3rd Qu.:35.70   3rd Qu.:20.00
##                      Max. :139.54   Max. :267.00   Max. :23.00
##
##   NO2_AQI      O3_Units      O3_Mean      O3_1st_Max_Value
##   Min. : 0.0   Length:1746661   Min. :0.00000   Min. :0.0000
##   1st Qu.:12.0   Class :character 1st Qu.:0.01787   1st Qu.:0.0290
##   Median :23.0   Mode :character  Median :0.02587   Median :0.0380
##   Mean :23.9                      Mean :0.02612   Mean :0.0392
##   3rd Qu.:33.0                      3rd Qu.:0.03392   3rd Qu.:0.0480
##   Max. :132.0                      Max. :0.09508   Max. :0.1410
##
##   O3_1st_Max_Hour O3_AQI      SO2_Units      SO2_Mean
##   Min. : 0.00   Min. : 0.00   Length:1746661   Min. : -2.0000
##   1st Qu.: 9.00   1st Qu.:25.00   Class :character 1st Qu.: 0.2565
##   Median :10.00   Median :33.00   Mode :character  Median : 0.9875
##   Mean :10.17   Mean :36.05                      Mean : 1.8704
##   3rd Qu.:11.00   3rd Qu.:42.00                      3rd Qu.: 2.3250
##   Max. :23.00   Max. :218.00                      Max. :321.6250
##
##   SO2_1st_Max_Value SO2_1st_Max_Hour SO2_AQI      CO_Units
##   Min. : -2.000   Min. : 0.000   Min. : 0.0   Length:1746661
##   1st Qu.: 0.800   1st Qu.: 5.000   1st Qu.: 1.0   Class :character
##   Median : 2.000   Median : 8.000   Median : 3.0   Mode :character
##   Mean : 4.492   Mean : 9.665   Mean : 7.1
##   3rd Qu.: 5.000   3rd Qu.:14.000   3rd Qu.: 9.0
##   Max. :351.000   Max. :23.000   Max. :200.0
##                      NA's :872907
##   CO_Mean      CO_1st_Max_Value CO_1st_Max_Hour CO_AQI
##   Min. : -0.4375   Min. : -0.4000   Min. : 0.000   Min. : 0
##   1st Qu.: 0.1835   1st Qu.: 0.2920   1st Qu.: 0.000   1st Qu.: 2
##   Median : 0.2926   Median : 0.4000   Median : 6.000   Median : 5
##   Mean : 0.3682   Mean : 0.6201   Mean : 7.875   Mean : 6
##   3rd Qu.: 0.4667   3rd Qu.: 0.8000   3rd Qu.:13.000   3rd Qu.: 8
##   Max. : 7.5083   Max. :19.9000   Max. :23.000   Max. :201
##                      NA's :873323
#str(Pollution)


```

summary()를 통해 출력한 결과 농도를 나타내는 mean 과 max_value 값에 0 또는 음수 값이 존재했다. 그래서 이상함을 느끼고 오염물질의 측정 단위인 ppm 과 ppb 구하는 방법³을 찾아보았다. 참고로, 측정 단위는 NO2 와 SO2 는 ppb, O3 와 CO 는 ppm 이었다.

³ <http://slideplayer.com/slide/5675334>(접속일 2017.12.21)

Parts per million/billion (ppm & ppb)

- $\text{ppm} = \frac{\text{mass solute}}{\text{volume solution}} \times 10^6$
- $\text{ppb} = \frac{\text{mass solute}}{\text{volume solution}} \times 10^9$



Mass and volume units must match.

(g & mL) or (Kg & L)

or $\frac{\text{mg}}{\text{L}} = \text{ppm}$

or $\frac{\mu\text{g}}{\text{L}} = \text{ppb}$

AND

For very low concentrations:

parts per trillion $\frac{\text{ng}}{\text{L}} = \text{ppt}$

공식을 확인해본 결과, 전체 용질 중 특정 물질의 용질이 0 또는 음수가 될 수 없기 때문에 이는 잘못된 관측이라고 판단했다. 그런데, 실제로 잘못된 관측이 일어난 것인지 의문을 품게 되었다. 따라서 지인을 통해 이화여대 환경공학과 학생에게 자문을 구했고, 이는 기계의 이상으로 발생하는 이상치라는 것을 알게 되었다. 0 인 값은 특히 그 관측 수가 꽤 많았기 때문에 이를 제거하고 분석을 하는 것이 맞다고 판단하고, 이러한 이상치들을 제거했다.

#mean, max_value 0 또는 음수 제외

```
Pollution_NO2 <- Pollution_NO2 %>% filter(NO2_Mean > 0, NO2_1st_Max_Value > 0)
```

```
Pollution_SO2 <- Pollution_SO2 %>% filter(SO2_Mean > 0, SO2_1st_Max_Value > 0)
```

```
Pollution_O3 <- Pollution_O3 %>% filter(O3_Mean > 0, O3_1st_Max_Value > 0)
```

```
Pollution_CO <- Pollution_CO %>% filter(CO_Mean > 0, CO_1st_Max_Value > 0)
```

제거하고나니 각각의 레코드수가 NO2 는 411471, O3 는 415980, SO2 는 755350, CO 는 814443 이 되었다. 이는 원래 레코드수에 현저히 줄은 수치이다.

이제, 분석을 위한 데이터셋이 마련되었다. 데이터셋을 넷으로 나눔으로서 단위의 압박에서 자연스럽게 벗어났고 필요없는 중복 데이터를 제거함으로써 R 상에서 더욱 메모리를 효율적으로 관리할 수 있게 되었다. 더 필요한 자료는 분석 과정에서 추가로 생성했다.

III. DATA ANALYSIS

【 VARIABLES 】

다음 표는 변수를 정리한 내용이다.

Variables Description

Variable		Type	Description
State_Code		int	미국 환경보호국(EPA)이 지정한 주 코드
County_Code		int	미국 환경보호국(EPA)이 지정한 군 코드
Site_Num		int	미국 환경보호국(EPA)이 지정한 관측 장소 코드
Address		char	관측 장소의 실제 주소
State		char	관측 장소의 주 명칭
County		char	관측 장소의 군 명칭
City		char	관측 장소의 시 명칭
Date_Local		date	관측 일자
NO2(이산화질소) O3(오존) SO2(이산화황) CO2(일산화탄소)	_Units	char	이산화질소 측정 단위
	_Mean	num	하루 동안 측정된 이산화질소의 평균 농도
	_1st_Max_Value	num	하루 동안 측정된 이산화질소 농도 중 최대치
	_1st_Max_Hour	int	이산화질소 농도 최대치가 측정된 시간
	_AQI	int	하루 동안 측정된 이산화질소의 공기 품질 지표
		char	공기 품질 지표의 범주 - 6가지
Year		int	관측한 연도
Month		int	관측한 달
Day		int	관측한 날짜

*파란색: 기존 변수 / 빨간색: 추가 변수

위 표의 변수는 앞서 정리한 데이터셋을 토대로 변수들의 이름과 설명을 작성한 것이다.

다음으로, 각 위치를 구별할 수 있는 KEY 조합을 찾았다. 참고로 primary key로 선정할 좋은 natural key 조합은 없었기 때문에 surrogate key를 이용해야하는 상황이었다. 하지만 primary key를 사용할 일이 없었으므로 따로 추가하지 않았다.

중복 없는 State_Code와 State의 수는 같았지만, County_Code와 County 수는 달랐다. 또한 Site_Num과 address 수도 달랐다. 이에 의문을 품고 여러 조합을 시도해본 결과,

#조합

```
Pollution %>% count(State, County) %>% print(n=0)

## # A tibble: 139 x 3
## # ... with 3 variables: State <chr>, County <chr>, n <int>

Pollution %>% count(State_Code, County_Code) %>% print(n=0)

## # A tibble: 139 x 3
## # ... with 3 variables: State_Code <int>, County_Code <int>, n <int>
```

#개수 같음

```
Pollution %>% count(Site_Num) %>% print(n=0)

## # A tibble: 110 x 2
## # ... with 2 variables: Site_Num <int>, n <int>

Pollution %>% count(City, Address) %>% print(n=0)

## # A tibble: 204 x 3
## # ... with 3 variables: City <chr>, Address <chr>, n <int>
```

#개수 다름

```
Pollution %>% count(Site_Num) %>% print(n=0)

## # A tibble: 110 x 2
## # ... with 2 variables: Site_Num <int>, n <int>

Pollution %>% count(Address) %>% print(n=0)

## # A tibble: 204 x 2
## # ... with 2 variables: Address <chr>, n <int>
```

#개수 다름

```
Pollution %>% count(State, County, City, Address) %>% print(n=0)

## # A tibble: 204 x 5
## # ... with 5 variables: State <chr>, County <chr>, City <chr>,
## #   Address <chr>, n <int>

Pollution %>% count(State_Code, County_Code, Site_Num) %>% print(n=0)

## # A tibble: 204 x 4
## # ... with 4 variables: State_Code <int>, County_Code <int>,
## #   Site_Num <int>, n <int>
```

#개수 같음!!

Code 의 경우 상위 지역 Code 와 함께 사용해야한다는 사실을 알게되었다. 또한, (State, County, City, Address) 조합과 (State_Code, County_Code, Site_Num) 조합이 같은 정보를 나타내는 것을 알게되었다. 따라서 고유한 위치를 구별하기 위해서 (State, County, City, Address)조합, (State_Code, County_Code, Site_Num)조합, (State, County, Site_Num)조합 중 하나를 사용하기로 했다.

【 OBSERVATIONS 】

관측 자체에 관해 살펴보았다.

먼저, 기본적으로 중복되지 않는 State 의 수와 county 수를 확인했다.

```
Pollution %>% count(n_distinct(State))

## # A tibble: 1 x 2
##   `n_distinct(State)`      n
##           <int>   <int>
## 1                47 1746661

Pollution %>% count(n_distinct(County))

## # A tibble: 1 x 2
##   `n_distinct(County)`      n
##           <int>   <int>
## 1                133 1746661

Pollution %>% count(n_distinct(City))

## # A tibble: 1 x 2
##   `n_distinct(City)`      n
##           <int>   <int>
## 1                144 1746661

Pollution %>% count(n_distinct(Address))

## # A tibble: 1 x 2
##   `n_distinct(Address)`      n
##           <int>   <int>
## 1                 204 1746661
```

확인해본 결과 관측은 47 개의 주이자, 133 개의 군, 144 개의 시에서 실시되었다. 미국의 주는 총 50 개이므로 미국 전역에서 관측되었다고 볼 수 있다. 그러나, 어느 지역에서 특히 많은 관측이 이루어졌는지 살펴보았다.

```
Pollution_N02 %>% count(State) %>% arrange(desc(n)) %>% print(n = 3)

## # A tibble: 47 x 2
##       State      n
##       <chr> <int>
## 1 California 139105
## 2 Pennsylvania 47190
## 3 Texas      29886
## # ... with 44 more rows

Pollution_S02 %>% count(State) %>% arrange(desc(n)) %>% print(n = 3)

## # A tibble: 47 x 2
##       State      n
##       <chr> <int>
## 1 California 245560
## 2 Pennsylvania 90834
## 3 Texas      46766
## # ... with 44 more rows

Pollution_O3 %>% count(State) %>% arrange(desc(n)) %>% print(n = 3)
```

```
## # A tibble: 47 x 2
##       State      n
##     <chr> <int>
## 1 California 140541
## 2 Pennsylvania 47230
## 3 Texas      30140
## # ... with 44 more rows

Pollution_CO %>% count(State) %>% arrange(desc(n)) %>% print(n = 3)

## # A tibble: 47 x 2
##       State      n
##     <chr> <int>
## 1 California 280369
## 2 Pennsylvania 82153
## 3 Texas      58908
## # ... with 44 more rows
```

State 별로 관측 수를 구하고 내림차순으로 정렬했을 때, California 주에서 압도적인 관측이 발생했다. NO2의 경우 2위에 비해 거의 4배 이상의 관측이 California 주에서 이루어졌다는 것을 알 수 있었다. 2위는 Pennsylvania 주, 3위는 Texas 주가 차지했다. 이 주에 어떤 군과 도시가 있는지 잠깐 확인해보자.

```
Pollution %>% filter(State == "California") %>% group_by(State, County) %>%
summarise(n_distinct(City))

## # A tibble: 18 x 3
## # Groups:   State [?]
##       State      County `n_distinct(City)`
##     <chr>      <chr>          <int>
## 1 California Alameda              2
## 2 California Contra Costa          5
## 3 California Fresno              1
## 4 California Humboldt            2
## 5 California Imperial            1
## 6 California Kern                1
## 7 California Los Angeles          5
## 8 California Orange              1
## 9 California Riverside           1
## 10 California Sacramento          2
## 11 California San Bernardino      2
## 12 California San Diego           3
## 13 California San Francisco        1
## 14 California Santa Barbara        4
## 15 California Santa Clara          2
## 16 California Santa Cruz           1
## 17 California Solano              2
## 18 California Ventura             1

Pollution %>% filter(State == "Pennsylvania") %>% group_by(State, County) %>%
summarise(n_distinct(City))

## # A tibble: 19 x 3
## # Groups:   State [?]
##       State      County `n_distinct(City)`
##     <chr>      <chr>          <int>
## 1 Pennsylvania Adams              1
## 2 Pennsylvania Allegheny          1
## 3 Pennsylvania Beaver             1
## 4 Pennsylvania Berks              2
## 5 Pennsylvania Blair              1
## 6 Pennsylvania Bucks              1
```

```
## 7 Pennsylvania Cambria 1
## 8 Pennsylvania Dauphin 1
## 9 Pennsylvania Erie 1
## 10 Pennsylvania Lackawanna 1
## 11 Pennsylvania Lancaster 1
## 12 Pennsylvania Lawrence 1
## 13 Pennsylvania Luzerne 1
## 14 Pennsylvania Montgomery 1
## 15 Pennsylvania Northampton 1
## 16 Pennsylvania Philadelphia 1
## 17 Pennsylvania Washington 1
## 18 Pennsylvania Westmoreland 1
## 19 Pennsylvania York 1

Pollution %>% filter(State == "Texas") %>% group_by(State, County) %>%
summarise(n_distinct(City))

## # A tibble: 6 x 3
## # Groups:   State [?]
##   State County `n_distinct(City)`
##   <chr> <chr> <int>
## 1 Texas Bexar 2
## 2 Texas Dallas 1
## 3 Texas El Paso 1
## 4 Texas Harris 2
## 5 Texas McLennan 1
## 6 Texas Travis 1
```

California 에 총 18 개의 군에서 관측이 이루어졌으며, 군별 관측된 도시 숫자를 보아 California 내에서도 전역에서 관측이 이루어진 것을 알 수 파악할 수 있다. Pennsylvania 의 경우 19 개의 군에서, Texas 의 경우 6 개의 군에서 관측이 이루어졌다. 다시 돌아와서, 미국 전역에서 관측이 많이 이루어진 군을 확인해보았다.

```
Pollution_NO2 %>% count(State, County) %>% arrange(desc(n)) %>% print(n = 5)

## # A tibble: 139 x 3
##   State County n
##   <chr> <chr> <int>
## 1 California Los Angeles 22360
## 2 California Contra Costa 21003
## 3 California Santa Barbara 19403
## 4 California San Diego 12741
## 5 Arizona Maricopa 11630
## # ... with 134 more rows

Pollution_SO2 %>% count(State, County) %>% arrange(desc(n)) %>% print(n = 5)

## # A tibble: 139 x 3
##   State County n
##   <chr> <chr> <int>
## 1 California Los Angeles 40050
## 2 California Contra Costa 39987
## 3 California Santa Barbara 30666
## 4 California San Diego 24977
## 5 Arizona Maricopa 21891
## # ... with 134 more rows

Pollution_O3 %>% count(State, County) %>% arrange(desc(n)) %>% print(n = 5)

## # A tibble: 139 x 3
##   State County n
##   <chr> <chr> <int>
```

```
## 1 California Los Angeles 22356
## 2 California Contra Costa 21001
## 3 California Santa Barbara 20755
## 4 California San Diego 12747
## 5 Arizona Maricopa 11231
## # ... with 134 more rows

Pollution_CO %>% count(State, County) %>% arrange(desc(n)) %>% print(n = 5)

## # A tibble: 138 x 3
##       State      County      n
##       <chr>      <chr> <int>
## 1 California Los Angeles 46113
## 2 California Contra Costa 41974
## 3 California Santa Barbara 37281
## 4 California San Diego 25469
## 5 Arizona Maricopa 22417
## # ... with 133 more rows
```

확인해본 결과 네 오염물질 모두 1 위는 Californi 주의 Los Angeles 군이, 2 위는 California 주의 Contra Costa 군, 3 위는 California 주의 Santa Barbara 군이, 4 위는 California 주의 San Diego 군이, 5 위는 Arizona 주의 Maricopa 군이 차지했다.

【 POLLUTANTS 】

먼저, 우리의 관심 오염물질 – NO₂, O₃, SO₂, CO 에 대해 간단히 알아보았다.

NO₂ (이산화질소) 4 5	독성이 있는 적갈색의 기체 극성물질로 물에 잘 녹음. 산성비의 주범 중 하나 두통, 구역질, 호흡 곤란을 일으키며 미치료시 폐수종으로 사망 가능
O₃ (오존) 6 7 8	산화력이 강해 살균, 악취제거에 이용 대기 상층부(성층권)의 오존층(약 90%)은 자외선을 차단 지표 부근(대류권)에 있는 오존(10%)은 일정 기준 이상 높아지면 호흡기와 눈에 자극, 폐기능 저하 및 농작물에 피해
SO₂ (이산화황) 9	자극적인 냄새의 유독성 기체 주로 공장이나 발전소 등에서 황을 포함한 금속 물질을 태울 때 대기로 유입 눈 염증, 호흡기 질환 및 알레르기 유발, 심하면 사망 가능

⁴ https://ko.wikipedia.org/wiki/%EC%9D%B4%EC%82%B0%ED%99%94_%EC%A7%88%EC%86%8C(접속일 2017.12.22.)

⁵ https://namu.wiki/w/%EC%9D%B4%EC%82%B0%ED%99%94_%EC%A7%88%EC%86%8C(접속일 2017.12.22.)

⁶ <https://ko.wikipedia.org/wiki/%EC%98%A4%EC%A1%B4>(접속일 2017.12.22.)

⁷ <https://namu.wiki/w/%EC%98%A4%EC%A1%B4>(접속일 2017.12.22.)

⁸ <http://ecopia.incheon.go.kr/posts/329/1026>(접속일 2017.12.22.)

⁹ https://ko.wikipedia.org/wiki/%EC%9D%B4%EC%82%B0%ED%99%94_%ED%99%A9(접속일 2017.12.22.)

CO (일산화탄소) 10	가연성, 독성의 기체 가정의 도시가스, 공업 지대에서 발생 산소보다 헤모글로빈과의 친화력이 200배 정도 좋아서 소량 호흡시에도 문제 발생 가능
-----------------------------------	--

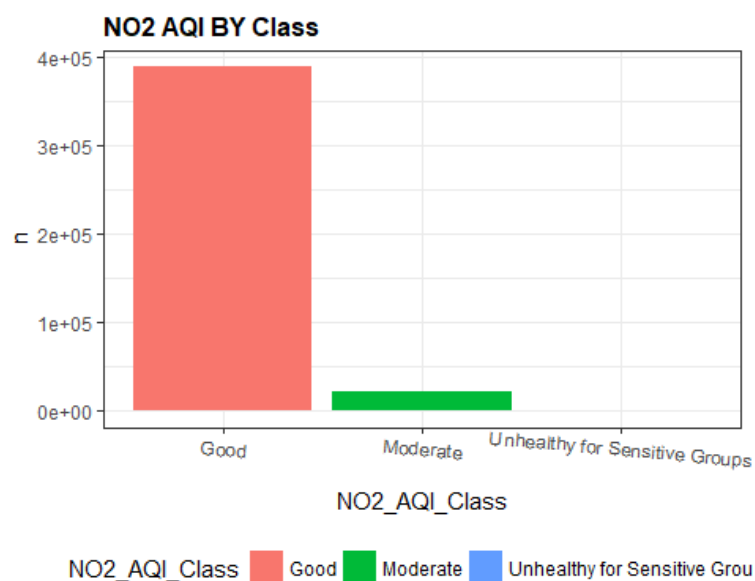
네 물질 모두 인체에 매우 유해하며 흡입 시 사망까지 가능하다는 것을 확인했다.

[AQI Class 확인]

이제, 앞서 추가한 AQI_Class 변수를 이용하여 오염물질의 심각성을 비교해보고자 한다. 오염물질 별로 단위가 다르기 때문에 통일된 공기 품질 지표인 AQI를 이용하여 확인했다. 네 오염물질별로 먼저 AQI_Class 별로 막대 그래프를 그리고, AQI 범주가 "Good" 또는 "Moderate"가 아닌 범주에 대해서는 더 자세히 살펴보았다. 이들은 이상치로 칭했고, NO2 부터 살펴보았다.

심각성의 기준은 16p 표 참고.

```
#NO
Pollution_NO2 %>% group_by(NO2_AQI_Class) %>% summarise(n=n()) %>%
ggplot(aes(NO2_AQI_Class, n)) + geom_bar(aes(fill = NO2_AQI_Class), stat = "identity") +
theme_bw() + labs(title = "NO2 AQI BY Class") + theme(axis.text.x = element_text(angle = -
5), legend.position = "bottom", plot.title = element_text(face = "bold", size = 12))
```



```
#이상치 관측 위치 확인
Pollution_NO2 %>% filter(NO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County) %>% summarise(AQI_mean = mean(NO2_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 34 x 3
## # Groups:   State [16]
##       State      County AQI_mean
```

¹⁰ https://ko.wikipedia.org/wiki/%EC%9D%BC%EC%82%B0%ED%99%94_%ED%83%84%EC%86%8C(접속일 2017.12.22.)


```
##           <chr>           <chr>    <dbl>
## 1    Michigan           Wayne 128.0000
## 2 Pennsylvania          York 117.0000
## 3     Arizona           Maricopa 114.1754
## 4     Florida           Orange 109.0000
## 5    California          Riverside 106.5000
## 6     Illinois           Cook 106.5000
## 7    Connecticut        Litchfield 106.0000
## 8    Louisiana East Baton Rouge 105.5000
## 9     Colorado           Denver 105.0000
## 10   Illinois           Saint Clair 105.0000
## # ... with 24 more rows

#이상치 관측 날짜 확인
Pollution_NO2 %>% filter(NO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(Date_Local) %>% summarise(AQI_mean = mean(NO2_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 413 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2000-01-10     132
## 2 2002-01-11     131
## 3 2001-12-08     129
## 4 2000-01-19     128
## 5 2002-06-27     128
## 6 2000-01-11     127
## 7 2000-02-04     125
## 8 2000-01-16     124
## 9 2000-01-20     123
## 10 2000-01-21     123
## # ... with 403 more rows

#이상치 관측 위치 + 날짜 확인
Pollution_NO2 %>% filter(NO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County, Date_Local) %>% summarise(AQI_mean = mean(NO2_AQI)) %>%
arrange(desc(AQI_mean))

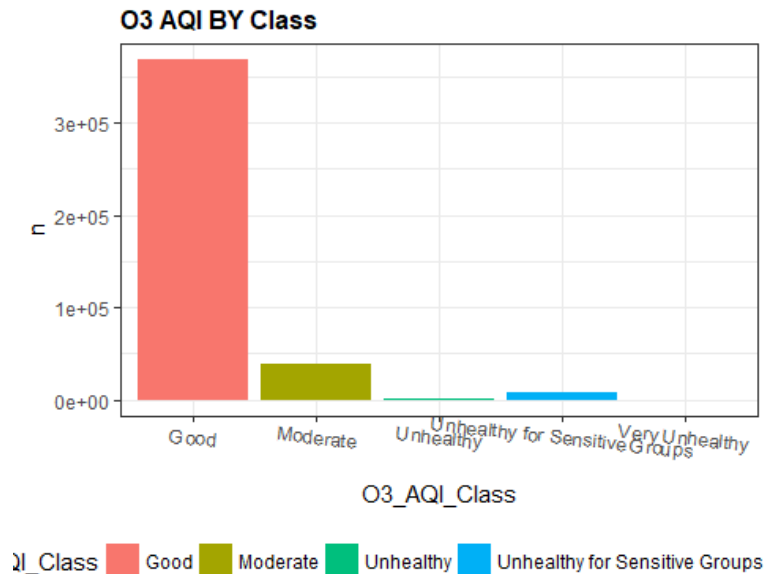
## # A tibble: 494 x 4
## # Groups:   State, County [34]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1  Arizona   Maricopa 2000-01-10     132
## 2 California Los Angeles 2002-01-11     131
## 3 California Los Angeles 2001-12-08     129
## 4  Arizona   Maricopa 2000-01-19     128
## 5  Michigan   Wayne 2002-06-27     128
## 6  Arizona   Maricopa 2000-01-11     127
## 7  Arizona   Maricopa 2000-02-09     126
## 8  Arizona   Maricopa 2000-02-04     125
## 9  Arizona   Maricopa 2000-01-12     124
## 10 Arizona   Maricopa 2000-01-15     124
## # ... with 484 more rows
```

NO2의 경우, 대부분 "Good" 범주에 속하며 일부만 매우 일부만 "Unhealthy for Sensitive Groups" 범주에서 관측되었다. "Unhealthy for Sensitive Groups"가 관측된 위치와 그곳의 AQI_mean을 출력했다. 그 결과 Michigan 주의 Wayne 군에서 가장 높은 AQI_mean을 보였다. 다시, "Unhealthy for Sensitive Groups"가 관측된 날짜와 그곳의 AQI_mean을 출력했다. 그 결과 2000년 1월 10일에 가장 높은 AQI_mean을 보였으며 전반적으로 2000년도에 높은 AQI_mean이 출력되었다. 이번엔 "Unhealthy for Sensitive Groups"가 관측된 위치, 날짜와 함께 AQI_mean을 출력했다. 그

결과 날짜는 같지만 위치는 다르게 2000 년 1 월 10 일 Arizonz 주의 Maricopa 에서 가장 높은 AQI_mean 이 출력되었다.

두 번째로, O3 를 같은 방법으로 살펴보았다.

```
Pollution_O3 %>% filter(!is.na(O3_AQI)) %>% group_by(O3_AQI_Class) %>%
summarise(n=n()) %>% ggplot(aes(O3_AQI_Class, n)) + geom_bar(aes(fill = O3_AQI_Class), stat =
"identity") + theme_bw() + labs(title = "O3 AQI BY Class") + theme(axis.text.x =
element_text(angle = -6), legend.position = "bottom", plot.title = element_text(face =
"bold", size = 12))
```



이상치 관측 위치 확인

O3_AQI_Class = Unhealthy for Sensitive

```
Pollution_O3 %>% filter(O3_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County) %>% summarise(AQI_mean = mean(O3_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 113 x 3
## # Groups:   State [41]
##       State      County AQI_mean
##       <chr>      <chr>   <dbl>
## 1 Louisiana Jefferson 122.0000
## 2 Tennessee      Meigs 120.6176
## 3 California Santa Clara 118.5000
## 4 Kentucky      Boyd 118.1475
## 5 Ohio          Cuyahoga 118.0000
## 6 Michigan      Wayne 117.9600
## 7 California Riverside 117.7892
## 8 California      Fresno 116.8867
## 9 Pennsylvania Northampton 116.8615
## 10 Connecticut Fairfield 116.7143
## # ... with 103 more rows
```

O3_AQI_Class = Unhealthy

```
Pollution_O3 %>% filter(O3_AQI_Class == "Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(O3_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 65 x 3
## # Groups:   State [24]
##       State      County AQI_mean
##       <chr>      <chr>   <dbl>
```

```
## 1 Maryland Baltimore 173.8000
## 2 Indiana Marion 173.5000
## 3 Pennsylvania Bucks 171.4062
## 4 California Imperial 170.8889
## 5 New Jersey Camden 169.7586
## 6 Louisiana East Baton Rouge 169.5000
## 7 Missouri St. Louis City 168.9000
## 8 Pennsylvania Northampton 168.7273
## 9 District Of Columbia District of Columbia 168.7000
## 10 Illinois Saint Clair 168.3000
## # ... with 55 more rows
```

#03_AQI_Class = Very Unhealthy

```
Pollution_03 %>% filter(03_AQI_Class == "Very Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(03_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 24 x 3
## # Groups:   State [12]
##           State County AQI_mean
##           <chr>   <chr>   <dbl>
## 1 Country Of Mexico BAJA CALIFORNIA NORTE 211.00
## 2 District Of Columbia District of Columbia 206.00
## 3 California Fresno 205.25
## 4 North Carolina Mecklenburg 205.00
## 5 Pennsylvania Montgomery 204.50
## 6 California San Bernardino 204.30
## 7 California Los Angeles 204.00
## 8 Pennsylvania Bucks 203.60
## 9 Pennsylvania Allegheny 203.00
## 10 Pennsylvania Beaver 203.00
## # ... with 14 more rows
```

#이상치 출력 날짜 확인

#03_AQI_Class = Unhealthy for Sensitive

```
Pollution_03 %>% filter(03_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(Date_Local) %>% summarise(AQI_mean = mean(03_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 2,080 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2000-09-04 147.0
## 2 2000-09-17 147.0
## 3 2007-07-06 147.0
## 4 2015-09-30 147.0
## 5 2001-09-16 145.0
## 6 2008-05-15 145.0
## 7 2008-10-29 145.0
## 8 2011-08-07 145.0
## 9 2012-09-08 145.0
## 10 2008-08-02 143.5
## # ... with 2,070 more rows
```

#03_AQI_Class = Unhealthy

```
Pollution_03 %>% filter(03_AQI_Class == "Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(03_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 522 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2004-06-05 197
## 2 2006-06-03 196
## 3 2000-06-13 195
## 4 2003-06-29 195
## 5 2003-08-10 195
```

```

## 6 2004-09-01      195
## 7 2006-06-04      195
## 8 2007-08-02      195
## 9 2013-06-01      195
## 10 2013-06-29     195
## # ... with 512 more rows

#03_AQI_Class = Very Unhealthy
Pollution_03 %>% filter(O3_AQI_Class == "Very Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(O3_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 64 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2013-06-29     218
## 2 2007-07-04     211
## 3 2003-08-17     210
## 4 2008-06-27     207
## 5 2002-07-02     206
## 6 2002-07-18     206
## 7 2003-05-31     206
## 8 2005-05-22     206
## 9 2006-07-22     206
## 10 2008-07-09     206
## # ... with 54 more rows

#이상치 관측 위치 + 날짜 확인
#03_AQI_Class = Unhealthy for Sensitive
Pollution_03 %>% filter(O3_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County, Date_Local) %>% summarise(AQI_mean = mean(O3_AQI)) %>%
arrange(desc(AQI_mean))

## # A tibble: 7,562 x 4
## # Groups:   State, County [113]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1 Alabama Jefferson 2015-08-04     147
## 2 California Fresno 2007-07-06     147
## 3 California Fresno 2008-07-26     147
## 4 California Fresno 2008-09-05     147
## 5 California Fresno 2011-09-07     147
## 6 California Imperial 2001-07-02     147
## 7 California Imperial 2007-06-15     147
## 8 California Imperial 2007-07-04     147
## 9 California Los Angeles 2000-09-17     147
## 10 California Los Angeles 2002-07-09     147
## # ... with 7,552 more rows

#03_AQI_Class = Unhealthy
Pollution_03 %>% filter(O3_AQI_Class == "Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(O3_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 890 x 4
## # Groups:   State, County [65]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1 California Los Angeles 2006-07-15     197
## 2 California Riverside 2003-05-28     197
## 3 California Riverside 2003-07-15     197
## 4 California Riverside 2004-06-05     197
## 5 California Riverside 2006-06-03     197
## 6 California Riverside 2014-09-13     197
## 7 California Sacramento 2002-08-11     197
## 8 District Of Columbia District of Columbia 2002-06-25     197

```

```
## 9 Louisiana East Baton Rouge 2003-07-18 197
## 10 New Jersey Camden 2001-08-07 197
## # ... with 880 more rows

#03_AQI_Class = Very Unhealthy
Pollution_03 %>% filter(03_AQI_Class == "Very Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(03_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 75 x 4
## # Groups: State, County [24]
## State County Date_Local AQI_mean
## <chr> <chr> <date> <dbl>
## 1 California San Bernardino 2013-06-29 218
## 2 Country Of Mexico BAJA CALIFORNIA NORTE 2007-07-04 211
## 3 California Riverside 2003-08-17 210
## 4 California Fresno 2008-06-27 207
## 5 Pennsylvania Bucks 2000-06-10 207
## 6 Pennsylvania Montgomery 2000-06-10 207
## 7 California Fresno 2008-07-09 206
## 8 California Los Angeles 2006-07-22 206
## 9 California Riverside 2005-05-22 206
## 10 California San Bernardino 2009-07-18 206
## # ... with 65 more rows
```

O3 의 경우, 대부분 "Good" 범주에 속하지만 "Unhealthy for Sensitive Groups" 범주도 꽤 보였고, 심지어 "Unhealthy"와 "Very Unhealthy" 범주도 관측되었다.

"Unhealthy for Sensitive Groups"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Louisiana 주의 Jefferson 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy for Sensitive Groups"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2000 년 9 월 4 일에 가장 높은 AQI_mean 을 보였으며 NO2 와 다르게 최근에도 높은 AQI_mean 이 관측되었다. 이번엔 "Unhealthy for Sensitive Groups"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2015 년 8 월 4 일 Alabama 주의 Jefferson 에서 가장 높은 AQI_mean 이 출력되었다.

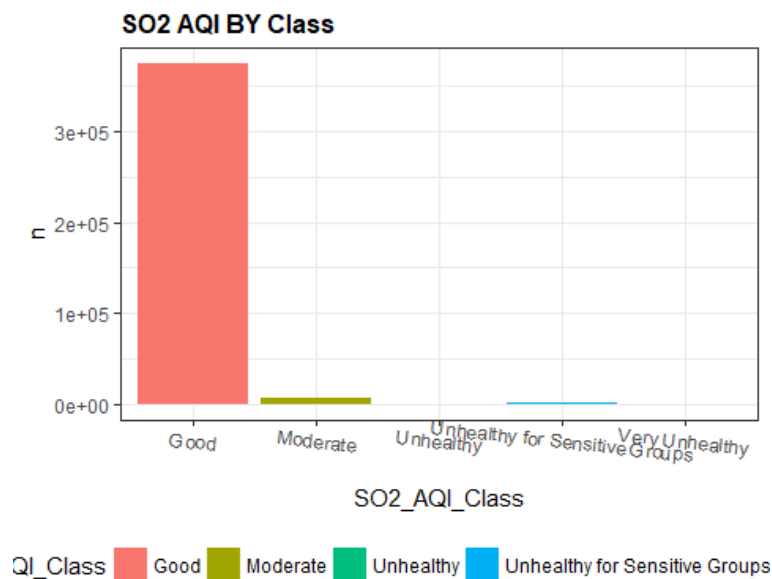
"Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Maryland 주의 Baltimore 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2004 년 6 월 5 일에 가장 높은 AQI_mean 을 보였으며 NO2 와 다르게 높은 AQI_mean 이 출력된 연도가 다양했다. 이번엔 "Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2006 년 7 월 15 일 California 주의 Los Angeles 에서 가장 높은 AQI_mean 이 출력되었다.

"Very Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Country of Mexico 의 BAJA CALIFORNIA NORTE 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Very Unhealthy"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2013 년 6 월 29 일에 가장 높은 AQI_mean 을 보였으며 전반적으로 2000 년도에 높은 AQI_mean 이 출력되었다. 이번엔 "Very Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2013 년 6 월 29 일 California 주의 San Bernardino 에서 가장 높은 AQI_mean 이 출력되었다.

특이한 점은, 높은 AQI_mean 이 관측된 날씨가 주로 6, 7, 8 월, 즉 여름이라는 것이다. 계절별 오염물질 농도는 뒤에서 더 자세히 알아보았다.

세 번째로, SO2 를 같은 방법으로 살펴보았다.

```
#SO2
Pollution_SO2 %>% filter(!is.na(SO2_AQI) == TRUE) %>% group_by(SO2_AQI_Class) %>%
summarise(n=n()) %>% ggplot(aes(SO2_AQI_Class, n)) + geom_bar(aes(fill = SO2_AQI_Class),
stat = "identity") + theme_bw() + labs(title = "SO2 AQI BY Class") + theme(axis.text.x =
element_text(angle = -6), legend.position = "bottom", plot.title = element_text(face =
"bold", size = 12))
```



#이상치 관측 위치 확인

#SO2_AQI_Class = Unhealthy for Sensitive

```
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County) %>% summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 58 x 3
## # Groups:   State [23]
##       State                County AQI_mean
##       <chr>                <chr>    <dbl>
## 1 Country Of Mexico BAJA CALIFORNIA NORTE 132.0000
## 2 Pennsylvania Philadelphia 124.0000
## 3 Maryland Baltimore 119.9167
## 4 Ohio Athens 119.0000
## 5 Kentucky Henderson 118.4000
## 6 California Los Angeles 117.4000
## 7 Illinois Saint Clair 115.6696
## 8 North Carolina Forsyth 114.5000
## 9 New Jersey Essex 114.0000
## 10 Ohio Hamilton 114.0000
## # ... with 48 more rows
```

#SO2_AQI_Class = Unhealthy

```
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 13 x 3
## # Groups:   State [9]
##       State                County AQI_mean
```

```
##           <chr>           <chr>    <dbl>
## 1 Pennsylvania Lawrence  195.000
## 2 Colorado      Adams   176.000
## 3 New York      Queens  176.000
## 4 Texas         Harris  172.000
## 5 Pennsylvania Cambria  163.500
## 6 Illinois     Saint Clair 162.625
## 7 Indiana      Marion   161.000
## 8 Michigan     Wayne   161.000
## 9 Ohio         Hamilton 161.000
## 10 Pennsylvania Beaver   155.000
## 11 California  Imperial 153.000
## 12 Pennsylvania Berks    153.000
## 13 Pennsylvania York     152.500
```

#SO2_AQI_Class = Very Unhealthy

```
Pollution_SO2 %>% filter(SO2_AQI_Class == "Very Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 2 x 3
## # Groups:   State [2]
##       State      County AQI_mean
##       <chr>      <chr>    <dbl>
## 1 Illinois Saint Clair     200
## 2 Oklahoma  Cherokee     200
```

#이상치 관측 날짜 확인

#SO2_AQI_Class = Unhealthy for Sensitive

```
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(Date_Local) %>% summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 641 x 2
##   Date_Local AQI_mean
##   <date>      <dbl>
## 1 2007-04-21    148
## 2 2004-12-04    147
## 3 2015-12-17    147
## 4 2005-09-11    146
## 5 2002-01-25    145
## 6 2007-06-08    145
## 7 2000-09-01    144
## 8 2001-10-13    141
## 9 2002-12-15    140
## 10 2003-06-15    140
## # ... with 631 more rows
```

#SO2_AQI_Class = Unhealthy

```
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))
```

```
## # A tibble: 31 x 2
##   Date_Local AQI_mean
##   <date>      <dbl>
## 1 2001-06-27    195
## 2 2001-11-16    180
## 3 2001-11-15    177
## 4 2004-10-12    177
## 5 2002-08-11    176
## 6 2007-09-11    176
## 7 2001-10-04    173
## 8 2003-06-09    173
## 9 2000-10-14    172
## 10 2005-03-06    169
## # ... with 21 more rows
```

```

#SO2_AQI_Class = Very Unhealthy
Pollution_SO2 %>% filter(SO2_AQI_Class == "Very Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 2 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2002-02-14     200
## 2 2006-05-04     200

#이상치 관측 위치 + 날짜 확인
#SO2_AQI_Class = Unhealthy for Sensitive
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County, Date_Local) %>% summarise(AQI_mean = mean(SO2_AQI)) %>%
arrange(desc(AQI_mean))

## # A tibble: 728 x 4
## # Groups:   State, County [58]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1 Pennsylvania York 2005-08-15     148
## 2 Pennsylvania York 2007-04-21     148
## 3 California San Bernardino 2015-12-17     147
## 4 Illinois Saint Clair 2004-12-04     147
## 5 Michigan Wayne 2005-09-11     146
## 6 Illinois Saint Clair 2002-01-25     145
## 7 Illinois Saint Clair 2003-06-28     145
## 8 Maryland Baltimore 2007-06-08     145
## 9 Missouri Saint Louis 2000-09-01     144
## 10 California Los Angeles 2000-05-22     141
## # ... with 718 more rows

#SO2_AQI_Class = Unhealthy
Pollution_SO2 %>% filter(SO2_AQI_Class == "Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 31 x 4
## # Groups:   State, County [13]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1 Pennsylvania Lawrence 2001-06-27     195
## 2 Illinois Saint Clair 2001-11-16     180
## 3 Illinois Saint Clair 2001-11-15     177
## 4 Pennsylvania Cambria 2004-10-12     177
## 5 Colorado Adams 2007-09-11     176
## 6 New York Queens 2002-08-11     176
## 7 Illinois Saint Clair 2001-10-04     173
## 8 Illinois Saint Clair 2003-06-09     173
## 9 Texas Harris 2000-10-14     172
## 10 Illinois Saint Clair 2005-03-06     169
## # ... with 21 more rows

#SO2_AQI_Class = Very Unhealthy
Pollution_SO2 %>% filter(SO2_AQI_Class == "Very Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(SO2_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 2 x 4
## # Groups:   State, County [2]
##   State County Date_Local AQI_mean
##   <chr>   <chr>   <date>     <dbl>
## 1 Illinois Saint Clair 2002-02-14     200
## 2 Oklahoma Cherokee 2006-05-04     200

```

SO2의 경우, 대부분 "Good" 범주에 속하지만 다른 범주에서도 모두 관측되었다.

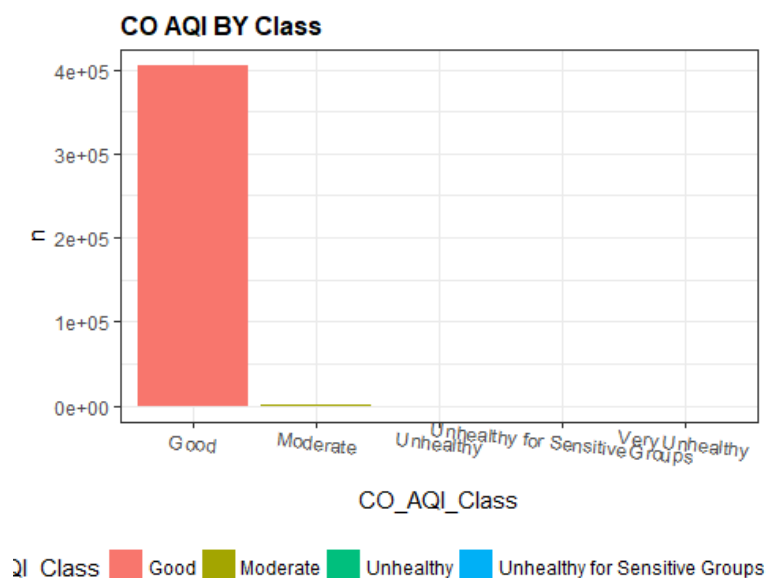
"Unhealthy for Sensitive Groups"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Country of Mexico 의 BAJA CALIFORNIA NORTE 에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy for Sensitive Groups"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2007 년 4 월 21 일에 가장 높은 AQI_mean 을 보였다. 이번엔 "Unhealthy for Sensitive Groups"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2005 년 8 월 15 일 Pennsylvania 주의 York 에서 가장 높은 AQI_mean 이 출력되었다.

"Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Pennsylvania 주의 Lawrence 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2001 년 6 월 27 일에 가장 높은 AQI_mean 을 보였다. 이번엔 "Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 위치와 날짜 모두 같은 2001 년 6 월 27 일 Pennsylvania 주의 Lawrence 군에서 가장 높은 AQI_mean 이 출력되었다.

"Very Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 Illinois 주의 Saint Clair 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Very Unhealthy"가 관측된 날짜와 그 곳의 AQI_mean 을 출력했다. 그 결과 2007 년 4 월 21 일에 가장 높은 AQI_mean 을 보였으며 전반적으로 2000 년도에 높은 AQI_mean 이 출력되었다. 이번엔 "Very Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2002 년 2 월 14 일 Illinois 주의 Saint Clair 군에서 가장 높은 AQI_mean 이 출력되었다.

마지막으로, CO 를 같은 방법으로 살펴보았다.

```
#CO
Pollution_CO %>% filter(!is.na(CO_AQI) == TRUE) %>% group_by(CO_AQI_Class) %>%
summarise(n=n()) %>% ggplot(aes(CO_AQI_Class, n)) + geom_bar(aes(fill = CO_AQI_Class), stat
= "identity") + theme_bw() + labs(title = "CO AQI BY Class") + theme(axis.text.x =
element_text(angle = -6), legend.position = "bottom", plot.title = element_text(face =
"bold", size = 12))
```



```
#이상치 관측 위치 확인
#CO_AQI_Class = Unhealthy for Sensitive
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County) %>% summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))
```

```

## # A tibble: 2 x 3
## # Groups:   State [2]
##       State      County AQI_mean
##       <chr>      <chr>    <dbl>
## 1      California Imperial 119.0833
## 2 Country Of Mexico BAJA CALIFORNIA NORTE 117.6667

#CO_AQI_Class = Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 2 x 3
## # Groups:   State [2]
##       State      County AQI_mean
##       <chr>      <chr>    <dbl>
## 1      California Imperial 171.6667
## 2 Country Of Mexico BAJA CALIFORNIA NORTE 150.0000

#CO_AQI_Class = Very Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Very Unhealthy") %>% group_by(State, County) %>%
summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 1 x 3
## # Groups:   State [1]
##       State      County AQI_mean
##       <chr>      <chr>    <dbl>
## 1 California Imperial      201

#이상치 관측 날짜 확인
#CO_AQI_Class = Unhealthy for Sensitive
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(Date_Local) %>% summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 17 x 2
##   Date_Local AQI_mean
##   <date>      <dbl>
## 1 2006-01-07     143
## 2 2001-12-14     138
## 3 2000-11-21     136
## 4 2002-01-14     136
## 5 2006-11-17     135
## 6 2000-01-09     126
## 7 2002-11-07     126
## 8 2001-10-20     118
## 9 2006-01-14     118
## 10 2004-01-30     115
## 11 2002-12-04     113
## 12 2001-10-15     106
## 13 2001-12-13     106
## 14 2006-02-14     106
## 15 2000-01-15     103
## 16 2006-11-16     103
## 17 2006-10-29     101

#CO_AQI_Class = Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 4 x 2
##   Date_Local AQI_mean
##   <date>      <dbl>
## 1 2000-11-26     183
## 2 2000-12-21     173

```

```

## 3 2000-11-22      159
## 4 2006-12-21      150

#CO_AQI_Class = Very Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Very Unhealthy") %>% group_by(Date_Local) %>%
summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 1 x 2
##   Date_Local AQI_mean
##   <date>     <dbl>
## 1 2000-12-20      201

#이상치 관측 위치 + 날짜 확인
#CO_AQI_Class = Unhealthy for Sensitive
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy for Sensitive Groups") %>%
group_by(State, County, Date_Local) %>% summarise(AQI_mean = mean(CO_AQI)) %>%
arrange(desc(AQI_mean))

## # A tibble: 18 x 4
## # Groups:   State, County [2]
##       State      County Date_Local AQI_mean
##       <chr>      <chr>    <date>    <dbl>
## 1 Country Of Mexico BAJA CALIFORNIA NORTE 2006-01-07      143
## 2 California        Imperial 2001-12-14      138
## 3 California        Imperial 2000-11-21      136
## 4 California        Imperial 2002-01-14      136
## 5 Country Of Mexico BAJA CALIFORNIA NORTE 2006-11-17      135
## 6 California        Imperial 2000-01-09      126
## 7 California        Imperial 2002-11-07      126
## 8 California        Imperial 2001-10-20      118
## 9 Country Of Mexico BAJA CALIFORNIA NORTE 2006-01-14      118
## 10 California        Imperial 2004-01-30      115
## 11 California        Imperial 2002-12-04      113
## 12 California        Imperial 2001-10-15      106
## 13 California        Imperial 2001-12-13      106
## 14 California        Imperial 2006-02-14      106
## 15 Country Of Mexico BAJA CALIFORNIA NORTE 2006-02-14      106
## 16 California        Imperial 2000-01-15      103
## 17 Country Of Mexico BAJA CALIFORNIA NORTE 2006-11-16      103
## 18 Country Of Mexico BAJA CALIFORNIA NORTE 2006-10-29      101

#CO_AQI_Class = Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 4 x 4
## # Groups:   State, County [2]
##       State      County Date_Local AQI_mean
##       <chr>      <chr>    <date>    <dbl>
## 1 California        Imperial 2000-11-26      183
## 2 California        Imperial 2000-12-21      173
## 3 California        Imperial 2000-11-22      159
## 4 Country Of Mexico BAJA CALIFORNIA NORTE 2006-12-21      150

#CO_AQI_Class = Very Unhealthy
Pollution_CO %>% filter(CO_AQI_Class == "Very Unhealthy") %>% group_by(State, County,
Date_Local) %>% summarise(AQI_mean = mean(CO_AQI)) %>% arrange(desc(AQI_mean))

## # A tibble: 1 x 4
## # Groups:   State, County [1]
##       State      County Date_Local AQI_mean
##       <chr>      <chr>    <date>    <dbl>
## 1 California Imperial 2000-12-20      201

```

SO₂ 의 경우, 대부분 "Good" 범주에 속하고 다른 범주에서 관측되긴 했지만 CO 의 레코드수가 적지 않은 만큼 이는 다른 오염물질에 비해 굉장히 적은 수치로 볼 수 있다.

"Unhealthy for Sensitive Groups"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 California 주의 Imperial 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy for Sensitive Groups"가 관측된 날짜와 그곳의 AQI_mean 을 출력했다. 그 결과 2007 년 1 월 7 일에 가장 높은 AQI_mean 을 보였다. 이번엔 "Unhealthy for Sensitive Groups"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2006 년 1 월 7 일 Country of Mexico 의 BAJA CALIFORNIA NORTE 에서 가장 높은 AQI_mean 이 출력되었다.

"Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 California 주의 Imperial 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Unhealthy"가 관측된 날짜와 그곳의 AQI_mean 을 출력했다. 그 결과 2000 년 11 월 26 일에 가장 높은 AQI_mean 을 보였다. 이번엔 "Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 위치와 날짜 모두 같은 2000 년 11 월 26 일 California 주의 Imperial 군에서 가장 높은 AQI_mean 이 출력되었다.

"Very Unhealthy"가 관측된 위치와 그곳의 AQI_mean 을 출력했다. 그 결과 California 주의 Imperial 군에서 가장 높은 AQI_mean 을 보였다. 다시, "Very Unhealthy"가 관측된 날짜와 그곳의 AQI_mean 을 출력했다. 그 결과 2000 년 12 월 20 일에 가장 높은 AQI_mean 을 보였다. 이번엔 "Very Unhealthy"가 관측된 위치, 날짜와 함께 AQI_mean 을 출력했다. 그 결과 2000 년 12 월 20 일 California 주의 Imperial 군에서 가장 높은 AQI_mean 이 출력되었다.

CO 의 경우 특히 California 주의 Imperial 군에서 심각한 수치를 보인 것으로 나타났다.

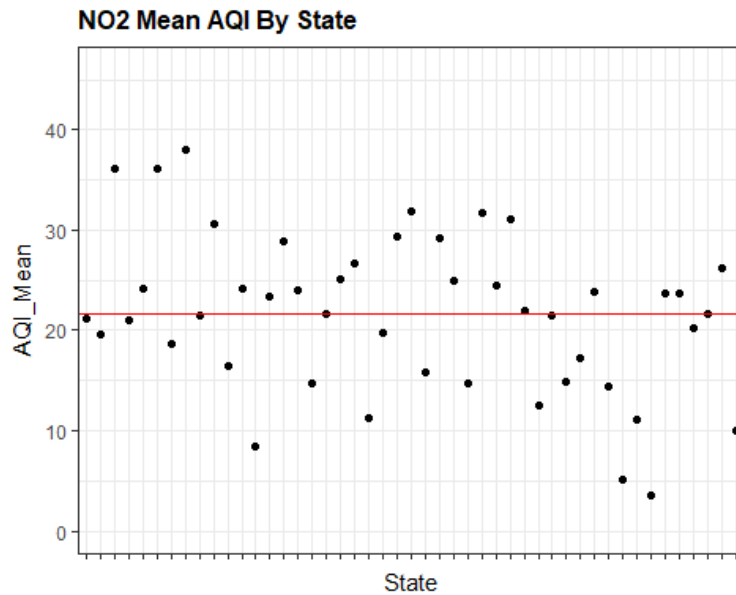
AQI 에 관한 조사를 하며 발견한 흥미로운 사이트¹¹도 첨부한다. 이 사이트에서 전 세계 AQI 를 실시간으로 확인할 수 있으며 미국의 경우 항상 California 지역의 AQI 가 높게 측정되고 있다는 것을 한 눈에 파악할 수 있다.

[지역별 AQI 지표 확인]

다음으로, AQI 의 지역별 차이를 알아보기 위해 산점도를 그렸다. 각 오염물질의 AQI 평균 또한 나타내어 지역별 AQI 의 비교를 좀더 용이하게 했다.

```
Pollution_N02 %>% group_by(State) %>% summarise(AQI_Mean = mean(N02_AQI)) %>%  
ggplot(aes(State, AQI_Mean)) + geom_point() + geom_hline(aes(yintercept = mean(AQI_Mean)),  
color = "red") + theme_bw() + labs(title = "N02 Mean AQI By State") + theme(plot.title =  
element_text(face = "bold", size = 12), axis.text.x = element_blank())
```

¹¹ <http://aqicn.org/here/kr/>(접속일 2017.12.22.)



#평균에 비해 훨씬 높은 AQI_Mean 이 35 이상인 경우와, AQI_Mean 이 10 이하인 경우만 살펴보면,
 Pollution_NO2 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(NO2_AQI)) %>%
filter(AQI_Mean > 35) %>% **arrange**(**desc**(AQI_Mean))

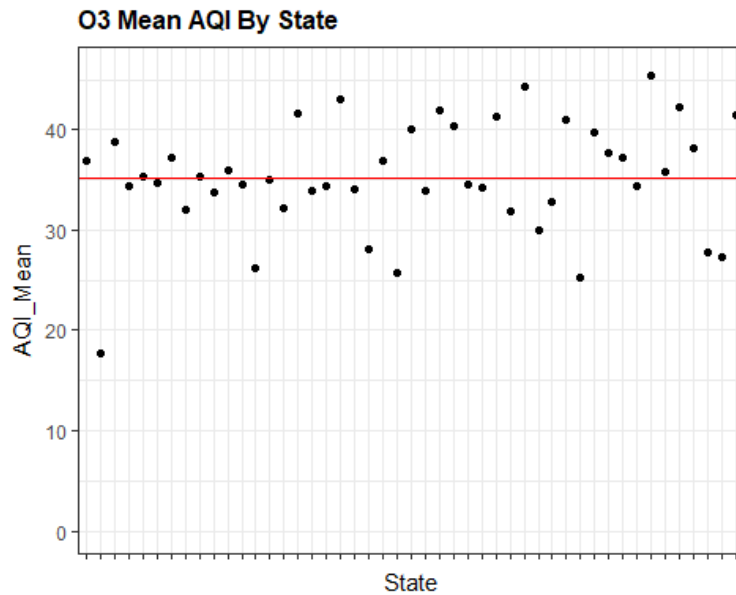
```
## # A tibble: 3 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Country Of Mexico 37.96057
## 2 Arizona 36.16699
## 3 Colorado 36.07811
```

Pollution_NO2 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(NO2_AQI)) %>%
filter(AQI_Mean < 10) %>% **arrange**(AQI_Mean)

```
## # A tibble: 4 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Tennessee 3.585349
## 2 South Carolina 5.126671
## 3 Hawaii 8.496006
## 4 Wyoming 9.974735
```

두 번째로, O3 를 살펴보았다.

```
#O3
Pollution_O3 %>% group_by(State) %>% summarise(AQI_Mean = mean(O3_AQI, na.rm = TRUE)) %>%
ggplot(aes(State, AQI_Mean)) + geom_point() + geom_hline(aes(yintercept = mean(AQI_Mean,
na.rm = TRUE)), color = "red") + theme_bw() + labs(title = "O3 Mean AQI By State") +
theme(plot.title = element_text(face = "bold", size = 12), axis.text.x = element_blank())
```



#평균에 비해 훨씬 높은 AQI_Mean 이 40 이상인 경우와, AQI_Mean 이 30 이하인 경우만 살펴보면,
 Pollution_O3 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(O3_AQI, **na.rm** = **TRUE**)) %>%
filter(AQI_Mean > 40) %>% **arrange**(**desc**(AQI_Mean))

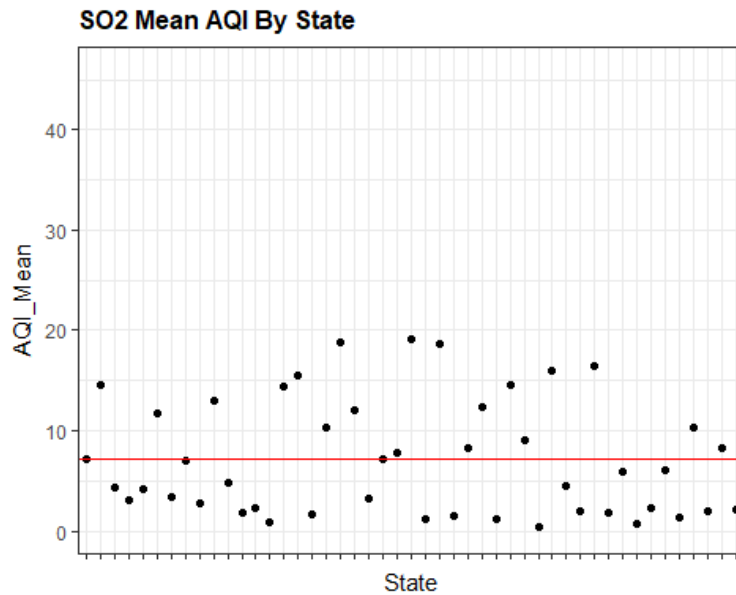
```
## # A tibble: 11 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Tennessee 45.35841
## 2 North Carolina 44.35932
## 3 Kentucky 42.96436
## 4 Utah 42.23614
## 5 Missouri 41.97449
## 6 Indiana 41.59765
## 7 Wyoming 41.45067
## 8 New Mexico 41.33240
## 9 Oklahoma 41.03965
## 10 Nevada 40.38662
## 11 Michigan 40.02296
```

Pollution_O3 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(O3_AQI, **na.rm** = **TRUE**)) %>%
filter(AQI_Mean < 30) %>% **arrange**(AQI_Mean)

```
## # A tibble: 8 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Alaska 17.74848
## 2 Oregon 25.22978
## 3 Massachusetts 25.69598
## 4 Hawaii 26.28520
## 5 Wisconsin 27.32190
## 6 Washington 27.70539
## 7 Maine 28.07224
## 8 North Dakota 29.97895
```

세 번째로, SO2 를 살펴보았다.

```
#SO2
Pollution_SO2 %>% group_by(State) %>% summarise(AQI_Mean = mean(SO2_AQI, na.rm = TRUE)) %>%
ggplot(aes(State, AQI_Mean)) + geom_point() + geom_hline(aes(yintercept = mean(AQI_Mean,
na.rm = TRUE)), color = "red") + theme_bw() + labs(title = "SO2 Mean AQI By State") +
theme(plot.title = element_text(face = "bold", size = 12), axis.text.x = element_blank())
```



#평균에 비해 훨씬 높은 AQI_Mean 이 15 이상인 경우와, AQI_Mean 이 2 이하인 경우만 살펴보면,
 Pollution_SO2 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(SO2_AQI, na.rm = TRUE)) %>%
filter(AQI_Mean > 15) %>% **arrange**(**desc**(AQI_Mean))

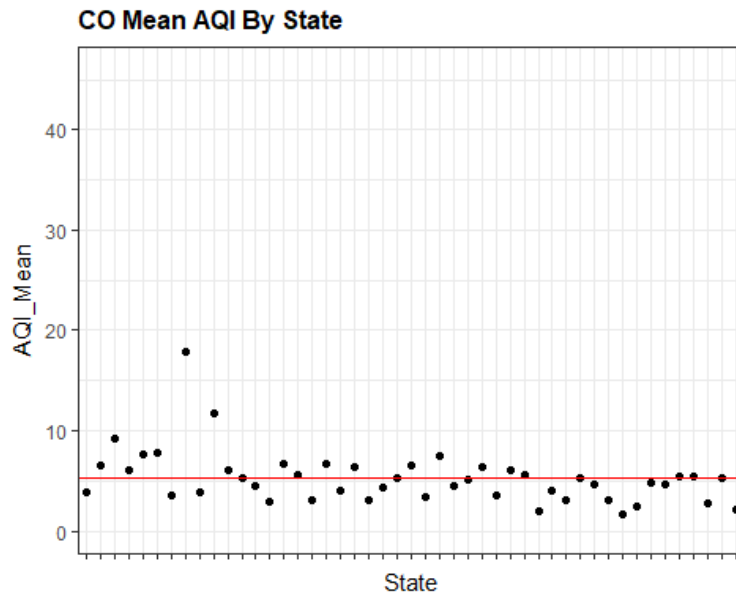
```
## # A tibble: 6 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1  Michigan 19.20731
## 2  Kentucky 18.76802
## 3  Missouri 18.59471
## 4 Pennsylvania 16.40746
## 5    Ohio 16.06159
## 6   Indiana 15.47681
```

Pollution_SO2 %>% **group_by**(State) %>% **summarise**(AQI_Mean = **mean**(SO2_AQI, na.rm = TRUE)) %>%
filter(AQI_Mean < 2) %>% **arrange**(AQI_Mean)

```
## # A tibble: 11 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 North Dakota 0.4586193
## 2 South Dakota 0.6722973
## 3    Idaho 0.8452915
## 4 New Mexico 1.1937677
## 5  Minnesota 1.2041284
## 6    Utah 1.4225272
## 7   Nevada 1.4758507
## 8    Iowa 1.6709903
## 9 Rhode Island 1.7680965
## 10   Georgia 1.7944070
## 11 Washington 1.9541667
```

마지막으로, CO 를 살펴보았다.

```
#CO
Pollution_CO %>% group_by(State) %>% summarise(AQI_Mean = mean(CO_AQI, na.rm = TRUE)) %>%
ggplot(aes(State, AQI_Mean)) + geom_point() + geom_hline(aes(yintercept = mean(AQI_Mean,
na.rm = TRUE)), color = "red") + theme_bw() + labs(title = "CO Mean AQI By State") +
theme(plot.title = element_text(face = "bold", size = 12), axis.text.x = element_blank())
```



#평균에 비해 훨씬 높은 AQI_Mean 이 10 이상인 경우와, AQI_Mean 이 3 이하인 경우만 살펴보면,
 Pollution_CO %>% group_by(State) %>% summarise(AQI_Mean = mean(CO_AQI, na.rm = TRUE)) %>%
 filter(AQI_Mean > 10) %>% arrange(desc(AQI_Mean))

```
## # A tibble: 2 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Country Of Mexico 17.82587
## 2 District Of Columbia 11.70767
```

Pollution_CO %>% group_by(State) %>% summarise(AQI_Mean = mean(CO_AQI, na.rm = TRUE)) %>%
 filter(AQI_Mean < 3) %>% arrange(AQI_Mean)

```
## # A tibble: 6 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 South Carolina 1.720395
## 2 North Dakota 2.053358
## 3 Wyoming 2.127305
## 4 South Dakota 2.386692
## 5 Washington 2.730290
## 6 Idaho 2.894505
```

#이상치 확인

Pollution_CO %>% group_by(State) %>% summarise(AQI_Mean = mean(CO_AQI, na.rm = TRUE)) %>%
 filter(AQI_Mean > 15) %>% arrange(desc(AQI_Mean))

```
## # A tibble: 1 x 2
##       State AQI_Mean
##       <chr>   <dbl>
## 1 Country Of Mexico 17.82587
```

AQI의 경우 네 오염물질의 단위를 맞추기 힘든 상황에서 비교하기 매우 좋은 지표이다. 이번 네 그림에서는 y축의 limit을 동일하게 줌으로서, 대강의 점들이 어디에 분포해있는지 알 수 있었다. O3가 전반적으로 굉장히 높은 수치를 차지했다. 그 다음으로 높은 물질은 NO2였고, SO2와 CO는 다행이도 다른 두 오염물질에 비해 심각성이 덜한 편이었다. 하지만 이 둘은 오염물질 자체의 독성이 굉장한 만큼 계속해서 예의주시해야하는 물질이다.

【 DATE 】

[계절별]

네 오염물질에 대해 각각 계절별로 분석해보기 위해 spring/summer/fall/winter 로 나눈 새로운 데이터셋을 만들었다. (3-5 월 = 봄 / 6-8 월 = 여름 / 9-11 월 = 가을 / 12-2 월 = 겨울)

```
N02_spring<-Pollution_N02 %>% filter(Month %in% c(3:5)) %>% distinct()
N02_summer<-Pollution_N02 %>% filter(Month %in% c(6:8)) %>% distinct()
N02_fall<-Pollution_N02 %>% filter(Month %in% c(9:11)) %>% distinct()
N02_winter<-Pollution_N02 %>% filter(Month %in% c(1:2,12)) %>% distinct()
```

```
S02_spring<-Pollution_S02 %>% filter(Month %in% c(3:5)) %>% distinct()
S02_summer<-Pollution_S02 %>% filter(Month %in% c(6:8)) %>% distinct()
S02_fall<-Pollution_S02 %>% filter(Month %in% c(9:11)) %>% distinct()
S02_winter<-Pollution_S02 %>% filter(Month %in% c(1:2,12)) %>% distinct()
```

```
O3_spring<-Pollution_O3 %>% filter(Month %in% c(3:5)) %>% distinct()
O3_summer<-Pollution_O3 %>% filter(Month %in% c(6:8)) %>% distinct()
O3_fall<-Pollution_O3 %>% filter(Month %in% c(9:11)) %>% distinct()
O3_winter<-Pollution_O3 %>% filter(Month %in% c(1:2,12)) %>% distinct()
```

```
CO_spring<-Pollution_CO %>% filter(Month %in% c(3:5)) %>% distinct()
CO_summer<-Pollution_CO %>% filter(Month %in% c(6:8)) %>% distinct()
CO_fall<-Pollution_CO %>% filter(Month %in% c(9:11)) %>% distinct()
CO_winter<-Pollution_CO %>% filter(Month %in% c(1,2,12)) %>% distinct()
```

2000 년부터 2010 년까지의 각 날짜별로 관측한 횟수의 차이를 살펴보기 위해 count 를 세서 내림차순으로 정렬해보니, 2002 년 6 월 10 일 이 640 건으로 가장 많았고, 2001 년 2 월 12 일 이 136 건으로 가장 적었다.

최대값과 최소값의 차이가 약 500 건으로 매우 크기 때문에 앞으로의 날짜별 자료분석에 있어서 mean 값을 취하여 살펴보는 것이 더 적절할 것이라고 판단했다.

```
Pollution %>% group_by(Year,Month,Day) %>% summarise(count=n()) %>% arrange(desc(count))
```

```
## # A tibble: 5,996 x 4
## # Groups:   Year, Month [197]
##   Year Month   Day count
##   <int> <int> <int> <int>
## 1 2002     6    10  640
## 2 2002     6     9  636
## 3 2011     5    24  452
## 4 2008     6    13  416
## 5 2008     6    12  408
## 6 2013     4    17  408
## 7 2013     4    16  406
## 8 2008     7    10  400
## 9 2013     4     8  400
## 10 2013     4    15  400
## # ... with 5,986 more rows
```

```
Pollution %>% group_by(Year,Month,Day) %>% summarise(count=n()) %>% arrange(count)
```

```
## # A tibble: 5,996 x 4
## # Groups:   Year, Month [197]
##   Year Month Day count
##   <int> <int> <int> <int>
## 1  2016     5     1    28
## 2  2016     5     2    28
## 3  2016     5     3    28
## 4  2016     5     4    28
## 5  2016     5     5    28
## 6  2016     5     6    28
## 7  2016     5     7    28
## 8  2016     5     8    28
## 9  2016     5     9    28
## 10 2016     5    10    28
## # ... with 5,986 more rows
```

[각 날짜별 최대/최소값 찾기]

이번에는 각 날짜별 최대 & 최소값을 찾아보았다.

```
Pollution_NO2 %>% filter(NO2_Mean > 0) %>% mutate(mean_max=max(NO2_Mean),mean_min=min(NO2_Mean)) %>% select(Date_Local,mean_max,mean_min,everything()) %>% distinct()
```

```
## # A tibble: 411,471 x 18
##   Date_Local mean_max mean_min State_Code County_Code Site_Num
##   <date>     <dbl>     <dbl>     <int>     <int>     <int>
## 1 2000-01-01 1.395417 4.167e-05         4         13      3002
## 2 2000-01-02 1.395417 4.167e-05         4         13      3002
## 3 2000-01-03 1.395417 4.167e-05         4         13      3002
## 4 2000-01-04 1.395417 4.167e-05         4         13      3002
## 5 2000-01-05 1.395417 4.167e-05         4         13      3002
## 6 2000-01-06 1.395417 4.167e-05         4         13      3002
## 7 2000-01-07 1.395417 4.167e-05         4         13      3002
## 8 2000-01-08 1.395417 4.167e-05         4         13      3002
## 9 2000-01-09 1.395417 4.167e-05         4         13      3002
## 10 2000-01-10 1.395417 4.167e-05         4         13      3002
## # ... with 411,461 more rows, and 12 more variables: Address <chr>,
## #   State <chr>, County <chr>, City <chr>, Year <int>, Month <int>,
## #   Day <int>, NO2_Units <chr>, NO2_Mean <dbl>, NO2_1st_Max_Value <dbl>,
## #   NO2_1st_Max_Hour <int>, NO2_AQI <int>
```

```
Pollution_SO2 %>% filter(SO2_Mean > 0) %>% mutate(mean_max=max(SO2_Mean),mean_min=min(SO2_Mean)) %>% select(Date_Local,mean_max,mean_min,everything()) %>% distinct()
```

```
## # A tibble: 755,350 x 18
##   Date_Local mean_max mean_min State_Code County_Code Site_Num
##   <date>     <dbl>     <dbl>     <int>     <int>     <int>
## 1 2000-01-01 3.21625 4.167e-05         4         13      3002
## 2 2000-01-01 3.21625 4.167e-05         4         13      3002
## 3 2000-01-02 3.21625 4.167e-05         4         13      3002
## 4 2000-01-02 3.21625 4.167e-05         4         13      3002
## 5 2000-01-03 3.21625 4.167e-05         4         13      3002
## 6 2000-01-03 3.21625 4.167e-05         4         13      3002
## 7 2000-01-04 3.21625 4.167e-05         4         13      3002
## 8 2000-01-04 3.21625 4.167e-05         4         13      3002
## 9 2000-01-05 3.21625 4.167e-05         4         13      3002
## 10 2000-01-05 3.21625 4.167e-05         4         13      3002
## # ... with 755,340 more rows, and 12 more variables: Address <chr>,
## #   State <chr>, County <chr>, City <chr>, Year <int>, Month <int>,
```

```
## # Day <int>, S02_Units <chr>, S02_Mean <dbl>, S02_1st_Max_Value <dbl>,
## # S02_1st_Max_Hour <int>, S02_AQI <int>

Pollution_O3 %>% filter(O3_Mean > 0) %>% mutate(mean_max=max(O3_Mean),mean_min=min(O3_Mean)) %>% select(Date_Local,mean_max,mean_min,everything()) %>%distinct()

## # A tibble: 415,980 x 18
##   Date_Local mean_max mean_min State_Code County_Code Site_Num
##   <date>     <dbl>   <dbl>     <int>     <int>     <int>
## 1 2000-01-01 0.95083 0.00042         4         13      3002
## 2 2000-01-02 0.95083 0.00042         4         13      3002
## 3 2000-01-03 0.95083 0.00042         4         13      3002
## 4 2000-01-04 0.95083 0.00042         4         13      3002
## 5 2000-01-05 0.95083 0.00042         4         13      3002
## 6 2000-01-06 0.95083 0.00042         4         13      3002
## 7 2000-01-07 0.95083 0.00042         4         13      3002
## 8 2000-01-08 0.95083 0.00042         4         13      3002
## 9 2000-01-09 0.95083 0.00042         4         13      3002
## 10 2000-01-10 0.95083 0.00042         4         13      3002
## # ... with 415,970 more rows, and 12 more variables: Address <chr>,
## # State <chr>, County <chr>, City <chr>, Year <int>, Month <int>,
## # Day <int>, O3_Units <chr>, O3_Mean <dbl>, O3_1st_Max_Value <dbl>,
## # O3_1st_Max_Hour <int>, O3_AQI <int>

Pollution_CO %>% filter(CO_Mean > 0) %>% mutate(mean_max=max(CO_Mean),mean_min=min(CO_Mean)) %>% select(Date_Local,mean_max,mean_min,everything()) %>%distinct()

## # A tibble: 814,443 x 18
##   Date_Local mean_max mean_min State_Code County_Code Site_Num
##   <date>     <dbl>   <dbl>     <int>     <int>     <int>
## 1 2000-01-01 75.08333 0.00042         4         13      3002
## 2 2000-01-01 75.08333 0.00042         4         13      3002
## 3 2000-01-02 75.08333 0.00042         4         13      3002
## 4 2000-01-02 75.08333 0.00042         4         13      3002
## 5 2000-01-03 75.08333 0.00042         4         13      3002
## 6 2000-01-03 75.08333 0.00042         4         13      3002
## 7 2000-01-04 75.08333 0.00042         4         13      3002
## 8 2000-01-04 75.08333 0.00042         4         13      3002
## 9 2000-01-05 75.08333 0.00042         4         13      3002
## 10 2000-01-05 75.08333 0.00042         4         13      3002
## # ... with 814,433 more rows, and 12 more variables: Address <chr>,
## # State <chr>, County <chr>, City <chr>, Year <int>, Month <int>,
## # Day <int>, CO_Units <chr>, CO_Mean <dbl>, CO_1st_Max_Value <dbl>,
## # CO_1st_Max_Hour <int>, CO_AQI <int>
```

즉, 다음과 같이 각 오염물질별 평균의 최댓값 변수를 만들어 정리하였다.

이를 각 오염물질별로 간단한 그림과 함께 확인해보자.

```
#NO2

Pollution_NO2 %>% group_by(Date_Local) %>% summarise(count=n()) %>% arrange(desc(count))

## # A tibble: 5,996 x 2
##   Date_Local count
##   <date> <int>
## 1 2002-06-10 640
## 2 2002-06-09 636
## 3 2011-05-24 452
## 4 2008-06-13 416
## 5 2008-06-12 408
## 6 2013-04-17 408
## 7 2013-04-16 406
## 8 2008-07-10 400
```

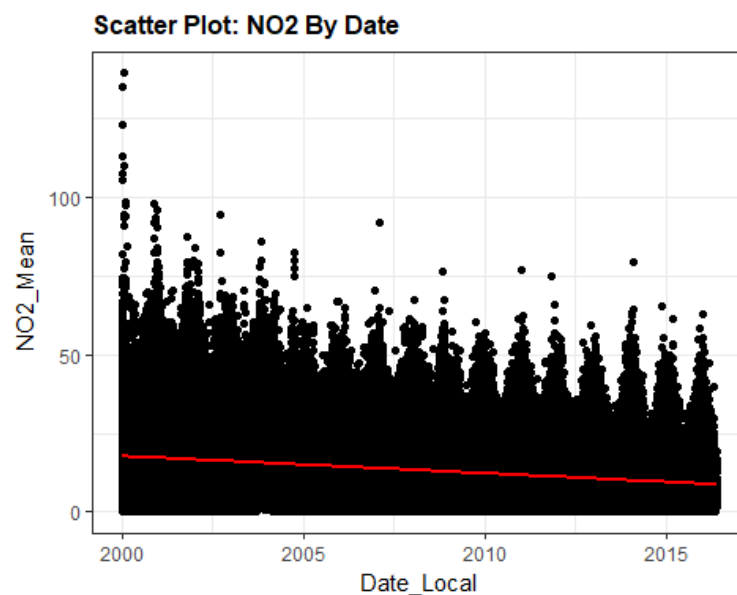
```
## 9 2013-04-08 400
## 10 2013-04-15 400
## # ... with 5,986 more rows

Pollution_N02 %>% group_by(Date_Local) %>% summarise(count=n()) %>% arrange(count)

## # A tibble: 5,996 x 2
##   Date_Local count
##   <date> <int>
## 1 2016-05-01 28
## 2 2016-05-02 28
## 3 2016-05-03 28
## 4 2016-05-04 28
## 5 2016-05-05 28
## 6 2016-05-06 28
## 7 2016-05-07 28
## 8 2016-05-08 28
## 9 2016-05-09 28
## 10 2016-05-10 28
## # ... with 5,986 more rows
```

NO2 의 오염도가 가장 낮은 날짜는 2016-05, 가장 높은 날짜는 2002-06-10 이다.

```
Pollution_N02 %>% group_by(Date_Local) %>%
ggplot(aes(Date_Local,NO2_Mean))+geom_point()+geom_smooth(method="lm",color="red")
```



```
Pollution_N02 %>% filter(NO2_Mean>0.01) %>% arrange(desc(NO2_Mean)) %>%
select(Date_Local,NO2_Mean,everything()) %>% distinct()
```

```
## # A tibble: 398,622 x 16
##   Date_Local NO2_Mean State_Code County_Code Site_Num
##   <date>     <dbl>     <int>     <int>     <int>
## 1 2000-01-19 1.3954167       4        13       3003
## 2 2000-01-13 1.3533333       4        13       3003
## 3 2000-01-10 1.3518750       4        13       3003
## 4 2000-01-12 1.2333333       4        13       3003
## 5 2000-01-15 1.1308333       4        13       3003
## 6 2000-01-18 1.1013636       4        13       3003
```

```
## 7 2000-01-11 1.0754546      4      13      3003
## 8 2000-01-05 1.0550000      4      13      3003
## 9 2000-02-04 0.9875000      4      13      3003
## 10 2000-11-27 0.9813044      6      37      1103
## # ... with 398,612 more rows, and 11 more variables: Address <chr>,
## #   State <chr>, County <chr>, City <chr>, Year <int>, Month <int>,
## #   Day <int>, NO2_Units <chr>, NO2_1st_Max_Value <dbl>,
## #   NO2_1st_Max_Hour <int>, NO2_AQI <int>
```

해가 지날수록 NO2 는 감소하는 경향을 보임을 알 수 있다.

#03

```
Pollution_03 %>% group_by(Date_Local) %>% arrange(desc(O3_1st_Max_Value))
```

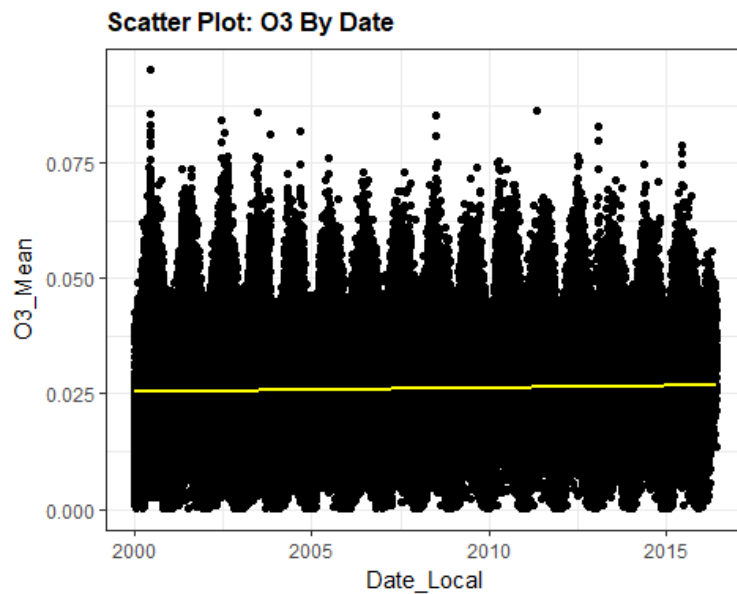
```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
##   <int>      <int>    <int>    <chr>
## 1      80        2      12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## 2      80        2      12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## 3      80        2      12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## 4      80        2      12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## 5        6       65     8001 5888 MISSION BLVD., RUBIDOUX
## 6        6       65     8001 5888 MISSION BLVD., RUBIDOUX
## 7        6       65     8001 5888 MISSION BLVD., RUBIDOUX
## 8        6       65     8001 5888 MISSION BLVD., RUBIDOUX
## 9      42      91      13 STATE ARMORY - 1046 BELVOIR RD
## 10     42      91      13 STATE ARMORY - 1046 BELVOIR RD
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, O3_Units <chr>, O3_Mean <dbl>, O3_1st_Max_Value <dbl>,
## #   O3_1st_Max_Hour <int>, O3_AQI <int>
```

```
Pollution_03 %>% group_by(Date_Local) %>% arrange(desc(O3_Mean))
```

```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
##   <int>      <int>    <int>    <chr>
## 1      42      91      13 STATE ARMORY - 1046 BELVOIR RD
## 2      42      91      13 STATE ARMORY - 1046 BELVOIR RD
## 3      42      91      13 STATE ARMORY - 1046 BELVOIR RD
## 4      42      91      13 STATE ARMORY - 1046 BELVOIR RD
## 5        6      83     1025 LFC #1-LAS FLORES CANYON
## 6        6      83     1025 LFC #1-LAS FLORES CANYON
## 7        6      83     1025 LFC #1-LAS FLORES CANYON
## 8        6      83     1025 LFC #1-LAS FLORES CANYON
## 9      36     103        9 57 DIVISION STREET
## 10     36     103        9 57 DIVISION STREET
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, O3_Units <chr>, O3_Mean <dbl>, O3_1st_Max_Value <dbl>,
## #   O3_1st_Max_Hour <int>, O3_AQI <int>
```

O3 평균 오염도를 내림차순, 오름차순으로 정렬해 보았다. 2000-06-10 에 최대치를 기록했다.

```
Pollution_03 %>% group_by(Date_Local) %>%
ggplot(aes(Date_Local,O3_Mean))+geom_point()+geom_smooth(method="lm",color="yellow")
```



O3 는 다른 오염물질과 다르게 해가 지날수록 평균이 증가한다.

#S02

#S02 의 평균을 크기가 큰 순과 작은 순으로 정렬해 보았다.

```
Pollution_S02 %>% group_by(Date_Local) %>% arrange(desc(S02_1st_Max_Value))
```

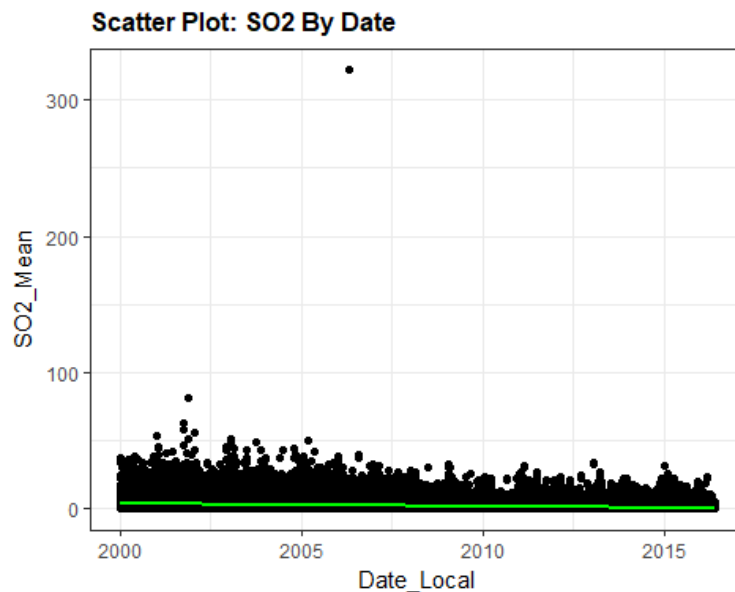
```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
##   <int>      <int>    <int>    <chr>
## 1      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 2      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 3      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 4      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 5      17        163      10    13TH & TUDOR
## 6      17        163      10    13TH & TUDOR
## 7      42         73      15    CROTON ST & JEFFERSON ST.
## 8      42         73      15    CROTON ST & JEFFERSON ST.
## 9      17        163      10    13TH & TUDOR
## 10     17        163      10    13TH & TUDOR
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, S02_Units <chr>, S02_Mean <dbl>, S02_1st_Max_Value <dbl>,
## #   S02_1st_Max_Hour <int>, S02_AQI <int>
```

```
Pollution_S02 %>% group_by(Date_Local) %>% arrange(desc(S02_Mean))
```

```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
##   <int>      <int>    <int>    <chr>
## 1      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 2      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 3      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 4      40         21    9002 P.O. BOX 948 TAHLEQUAH, OK 74464
## 5      17        163      10    13TH & TUDOR
## 6      17        163      10    13TH & TUDOR
```

```
## 7      17      163      10      13TH & TUDOR
## 8      17      163      10      13TH & TUDOR
## 9      17      163      10      13TH & TUDOR
## 10     17      163      10      13TH & TUDOR
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, SO2_Units <chr>, SO2_Mean <dbl>, SO2_1st_Max_Value <dbl>,
## #   SO2_1st_Max_Hour <int>, SO2_AQI <int>
```

```
Pollution_SO2 %>% group_by(Date_Local) %>% ggplot(aes(Date_Local,SO2_Mean))+geom_point()+ge
om_smooth(method="lm",color="green")
```



#CO

```
Pollution_CO %>% group_by(Date_Local) %>% arrange(desc(CO_1st_Max_Value))
```

```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
##   <int>      <int>    <int>    <chr>
## 1         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 2         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 3         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 4         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 5        22        51       1001 West Temple Pl
## 6        22        51       1001 West Temple Pl
## 7         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 8         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 9         6         71       306 14306 PARK AVE., VICTORVILLE, CA
## 10        6         71       306 14306 PARK AVE., VICTORVILLE, CA
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, CO_Units <chr>, CO_Mean <dbl>, CO_1st_Max_Value <dbl>,
## #   CO_1st_Max_Hour <int>, CO_AQI <int>
```

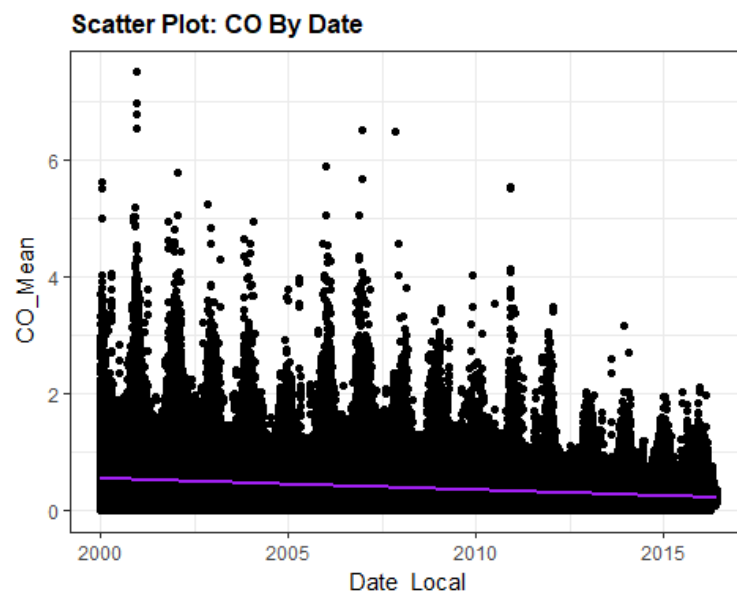
```
Pollution_CO %>% group_by(Date_Local) %>% arrange(desc(CO_Mean))
```

```
## # A tibble: 1,746,661 x 16
## # Groups:   Date_Local [5,996]
##   State_Code County_Code Site_Num Address
```

```
##      <int>      <int>      <int>      <chr>
## 1         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 2         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 3         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 4         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 5         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 6         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 7         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 8         6         25         5 1029 ETHEL ST, CALEXICO HIGH SCHOOL
## 9        80          2        12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## 10       80          2        12 UABC, CALZADA BENITO JUAREZ, MEXICALI
## # ... with 1,746,651 more rows, and 12 more variables: State <chr>,
## #   County <chr>, City <chr>, Date_Local <date>, Year <int>, Month <int>,
## #   Day <int>, CO_Units <chr>, CO_Mean <dbl>, CO_1st_Max_Value <dbl>,
## #   CO_1st_Max_Hour <int>, CO_AQI <int>
```

CO 의 평균을 오름차순, 내림차순으로 정렬했다. 최대치는 2000-12 이다.

```
Pollution_CO %>% group_by(Date_Local) %>% ggplot(aes(Date_Local,CO_Mean))+geom_point()+geom_smooth(method="lm",color="purple")
```



```
Pollution_CO %>% filter(CO_Mean>6) %>% arrange(desc(CO_Mean)) %>%
select(Date_Local,CO_Mean,everything()) %>% distinct()
```

```
## # A tibble: 6 x 17
##   Date_Local CO_Mean State_Code County_Code Site_Num
##   <date>     <dbl>      <int>      <int>      <int>
## 1 2000-12-20 7.508333         6         25         5
## 2 2000-12-21 6.975000         6         25         5
## 3 2000-12-20 6.795652         6         25         5
## 4 2000-12-21 6.543478         6         25         5
## 5 2006-12-21 6.533333        80          2        12
## 6 2007-11-01 6.500000        80          2        12
## # ... with 12 more variables: Address <chr>, State <chr>, County <chr>,
## #   City <chr>, Year <int>, Month <int>, Day <int>, CO_Units <chr>,
## #   CO_1st_Max_Value <dbl>, CO_1st_Max_Hour <int>, CO_AQI <int>,
## #   CO_AQI_Class <chr>
```

시간이 지날수록 CO 의 평균농도는 감소한다.

【 LOCATION 】

오염물질 별 평균이 높은 주를 확인해보았다. AQI mean 과는 조금 다른 결과가 출력되는 것을 알 수 있었다.

#오염물질 별로 평균이 높은 주는 어디일지 살펴보자.

```
Pollution_NO2%>%group_by(State)%>%summarise(mean_NO2=mean(NO2_Mean))%>%arrange(desc(mean_NO2))%>%head(5)
```

```
## # A tibble: 5 x 2
##       State mean_NO2
##   <chr>    <dbl>
## 1 Country Of Mexico 20.31479
## 2      Colorado 19.71049
## 3      Arizona 19.09904
## 4      New York 18.99439
## 5 Massachusetts 18.62205
```

```
Pollution_SO2%>%group_by(State)%>%summarise(mean_SO2=mean(SO2_Mean))%>%arrange(desc(mean_SO2))%>%head(5)
```

```
## # A tibble: 5 x 2
##       State mean_SO2
##   <chr>    <dbl>
## 1      Alaska 6.092910
## 2      New York 4.819131
## 3 District Of Columbia 4.256948
## 4      Pennsylvania 4.198356
## 5      Kentucky 3.800549
```

```
Pollution_CO%>%group_by(State)%>%summarise(mean_CO=mean(CO_Mean))%>%arrange(desc(mean_CO))%>%head(5)
```

```
## # A tibble: 5 x 2
##       State mean_CO
##   <chr>    <dbl>
## 1 Country Of Mexico 0.8577745
## 2 District Of Columbia 0.7958255
## 3      Arizona 0.4885040
## 4      Missouri 0.4684906
## 5      California 0.4578803
```

```
Pollution_O3%>%group_by(State)%>%summarise(mean_O3=mean(O3_Mean))%>%arrange(desc(mean_O3))%>%head(5)
```

```
## # A tibble: 5 x 2
##       State mean_O3
##   <chr>    <dbl>
## 1 Wyoming 0.03806685
## 2 Tennessee 0.03779823
## 3      Utah 0.03206618
## 4      Nevada 0.03199683
## 5 New Mexico 0.03171082
```

NO2 의 평균이 가장 높은 5 개의 주를 살펴보았다. Country Of Mexico, Colorado, Arizona, New York, Massachusetts 순으로 나왔다.

SO2 의 평균이 높은 주는 Alaska, New York, District of Columbia, Pennsylvania, Kentucky 순으로 나왔다.

CO 의 평균이 높은 주는 Country of Mexico, District of Columbia, Arizona, Missouri,

California 순으로 나왔다.

O3의 평균이 높은 주는 Wyoming, Tennessee, Utah, Nevada, New Mexico 순서로 나타났다.

NO2, SO2, CO의 평균이 높은 주는 겹치는 주가 많은데 O3의 평균이 높은 주와는 겹치는 주가 없는 것으로 보인다. NO2, SO2, CO와 O3의 농도가 서로 반대의 경향을 나타냈던 점을 고려해 O3 평균이 가장 낮은 5개의 주도 살펴보았다.

```
Pollution_O3 %>% group_by(State) %>% summarise(mean_O3 = mean(O3_Mean)) %>% arrange(desc(mean_O3)) %>% tail(5)
```

```
## # A tibble: 5 x 2
##       State      mean_O3
##       <chr>      <dbl>
## 1 Washington 0.02086637
## 2 Georgia    0.02059773
## 3 Massachusetts 0.02040808
## 4 Oregon     0.01930986
## 5 Alaska     0.01281565
```

Washington, Georgia, Massachusetts, Oregon, Alaska 순으로 나타났는데 Massachusetts는 NO2의 평균이 높은 주 중 하나였다. 또한 Alaska는 SO2의 평균이 가장 높은 주이다.

IV. HYPOTHESIS

지금까지 분석한 내용을 바탕으로 더욱 궁금한 점들에 대해 가설을 세우고 검정했다.

가설1: 출근시간의 Max_Value가 높을 것이다.

하루 동안 측정된 이산화질소 농도 중 최대치를 나타낸 변수 _1st_Max_Value와, 그러한 최대 농도가 측정된 시각을 나타낸 변수이다 _1st_Max_Hour을 이용하여 첫 번째 가설을 세웠다.

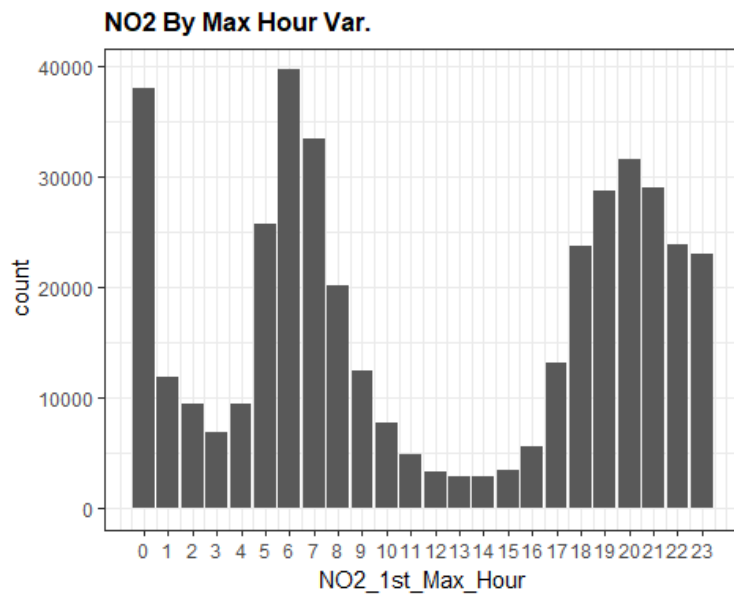
가설이 참인지 알아보기 위해 먼저 _1st_Max_Hour 을 count 한 후 그림을 그려 확인했다.

#NO2 의 Max_Hour count 하고 그림으로 표현하기

```
Pollution_NO2 %>% count(NO2_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   NO2_1st_Max_Hour      n
##             <int> <int>
## 1                 0 38051
## 2                 1 11910
## 3                 2  9516
## 4                 3  6916
## 5                 4  9482
## 6                 5 25747
## 7                 6 39639
## 8                 7 33429
## 9                 8 20098
## 10                9 12449
## 11                10  7795
## 12                11  4864
## 13                12  3323
## 14                13  2869
## 15                14  2918
## 16                15  3477
## 17                16  5643
## 18                17 13199
## 19                18 23788
## 20                19 28785
## 21                20 31522
## 22                21 29074
## 23                22 23899
## 24                23 23078
```

```
ggplot(Pollution_NO2, aes(NO2_1st_Max_Hour))+geom_bar()+scale_x_continuous(breaks=c(1:23))
```

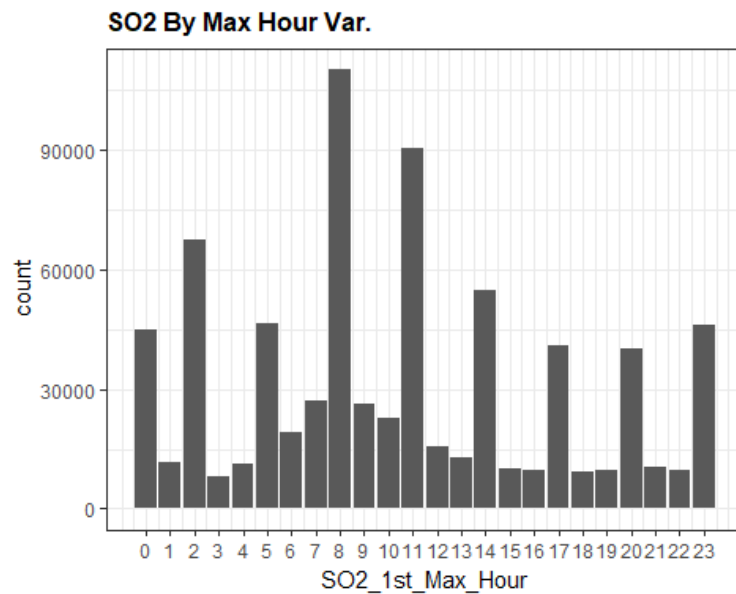


#S02 의 Max_Hour count 하고 그림으로 표현하기

```
Pollution_S02 %>% count(S02_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   S02_1st_Max_Hour      n
##   <int> <int>
## 1         0 45082
## 2         1 11583
## 3         2 67661
## 4         3  8218
## 5         4 11123
## 6         5 46669
## 7         6 19095
## 8         7 26949
## 9         8 110085
## 10        9 26396
## 11       10 22748
## 12       11 90579
## 13       12 15510
## 14       13 12748
## 15       14 54898
## 16       15 10183
## 17       16  9741
## 18       17 41073
## 19       18  9243
## 20       19  9610
## 21       20 40348
## 22       21 10390
## 23       22  9476
## 24       23 45942
```

```
ggplot(Pollution_S02,aes(S02_1st_Max_Hour))+geom_bar()+scale_x_continuous(breaks=c(1:23))
```

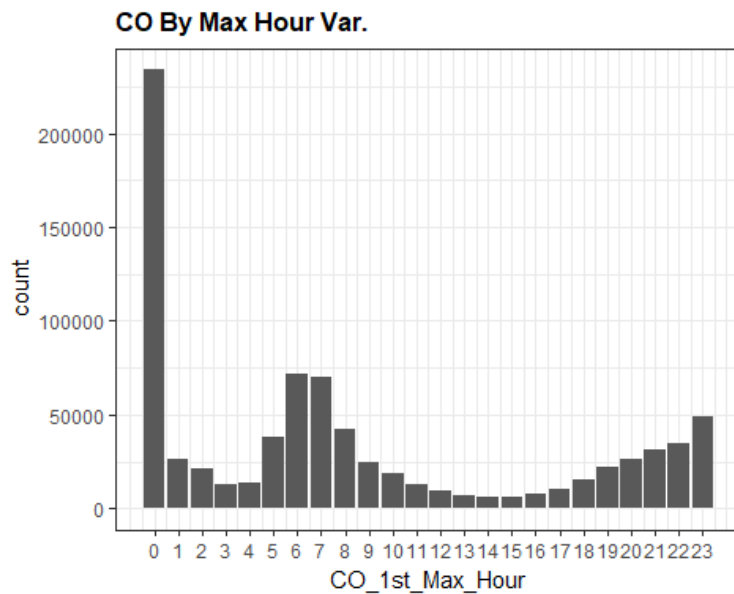


#CO 의 Max_Hour count 하고 그림으로 표현하기

```
Pollution_CO %>% count(CO_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   CO_1st_Max_Hour      n
##   <int> <int>
## 1         0 23366
## 2         1  2630
## 3         2  2096
## 4         3  1329
## 5         4  1415
## 6         5  3780
## 7         6  7206
## 8         7  7000
## 9         8  4232
## 10        9  2501
## 11       10  1851
## 12       11  1328
## 13       12   968
## 14       13   740
## 15       14   631
## 16       15   661
## 17       16   802
## 18       17  1038
## 19       18  1516
## 20       19  2179
## 21       20  2660
## 22       21  3118
## 23       22  3511
## 24       23  4876
```

```
ggplot(Pollution_CO,aes(CO_1st_Max_Hour))+geom_bar()+scale_x_continuous(breaks=c(1:23))
```



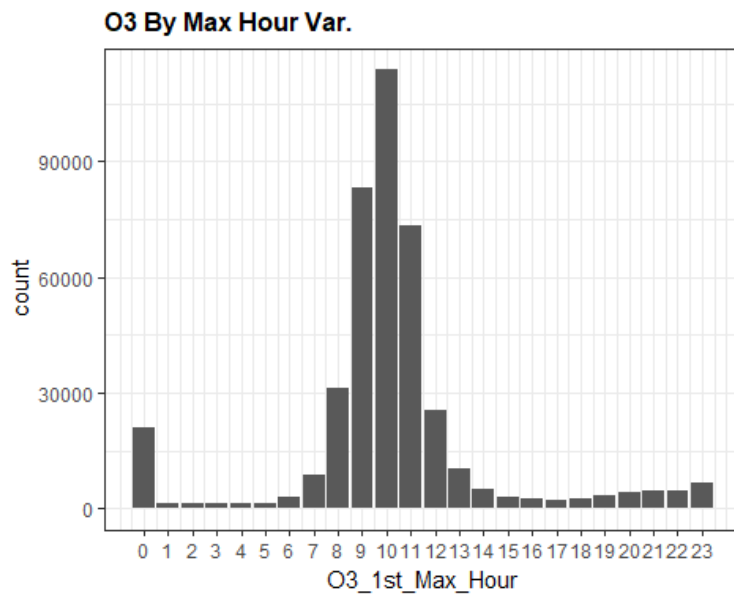
#03 의 Max_Hour count 하고 그림으로 표현하기

```
Pollution_03 %>% count(O3_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   O3_1st_Max_Hour      n
##   <int>    <int>
## 1         0 20984
## 2         1  1218
## 3         2  1216
## 4         3  1228
## 5         4  1217
## 6         5  1538
## 7         6  3088
## 8         7  8841
## 9         8 31219
## 10        9 83267
## 11       10 113611
## 12       11  73458
## 13       12 25547
## 14       13 10196
## 15       14  4919
## 16       15  3140
## 17       16  2467
## 18       17  2305
## 19       18  2670
## 20       19  3403
## 21       20  4211
## 22       21  4745
## 23       22  4649
## 24       23  6843
```

```
ggplot(Pollution_03,aes(O3_1st_Max_Hour))+geom_bar()+scale_x_continuous(breaks=c(1:23))+scale_x_continuous(breaks=c(1:23))
```

```
## Scale for 'x' is already present. Adding another scale for 'x', which
## will replace the existing scale.
```



그래프를 보면 NO2 는 출근시간에 많은 관측을 보였고 SO2 는 8 시와 11 시, CO 는 0 시에 다른 때와는 확연히 차이 나는 많은 수치를 달성했으며 O3 는 출근시간이 아닐 때 많이 관측되었다. 어느 시간대에 가장 많이 관측되었는지 확인했으므로 이제 각 물질마다 자세히 시간에 따른 농도를 알아보도록 하겠다.

#NO

#Hour 로 그룹을 나누었을 때 데이터의 수 구하기

```
Pollution_NO2 %>% count(NO2_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   NO2_1st_Max_Hour    n
##   <int> <int>
## 1         0 38051
## 2         1 11910
## 3         2  9516
## 4         3  6916
## 5         4  9482
## 6         5 25747
## 7         6 39639
## 8         7 33429
## 9         8 20098
## 10        9 12449
## 11        10  7795
## 12        11  4864
## 13        12  3323
## 14        13  2869
## 15        14  2918
## 16        15  3477
## 17        16  5643
## 18        17 13199
## 19        18 23788
## 20        19 28785
## 21        20 31522
## 22        21 29074
## 23        22 23899
## 24        23 23078
```

```
Pollution_NO2 %>% group_by(NO2_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(count)
```

```
## # A tibble: 24 x 2
##   NO2_1st_Max_Hour count
##   <int> <int>
## 1         13 2869
## 2         14 2918
## 3         12 3323
## 4         15 3477
## 5         11 4864
## 6         16 5643
## 7          3 6916
## 8         10 7795
## 9          4 9482
## 10        2 9516
## # ... with 14 more rows
```

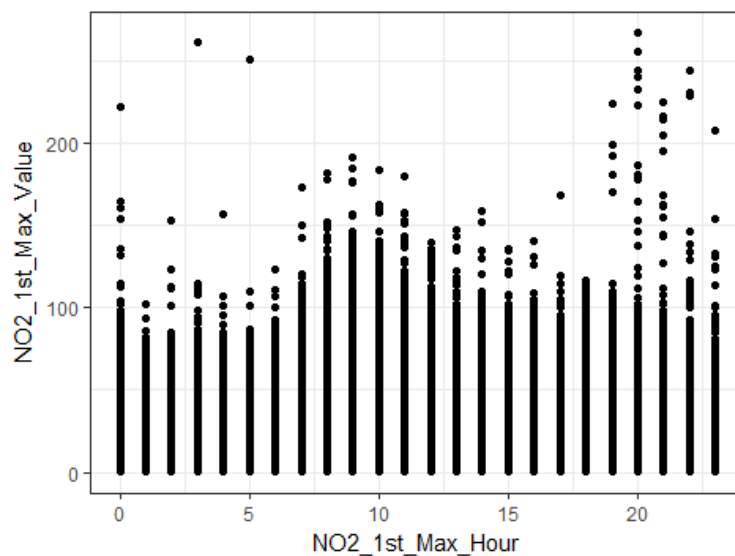
```
Pollution_NO2 %>% group_by(NO2_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(desc(count))
```

```
## # A tibble: 24 x 2
##   NO2_1st_Max_Hour count
##   <int> <int>
## 1          6 39639
## 2          0 38051
## 3          7 33429
## 4         20 31522
## 5         21 29074
## 6         19 28785
## 7          5 25747
## 8         22 23899
## 9         18 23788
## 10        23 23078
## # ... with 14 more rows
```

#시간별로 그룹을 나누고 Max_Value 를 산점도로 구하

```
ggplot(Pollution_NO2,aes(NO2_1st_Max_Hour,NO2_1st_Max_Value)) + geom_point()
```

Scatter Plot: NO2 Max Value Mean By Max Hour Var.

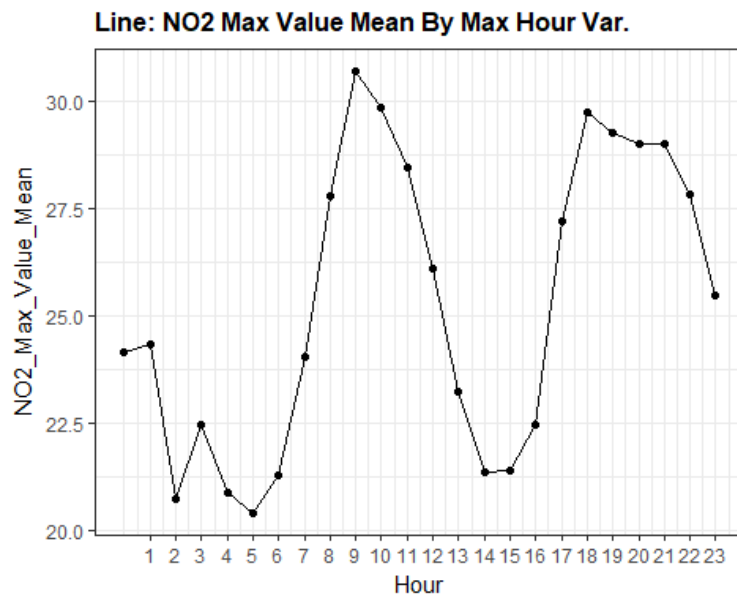


#시간별로 Max_Value 의 mean 을 구하기

```
Pollution_NO2_1 <- Pollution_NO2 %>% group_by(NO2_1st_Max_Hour) %>% summarise(Max_Value_mean_by_hour=mean(NO2_1st_Max_Value))
```


#시간별 Max_Value 의 mean 을 구해 그림을 그린다.

```
ggplot(Pollution_NO2_1,aes(NO2_1st_Max_Hour,Max_Value_mean_by_hour)) + geom_point()+geom_line(group=1) + scale_x_continuous(breaks=c(1:23))+xlab("Hour")+ylab("NO2_Max_Value_Mean")
```



실제 출근시간아 아닌 2 시와 3 시 사이에 엄청난 양이 줄었다는 것을 알 수 있었다. 또한 출근시간과 퇴근시간에는 높은 농도를 기록했다는 것을 알 수 있었다.

#S02

#Hour 로 그룹을 나누었을 때 데이터의 수 구하기

```
Pollution_S02 %>% count(S02_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   S02_1st_Max_Hour     n
##         <int> <int>
## 1             0 45082
## 2             1 11583
## 3             2 67661
## 4             3  8218
## 5             4 11123
## 6             5 46669
## 7             6 19095
## 8             7 26949
## 9             8 110085
## 10            9 26396
## 11           10 22748
## 12           11 90579
## 13           12 15510
## 14           13 12748
## 15           14 54898
## 16           15 10183
## 17           16  9741
## 18           17 41073
## 19           18  9243
## 20           19  9610
## 21           20 40348
## 22           21 10390
## 23           22  9476
## 24           23 45942
```

```
Pollution_S02 %>% group_by(S02_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(count)
```

```
## # A tibble: 24 x 2
##   S02_1st_Max_Hour count
##   <int> <int>
## 1         3    8218
## 2        18   9243
## 3        22   9476
## 4        19   9610
## 5        16   9741
## 6        15  10183
## 7        21  10390
## 8         4  11123
## 9         1  11583
## 10       13  12748
## # ... with 14 more rows
```

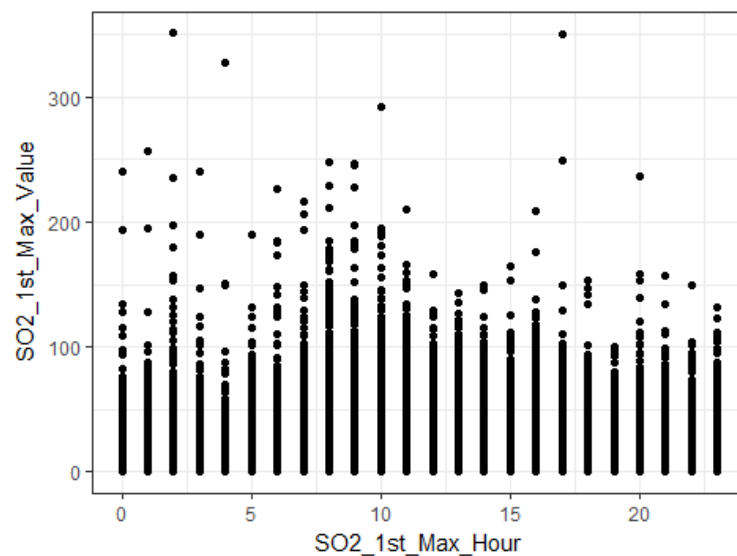
```
Pollution_S02 %>% group_by(S02_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(desc(count))
```

```
## # A tibble: 24 x 2
##   S02_1st_Max_Hour count
##   <int> <int>
## 1         8  110085
## 2        11   90579
## 3         2   67661
## 4        14   54898
## 5         5   46669
## 6        23   45942
## 7         0   45082
## 8        17   41073
## 9        20   40348
## 10        7   26949
## # ... with 14 more rows
```

#시간별로 그룹을 나누고 Max_Value 를 산점도로 구하

```
ggplot(Pollution_S02,aes(S02_1st_Max_Hour,S02_1st_Max_Value)) + geom_point()
```

Scatter Plot: SO2 Max Value Mean By Max Hour Var.

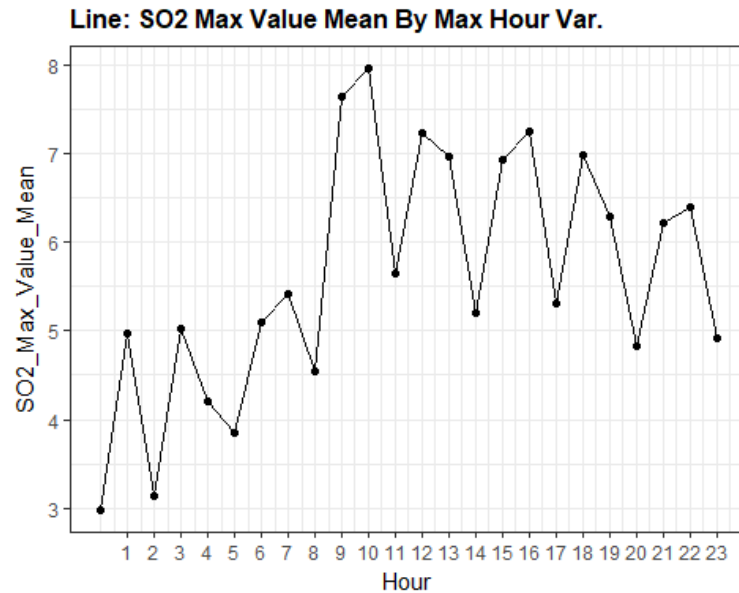


#시간별로 Max_Value 의 mean 을 구하기

```
Pollution_S02_1 <- Pollution_S02 %>% group_by(S02_1st_Max_Hour) %>% summarise(Max_Value_mean_by_hour=mean(S02_1st_Max_Value))
```

#시간별 Max_Value 의 mean 을 구해 그림을 그린다.

```
ggplot(Pollution_SO2_1,aes(SO2_1st_Max_Hour,Max_Value_mean_by_hour)) + geom_point()+geom_line(group=1) + scale_x_continuous(breaks=c(1:23))+xlab("Hour")+ylab("SO2_Max_Value_Mean")
```



SO2 는 이상하게 그래프가 형성되었는데 출근시간부터 퇴근시간을 포함한 대부분의 시간에서 높은 농도를 기록한다는 것을 알 수 있었다.

#CO

#Hour 로 그룹을 나누었을 때 데이터의 수 구하기

```
Pollution_CO %>% count(CO_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2
##   CO_1st_Max_Hour      n
##   <int> <int>
## 1         0 233666
## 2         1  26303
## 3         2  20967
## 4         3  13296
## 5         4  14159
## 6         5  37808
## 7         6  72067
## 8         7  70001
## 9         8  42320
## 10        9  25015
## 11       10  18517
## 12       11  13280
## 13       12   9682
## 14       13   7408
## 15       14   6313
## 16       15   6612
## 17       16   8021
## 18       17  10385
## 19       18  15169
## 20       19  21799
## 21       20  26600
## 22       21  31184
```

```
## 23      22 35111
## 24      23 48760
```

```
Pollution_CO %>% group_by(CO_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(count)
```

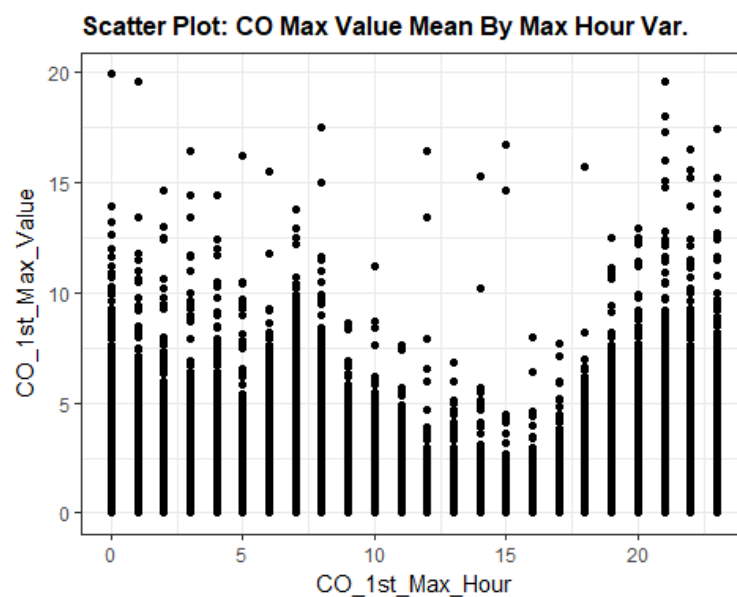
```
## # A tibble: 24 x 2
##   CO_1st_Max_Hour count
##         <int> <int>
## 1             14  6313
## 2             15  6612
## 3             13  7408
## 4             16  8021
## 5             12  9682
## 6             17 10385
## 7             11 13280
## 8              3 13296
## 9              4 14159
## 10            18 15169
## # ... with 14 more rows
```

```
Pollution_CO %>% group_by(CO_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(desc(count))
```

```
## # A tibble: 24 x 2
##   CO_1st_Max_Hour count
##         <int> <int>
## 1              0 233666
## 2              6  72067
## 3              7  70001
## 4             23 48760
## 5              8 42320
## 6              5  37808
## 7             22  35111
## 8             21  31184
## 9             20  26600
## 10             1  26303
## # ... with 14 more rows
```

#시간별로 그룹을 나누고 Max_Value 를 산점도로 구하

```
ggplot(Pollution_CO,aes(CO_1st_Max_Hour,CO_1st_Max_Value)) + geom_point()
```

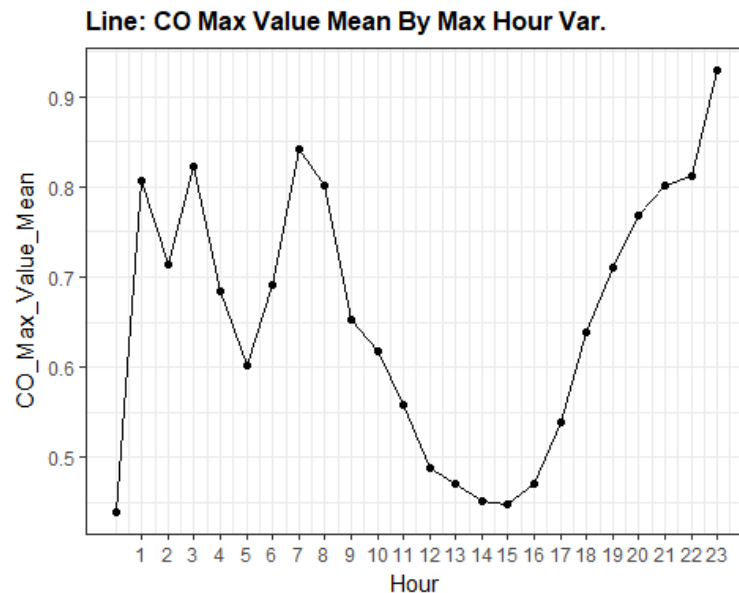


#시간별로 Max_Value 의 mean 을 구하기

```
Pollution_CO_1 <- Pollution_CO %>% group_by(CO_1st_Max_Hour) %>% summarise(Max_Value_mean_by_hour=mean(CO_1st_Max_Value))
```

#시간별 Max_Value 의 mean 을 구해 그림을 그린다.

```
ggplot(Pollution_CO_1,aes(CO_1st_Max_Hour,Max_Value_mean_by_hour)) + geom_point()+geom_line  
(group=1) + scale_x_continuous(breaks=c(1:23))+xlab("Hour")+ylab("CO_Max_Value_Mean")
```



#03

Hour 로 그룹을 나누었을 때 데이터의 수 구하기

```
Pollution_O3 %>% count(O3_1st_Max_Hour) %>% print(n=24)
```

```
## # A tibble: 24 x 2  
##   O3_1st_Max_Hour      n  
##           <int> <int>  
## 1             0 20984  
## 2             1  1218  
## 3             2  1216  
## 4             3  1228  
## 5             4  1217  
## 6             5  1538  
## 7             6  3088  
## 8             7  8841  
## 9             8 31219  
## 10            9 83267  
## 11           10 113611  
## 12           11  73458  
## 13           12 25547  
## 14           13 10196  
## 15           14  4919  
## 16           15  3140  
## 17           16  2467  
## 18           17  2305  
## 19           18  2670  
## 20           19  3403  
## 21           20  4211  
## 22           21  4745  
## 23           22  4649  
## 24           23  6843
```

```
Pollution_O3 %>% group_by(O3_1st_Max_Hour) %>% summarise(count=n()) %>% arrange(count)
```

```
## # A tibble: 24 x 2
##   O3_1st_Max_Hour count
##   <int> <int>
## 1         2    1216
## 2         4    1217
## 3         1    1218
## 4         3    1228
## 5         5    1538
## 6        17    2305
## 7        16    2467
## 8        18    2670
## 9         6    3088
## 10       15    3140
## # ... with 14 more rows
```

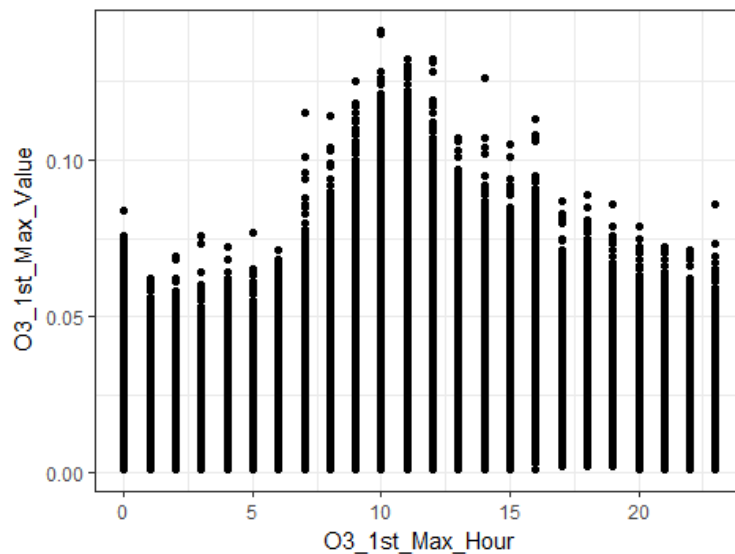
```
Pollution_O3 %>% group_by(O3_1st_Max_Hour) %>% summarise(count=n()) %>%
arrange(desc(count))
```

```
## # A tibble: 24 x 2
##   O3_1st_Max_Hour count
##   <int> <int>
## 1        10 113611
## 2         9  83267
## 3        11  73458
## 4         8  31219
## 5        12  25547
## 6         0  20984
## 7        13  10196
## 8         7   8841
## 9        23   6843
## 10       14   4919
## # ... with 14 more rows
```

#시간별로 그룹을 나누고 Max_Value 를 산점도로 구하

```
ggplot(Pollution_O3,aes(O3_1st_Max_Hour,O3_1st_Max_Value)) + geom_point()
```

Scatter Plot: O3 Max Value Mean By Max Hour Var.

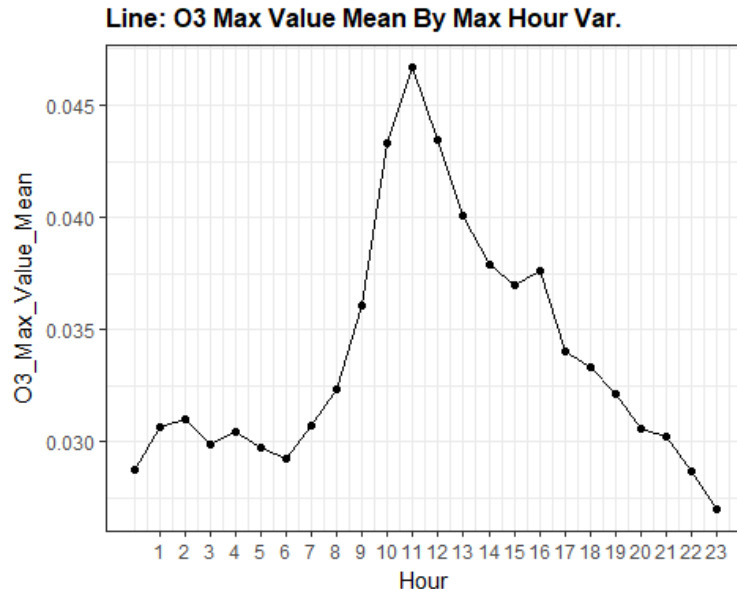


#시간별로 Max_Value 의 mean 을 구하기

```
Pollution_O3_1 <- Pollution_O3 %>% group_by(O3_1st_Max_Hour) %>%
summarise(Max_Value_mean_by_hour=mean(O3_1st_Max_Value))
```

#시간별 Max_Value 의 mean 을 구해 그림을 그린다.

```
ggplot(Pollution_O3_1,aes(O3_1st_Max_Hour,Max_Value_mean_by_hour)) +  
geom_point()+geom_line(group=1) +  
scale_x_continuous(breaks=c(1:23))+xlab("Hour")+ylab("O3_Max_Value_Mean")
```



O3 는 가설과는 정반대로 9 시부터 12 시까지 급속도로 오르다 12 시쯤에 절정을 찍고 다시 줄어드는 것을 관측할 수 있다.

가설 1 은 네 오염물질 모두 출근시간에 영향을 받을 것이라고 예상했다. NO2 는 가설과 거의 일치했고 CO 는 가설의 경향을 따르는 반면, SO2 는 아예 따르지 않았으며 O3 는 정반대의 결과로 나타났다. 이 결과를 알아보기로자 <관악과 시청별 요일별 오존농도>¹² 라는 논문을 참고해서 결과의 원인을 분석해보았다.

먼저, NO2는 교통량에 영향을 많이 받기 때문에 출근시간에 높은 농도를 형성해 가설과 맞는 결과를 보여주었다. 이러한 NO2는 O3의 합성작용을 방해한다. 그리하여 NO2의 농도가 낮은 출근시간이 아닌 때에 O3가 실제 높은 농도를 보여주고 있다는 것 또한 알게 되었다. 그래서 결과적으로 NO2와 O3가 상반되는 결과를 보였다고 예측했다.

CO는 출퇴근시간에 높은 농도를 달성하지만 퇴근시간이 지난 밤 또한 높은 농도를 기록했다. 이것은 CO는 약 78%로 주거환경에서 대부분 많이 형성되는데 모두 퇴근하고 집에서 있을 때 주거활동을 하며 CO를 발생시켜 이러한 결과가 나타난다고 예측했다.

SO2는 황을 태울 때 나타나는데 천연적으로는 온천, 화산지역에서 나타나고 인위적으로는 석탄, 경유, 중유 등을 태울 때 나타난다. 따라서 출근시간부터 노동시간을 거쳐 퇴근시간까지 계속하여 높은 농도를 달성하고 있다고 예측했다.

¹² 김정화, 김용표, 「관악과 시청의 요일별 오존 농도: 1996 ~ 2000년 측정 자료」.. 『한국대기환경학회지』 제19권 제5호.

가설 1의 결론을 좀더 확인하기 위하여 가장 많은 관측치 수를 보인 California에 국한해 확인해보고자 한다. 특히, 이 주의 경우 교통량이 많기 때문에 가설 1을 알아보기 위한 좋은 범위로 판단했다. 그리고 교통량에 가장 많은 영향을 받는 NO2와 NO2에 가장 영향을 받는 O3를 비교해보고자 이 둘의 그림을 출력해보았다.

```
Pollution_NO2_California <- Pollution_NO2 %>% filter(State=="California") %>% group_by(NO2_1st_Max_Hour) %>% summarise(mean1=mean(NO2_1st_Max_Value)) %>% mutate(Hour=NO2_1st_Max_Hour) %>% select(-NO2_1st_Max_Hour)
Pollution_NO2_California %>% print(n=24)
```

```
## # A tibble: 24 x 2
##       mean1 Hour
##       <dbl> <int>
## 1 22.46103     0
## 2 26.59426     1
## 3 17.54994     2
## 4 24.82181     3
## 5 20.14155     4
## 6 18.74158     5
## 7 19.96416     6
## 8 23.73555     7
## 9 28.18135     8
## 10 32.12478     9
## 11 31.77148    10
## 12 31.33539    11
## 13 28.94919    12
## 14 25.35160    13
## 15 22.95860    14
## 16 22.32907    15
## 17 22.96942    16
## 18 29.16240    17
## 19 30.06667    18
## 20 29.06320    19
## 21 28.51343    20
## 22 28.44058    21
## 23 27.74271    22
## 24 25.31729    23
```

```
Pollution_O3_California <- Pollution_O3 %>% filter(State=="California") %>% group_by(O3_1st_Max_Hour) %>% summarise(mean2=mean(O3_1st_Max_Value)) %>% mutate(Hour=O3_1st_Max_Hour) %>% select(-O3_1st_Max_Hour)
Pollution_O3_California %>% print(n=24)
```

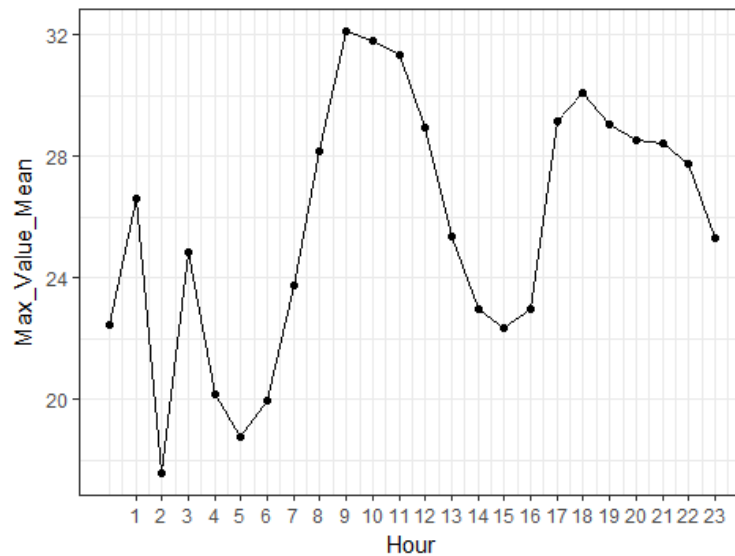
```
## # A tibble: 24 x 2
##       mean2 Hour
##       <dbl> <int>
## 1 0.03236607     0
## 2 0.03327297     1
## 3 0.03482058     2
## 4 0.03294848     3
## 5 0.03416997     4
## 6 0.03302653     5
## 7 0.03132909     6
## 8 0.03204350     7
## 9 0.03255304     8
## 10 0.03522398     9
## 11 0.04219769    10
## 12 0.04490975    11
## 13 0.04132764    12
## 14 0.03914744    13
## 15 0.03780542    14
## 16 0.03848673    15
```



```
## 17 0.04231562 16
## 18 0.03688421 17
## 19 0.03810441 18
## 20 0.03536568 19
## 21 0.03355636 20
## 22 0.03359295 21
## 23 0.03221314 22
## 24 0.02984097 23
```

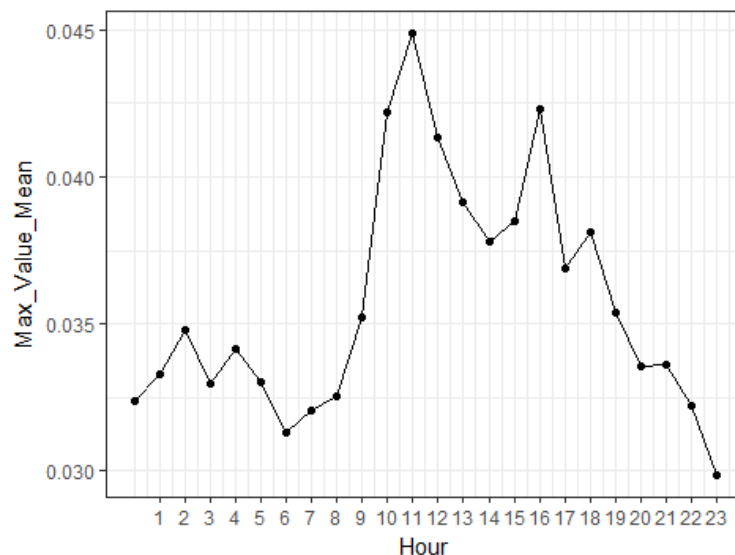
```
ggplot(Pollution_N02_California,aes(Hour,mean1)) + geom_point()+geom_line(group=1) + scale_x_continuous(breaks=c(1:23))+ylab("Max_Value_Mean")
```

Line: California NO2 Max Value Mean By Max Hour Var.

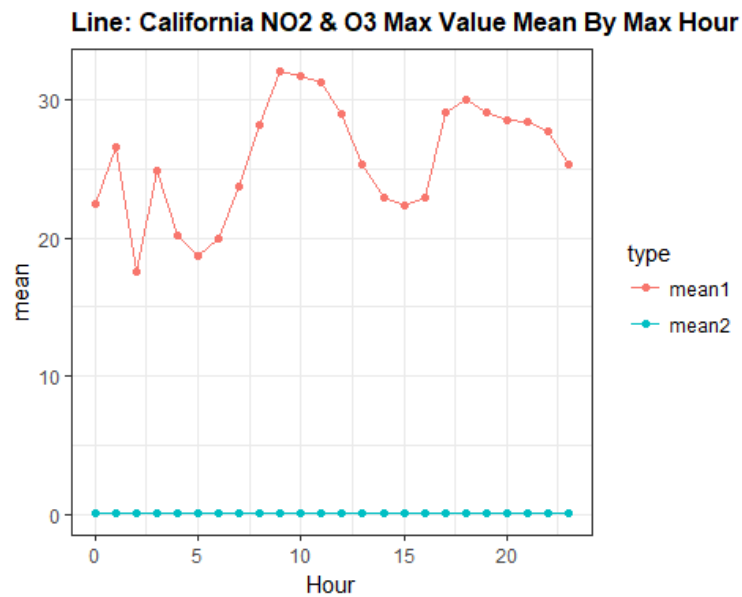


```
ggplot(Pollution_O3_California,aes(Hour,mean2)) + geom_point()+geom_line(group=1) + scale_x_continuous(breaks=c(1:23))+xlab("Hour")+ylab("Max_Value_Mean")
```

Line: California O3 Max Value Mean By Max Hour Var.

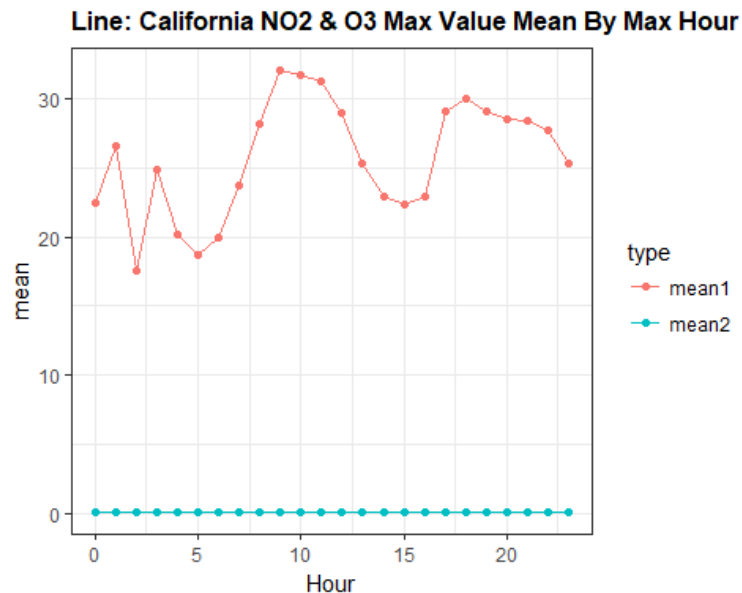


```
difference_O3_NO2 <- left_join(Pollution_N02_California,Pollution_O3_California,by="Hour")
%>% gather('mean1','mean2',key="type",value="mean")
ggplot(difference_O3_NO2,aes(x=Hour,y=mean,color=type))+geom_point()+geom_line()
```



#차이를 알아보기 위해 임의로 NO2 에 1000 을 곱한다. 단위 고려 x

```
Pollution_O3_California2 <- Pollution_O3_California %>% mutate (mean2=mean2*1000)
difference_O3_NO2_2 <- left_join(Pollution_NO2_California,Pollution_O3_California2,by="Hour") %>%
gather('mean1','mean2',key="type",value="mean")
ggplot(difference_O3_NO2_2,aes(x=Hour,y=mean,color=type))+geom_point()+geom_line() + theme_bw() +
labs(title = "Line: California NO2 & O3 Max Value Mean By Max Hour") + theme(plot.title = element_text(face = "bold", size = 12))
```



전역에서 살펴본 그림과 다르게 California 에 국한한 그림은 생각보다 뚜렷한 차이가 보이지 않는다. 하지만 이 결과를 보면 NO2 와 O3 가 대부분 같은 경향을 따르고 있다는 것을 확인할 수 있다. 이는 California 의 날씨나 교통량 등 다른 요인이 작용해서 이와 같은 결과를 나타낸 것이라 예측했다. 하지만 이 그래프에서도 실제 NO3 가 최저점을 찍은 후 O3 가 높아지고 다시 NO3 가 높아지며 O3 가 낮아지는 것을 작게나마 발견할 수 있었다.

가설2: 산업화가 진행됨에 따라 시간이 갈수록 오염도가 높아질 것이다.

이제 두 번째 가설을 살펴보도록 하자. 마찬가지로 네 오염물질로 나누어 분석했다. year로 group을 지은 데이터셋 네개를 새로 만들었다. 그리고 각 해의 평균을 구해 내림차순으로 정렬했다. 참고로 2016년은 5월 31일까지 있다.

#N02

```
N02_Year <- Pollution_N02 %>% group_by(Year) %>% summarise(mean=mean(N02_Mean))
N02_Year %>% arrange(desc(mean))
```

```
## # A tibble: 17 x 2
##   Year      mean
##   <int>    <dbl>
## 1  2001 17.687308
## 2  2000 17.591648
## 3  2002 16.467104
## 4  2003 15.942222
## 5  2004 15.026460
## 6  2005 15.018236
## 7  2006 14.607982
## 8  2007 13.250254
## 9  2008 12.301557
## 10 2009 11.576550
## 11 2010 11.490547
## 12 2011 11.305367
## 13 2012 10.784233
## 14 2013 10.556674
## 15 2016 10.284462
## 16 2014 10.108942
## 17 2015  9.642894
```

#O3

```
O3_Year <- Pollution_O3 %>% group_by(Year) %>% summarise(mean=mean(O3_Mean))
O3_Year %>% arrange(desc(mean))
```

```
## # A tibble: 17 x 2
##   Year      mean
##   <int>    <dbl>
## 1  2012 0.02745501
## 2  2002 0.02671733
## 3  2013 0.02670030
## 4  2010 0.02665000
## 5  2011 0.02659791
## 6  2015 0.02657774
## 7  2014 0.02646954
## 8  2016 0.02646303
## 9  2007 0.02640041
## 10 2008 0.02612328
## 11 2006 0.02584385
## 12 2005 0.02572325
## 13 2003 0.02550736
## 14 2001 0.02541016
## 15 2009 0.02539266
## 16 2004 0.02488524
## 17 2000 0.02431311
```

#SO2

```
S02_Year <- Pollution_S02 %>% group_by(Year) %>% summarise(mean=mean(S02_Mean))
S02_Year %>% arrange(desc(mean))
```

```
## # A tibble: 17 x 2
##   Year      mean
##   <int>    <dbl>
## 1  2000 3.8986894
## 2  2001 3.3544574
## 3  2005 3.1522731
## 4  2003 3.1202194
## 5  2004 3.0594133
## 6  2002 3.0010224
## 7  2006 2.8321886
## 8  2007 2.5816362
## 9  2008 2.1269940
## 10 2009 1.7673155
## 11 2010 1.4567611
## 12 2011 1.4057860
## 13 2012 1.1021100
## 14 2014 0.9924339
## 15 2013 0.9923925
## 16 2015 0.8592291
## 17 2016 0.7598214
```

#CO

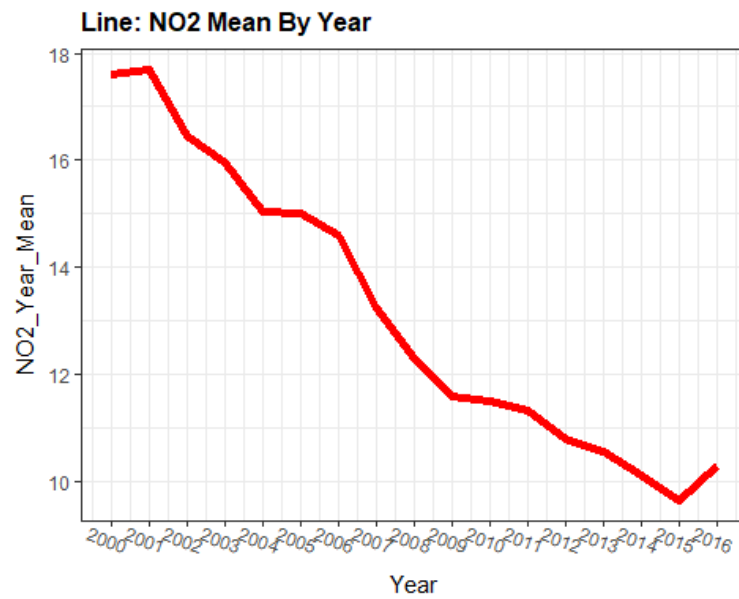
```
CO_Year <- Pollution_CO %>% group_by(Year) %>% summarise(mean=mean(CO_Mean))
CO_Year %>% arrange(desc(mean))
```

```
## # A tibble: 17 x 2
##   Year      mean
##   <int>    <dbl>
## 1  2000 0.5747190
## 2  2002 0.5274224
## 3  2001 0.5261365
## 4  2003 0.5105730
## 5  2004 0.4595850
## 6  2005 0.4369952
## 7  2006 0.4349994
## 8  2007 0.3934329
## 9  2008 0.3463136
## 10 2009 0.3359699
## 11 2010 0.3319805
## 12 2011 0.3141021
## 13 2012 0.3008574
## 14 2016 0.2942060
## 15 2015 0.2861630
## 16 2013 0.2857914
## 17 2014 0.2817015
```

이제 선그래프를 이용하여 시간이 지남에 따라 오염물질의 증감을 알아보고자 한다.

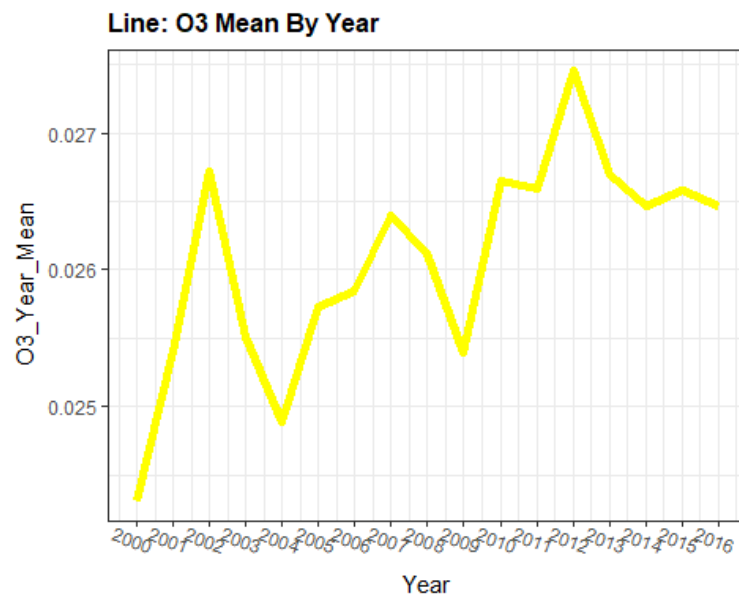
#NO2

```
NO2_Year %>% ggplot(aes(Year,mean))+geom_line(color="red", size = 1.8)
+scale_x_continuous(breaks=c(2000:2016))+ylab("NO2_Year_Mean") + theme_bw() + labs(title =
"Line: NO2 Mean By Year") + theme(axis.text.x = element_text(angle = -20), plot.title =
element_text(face = "bold", size = 12))
```



NO2는 단조롭게 지속적으로 감소하는 추세를 보였다.

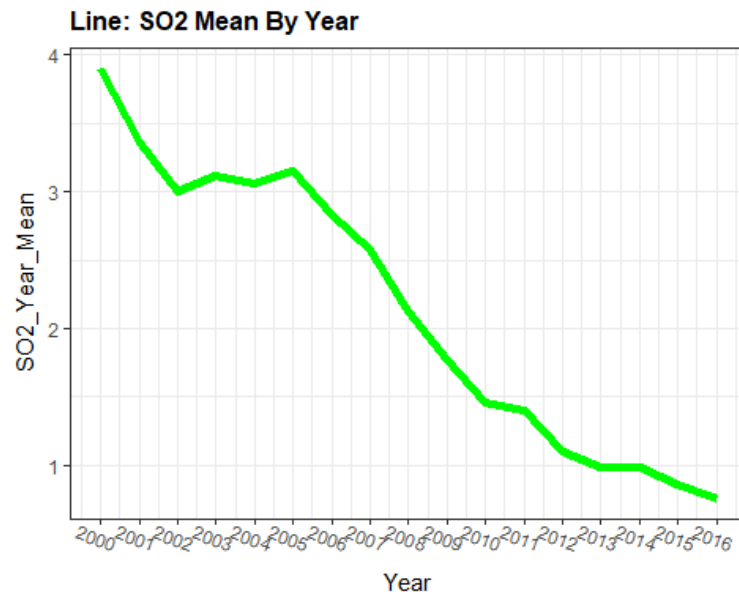
```
#O3
O3_Year %>% ggplot(aes(Year,mean))+geom_line(color="yellow", size =
1.8)+scale_x_continuous(breaks=c(2000:2016))+ylab("O3_Year_Mean") + theme_bw() + labs(title
= "Line: O3 Mean By Year") + theme(axis.text.x = element_text(angle = -20), plot.title =
element_text(face = "bold", size = 12))
```



O3 는 4~5 년을 주기로 증감을 반복하였고 CO 는 지속적으로 감소했다.

#SO2

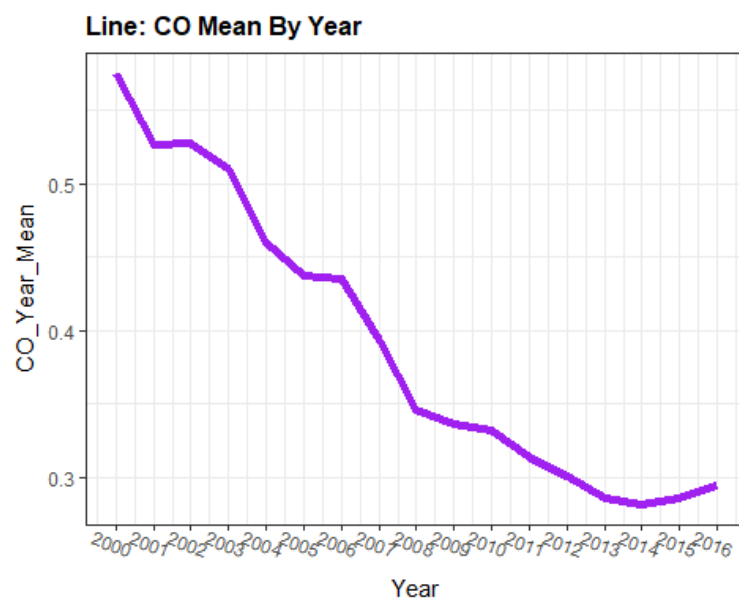
```
SO2_Year %>% ggplot(aes(Year,mean))+geom_line(color="green", size = 1.8)
+scale_x_continuous(breaks=c(2000:2016))+ylab("SO2_Year_Mean") + theme_bw() + labs(title =
"Line: SO2 Mean By Year") + theme(axis.text.x = element_text(angle = -20), plot.title =
element_text(face = "bold", size = 12))
```



SO2 는 2002 년에서 2005 년을 제외한 모든 구간에서 감소했다.

#CO

```
CO_Year %>% ggplot(aes(Year,mean))+geom_line(color="purple", size = 1.8)
+scale_x_continuous(breaks=c(2000:2016))+ylab("CO_Year_Mean") + theme_bw() + labs(title =
"Line: CO Mean By Year") + theme(axis.text.x = element_text(angle = -20), plot.title =
element_text(face = "bold", size = 12))
```



결론적으로, O3 를 제외한 모든 오염물질에서 평균농도가 감소한다는 것으로 앞선 가설은 기각된다. 다행히도 O3 를 제외한 오염물질은 점점 감소하는 편이며 O3 의 원인이 되는 물질들에는 특히 계속 주의를 가져야 하겠다.

이제 다음 가설을 살펴보자.

가설 3: 우리나라는 늦가을에서 겨울에 대기오염이 심하다고 한다. 마찬가지로 미국도 가을에서 겨울에 오염도가 높을 것이다.

가설 3 을 살펴보기 전에 먼저 각 오염물질 데이터셋에서 Month 별로 group 을 두고 평균을 구해 내림차순으로 정렬했다.

#NO2

```
NO2_Month <- Pollution_NO2 %>% group_by(Month) %>% summarise(mean=mean(NO2_Mean))
NO2_Month %>% arrange(desc(mean))
```

```
## # A tibble: 12 x 2
##   Month      mean
##   <int>    <dbl>
## 1      1 17.692570
## 2     12 17.146110
## 3     11 16.423364
## 4      2 15.952683
## 5     10 14.242358
## 6      3 13.532661
## 7      9 12.011193
## 8      4 11.581564
## 9      8 10.659577
## 10     5 10.583321
## 11     6 10.059093
## 12     7  9.756791
```

#O3

```
O3_Month <- Pollution_O3 %>% group_by(Month) %>% summarise(mean=mean(O3_Mean))
O3_Month %>% arrange(desc(mean))
```

```
## # A tibble: 12 x 2
##   Month      mean
##   <int>    <dbl>
## 1      5 0.03310833
## 2      4 0.03255699
## 3      6 0.03231249
## 4      7 0.03104386
## 5      8 0.02969873
## 6      3 0.02805734
## 7      9 0.02595027
## 8      2 0.02191817
## 9     10 0.02121490
## 10     11 0.01809053
## 11      1 0.01682277
## 12     12 0.01563792
```

#SO2

```
SO2_Month <- Pollution_SO2 %>% group_by(Month) %>% summarise(mean=mean(SO2_Mean))
SO2_Month %>% arrange(desc(mean))
```

```
## # A tibble: 12 x 2
##   Month     mean
##   <int>   <dbl>
## 1     1 2.395612
## 2    10 2.240831
## 3    12 2.201079
## 4     2 2.183828
## 5     4 2.142979
## 6     7 2.141768
## 7     8 2.121363
## 8     6 2.076444
## 9     5 2.027910
## 10    9 2.025420
## 11    11 1.942326
## 12     3 1.862431

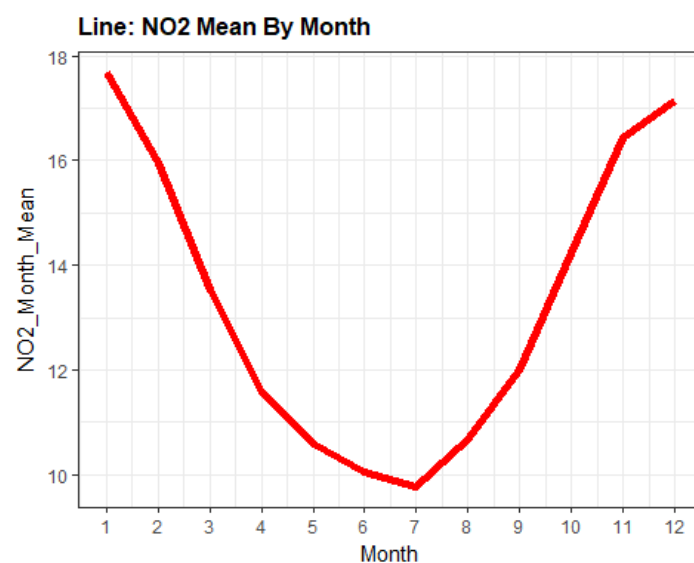
#CO

CO_Month <- Pollution_CO %>% group_by(Month) %>% summarise(mean=mean(CO_Mean))
CO_Month %>% arrange(desc(mean))

## # A tibble: 12 x 2
##   Month     mean
##   <int>   <dbl>
## 1     1 0.5361715
## 2    12 0.5353318
## 3    11 0.4966217
## 4     2 0.4561381
## 5    10 0.3973737
## 6     3 0.3820541
## 7     9 0.3376313
## 8     4 0.3370431
## 9     8 0.3119283
## 10    5 0.3108328
## 11     6 0.2961744
## 12    7 0.2926180
```

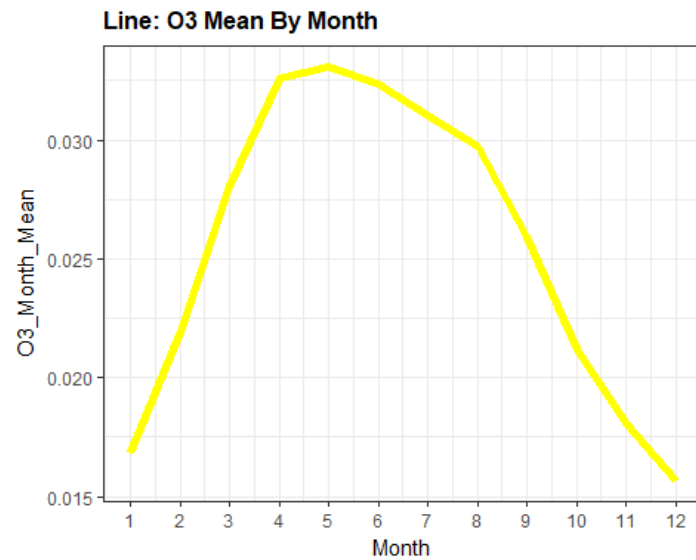
이제, Month 별 각 오염물질 평균을 선그래프로 나타냈다.

```
#NO2
NO2_Month %>% ggplot(aes(Month,mean))+geom_line(color="red", size =
1.8)+scale_x_continuous(breaks=c(1:12))+ylab("NO2_Month_Mean") + theme_bw() + labs(title =
"Line: NO2 Mean By Month") + theme(plot.title = element_text(face = "bold", size = 12))
```

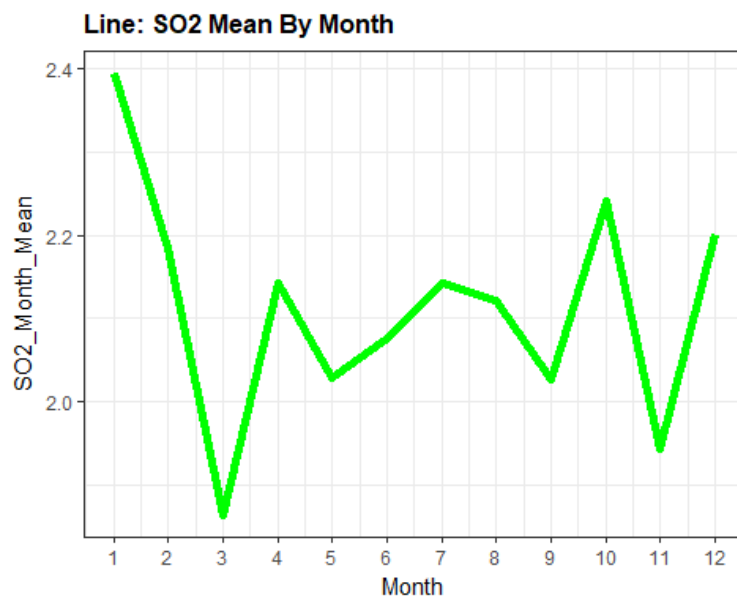


#O3

```
O3_Month %>% ggplot(aes(Month,mean))+geom_line(color="yellow", size =  
1.8)+scale_x_continuous(breaks=c(1:12)) +ylab("O3_Month_Mean") + theme_bw() + labs(title =  
"Line: O3 Mean By Month") + theme(plot.title = element_text(face = "bold", size = 12))
```



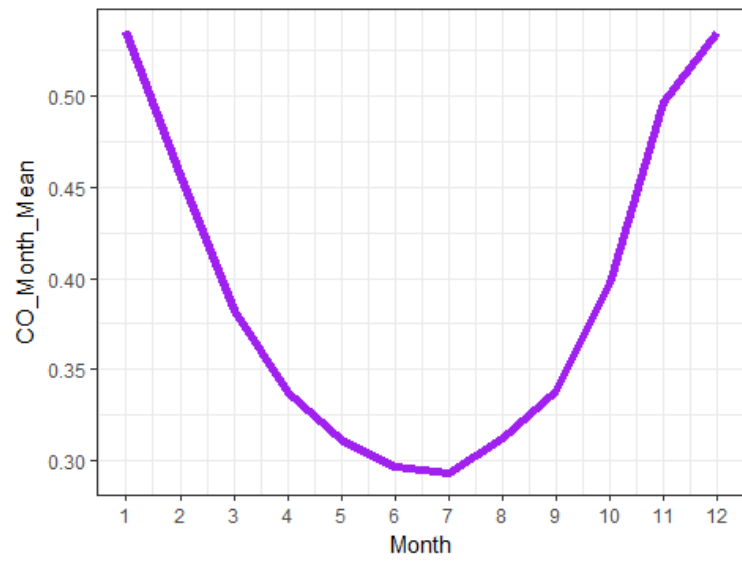
```
SO2_Month %>% ggplot(aes(Month,mean))+geom_line(color="green", size =  
1.8)+scale_x_continuous(breaks=c(1:12)) +ylab("SO2_Month_Mean") + theme_bw() + labs(title =  
"Line: SO2 Mean By Month") + theme(plot.title = element_text(face = "bold", size = 12))
```



#CO

```
CO_Month %>% ggplot(aes(Month,mean))+geom_line(color="purple", size =  
1.8)+scale_x_continuous(breaks=c(1:12))+ylab("CO_Month_Mean") + theme_bw() + labs(title =  
"Line: CO Mean By Month") + theme(plot.title = element_text(face = "bold", size = 12))
```

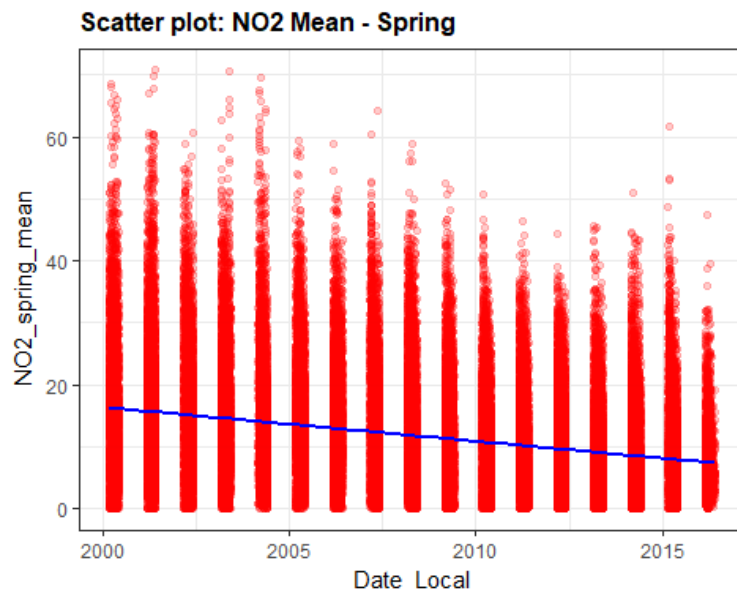
Line: CO Mean By Month



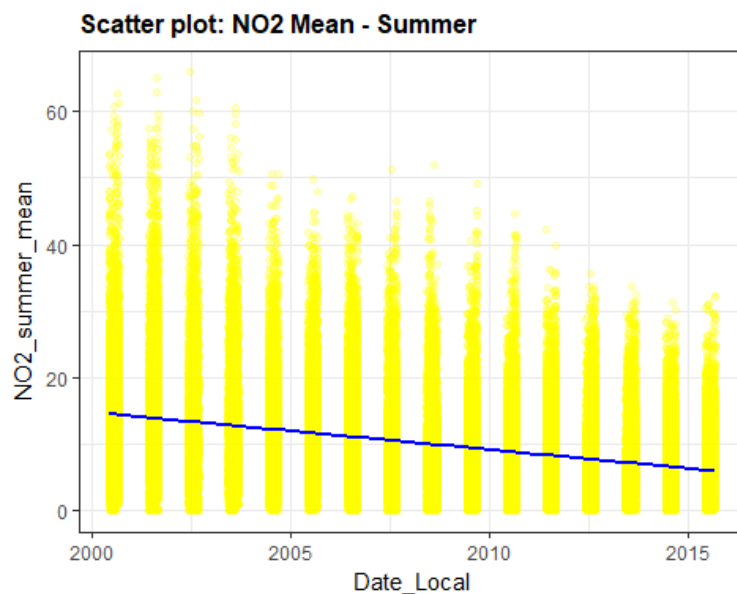
각 오염물질의 계절별 산점도 또한 그려보았다.

```
#NO2 - 4 Season
```

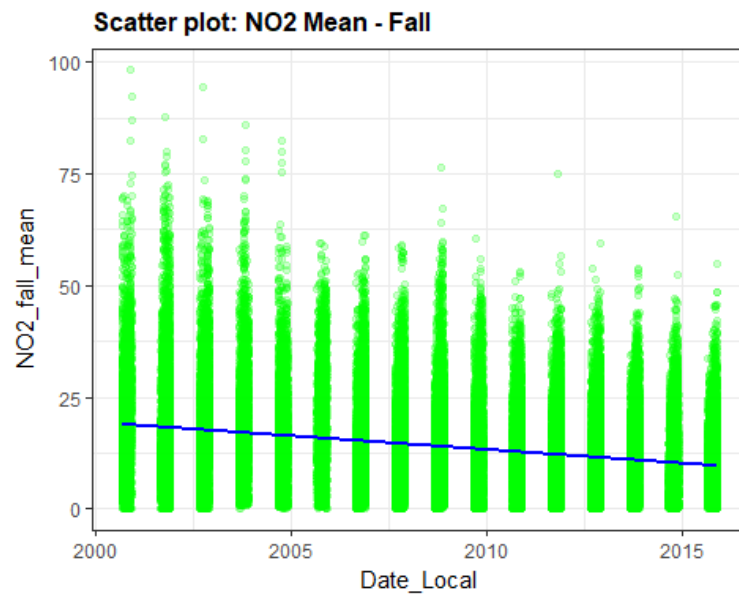
```
NO2_spring%>% ggplot(aes(Date_Local,NO2_Mean))+geom_point(color="red", alpha =  
1/5)+geom_smooth(method="lm",colour="blue")+ylab("NO2_spring_mean") + theme_bw() +  
labs(title = "Scatter plot: NO2 Mean - Spring") + theme(plot.title = element_text(face =  
"bold", size = 12))
```



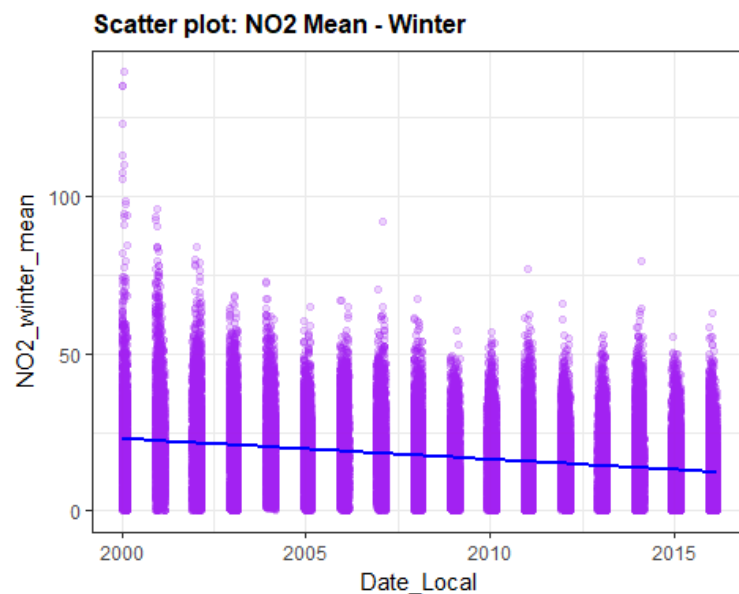
```
NO2_summer%>% ggplot(aes(Date_Local,NO2_Mean))+geom_point(color="yellow", alpha =  
1/5)+geom_smooth(method="lm",colour="blue")+ylab("NO2_summer_mean") + theme_bw() +  
labs(title = "Scatter plot: NO2 Mean - Summer") + theme(plot.title = element_text(face =  
"bold", size = 12))
```



```
NO2_fall%>% ggplot(aes(Date_Local,NO2_Mean))+ geom_point(color="green", alpha = 1/5)+  
geom_smooth(method="lm",colour="blue")+ylab("NO2_fall_mean") + theme_bw() + labs(title =  
"Scatter plot: NO2 Mean - Fall") + theme(plot.title = element_text(face = "bold", size =  
12))
```

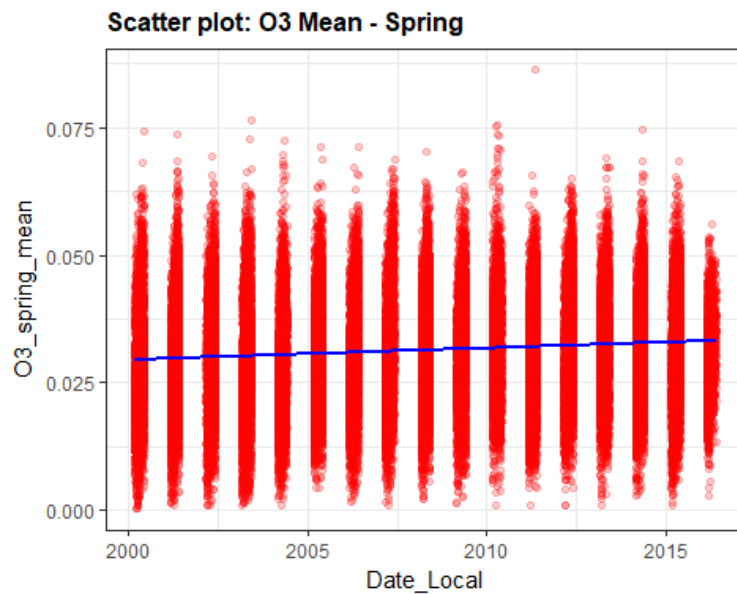


```
NO2_winter%>% ggplot(aes(Date_Local,NO2_Mean))+ geom_point(color="purple", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("NO2_winter_mean") + theme_bw() + labs(title =
"Scatter plot: NO2 Mean - Winter") + theme(plot.title = element_text(face = "bold", size =
12))
```

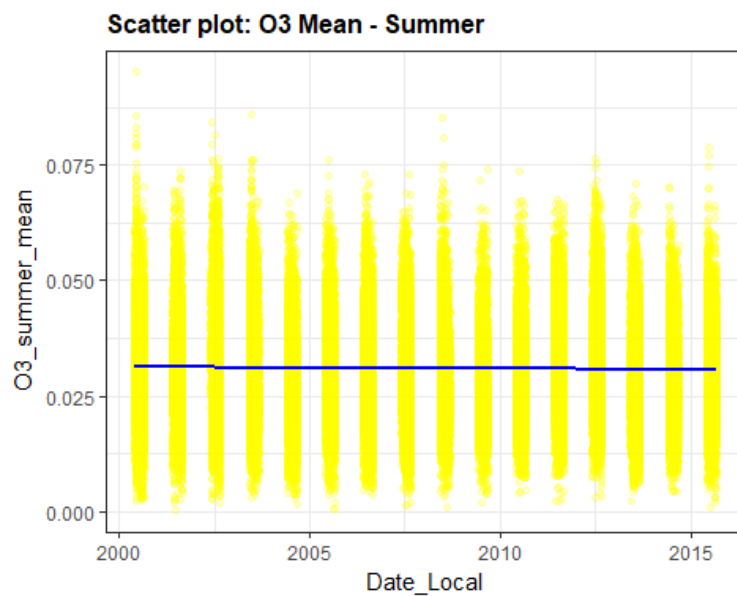


NO2 의 경우 가을, 겨울에 평균 농도가 높다.

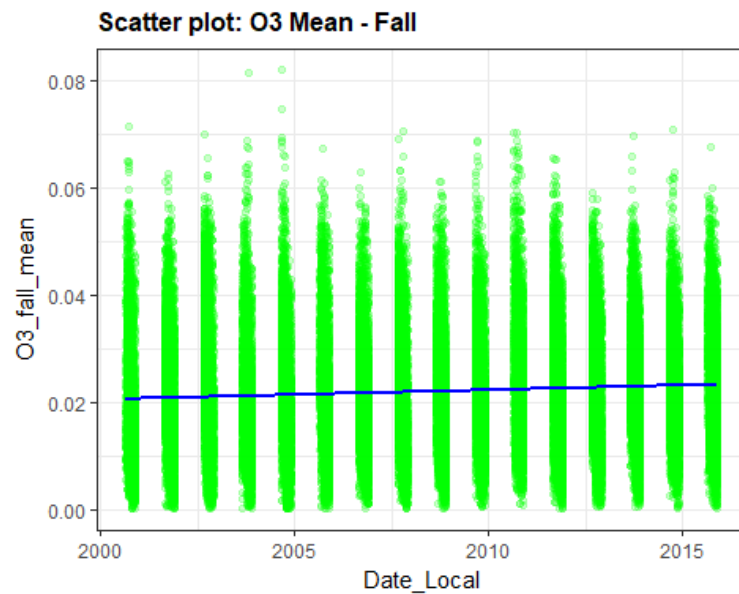
```
#03 - 4 Season
O3_spring%>% ggplot(aes(Date_Local,O3_Mean))+ geom_point(color="red", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("O3_spring_mean") + theme_bw() + labs(title =
"Scatter plot: O3 Mean - Spring") + theme(plot.title = element_text(face = "bold", size =
12))
```



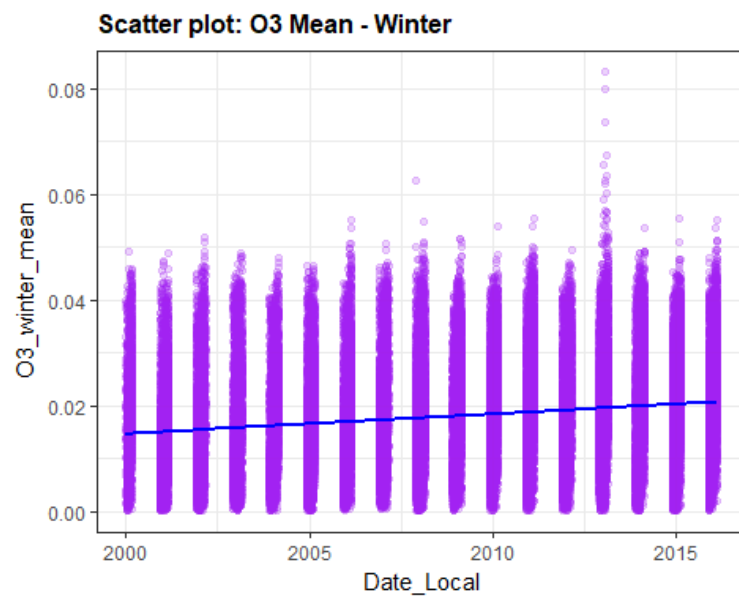
```
O3_summer%>% ggplot(aes(Date_Local,O3_Mean))+ geom_point(color="yellow", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("O3_summer_mean") + theme_bw() + labs(title =
"Scatter plot: O3 Mean - Summer") + theme(plot.title = element_text(face = "bold", size =
12))
```



```
O3_fall%>% ggplot(aes(Date_Local,O3_Mean))+ geom_point(color="green", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("O3_fall_mean") + theme_bw() + labs(title =
"Scatter plot: O3 Mean - Fall") + theme(plot.title = element_text(face = "bold", size =
12))
```

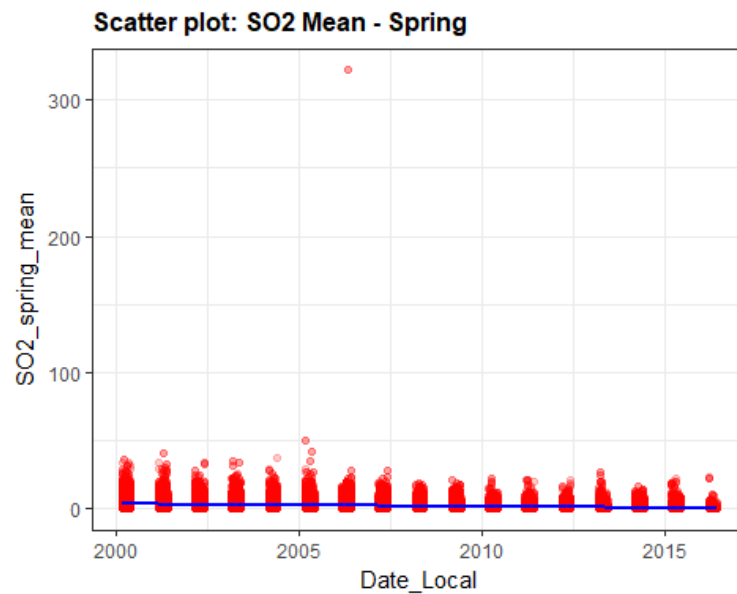


```
O3_winter%>% ggplot(aes(Date_Local,O3_Mean))+ geom_point(color="purple", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("O3_winter_mean") + theme_bw() + labs(title =
"Scatter plot: O3 Mean - Winter") + theme(plot.title = element_text(face = "bold", size =
12))
```

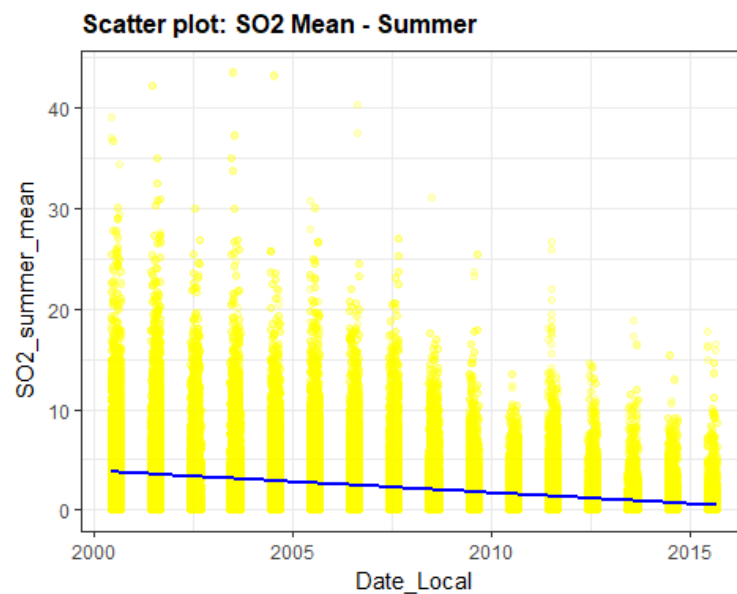


O3 의 경우 겨울에 평균 농도가 가장 낮다.

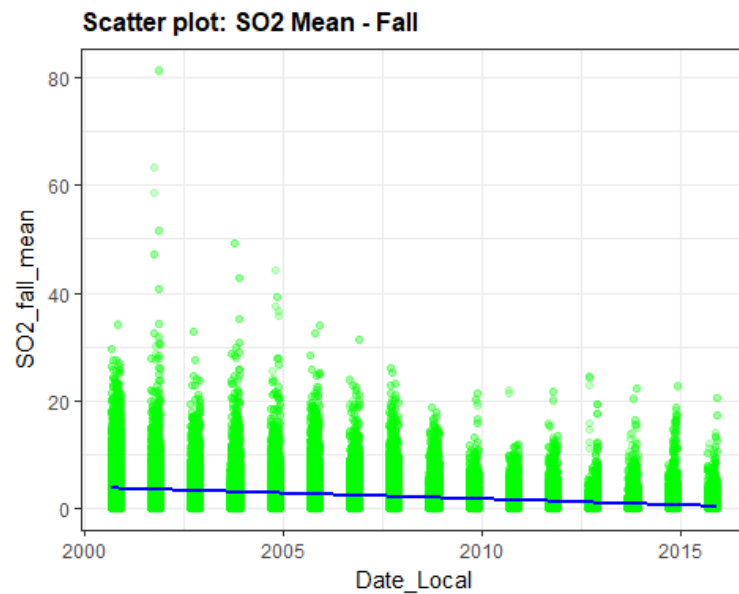
```
#S02 - 4 Season
S02_spring%>% ggplot(aes(Date_Local,S02_Mean))+ geom_point(color="red", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("S02_spring_mean") + theme_bw() + labs(title =
"Scatter plot: S02 Mean - Spring") + theme(plot.title = element_text(face = "bold", size =
12))
```



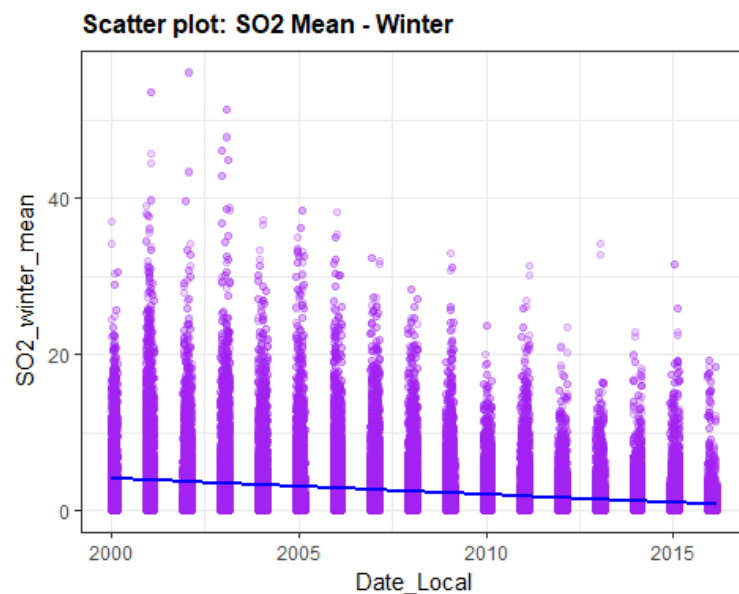
```
SO2_summer%>% ggplot(aes(Date_Local,SO2_Mean))+ geom_point(color="yellow", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("SO2_summer_mean") + theme_bw() + labs(title =
"Scatter plot: SO2 Mean - Summer") + theme(plot.title = element_text(face = "bold", size =
12)) + theme_bw() + labs(title = "Scatter plot: NO2 Mean - Summer") + theme(plot.title =
element_text(face = "bold", size = 12))
```



```
SO2_fall%>% ggplot(aes(Date_Local,SO2_Mean))+ geom_point(color="green", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("SO2_fall_mean") + theme_bw() + labs(title =
"Scatter plot: SO2 Mean - Fall") + theme(plot.title = element_text(face = "bold", size =
12))
```

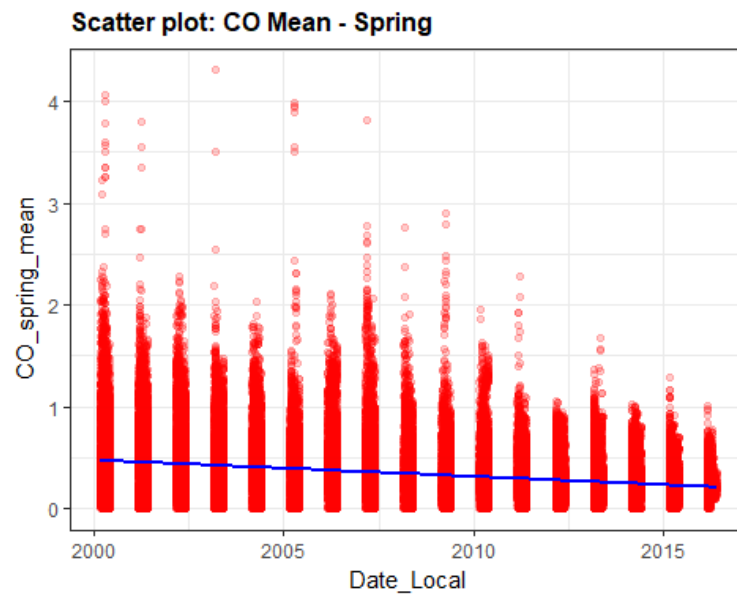


```
SO2_winter%>% ggplot(aes(Date_Local,SO2_Mean))+ geom_point(color="purple", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("SO2_winter_mean") + theme_bw() + labs(title =
"Scatter plot: SO2 Mean - Winter") + theme(plot.title = element_text(face = "bold", size =
12))
```

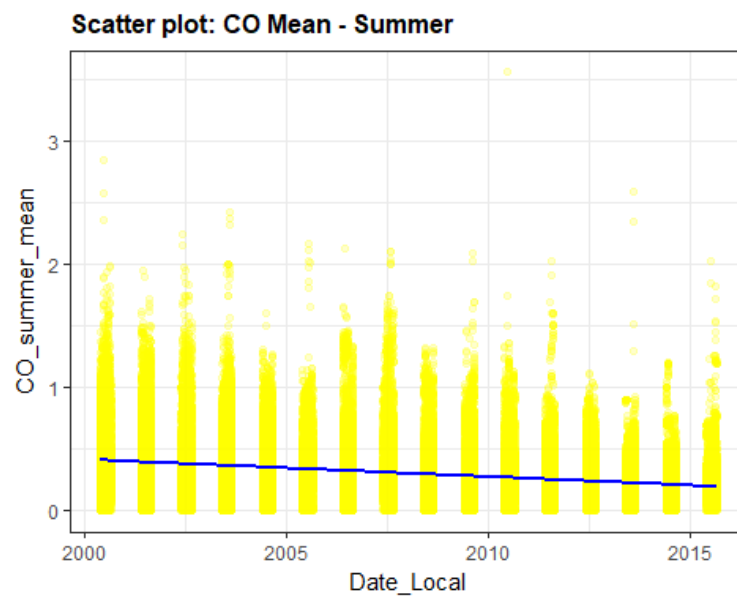


SO2 의 경우 봄, 가을에 평균 농도가 낮고, 여름, 겨울에 평균 농도가 높다.

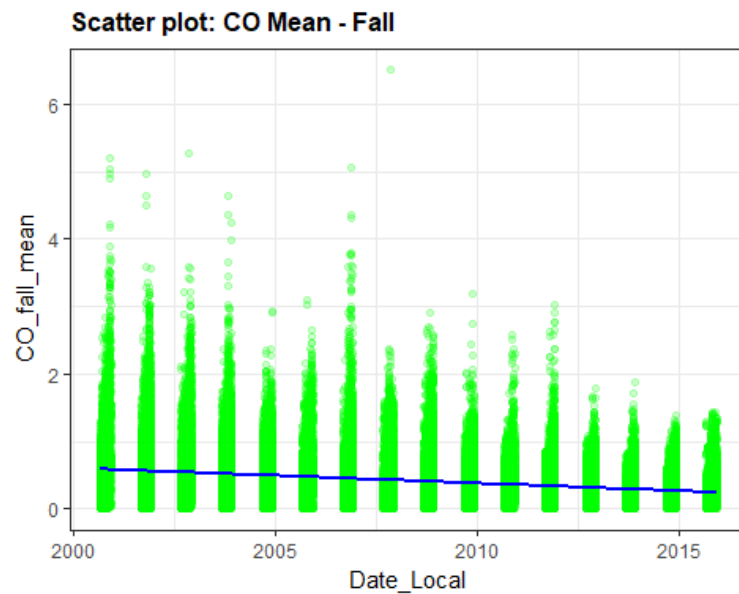
```
#CO - 4 Season
CO_spring%>% ggplot(aes(Date_Local,CO_Mean))+ geom_point(color="red",
alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("CO_spring_mean") + theme_bw()
+ labs(title = "Scatter plot: CO Mean - Spring") + theme(plot.title =
element_text(face = "bold", size = 12))
```

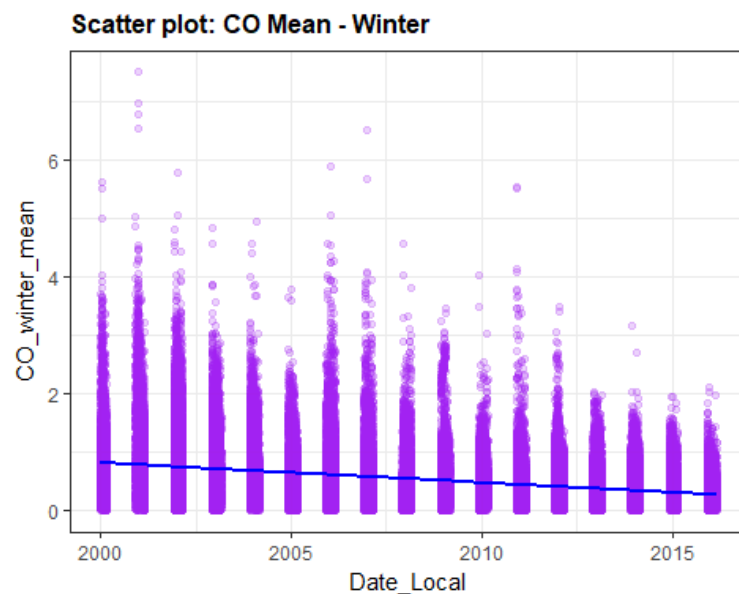
```
CO_summer%>% ggplot(aes(Date_Local,CO_Mean))+ geom_point(color="yellow", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("CO_summer_mean") + theme_bw() + labs(title =
"Scatter plot: CO Mean - Summer") + theme(plot.title = element_text(face = "bold", size =
12))
```



```
CO_fall%>% ggplot(aes(Date_Local,CO_Mean))+ geom_point(color="green", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("CO_fall_mean") + theme_bw() + labs(title =
"Scatter plot: CO Mean - Fall") + theme(plot.title = element_text(face = "bold", size =
12))
```



```
CO_winter%>% ggplot(aes(Date_Local,CO_Mean))+ geom_point(color="purple", alpha = 1/5)+
geom_smooth(method="lm",colour="blue")+ylab("CO_winter_mean") + theme_bw() + labs(title =
"Scatter plot: CO Mean - Winter") + theme(plot.title = element_text(face = "bold", size =
12))
```



CO 는 가을, 겨울에 평균 농도가 높다.

NO₂, CO 그리고 O₃ 가 서로 반대인 경향을 보인 이유는 생성 원인 때문이다. NO₂ 가 강한 자외선¹³을 받아 분해되고 불안정하게 재결합 되는 O₃ 는 자연적으로 여름철에 많았고 미국의

¹³ <https://www.epa.gov/sunsafety/sun-safety-monthly-average-uv-index#tab-12>(접속일 2017.12.22.)

질병관리기관의 조사에 따르면 CO 의 가장 큰 발생장소는 77.6%로 거주지¹⁴가 되었으며 이는 겨울철 난방을 통해 많은 양의 CO 가 발생됨을 알 수 있다.

다음으로, 위치와 관련된 가설들을 세워서 검토해보았다.

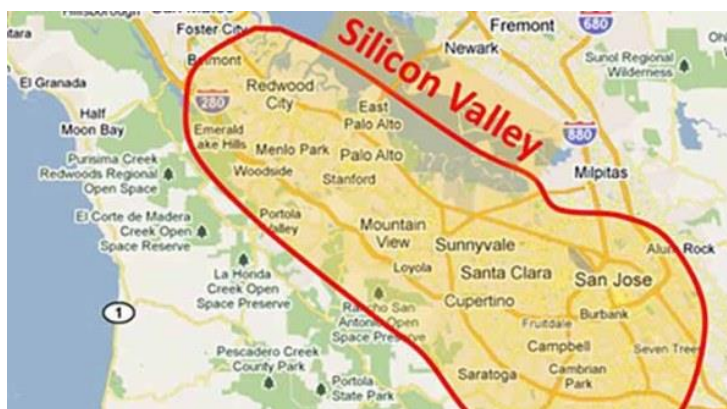


California map



가설4: 캘리포니아 주 안에서 공업지역(남동부)의 대기오염이 심각할 것이다.

캘리포니아 주 안에는 18 개의 군과 35 개의 도시가 있다. 전체 미국인 중 10 퍼센트 이상이 살고 있으며 도시화 수준에서 다른 주 보다 앞서고 있다. 또한 샌프란시스코 만 남쪽으로 실리콘 밸리라고 불리는 첨단 산업 기지가 있어 이 부근에서 오염물질의 농도가 높을 것이라고 생각하여 가설을 선정하였다.



¹⁴ <https://www.cdc.gov/mmwr/preview/mmwrhtml/mm6030a2.htm> (접속일 2017.12.22.)

먼저 그림과 같이 실리콘 벨리라고 불리는 지역의 county 를 조사했다.

```
Pollution%>%filter(State=="California")%>%count(County)
```

```
## # A tibble: 18 x 2
##   County      n
##   <chr> <int>
## 1 Alameda 14206
## 2 Contra Costa 84010
## 3 Fresno 15302
## 4 Humboldt 18546
## 5 Imperial 22420
## 6 Kern 434
## 7 Los Angeles 93381
## 8 Orange 21606
## 9 Riverside 30178
## 10 Sacramento 33032
## 11 San Bernardino 33973
## 12 San Diego 51110
## 13 San Francisco 14444
## 14 Santa Barbara 82998
## 15 Santa Clara 15264
## 16 Santa Cruz 13360
## 17 Solano 26238
## 18 Ventura 5640
```

이 18 개의 County 중에서 Santa Clara, San Francisco, Santa Cruz 이 세 개의 군이 실리콘벨리와 가까운 County 였다. 각 County 별로 오염물질의 평균 수치와 AQI 지수에 따라 순위를 보자. 먼저 mean 을 기준으로 살펴보았다.

```
##① NO2
```

```
Pollution_NO2%>%filter(State=="California")%>%group_by(County)%>%summarise(mean_NO2=mean(NO2_Mean))%>%arrange(desc(mean_NO2))
```

```
## # A tibble: 18 x 2
##   County mean_NO2
##   <chr>     <dbl>
## 1 Kern 28.953775
## 2 Los Angeles 24.696489
## 3 Riverside 18.914977
## 4 San Bernardino 18.616234
## 5 San Diego 18.398854
## 6 San Francisco 16.922068
## 7 Alameda 15.137418
## 8 Imperial 13.956229
## 9 Orange 13.940171
## 10 Fresno 13.144178
## 11 Santa Clara 12.359176
## 12 Ventura 12.051574
## 13 Sacramento 11.508663
## 14 Solano 10.289835
## 15 Contra Costa 10.228791
## 16 Santa Cruz 3.948229
## 17 Santa Barbara 3.936857
## 18 Humboldt 2.309601
```

Kern, LA, Riverside, San Bernardino San Diego 순으로 나타났고 San Francisco 는 6 위, Santa Clara 는 11 위, Santa Cruz 는 16 위로 높은 순위가 아니었다.

```
Pollution_S02%>%filter(State=="California")%>%group_by(County)%>%summarise(mean_S02=mean(SO
2_Mean))%>%arrange(desc(mean_S02))
```

```
## # A tibble: 18 x 2
##       County mean_S02
##       <chr>   <dbl>
## 1    San Diego 2.8075062
## 2      Kern 1.9255256
## 3    Ventura 1.8389816
## 4 San Francisco 1.7616449
## 5   Los Angeles 1.6467138
## 6 Contra Costa 1.4303085
## 7   Riverside 1.4297579
## 8     Orange 1.1628085
## 9    Santa Cruz 1.1331537
## 10 Sacramento 1.0588138
## 11    Alameda 1.0031799
## 12 San Bernardino 0.9905975
## 13     Solano 0.9879000
## 14     Fresno 0.9523963
## 15   Imperial 0.7031819
## 16 Santa Barbara 0.6598056
## 17   Humboldt 0.6136941
## 18   Santa Clara 0.4899848
```

SO2 역시 San Francisco 가 4 위를 차지했지만 나머지 지역들은 순위 밖에 있는 것을 확인할 수 있다. Santa Clara 는 가장 낮은 순위를 차지했다. 오히려 인구 밀집 지역인 Kern, San Diego, Los Angeles 가 더 높은 순위를 차지했다.

```
Pollution_CO%>%filter(State=="California")%>%group_by(County)%>%summarise(mean_CO=mean(CO_M
ean))%>%arrange(desc(mean_CO))
```

```
## # A tibble: 18 x 2
##       County mean_CO
##       <chr>   <dbl>
## 1      Kern 0.8259681
## 2   Imperial 0.7312792
## 3    San Diego 0.7250753
## 4   Los Angeles 0.6302228
## 5 San Francisco 0.5406450
## 6   Riverside 0.5334218
## 7    Alameda 0.4629573
## 8     Solano 0.4291068
## 9 Sacramento 0.4227994
## 10    Orange 0.3961061
## 11 Contra Costa 0.3852445
## 12   Santa Clara 0.3709360
## 13     Fresno 0.3657337
## 14   Santa Cruz 0.3354921
## 15 San Bernardino 0.3193747
## 16   Humboldt 0.3052056
## 17    Ventura 0.3047144
## 18 Santa Barbara 0.2263691
```

```
Pollution_O3%>%filter(State=="California")%>%group_by(County)%>%summarise(mean_O3=mean(O3_M
ean))%>%arrange(desc(mean_O3))
```

```
## # A tibble: 18 x 2
##       County mean_O3
##       <chr>   <dbl>
## 1 Santa Barbara 0.03342543
## 2 San Bernardino 0.03159063
## 3     Fresno 0.03037633
```

```
## 4      Orange 0.02894573
## 5      Riverside 0.02756996
## 6      San Diego 0.02649092
## 7      Imperial 0.02636976
## 8      Santa Cruz 0.02565134
## 9      Sacramento 0.02524203
## 10     Contra Costa 0.02356595
## 11     Ventura 0.02313384
## 12     Humboldt 0.02274430
## 13     Los Angeles 0.02259353
## 14     Solano 0.02255754
## 15     Santa Clara 0.02072476
## 16     San Francisco 0.01959586
## 17     Alameda 0.01856689
## 18     Kern 0.01662702
```

네 개의 오염물질을 모두 살펴보아도 Santa Clara, San Francisco, Santa Cruz 는 두드러지게 높은 순위를 보이지 않았다. San Francisco 는 NO₂, SO₂,와 CO 의 평균 농도에서는 4-6 위에 머무르고 오히려 Kern 군이 1,2 위를 차지했다. O₃ 는 앞에서 살펴보았듯 다른 물질들과 반대로 Kern 군에서 제일 적었고 Santa Clara 는 4 개 오염물질 모두 10 위권 바깥에 있었다. 다음으로 AQI 를 기준으로 보자.

```
Pollution_NO2%>%filter(State=="California")%>%group_by(County)%>%summarise(AQI_NO2=mean(NO2_AQI))%>%arrange(desc(AQI_NO2))
```

```
## # A tibble: 18 x 2
##       County    AQI_NO2
##       <chr>    <dbl>
## 1      Kern 51.834862
## 2 Los Angeles 39.513417
## 3 San Bernardino 34.855376
## 4      Riverside 32.269690
## 5      San Diego 31.985323
## 6      Imperial 31.393723
## 7 San Francisco 27.171144
## 8      Orange 25.128101
## 9      Alameda 24.527309
## 10     Fresno 24.510517
## 11     Sacramento 22.676132
## 12     Ventura 21.807528
## 13 Santa Clara 21.336478
## 14     Solano 18.771494
## 15 Contra Costa 18.106413
## 16 Santa Cruz 9.503593
## 17 Santa Barbara 8.348915
## 18     Humboldt 5.537740
```

```
Pollution_SO2%>%filter(State=="California")%>%group_by(County)%>%summarise(AQI_SO2=mean(SO2_AQI,na.rm=TRUE))%>%arrange(desc(AQI_SO2))
```

```
## # A tibble: 18 x 2
##       County    AQI_SO2
##       <chr>    <dbl>
## 1      Kern 8.6330275
## 2      San Diego 7.1078064
## 3      Los Angeles 5.7899532
## 4 San Francisco 5.6197596
## 5      Ventura 5.2491525
## 6 Contra Costa 5.0002496
## 7      Alameda 3.5605231
```

```
## 8      Orange 3.5385477
## 9      Santa Cruz 3.4392757
## 10     Riverside 3.3675454
## 11     Sacramento 3.2312291
## 12     Solano 3.1518012
## 13 San Bernardino 2.9006227
## 14     Imperial 2.6175767
## 15 Santa Barbara 2.2635795
## 16     Fresno 2.2530468
## 17     Santa Clara 1.1225989
## 18     Humboldt 0.5879687
```

```
Pollution_CO>%>%filter(State=="California")>%>%group_by(County)>%>%summarise(AQI_CO=mean(CO_AQ
I,na.rm=TRUE))>%>%arrange(desc(AQI_CO))
```

```
## # A tibble: 18 x 2
##       County    AQI_CO
##       <chr>    <dbl>
## 1      Kern 16.074074
## 2    Imperial 14.712884
## 3    San Diego 11.372586
## 4 Los Angeles 10.960348
## 5    Riverside 8.988556
## 6 San Francisco 8.362010
## 7    Alameda 7.883913
## 8    Sacramento 7.572416
## 9      Solano 7.372693
## 10     Orange 7.277496
## 11     Fresno 6.647142
## 12     Santa Clara 5.794102
## 13 Contra Costa 5.731207
## 14 San Bernardino 5.679567
## 15    Ventura 5.036222
## 16     Santa Cruz 4.744311
## 17    Humboldt 4.019072
## 18 Santa Barbara 3.237864
```

```
Pollution_O3>%>%filter(State=="California")>%>%group_by(County)>%>%summarise(AQI_O3=mean(O3_AQ
I,na.rm=TRUE))>%>%arrange(desc(AQI_O3))
```

```
## # A tibble: 18 x 2
##       County    AQI_O3
##       <chr>    <dbl>
## 1    Riverside 55.71802
## 2      Fresno 52.80754
## 3 San Bernardino 52.73713
## 4    Imperial 39.72325
## 5    Sacramento 39.24193
## 6      Orange 36.50639
## 7 Santa Barbara 35.73963
## 8    San Diego 34.23213
## 9    Los Angeles 33.81218
## 10    Ventura 32.91412
## 11 Contra Costa 30.54459
## 12      Kern 30.35780
## 13     Santa Clara 28.17939
## 14    Santa Cruz 28.05629
## 15      Solano 27.96356
## 16    Humboldt 25.27003
## 17    Alameda 22.94480
## 18 San Francisco 22.61839
```

AQI 역시 mean 과 마찬가지로 Santa Clara, San Francisco, Santa Cruz 의 순위가 높지 않은 것을 확인할 수 있었다. 첨단산업지역이라고 불리는 곳임에도 캘리포니아 주의 다른 군에 비해 대기오염 지수가 낮은 것을 보면 대기오염에는 공업보다 큰 영향을 끼치는 요인들(환경규제, 교통, 주거생활 등)이 있을 것이라고 판단된다.

마지막으로, California 에 국한되지 않은 전역에 관련한 가설을 세우고 검정했다.

가설5: 공업지역이 많은 주가 대기오염이 심각할 것이다.

캘리포니아 주 안에는 18 개의 군과 35 개의 도시가 있다. 전체 미국인 중 10 퍼센트 이상이 살고 있으며 도시화 수준에서 다른 주 보다 앞서고 있다. 또한 샌프란시스코 만 남쪽으로 실리콘 밸리라고 불리는 첨단 산업 기지가 있어 이 부근에서 오염물질의 농도가 높을 것이라고 생각하여 가설을 선정하였다.

#N02

```
Pollution_N02 %>% group_by(State) %>% summarise(mean_N02=mean(N02_Mean)) %>% arrange(desc(mean_N02))
```

```
## # A tibble: 47 x 2
##       State mean_N02
##       <chr>   <dbl>
## 1 Country Of Mexico 20.31479
## 2 Colorado 19.71049
## 3 Arizona 19.09904
## 4 New York 18.99439
## 5 Massachusetts 18.62205
## 6 New Jersey 18.60761
## 7 District Of Columbia 17.68299
## 8 Michigan 16.84291
## 9 Illinois 15.98321
## 10 Missouri 14.97340
## # ... with 37 more rows
```

```
Pollution_N02 %>% group_by(State) %>% summarise(AQI_N02=mean(N02_AQI)) %>% arrange(desc(AQI_N02))
```

```
## # A tibble: 47 x 2
##       State AQI_N02
##       <chr>   <dbl>
## 1 Country Of Mexico 37.96057
## 2 Arizona 36.16699
## 3 Colorado 36.07811
## 4 Michigan 31.89824
## 5 New Jersey 31.68738
## 6 New York 31.14869
## 7 District Of Columbia 30.59335
## 8 Massachusetts 29.31965
## 9 Missouri 29.19587
## 10 Illinois 28.84914
## # ... with 37 more rows
```

N02 조사결과에서는 공업지역의 주의 수와 대기오염도가 큰 상관관계를 가졌다.

#S02

```
Pollution_S02%>%group_by(State)%>%summarise(mean_S02=mean(S02_Mean))%>%arrange(desc(mean_S02))
```

```
## # A tibble: 47 x 2
##       State mean_S02
##       <chr>   <dbl>
## 1      Alaska 6.092910
## 2    New York 4.819131
## 3 District Of Columbia 4.256948
## 4    Pennsylvania 4.198356
## 5      Kentucky 3.800549
## 6      Missouri 3.502218
## 7    New Jersey 3.457316
## 8      Michigan 3.326067
## 9      Indiana 3.282022
## 10     Virginia 3.160402
## # ... with 37 more rows
```

```
Pollution_S02%>%group_by(State)%>%summarise(AQI_S02=mean(S02_AQI,na.rm=TRUE))%>%arrange(desc(AQI_S02))
```

```
## # A tibble: 47 x 2
##       State AQI_S02
##       <chr>   <dbl>
## 1    Michigan 19.20731
## 2    Kentucky 18.76802
## 3    Missouri 18.59471
## 4    Pennsylvania 16.40746
## 5      Ohio 16.06159
## 6    Indiana 15.47681
## 7    New York 14.56820
## 8      Alaska 14.50607
## 9    Illinois 14.46597
## 10 District Of Columbia 13.02168
## # ... with 37 more rows
```

반면 S02 조사결과에서는 공업지역의 주의 수와 대기오염도와는 관계를 지을 만큼 관련도가 크지 않았다.

#③ CO

```
Pollution_CO%>%group_by(State)%>%summarise(mean_CO=mean(CO_Mean))%>%arrange(desc(mean_CO))
```

```
## # A tibble: 47 x 2
##       State mean_CO
##       <chr>   <dbl>
## 1 Country Of Mexico 0.8577745
## 2 District Of Columbia 0.7958255
## 3      Arizona 0.4885040
## 4      Missouri 0.4684906
## 5    California 0.4578803
## 6      Colorado 0.4451940
## 7    Tennessee 0.4383056
## 8      Florida 0.4308567
## 9      Arkansas 0.4249777
## 10     Alaska 0.4234378
## # ... with 37 more rows
```

```
Pollution_CO%>%group_by(State)%>%summarise(AQI_CO=mean(CO_AQI,na.rm=TRUE))%>%arrange(desc(AQI_CO))
```

```
## # A tibble: 47 x 2
##       State      AQI_CO
##       <chr>    <dbl>
## 1 Country Of Mexico 17.825866
## 2 District Of Columbia 11.707673
## 3 Arizona 9.174156
## 4 Colorado 7.760302
## 5 California 7.606910
## 6 Missouri 7.438742
## 7 Kansas 6.730588
## 8 Illinois 6.668686
## 9 Alaska 6.528340
## 10 Michigan 6.508108
## # ... with 37 more rows
```

c0 조사 결과 공업지역의 수와 대기오염도는 거의 비례한다고 볼 수 있다.

#03

```
Pollution_03%>%group_by(State)%>%summarise(mean_03=mean(O3_Mean))%>%arrange(desc(mean_03))
```

```
## # A tibble: 47 x 2
##       State      mean_03
##       <chr>    <dbl>
## 1 Wyoming 0.03806685
## 2 Tennessee 0.03779823
## 3 Utah 0.03206618
## 4 Nevada 0.03199683
## 5 New Mexico 0.03171082
## 6 South Carolina 0.03166799
## 7 Oklahoma 0.03138986
## 8 Rhode Island 0.03008086
## 9 South Dakota 0.03004211
## 10 Indiana 0.02942556
## # ... with 37 more rows
```

```
Pollution_03%>%group_by(State)%>%summarise(AQI_03=mean(O3_AQI,na.rm=TRUE))%>%arrange(desc(AQI_03))
```

```
## # A tibble: 47 x 2
##       State      AQI_03
##       <chr>    <dbl>
## 1 Tennessee 45.35841
## 2 North Carolina 44.35932
## 3 Kentucky 42.96436
## 4 Utah 42.23614
## 5 Missouri 41.97449
## 6 Indiana 41.59765
## 7 Wyoming 41.45067
## 8 New Mexico 41.33240
## 9 Oklahoma 41.03965
## 10 Nevada 40.38662
## # ... with 37 more rows
```

03 조사결과에서는 공업지역의 주의 수와 대기오염도와는 관계를 지을 만큼 관련도가 크지 않았다.

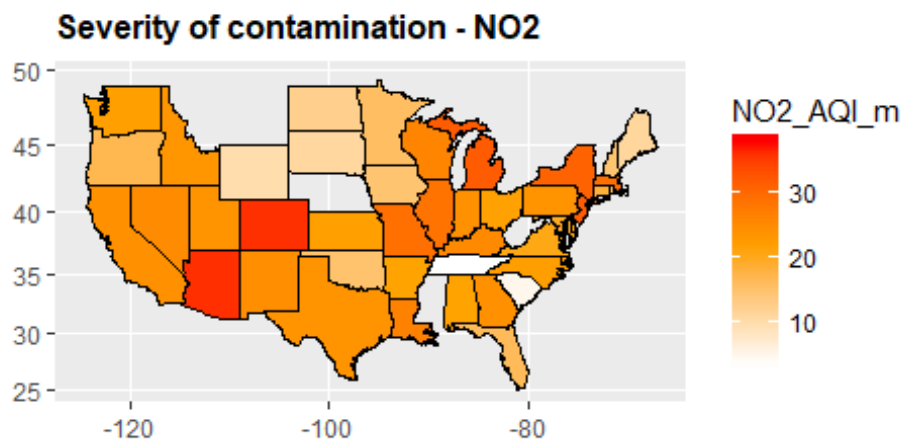
이를 실제 지도에서 미국 지도로 나타내보았다.

```
states_map <- map_data("state")

#NO2
Pollution_NO2_AQI <- Pollution_NO2 %>% group_by(State) %>%
  summarise(NO2_AQI_m = mean(NO2_AQI, na.rm = TRUE))

#state 이름 소문자로 바꾸기
Pollution_NO2_AQI$State <- tolower(Pollution_NO2_AQI$State)

#NO2 지도
ggChoropleth(data=Pollution_NO2_AQI, aes(fill=NO2_AQI_m, map_id=State),
  map=states_map) + labs(title = "Severity of contamination - NO2") +
  theme(plot.title = element_text(face = "bold", size = 12))
```

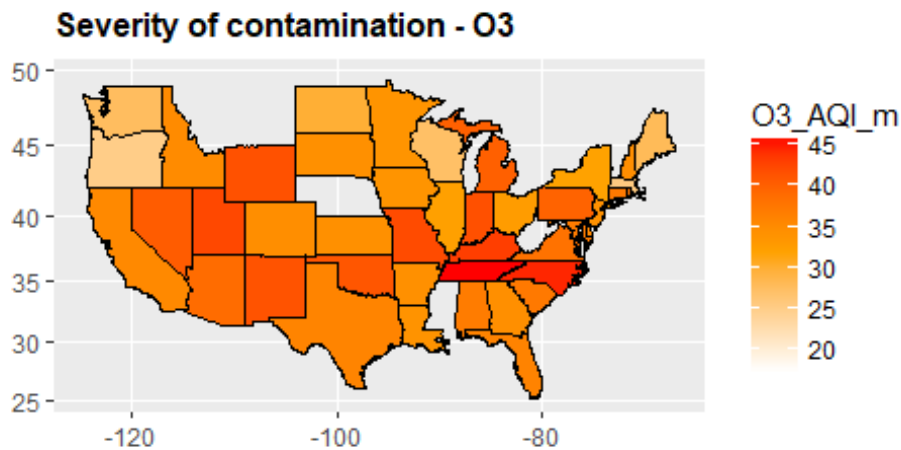


특히 California 주는 눈에 띄게 붉은 색을 띄고 있었다.

```
#O3
Pollution_O3_AQI <- Pollution_O3 %>% group_by(State) %>%
  summarise(O3_AQI_m = mean(O3_AQI, na.rm = TRUE))

#state 이름 소문자로 바꾸기
Pollution_O3_AQI$State <- tolower(Pollution_O3_AQI$State)

#O3 지도
ggChoropleth(data=Pollution_O3_AQI, aes(fill=O3_AQI_m, map_id=State),
  map=states_map) + labs(title = "Severity of contamination - O3") +
  theme(plot.title = element_text(face = "bold", size = 12))
```

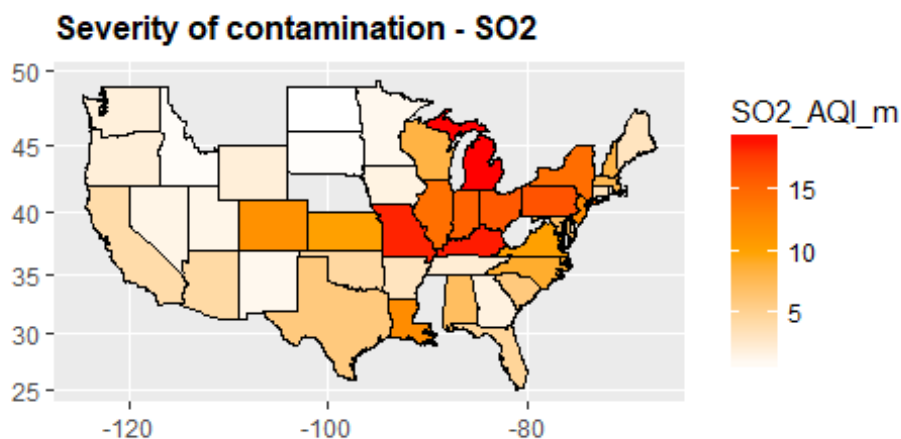


O3 의 경우 지금까지 보았던 것 처럼, 전반적으로 오염물질 농도가 확연히 높아보였다.

```
#SO2
Pollution_SO2_AQI <- Pollution_SO2 %>% group_by(State) %>%
  summarise(SO2_AQI_m = mean(SO2_AQI, na.rm = TRUE))

#state 이름 소문자로 바꾸기
Pollution_SO2_AQI$State <- tolower(Pollution_SO2_AQI$State)

#SO2 지도
ggChoropleth(data=Pollution_SO2_AQI, aes(fill=SO2_AQI_m, map_id=State),
  map=states_map) + labs(title = "Severity of contamination - SO2") +
  theme(plot.title = element_text(face = "bold", size = 12))
```



SO2 는 특이하게 미국 동부에서 붉은 색을 많이 띄었다.

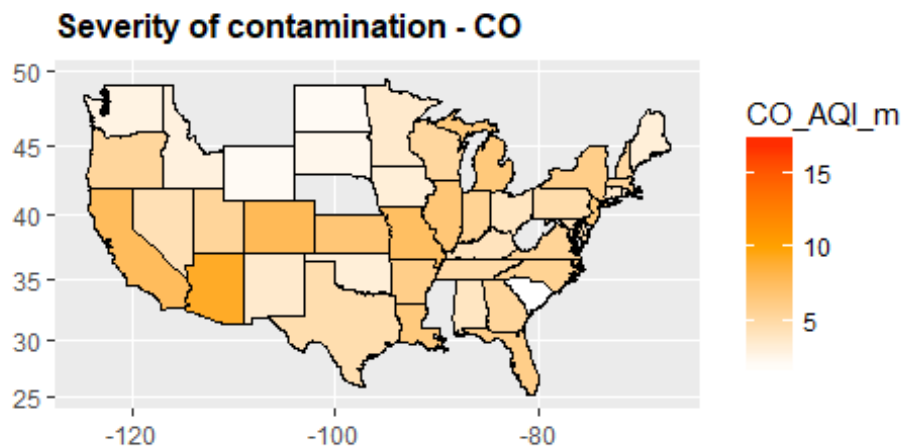
```
#CO
Pollution_CO_AQI <- Pollution_CO %>% group_by(State) %>%
  summarise(CO_AQI_m = mean(CO_AQI, na.rm = TRUE))

#state 이름 소문자로 바꾸기
```

```
Pollution_CO_AQI$State <- tolower(Pollution_CO_AQI$State)
```

#CO 지도

```
ggChoropleth(data=Pollution_CO_AQI, aes(fill=CO_AQI_m, map_id=State),  
map=states_map) + labs(title = "Severity of contamination - CO") +  
theme(plot.title = element_text(face = "bold", size = 12))
```



CO 는 다른 물질에 비해 다행히 붉음이 덜했다.

결과적으로, 미국 공업지역은 대부분 오대호 연안/선벨트지역 - 노스캐롤라이나 주~남부 캘리포니아, 북쪽 지역¹⁵이 많다. 결론적으로, 실제 농도가 높게 나온 주 10 개 찾아서 대조해본 결과 대다수가 미국 남부 지방의 공업지역이었다.

¹⁵ <http://study.zumst.com/upload/00-W33-00-42-02/W33-00-42-02-%EB%B6%81%EC%95%84%EB%A9%94%EB%A6%AC%EC%B9%B4%20%EA%B3%B5%EC%97%85%20%EC%A7%80%EC%97%AD.png>(접속일 2017.12.22)

02-%EB%B6%81%EC%95%84%EB%A9%94%EB%A6%AC%EC%B9%B4%20%EA%B3%B5%EC%97%85%20%EC%A7%80%EC%97%AD.png(접속일 2017.12.22)

V. CONCLUSION

[분석내용 요약]

1. dataset 정리

분석에 용이하지 않은 자료들에 다음과 같이 정리했다.

(기존자료) Pollution - 관측치 수: 1746661 / 변수 개수: 29 (범주형(9) + 연속형(21))

(수정자료) Pollution_NO2 - 관측치 수: 413759 / 변수 개수 (범주형(6)+연속형(10))

Pollution_SO2 - 관측치 수: 836741 / 변수 개수 (범주형(6)+연속형(10))

Pollution_O3 - 관측치 수: 416130 / 변수 개수 (범주형(6)+연속형(10))

Pollution_CO - 관측치 수: 843691 / 변수 개수 (범주형(6)+연속형(10))

2. 각 가설에 대한 내용 및 결론 요약

1) 출퇴근시간의 오염물질의 Max_Value가 높게 측정될 것이다.

자료 분석 과정 중 네 가지 오염물질의 농도가 같이 움직이지 않는다는 것을 알게 되었고, 출·퇴근 시간에 높은 농도를 기록한 물질은 NO2와 CO였고, O3는 정 반대의 결과를 나타냈다.

2) 2000년에서 2016년으로 갈수록 오염물질의 농도가 높아질 것이다.

O3를 제외한 NO2, SO2, CO는 지속적으로 감소하는 추세를 보였다. O3는 4-5년을 주기로 증감을 반복하였으나 2000년에 비해 2016년에는 월등히 높다. O3의 원인이 되는 물질들에는 계속 주의를 가져야 하겠다.

3) 미국의 늦가을부터 겨울까지의 오염물질의 농도가 가장 높을 것이다.

1월부터 12월까지 NO2, SO2, CO 모두 11월~2월이 가장 농도가 높았고 O3는 포물선 모양을 그리며 5~9월에 가장 농도가 높았다. NO2가 강한 자외선을 받아 분해되어 불안정하게 재결합 되는 O3는 자연적으로 여름철에 많았고, CO는 겨울철 난방으로 인해 발생되며 농도가 높은 것을 알게 되었다.

4) 캘리포니아 주 안에서 공업지역(남동부)의 대기오염이 많을 것이다.

Mean과 AQI 모두 공업지역의 순위는 높지 않았고 오히려 다른 군에 비해 대기 오염 수준이 낮았다. 이는 대기 오염에 큰 영향을 끼치는 요인은 공업단지 자체 보다는 인구밀집(교통, 주거생활), 환경규제가 있다고 판단된다.

5) 공업지역이 많은 주가 대기오염이 높을 것이다.

이 가설은 참으로 NO₂는 미국 남서부의 California 지역에서, SO₂는 동부에서 특히 심각함을 보였다.

[팀프로젝트를 마치며]

검색을 통해 가장 분석하기 쉬운 데이터를 선정하려고 노력했으나, 데이터 셋에 처음부터 있었던 문제는 피할 수 없었다. 이를 정리하는데 꽤나 오랜 시간이 걸렸으며, 사실 정리를 했어도 하루에 max_value가 두 번 관측되는 경우 등으로 인해 분석하기 완벽한 자료로는 볼 수 없었다. 이는 기본적으로 어떠한 방식으로 이 데이터가 합쳐졌는지 자세한 설명 없이 kaggle 과 EPA 사이트에서 추측해야 했고, 최선의 결과를 얻기 위해 어느 정도 묵인하게 되었다. 이 과정에서 우리 조가 발견하지 못한 또 다른 문제점도 있을 수 있다.

하지만 이번 팀플로 조원 모두 마찬가지로 자료 정리가 데이터 분석의 거의 반절이라는 점에 매우 동감했다. 이를 어떻게 처리할 것인지에 대한 고민을 했고, 팀원들의 지식뿐만 아니라 주변 인물에게까지 도움을 받아 이를 해결하기 위해 애썼다. 결과적으로, 실제 자료 분석에 한 발자국이라도 다가간 만큼 뿌듯했다.