# Project Seoul Air

서울시 시간 평균 대기오염도 정보에 따른 초미세먼지 농도 예측 분석

과목: 빅데이터의 분석과 이해
날짜: 2018.12.13
이름: 강민정
학번: 2014100037

# 목차

1장. 서론

1.1 분석의 목적
1.2 분석 데이터에 관한 기본 정보

2장. Codebook

2.1 데이터 클렌징
2.2 코드북

3장. Descriptive Analytics

4장. Predictive Analytics

4.1 Preprocessing

4.2 Data Splitting

4.3 Modeling

4.4 Model Evaluation

5장. 결언

# 1장. 서론

## 1.1 분석의 목적

본 보고서에서는 최근 몇 년 사이 환경 및 정치 분야를 넘어 시민들의 일상 속 가장 큰 관심사로 부상한 미세먼지에 대해 알아보고자 한다.

흔히 직경 10 μg/m 이하의 작은 먼지 입자를 아울러 말하는 '미세먼지' 중에서도 입자가 2.5μg/m 이하로, 보다 작고 위험성은 높은 초미세먼지(PM2.5)에 집중하였고, 어떠한 지역 및 시간대에서, 혹은 어떠한 대기 환경이 조성되었을 때 농도가 높이 나타나는지 예측분석을통해 알아보고자한다.

## 1.2 분석 데이터에 관한 기본 정보

- 출처: 서울열린데이터광장(data.seoul.go.kr)의 실시간 데이터를 추출하여 저장한 Kaggle 의 "Air pollutants measured in Seoul"
- URL: https://www.kaggle.com/jihyeseo/seoulairreport
- 내용: 2017 년 11 월 17 일 23 시부터 2017 년 11 월 24 일 23 시까지 서울시내 25 개 구의 대기오염 측정소에서 한 시간 간격으로 측정한 시간 평균 대기오염도
- 크기: 4225 행 * 9 열

```
head(A)
```

```
##         Date Time District   NO2    O3  CO   SO2 PM10 PM2.5
## 1 2017-11-24 2300  Gangnam 0.038 0.004 0.4 0.005   16    10
## 2 2017-11-24 2200  Gangnam 0.031 0.008 0.4 0.005   17     9
## 3 2017-11-24 2100  Gangnam 0.025 0.012 0.4 0.005   18    11
## 4 2017-11-24 2000  Gangnam 0.033 0.007 0.4 0.005   21    12
## 5 2017-11-24 1900  Gangnam 0.033 0.008 0.4 0.005   20    10
## 6 2017-11-24 1800  Gangnam 0.026 0.011 0.4 0.005   21    10
```

데이터는 총 4225 개의 행과 9 개의 열로 이루어져있으며, 각각의 행은 한 측정소에서 측정한 시간 별 평균 대기오염도 관측 개별 값에 해당한다.

9 개의 열은 위와 같이 날짜(Date), 시간(Time), 구(District) 등 관측 환경에 관한 항목과, 이산화질소(NO2), 오존(O3), 일산화탄소(O3), 아황산가스(SO2), 미세먼지(PM10), 초미세먼지(PM2.5) 등 여섯 가지 대기오염원으로 구성되어 있다. 여섯 가지 오염원 중 ppm(parts per million) 단위를

사용하는 이산화질소, 오존, 일산화탄소, 아황산가스는 소수점 단위의 실수 값(numeric), 세제곱미터 당 마이크로그램(μg/m3)을 단위로 사용하는 미세먼지와 초미세먼지는 정수 값(integer)을 가지는 것을 볼 수 있다.

이 보고서에서는 분석을 위해 R 프로그램의 ggplot, caret, tidyverse, ROCR, pROC, rpart 와 rpart.plot 등의 라이브러리를 사용하였다.

# 2장. Codebook

## 2.1 데이터 클렌징

데이터 코드북 작성에 앞서 데이터 클렌징 작업을 진행하였다. 클렌징 작업에는 엑셀과 R 두 가지 툴을 사용하였다.

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | 측정일시 | 측정소명 | 이산화질소농도(ppm) | 오존농도(ppm) | 일산화탄소농도(ppm) | 아황산가스(ppm) | 미세먼지(㎍/㎥) | 초미세먼지(㎍/㎥) |
| 2 | 201711242300 | 강남구 | 0.038 | 0.004 | 0.4 | 0.005 | 16 | 10 |
| 3 | 201711242200 | 강남구 | 0.031 | 0.008 | 0.4 | 0.005 | 17 | 9 |
| 4 | 201711242100 | 강남구 | 0.025 | 0.012 | 0.4 | 0.005 | 18 | 11 |
| 5 | 201711242000 | 강남구 | 0.033 | 0.007 | 0.4 | 0.005 | 21 | 12 |

*Figure 1. Original Dataset*

우선 원본 데이터를 엑셀을 이용하여 1 차 클렌징을 진행하였다. 한글로 표기되어있는 열 이름을 모두 영문으로 변경하고, 찾기와 모두 바꾸기 기능을 이용해 측정소 명을 모두 영문으로 변경하였다. 이후 데이터 나누기 기능을 이용해 날짜와 시간을 두 개의 열로 분리하였고, 셀 서식 기능을 이용해 날짜의 형태를 20171124 에서 2017-11-24 와 같은 형태로 변경하였다.

| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Date | Time | District | NO2 | O3 | CO | SO2 | PM10 | PM2.5 |
| 2 | 2017-11-24 | 2300 | Gangnam | 0.038 | 0.004 | 0.4 | 0.005 | 16 | 10 |
| 3 | 2017-11-24 | 2200 | Gangnam | 0.031 | 0.008 | 0.4 | 0.005 | 17 | 9 |
| 4 | 2017-11-24 | 2100 | Gangnam | 0.025 | 0.012 | 0.4 | 0.005 | 18 | 11 |
| 5 | 2017-11-24 | 2000 | Gangnam | 0.033 | 0.007 | 0.4 | 0.005 | 21 | 12 |

*Figure 2. Primary Cleansed Data*

이후 엑셀로 1 차 클렌징이 완료된 csv 파일을 R 의 na.omit 함수를 이용해 누락 값인 NA 값을 모두 제거하였다. 더 나아가서, 데이터 내 포함된 관측 값이 11 월 17 일 23 시부터 24 일 23 시까지였는데, 17 일은 나머지 7 일과 달리 24 시간 전체의 데이터가 아닌 한 시간의 데이터 밖에 포함되지 않았기 때문에 제외하기로 결정하였다. 또 substr 함수를 이용하여 2300 과 같이 네 자리로 표기 된 시간을 23 처럼 두 자리로 줄였다.

```
> A <- na.omit(A)
> A <- filter(A,Date!="2017-11-17")

> A$Time <- substr(A$Time, 1, 2)
```

마지막으로 예측 분석을 통해 알아보고자 하는 값인 PM2.5(초미세먼지)를 농도에 따라 다섯 가지 등급으로 분류하였다. 다섯 가지 등급의 분류기준은 PM2.5 의 24 시간 평균치에 대한 WHO 의 Air Quality Guideline 권고 기준을 따랐다. 등급별 자세한 내용은 뒤에서 살펴 볼 코드 북에 설명되어 있다.

```
> A <- dplyr::mutate(A, Y = ifelse( A$PM2.5 < 25, "AQG",
+                                   ifelse( A$PM2.5 < 37.5, "IT3",
+                                           ifelse( A$PM2.5 < 50,"IT2",
+                                                   ifelse( A$PM2.5 < 70,"IT1
","Off.T")))))
```

위와 같은 클렌징 과정을 모두 거친 데이터의 형태는 다음과 같다.

```
head(air)

##         Date Time District   NO2    O3  CO   SO2 PM10 PM2.5   Y
## 1 2017-11-24   23  Gangnam 0.038 0.004 0.4 0.005   16    10 AQG
## 2 2017-11-24   22  Gangnam 0.031 0.008 0.4 0.005   17     9 AQG
## 3 2017-11-24   21  Gangnam 0.025 0.012 0.4 0.005   18    11 AQG
## 4 2017-11-24   20  Gangnam 0.033 0.007 0.4 0.005   21    12 AQG
## 5 2017-11-24   19  Gangnam 0.033 0.008 0.4 0.005   20    10 AQG
## 6 2017-11-24   18  Gangnam 0.026 0.011 0.4 0.005   21    10 AQG

dim(air)

## [1] 4095   10
```

## 2.2 코드북

다음은 각 Column Name 에 대한 설명과, 예측하고자 하는 PM2.5 의 등급에 대한 정보가 담긴 kable 코드북이다.

### 2.2.1 Description of Variables

| Variable | Description | Value |
|---|---|---|
| Date | Date of Measurement | YYYY-MM-DD |
| Time | Time of Measurement | HHMM |
| District | District Name of Measurement Station | Gangnam |
| NO2 | Nitrogen Dioxide (ppm) | 0.038 |
| O3 | Ozone (ppm) | 0.004 |

| | | |
|---|---|---|
| CO | Carbon Monoxide (ppm) | 0.4 |
| SO2 | Sulfur Dioxide (ppm) | 0.005 |
| PM10 | PM10 Fine Dust (microgram/m3) | 16 |
| PM2.5 | PM2.5 Fine Dust (microgram/m3) | 10 |
| Y | PM2.5 Classification | AQG |

### 2.2.2. Classification of PM2.5

| PM2.5 | Y | Description |
|---|---|---|
| 25 | AQG | Threshold of cardiopulmonary and lung cancer mortality increase in response to long-term exposure to PM2.5 |
| 37.5 | IT3 | Mortality risk 3% higher than AQG |
| 50 | IT2 | Mortality risk 9% higher than AQG |
| 75 | IT1 | Mortality risk 15% higher than AQG |
| 75+ | Off.T | Extremely high mortality risk |

## 3장. Descriptive Analytics

Predictive Analytics 를 진행하기에 앞서 Descriptive Analytics 를 통한 과거 데이터의 시각화를 진행해보았다. ggplot2 를 사용하여 boxplot 과 density graph 를 통해 날짜 별 PM2.5 의 분포를 살펴본 결과, 11 월 21 일과 22 일에 초미세먼지 농도가 높았음을 알 수 있다.

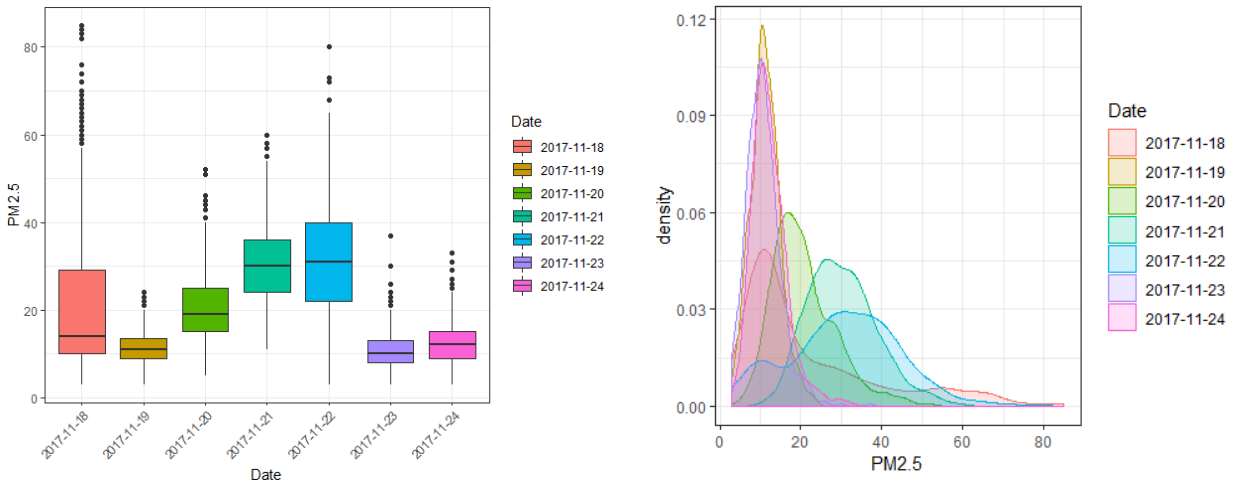



*Figure 3. PM2.5 Concentration by Date*

이후 구역별, 시간대별 PM2.5 분포를 살펴보기 위해 아래와 같이 bot plot 을 먼저 사용해보았으나, 눈에 띄는 차이를 발견하기 어려웠다. 따라서 PM2.5 를 등급별로 묶은 Y 열을 이용하여 bar graph 를 그려 다시 비교해보았다. 그 결과 동작구, 광진구, 양천구가 다른 구역에 비해 PM2.5 농도가 더 높게 나타나고, 12 시에서 5 시 정도의 새벽시간대가 PM2.5 농도가 높음을 알 수 있었다. Bar graph 를 그림에 있어서는 데이터 클렌징 과정에서 NA 값을 제외함으로써 각 항목당 데이터의 개수가 미세하게 차이가 있어 stack 방식이 아닌 fill 방식을 이용해 그래프를 그렸다.

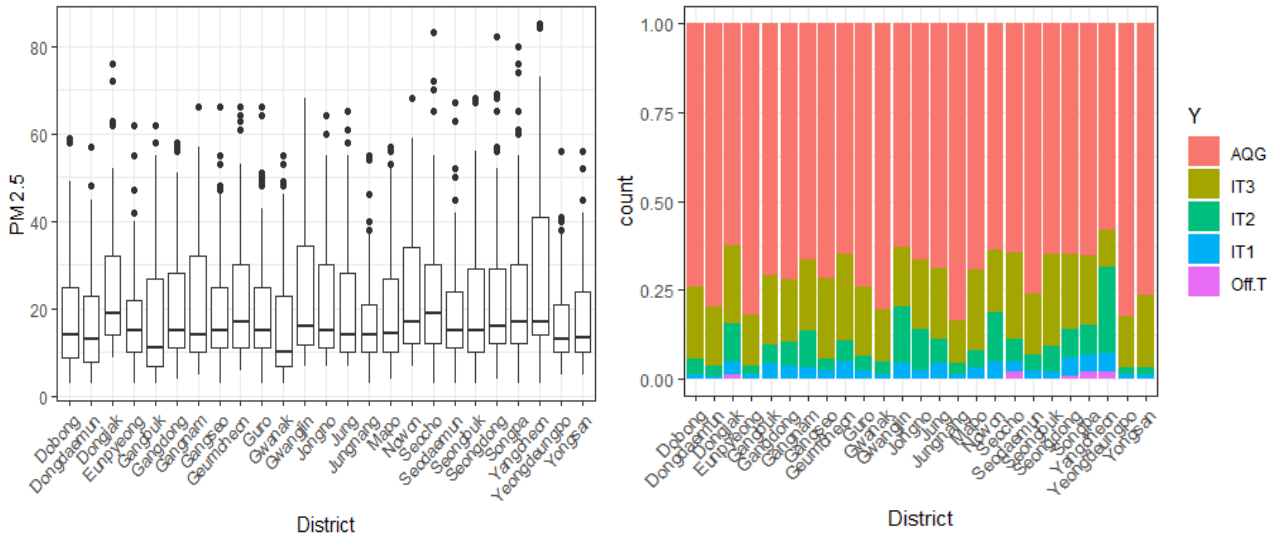



*Figure 4. PM2.5 Concentration by District*

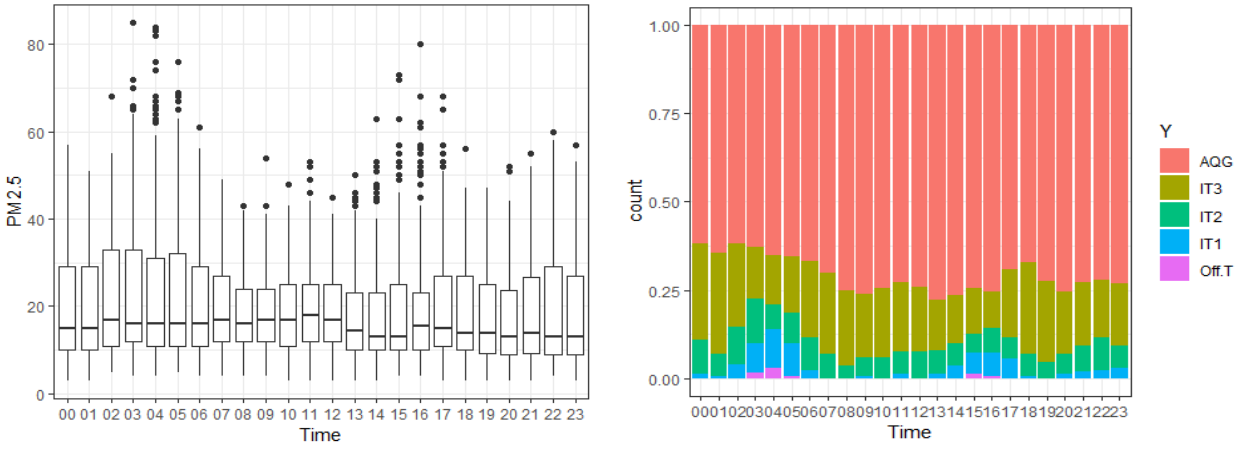



*Figure 5. PM2.5 Concentration by Time*

다음으로는 아래와 같이 PM2.5 와 나머지 다섯 가지 대기오염원의 상관관계를 scatter plot 으로 나타내보았다. 그 결과 PM2.5 는 PM10 과는 강한 양의 상관관계, NO2 와 CO 와는 약한 양의 상관관계를 가지며, O3 과 SO2 는 거의 상관관계가 없음을 볼 수 있었다. ggplot2 이 아닌 caret 의 featureplot 을 이용해 density 그래프를 그려보았을 때에도 비슷한 결과를 확인할 수 있었다.

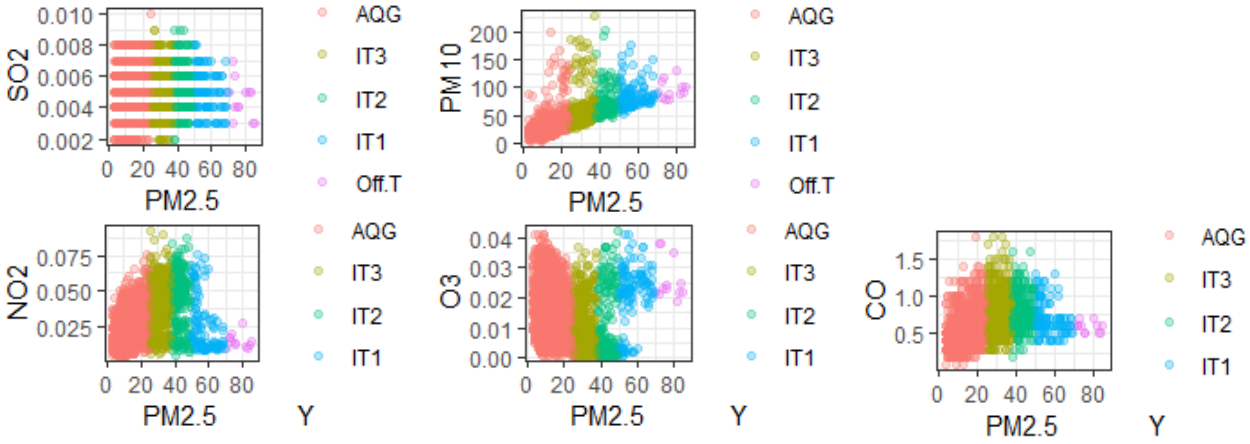



*Figure 6. PM2.5 and other pollutants (scatter plot)*

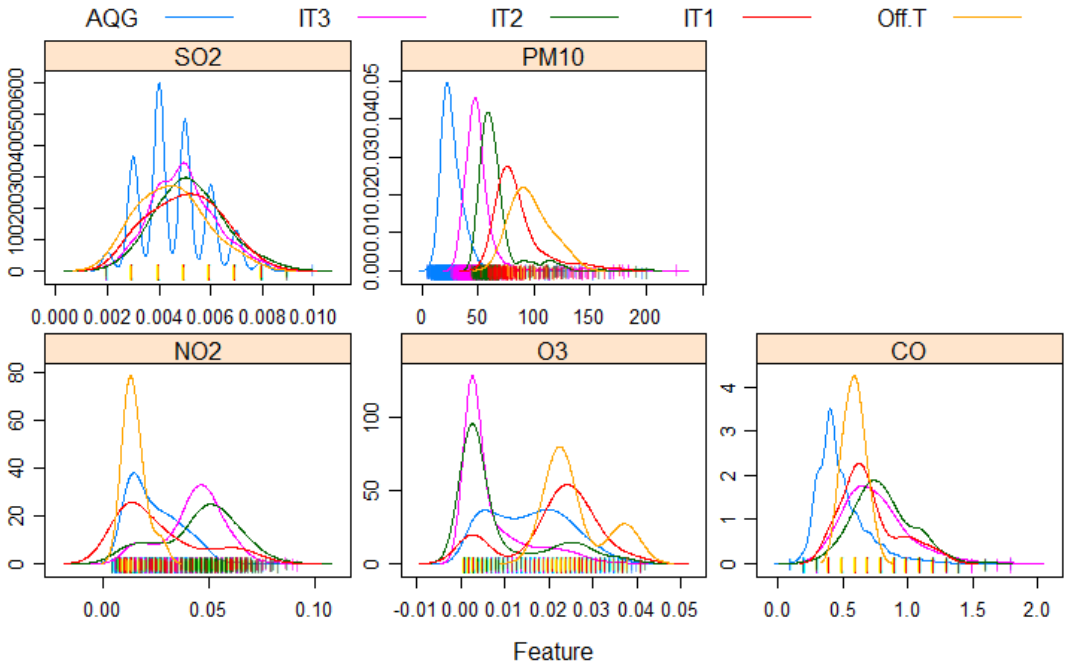



*Figure 7. PM2.5 and other pollutants (density)*

# 4장. Predictive Analytics

## 4.1 Preprocessing

```
> nzv <- nearZeroVar(A[,4:8], saveMetrics= TRUE)
> nzv

     freqRatio percentUnique zeroVar   nzv
NO2   1.123377     1.9780220   FALSE FALSE
O3    1.762712     1.0256410   FALSE FALSE
CO    1.594880     0.4395604   FALSE FALSE
SO2   1.065237     0.2197802   FALSE FALSE
PM10  1.006135     3.6874237   FALSE FALSE
```

Predictive Analytics 모델을 만들기 위해 우선 데이터 프로세싱을 시도하였다. 우선 분산이 0 이거나 0 에 가까운 변수가 있는지 nearZeroVar 을 이용해 살펴보았으나 모두 FALSE 값을 얻어 nzv 값을 제거할 필요는 없었다.

이후 preProcess 함수를 이용해 numeric 및 integer 값을 갖는 다섯 가지 대기 오염 수치에 대한 표준화 과정을 거쳤다.

## 4.2 Data Splitting

데이터를 Training Data 와 Testing Data 로 나누기 전에, predictive analytics 를 진행할 predictor 로는 Time, District, 그리고 NO2, O3, CO, SO2, PM10 등 다섯 가지 대기 오염원까지 총 7 가지 predictor 를 사용하기로 하였다. 앞선 descriptive analytics 에서는 과거 데이터 분석 위해 Date 항목도 사용하였지만, 대기 환경에 따른 PM2.5 농도를 예측하는데 있어서 날짜를 사용하는 것은 예측의 정확도를 실제 정확도와 차이가 발생하게 할 것으로 판단하여 제외하였다.

데이터는 Training Data 70%, Testing Data 30%로 나누었다.

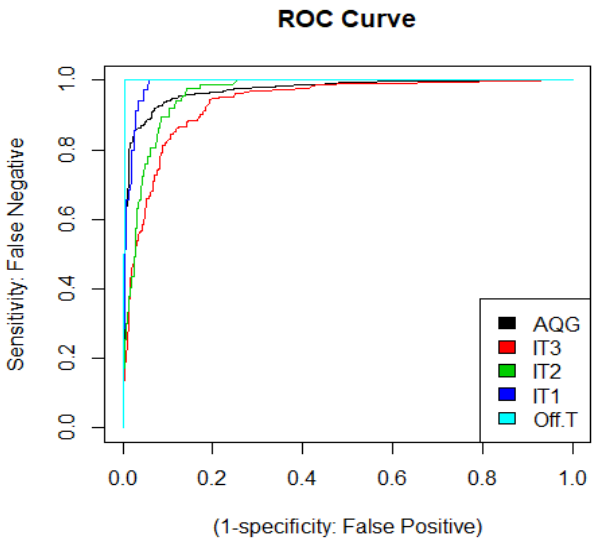
## 4.3 Modeling

모델링은 caret 패키지의 Random Forest, R Partition, Neural Network 등의 기법을 사용하여 보았다. Random Forest 의 샘플링 방식은 bootstrapping, K-fold cross validation, 그리고 repeated cross validation 의 세가지를 비교해보았다.

## 4.4 Model Evaluation

### 4.4.1 Random Forest

```
> model.rf <- caret::train(myModel, method = "rf", data = trData)
2870 samples
   5 predictor
   5 classes: 'AQG', 'IT3', 'IT2', 'IT1', 'Off.T'
Resampling: Bootstrapped (25 reps)
Accuracy : 0.862

                     Class: AQG Class: IT3 Class: IT2 Class: IT1 Class: Off.T
Sensitivity              0.9505     0.7241    0.44828    0.68571    0.0000000
Specificity              0.8543     0.9204    0.98155    0.98655    0.9991817
```




```
AUC

 [[1]] AQG
[1] 0.9720727

[[1]] IT3
[1] 0.9361609

[[1]] IT2
[1] 0.9620074

[[1]] IT1
[1] 0.9883794

[[1]] Off.T
[1] 0.9976814
```

**K-Fold Cross Validation**

```
> my_trControl1 <- trainControl(method = "cv",
+                               number = 5)
> model.rf1 <- train(Y ~ ., method = "rf", data = trData, trControl = my_trCo
ntrol1)
```

**Accuracy : 0.8645**

**Cross Validation**

```
> my_trControl2 <- trainControl(method = "repeatedcv",
+                               number = 5,
+                               repeats = 3)
```

```
> model.rf2 <- train(Y ~ ., method = "rf", data = trData, trControl = my_trCo
ntrol2)
```

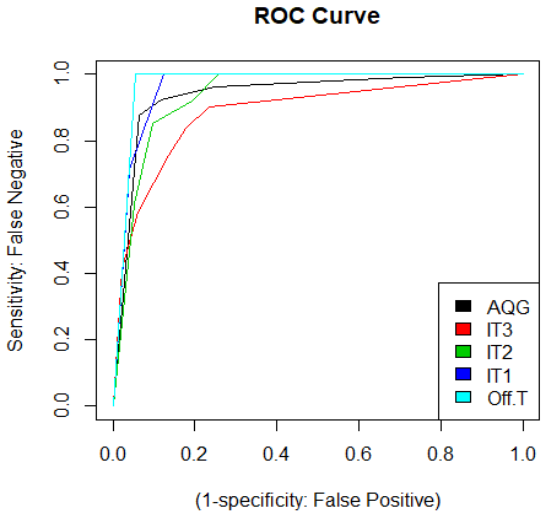
**Accuracy : 0.862**

랜덤포레스트 시행 시 여러 가지 샘플링 방법을 사용해보았으나 Accuracy 와 AUC 등에서 큰 차이는 없었다.

### 4.4.2 R Partition

**Accuracy : 0.8294**

| | Class: AQG | Class: IT3 | Class: IT2 | Class: IT1 | Class: Off.T |
|---|---|---|---|---|---|
| Sensitivity | 0.9240 | 0.5862 | 0.60920 | 0.71429 | 0.000000 |
| Specificity | 0.8824 | 0.9376 | 0.94903 | 0.96050 | 1.000000 |




AUC

[[1]] AQG
[1] 0.938782

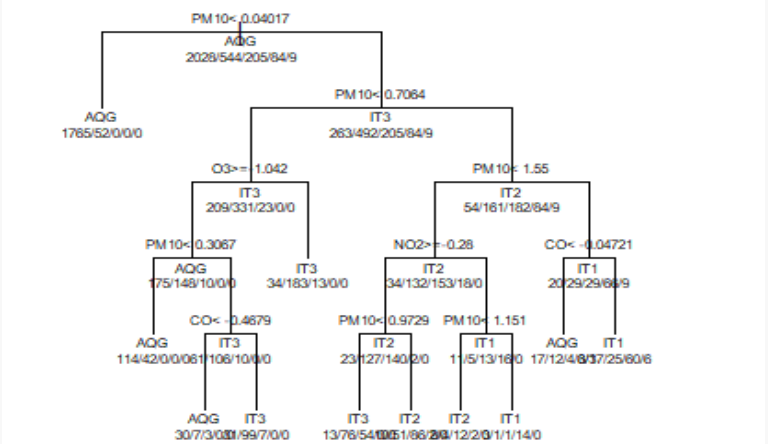
[[1]] IT3
[1] 0.8845301

[[1]] IT2
[1] 0.9386249

**[[1]] IT1**
**[1] 0.962485**

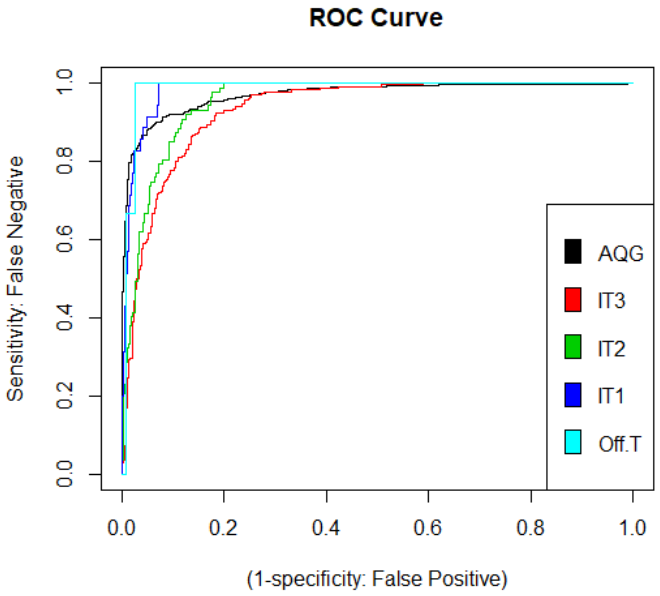
**[[1]] Off.T**
**[1] 0.9717676**

**R Part Plot Tree**



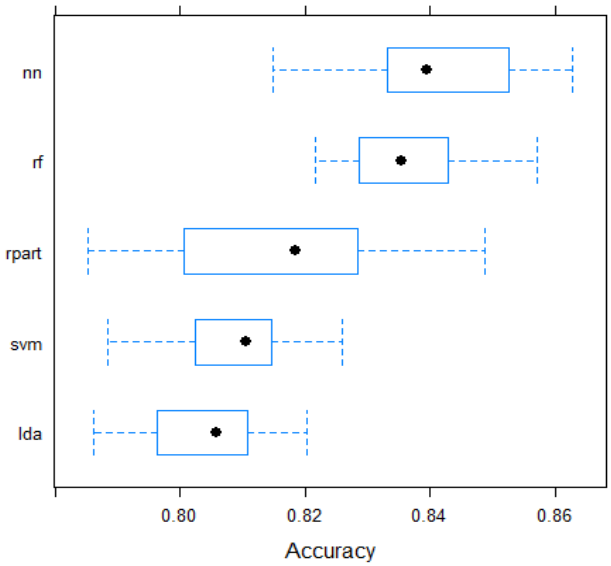
### 4.4.3 Neural Network

```
Accuracy : 0.8588

Statistics by Class:
                     Class: AQG Class: IT3 Class: IT2 Class: IT1 Class: Off.T
Sensitivity              0.9516     0.6983    0.43678    0.74286     0.000000
Specificity              0.8095     0.9305    0.98418    0.98487     1.000000
```




```
AUC
[[1]] AQG
[1] 0.9710045

[[1]] IT3
[1] 0.9339601

[[1]] IT2
[1] 0.9548108


[[1]] IT1
[1] 0.9837935


[[1]] Off.T
[1] 0.9858156
```

보고서에서는 생략되었지만 Random Forest, R Partition, Neural Network, Support Vector Machine, Linear Discriminant Analysis 등 다섯 가지 모델링 방식을 사용해본 결과, Random forest 와 Neural Network 가 비교적 높은 Accuracy 를 보이는 것을 알 수 있었다.




모든 모델에서 PM2.5 값이 가장 높은 두 등급인 IT1 과 Off.T 의 AUC 가 가장 높았다.

## 5장. 결언

본 보고서에서는 Descriptive Analytics 를 통해 PM2.5 의 농도가 보이는 패턴을 읽어내려 시도하였고, Predictive Analytics 를 통해 PM2.5 가 높게 나타나는 환경을 예측해보려 하였다. 그 결과 동작구, 광진구, 양천구에서, 또는 밤 12 시에서 새벽 5 시 정도의 시간대에 PM2.5 가 높게 나타나는 패턴을 볼 수 있었고, PM2.5 농도는 PM10, NO2, CO 농도와 양의 상관관계를 보였다. 또 Random Forest 와 Neural Network 등의 기법으로 모델링을 통해 PM2.5 의 수준을 예측할 수 있는 모델을 작성하였다.

하지만 PM2.5 농도를 비롯한 대기오염도는 본 보고서에서 분석한 변수 외에도 풍향, 풍속, 강수량과 같은 날씨 및 시내 교통량 등 수많은 요소로부터 영향을 받는다. 따라서 이 분석은 지역, 시간대, 대기오염도 외의 관련된 중요 변수를 모두 포함하지 못했다는 한계를 지닌다.

비록 불완전한 분석이지만, 본 분석을 통해 환경 관측 분야에서 데이터 마이닝을 사용할 수 있는 가능성에 대해 고민해볼 수 있었다. 본 분석이 조금이나마 일반 시민들에게는 어떠한 시간대와 지역에서 초미세먼지를 조심해야하는지에 대한 제언이 될 수 있고, 지자체로 하여금 초미세먼지 농도가 높이 나타날 상황을 미리 예측하게 하여 통합적인 대기오염 관리 대책에 참조할 수 있는 분석 자료가 될 수 있기를 기대한다.

## 참조문헌


- World Health Organization, 2005, WHO Air Quality Guidelines for Particulate Matter, Ozone, Nitrogen Dioxide and Sulfur Dioxide
- Kaggle, ttps://www.kaggle.com/jihyeseo/seoulairreport
- Max Kuhn, 2018, The Caret Package (http://topepo.github.io/)
- 정호원 교수님 수업 자료