8개 도시 월별 미세먼지 측정량

서울 미세먼지

미세먼지량 예측 모형

20160131 김지현

2020 10 25

```
library(tidyverse)
```

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

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

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

```
library(tsibble)
library(fpp3)
```

```
## -- Attaching packages --------------------- fpp3 0.3 --
```

```
## √ lubridate   1.7.9     √ feasts      0.1.5
## √ tsibbledata 0.2.0     √ fable       0.2.1
```

```
## -- Conflicts ------------------------ fpp3_conflicts --
## x lubridate::date()     masks base::date()
## x dplyr::filter()       masks stats::filter()
## x lubridate::interval() masks tsibble::interval()
## x dplyr::lag()          masks stats::lag()
```

```
setwd('C:/Users/JIHYUN/Desktop/수업/통계학특강/2차과제')
```

8개 도시 월별 미세먼지 측정량

데이터 전처리

- tot는 제거

```
pm10w <- readr::read_csv('PM10w.csv') %>%
            select(-tot)
```

```
## Parsed with column specification:
## cols(
##   yymm = col_character(),
##   tot = col_double(),
##   seoul = col_double(),
##   busan = col_double(),
##   daegu = col_double(),
##   incheon = col_double(),
##   gwangju = col_double(),
##   daejeon = col_double(),
##   ulsan = col_double(),
##   sejong = col_double()
## )
```

```
head(pm10w)
```

```
## # A tibble: 6 x 9
##   yymm     seoul busan daegu incheon gwangju daejeon ulsan sejong
##   <chr>    <dbl> <dbl> <dbl>   <dbl>   <dbl>   <dbl> <dbl>  <dbl>
## 1 2010. 01    59    47    56      64      46      48    46     NA
## 2 2010. 02    50    44    49      54      39      39    44     NA
## 3 2010. 03    61    64    69      67      65      52    60     NA
## 4 2010. 04    49    50    47      55      42      41    47     NA
## 5 2010. 05    56    56    55      62      62      52    54     NA
## 6 2010. 06    51    46    47      57      39      40    50     NA
```

결측확인

```
colSums(is.na(pm10w))
```

```
##    yymm   seoul   busan   daegu incheon gwangju daejeon   ulsan  sejong
##       0       0       0       0       0       0       0       0      72
```

```
pm10w$sejong[is.na(pm10w$sejong)]
```

```
##  [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [26] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [51] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
```

WDF를 LDF로 변환

```
pm10w <- pivot_longer(pm10w,
             col = c(-yymm,seoul,busan,incheon,
                     gwangju,daejeon,ulsan,sejong),
             names_to = 'city',
             values_to = 'y')
pm10w
```

```
## # A tibble: 984 x 3
##    yymm     city        y
##    <chr>    <chr>   <dbl>
##  1 2010. 01 seoul      59
##  2 2010. 01 busan      47
##  3 2010. 01 daegu      56
##  4 2010. 01 incheon    64
##  5 2010. 01 gwangju    46
##  6 2010. 01 daejeon    48
##  7 2010. 01 ulsan      46
##  8 2010. 01 sejong     NA
##  9 2010. 02 seoul      50
## 10 2010. 02 busan      44
## # ... with 974 more rows
```

tisbble로 변환

- yymm칼럼의 데이터형을 시간형태로 변환

```
pm10w <- pm10w %>%
  mutate(yymm = yearmonth(yymm)) %>%
  as_tsibble(key=city,index= yymm)
pm10w
```

```
## # A tsibble: 984 x 3 [1M]
## # Key:       city [8]
##       yymm city      y
##      <mth> <chr> <dbl>
##  1  2010 1 busan    47
##  2  2010 2 busan    44
##  3  2010 3 busan    64
##  4  2010 4 busan    50
##  5  2010 5 busan    56
##  6  2010 6 busan    46
##  7  2010 7 busan    41
##  8  2010 8 busan    42
##  9  2010 9 busan    38
## 10 2010 10 busan    41
## # ... with 974 more rows
```

데이터 탐색

기초 통계량

- na.rm=T -> 결측 대체하고 계산

연별 미세먼지 평균

```
pm10w %>%
  index_by(Year=year(yymm))%>%
  summarize(n=n(),my=mean(y, na.rm= T))
```

```
## # A tsibble: 11 x 3 [1Y]
##     Year     n    my
##    <dbl> <int> <dbl>
##  1  2010    96  48.6
##  2  2011    96  47.5
##  3  2012    96  42.4
##  4  2013    96  45.5
##  5  2014    96  45.2
##  6  2015    96  45.7
##  7  2016    96  44.7
##  8  2017    96  43.8
##  9  2018    96  40.9
## 10  2019    96  40.5
## 11  2020    24  38.1
```

분기별 미세먼지 평균

```
pm10w %>%
  index_by(Quarter=quarter(yymm))%>%
  summarize(n=n(),my=mean(y, na.rm= T))
```

```
## # A tsibble: 4 x 3 [1]
##   Quarter     n    my
##     <int> <int> <dbl>
## 1       1   264  51.7
## 2       2   240  49.8
## 3       3   240  31.5
## 4       4   240  43.1
```

월별 미세먼지 평균

```
pm10w %>%
  index_by(Month=month(yymm))%>%
  summarize(n=n(),my=mean(y, na.rm= T))
```

```
## # A tsibble: 12 x 3 [1]
##    Month     n    my
##    <dbl> <int> <dbl>
##  1     1    88  49.9
##  2     2    88  50.3
##  3     3    88  54.9
##  4     4    80  52.0
##  5     5    80  55.8
##  6     6    80  41.6
##  7     7    80  32.8
##  8     8    80  30.6
##  9     9    80  31.0
## 10    10    80  37.1
## 11    11    80  47.2
## 12    12    80  45.1
```

도시별 연평균 미세먼지 측정량

```
yyfd <- pm10w %>%
  index_by(Year=year(yymm))%>%
  group_by(city)%>%
  summarize(n=n(),my=mean(y, na.rm= T))
yyfd
```

```
## # A tsibble: 88 x 4 [1Y]
## # Key:       city [8]
##    city   Year     n    my
##    <chr> <dbl> <int> <dbl>
##  1 busan  2010    12  48.7
##  2 busan  2011    12  47.6
##  3 busan  2012    12  43.4
##  4 busan  2013    12  48.5
##  5 busan  2014    12  48.4
##  6 busan  2015    12  45.1
##  7 busan  2016    12  43.8
##  8 busan  2017    12  43.8
##  9 busan  2018    12  41.6
## 10 busan  2019    12  36.9
## # ... with 78 more rows
```

도시별 월평균 미세먼지 측정량

```
mmfd <- pm10w %>%
  index_by(Month=month(yymm))%>%
  group_by(city)%>%
  summarize(n=n(),my=mean(y, na.rm= T))
mmfd
```

```
## # A tsibble: 96 x 4 [1]
## # Key:       city [8]
##    city  Month     n    my
##    <chr> <dbl> <int> <dbl>
##  1 busan     1    11  45.9
##  2 busan     2    11  47.6
##  3 busan     3    11  52.1
##  4 busan     4    10  52.6
##  5 busan     5    10  58.1
##  6 busan     6    10  44
##  7 busan     7    10  38.3
##  8 busan     8    10  35.5
##  9 busan     9    10  33.2
## 10 busan    10    10  37.4
## # ... with 86 more rows
```

도시별 분기별 미세먼지 측정량

```
qqfd <- pm10w %>%
  index_by(quarter=quarter(yymm))%>%
  group_by(city)%>%
  summarize(n=n(),my=mean(y, na.rm= T))
qqfd
```

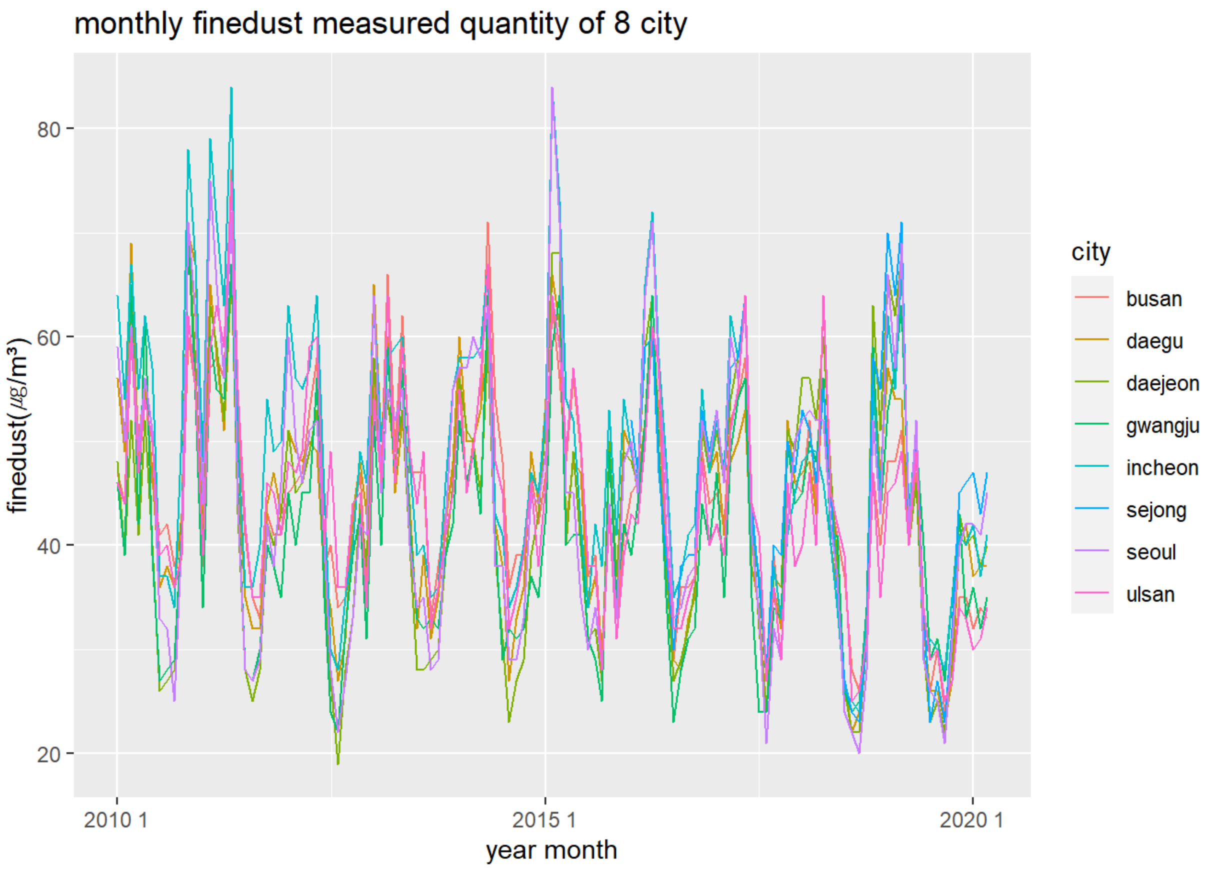
```
## # A tsibble: 32 x 4 [1]
## # Key:       city [8]
##    city    quarter     n    my
##    <chr>     <int> <int> <dbl>
##  1 busan         1    33  48.5
##  2 busan         2    30  51.6
##  3 busan         3    30  35.7
##  4 busan         4    30  41.8
##  5 daegu         1    33  51.6
##  6 daegu         2    30  47.3
##  7 daegu         3    30  31.1
##  8 daegu         4    30  44.7
##  9 daejeon       1    33  52.4
## 10 daejeon       2    30  47.4
## # ... with 22 more rows
```

시계열 그림

- 추세가 없고 등분산이지만 계절성이 있어 비정상 시계열로보인다.

```
pm10w %>%
  autoplot(y) +
  ylab("finedust(㎍/㎥ )") +
  labs(title="monthly finedust measured quantity of 8 city")+
  xlab("year month")
```

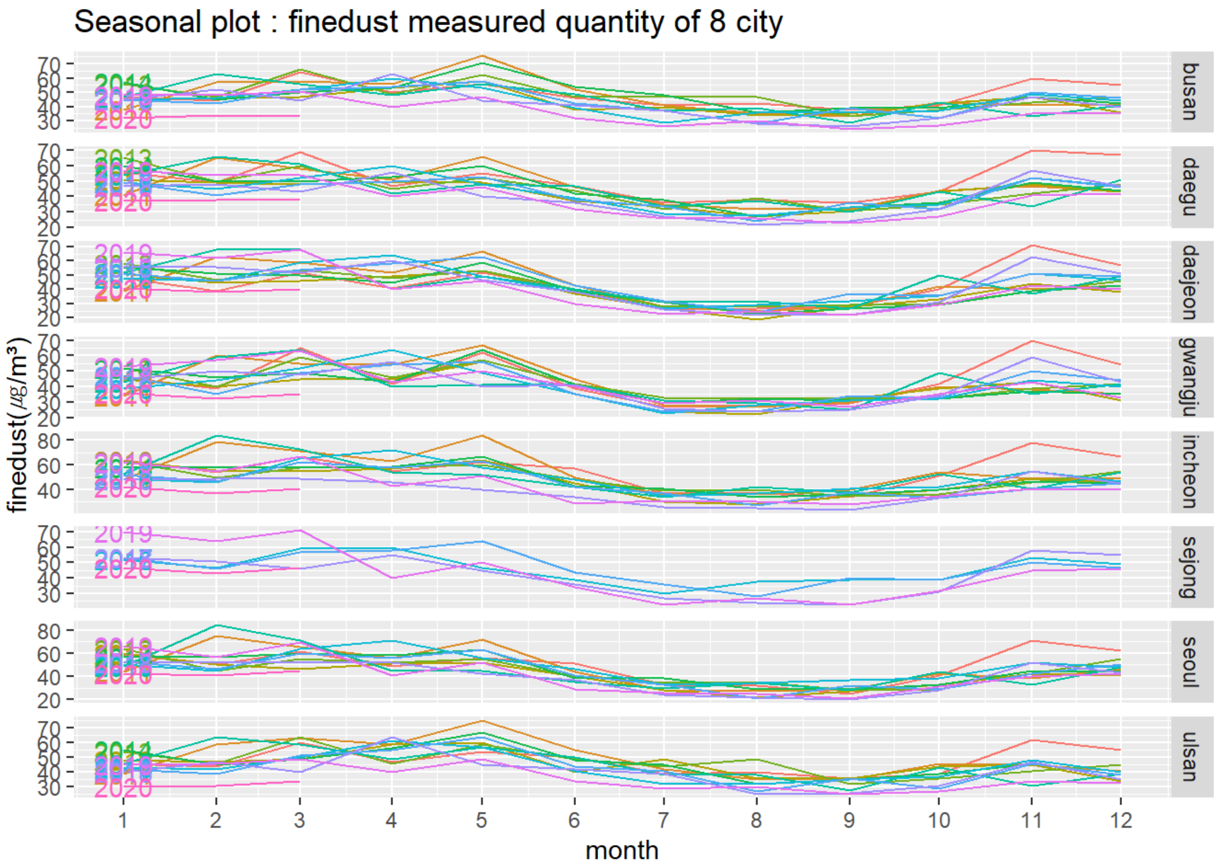
```
## Warning: Removed 72 row(s) containing missing values (geom_path).
```



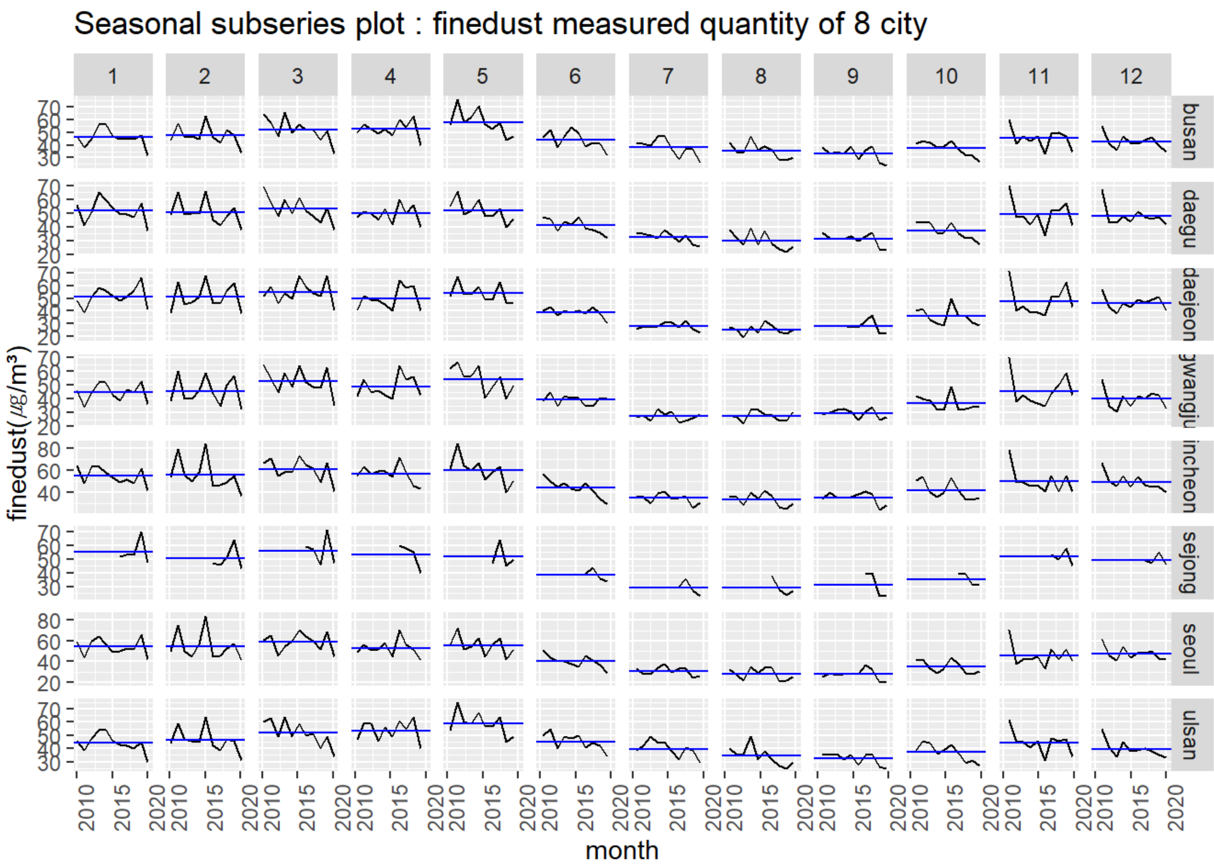
계절성 그림 (gg_series, gg_subseries)

```
pm10w %>% gg_season(y, labels = "left")+
  ylab("finedust(㎍/m³ )")+
  xlab("month")+
  ggtitle("Seasonal plot : finedust measured quantity of 8 city")
```

```
## Warning: Removed 6 rows containing missing values (geom_text).
```




```
pm10w %>%
  gg_subseries(y) +
  ylab("finedust(㎍/m³ )") +
  xlab("month")+
  ggtitle("Seasonal subseries plot : finedust measured quantity of 8 city")
```



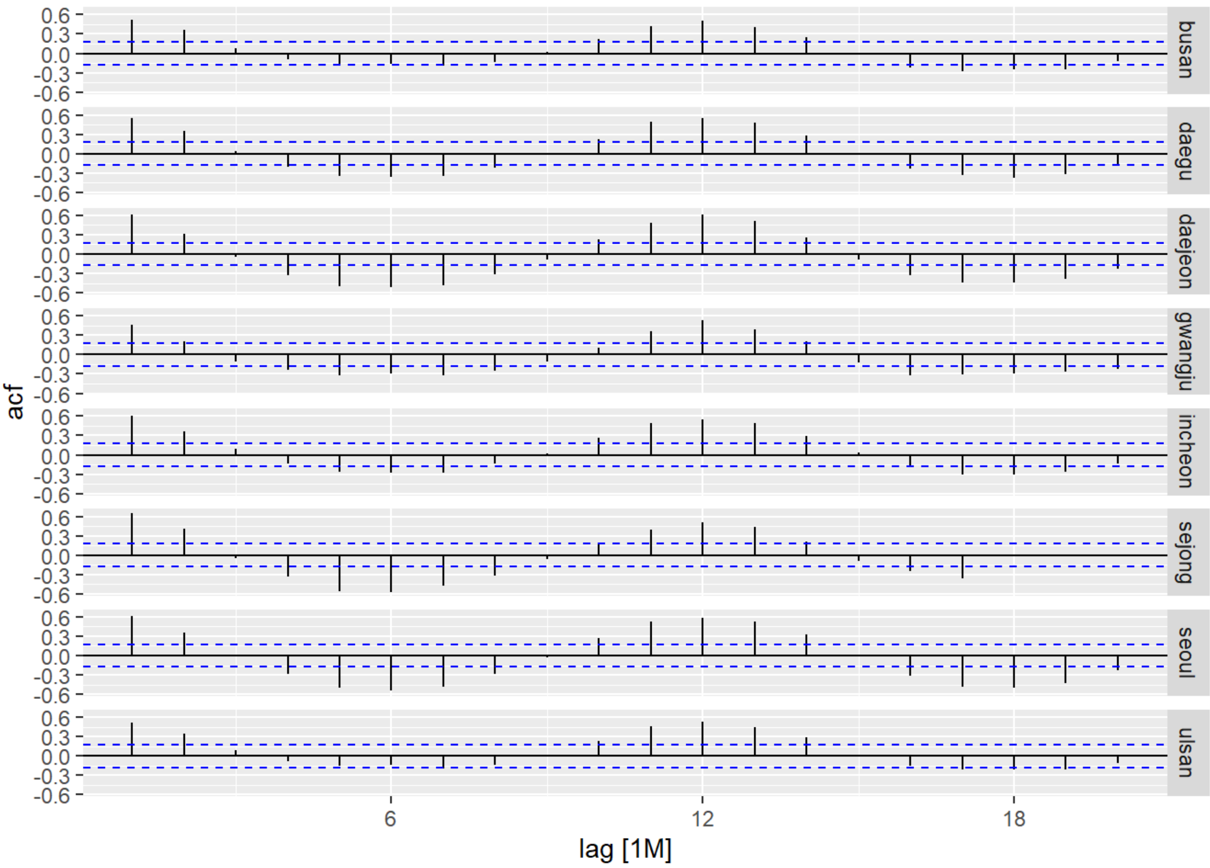
ACF의 특징 기술

- 계절성이있는 비정상 시계열의 acf모양인 scalloped pattern을 보인다.

```
pm10w %>% ACF(y, lag_max=12)
```

```
## # A tsibble: 96 x 3 [1M]
## # Key:       city [8]
##    city    lag     acf
##    <chr> <lag>   <dbl>
##  1 busan    1M  0.517
##  2 busan    2M  0.369
##  3 busan    3M  0.0834
##  4 busan    4M -0.0903
##  5 busan    5M -0.170
##  6 busan    6M -0.169
##  7 busan    7M -0.197
##  8 busan    8M -0.128
##  9 busan    9M  0.0274
## 10 busan   10M  0.220
## # ... with 86 more rows
```

```
autoplot(ACF(pm10w,y,type='cor'))
```



Ljung-Box 검정

- p-value가 $\alpha = 0.05$보다 작으므로 $H_0 = \rho_1 = \ldots = \rho_{12} = 0$를 기각한다. 따라서 y를 백색잡음으로 보기 어렵다.

```
pm10w %>% features(y, ljung_box,lag=12, dof=0)
```

```
## # A tibble: 8 x 3
##   city     lb_stat lb_pvalue
##   <chr>      <dbl>     <dbl>
## 1 busan       134.         0
## 2 daegu       195.         0
## 3 daejeon     275.         0
## 4 gwangju     149.         0
## 5 incheon     184.         0
## 6 sejong      127.         0
## 7 seoul       281.         0
## 8 ulsan       139.         0
```

서울 미세먼지

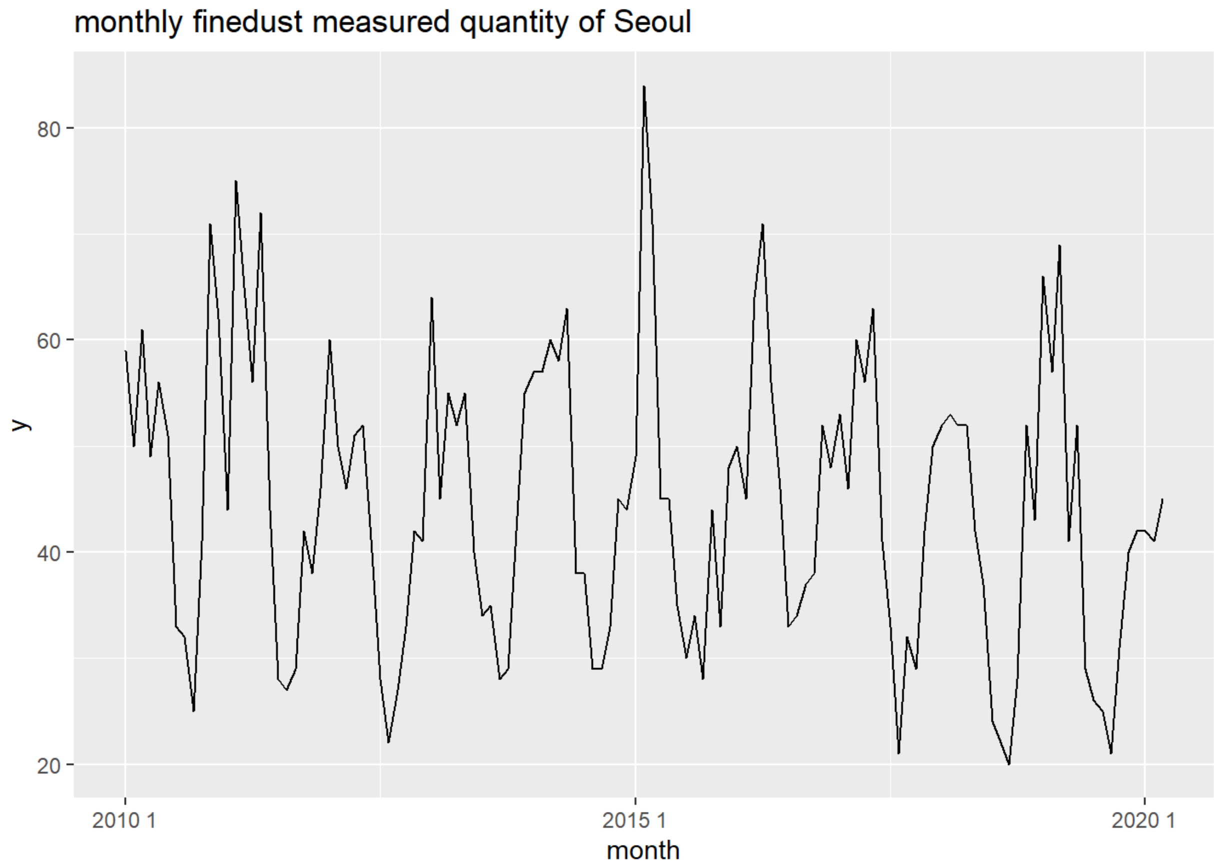
```
pm10s <- pm10w %>%
          filter(city=='seoul') %>%
          select(-city)
pm10s
```

```
## # A tsibble: 123 x 2 [1M]
##        yymm     y
##       <mth> <dbl>
##  1  2010 1    59
##  2  2010 2    50
##  3  2010 3    61
##  4  2010 4    49
##  5  2010 5    56
##  6  2010 6    51
##  7  2010 7    33
##  8  2010 8    32
##  9  2010 9    25
## 10 2010 10    41
## # ... with 113 more rows
```

시계열 그림

```
autoplot(pm10s)+
  labs(title="monthly finedust measured quantity of Seoul")+
  xlab("month")
```

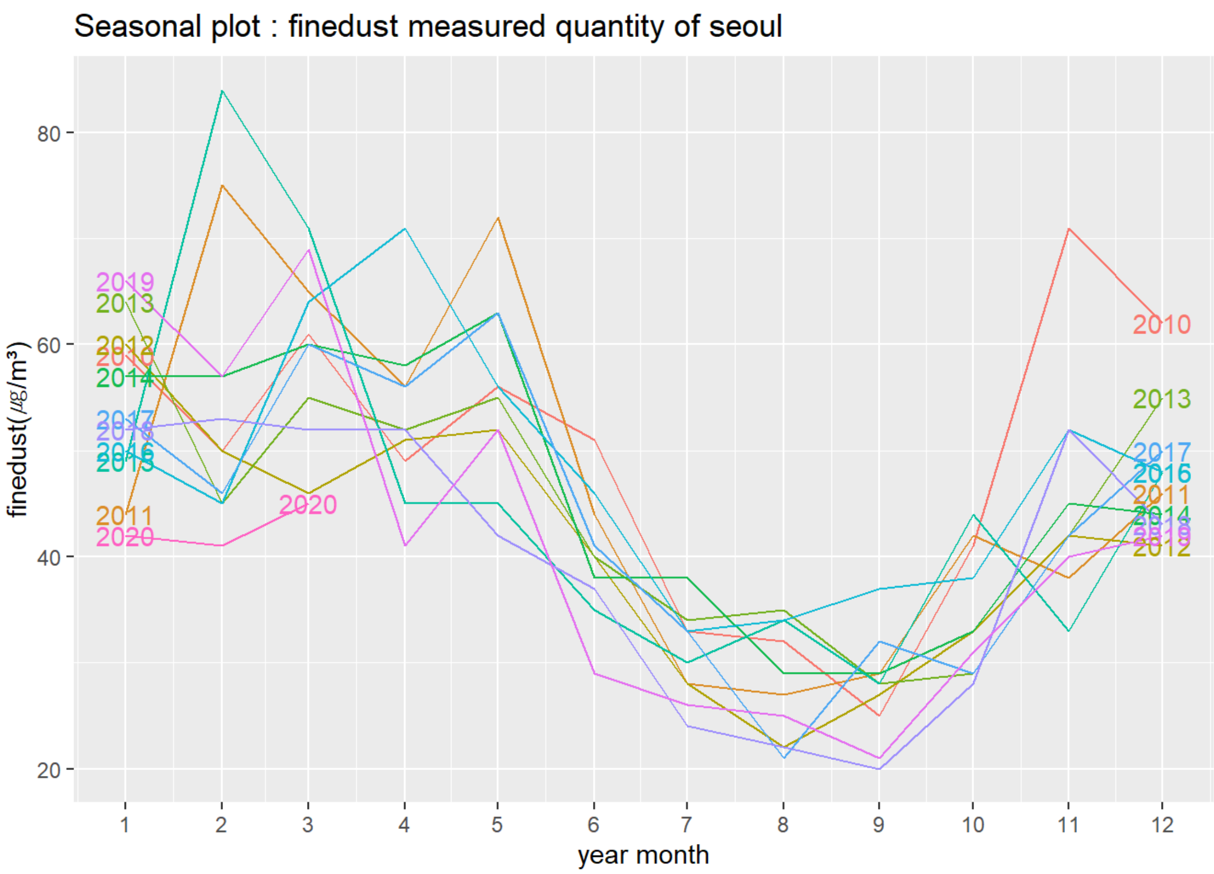
```
## Plot variable not specified, automatically selected `.vars = y`
```



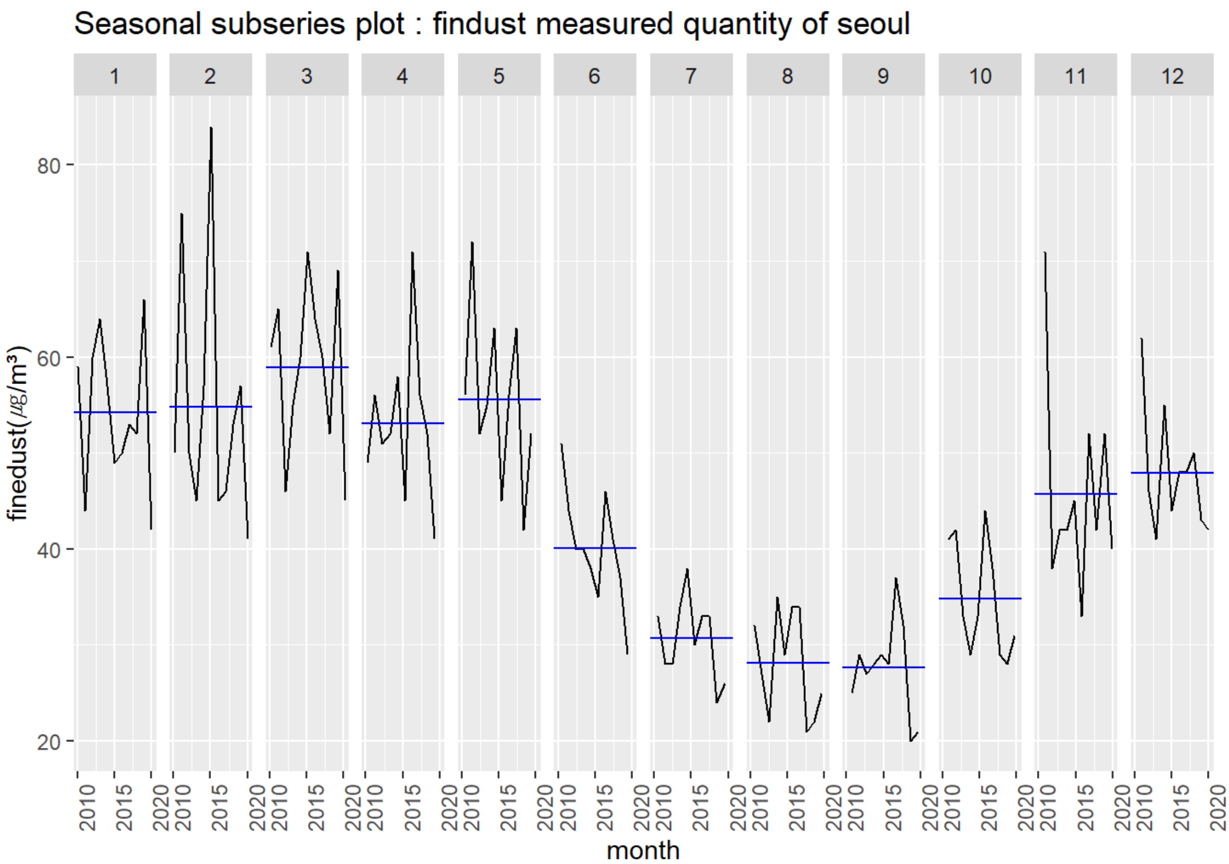
계절성 검토

- 서울의 미세먼지 측정량은 2월~5월 , 10월~12월에 증가하고, 7~9월에 감소하는 계절성을 보인다.

```
pm10s %>% gg_season(y, labels = "both")+
  ylab("finedust(㎍/㎥ )")+
  xlab("year month")+
  ggtitle("Seasonal plot : finedust measured quantity of seoul")
```




```
pm10s %>%
  gg_subseries(y) +
  ylab("finedust(㎍/m³ )") +
  xlab("month")+
  ggtitle("Seasonal subseries plot : findust measured quantity of seoul")
```



시계열 그림, 계절성 검토. 추세여부, 등분산성 등을 설명하시오

- 서울의 미세먼지 측정량은 2월~5월 , 10월~12월에 증가하고, 7~9월에 감소하는 계절성을 보인다.
- 추세는 존재하지않는 것으로 보이며 등분산성을 가지는 것으로 보인다.

자료 분할

- TRN(적합용) : 2010.1~2017.12 월별 미세먼지 측정량
- TST(검정용) : 2018.1~2019.12ㅎ 월별 미세먼지 측정량

```
TRN <- filter_index(pm10s, ~'2017 12')
TST <- filter_index(pm10s,'2018 1'~'2020 1')
```

TRN를 X11, SEATS, STL로 분해하고 설명하시오

x11 decomposition

- 분해결과 미세먼지 측정량은 몇년도든 간 3월에 가장 높다.
- 뚜렷한 계절성을 갖고, 등분산이며 결정적 추세는 존재하지않는 것으로 보인다.

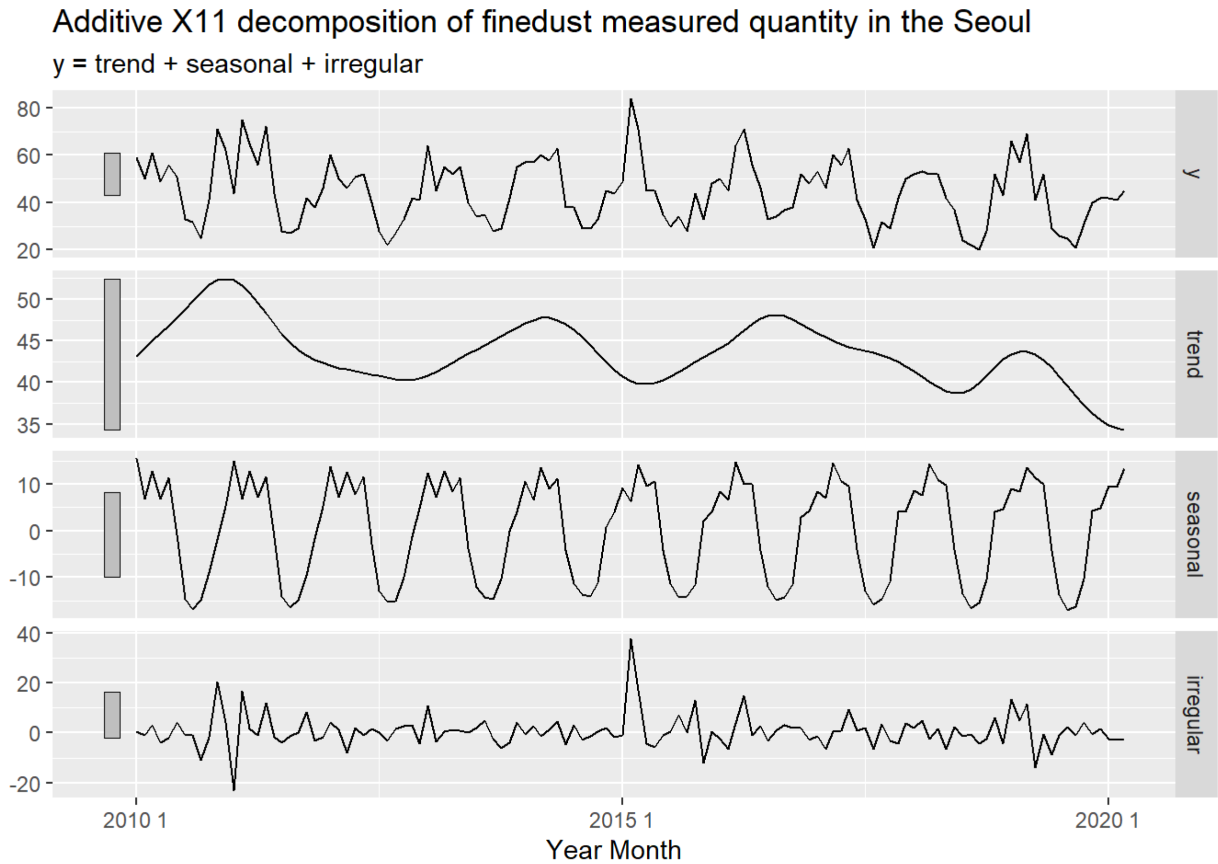
```
library(seasonal)
```

```
## Warning: package 'seasonal' was built under R version 4.0.3
```

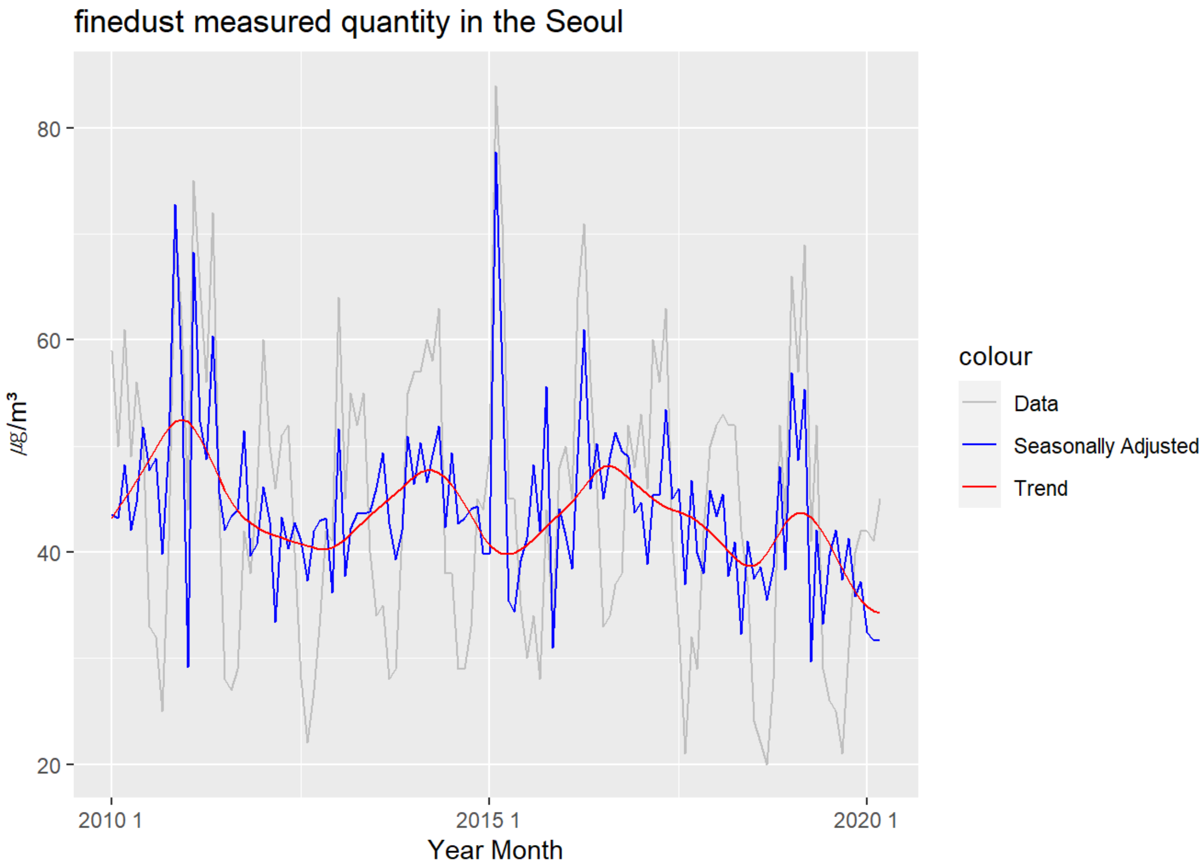
```
##
## Attaching package: 'seasonal'
```

```
## The following object is masked from 'package:tibble':
##
##     view
```

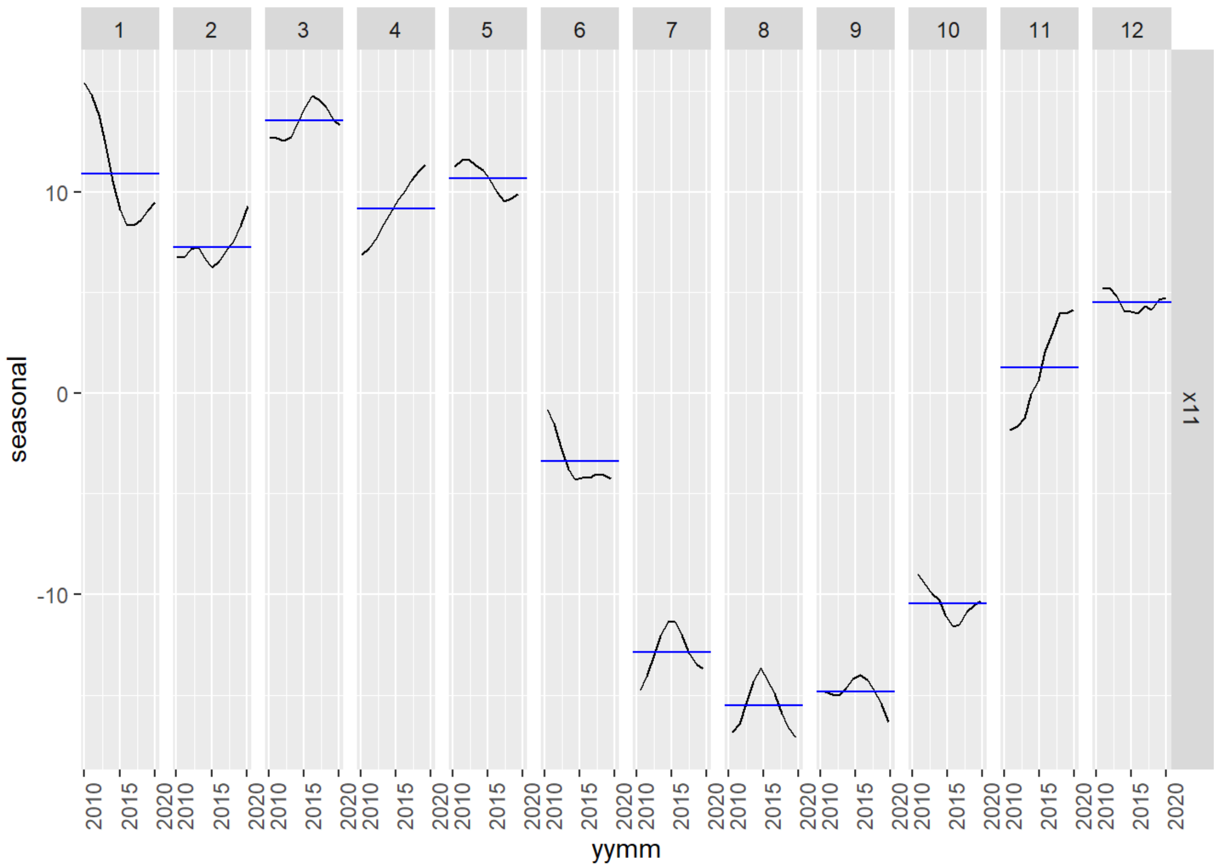
```
x11_dcmp <- pm10s %>%
  model(x11 = feasts:::X11(y, type = "additive")) %>%
  components()
autoplot(x11_dcmp) + xlab("Year Month") +
  ggtitle("Additive X11 decomposition of finedust measured quantity in the Seoul")
```




```
x11_dcmp %>%
  ggplot(aes(x = yymm)) +
  geom_line(aes(y = y, colour = "Data")) +
  geom_line(aes(y = season_adjust, colour = "Seasonally Adjusted")) +
  geom_line(aes(y = trend, colour = "Trend")) +
  xlab("Year Month") + ylab("㎍/㎥ ") +
  ggtitle("finedust measured quantity in the Seoul") +
  scale_colour_manual(values=c("gray","blue","red"),
             breaks=c("Data","Seasonally Adjusted","Trend"))
```



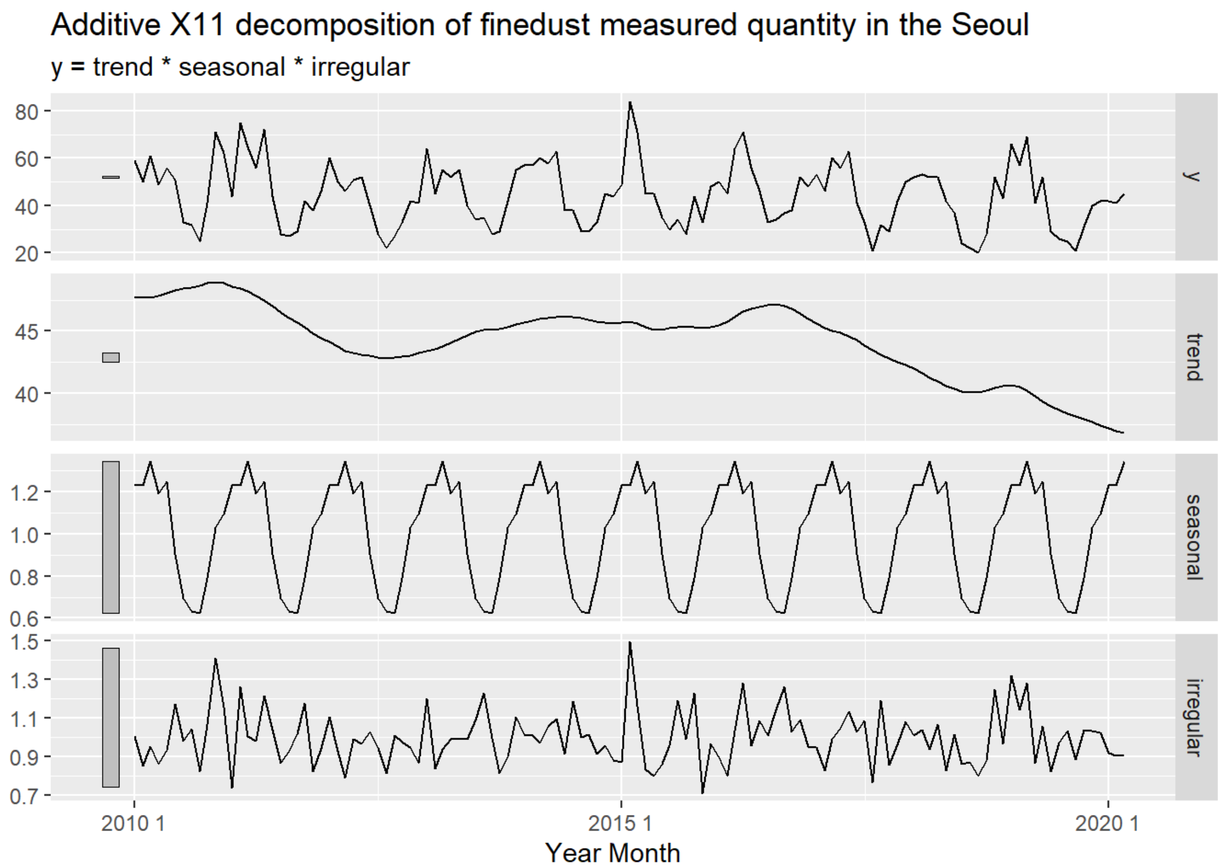

```
x11_dcmp %>%
  gg_subseries(seasonal)
```



SEATS decomposition

- 분해결과 서울시 미세먼지 측정량은 감소하는 추세이며 등분산이고 뚜렷한 계절성을 가지는 것으로 보인다.

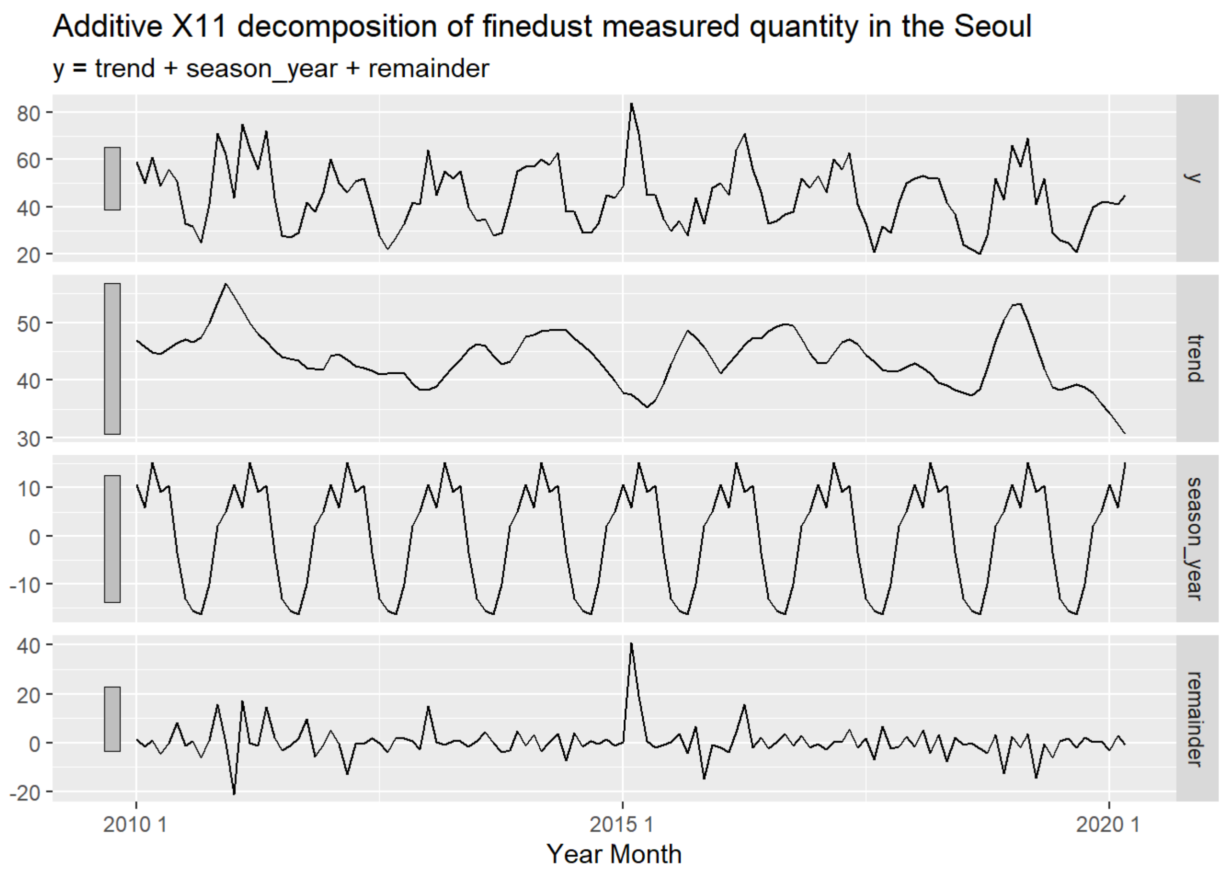
```
seats_dcmp <- pm10s %>%
  model(seats = feasts:::SEATS(y)) %>%
  components()
autoplot(seats_dcmp)+ xlab("Year Month") +
  ggtitle("Additive X11 decomposition of finedust measured quantity in the Seoul")
```



STL decomposition

- 분해결과 결정적 추세는 존재하지 않으며 등분산이며 뚜렷한 계절성을 가지는 것으로보인다.

```
pm10s %>%
  model(STL(y ~ trend(window=7) + season(window='periodic'),
    robust = TRUE)) %>%
  components() %>%
  autoplot()+ xlab("Year Month") +
  ggtitle("Additive X11 decomposition of finedust measured quantity in the Seoul")
```



단순예측법 실행

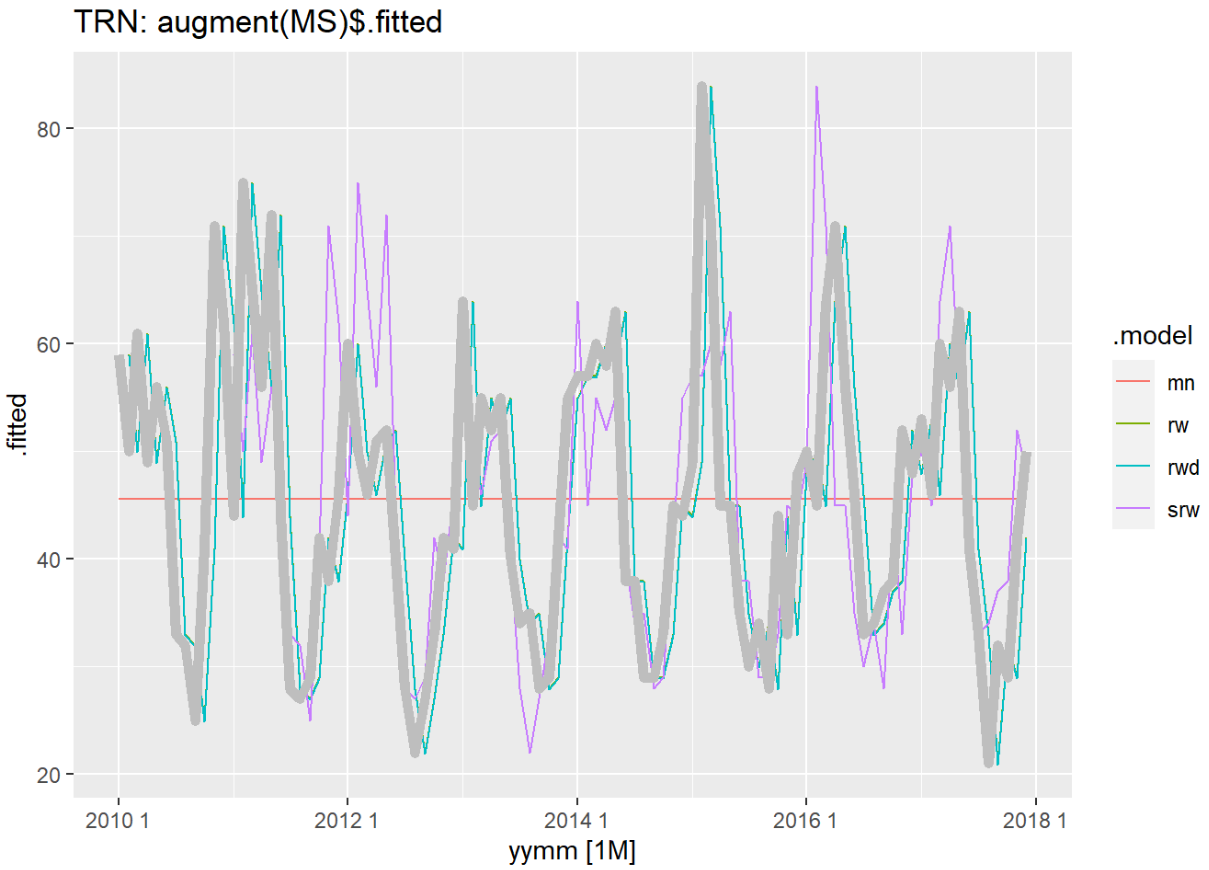
MBL 생성

```
MS <- model(TRN,
      mn = MEAN(y),
      rw = NAIVE(y),
      rwd = RW(y~drift()),
      srw = SNAIVE(y))
MS
```

```
## # A mable: 1 x 4
##         mn      rw          rwd       srw
##    <model> <model>      <model>   <model>
## 1   <MEAN> <NAIVE> <RW w/ drift> <SNAIVE>
```

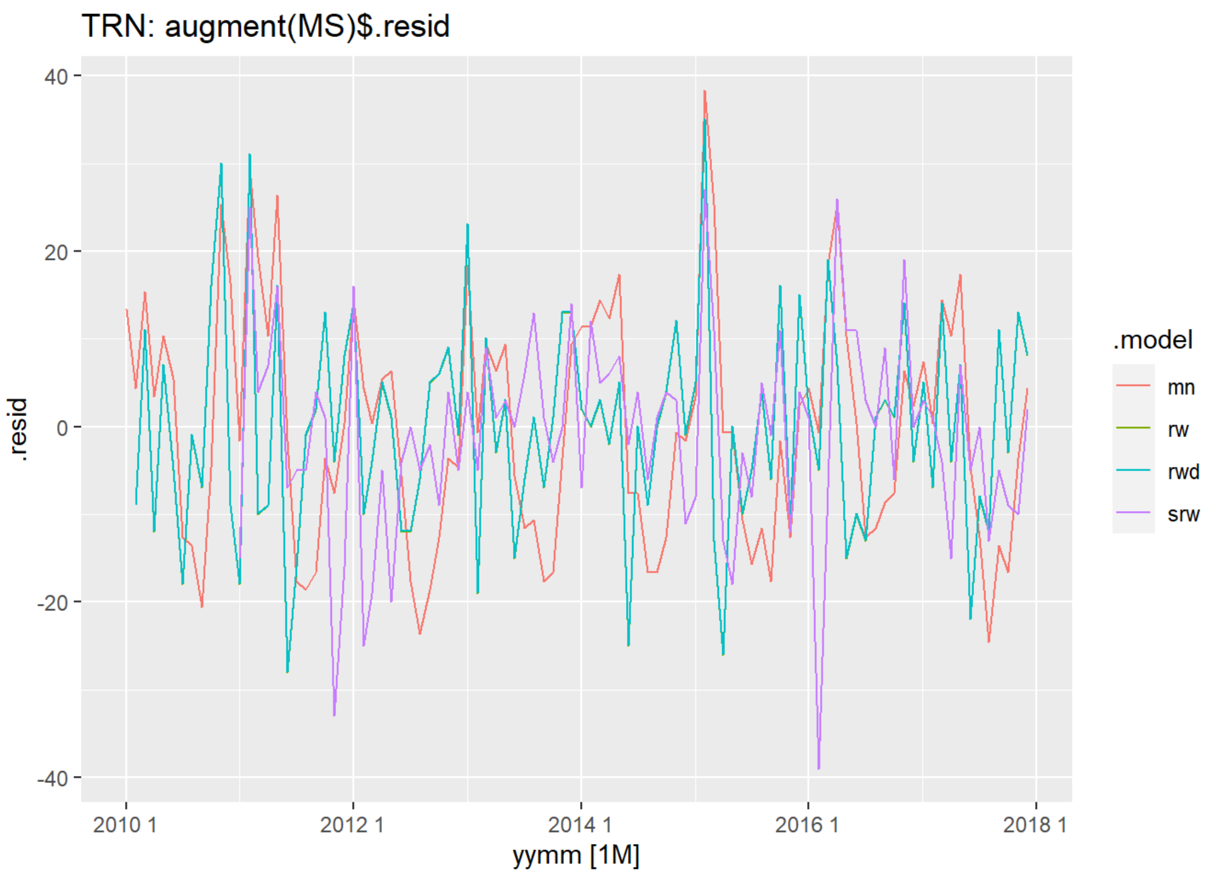
```
AS <- augment(MS)
autoplot(AS, .fitted)+
  autolayer(AS,y,color='gray',size=2)+
  ggtitle('TRN: augment(MS)$.fitted')
```

```
## Warning: Removed 14 row(s) containing missing values (geom_path).
```




```
autoplot(AS,.resid)+
  ggtitle('TRN: augment(MS)$.resid')
```

```
## Warning: Removed 14 row(s) containing missing values (geom_path).
```



```
features(AS,.resid,ljung_box, lag=4, dof=0)
```

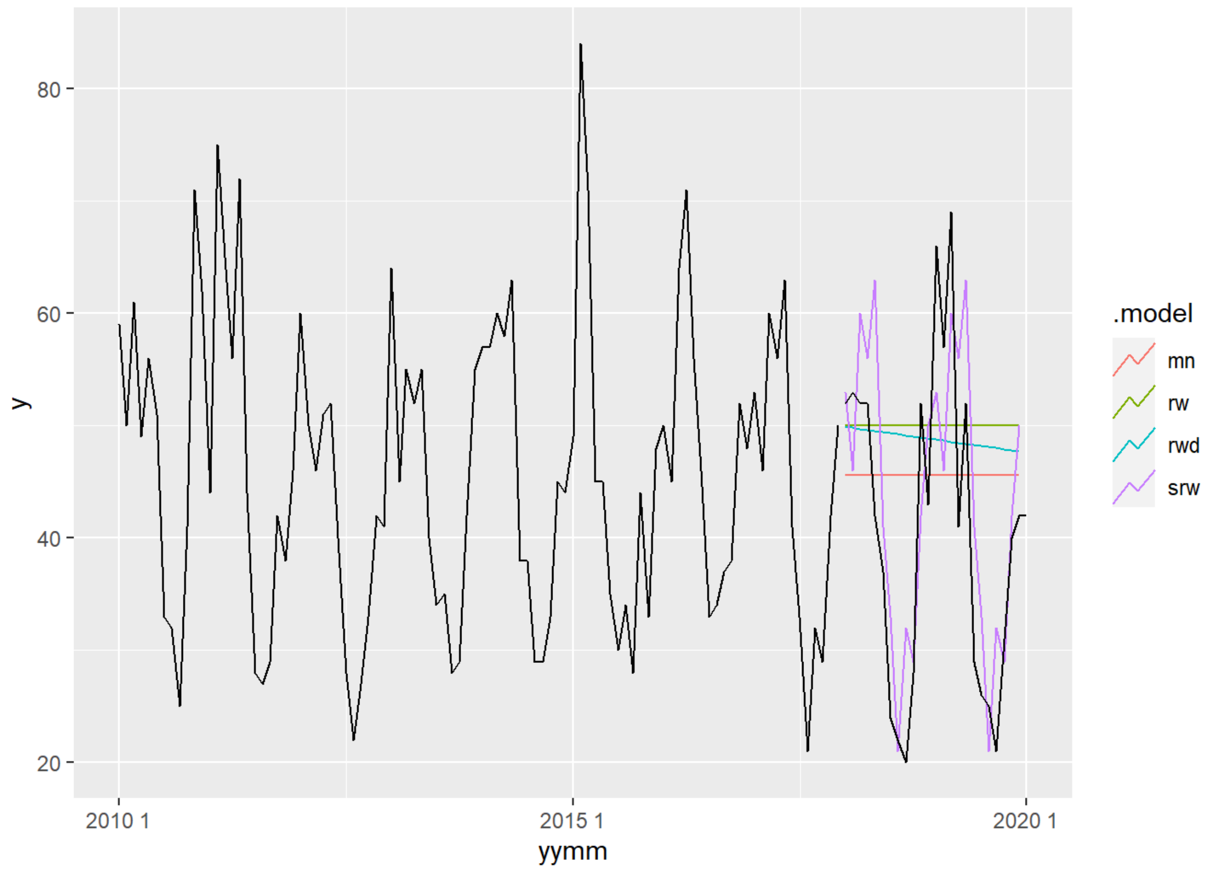
```
## # A tibble: 4 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 mn       51.6   1.65e-10
## 2 rw        4.64  3.27e- 1
## 3 rwd       4.64  3.27e- 1
## 4 srw       2.84  5.85e- 1
```

FBL생성

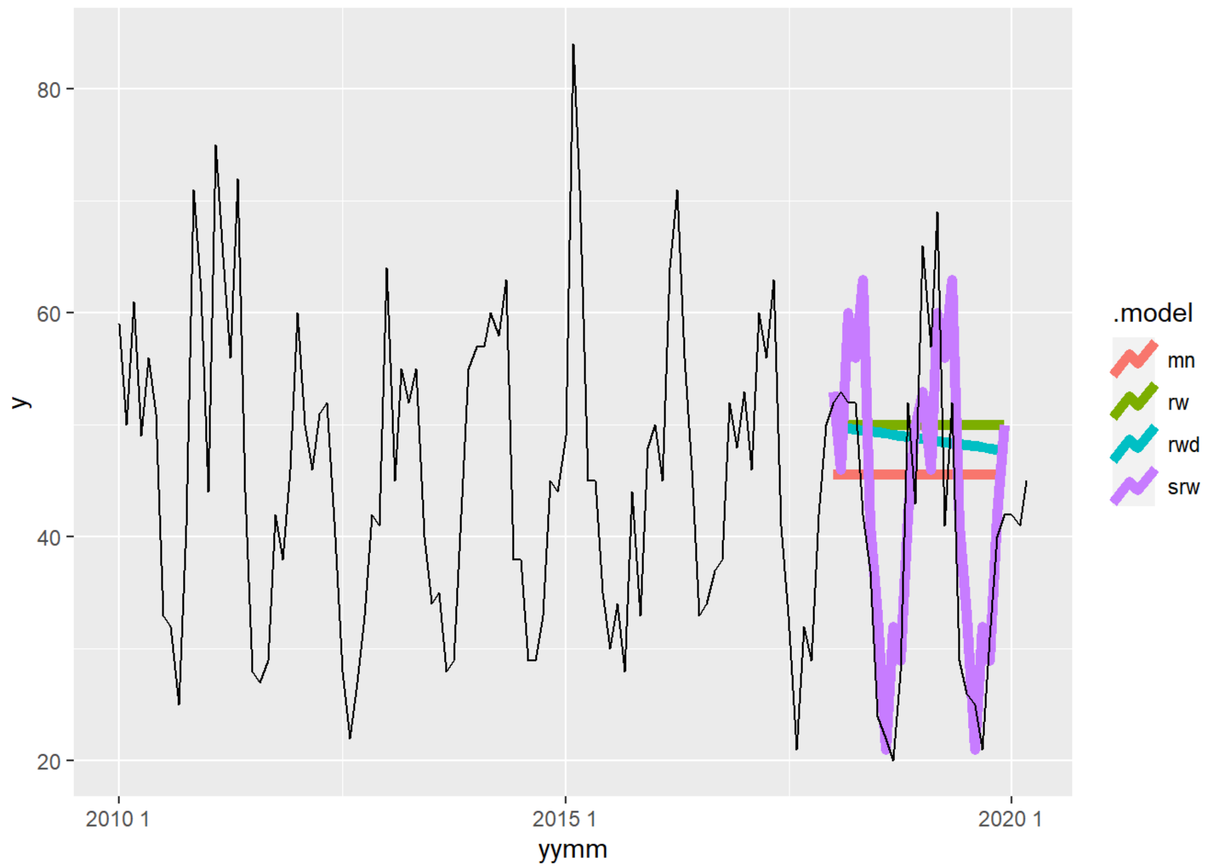
```
FS <- forecast(MS, data=pm10s)
FS
```

```
## # A fable: 96 x 4 [1M]
## # Key:     .model [4]
##    .model    yymm          y .mean
##    <chr>    <mth>     <dist> <dbl>
##  1 mn      2018 1 N(46, 181)  45.6
##  2 mn      2018 2 N(46, 181)  45.6
##  3 mn      2018 3 N(46, 181)  45.6
##  4 mn      2018 4 N(46, 181)  45.6
##  5 mn      2018 5 N(46, 181)  45.6
##  6 mn      2018 6 N(46, 181)  45.6
##  7 mn      2018 7 N(46, 181)  45.6
##  8 mn      2018 8 N(46, 181)  45.6
##  9 mn      2018 9 N(46, 181)  45.6
## 10 mn     2018 10 N(46, 181)  45.6
## # ... with 86 more rows
```

```
autoplot(FS,TRN, level=NULL)+
  autolayer(TST,y)
```




```
autoplot(FS,pm10s,level=NULL,size=2)
```



성능 평가

TRN에 대한 성능평가

- TRN에서의 성능은 rwd의 RMSE,MAE,MAPE가 각각 06916,9.445097,21.17628로 rwd 모델이 가장 우수한 것으로 나타났다.

```
as.data.frame(accuracy(MS))
```

```
##   .model    .type            ME     RMSE       MAE       MPE     MAPE     MASE
## 1     mn Training -2.368688e-15 13.30051 10.882161 -9.328597 26.79283 1.291104
## 2     rw Training -9.473684e-02 12.06954  9.442105 -3.504211 21.19058 1.120250
## 3    rwd Training -1.498984e-16 12.06916  9.445097 -3.276447 21.17628 1.120605
## 4    srw Training -7.619048e-01 11.44552  8.428571 -4.414184 18.61234 1.000000
##          ACF1
## 1  0.58671564
## 2 -0.12066405
## 3 -0.12066405
## 4  0.06814661
```

TST에 대한 성능평가

- TST에서의 성능은 SRW의 RMSE,MAE,MAPE가 각각 9.313968, 7.916667,21.54606로 가장 우수한것으로 나타났다.

```
as.data.frame(accuracy(FS, data=pm10s))
```

```
##   .model .type        ME      RMSE       MAE       MPE     MAPE      MASE
## 1     mn  Test -4.947917 15.009578 12.820312 -28.32122 41.66175 1.5210540
## 2     rw  Test -9.333333 16.968107 13.916667 -40.65811 48.07065 1.6511299
## 3    rwd  Test -8.149123 16.225114 13.364035 -37.18026 45.69019 1.5855635
## 4    srw  Test -3.166667  9.313968  7.916667 -12.25656 21.54606 0.9392655
##        ACF1
## 1 0.6312537
## 2 0.6312537
## 3 0.6260702
## 4 0.0960204
```

최종모형 -SNAIVE

- TST에서 성능이 가장 좋은 snaive모델을 최종모형으로 선택

```
MSRW <- model(TRN,
      srw = SNAIVE(y))
MSRW
```

```
## # A mable: 1 x 1
##         srw
##     <model>
## 1 <SNAIVE>
```

```
ASRW <- augment(MSRW)
ASRW
```

```
## # A tsibble: 96 x 6 [1M]
## # Key:       .model [1]
##    .model     yymm     y .fitted .resid .innov
##    <chr>     <mth> <dbl>   <dbl>  <dbl>  <dbl>
##  1 srw      2010 1    59      NA     NA     NA
##  2 srw      2010 2    50      NA     NA     NA
##  3 srw      2010 3    61      NA     NA     NA
##  4 srw      2010 4    49      NA     NA     NA
##  5 srw      2010 5    56      NA     NA     NA
##  6 srw      2010 6    51      NA     NA     NA
##  7 srw      2010 7    33      NA     NA     NA
##  8 srw      2010 8    32      NA     NA     NA
##  9 srw      2010 9    25      NA     NA     NA
## 10 srw    2010 10    41      NA     NA     NA
## # ... with 86 more rows
```

예측값 생성

```
FSRW <- forecast(MSRW, data=pm10s)
```

모형평가

- TRN평가

```
accuracy(MSRW)
```

```
## # A tibble: 1 x 9
##   .model .type        ME  RMSE   MAE   MPE  MAPE  MASE   ACF1
##   <chr>  <chr>     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 srw    Training -0.762  11.4  8.43 -4.41  18.6     1 0.0681
```

- TST평가

```
accuracy(FSRW, pm10s)
```

```
## # A tibble: 1 x 9
##   .model .type    ME  RMSE   MAE   MPE  MAPE  MASE   ACF1
##   <chr>  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 srw    Test  -3.17  9.31  7.92 -12.3  21.5 0.939 0.0960
```

잔차 검토

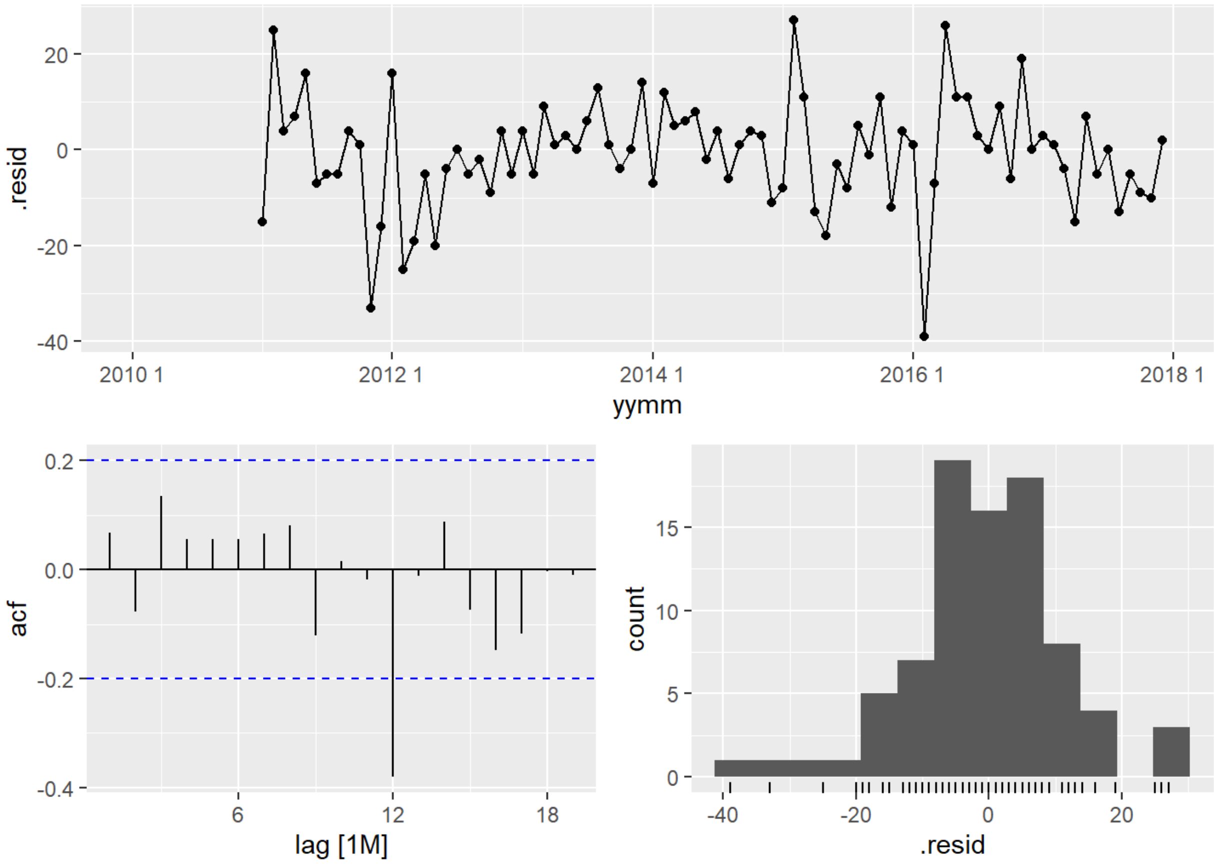
- 잔차는 등분산에 가깝고, 잔차의 ACF에서 잔차의 자기상관이 없고, 정규분포를 따르는 것으로 보인다.

```
MSRW %>%
  gg_tsresiduals()
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 12 rows containing missing values (geom_point).
```

```
## Warning: Removed 12 rows containing non-finite values (stat_bin).
```



잔차의 백색잡음 검정

- p-value가 $\alpha = 0.05$보다 크다.따라서 $H_0 = \rho_1 = \ldots = \rho_{12} = 0$를 기각할 수 없다.

```
features(ASRW,.resid, ljung_box, lag=12,dof=0)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 srw       20.5    0.0589
```