DTW를 이용한 경부선 역 clustering

방태모

1. 패키지 및 데이터 로딩

```
> library(tidyverse)
> library(lubridate)
> library(mlr) # normalizeFeatures{mlr}
> library(dtw) # for method = "DTW"
> # library(TSclust)
> # dtw 패키지 로딩후, dist(자료, method = "DTW") = diss(자료, "DTWARP", p = 0.05){TSclust} 동일함.
> library(openair) # timePlot{openair}
> library(showtext)
> font_add_google("Nanum Gothic", "nanumgothic")
> showtext_auto()

> # 데이터 로딩
> gb <- as_tibble(read.table("./data/GyungBu.txt", header = TRUE))
> glimpse(gb) # 40개역, 2005년 - 2018년 총 14년치 월별 자료
Observations: 168
Variables: 40
$ 서울 <int> 977197, 853344, 1018056, 990384, 1031283, 970389, 960455,  ...
$ 남영 <int> 444389, 376620, 548903, 528825, 558055, 516260, 473828, 445034, ...
$ 용산 <int> 1512145, 1413353, 1451379, 1407101, 1481852, 1409732, 1497541, ...
...
$ 두정 <int> 31578, 88218, 166193, 166860, 170937, 139381, 106491, 125036, ...
$ 천안 <int> 123877, 330325, 504886, 500142, 504825, 404852, 314218, 379299, ...
$ 합계 <int> 17887556, 16409572, 20771582, 20583827, 21173353, 19363309, ...

> col_na <- function(df, fun){
+   out <- vector("double", length(df))
+   for(i in seq_along(df)) out[i] <- fun(is.na(df[[i]]))
+   names(out) <- names(df)
+   sort(out[out > 0], decreasing = TRUE)
+ }
> col_na(gb, sum) # 결측치 개수
서동탄   당정   광명   진위   지제   세마 오산대
    61     60     23     17     17     11     11
> col_na(gb, mean) # 결측치 비율
    서동탄        당정        광명        진위        지제        세마       오산대
0.36309524 0.35714286 0.13690476 0.10119048 0.10119048 0.06547619 0.06547619
```

```
> gb1 <- map(gb, na.omit) # 결측치 제거한 경부선 역별 수송량 list 객체
> gb1_scale <- as_tibble(normalizeFeatures(gb, method = "standardize")) %>%
+    map(., na.omit) # 결측치 제거후, 표준화시킨 경부선 역별 수송량 list 객체
```

2. DTW distance를 이용하여 계층적 군집 수행

(1) DTW distance 계산

```
> distMat <- dist(gb1, method = "DTW")
> distMat_scale <- dist(gb1_scale, method = "DTW")
```

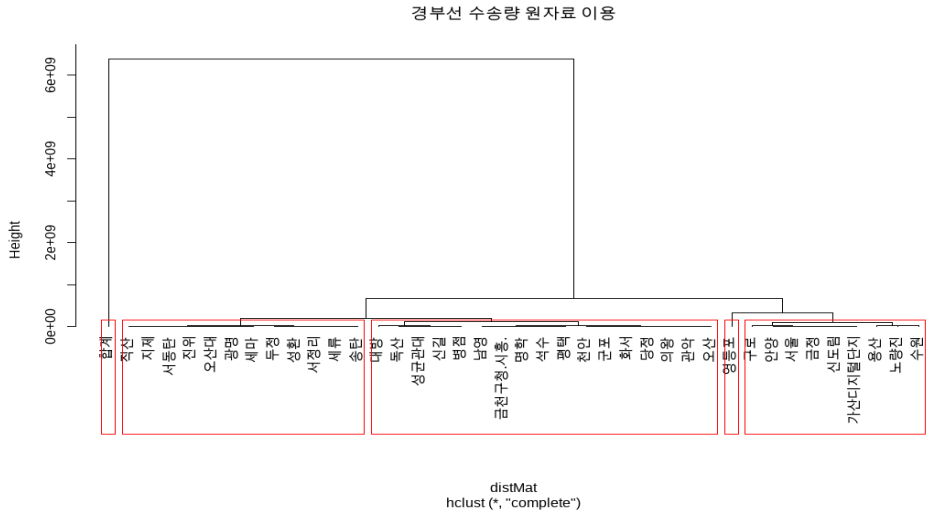
(2) 계층적 군집 수행

군집 간 거리측정 방법으로는 내부 응집성에 중점을 둔 완전연결법(최장연결법)을 사용함.

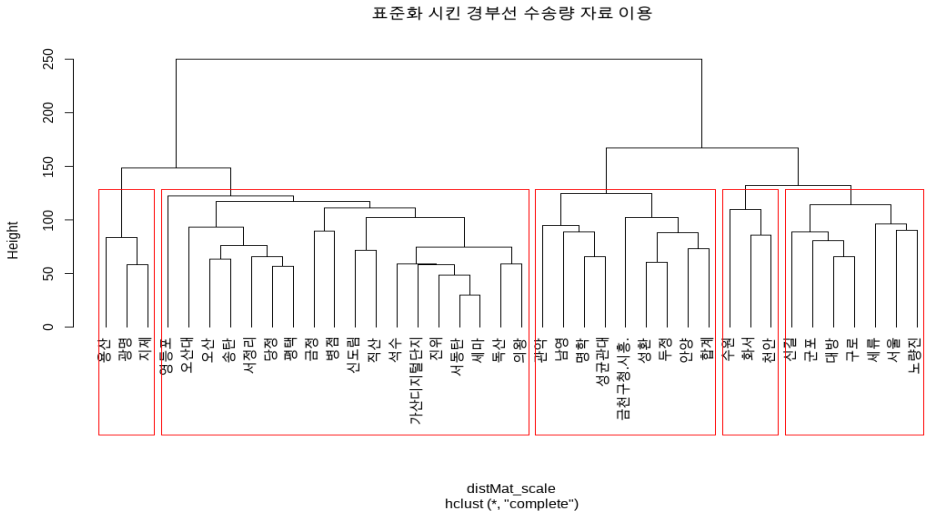
```
> hc <- hclust(distMat, method = "complete")
> hc_scale <- hclust(distMat_scale, method = "complete")
```

(3) 시각화

```
> ## 군집 형성 시각화
> windows()
> plot(hc, hang = -1, main = "경부선 수송량 원자료 이용")
> rect.hclust(hc, k = 5, border = "red")
```

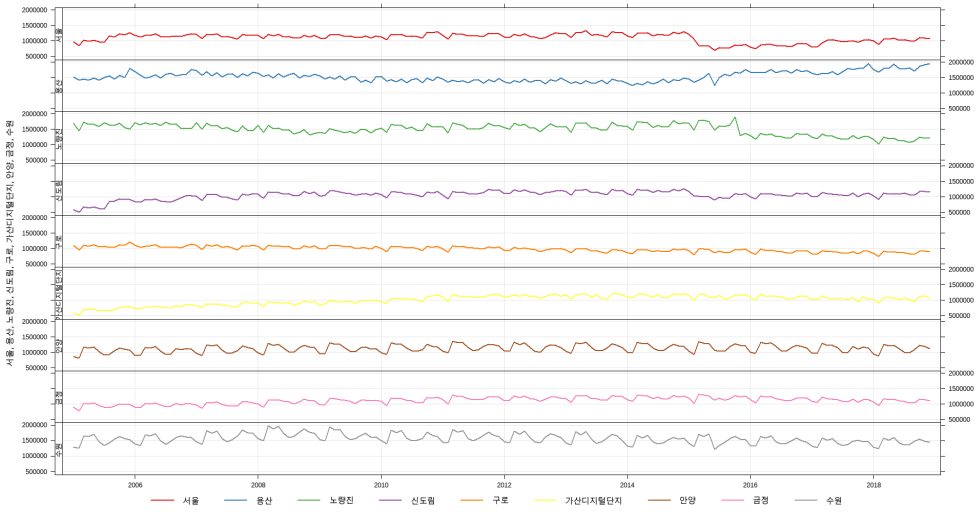



```
> windows()
> plot(hc_scale, hang = -1, main = "표준화 시킨 경부선 수송량 자료 이용")
> rect.hclust(hc_scale, k = 5, border = "red")
```

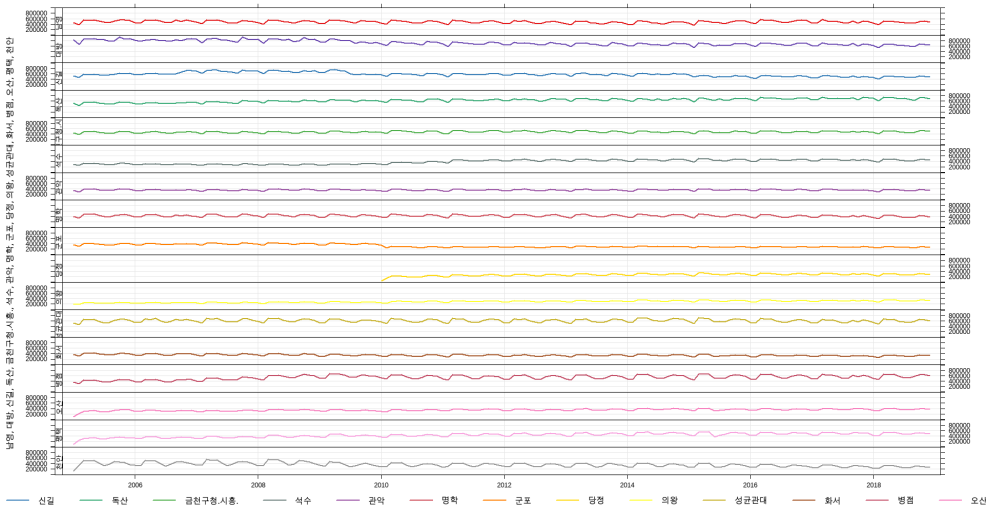



```
> ## 군집별 시도표 시각화
> groups <- cutree(hc, k = 5)
> groups_scale <- cutree(hc_scale, k = 5)
> for(i in 1:5){ # 각 군집의 역 이름 저장
+   assign(str_c("hc_",i) , names(which(groups==i)) )
+   assign(str_c("hc_scale_",i) , names(which(groups_scale==i)))
+ }
> for(i in 1:5){ # 군집별 dataset 만들기
+   assign(str_c("group_", i), gb %>%
+            select(get(str_c("hc_", i))))
+   assign(str_c("group_scale_", i), gb %>%
+            select(get(str_c("hc_scale_", i))))
+ }

> # 원자료로 형성한 군집별 시도표
> windows()
> group_1 %>%
+   mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+   timePlot(., pollutant = colnames(group_1), lwd = 2)
```

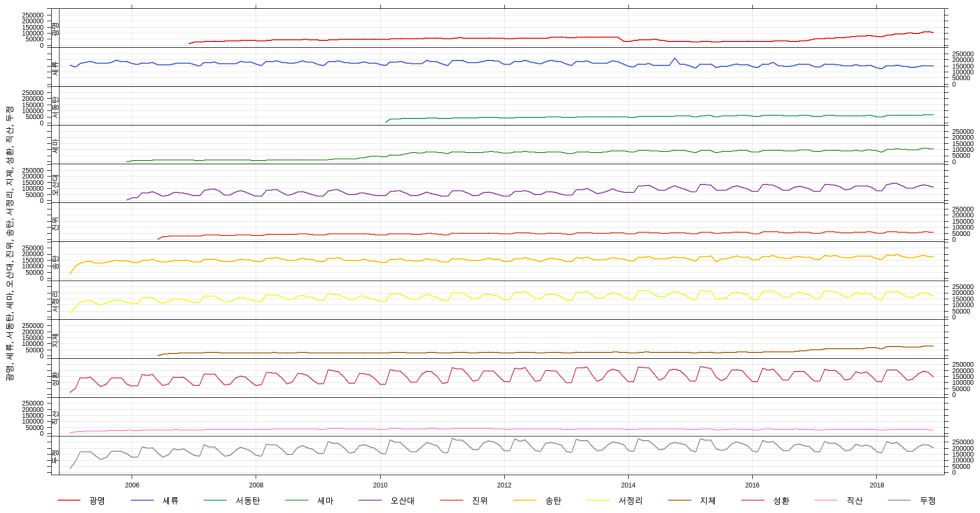



```
> windows()
> group_2 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_2), lwd = 2)
```



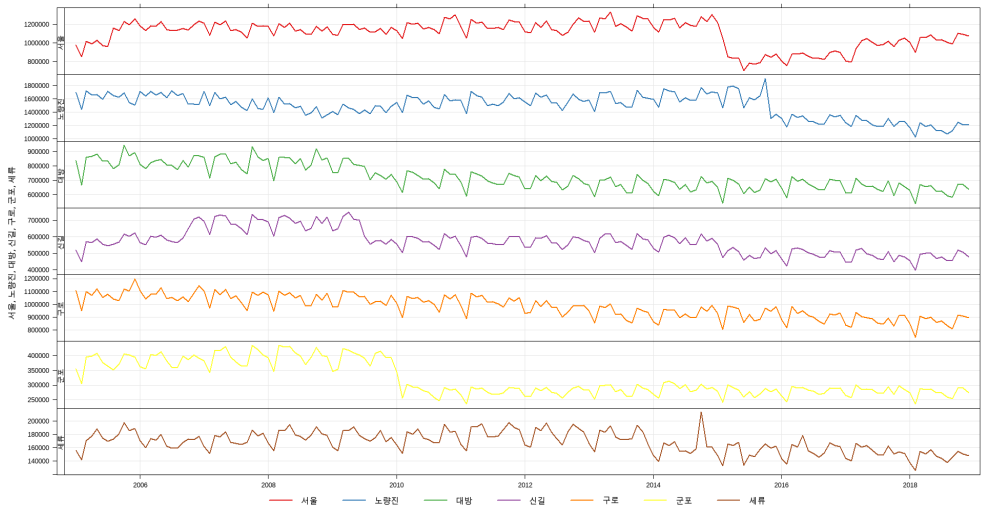

```
> windows()
> group_3 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_3), lwd = 2)
```

```
> windows()
> group_4 %>%
+    mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+    timePlot(., pollutant = colnames(group_4), lwd = 2)
```

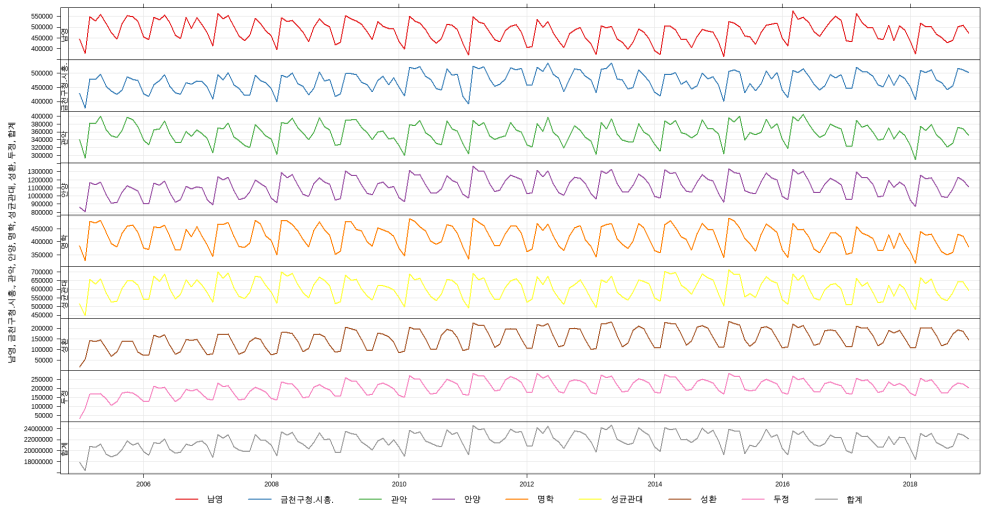



```
> windows()
> group_5 %>%
+    mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+    timePlot(., pollutant = colnames(group_5), lwd = 2)
```

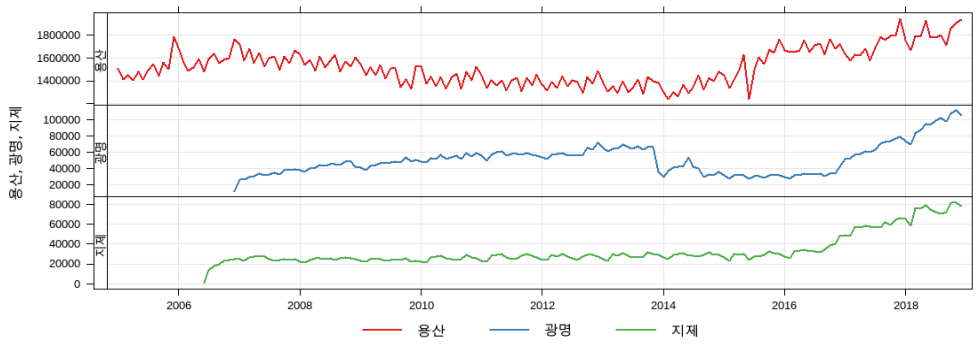

```
> # 표준화시킨 자료로 형성한 군집별 시도표
> windows()
> group_scale_1 %>%
+    mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+    timePlot(., pollutant = colnames(group_scale_1), lwd = 2, y.relation = "free")
```



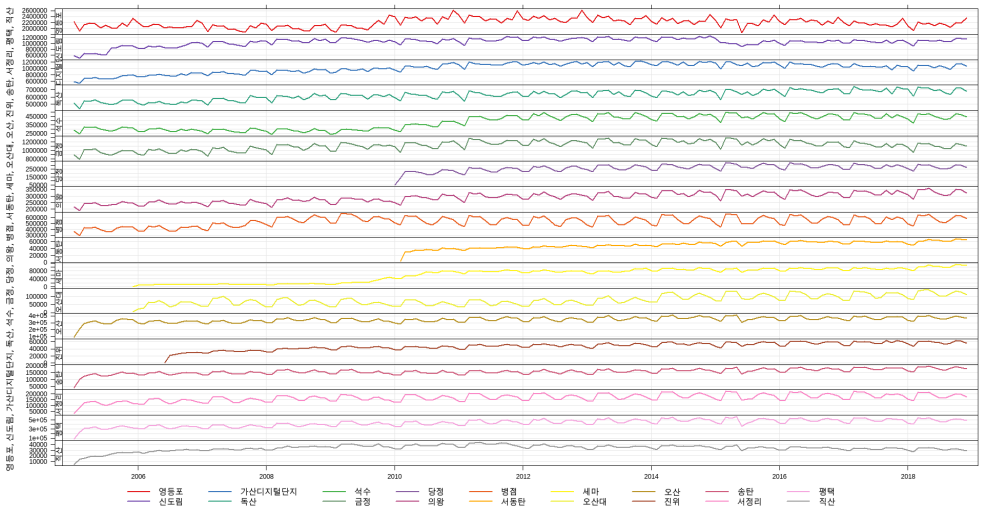

```
> windows()
> group_scale_2 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_scale_2), lwd = 2, y.relation = "free")
```




```
> windows()
> group_scale_3 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_scale_3), lwd = 2, y.relation = "free")
```




```
> windows()
> group_scale_4 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_scale_4), lwd = 2, y.relation = "free")
```



```
> windows()
> group_scale_5 %>%
+     mutate(date = seq(ymd("2005-01-01"), by = "month", length.out = 12*14)) %>%
+     timePlot(., pollutant = colnames(group_scale_5), lwd = 2, y.relation = "free")
```