Assignment #1

지하철 시각화

20190552 손지영

사용 data

- metro.csv: 2019년 11월의 일별 시간대별 지하철역 이용인원 데이터
- metro_coord.csv : 7호선 지하철역 위치 (위도, 경도) 데이터

데이터 전처리

- 문제를 풀기 전 전처리를 수행한다.

```
# 사용할 패키지 추가
library(ggplot2)
library(tidyr)
library(dplyr)
library(ggthemes)
library(ggmap)
```

```
# 데이터파일 읽기
metro = read.csv('metro.csv', fileEncoding = "euc-kr")
str(metro)
```

```
## 'data.frame':    16500 obs. of  30 variables:
##  $ X       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 날짜    : chr  "2019-11-01" "2019-11-01" "2019-11-01" "2019-11-01" ...
##  $ 호선    : chr  "1호선" "1호선" "1호선" "1호선" ...
##  $ 역번호  : int  150 150 151 151 152 152 153 153 154 154 ...
##  $ 역명    : chr  "서울역" "서울역" "시청" "시청" ...
##  $ 구분    : chr  "승차" "하차" "승차" "하차" ...
##  $ X04...05: int  32 2 3 0 5 0 11 1 2 0 ...
##  $ X05...06: int  438 353 89 182 143 211 187 127 83 175 ...
##  $ X06...07: int  529 2019 152 852 161 1078 154 477 115 622 ...
##  $ X07...08: int  1612 4520 289 2926 288 4395 302 1044 219 1817 ...
##  $ X08...09: int  3405 9906 435 9348 482 13000 386 3662 366 5234 ...
##  $ X09...10: int  2360 6525 481 4124 631 6669 550 3510 494 3292 ...
##  $ X10...11: int  2377 3571 716 2064 768 2964 841 2593 843 2292 ...
##  $ X11...12: int  2853 2951 1090 1889 1359 2501 1686 2813 1262 2349 ...
##  $ X12...13: int  3334 3190 1073 1538 1531 2127 1781 2646 1583 2160 ...
##  $ X13...14: int  3545 3348 1367 1751 1937 2108 2059 2718 1868 2159 ...
##  $ X14...15: int  2850 3179 1782 1403 2466 1926 2405 2579 2303 2071 ...
##  $ X15...16: int  4606 3265 2235 1431 2821 1718 3125 2103 2479 1559 ...
##  $ X16...17: int  4915 3575 2345 1218 3403 1778 3241 2010 2656 1777 ...
##  $ X17...18: int  7472 4191 3627 1249 5807 2396 3796 2033 3583 1599 ...
##  $ X18...19: int  11107 5445 7462 1486 10738 3746 4836 2582 5246 1776 ...
##  $ X19...20: int  5754 3882 2943 816 4680 2557 3192 1682 2709 1261 ...
##  $ X20...21: int  3920 2596 2249 439 3670 935 2107 675 1782 548 ...
##  $ X21...22: int  3799 2177 2199 288 4495 510 2452 512 1565 341 ...
##  $ X22...23: int  3369 1624 1460 296 4118 384 2407 380 1094 260 ...
##  $ X23...24: int  1678 912 640 202 2366 299 1394 323 596 153 ...
##  $ X00...01: int  228 478 62 47 271 75 236 143 66 73 ...
##  $ X01...02: int  2 39 0 1 1 0 6 10 1 1 ...
##  $ X02...03: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X03...04: int  0 0 0 0 0 0 0 0 0 0 ...
```

```
# 날짜 타입 바꾸기
metro$날짜 = as.Date(metro$날짜)

# 시간대 열을 한 열로 변경
metro_time = metro %>% gather(key='time', value='value', X04...05:X03...04)
str(metro_time)
```

```
## 'data.frame':    396000 obs. of  8 variables:
##  $ X     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 날짜  : Date, format: "2019-11-01" "2019-11-01" ...
##  $ 호선  : chr  "1호선" "1호선" "1호선" "1호선" ...
##  $ 역번호: int  150 150 151 151 152 152 153 153 154 154 ...
##  $ 역명  : chr  "서울역" "서울역" "시청" "시청" ...
##  $ 구분  : chr  "승차" "하차" "승차" "하차" ...
##  $ time  : chr  "X04...05" "X04...05" "X04...05" "X04...05" ...
##  $ value : int  32 2 3 0 5 0 11 1 2 0 ...
```

```
# 시간대 표시를 명확하게 바꾸기
metro_time$time = factor(metro_time$time, levels = c('X04...05', 'X05...06', 'X06...07', 'X07...08', 'X08...09',
'X09...10',
                                                     'X10...11', 'X11...12', 'X12...13', 'X13...14', 'X14...15', 'X15..
.16',
                                                     'X16...17', 'X17...18', 'X18...19', 'X19...20', 'X20...21', 'X21..
.22',
                                                     'X22...23', 'X23...24', 'X00...01', 'X01...02', 'X02...03',  'X03.
..04'),
                         labels= c('04-05', '05-06', '06-07', '07-08', '08-09', '09-10',
                                   '10-11', '11-12', '12-13', '13-14', '14-15', '15-16',
                                   '16-17', '17-18', '18-19', '19-20', '20-21', '21-22',
                                   '22-23', '23-24', '00-01', '01-02', '02-03', '03-04'))
```

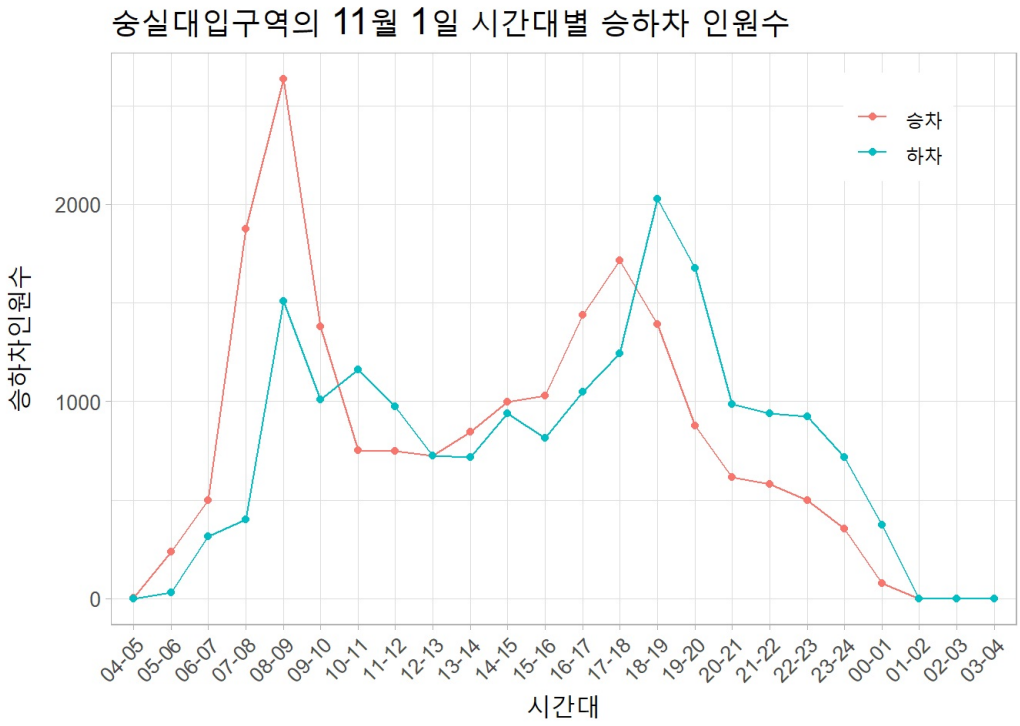
1번문제

숭실대입구(살피재) 역의 11월 1일의 시간대별 승차 및 하차 인원 수를 하나의 그래프로 시각화해보자.

```
# 숭실대입구(살피재) 역의 11월 1일 데이터 추출
df_time = subset(metro_time, (역명=='숭실대입구(살피재)')&(날짜=='2019-11-01'))
str(df_time)
```

```
## 'data.frame':    48 obs. of  8 variables:
##  $ X     : int  473 474 473 474 473 474 473 474 473 474 ...
##  $ 날짜  : Date, format: "2019-11-01" "2019-11-01" ...
##  $ 호선  : chr  "7호선" "7호선" "7호선" "7호선" ...
##  $ 역번호: int  2740 2740 2740 2740 2740 2740 2740 2740 2740 2740 ...
##  $ 역명  : chr  "숭실대입구(살피재)" "숭실대입구(살피재)" "숭실대입구(살피재)" "숭실대입구(살피재)" ...
##  $ 구분  : chr  "승차" "하차" "승차" "하차" ...
##  $ time  : Factor w/ 24 levels "04-05","05-06",..: 1 1 2 2 3 3 4 4 5 5 ...
##  $ value : int  3 0 237 31 498 314 1875 400 2637 1510 ...
```

```
# 그래프 그리기
ggplot(df_time, aes(x=time, y=value, group=구분, color=구분)) +
  geom_line() +
  theme_light() +
  geom_point(data=df_time, aes(x=time, y=value, group=구분, color=구분)) +
  labs(x='시간대', y='승하차인원수', title = '숭실대입구역의 11월 1일 시간대별 승하차 인원수') +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1 )) +
  theme(legend.title=element_blank(),
        legend.position = c(0.87, 0.87),
        text = element_text(size=15),
        axis.text = element_text(size=10),
        axis.title.x = element_text(margin=margin(t=3)),
        axis.title.y = element_text(margin=margin(r=10))) +
  scale_fill_brewer(palette='Set2')
```



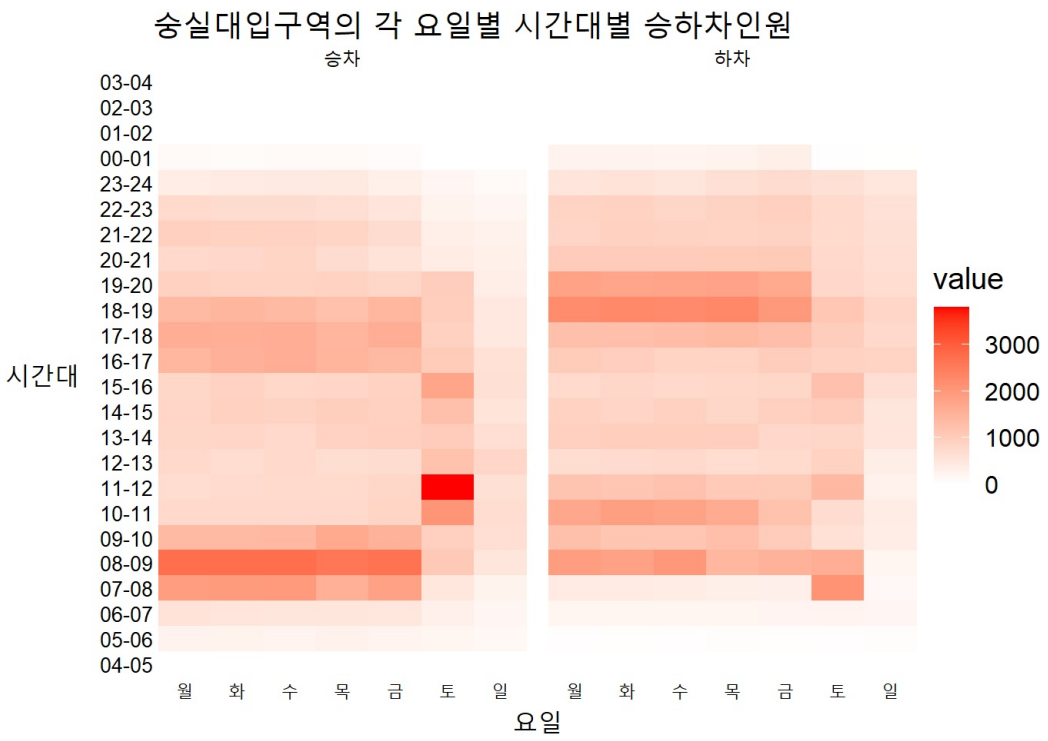
2번 문제

숭실대입구(살피재) 역의 11월 10일(일)부터 11월16일(토)까지 일주일간 각 요일별 시간대별 승차인원과 하차인원의 분포를 각각 heat map으로 시각화해보자.

```
# 숭실대입구역만 추출
df_ssu = subset(metro_time, (역명=='숭실대입구(살피재)'))

# 11.10-11.16 데이터만 추출
df_week = df_ssu[('2019-11-10' <= df_ssu$날짜 & df_ssu$날짜 <= '2019-11-16'),]

# 요일 데이터 추가 및 순서 지정
df_week$weekday = weekdays(df_week$날짜)
df_week$weekday = factor(df_week$weekday, levels = c('월요일','화요일','수요일','목요일','금요일','토요일','일요일'),
                         labels = c('월','화','수','목','금','토','일'))
str(df_week)
```

```
## 'data.frame':    336 obs. of  9 variables:
##  $ X      : int  5423 5424 5973 5974 6523 6524 7073 7074 7623 7624 ...
##  $ 날짜   : Date, format: "2019-11-10" "2019-11-10" ...
##  $ 호선   : chr  "7호선" "7호선" "7호선" "7호선" ...
##  $ 역번호 : int  2740 2740 2740 2740 2740 2740 2740 2740 2740 2740 ...
##  $ 역명   : chr  "숭실대입구(살피재)" "숭실대입구(살피재)" "숭실대입구(살피재)" "숭실대입구(살피재)" ...
##  $ 구분   : chr  "승차" "하차" "승차" "하차" ...
##  $ time   : Factor w/ 24 levels "04-05","05-06",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ value  : int  0 0 1 0 1 0 1 0 4 0 ...
##  $ weekday: Factor w/ 7 levels "월","화","수",..: 7 7 1 1 2 2 3 3 4 4 ...
```

```
# 그래프 그리기
ggplot(df_week, aes(x=weekday, y=time, fill=value)) +
  geom_tile() +
  theme_void() +
  labs(x='요일',y='시간대',title = '숭실대입구역의 각 요일별 시간대별 승하차인원') +
  scale_fill_gradient(low = "white", high = "red") +
  theme(text = element_text(size=15),
        axis.text = element_text(size=10),
        axis.title.x = element_text(margin=margin(t=3)),
        axis.title.y = element_text(margin=margin(r=10))) +
  facet_wrap(~구분)
```



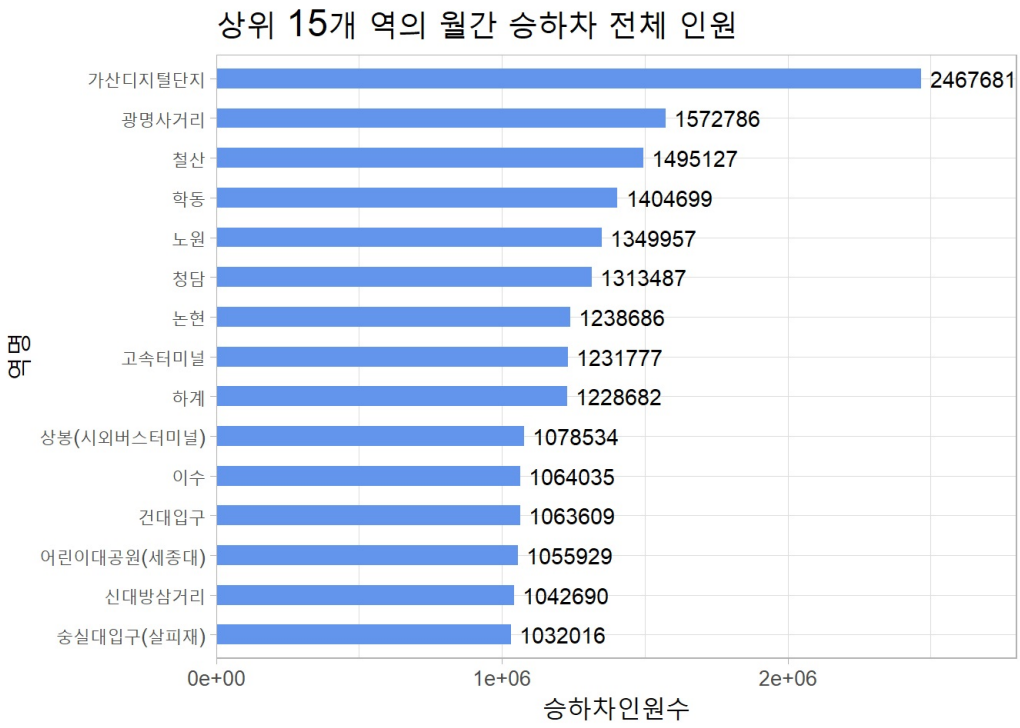
3번 문제

7호선의 모든 역 중에서 유동인구(월간 승하차 전체인원)가 가장 많은 15개 역에 대한 유동인구 수를 그래프로 시각화해보자.

```
# 7호선 데이터만 추출
all_7 = subset(metro_time, 호선=='7호선')

# 역명을 기준으로 총합 구하기
df_sum = all_7 %>% group_by(역명) %>% summarize(total = sum(value))

# 유동인구수를 기준으로 높은 것부터 정렬
df_sort = df_sum[order(df_sum$total,decreasing = TRUE),]

# 상위 15개의 역 추출
df_top = df_sort[1:15,]
str(df_top)
```

```
## tibble [15 × 2] (S3: tbl_df/tbl/data.frame)
##  $ 역명 : chr [1:15] "가산디지털단지" "광명사거리" "철산" "학동" ...
##  $ total: int [1:15] 2467681 1572786 1495127 1404699 1349957 1313487 1238686 1231777 1228682 1078534 ...
```

```
# 그래프 그리기
ggplot(df_top, aes(x=reorder(역명, total), y=total)) +
  geom_bar(stat='identity', width=0.5, fill='cornflowerblue') +
  theme_light() +
  coord_flip() +
  labs(x='역명', y='승하차인원수', title = '상위 15개 역의 월간 승하차 전체 인원') +
  scale_y_continuous(expand=c(0,0), limits=c(0, 2800000)) +
  theme(text = element_text(size=15),
        axis.text = element_text(size=10)) +
  geom_text(aes(label=total), hjust=-0.1)
```



4번 문제

7호선 지하철역 위치 정보를 활용하여 7호선의 모든 역에 대한 유동인구 분포를 지도 위에 시각화해보자. 크기, 투명도 등을 활용하여 분포를 표현할 수 있다.

```
# 위치 데이터 불러오기
crd = read.csv('metro_coord.csv', fileEncoding = "euc-kr")
str(crd)
```

```
## 'data.frame':    51 obs. of  6 variables:
##  $ X     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 역번호: int  2748 2732 2729 2736 2718 2750 2727 2760 2753 2747 ...
##  $ 역명  : chr  "가산디지털단지" "강남구청" "건대입구" "고속터미널" ...
##  $ 호선  : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ lat   : num  37.5 37.5 37.5 37.5 37.6 ...
##  $ lon   : num  127 127 127 127 127 ...
```

```
# 모든 역에 대해 역명을 기준으로 총유동인구수 구하기
all_sum = all_7 %>% group_by(역명) %>% summarize(sum = sum(value))
str(all_sum)
```

```
## tibble [51 × 2] (S3: tbl_df/tbl/data.frame)
##  $ 역명: chr [1:51] "가산디지털단지" "강남구청" "건대입구" "고속터미널" ...
##  $ sum : int [1:51] 2467681 1005604 1063609 1231777 846742 1572786 825795 601076 465364 1012929 ...
```

```
# 위치데이터와 결합하기
df = left_join(crd, all_sum, by='역명')
str(df)
```

```
## 'data.frame':    51 obs. of  7 variables:
##  $ X     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ 역번호: int  2748 2732 2729 2736 2718 2750 2727 2760 2753 2747 ...
##  $ 역명  : chr  "가산디지털단지" "강남구청" "건대입구" "고속터미널" ...
##  $ 호선  : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ lat   : num  37.5 37.5 37.5 37.5 37.6 ...
##  $ lon   : num  127 127 127 127 127 ...
##  $ sum   : int  2467681 1005604 1063609 1231777 846742 1572786 825795 601076 465364 1012929 ...
```

```
# 서울 위치 설정
bbox_seoul = c(left=126.691422, bottom=37.455942, right=127.129972, top=37.704339)
seoul = get_stamenmap(bbox=bbox_seoul, zoom=11, maptype='terrain')
```

```
## i Map tiles by Stamen Design, under CC BY 3.0. Data by OpenStreetMap, under ODbL.
```

```
# 그래프 그리기
ggmap(seoul, base_layer=ggplot(df, aes(x=lon, y=lat, size=sum, color=역명))) +
  geom_point(alpha=0.5) +
  labs(title = '7호선의 모든 역에 대한 유동인구 분포') +
  scale_size(range=c(1,20)) +
  theme(legend.title=element_blank()) +
  scale_color_discrete(guide=FALSE) +
  theme_void()
```

```
## Warning in missing(base_layer) || base_layer == "auto": 'length(x) = 9 > 1' in
## coercion to 'logical(1)'
```

```
## Warning: The `guide` argument in `scale_*()` cannot be `FALSE`. This was deprecated in
## ggplot2 3.3.4.
## ℹ Please use "none" instead.
```

7호선의 모든 역에 대한 유동인구 분포