데이터과학 실습 보고서

-R을 이용한 타슈데이터 분석-

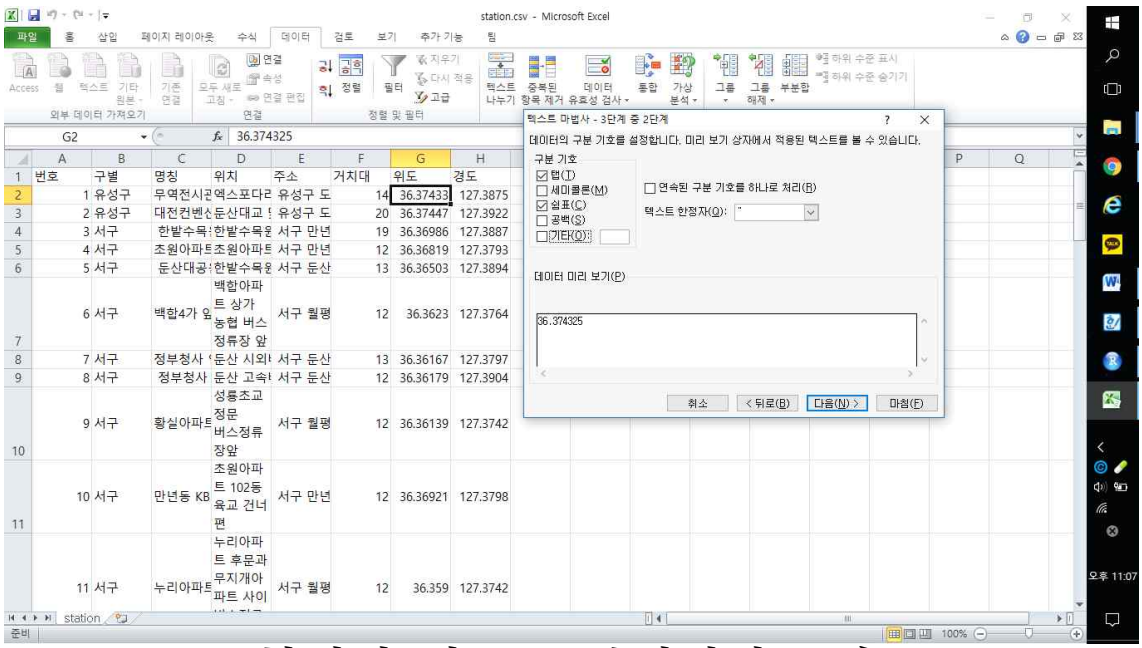
2017.3.22

201201185 장진우

1.데이터 가공 및 합치기

2주차 과제에서 사용했던 데이터들을 그대로 사용하지만 과제 1-2 GoogleMap에 사용하기 위해서는 station.csv의 좌표를 가공해야 합니다.

이번에는 간단하게 엑셀기능을 이용하여 데이터 -> 텍스트 나누기 -> 쉼표체크를 이용하여 손쉽게 합쳐진 좌표를 분리해낼 수 있습니다.



<합쳐진 좌표를 분리시킨 모습>

과제1.

1)데이터 분석 과정

이번 사용정류장 TOP10은 앞서 Python으로 작성했던 코드에서 조금 응용을 하여 빈도수를 구하는 함수(table())를 사용하여 rent_station과 return_station 각각의 빈도수를 구한 후 오름차순 정렬하여 더하는 것으로 해결할 수 있습니다.

2)코드 설명 및 결과화면

-데이터 분석

```
#필요한 Library 설정
library(ggplot2)
library(ggmap)

#.csv 데이터 읽어 들이기
tashu = read.csv(file='tashu.csv',encoding = 'UTF-8')
station <- read.csv("station.csv")

#필요한 데이터 추출
rent_station <-data.frame(as.numeric(tashu$RENT_STATION))
return_station <-data.frame(as.numeric(tashu$RETURN_STATION))
station_info <- as.numeric(station$'번호')
#table로 변형시켜 빈도수 추출 및 정렬
rent_station_table <-table(rent_station)
return_station_table <-table(return_station)

rent_station_index <-order(rent_station_table,decreasing=TRUE)
return_station_index <-order(return_station_table,decreasing=TRUE)

rent_station_sort <-sort(rent_station_table,decreasing=TRUE)
return_station_sort <-sort(return_station_table,decreasing=TRUE)
#rent_station의 빈도수에 return_station의 빈도수를 더함
total_station_table=rent_station_table
count=1
for(i in return_station_index){
  total_station_table[i]=total_station_table[i]+return_station_sort[count]
  count=count+1
  print(i)
}
#최종 합쳐진 빈도수의 정렬
total_station_index <- order(total_station_table,decreasing = TRUE)
total_station_table_sort <- sort(total_station_table,decreasing = TRUE)
#TOP10 추출
top_10 <-total_station_index[c(1:10)]
top_10
station_num <-station_info[top_10[c(1:10)]]
Freq <- total_station_table_sort[c(1:10)]
result <- cbind(station_num,Freq)
result <- as.data.frame(result)

result
```

우선 tashu.csv와 station.csv를 읽어온 후

numeric 형태로 rent_station과 return_station을 각각 저장

하고, station_info또한 numeric형태로 저장을 합니다. 그 후, table()을 사용하여 rent_station과 return_station의 빈도수를 계산하여 각각 저장합니다. 그 후, 빈도수를 이용하여 오름차순 정렬하여 위치와 값을 따로 저장하고 rent_station의 오름차순 상위 10개의 데이터에 return_station의 오름차순 상위 10개의 데이터를 합치면 최종 인기정류장 TOP10이 완성되게 됩니다.

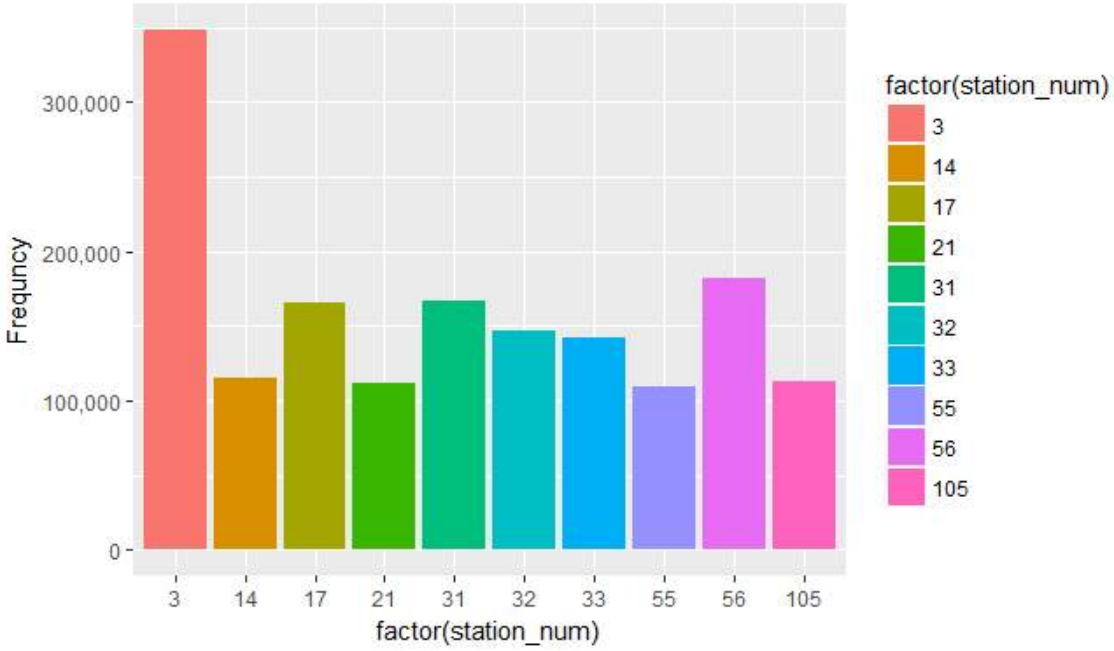
```
> result
    station_num   Freq
3             3 348977
56           56 182114
31           31 166866
17           17 165778
32           32 147063
33           33 142310
14           14 114878
105         105 112921
21           21 111715
55           55 110045
>
```

<결과 화면>

-막대 그래프

```
dia_bar <- ggplot(result, aes(x=factor(station_num), y=Freq, fill=factor(station_num))) +
  geom_bar(stat='identity') + scale_y_continuous(name="Frequncy", labels = scales::comma)

dia_bar
```

위의 결과 데이터를 가지고 ggplot()이라는 함수를 사용하여 막대그래프를 그릴 수 있는데, 각각의 인자에 result는 데이터를 의미하고 aes()는 x,y축의 값들을 의미합니다. 또한 fill을 이용하여 그래프 우측에 데이터를 시각화 할 수 있고, geom_bar()를 이용하여 bar 그래프의 설정을 할 수 있습니다.



<결과 막대그래프>

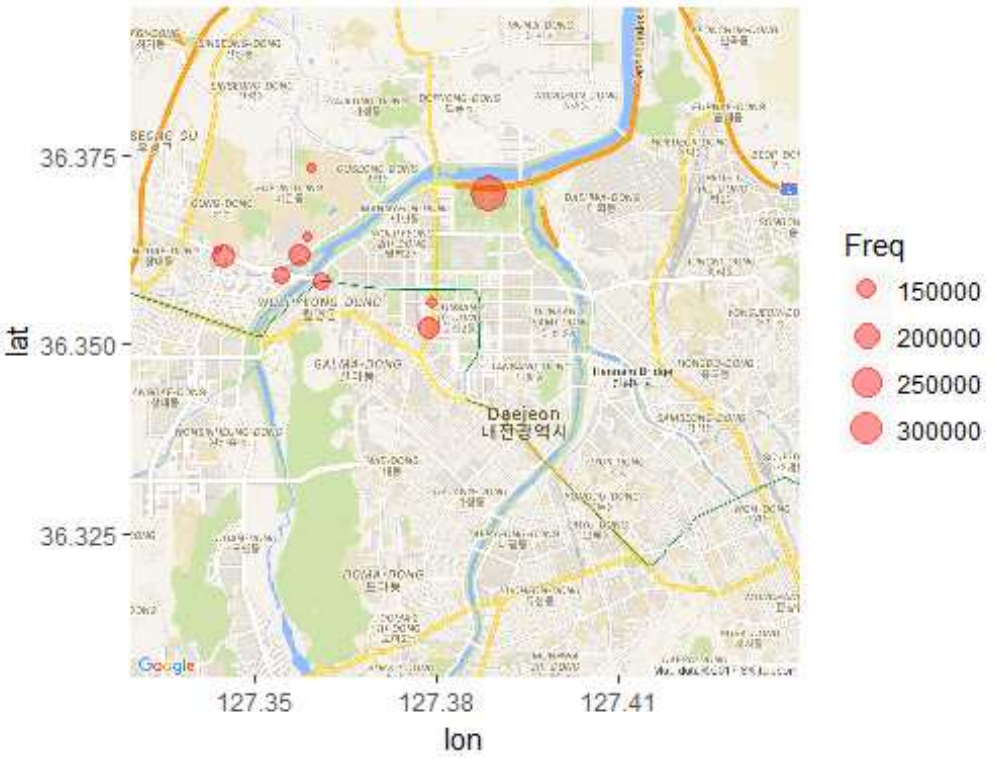
-구글맵

```
#------------------TOP_10에 대한 GoogleMap-----------------------------------------------------------------------------
station_location <- subset(station, select=c("번호", "위도","경도"))
colnames(station_location) <- c("번호", "위도","경도")

top10_station_location <- merge(result, station_location,by=0)
top10_station_location

map_point <- ggmap(get_googlemap("Daejon", maptype="roadmap", zoom=13))+geom_point(data=top10_station_location,
                                                    aes(경도,위도,size=Freq),alpha=0.4,colour="red" )
map_point
#---------------------------------------------------------------------------------------------------------------------
```

station_location이라는 변수에 station의 정보 중 “번호”,“위도”,“경도”만을 뽑아내어 저장을 합니다. 그리고 각각 열의 이름을 “번호”,“위도”,“경도”로 지정해 줍니다. 그 후, result에서의 번호와 station_location의 번호를 기준으로 merge하여 top10의 location정보를 저장합니다.

구글맵은 ggmap()함수를 통해 구현할 수 있는데 우선 “Dajeon”의 지도를 roadmap 타입 이고 13크

기인 지도로 만들어 기본 대전 지도의 틀을 제작합니다. 거기에 geom_point를 이용하여 점을 찍을 수 있는데, data=을 통해 데이터를 집어넣을 수 있고, 위에서 구했던 top10_location을 삽입한 후, aes()를 통해 x,y축의 데이터를 삽입하고, size를 Frequency의 값으로 설정하여 Frequency의 값에 따라 점의 크기가 달라지게 설정을 합니다. 그 후 부수적으로 투명도나 색깔을 설정하여 그래프를 완성합니다.



<결과 Google Map>

과제2.

1)데이터 분석 과정

앞서 구했던 사용정류장 TOP10의 데이터는 rent_station과 return_station의 빈도수를 따로 구하여 더하였지만 지금 구하는 인기 경로 TOP20에서는 rent와 return station의 정보를 합쳐 빈도수를 구하여 오름차순 정렬하면 된다.

2)코드 설명 및 결과 화면

-데이터 분석

```
#----------------------- TOP20경로 ------------------------------------------------------------
rent_return <- table(tashu$RENT_STATION, tashu$RETURN_STATION)
rent_return

rent_return = as.data.frame(rent_return)
rent_return
top20_rent_return <- head(rent_return[with(rent_return, order(-Freq)),], 20)
colnames(top20_rent_return) <- c("rent", "return", "Freq")
top20_rent_return
#-------------------------------------------------------------------------------------------
```

우선 rent_return 변수에 table()을 이용하여 rent_station과 return_station의 정보를 합친 다음 data.frame()으로 형 변환을 시켜준다. 그 후 , rent_return값의 정보를 with()함수를 이용하여 추출하고 head()함수로 20개까지 뽑아냅니다. 그리고 열의 이름을 “rent”,“return”,“Freq”로 설정해 준 후 출력해 줍니다.

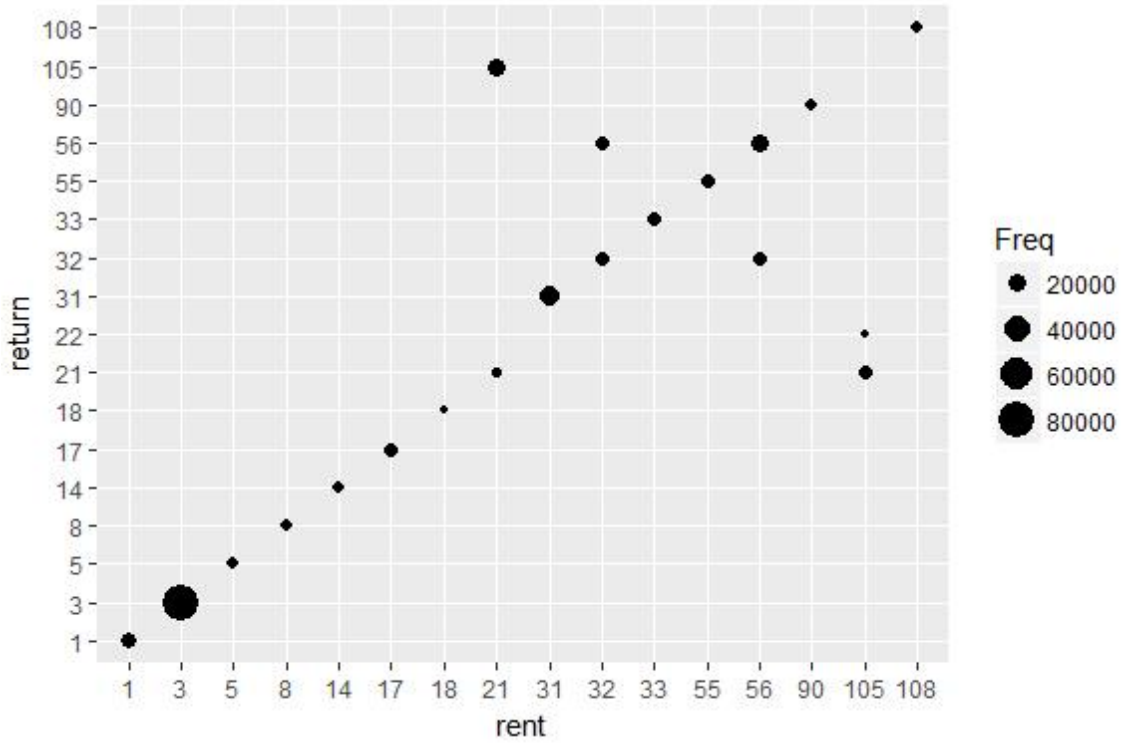
	Var1	Var2	Freq
407	3	3	84496
6091	31	31	21749
11166	56	56	18343
21029	21	105	17220
1	1	1	14489
6294	32	32	12177
4145	105	21	12154
6497	33	33	11973
3249	17	17	11966
6318	56	32	11868
11142	32	56	11118
10963	55	55	11111
813	5	5	10798
21722	108	108	9926
18068	90	90	9650
1422	8	8	9560
2640	14	14	9231
4061	21	21	9006
3452	18	18	8192
4347	105	22	8074

<결과 화면>

-그래프

```
#----------------------- TOP20경로 그래프 ----------------------------------
ggplot(top20_rent_return, aes(x=factor(rent), y=factor(return), size=Freq))+
  geom_point() + xlab("rent") + ylab("return")
```

ggplot()함수를 이용하여 앞에서 구한 데이터인 top20_rent_return의 정보를 넣고, x,y축에 각각 factor(rent),factor(return)의 값을 넣어 주고, 값에 따라 크기가 달라지기 위해 size를 Freq로 넣어줍니다. 그 후 x좌표는 rent, y좌표에 return을 대입하여 점을 찍어 줍니다.


return
108
105
90
56
55
33
32
31
22
21
18
17
14
8
5
3
1
rent
1
3
5
8
14
17
18
21
31
32
33
55
56
90
105
108
Freq
20000
40000
60000
80000

<결과 그래프>