

위치기반데이터 분석(Location based Data Analytics)

14강 R 공간분석프로그래밍



ggmap 라이브러리 사용하기(1)

get_googlemap 함수

Parameter	설명
center	지도의 중심좌표 the central coordinates of a map
zoom	지도의 확대크기로서 3(대륙) ~ 21(빌딩) 기본값은 10(도시) Default value 10(city)
size	지도의 가로와 세로 픽셀 크기 기본값은 640 * 640 (c(640, 640))
maptype	출력될 지도 유형 기본값은 "terrain" roadmap, terrain, stellite, hybrid 선택

ggmap 설치

```
install.packages('devtools')  
library('devtools')
```

```
install_github('dkahle/ggmap', ref='tidyup')
```

```
library('ggmap')
```

ggmap 라이브러리 사용하기(2)

● 소스코드

```
install.packages('devtools')
library('devtools')

install_github('dkahle/ggmap', ref='tidyup')

library('ggmap')

usethis::edit_r_environ()

mykey <- Sys.getenv('GOOGLE_API_KEY')
register_google(key=mykey)

names<-c("용두암", "성산일출봉", "정방폭포", "중문관광단지", "한라산1100고지",
         "차귀도")

addr<-c("제주시 용두암길 15",
        "서귀포시 성산읍 성산리",
        "서귀포시 동홍동 299-3",
        "서귀포시 중문동 2624-1",
        "서귀포시 색달동 산1-2",
        "제주시 한경면 고산리 125"
        )
gc<-geocode(encutf8(addr))

df<- data.frame(name=names, lon=gc$lon, lat=gc$lat)

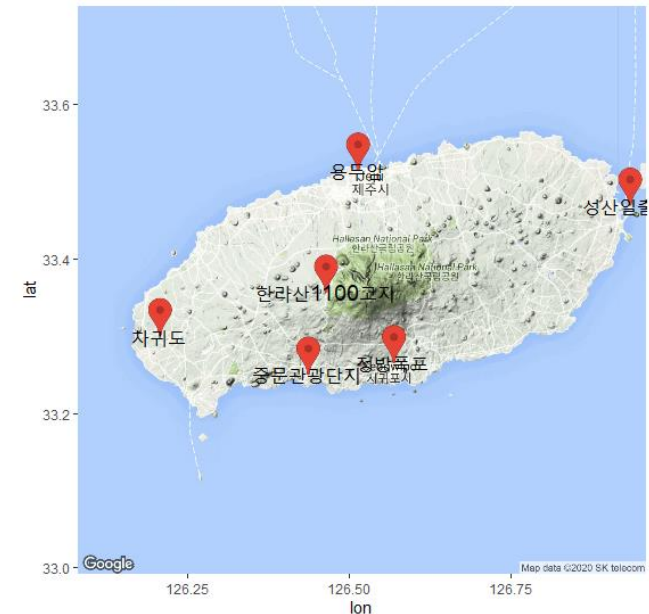
cen<-c(mean(df$lon), mean(df$lat))

map<-get_googlemap(center=cen, maptype = "terrain", zoom=10, size = c(640,640),
                   markers = gc)

gmap<-ggmap(map)

gmap+geom_text(data=df, aes(x=lon, y=lat), size=5, label=df$name)
```

● 결과



공용API를 활용하여 부동산 실거래가 분석하기(1)

data.go.kr에서 제공하는 공용API에서 실거래가 정보 가져오기

```
install.packages("tidyverse")
install.packages("httr")
install.packages("rvest")
install.packages("jsonlite")

library(tidyverse)
library(httr)
library(rvest)
library(jsonlite)

mykey <- Sys.getenv("DATAGOKR_KEY")

url <- "http://openapi.molit.go.kr:8081/OpenAPI_ToolInstallPackage/service/rest/RTMSOBSvc/getRTMSDataSvcAptTrade?"

res <- GET(url = url, query=list(LAWD_CD = '41135', DEAL_YMD='202003', serviceKey=mykey %>% I()))

res %>% content(as='text', encoding = "UTF-8") %>% fromJSON() -> json

df<-json$response$body$items$item
```

dataframe에 실거래가 정보

	거래 금액	건축 년도	년	법정 동	아파트	월	일	전용 면적	지번	지역 코드	층
1	64,000	1993	2020	분당동	장안타운(건영)	3	4	70.6800	66	41135	4
2	60,500	1992	2020	분당동	셋별마을(동성)	3	4	59.4000	35	41135	2
3	85,000	1993	2020	분당동	장안타운(건영)	3	5	131.5500	66	41135	7
4	78,000	1993	2020	분당동	셋별마을(라이프)	3	6	84.9900	34	41135	1
5	89,900	1994	2020	분당동	셋별마을(우방)	3	7	84.9900	38	41135	2
6	62,800	1992	2020	분당동	셋별마을(라이프)	3	10	58.1700	34	41135	12
7	89,000	1992	2020	분당동	셋별마을(동성)	3	12	84.8200	35	41135	2
8	85,500	1994	2020	분당동	장안타운(건영)	3	17	131.5500	66	41135	10
9	51,000	1993	2020	분당동	장안타운(건영)	3	17	67.7300	66	41135	1
10	112,500	1994	2020	분당동	셋별마을(우방)	3	24	133.8600	38	41135	9
11	100,000	1993	2020	수내동	푸른마을(쌍용)	3	3	84.7200	75	41135	6
12	102,000	1993	2020	수내동	푸른마을(신성)	3	3	84.7200	73	41135	16

공용API를 활용하여 부동산 실거래가 분석하기(2)

일정기간 동안의 실거래가 정보 수집 및 범주형 Vector로 변경

```
months<-seq(from=as.Date(x='2019-01-01'), to=as.Date(x='2020-05-01'), by='1 month')
months<-format(x=months, format='%Y%m')

result<-data.frame()

for(month in months) {
  cat('현재 ', month, '기간의 데이터를 수집중!!!\n')
  res <- GET(url = url, query=list(LAWD_CD = '41135', DEAL_YMD=month, serviceKey=mykey %>% I()))
  res %>% content(as='text', encoding = "UTF-8") %>% fromJSON() -> json
  df<-json$response$body$items$item
  result<-rbind(result, df)
  Sys.sleep(time=1)
}
result$거래금액 %>% str_remove(pattern = ',') %>% as.numeric() -> result$거래금액

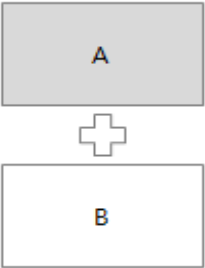

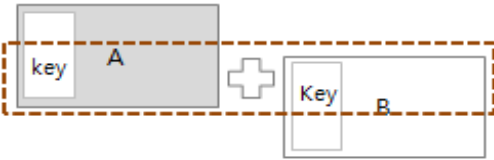
head(x=result$법정동)
as.factor(x=result$법정동) %>% head()

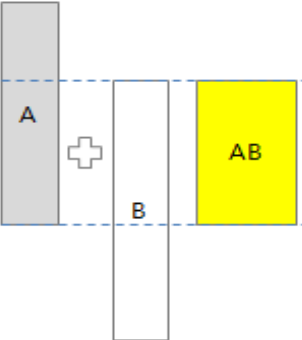
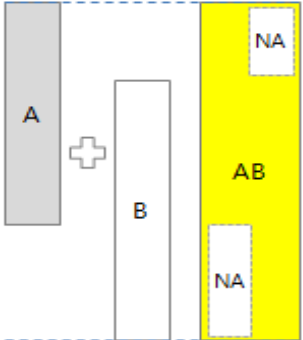
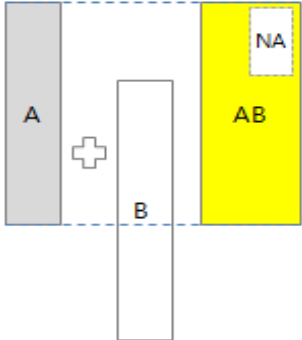
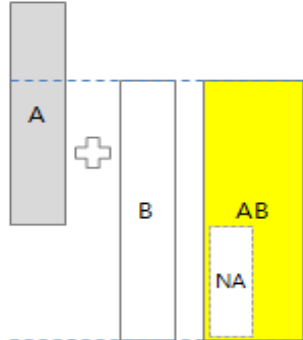
map_df(.x=result[,c(4,5,10)], .f=as.factor) -> result[,c(4,5,10)]

saveRDS(object=result, file='apt_price_bundang')
```

공용API를 활용하여 부동산 실거래가 분석하기(3)

데이터프레임에서 데이터 연결 방법

rbind(A, B)	cbind(A, B)	merge(A, B, by='key')
 <p>행 결합</p>	 <p>열 결합</p>	 <p>동일 key 값 기준 결합</p> <p>http://rfriend.tistory.com</p>

Inner Join	Outer Join	Left Outer Join	Right Outer Join
<p>merge(A, B, by='key')</p>  <p>* Key 값 기준 정렬 가정</p>	<p>merge(x=A, y=B, by='key', all = TRUE)</p> 	<p>merge(x=A, y=B, by='key', all.x = TRUE)</p> 	<p>merge(x=A, y=B, by='key', all.y = TRUE)</p>  <p>http://rfriend.tistory.com</p>

공용API를 활용하여 부동산 실거래가 분석하기(4)

통계분석하기

```
mean(x=result$거래금액)

median(x=result$거래금액)

result$건축년도 %>% table() %>% sort(decreasing = TRUE)

min(x=result$거래금액)

max(x=result$거래금액)

range(x=result$거래금액)

range(x=result$거래금액) %>% diff()

quantile(x=result$거래금액)

quantile(x=result$거래금액, probs = c(0.95, 0.99))

IQR(x=result$거래금액)

var(x=result$거래금액)

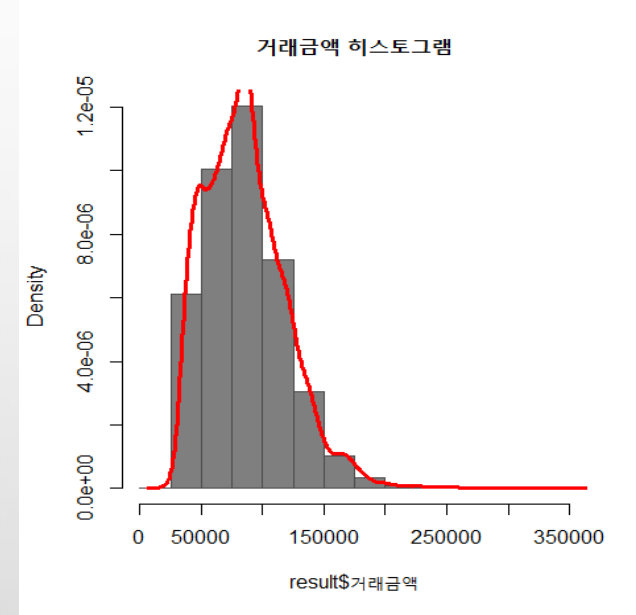
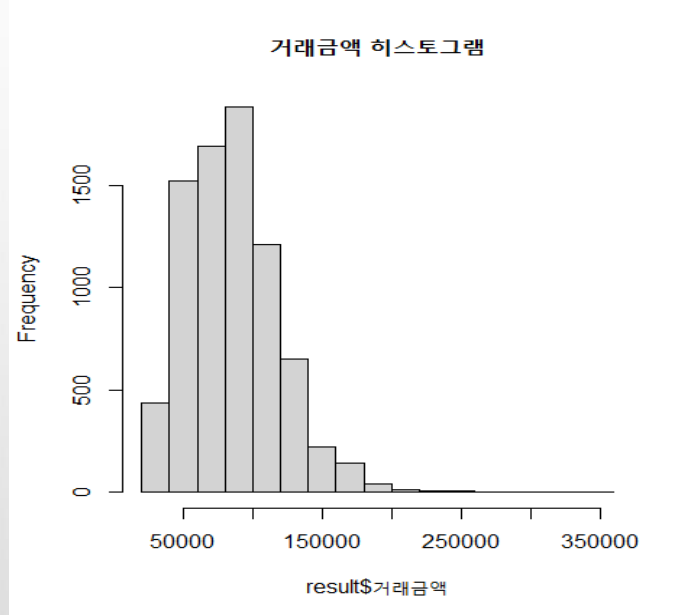
sd(x=result$거래금액)
```

```
> mean(x=result$거래금액)
[1] 86040.16
> median(x=result$거래금액)
[1] 83750
> result$건축년도 %>% table() %>% sort(decreasing = TRUE)
.
1995 1993 1992 1994 2009 1996 2003 1991 2004 2008 2015 2007 2010 2011 1998 2005 2013 2012
1740 1371 1120 1104 951 370 330 304 194 117 58 43 42 42 14 14 9 7
> min(x=result$거래금액)
[1] 20600
> max(x=result$거래금액)
[1] 350000
> range(x=result$거래금액)
[1] 20600 350000
> range(x=result$거래금액) %>% diff()
[1] 329400
> quantile(x=result$거래금액)
 0%    25%   50%   75%  100%
20600 60500 83750 105000 350000
> quantile(x=result$거래금액, probs = c(0.95, 0.99))
 95%    99%
144500 180000
> IQR(x=result$거래금액)
[1] 44500
> var(x=result$거래금액)
[1] 1089309605
> sd(x=result$거래금액)
[1] 33004.69
```

공용API를 활용하여 부동산 실거래가 분석하기(5)

그래프 그리기(히스토그램)

```
hist(x=result$거래금액, main='거래금액 히스토그램')  
  
hist(x=result$거래금액, breaks = seq(from=0, to = 350000, by = 25000),  
     freq = FALSE,  
     col='gray50',  
     border = 'gray30',  
     main='거래금액 히스토그램')  
lines(x=density(x=result$거래금액),  
      lwd = 3,  
      col='red')
```

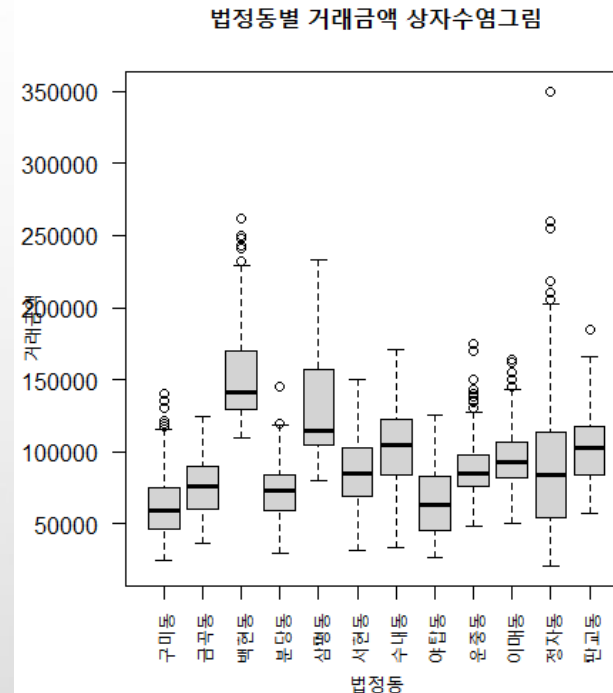
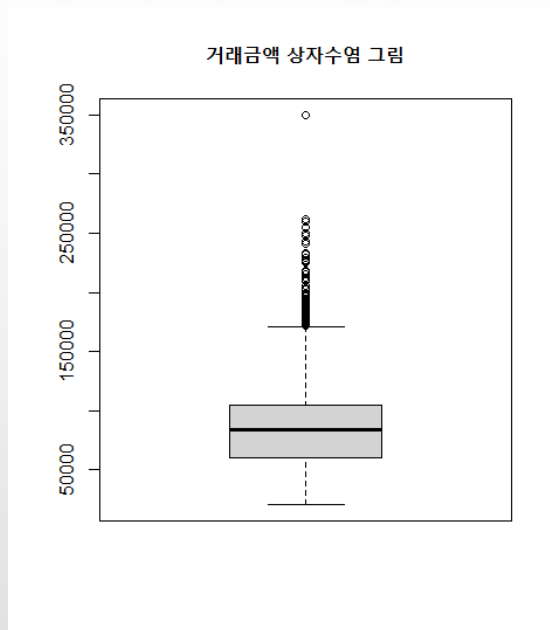


공용API를 활용하여 부동산 실거래가 분석하기(6)

그래프 그리기(상자수염그림)

```
boxplot(x=result$거래금액, main='거래금액 상자수염 그림')
```

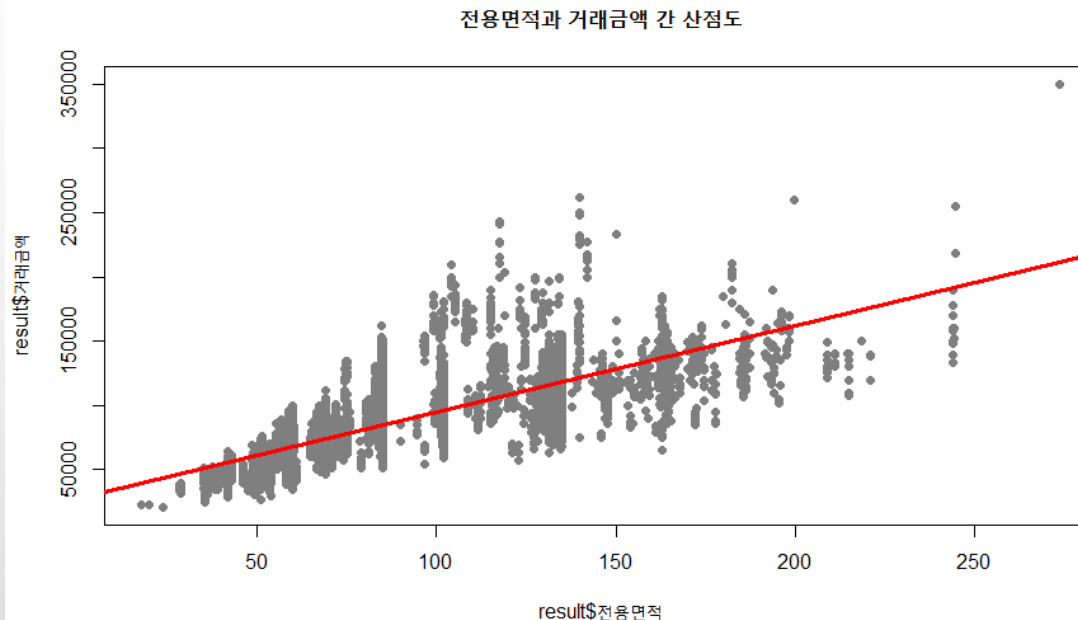
```
boxplot(formula = 거래금액 ~ 법정동,  
        data= result,  
        las=2,  
        main='법정동별 거래금액 상자수염그림')
```



공용API를 활용하여 부동산 실거래가 분석하기(7)

그래프 그리기(산점도)

```
plot(x=result$전용면적, y=result$거래금액,  
     pch=19,  
     col = 'gray50',  
     main='전용면적과 거래금액 간 산점도')  
abline(reg=lm(formula = 거래금액 ~ 전용면적, data=result),  
       lwd = 3,  
       col = 'red')
```



공용API를 활용하여 부동산 실거래가 분석하기(8)

그래프 그리기(산점도)

```
result %>% mutate(지번주소=str_c('경기도 성남시 분당구', 법정동, 지번, sep=''),
                  면적평균 = 거래금액/전용면적) %>%
  select(아파트, 지번주소, 면적평균) %>% group_by(아파트, 지번주소) %>%
  summarise(단지평균 = mean(면적평균)) -> df

library(ggmap)
my_gokey <- Sys.getenv('GOOGLE_API_KEY')
register_google(key=my_gokey)

addr_v = c()

for(atmp in df['지번주소']) {addr_v <- atmp}

gc<-geocode(enc2utf8(addr_v))

df$위도 <- NA
df$경도 <- NA

for(i in 1:nrow(x=df)) {
  df$위도[i] <- gc$lat[i]
  df$경도[i] <- gc$lon[i]
}

center <- c(lon=median(x=df$경도), lat= median(x=df$위도))

qmap(location = c(lon = center[1], lat = center[2]),
      zoom = 13,
      maptype = 'hybrid',
      source = 'google') + geom_point(data=df, mapping= aes(x=경도, y=위도, color=단지평균),
                                       shape = 19,
                                       size = 2) + scale_color_gradient(low='yellow', high='red') +
  theme(legend.position = 'None')
```

