final EDA base

2015122016

2019년 5월 21일

#import data

```
train <- read.csv("C:/Users/Jooyeon Kim/Desktop/YONSEI/ESC/2019-1/[ESC 19-Spring] Final Project/dataset/
school <- read.csv("C:/Users/Jooyeon Kim/Desktop/YONSEI/ESC/2019-1/[ESC 19-Spring] Final Project/dataset
subway <- read.csv("C:/Users/Jooyeon Kim/Desktop/YONSEI/ESC/2019-1/[ESC 19-Spring] Final Project/dataset
```

#data summary

```
glimpse(train)
```

```
## Observations: 90,000
## Variables: 24
## $ key                                <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1...
## $ apartment_id                       <int> 2944, 4307, 4307, 1337, 936, 936...
## $ transaction_year_month             <int> 201710, 201710, 201710, 201710, ...
## $ transaction_date                   <fct> 11~20, 11~20, 11~20, 11~20, 11~2...
## $ year_of_completion                 <int> 1993, 1995, 1995, 1989, 1983, 19...
## $ exclusive_use_area                 <dbl> 112.24, 59.52, 59.52, 35.10, 51....
## $ floor                              <int> 1, 4, 4, 10, 8, 6, 5, 5, 6, 21, ...
## $ latitude                           <dbl> 37.63981, 37.49666, 37.49666, 37...
## $ longitude                          <dbl> 127.0756, 126.8471, 126.8471, 12...
## $ address_by_law                     <int> 1135010400, 1153010800, 11530108...
## $ total_parking_capacity_in_site     <int> 666, 195, 195, NA, 360, 360, 674...
## $ total_household_count_in_sites     <int> 498, 218, 218, 1635, 360, 360, 1...
## $ apartment_building_count_in_sites  <int> 6, 1, 1, 9, 5, 5, 14, 14, 1, 2, ...
## $ tallest_building_in_sites          <int> 15, 22, 22, 15, 11, 11, 15, 15, ...
## $ lowest_building_in_sites           <int> 11, 21, 21, 15, 10, 10, 15, 15, ...
## $ heat_type                          <fct> individual, individual, individu...
## $ heat_fuel                          <fct> gas, gas, gas, gas, gas, gas, ga...
## $ room_id                            <int> 6316, 7677, 7678, 3643, 2441, 24...
## $ supply_area                        <dbl> 132.66, 80.63, 80.90, 46.57, 72....
## $ total_household_count_of_area_type <int> 274, 1, 87, 420, 170, 170, 120, ...
## $ room_count                         <int> 4, 2, 2, 2, 2, 2, 3, 3, 3, 3, 4,...
## $ bathroom_count                     <int> 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2,...
## $ front_door_structure               <fct> stairway, corridor, corridor, co...
## $ transaction_real_price             <dbl> 5.350e+08, 2.400e+08, 2.400e+08,...
```

```
colSums(is.na(train))
```

```
##                                key                       apartment_id
##                                  0                                  0
##              transaction_year_month                   transaction_date
##                                  0                                  0
##                  year_of_completion                 exclusive_use_area
```

```
##                                0                                0
##                            floor                         latitude
##                                0                                0
##                        longitude                   address_by_law
##                                0                                0
##    total_parking_capacity_in_site   total_household_count_in_sites
##                             3623                                0
##  apartment_building_count_in_sites        tallest_building_in_sites
##                                0                                2
##          lowest_building_in_sites                        heat_type
##                                2                              360
##                        heat_fuel                          room_id
##                             1032                                0
##                     supply_area total_household_count_of_area_type
##                                0                                0
##                      room_count                   bathroom_count
##                               28                               28
##              front_door_structure           transaction_real_price
##                              888                                0
```

```
glimpse(school)
```

```
## Observations: 1,311
## Variables: 9
## $ school_id       <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, ...
## $ latitude        <dbl> 37.49088, 37.57778, 37.48137, 37.48574, 37.48081, 3...
## $ longitude       <dbl> 127.0151, 127.0029, 127.0591, 127.0580, 127.0519, 1...
## $ school_class    <fct> elementary, elementary, elementary, elementary, ele...
## $ operation_type  <fct> national, national, public, public, public, public,...
## $ highschool_type <fct> , , , , , , , , , , , , , , , , , , , , , , , , , ,
## $ gender          <fct> both, both, both, both, both, both, both, both, bot...
## $ foundation_date <fct> 1953.1.31, 1946.8.22, 1982.9.20, 1987.11.17, 1983.1...
## $ address_by_law  <dbl> 1165010800, 1111016800, 1168010300, 1168010300, 116...
```

```
colSums(is.na(school))
```

```
##       school_id        latitude       longitude    school_class  operation_type
##               0               0               0               0               0
## highschool_type          gender foundation_date  address_by_law
##               0               0               0               0
```

```
glimpse(subway)
```

```
## Observations: 283
## Variables: 5
## $ station_id     <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 1...
## $ latitude       <dbl> 37.55573, 37.56562, 37.57017, 37.57157, 37.57099, 37...
## $ longitude      <dbl> 126.9721, 126.9769, 126.9831, 126.9919, 127.0019, 12...
## $ subway_line    <fct> "1,4,KJ,AP", "1,2", "1", "1,3,5", "1", "1,4", "1,2,U...
## $ address_by_law <int> 1114012000, 1114016700, 1111012600, 1111015600, 1111...
```

```
colSums(is.na(subway))
```

```
##     station_id       latitude      longitude    subway_line address_by_law
##              0              0              0              0              8
```

train dataset NA

variable	# of NA
total_parking_capacity_in_site	3,623
tallest_building_in_sites	2
lowest_building_in_sites	2
heat_type	360
heat_fuel	1,032
room_count	28
bathroom_count	28
front_door_structure	888

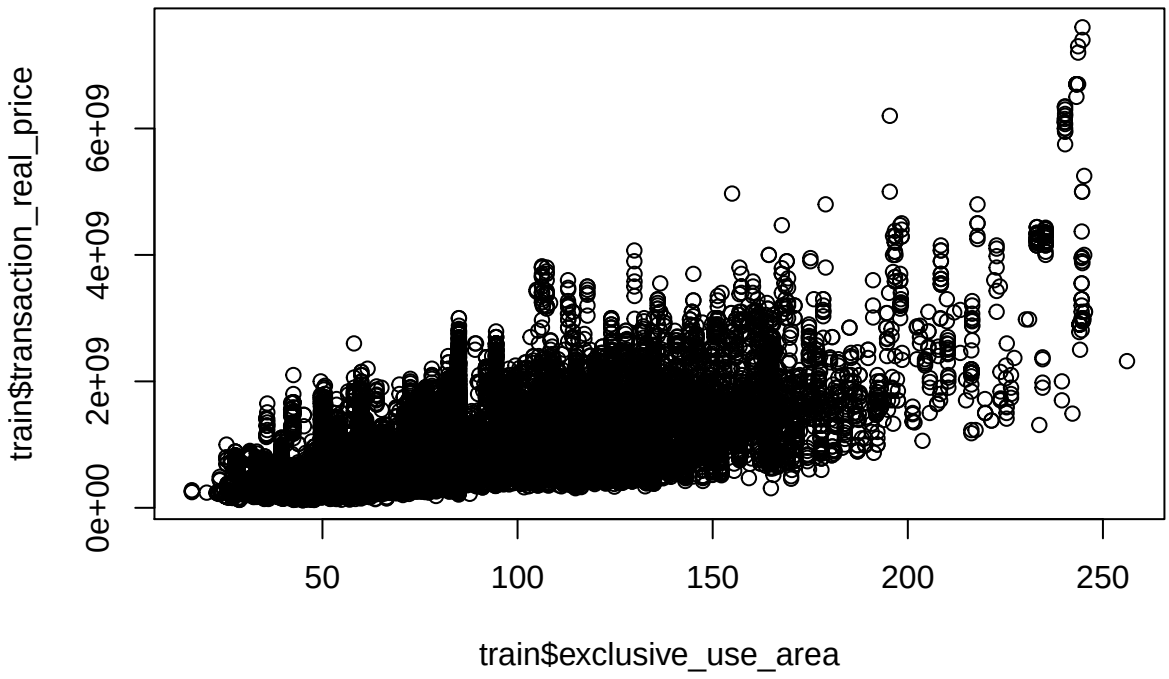
subway dataset NA

variable	# of NA
address_by_law	8

Response Variable

transaction_real_price

```
plot(train$exclusive_use_area, train$transaction_real_price)
```



```
train$price <- train$transaction_real_price/train$exclusive_use_area
```

아파트 실거래가는 공급면적에 따라 증가하는 추세를 보인다.
이것은 직관적으로도 당연한 현상으로, 집의 크기가 넓을수록 실거래가가 높은 것 당연하기 때문에
분석을 진행하기 위해서 평당 가격으로 계산하자.

Explanatory Variable

key

data 순서를 나타낼 뿐 의미가 없기 때문에 제거.

apartment_id

```
summary(train$apartment_id)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       3    1498    5394    9024   16156   50075
```

```
length(unique(train$apartment_id))
```

```
## [1] 2595
```

아파트 ID는 2595개가 존재한다.
너무 많아서 특정 아파트에 대한 선호도를 보기에는 복잡할 것 같다.
우선 빼기로 결정.

transaction_year_month

```
summary(train$transaction_year_month)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  201710  201712  201803  201781  201807  201809
```

```
train$transaction_year <- train$transaction_year_month %>% str_sub(1,4) %>% as.numeric()
train$transaction_month <- train$transaction_year_month %>% str_sub(5,6) %>% as.numeric()
```

거래 연도와 월로 나누자.
거래 연도는 아파트의 연식을 파악하기 위해서 아파트 준공 연도와 함께 사용.
거래 월로 가격과의 관계가 있는지 파악해보자.

transaction_data

```
summary(train$transaction_date)
```

```
##  1~10 11~20 21~28 21~30 21~31
## 27715 30754  2536  5744 23251
```

```
train$trans_date1 <- ifelse(train$transaction_date=='1~10',1,0)
train$trans_date2 <- ifelse(train$transaction_date=='11~20',1,0)
train$trans_date3 <- ifelse((train$transaction_date=='21~28')|(train$transaction_date=='21~30')|(train$
```

21~28, 21~30, 21~31 3개의 class를 하나로 봐도 무방할 것 같다.
월초, 중순. 월말 3개의 category로 나누어서 진행하자.

year_of_completion

```
train$age <- train$transaction_year - train$year_of_completion
```

준공연도 자체가 가지고 있는 의미보다 거래 당시 아파트가 얼마나 되었는지(연식)가 더 유의미할 것 같다.
위에서 구한 transaction_year와의 차이를 구해서 age 변수를 추가하자.

exclusive_use_area

아파트 거래가를 평당 가격으로 계산하기 때문에 전용면적과의 비례 관계를 제거했으므로 설명변수로서 그냥 사용하면 될 것 같다.

floor & tallest_building_in_sites & lowest_building_in_sites

```
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
mode.tallest <- getmode(train$tallest_building_in_sites)
mode.lowest <- getmode(train$lowest_building_in_sites)
train$tallest_building_in_sites[is.na(train$tallest_building_in_sites)] <- mode.tallest
train$lowest_building_in_sites[is.na(train$lowest_building_in_sites)] <- mode.lowest

train$tallest_building_in_sites <- as.numeric(train$tallest_building_in_sites)
train$lowest_building_in_sites <- as.numeric(train$lowest_building_in_sites)
train$mean.floor <- (train$tallest_building_in_sites + train$lowest_building_in_sites)/2
train$lowfloor <- ifelse(train$floor < 1/3 * train$mean.floor, 1, 0)
train$middlefloor <- ifelse(train$floor > 1/3 * train$mean.floor &
                            train$floor <= 2/3 * train$mean.floor, 1, 0)
train$highfloor <- ifelse(train$floor > 2/3 * train$mean.floor, 1, 0)
train$floor5 <- ifelse(train$tallest_building_in_sites<=5, 1, 0)
```

결측치가 2개씩 있는데 9만개를 유지하고 싶으니 대체하자.
skewed 정도가 크기 때문에 median이나 mode가 나을 것 같은데 mode로 가자.
절대적인 층수도, 상대적인 층수도 중요할 것 같다.
상대적인 층수는 최저/최고 평균에서 1/3씩을 이용하여 저,중,고로 나누자.
단지 내 최고층이 5층인 case를 기준으로 뭔가 차이가 나타난다.
5층 이상 이하에 대한 dummy variable도 추가해보자.

latitude & longitude

283개의 subway역과 1311개의 school의 경도, 위도를 이용하여 아파트 주변에 지하철역이나 학교가 있는지 여부에 대한 dummy variable을 만들어 사용하자.
소수점까지 계산하기에는 복잡하니 반올림을 해서 정리해보자.

address_by_law : 법정동코드

```
length(unique(train$address_by_law %>% str_sub(3,4)))
```

```
## [1] 25
```

```
length(unique(train$address_by_law))
```

```
## [1] 263
```

25개의 구와 263개의 동으로 나누어져 있다.
동으로 분석하기에는 복잡할 것 같으니 우선 구를 기준으로 나우어 분석을 진행하고
다른 구들과 차이를 보이는 구가 있다면 그 구에 대해서만 동으로 세부적으로 나누어서 분석을 진행해볼 예정.

total_household_count_in_sites

& total_household_count_of_area_type

& total_parking_capacity_in_site & apartment_building_count_in_sites

```
train2 <- train[is.na(train$total_parking_capacity_in_site) == F,]
corrplot.mixed(cor(train2[,c(11,12,13,20)]), lower='number', upper='ellipse')
```

total_parking_capacity_in_site에 NA가 너무 많다.
NA를 채울 수 있는, 혹은 total_parking_capacity_in_site를 대체할 만한 변수를 찾아보니
total_household_count_in_sites로 total_parking_capacity_in_site를 설명할 수 있을 것 같다.
이와 유사하게 apartment_building_count_in_sites도 설명할 수 있을 것 같다.
multicollinerity가 발생할 수 있으니 우선 total_household_count_in_sites만 사용해보자.

total_household_count_of_area_type는 별 연관성이 없어보이니 그대로 사용하자.

room_id

전용면적이나 공급면적이랑 유사한 정보를 가지고 있는 것으로 보인다.
id보다는 실제 면적으로 예측하는 것이 더 타당하다고 판단하여 면적 정보를 이용하는 것으로 하고 이 변수는 제거하자.

supply_area

```
cor(train$exclusive_use_area, train$supply_area)
```

```
## [1] 0.9829462
```

```
train$commom_area <- train$supply_area - train$exclusive_use_area
```

공급면적의 정의 상 전용면적 + 주거공용면적이기 때문에 전용면적과 공급면적의 correlation이 매우 높다.
하지만 주거공용면적도 무시할 수 없는 정보이기 때문에 common_area라는 변수를 추가하자.

room_count & bathroom_count

```
summary(train$room_count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   3.000   3.000   2.987   3.000   7.000      28
```

```
which(is.na(train$room_count))
```

```
##  [1]  9227  9228 15045 19180 20257 23598 24099 28305 31218 31287 31531 35916
## [13] 38136 38429 41545 42369 43225 45755 54252 55673 59157 60738 61678 66423
## [25] 69150 81649 84862 84863
```

```
summary(train$bathroom_count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   1.000   2.000   1.646   2.000   4.000      28
```

```
which(is.na(train$bathroom_count))
```

```
##  [1]  9227  9228 15045 19180 20257 23598 24099 28305 31218 31287 31531 35916
## [13] 38136 38429 41545 42369 43225 45755 54252 55673 59157 60738 61678 66423
## [25] 69150 81649 84862 84863
```

```
unique(train$exclusive_use_area[which(is.na(train$room_count))])
```

```
##  [1]  59.00  36.16  36.35  36.47  75.15  62.70  49.14  77.46  55.44  36.27
## [11]  41.64 101.88  86.55  60.81  84.60
```

```
train$room_count[train$exclusive_use_area == 59]
```

```
##  [1]  2 NA NA  2  2  2  2  2  2  2  2  2
```

```
train$room_count[is.na(train$room_count) & 36 < train$exclusive_use_area &
                   train$exclusive_use_area < 42] <- 1
train$room_count[is.na(train$room_count) & 49 < train$exclusive_use_area &
```

```
                  train$exclusive_use_area < 60] <- 2
train$room_count[is.na(train$room_count) & 60 < train$exclusive_use_area] <- 3

train$bathroom_count[is.na(train$bathroom_count) & train$exclusive_use_area < 60] <- 1
train$bathroom_count[is.na(train$bathroom_count) & 60 < train$exclusive_use_area] <- 32
```

동일한 아파트 단지들에 대해서 room_count와 bathroom_count에서 NA 발생한다.
동일한 혹은 비슷한 크기의 exclusive_use_area를 가진 곳의 room_count의 mode로 대체하자.

heat_type & heat_fuel

```
summary(train$heat_type)
```

```
##    central   district individual       NA's
##       8865      28183      52592        360
```

```
summary(train$heat_fuel)
```

```
## cogeneration          gas         NA's
##        28337        60631         1032
```

```
summary(train$heat_type[which(is.na(train$heat_fuel))])
```

```
##    central   district individual       NA's
##        229         37        406        360
```

```
summary(train$heat_fuel[train$heat_type == "individual"])
```

```
## cogeneration          gas         NA's
##          667        51519          766
```

```
summary(train$heat_fuel[train$heat_type == "central"])
```

```
## cogeneration          gas         NA's
##          413         8223          589
```

```
summary(train$heat_fuel[train$heat_type == "district"])
```

```
## cogeneration          gas         NA's
##        27257          889          397
```

```
unique(train$year_of_completion[which(is.na(train$heat_fuel) & is.na(train$heat_type))])
```

```
## [1] 2015 2016 2017 2018 1997
```

```
summary(train$heat_fuel[train$year_of_completion == 2015])
```

```
## cogeneration          gas         NA's
##         1589         1092          139
```

```
summary(train$heat_type[train$year_of_completion == 2015])
```

```
##    central   district individual       NA's
##         21       1568       1092        139
```

```
summary(train$heat_fuel[train$year_of_completion == 2016])
```

```
## cogeneration          gas         NA's
##          679         1091          111
```

```r
summary(train$heat_type[train$year_of_completion == 2016])
```

```
##    central   district individual       NA's
##          0        693       1091         97
```

```r
summary(train$heat_fuel[train$year_of_completion == 2017])
```

```
## cogeneration          gas         NA's
##           58          139          170
```

```r
summary(train$heat_type[train$year_of_completion == 2017])
```

```
##    central   district individual       NA's
##          0         72        210         85
```

```r
summary(train$heat_fuel[train$year_of_completion == 2018])
```

```
## cogeneration          gas         NA's
##            1           49           37
```

```r
summary(train$heat_type[train$year_of_completion == 2018])
```

```
##    central   district individual       NA's
##          0          1         49         37
```

```r
summary(train$heat_fuel[train$year_of_completion == 1997])
```

```
## cogeneration          gas         NA's
##          777         2209            2
```

```r
summary(train$heat_type[train$year_of_completion == 1997])
```

```
##    central   district individual       NA's
##        766        881       1339          2
```

```r
train$heat_fuel[is.na(train$heat_fuel) &
                        train$heat_type == "individual"] <- "gas"
train$heat_fuel[is.na(train$heat_fuel) &
                        train$heat_type == "district"] <- "cogeneration"
train$heat_fuel[is.na(train$heat_fuel) &
                        train$heat_type == "central"] <- "gas"

train$heat_fuel[is.na(train$heat_fuel) &
                        train$year_of_completion == "2015"] <- "cogeneration"
train$heat_type[is.na(train$heat_type) &
                        train$year_of_completion == "2015"] <- "district"
train$heat_fuel[is.na(train$heat_fuel) &
                        train$year_of_completion == "2016"] <- "gas"
train$heat_type[is.na(train$heat_type) &
                        train$year_of_completion == "2016"] <- "individual"
train$heat_fuel[is.na(train$heat_fuel) &
                        train$year_of_completion == "2017"] <- "gas"
train$heat_type[is.na(train$heat_type) &
                        train$year_of_completion == "2017"] <- "individual"
train$heat_fuel[is.na(train$heat_fuel) &
                        train$year_of_completion == "2018"] <- "gas"
train$heat_type[is.na(train$heat_type) &
                        train$year_of_completion == "2018"] <- "individual"
```

```
train$heat_fuel[is.na(train$heat_fuel) &
                        train$year_of_completion == "1997"] <- "gas"
train$heat_type[is.na(train$heat_type) &
                        train$year_of_completion == "1997"] <- "individual"
```

위의 결과를 이용해서 central을 가진 229개의 na는 gas로
district인 37개의 NA는 cogeneration으로
individual인 406개는 gas로 대체하자.
둘 다 NA인 경우에서는 관련된 변수로 연도를 선택하자.
2015년에는 cogeneration과 district, 2016년에는 gas와 individual이,
2017년에는 gas와 individual이, 2018년에는 gas와 individual이,
1997년에는 ags와 individual이 제일 많다.
정리하면, central - gas, district - cogenretaion, individual - gas로 매칭할 수 있을 같다.

front_door_structure

```
summary(train$front_door_structure)
```

```
## corridor    mixed stairway     NA's
##    24303     1307    63502      888
```

```
yr <- unique(train$year_of_completion[is.na(train$front_door_structure)])
summary(train$front_door_structure[train$year_of_completion == yr[21]])
```

```
## corridor    mixed stairway     NA's
##     1525       89     1896        1
```

```
unique(train$exclusive_use_area[is.na(train$front_door_structure) &
                                  train$year_of_completion == 1992])
```

```
## [1] 59.96 84.63 84.44
```

```
summary(train$exclusive_use_area[is.na(train$front_door_structure)])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.25   81.58   97.24  100.58  126.58  225.13
```

```
summary(train$front_door_structure[train$exclusive_use_area < 30])
```

```
## corridor    mixed stairway     NA's
##      177        0      217       17
```

```
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion >= 2000] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion <= 1988] <- "corridor"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion == 1999] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion == 1997] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion == 1993] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion == 1998] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                             train$year_of_completion == 1995] <- "stairway"
```

```
train$front_door_structure[is.na(train$front_door_structure) &
                           train$year_of_completion == 1996] <- "stairway"
train$front_door_structure[is.na(train$front_door_structure) &
                           train$year_of_completion == 1992] <- "corridor"
```

주어진 data의 복도식(corridor), 계단식(stairway), 복합식(mixed) 비율을 이용해서 결측치를 채워보자.
조사해보니 준공연도 혹은 전용 면적과 관련이 있을 것 같다.
동일한 준공연도의 최빈값으로 대체하고,
준공연도에 의한 구분이 유력하지 않은 경우 동일한 전용면적의 최빈값으로 대체하자.

##Final dataset

```
glimpse(train)
```

```
## Observations: 90,000
## Variables: 37
## $ key                                 <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1...
## $ apartment_id                        <int> 2944, 4307, 4307, 1337, 936, 936...
## $ transaction_year_month              <int> 201710, 201710, 201710, 201710, ...
## $ transaction_date                    <fct> 11~20, 11~20, 11~20, 11~20, 11~2...
## $ year_of_completion                  <int> 1993, 1995, 1995, 1989, 1983, 19...
## $ exclusive_use_area                  <dbl> 112.24, 59.52, 59.52, 35.10, 51....
## $ floor                               <int> 1, 4, 4, 10, 8, 6, 5, 5, 6, 21, ...
## $ latitude                            <dbl> 37.63981, 37.49666, 37.49666, 37...
## $ longitude                           <dbl> 127.0756, 126.8471, 126.8471, 12...
## $ address_by_law                      <int> 1135010400, 1153010800, 11530108...
## $ total_parking_capacity_in_site      <int> 666, 195, 195, NA, 360, 360, 674...
## $ total_household_count_in_sites      <int> 498, 218, 218, 1635, 360, 360, 1...
## $ apartment_building_count_in_sites   <int> 6, 1, 1, 9, 5, 5, 14, 14, 1, 2, ...
## $ tallest_building_in_sites           <dbl> 15, 22, 22, 15, 11, 11, 15, 15, ...
## $ lowest_building_in_sites            <dbl> 11, 21, 21, 15, 10, 10, 15, 15, ...
## $ heat_type                           <fct> individual, individual, individu...
## $ heat_fuel                           <fct> gas, gas, gas, gas, gas, gas, ga...
## $ room_id                             <int> 6316, 7677, 7678, 3643, 2441, 24...
## $ supply_area                         <dbl> 132.66, 80.63, 80.90, 46.57, 72....
## $ total_household_count_of_area_type  <int> 274, 1, 87, 420, 170, 170, 120, ...
## $ room_count                          <dbl> 4, 2, 2, 2, 2, 2, 3, 3, 3, 3, 4,...
## $ bathroom_count                      <dbl> 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2,...
## $ front_door_structure                <fct> stairway, corridor, corridor, co...
## $ transaction_real_price              <dbl> 5.350e+08, 2.400e+08, 2.400e+08,...
## $ price                               <dbl> 4766572, 4032258, 4032258, 47720...
## $ transaction_year                    <dbl> 2017, 2017, 2017, 2017, 2017, 20...
## $ transaction_month                   <dbl> 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ trans_date1                         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ trans_date2                         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ trans_date3                         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ age                                 <dbl> 24, 22, 22, 28, 34, 34, 32, 32, ...
## $ mean.floor                          <dbl> 13.0, 21.5, 21.5, 15.0, 10.5, 10...
## $ lowfloor                            <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ middlefloor                         <dbl> 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1,...
## $ highfloor                           <dbl> 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0,...
## $ floor5                              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ commom_area                         <dbl> 20.42, 21.11, 21.38, 11.47, 20.8...
```

```
train_final <- train[-c(1:5, 10:11, 13:15, 18:19, 24)]
glimpse(train_final)
```

```
## Observations: 90,000
## Variables: 24
## $ exclusive_use_area                <dbl> 112.24, 59.52, 59.52, 35.10, 51....
## $ floor                             <int> 1, 4, 4, 10, 8, 6, 5, 5, 6, 21, ...
## $ latitude                          <dbl> 37.63981, 37.49666, 37.49666, 37...
## $ longitude                         <dbl> 127.0756, 126.8471, 126.8471, 12...
## $ total_household_count_in_sites    <int> 498, 218, 218, 1635, 360, 360, 1...
## $ heat_type                         <fct> individual, individual, individu...
## $ heat_fuel                         <fct> gas, gas, gas, gas, gas, gas, ga...
## $ total_household_count_of_area_type <int> 274, 1, 87, 420, 170, 170, 120, ...
## $ room_count                        <dbl> 4, 2, 2, 2, 2, 2, 3, 3, 3, 3, 4,...
## $ bathroom_count                    <dbl> 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2,...
## $ front_door_structure              <fct> stairway, corridor, corridor, co...
## $ price                             <dbl> 4766572, 4032258, 4032258, 47720...
## $ transaction_year                  <dbl> 2017, 2017, 2017, 2017, 2017, 20...
## $ transaction_month                 <dbl> 10, 10, 10, 10, 10, 10, 10, 10, ...
## $ trans_date1                       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ trans_date2                       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ trans_date3                       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ age                               <dbl> 24, 22, 22, 28, 34, 34, 32, 32, ...
## $ mean.floor                        <dbl> 13.0, 21.5, 21.5, 15.0, 10.5, 10...
## $ lowfloor                          <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ middlefloor                       <dbl> 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1,...
## $ highfloor                         <dbl> 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0,...
## $ floor5                            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ commom_area                       <dbl> 20.42, 21.11, 21.38, 11.47, 20.8...
```

```
colSums(is.na(train_final))
```

```
##                 exclusive_use_area                              floor
##                                  0                                  0
##                           latitude                          longitude
##                                  0                                  0
##     total_household_count_in_sites                          heat_type
##                                  0                                  0
##                          heat_fuel total_household_count_of_area_type
##                                  0                                  0
##                         room_count                      bathroom_count
##                                  0                                  0
##               front_door_structure                              price
##                                  0                                  0
##                   transaction_year                  transaction_month
##                                  0                                  0
##                        trans_date1                        trans_date2
##                                  0                                  0
##                        trans_date3                                age
##                                  0                                  0
##                         mean.floor                           lowfloor
##                                  0                                  0
##                        middlefloor                          highfloor
##                                  0                                  0
```

```
##                    floor5                  commom_area
##                         0                            0
```