EDA, Regression Modeling and More with Seoul Officetel Rentals Data (2011-2021)

Introduction

- to be added

Preprocessing

Set Hangeul Font, 한글 폰트 설정

- For plotting purposes
- matplotlib. `rc("font",family=font_name)`

In [1]:
```
import matplotlib as mpl
import matplotlib.font_manager as fm

# Nanum Gothic Coding
# font_path= r"C:\tmp\NanumGothicCoding-Bold.ttf"

# D2Coding
font_path= r"C:\tmp\D2CodingBold-Ver1.3.2-20180524.ttf".replace("\\","/")

font_name= fm.FontProperties(fname=font_path).get_name() # D2Coding

mpl.rc("font",family=font_name)
```

In [2]:
```
import pandas as pd

path= "./data/"
#csv_2021= "seoul_rental_2021.csv"
csv_2020= "seoul_rental_2020.csv"
csv_2019= "seoul_rental_2019.csv"
csv_2018= "seoul_rental_2018.csv"
csv_2017= "seoul_rental_2017.csv"
csv_2016= "seoul_rental_2016.txt"
csv_2015= "seoul_rental_2015.txt"
csv_2014= "seoul_rental_2014.txt"
csv_2014_clean= "seoul_rental_2014_clean.txt"
csv_2013= "seoul_rental_2013.txt"
csv_2012= "seoul_rental_2012.txt"
csv_2011= "seoul_rental_2011.txt"

# df_2021= pd.read_csv(path+csv_2021,encoding="cp949")
# df_2021.shape
df_2020= pd.read_csv(path+csv_2020,encoding="cp949")
df_2019= pd.read_csv(path+csv_2019,encoding="cp949")
df_2018= pd.read_csv(path+csv_2018,encoding="cp949")
df_2017= pd.read_csv(path+csv_2017,encoding="cp949")
df_2016= pd.read_csv(path+csv_2016,encoding="utf-8")
df_2015= pd.read_csv(path+csv_2015,encoding="utf-8")
df_2014= pd.read_csv(path+csv_2014,encoding="utf-8")
df_2013= pd.read_csv(path+csv_2013,encoding="utf-8")
df_2012= pd.read_csv(path+csv_2012,encoding="utf-8")
df_2011= pd.read_csv(path+csv_2011,encoding="utf-8")
```

Merge 10-year records into one dataframe

- Check the shape of all the dataframes

In [3]:
```
df_list= [df_2020,df_2019,df_2018,df_2017,df_2016,df_2015,df_2014,df_2013, df_2012,\
          df_2011]
for i,df in enumerate(df_list):
    year=2020-i
    print(year,":",df.shape)
```

```
2020 : (49971, 14)
2019 : (48289, 14)
2018 : (40030, 14)
2017 : (34674, 14)
2016 : (27592, 14)
2015 : (24205, 14)
2014 : (20820, 16)
2013 : (16209, 14)
2012 : (12529, 14)
2011 : (10466, 14)
```

- The two unnamed columns are from the year 2014.
- tabulation of 1909 rows incorrect; has to be manually adjusted

In [4]:
```
df_2014.isna().sum()
```

Out[4]:
```
시군구              0
번지              16
본번               0
부번               0
단지명           1909
전월세구분          0
전용면적(㎡)     1909
계약년월            0
계약일             0
보증금(만원)        0
월세(만원)          0
층               0
건축년도            1
도로명             0
Unnamed: 14    18911
Unnamed: 15    18911
dtype: int64
```

In [5]:
```
df_2014_clean= pd.read_csv(path+csv_2014_clean,encoding="utf-8")
df_2014_clean.shape
```

Out[5]:
```
(20820, 14)
```

In [40]:
```
df_list= [df_2020,df_2019,df_2018,df_2017,df_2016,df_2015,df_2014_clean,df_2013,\
          df_2012,df_2011]
df_backup= pd.concat(df_list,ignore_index=True)
df= df_backup.copy()
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284785 entries, 0 to 284784
Data columns (total 14 columns):
 #   Column    Non-Null Count   Dtype
---  ------    --------------   -----
 0   시군구       284785 non-null  object
 1   번지        283949 non-null  object
 2   본번        284785 non-null  int64
 3   부번        284785 non-null  int64
 4   단지명       284785 non-null  object
 5   전월세구분     284785 non-null  object
 6   전용면적(㎡)   284785 non-null  float64
 7   계약년월      284785 non-null  int64
 8   계약일       284785 non-null  int64
 9   보증금(만원)   284785 non-null  object
 10  월세(만원)    284785 non-null  int64
 11  층        284785 non-null  int64
 12  건축년도      276309 non-null  float64
 13  도로명       284770 non-null  object
dtypes: float64(2), int64(6), object(6)
memory usage: 30.4+ MB
```

In [13]:
```
df.head(1)
```

Out[13]:

	시군구	번지	본번	부번	단지명	전월세구분	전용면적(㎡)	계약년월	계약일	보증금(만원)	월세(만원)	층	건축년도	도로명
0	서울특별시 강남구 개포동	1237-3	1237	3	(1237-3)	월세	29.51	202004	11	500	100	2	2020.0	논현로20길 12

Columns to be merged/dropped

- 번지 (lot number)
- 본번 (primary lot number)
- 부번 (secondary lot number)
- 단지명 (building/estate name)
- 도로명 (street address)

The street address is the only address that is legally valid in South Korea since the Road Name Address Act came fully into effect on January 1, 2014. The estate name has additional information and will be merged with the street name. The empty cells of the street address column will be filled the lot number and/or the estate name. The lot number is made up of a primary number hyphenated with a secondary number, e.g., 1237-3.

KR

> 도로명주소법이 전면적으로 시행되면서 2014년 1월 1일부터는 토지대장을 제외한 모든 곳에 도로명주소만을 쓸 수 있다. 따라서 도로명 주소와 단지명을 합쳐 각 건물의 전체 주소를 표시하되 도로명 주소 또는 단지명 컬럼이 비어 있으면 번지를 사용한다.

Rename data columns

- 시군구 → district1
- 번지 → lot_num
- 본번 → lot_num_primary
- 부번 → lot_num_secondary
- 단지명 → estate_name
- 전월세구분 → rent_type (lump-sum or monthly)
- 전용면적(㎡) → unit_size (m^2)
- 계약년월 → sign_yrmon
- 계약일 → sign_day
- 보증금(만원) → deposit (in 10,000 won)
- 월세(만원) → rent_price (in 10,000 won)
- 층 → floor
- 건축년도 → yr_built
- 도로명 → str_addr

In [41]:
```
cols= ["district1","lot_num","lot_num_primary","lot_num_secondary","estate_name",\
       "rent_type","unit_size","sign_yymm","sign_dd","deposit","rent_price",\
       "floor","yr_built","str_addr"]

df.columns= cols
df.head(1)
```

Out[41]:

	district1	lot_num	lot_num_primary	lot_num_secondary	estate_name	rent_type	unit_size	sign_yymm	sigr
0	서울특별시 강남구 개포동	1237-3	1237	3	(1237-3)	월세	29.51	202004	

In [9]:
```
df.isna().sum()
```

Out[9]:
```
district1               0
lot_num               836
lot_num_primary         0
lot_num_secondary       0
estate_name             0
rent_type               0
unit_size               0
sign_yymm               0
sign_dd                 0
deposit                 0
rent_price              0
floor                   0
yr_built             8476
str_addr               15
dtype: int64
```

In [42]:
```
import numpy as np
nan_index= np.where(df.str_addr.isna())
nan_index
```

Out[42]:
```
(array([238349, 238350, 238351, 238352, 238353, 238354, 238355, 238356,
        238357, 238358, 238359, 238360, 238361, 238362, 238363],
       dtype=int64),)
```

In [19]:
```
nan_index[0]#.flatten()
```

Out[19]:
```
array([238349, 238350, 238351, 238352, 238353, 238354, 238355, 238356,
       238357, 238358, 238359, 238360, 238361, 238362, 238363],
      dtype=int64)
```

Merge `str_addr` and `estate_name`

- into new column `street_addr`, and
- drop the two columns

In [38]:
```
del df["street_addr"]
```

In [43]:
```
import numpy as np
df["estate_name"]= df["estate_name"].astype(str)
df["str_addr"]= df.str_addr.astype(str)
str_addr_series= [row["str_addr"].replace("nan","")+row["estate_name"] \
                  if row["str_addr"]=="nan" else row["str_addr"]+", "+\
                  row["estate_name"] for i,row in df.iterrows()]
df.insert(0,"street_addr",str_addr_series)
#df["str_addr"].replace(np.NaN,"",regex=True) + ", "+ df["estate_name"]
#df
```

In [24]:
```
df.isna().sum()
```

Out[24]:
```
district1               0
lot_num               836
lot_num_primary         0
lot_num_secondary       0
estate_name             0
rent_type               0
unit_size               0
sign_yymm               0
sign_dd                 0
deposit                 0
rent_price              0
floor                   0
yr_built             8476
str_addr                0
street_addr             0
dtype: int64
```

In [37]:
```
# df.iloc[nan_index[0]]
```

Drop unused columns

- lot_num
- lot_num_primary
- lot_num_secondary
- estate_name
- str_addr

In [44]:
```
df.drop(["lot_num","lot_num_primary","lot_num_secondary","estate_name","str_addr"],\
        axis=1,inplace=True)
df.head(1)
```

Out[44]:

	street_addr	district1	rent_type	unit_size	sign_yymm	sign_dd	deposit	rent_price	floor	yr_built
0	논현로20길 12, (1237-3)	서울특별시 강남구 개포동	월세	29.51	202004	11	500	100	2	2020.0

New column `district`

KR

- 전체 데이터가 서울 지역에 한정되어 있으므로 "서울특별시", 동 이름 제거

In [45]:
```
df.insert(0,"district",[val.split()[1] for i,val in df.district1.iteritems() ])
df.head(2)
```

Out[45]:

	district	street_addr	district1	rent_type	unit_size	sign_yymm	sign_dd	deposit	rent_price	floor	yr_built
0	강남구	논현로20길 12, (1237-3)	서울특별시 강남구 개포동	월세	29.51	202004	11	500	100	2	2020.0
1	강남구	논현로20길 12, (1237-3)	서울특별시 강남구 개포동	월세	29.95	202005	30	3000	80	4	2020.0

In [50]:
```
df.insert(2,"district_sub",[f"{val.split()[2]}" for i,val \
                            in df.district1.iteritems()])
df.head(1)
```

Out[50]:

	district	street_addr	district_sub	district1	rent_type	unit_size	sign_yymm	sign_dd	deposit	rent_price	f
0	강남구	논현로20길 12, (1237-3)	개포동	서울특별시 강남구 개포동	월세	29.51	202004	11	500	100	

Drop columns

- `district1`

In [51]:
```
# drop "district1" column
df.drop("district1",axis=1,inplace=True)
df.head(1)
```

Out[51]:

	district	street_addr	district_sub	rent_type	unit_size	sign_yymm	sign_dd	deposit	rent_price	floor	yr_b
0	강남구	논현로20길 12, (1237-3)	개포동	월세	29.51	202004	11	500	100	2	202