capstone-real-estate

October 4, 2023

[1]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

[2]:
```python
df_train = pd.read_csv ("train.csv")
```

[3]:
```python
df_test = pd.read_csv ("test.csv")
```

[4]:
```python
df_train.shape
```

[4]:
```
(27321, 80)
```

[5]:
```python
df_test.shape
```

[5]:
```
(11709, 80)
```

[6]:
```python
df_train.head()
```

[6]:
```
      UID  BLOCKID  SUMLEVEL  COUNTYID  STATEID        state state_ab  \
0  267822      NaN       140        53       36     New York       NY
1  246444      NaN       140       141       18      Indiana       IN
2  245683      NaN       140        63       18      Indiana       IN
3  279653      NaN       140       127       72  Puerto Rico       PR
4  247218      NaN       140       161       20       Kansas       KS

         city             place   type  … female_age_mean  female_age_median  \
0    Hamilton          Hamilton   City  …        44.48629           45.33333
1  South Bend          Roseland   City  …        36.48391           37.58333
2    Danville          Danville   City  …        42.15810           42.83333
3    San Juan          Guaynabo  Urban  …        47.77526           50.58333
4   Manhattan  Manhattan City   City  …        24.17693           21.58333

   female_age_stdev  female_age_sample_weight  female_age_samples  pct_own  \
0          22.51276                 685.33845              2618.0  0.79046
```

```
1          23.43353                    267.23367                   1284.0  0.52483
2          23.94119                    707.01963                   3238.0  0.85331
3          24.32015                    362.20193                   1559.0  0.65037
4          11.10484                   1854.48652                   3051.0  0.13046

   married  married_snp  separated  divorced
0  0.57851      0.01882    0.01240   0.08770
1  0.34886      0.01426    0.01426   0.09030
2  0.64745      0.02830    0.01607   0.10657
3  0.47257      0.02021    0.02021   0.10106
4  0.12356      0.00000    0.00000   0.03109

[5 rows x 80 columns]
```

[7]:

```
df_train.columns
```

[7]:

```
Index(['UID', 'BLOCKID', 'SUMLEVEL', 'COUNTYID', 'STATEID', 'state',
       'state_ab', 'city', 'place', 'type', 'primary', 'zip_code', 'area_code',
       'lat', 'lng', 'ALand', 'AWater', 'pop', 'male_pop', 'female_pop',
       'rent_mean', 'rent_median', 'rent_stdev', 'rent_sample_weight',
       'rent_samples', 'rent_gt_10', 'rent_gt_15', 'rent_gt_20', 'rent_gt_25',
       'rent_gt_30', 'rent_gt_35', 'rent_gt_40', 'rent_gt_50',
       'universe_samples', 'used_samples', 'hi_mean', 'hi_median', 'hi_stdev',
       'hi_sample_weight', 'hi_samples', 'family_mean', 'family_median',
       'family_stdev', 'family_sample_weight', 'family_samples',
       'hc_mortgage_mean', 'hc_mortgage_median', 'hc_mortgage_stdev',
       'hc_mortgage_sample_weight', 'hc_mortgage_samples', 'hc_mean',
       'hc_median', 'hc_stdev', 'hc_samples', 'hc_sample_weight',
       'home_equity_second_mortgage', 'second_mortgage', 'home_equity', 'debt',
       'second_mortgage_cdf', 'home_equity_cdf', 'debt_cdf', 'hs_degree',
       'hs_degree_male', 'hs_degree_female', 'male_age_mean',
       'male_age_median', 'male_age_stdev', 'male_age_sample_weight',
       'male_age_samples', 'female_age_mean', 'female_age_median',
       'female_age_stdev', 'female_age_sample_weight', 'female_age_samples',
       'pct_own', 'married', 'married_snp', 'separated', 'divorced'],
      dtype='object')
```

[8]:

```
df_train.describe()
```

[8]:

```
                 UID  BLOCKID  SUMLEVEL      COUNTYID       STATEID  \
count   27321.000000      0.0   27321.0  27321.000000  27321.000000
mean   257331.996303      NaN     140.0     85.646426     28.271806
std     21343.859725      NaN       0.0     98.333097     16.392846
min    220342.000000      NaN     140.0      1.000000      1.000000
25%    238816.000000      NaN     140.0     29.000000     13.000000
50%    257220.000000      NaN     140.0     63.000000     28.000000
75%    275818.000000      NaN     140.0    109.000000     42.000000
```

```
max    294334.000000      NaN     140.0    840.000000     72.000000

           zip_code     area_code           lat           lng         ALand  \
count  27321.000000  27321.000000  27321.000000  27321.000000  2.732100e+04
mean   50081.999524    596.507668     37.508813    -91.288394  1.295106e+08
std    29558.115660    232.497482      5.588268     16.343816  1.275531e+09
min      602.000000    201.000000     17.929085   -165.453872  4.113400e+04
25%    26554.000000    405.000000     33.899064    -97.816067  1.799408e+06
50%    47715.000000    614.000000     38.755183    -86.554374  4.866940e+06
75%    77093.000000    801.000000     41.380606    -79.782503  3.359820e+07
max    99925.000000    989.000000     67.074018    -65.379332  1.039510e+11

       …  female_age_mean  female_age_median  female_age_stdev  \
count  …     27115.000000       27115.000000      27115.000000
mean   …        40.319803          40.355099         22.178745
std    …         5.886317           8.039585          2.540257
min    …        16.008330          13.250000          0.556780
25%    …        36.892050          34.916670         21.312135
50%    …        40.373320          40.583330         22.514410
75%    …        43.567120          45.416670         23.575260
max    …        79.837390          82.250000         30.241270

       female_age_sample_weight  female_age_samples       pct_own  \
count              27115.000000        27115.000000  27053.000000
mean                 544.238432         2208.761903      0.640434
std                  283.546896         1089.316999      0.226640
min                    0.664700            2.000000      0.000000
25%                  355.995825         1471.000000      0.502780
50%                  503.643890         2066.000000      0.690840
75%                  680.275055         2772.000000      0.817460
max                 6197.995200        27250.000000      1.000000

            married   married_snp     separated      divorced
count  27130.000000  27130.000000  27130.000000  27130.000000
mean       0.508300      0.047537      0.019089      0.100248
std        0.136860      0.037640      0.020796      0.049055
min        0.000000      0.000000      0.000000      0.000000
25%        0.425102      0.020810      0.004530      0.065800
50%        0.526665      0.038840      0.013460      0.095205
75%        0.605760      0.065100      0.027487      0.129000
max        1.000000      0.714290      0.714290      1.000000

[8 rows x 74 columns]
```

[9]:
```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 27321 entries, 0 to 27320
Data columns (total 80 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   UID                          27321 non-null  int64
 1   BLOCKID                      0 non-null      float64
 2   SUMLEVEL                     27321 non-null  int64
 3   COUNTYID                     27321 non-null  int64
 4   STATEID                      27321 non-null  int64
 5   state                        27321 non-null  object
 6   state_ab                     27321 non-null  object
 7   city                         27321 non-null  object
 8   place                        27321 non-null  object
 9   type                         27321 non-null  object
 10  primary                      27321 non-null  object
 11  zip_code                     27321 non-null  int64
 12  area_code                    27321 non-null  int64
 13  lat                          27321 non-null  float64
 14  lng                          27321 non-null  float64
 15  ALand                        27321 non-null  float64
 16  AWater                       27321 non-null  int64
 17  pop                          27321 non-null  int64
 18  male_pop                     27321 non-null  int64
 19  female_pop                   27321 non-null  int64
 20  rent_mean                    27007 non-null  float64
 21  rent_median                  27007 non-null  float64
 22  rent_stdev                   27007 non-null  float64
 23  rent_sample_weight           27007 non-null  float64
 24  rent_samples                 27007 non-null  float64
 25  rent_gt_10                   27007 non-null  float64
 26  rent_gt_15                   27007 non-null  float64
 27  rent_gt_20                   27007 non-null  float64
 28  rent_gt_25                   27007 non-null  float64
 29  rent_gt_30                   27007 non-null  float64
 30  rent_gt_35                   27007 non-null  float64
 31  rent_gt_40                   27007 non-null  float64
 32  rent_gt_50                   27007 non-null  float64
 33  universe_samples             27321 non-null  int64
 34  used_samples                 27321 non-null  int64
 35  hi_mean                      27053 non-null  float64
 36  hi_median                    27053 non-null  float64
 37  hi_stdev                     27053 non-null  float64
 38  hi_sample_weight             27053 non-null  float64
 39  hi_samples                   27053 non-null  float64
 40  family_mean                  27023 non-null  float64
 41  family_median                27023 non-null  float64
 42  family_stdev                 27023 non-null  float64
 43  family_sample_weight         27023 non-null  float64
```

```
 44  family_samples              27023 non-null  float64
 45  hc_mortgage_mean            26748 non-null  float64
 46  hc_mortgage_median          26748 non-null  float64
 47  hc_mortgage_stdev           26748 non-null  float64
 48  hc_mortgage_sample_weight   26748 non-null  float64
 49  hc_mortgage_samples         26748 non-null  float64
 50  hc_mean                     26721 non-null  float64
 51  hc_median                   26721 non-null  float64
 52  hc_stdev                    26721 non-null  float64
 53  hc_samples                  26721 non-null  float64
 54  hc_sample_weight            26721 non-null  float64
 55  home_equity_second_mortgage 26864 non-null  float64
 56  second_mortgage             26864 non-null  float64
 57  home_equity                 26864 non-null  float64
 58  debt                        26864 non-null  float64
 59  second_mortgage_cdf         26864 non-null  float64
 60  home_equity_cdf             26864 non-null  float64
 61  debt_cdf                    26864 non-null  float64
 62  hs_degree                   27131 non-null  float64
 63  hs_degree_male              27121 non-null  float64
 64  hs_degree_female            27098 non-null  float64
 65  male_age_mean               27132 non-null  float64
 66  male_age_median             27132 non-null  float64
 67  male_age_stdev              27132 non-null  float64
 68  male_age_sample_weight      27132 non-null  float64
 69  male_age_samples            27132 non-null  float64
 70  female_age_mean             27115 non-null  float64
 71  female_age_median           27115 non-null  float64
 72  female_age_stdev            27115 non-null  float64
 73  female_age_sample_weight    27115 non-null  float64
 74  female_age_samples          27115 non-null  float64
 75  pct_own                     27053 non-null  float64
 76  married                     27130 non-null  float64
 77  married_snp                 27130 non-null  float64
 78  separated                   27130 non-null  float64
 79  divorced                    27130 non-null  float64
dtypes: float64(62), int64(12), object(6)
memory usage: 16.7+ MB
```

[10]:
```
df_train.set_index(keys=["UID"],inplace=True)
```

[11]:
```
df_test.set_index(keys=["UID"],inplace=True)
```

[12]:
```
df_train.isnull().sum().any()
```

[12]:
```
True
```

[13]:
```
df_test.isnull().sum().any()
```

[13]:
```
True
```

[14]:
```
df_train.isnull().sum()[df_train.isnull().sum()>0]
```

[14]:
```
BLOCKID                        27321
rent_mean                        314
rent_median                      314
rent_stdev                       314
rent_sample_weight               314
rent_samples                     314
rent_gt_10                       314
rent_gt_15                       314
rent_gt_20                       314
rent_gt_25                       314
rent_gt_30                       314
rent_gt_35                       314
rent_gt_40                       314
rent_gt_50                       314
hi_mean                          268
hi_median                        268
hi_stdev                         268
hi_sample_weight                 268
hi_samples                       268
family_mean                      298
family_median                    298
family_stdev                     298
family_sample_weight             298
family_samples                   298
hc_mortgage_mean                 573
hc_mortgage_median               573
hc_mortgage_stdev                573
hc_mortgage_sample_weight        573
hc_mortgage_samples              573
hc_mean                          600
hc_median                        600
hc_stdev                         600
hc_samples                       600
hc_sample_weight                 600
home_equity_second_mortgage      457
second_mortgage                  457
home_equity                      457
debt                             457
second_mortgage_cdf              457
home_equity_cdf                  457
debt_cdf                         457
```

```
hs_degree                          190
hs_degree_male                     200
hs_degree_female                   223
male_age_mean                      189
male_age_median                    189
male_age_stdev                     189
male_age_sample_weight             189
male_age_samples                   189
female_age_mean                    206
female_age_median                  206
female_age_stdev                   206
female_age_sample_weight           206
female_age_samples                 206
pct_own                            268
married                            191
married_snp                        191
separated                          191
divorced                           191
dtype: int64
```

[15]:
```
df_test.isnull().sum()[df_test.isnull().sum()>0].shape
```

[15]:
```
(59,)
```

[16]:
```
df_test.isnull().sum()[df_test.isnull().sum()>0]
```

[16]:
```
BLOCKID                          11709
rent_mean                          148
rent_median                        148
rent_stdev                         148
rent_sample_weight                 148
rent_samples                       148
rent_gt_10                         149
rent_gt_15                         149
rent_gt_20                         149
rent_gt_25                         149
rent_gt_30                         149
rent_gt_35                         149
rent_gt_40                         149
rent_gt_50                         149
hi_mean                            122
hi_median                          122
hi_stdev                           122
hi_sample_weight                   122
hi_samples                         122
family_mean                        136
family_median                      136
```

```
family_stdev                   136
family_sample_weight           136
family_samples                 136
hc_mortgage_mean               268
hc_mortgage_median             268
hc_mortgage_stdev              268
hc_mortgage_sample_weight      268
hc_mortgage_samples            268
hc_mean                        290
hc_median                      290
hc_stdev                       290
hc_samples                     290
hc_sample_weight               290
home_equity_second_mortgage    220
second_mortgage                220
home_equity                    220
debt                           220
second_mortgage_cdf            220
home_equity_cdf                220
debt_cdf                       220
hs_degree                       85
hs_degree_male                  89
hs_degree_female               105
male_age_mean                   84
male_age_median                 84
male_age_stdev                  84
male_age_sample_weight          84
male_age_samples                84
female_age_mean                 96
female_age_median               96
female_age_stdev                96
female_age_sample_weight        96
female_age_samples              96
pct_own                        122
married                         84
married_snp                     84
separated                       84
divorced                        84
dtype: int64
```

[17]:
```
df_test.isnull().sum()[df_test.isnull().sum()>0].shape
```

[17]:
```
(59,)
```

[18]:
```
percent_train=df_train.isnull().sum()/len(df_train)*100
df_percent_train=pd.DataFrame(percent_train,columns=["Percentage of missing␣
↪values"])
```

[19]:
```
df_percent_train.sort_values(by=["Percentage of missing␣
↪values"],inplace=True,ascending=False)
```

[20]:
```
df_percent_train
```

[20]:
```
             Percentage of missing values
BLOCKID                        100.000000
hc_samples                       2.196113
hc_mean                          2.196113
hc_median                        2.196113
hc_stdev                         2.196113
…                                     …
state                            0.000000
zip_code                         0.000000
city                             0.000000
place                            0.000000
state_ab                         0.000000

[79 rows x 1 columns]
```

[21]:
```
percent_test=df_test.isnull().sum()/len(df_test)*100
df_percent_test=pd.DataFrame(percent_test,columns=["Percentage of missing␣
↪values"])
```

[22]:
```
df_percent_test.sort_values(by=["Percentage of missing␣
↪values"],inplace=True,ascending=False)
```

[23]:
```
df_percent_test
```

[23]:
```
             Percentage of missing values
BLOCKID                        100.000000
hc_samples                       2.476727
hc_mean                          2.476727
hc_median                        2.476727
hc_stdev                         2.476727
…                                     …
type                             0.000000
place                            0.000000
city                             0.000000
state                            0.000000
state_ab                         0.000000

[79 rows x 1 columns]
```

[24]:
```
df_train.drop(columns=['BLOCKID','SUMLEVEL'],inplace=True)
```

[25]:
```
df_test.drop(columns=['BLOCKID','SUMLEVEL'],inplace=True)
```

[26]:
```
missing_values_train=[]
for col in df_train.columns:
    if df_train[col].isnull().sum()!=0:
        missing_values_train.append(col)
```

[27]:
```
missing_values_test=[]
for col in df_test.columns:
    if df_test[col].isnull().sum()!=0:
        missing_values_test.append(col)
```

[28]:
```
for col in df_train.columns:
    if col in (missing_values_train):
        df_train[col].replace(np.nan,df_train[col].mean(),inplace=True)
```

[29]:
```
for col in df_test.columns:
    if col in (missing_values_test):
        df_test[col].replace(np.nan,df_test[col].mean(),inplace=True)
```

[30]:
```
df_train.isnull().sum().any()
```

[30]:
```
False
```

[31]:
```
df_test.isnull().sum().any()
```

[31]:
```
False
```

[32]:
```
pip install pandasql
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pandasql in ./.local/lib/python3.7/site-packages
(0.7.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/site-packages
(from pandasql) (1.21.5)
Requirement already satisfied: sqlalchemy in /usr/local/lib/python3.7/site-
packages (from pandasql) (1.3.15)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/site-packages
(from pandasql) (1.1.5)
Requirement already satisfied: python-dateutil>=2.7.3 in
/usr/local/lib/python3.7/site-packages (from pandas->pandasql) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/site-
packages (from pandas->pandasql) (2019.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-
packages (from python-dateutil>=2.7.3->pandas->pandasql) (1.14.0)
```

```
WARNING: You are using pip version 22.0.3; however, version 23.1.2 is
available.
You should consider upgrading via the '/usr/local/bin/python3.7 -m pip install
--upgrade pip' command.
Note: you may need to restart the kernel to use updated packages.
```

[33]:
```
from pandasql import sqldf
```

[34]:
```
q1="select place,pct_own,second_mortgage,lat,lng from df_train where pct_own>0.
↪10 and second_mortgage<0.5 order by second_mortgage DESC LIMIT 2500;"
```

[35]:
```
Query_fun=lambda q:sqldf(q,globals())
df_train_location=Query_fun(q1)
```

[36]:
```
df_train_location
```

[36]:
```
                  place  pct_own  second_mortgage        lat         lng
0        Worcester City  0.20247          0.43363  42.254262  -71.800347
1          Harbor Hills  0.15618          0.31818  40.751809  -73.853582
2           Glen Burnie  0.22380          0.30212  39.127273  -76.635265
3       Egypt Lake-leto  0.11618          0.28972  28.029063  -82.495395
4           Lincolnwood  0.14228          0.28899  41.967289  -87.652434
…                   …        …                …          …           …
2495     Marina Del Rey  0.44682          0.06818  33.983203 -118.466139
2496       Raleigh City  0.12827          0.06818  35.757135  -78.704288
2497           Lochearn  0.84707          0.06815  39.353095  -76.733315
2498       Manteca City  0.67116          0.06814  37.732143 -121.242902
2499  Philadelphia City  0.70507          0.06814  40.039070  -75.125135

[2500 rows x 5 columns]
```

[37]:
```
df_train['bad_debt']=df_train['second_mortgage']+df_train['home_equity']-df_train['home_equity
```

[38]:
```
df_train['bad_debt']
```

[38]:
```
UID
267822    0.09408
246444    0.04274
245683    0.09512
279653    0.01086
247218    0.05426
            …
279212    0.00000
277856    0.20908
233000    0.07857
```

```
287425    0.14305
265371    0.18362
Name: bad_debt, Length: 27321, dtype: float64
```

[39]:
```
df_train['city']
```

[39]:
```
UID
267822       Hamilton
246444     South Bend
245683       Danville
279653       San Juan
247218      Manhattan
             …
279212          Coamo
277856      Blue Bell
233000        Weldona
287425    Colleyville
265371      Las Vegas
Name: city, Length: 27321, dtype: object
```

[40]:
```
df_ham=df_train.loc[df_train['city']=='Hamilton']
df_Man=df_train.loc[df_train['city']=='Manhattan']
```

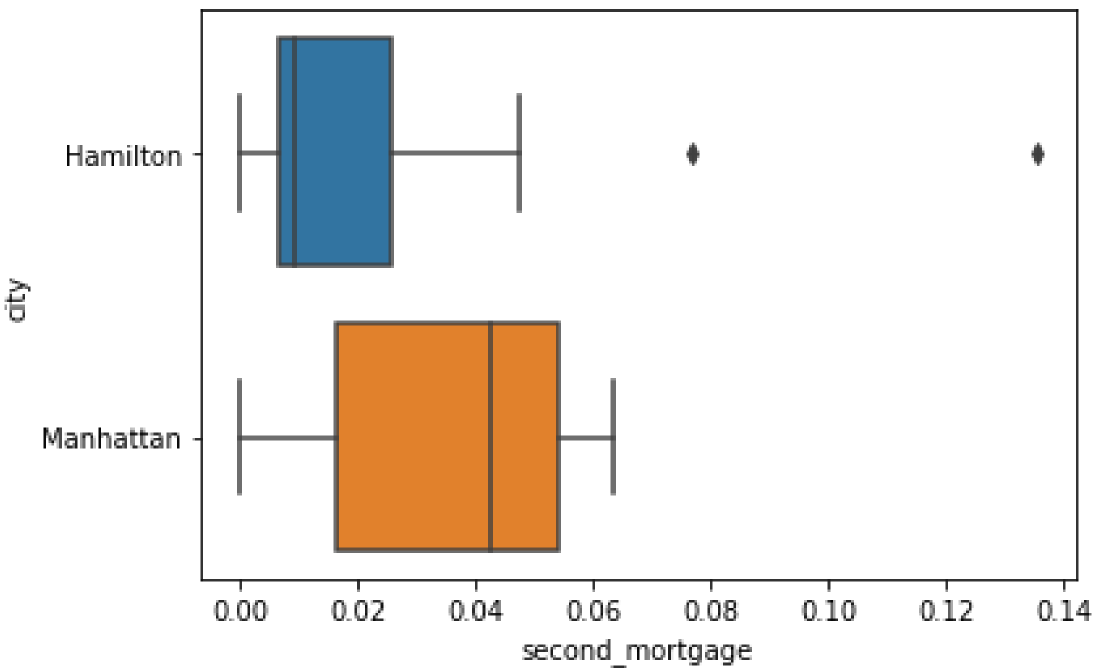
[41]:
```
df_box_city=pd.concat([df_ham,df_Man])
```

[42]:
```
df_box_city.head()
```

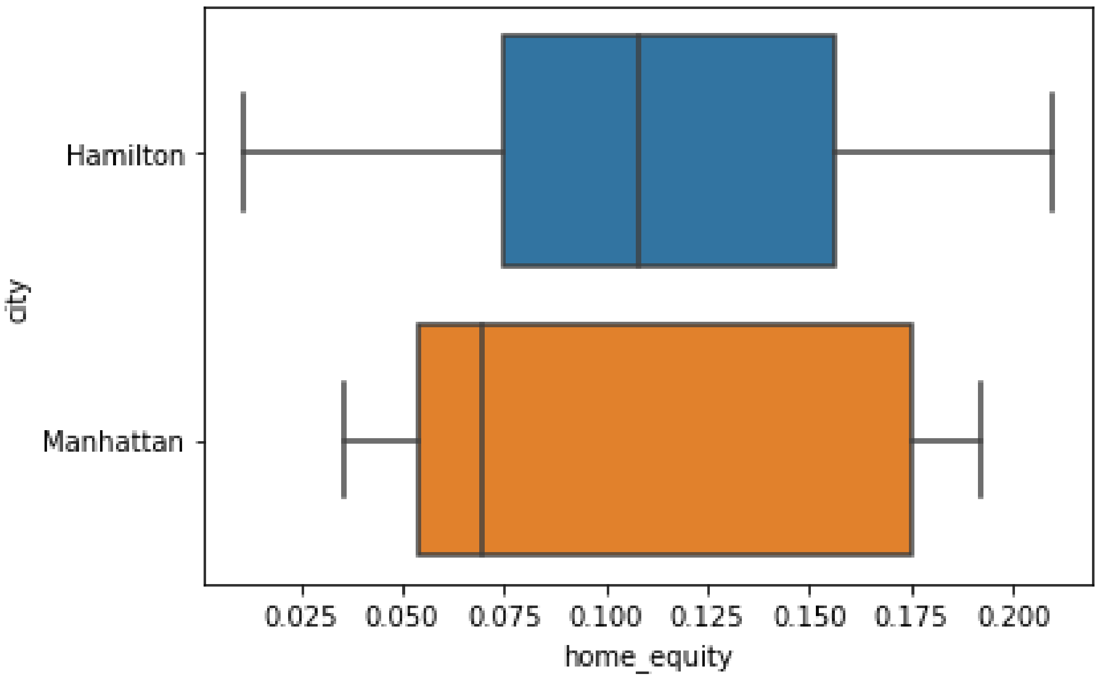
[42]:
```
        COUNTYID  STATEID        state state_ab      city           place  \
UID
267822        53       36     New York       NY  Hamilton        Hamilton
263797        21       34   New Jersey       NJ  Hamilton       Yardville
270979        17       39         Ohio       OH  Hamilton   Hamilton City
259028        95       28  Mississippi       MS  Hamilton        Hamilton
270984        17       39         Ohio       OH  Hamilton       New Miami

           type primary  zip_code  area_code  …  female_age_median  \
UID                                           …
267822     City   tract     13346        315  …           45.33333
263797     City   tract      8610        609  …           55.00000
270979  Village   tract     45015        513  …           31.66667
259028      CDP   tract     39746        662  …           35.91667
270984  Village   tract     45013        513  …           52.33333

        female_age_stdev  female_age_sample_weight  female_age_samples  \
UID
267822          22.51276                 685.33845              2618.0
263797          24.05831                 732.58443              3124.0
```

```
270979          22.66500                    565.32725             2528.0
259028          22.79602                    483.01311             1954.0
270984          24.55724                    682.81171             2912.0

        pct_own  married  married_snp  separated  divorced  bad_debt
UID
267822  0.79046  0.57851      0.01882    0.01240   0.08770   0.09408
263797  0.64400  0.56377      0.01980    0.00990   0.04892   0.18071
270979  0.61278  0.47397      0.04419    0.02663   0.13741   0.15005
259028  0.83241  0.58678      0.01052    0.00000   0.11721   0.02130
270984  0.63194  0.55697      0.01322    0.00000   0.15209   0.15651

[5 rows x 78 columns]
```

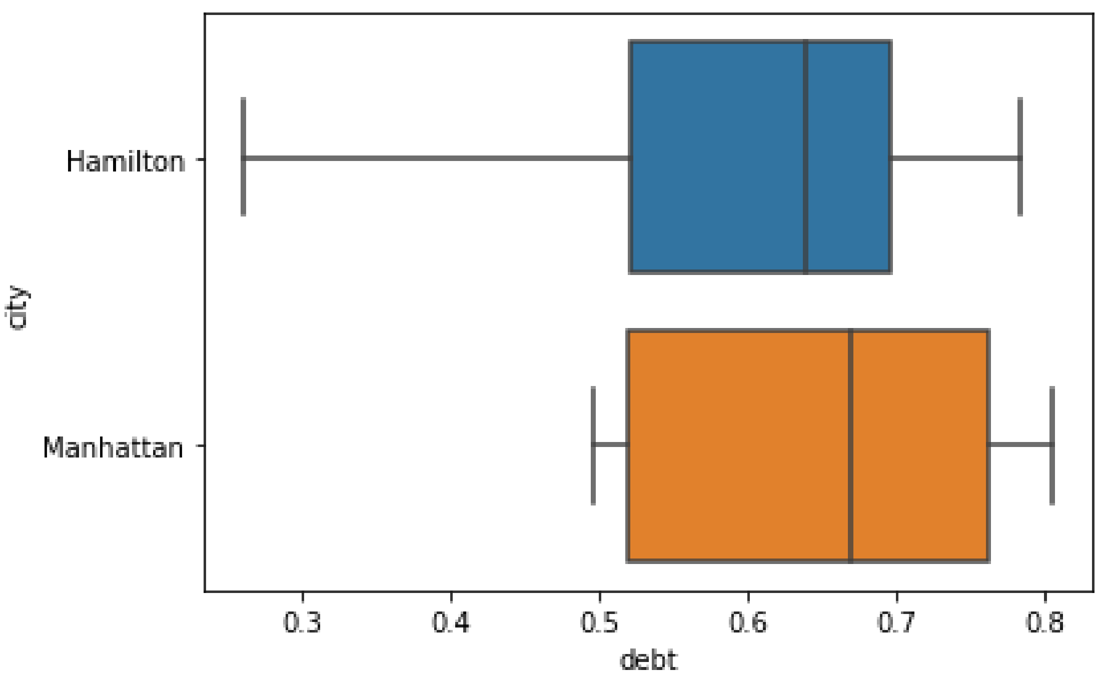
[43]:
```python
sns.boxplot(data=df_box_city,x='second_mortgage',y='city')
plt.show()
```



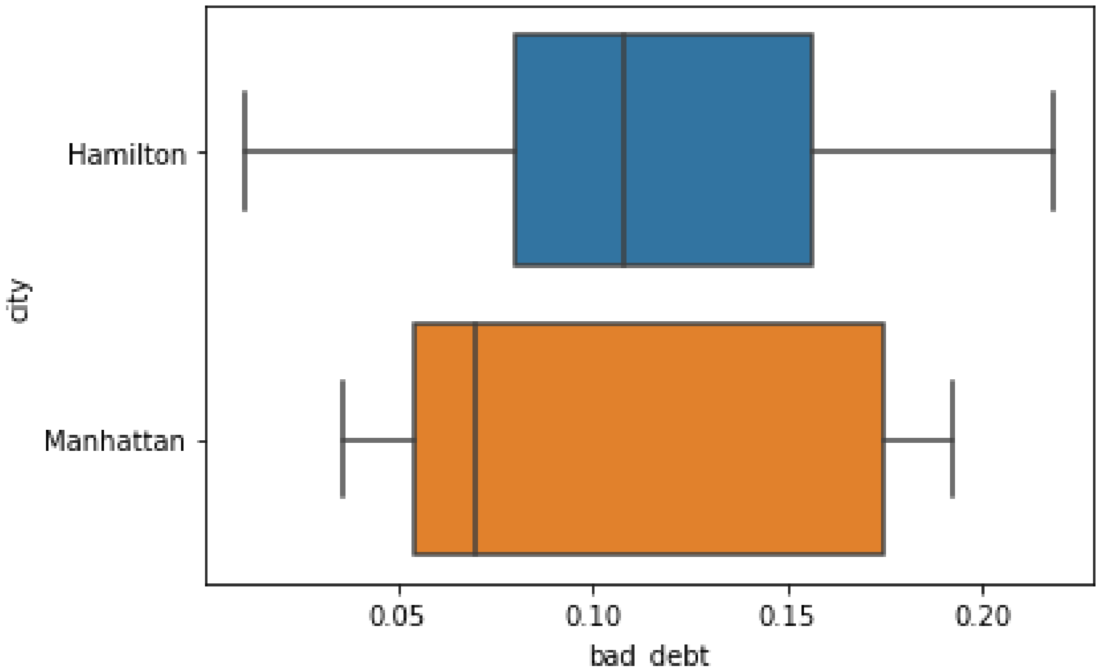
[44]:
```python
sns.boxplot(data=df_box_city,x='home_equity',y='city')
plt.show()
```



[45]:
```
sns.boxplot(data=df_box_city,x='debt',y='city')
plt.show()
```

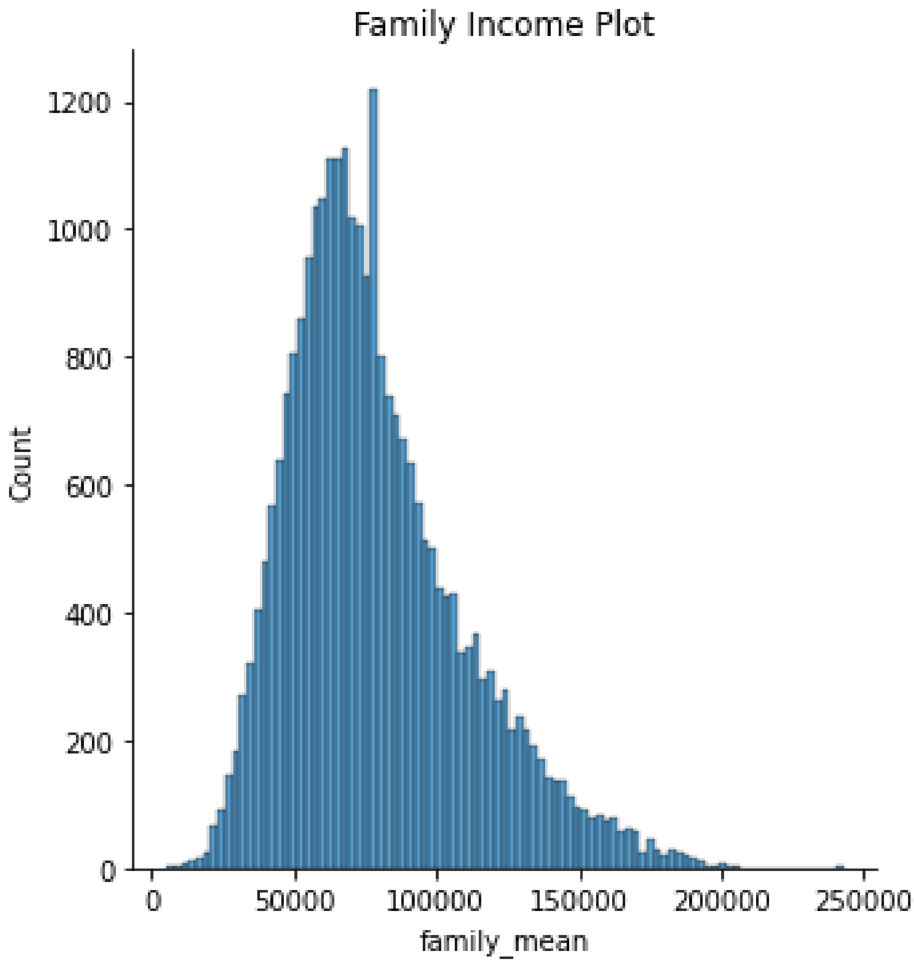


[46]:
```
sns.boxplot(data=df_box_city,x='bad_debt',y='city')
plt.show()
```



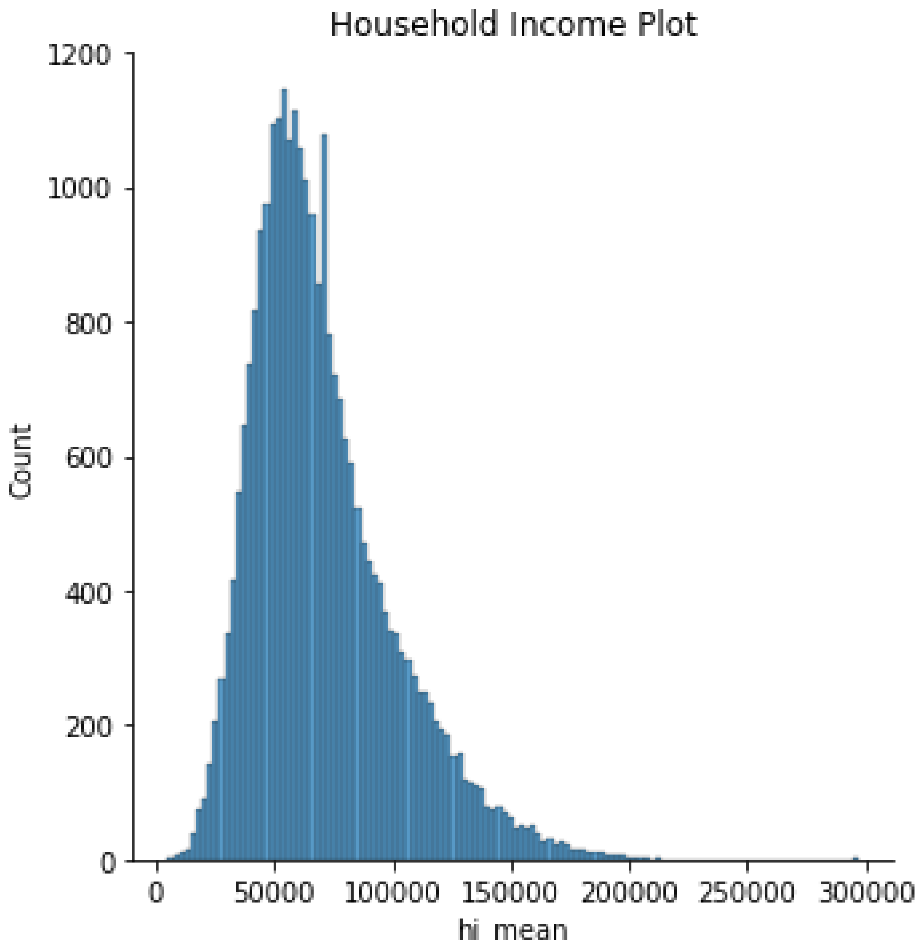
[47]:
```
sns.displot(df_train['family_mean'])
plt.title('Family Income Plot')
```

[47]:
```
Text(0.5, 1.0, 'Family Income Plot')
```



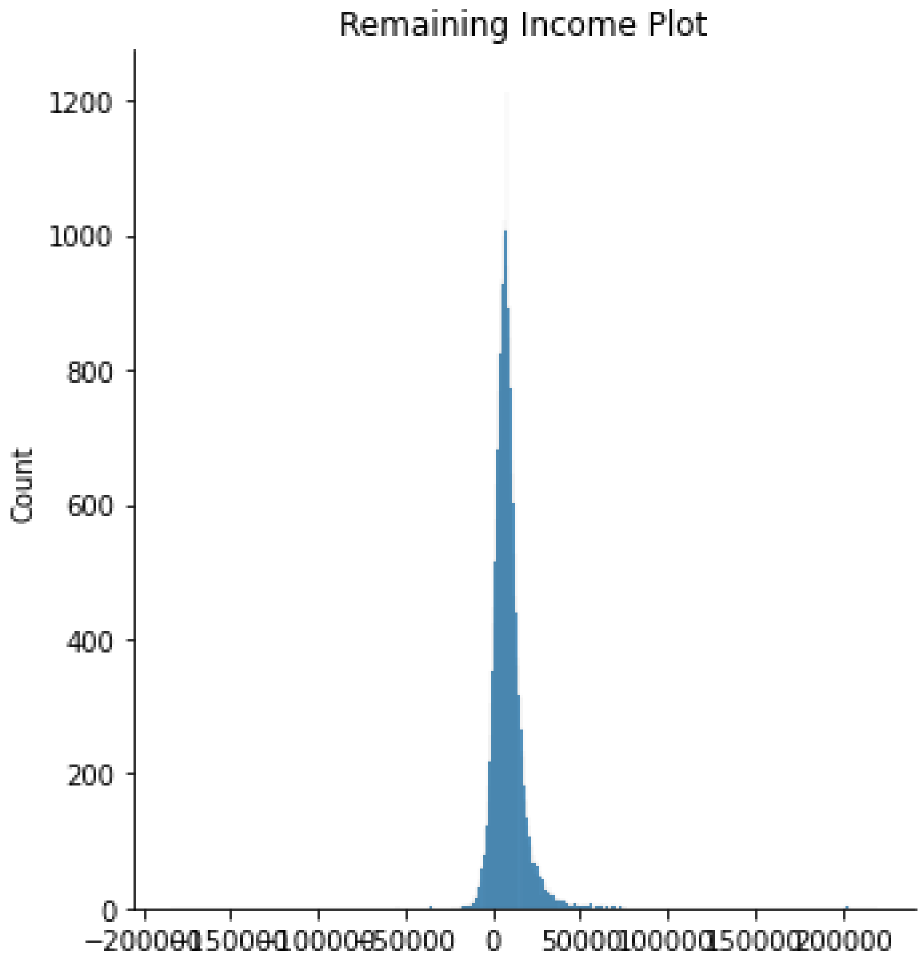
[48]:
```
sns.displot(df_train['hi_mean'])
plt.title('Household Income Plot')
```

[48]:
```
Text(0.5, 1.0, 'Household Income Plot')
```



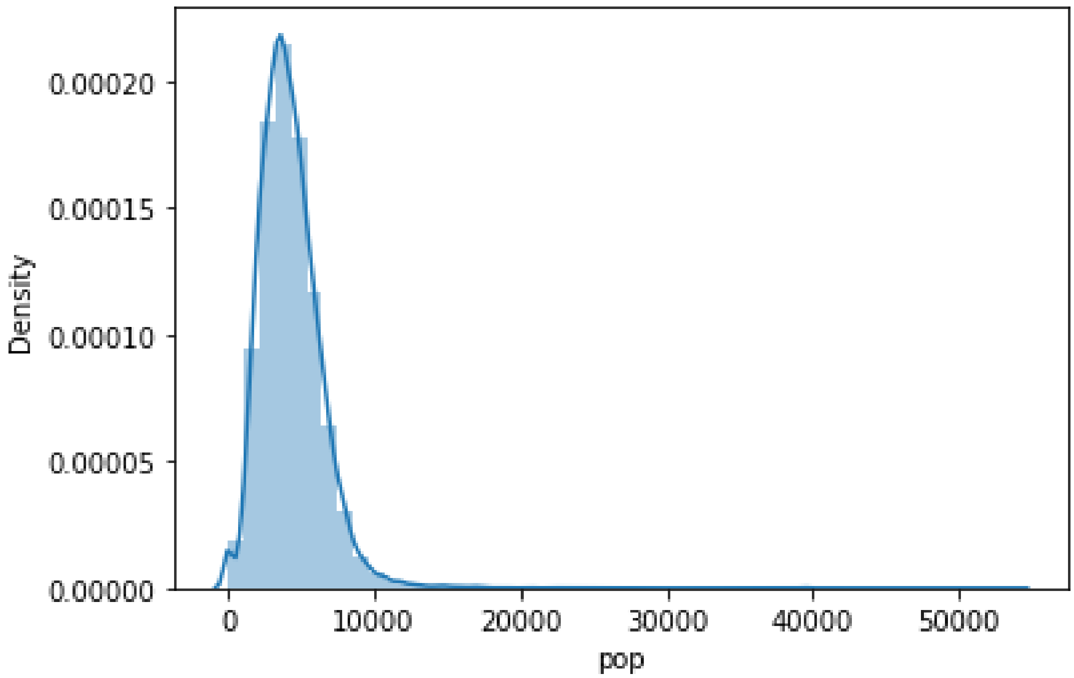
[49]:
```
sns.displot(df_train['family_mean']-df_train['hi_mean'])
plt.title('Remaining Income Plot')
```

[49]: Text(0.5, 1.0, 'Remaining Income Plot')



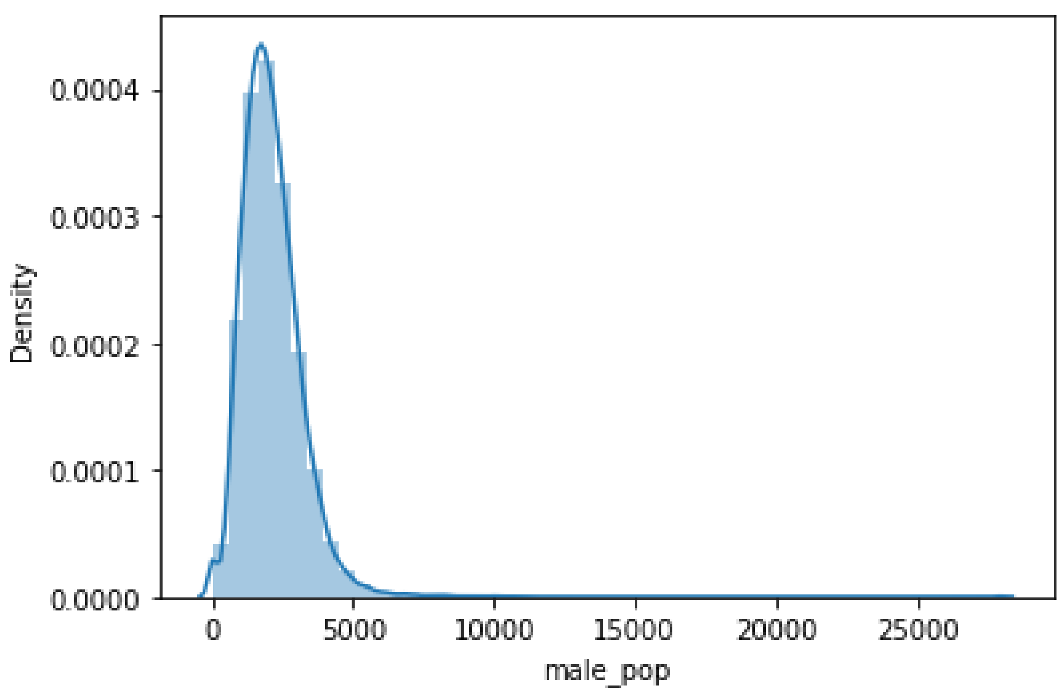
[50]:
```
sns.distplot(df_train['pop'])
```

[50]: <AxesSubplot:xlabel='pop', ylabel='Density'>



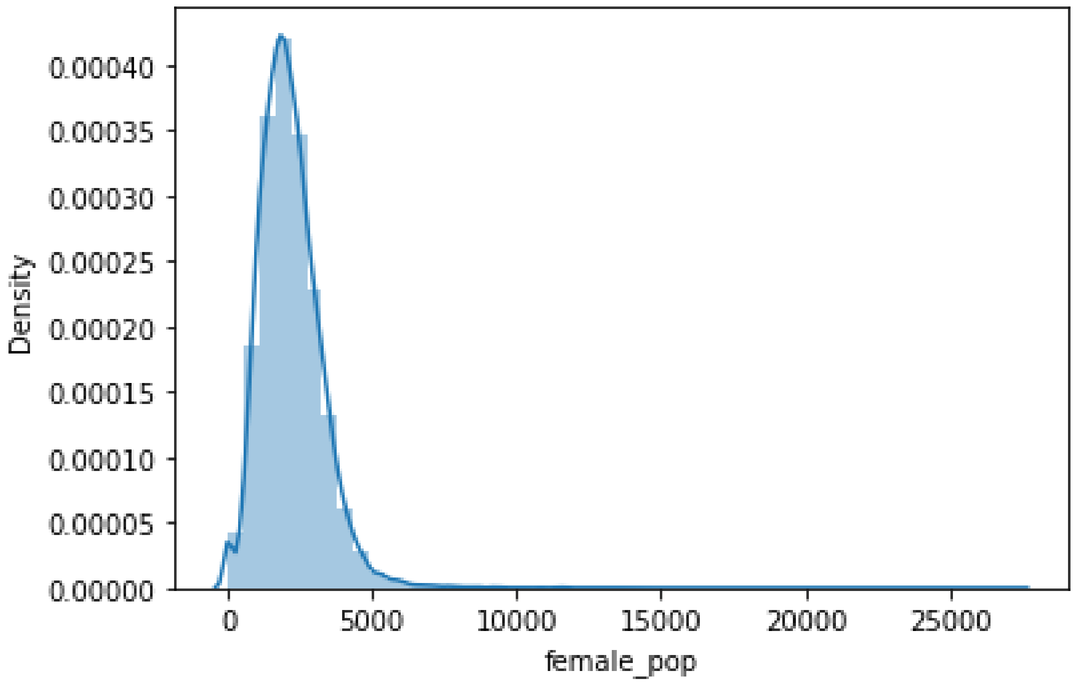
[51]:
```
sns.distplot(df_train['male_pop'])
```

[51]: <AxesSubplot:xlabel='male_pop', ylabel='Density'>



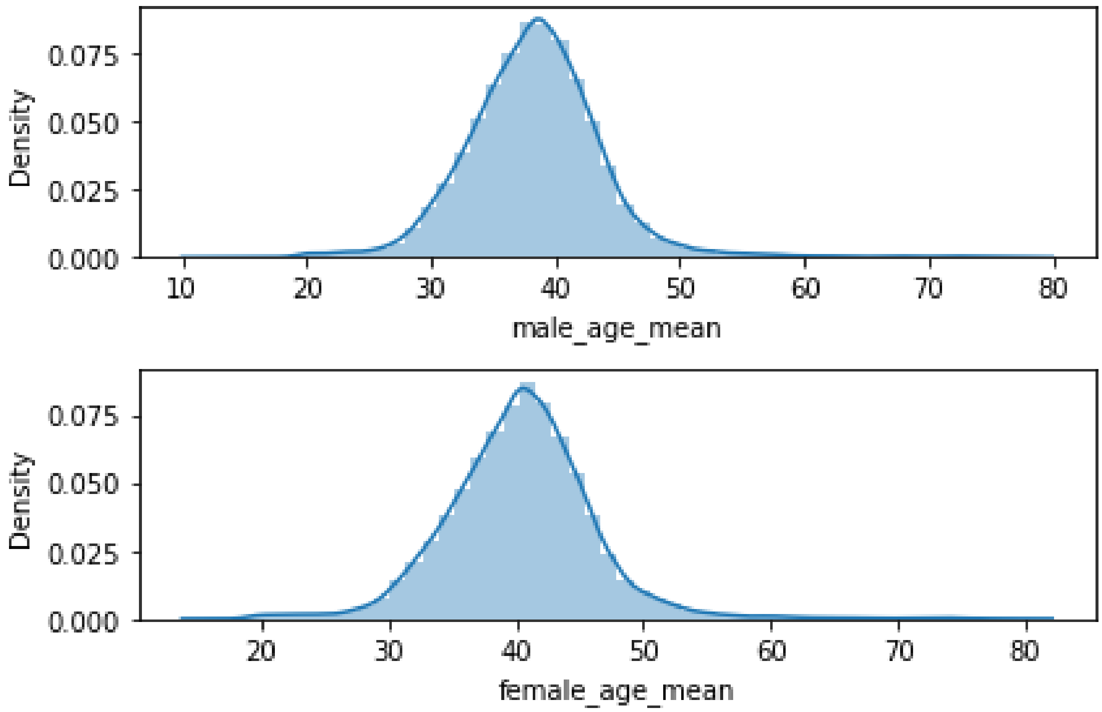
[52]:
```
sns.distplot(df_train['female_pop'])
```

[52]: <AxesSubplot:xlabel='female_pop', ylabel='Density'>



[53]:
```
fig,(ax1,ax2)=plt.subplots(2,1)
plt.subplots_adjust(wspace=0.8,hspace=0.9)
sns.distplot(df_train['male_age_mean'],ax=ax1)
sns.distplot(df_train['female_age_mean'],ax=ax2)
plt.tight_layout()
plt.show
```

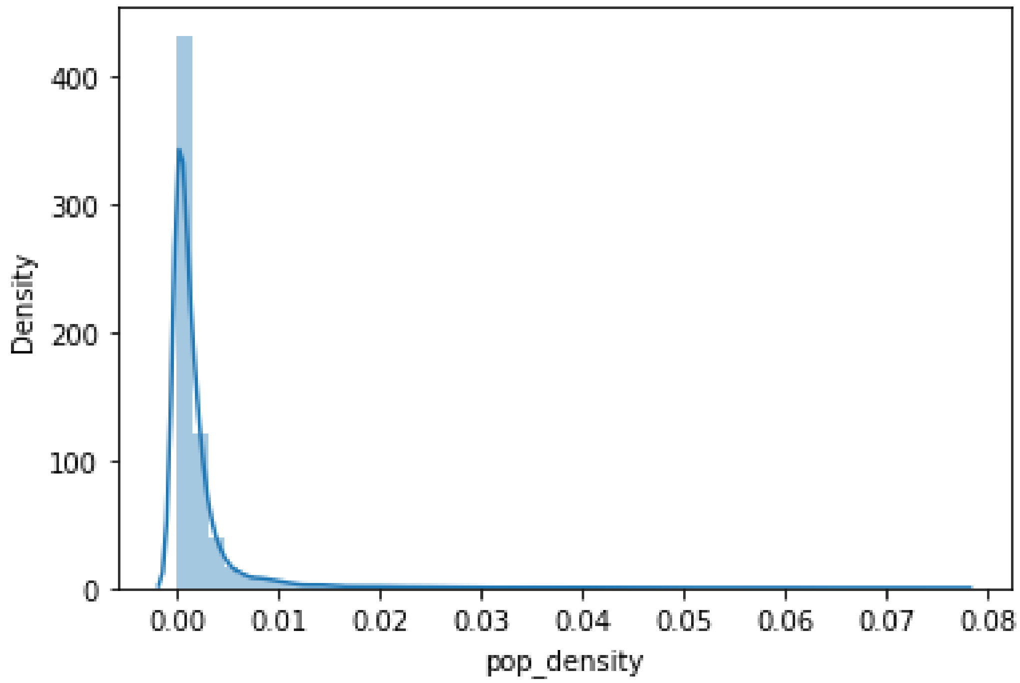
[53]: <function matplotlib.pyplot.show(close=None, block=None)>



[54]:
```
df_train['pop_density']=df_train['pop']/df_train['ALand']
```

[55]:
```
df_test['pop_density']=df_test['pop']/df_test['ALand']
```

[56]:
```
sns.distplot(df_train['pop_density'])
plt.show()
```



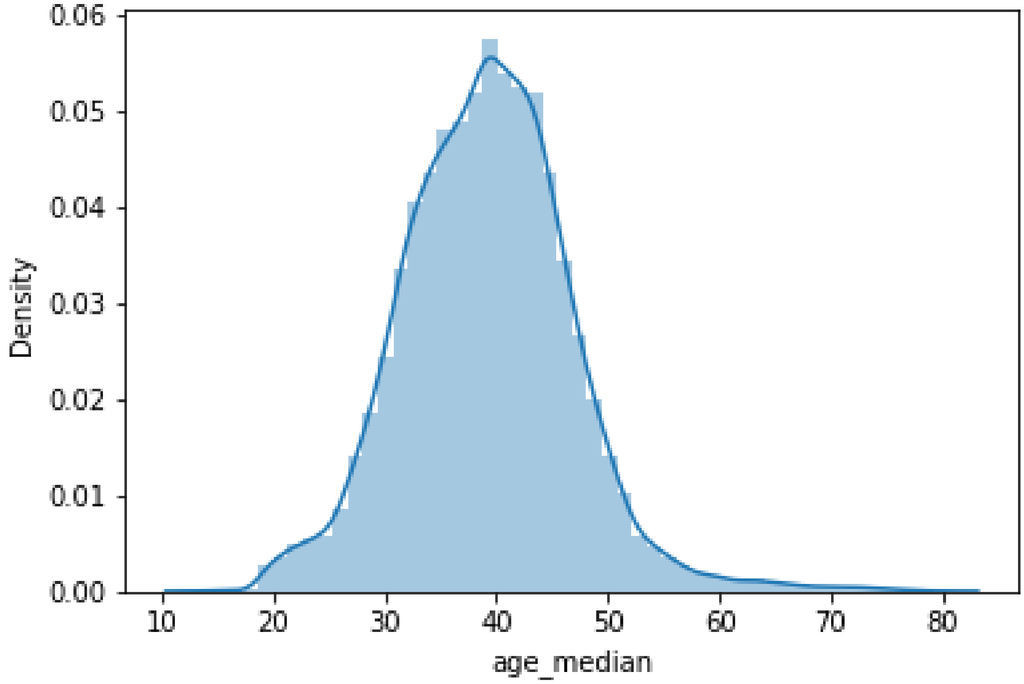
[57]:
```
df_train['age_median']=(df_train['male_age_median']+df_train['female_age_median'])/
↪2
```

[58]:
```
df_test['age_median']=(df_test['male_age_median']+df_test['female_age_median'])/
↪2
```

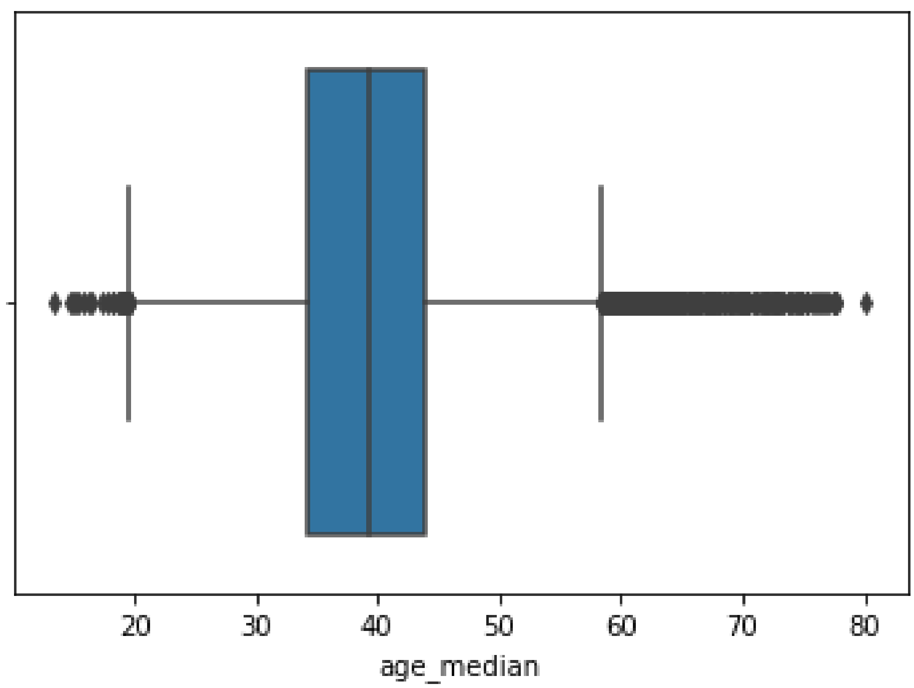
[59]:
```
df_train['age_median']
```

[59]:
```
UID
267822    44.666665
246444    34.791665
245683    41.833330
279653    49.750000
247218    22.000000
            …
279212    40.916670
277856    39.166665
233000    44.166665
287425    45.041670
265371    31.166665
Name: age_median, Length: 27321, dtype: float64
```

[ ]:
```
sns.distplot(df_train['age_median'])
plt.show()
```



[ ]:
```
sns.boxplot(df_train['age_median'])
plt.show()
```



[ ]:
```
#apply function
def func(num):
    if num<7000:
        return 'low'
```

[ ]:
```
df_train['pop_bin']=df_train['pop'].apply(func)
```

[ ]:
```
df_train['pop_bins']=pd.cut(df_train['pop'],bins=5,labels=['very␣
↪low','low','medium','high','very high'])
```

[ ]:
```
df_train['pop_bins'].value_counts()
```

[ ]:
```
very low     27058
low            246
medium           9
high             7
very high        1
Name: pop_bins, dtype: int64
```

[ ]:
```
df_train[['pop','pop_bins']].head()
```

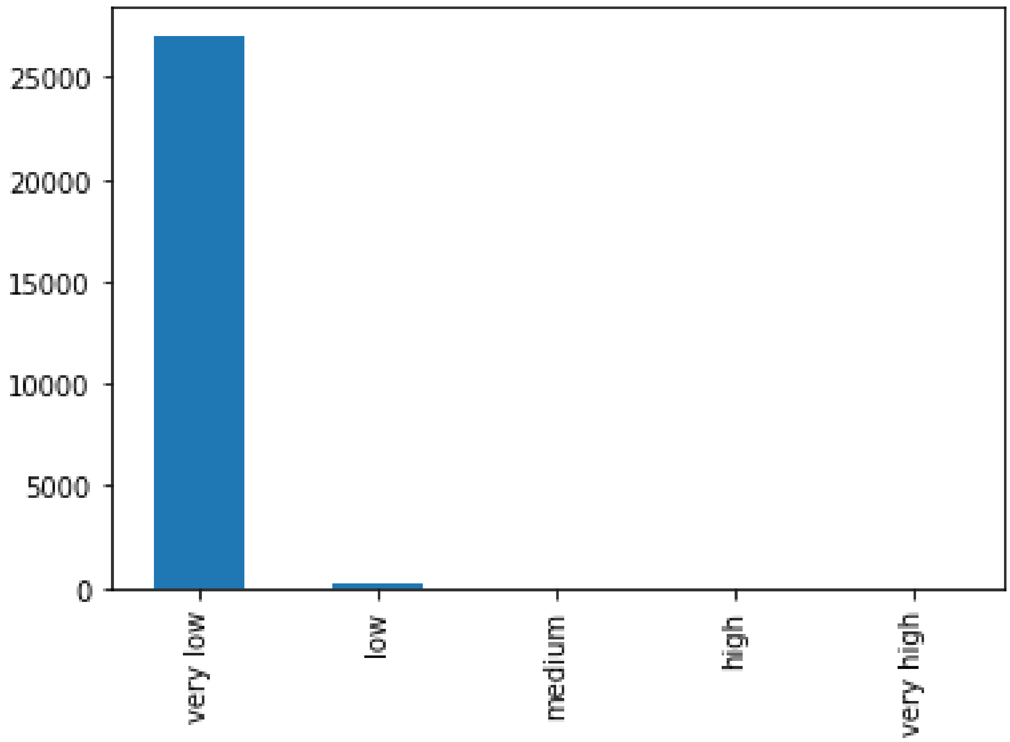
[ ]:
```
         pop  pop_bins
UID
267822  5230  very low
246444  2633  very low
245683  6881  very low
279653  2700  very low
247218  5637  very low
```

[ ]:
```
df_train['pop'].describe()
```

[ ]:
```
count    27321.000000
mean      4316.032685
std       2169.226173
min          0.000000
25%       2885.000000
50%       4042.000000
75%       5430.000000
max      53812.000000
Name: pop, dtype: float64
```

[ ]:
```
df_train['pop_bins'].value_counts().plot(kind='bar')
```

[ ]:
```
<AxesSubplot:>
```




```
[ ]: df_train.groupby(by='pop_bins')[['married','separated', 'divorced']].count()
```

```
[ ]:            married  separated  divorced
     pop_bins
     very low     27058      27058     27058
     low            246        246       246
     medium           9          9         9
     high             7          7         7
     very high        1          1         1
```

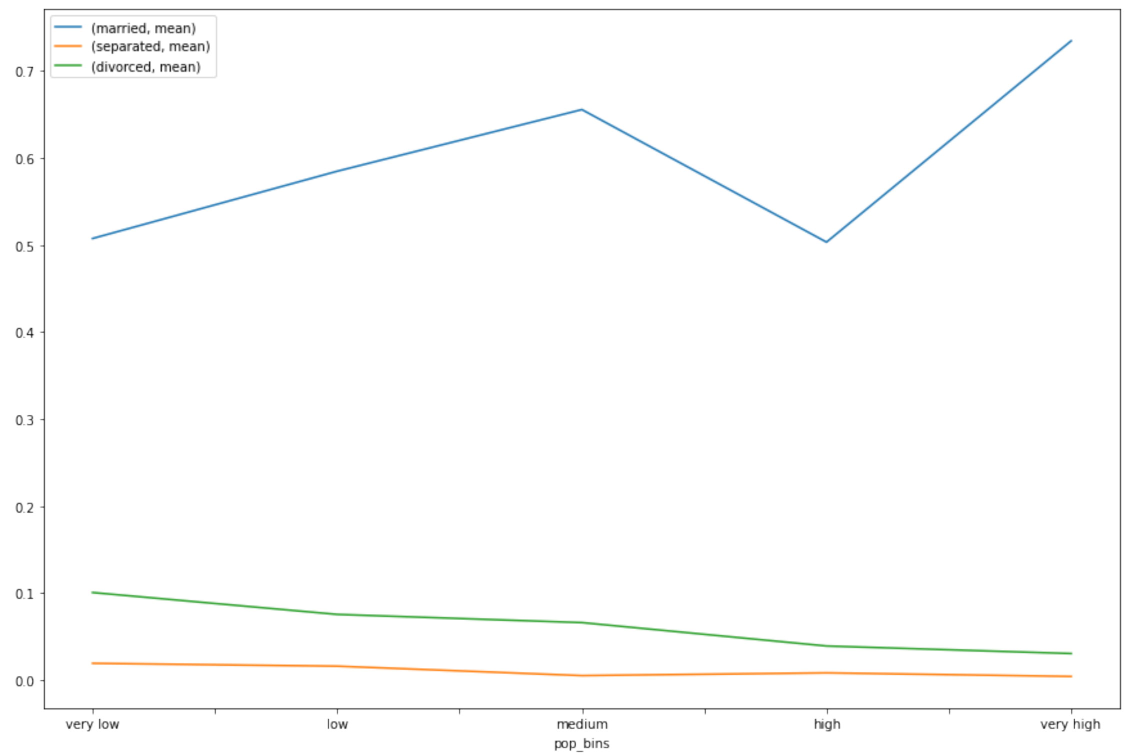
```
[ ]: df_train.groupby(by='pop_bins')[['married','separated', 'divorced']].
       ↪agg(['sum','mean','median','count'])
```

```
[ ]:               married                                  separated            \
                       sum      mean    median  count        sum      mean
     pop_bins
     very low  13733.22489  0.507548  0.524680  27058  517.52126  0.019126
     low         143.88385  0.584894  0.593135    246    3.89480  0.015833
     medium        5.90163  0.655737  0.618710      9    0.04503  0.005003
     high          3.52351  0.503359  0.335660      7    0.05699  0.008141
     very high     0.73474  0.734740  0.734740      1    0.00405  0.004050

                          divorced
```

	median	count	sum	mean	median	count
pop_bins						
very low	0.013650	27058	2719.430721	0.100504	0.096020	27058
low	0.011195	246	18.535600	0.075348	0.070045	246
medium	0.004120	9	0.593340	0.065927	0.064890	9
high	0.002500	7	0.273210	0.039030	0.010320	7
very high	0.004050	1	0.030360	0.030360	0.030360	1

[ ]:
```
df_train.groupby(by='pop_bins')[['married','separated', 'divorced']].
 ↪agg(['mean']).plot(figsize=(15,10))
plt.legend(loc='best')
```

[ ]: <matplotlib.legend.Legend at 0x7fd9ebcdb710>



[ ]:
```
rent_state_mean=df_train.groupby(by='state')['rent_mean'].agg(["mean"])
```

[ ]:
```
rent_state_mean
```

[ ]:

	mean
state	
Alabama	774.004927
Alaska	1185.763570
Arizona	1097.753511

Arkansas	720.918575
California	1471.133857
Colorado	1198.191514
Connecticut	1317.100534
Delaware	1127.309811
District of Columbia	1417.097934
Florida	1141.758549
Georgia	964.575973
Hawaii	1710.629412
Idaho	800.486650
Illinois	1034.887921
Indiana	810.910355
Iowa	737.246152
Kansas	831.215856
Kentucky	742.199763
Louisiana	846.375506
Maine	829.941899
Maryland	1412.009565
Massachusetts	1211.811159
Michigan	928.123200
Minnesota	957.376502
Mississippi	738.111770
Missouri	829.011192
Montana	776.337306
Nebraska	835.165893
Nevada	1128.641766
New Hampshire	1083.090073
New Jersey	1379.709933
New Mexico	853.611858
New York	1248.850743
North Carolina	885.593430
North Dakota	771.423137
Ohio	820.004760
Oklahoma	777.702422
Oregon	1024.616948
Pennsylvania	949.580140
Puerto Rico	550.079459
Rhode Island	1039.482069
South Carolina	859.919160
South Dakota	685.325569
Tennessee	856.649930
Texas	977.074993
Utah	1068.930520
Vermont	937.119939
Virginia	1305.707687
Washington	1126.649264
West Virginia	667.193267

```
Wisconsin               841.670190
Wyoming                 861.395327
```

[ ]:
```
income_state_mean=df_train.groupby(by='state')['family_mean'].agg(["mean"])
```

[ ]:
```
income_state_mean.head()
```

[ ]:
```
                    mean
state
Alabama     67030.064213
Alaska      92136.545109
Arizona     73328.238798
Arkansas    64765.377850
California  87655.470820
```

[ ]:
```
# calculate rent percentage
rent_perc=rent_state_mean['mean']/income_state_mean['mean']
```

[ ]:
```
rent_perc.head()
```

[ ]:
```
state
Alabama       0.011547
Alaska        0.012870
Arizona       0.014970
Arkansas      0.011131
California    0.016783
Name: mean, dtype: float64
```

[ ]:
```
df_train.columns
```

[ ]:
```
Index(['COUNTYID', 'STATEID', 'state', 'state_ab', 'city', 'place', 'type',
       'primary', 'zip_code', 'area_code', 'lat', 'lng', 'ALand', 'AWater',
       'pop', 'male_pop', 'female_pop', 'rent_mean', 'rent_median',
       'rent_stdev', 'rent_sample_weight', 'rent_samples', 'rent_gt_10',
       'rent_gt_15', 'rent_gt_20', 'rent_gt_25', 'rent_gt_30', 'rent_gt_35',
       'rent_gt_40', 'rent_gt_50', 'universe_samples', 'used_samples',
       'hi_mean', 'hi_median', 'hi_stdev', 'hi_sample_weight', 'hi_samples',
       'family_mean', 'family_median', 'family_stdev', 'family_sample_weight',
       'family_samples', 'hc_mortgage_mean', 'hc_mortgage_median',
       'hc_mortgage_stdev', 'hc_mortgage_sample_weight', 'hc_mortgage_samples',
       'hc_mean', 'hc_median', 'hc_stdev', 'hc_samples', 'hc_sample_weight',
       'home_equity_second_mortgage', 'second_mortgage', 'home_equity', 'debt',
       'second_mortgage_cdf', 'home_equity_cdf', 'debt_cdf', 'hs_degree',
       'hs_degree_male', 'hs_degree_female', 'male_age_mean',
       'male_age_median', 'male_age_stdev', 'male_age_sample_weight',
       'male_age_samples', 'female_age_mean', 'female_age_median',
       'female_age_stdev', 'female_age_sample_weight', 'female_age_samples',
```

```
 'pct_own', 'married', 'married_snp', 'separated', 'divorced',
 'bad_debt', 'pop_density', 'age_median', 'pop_bin', 'pop_bins'],
dtype='object')
```

[ ]:
```
df_num=df_train.select_dtypes(exclude="object")
```

[ ]:
```
df_num.shape
```

[ ]:
```
(27321, 75)
```

[ ]:
```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 27321 entries, 267822 to 265371
Data columns (total 82 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   COUNTYID                      27321 non-null  int64
 1   STATEID                       27321 non-null  int64
 2   state                         27321 non-null  object
 3   state_ab                      27321 non-null  object
 4   city                          27321 non-null  object
 5   place                         27321 non-null  object
 6   type                          27321 non-null  object
 7   primary                       27321 non-null  object
 8   zip_code                      27321 non-null  int64
 9   area_code                     27321 non-null  int64
 10  lat                           27321 non-null  float64
 11  lng                           27321 non-null  float64
 12  ALand                         27321 non-null  float64
 13  AWater                        27321 non-null  int64
 14  pop                           27321 non-null  int64
 15  male_pop                      27321 non-null  int64
 16  female_pop                    27321 non-null  int64
 17  rent_mean                     27321 non-null  float64
 18  rent_median                   27321 non-null  float64
 19  rent_stdev                    27321 non-null  float64
 20  rent_sample_weight            27321 non-null  float64
 21  rent_samples                  27321 non-null  float64
 22  rent_gt_10                    27321 non-null  float64
 23  rent_gt_15                    27321 non-null  float64
 24  rent_gt_20                    27321 non-null  float64
 25  rent_gt_25                    27321 non-null  float64
 26  rent_gt_30                    27321 non-null  float64
 27  rent_gt_35                    27321 non-null  float64
 28  rent_gt_40                    27321 non-null  float64
 29  rent_gt_50                    27321 non-null  float64
```

```
30  universe_samples            27321 non-null  int64
31  used_samples                27321 non-null  int64
32  hi_mean                     27321 non-null  float64
33  hi_median                   27321 non-null  float64
34  hi_stdev                    27321 non-null  float64
35  hi_sample_weight            27321 non-null  float64
36  hi_samples                  27321 non-null  float64
37  family_mean                 27321 non-null  float64
38  family_median               27321 non-null  float64
39  family_stdev                27321 non-null  float64
40  family_sample_weight        27321 non-null  float64
41  family_samples              27321 non-null  float64
42  hc_mortgage_mean            27321 non-null  float64
43  hc_mortgage_median          27321 non-null  float64
44  hc_mortgage_stdev           27321 non-null  float64
45  hc_mortgage_sample_weight   27321 non-null  float64
46  hc_mortgage_samples         27321 non-null  float64
47  hc_mean                     27321 non-null  float64
48  hc_median                   27321 non-null  float64
49  hc_stdev                    27321 non-null  float64
50  hc_samples                  27321 non-null  float64
51  hc_sample_weight            27321 non-null  float64
52  home_equity_second_mortgage 27321 non-null  float64
53  second_mortgage             27321 non-null  float64
54  home_equity                 27321 non-null  float64
55  debt                        27321 non-null  float64
56  second_mortgage_cdf         27321 non-null  float64
57  home_equity_cdf             27321 non-null  float64
58  debt_cdf                    27321 non-null  float64
59  hs_degree                   27321 non-null  float64
60  hs_degree_male              27321 non-null  float64
61  hs_degree_female            27321 non-null  float64
62  male_age_mean               27321 non-null  float64
63  male_age_median             27321 non-null  float64
64  male_age_stdev              27321 non-null  float64
65  male_age_sample_weight      27321 non-null  float64
66  male_age_samples            27321 non-null  float64
67  female_age_mean             27321 non-null  float64
68  female_age_median           27321 non-null  float64
69  female_age_stdev            27321 non-null  float64
70  female_age_sample_weight    27321 non-null  float64
71  female_age_samples          27321 non-null  float64
72  pct_own                     27321 non-null  float64
73  married                     27321 non-null  float64
74  married_snp                 27321 non-null  float64
75  separated                   27321 non-null  float64
76  divorced                    27321 non-null  float64
77  bad_debt                    27321 non-null  float64
```

```
 78  pop_density                   27321 non-null  float64
 79  age_median                    27321 non-null  float64
 80  pop_bin                       24883 non-null  object
 81  pop_bins                      27321 non-null  category
dtypes: category(1), float64(64), int64(10), object(7)
memory usage: 18.4+ MB
```

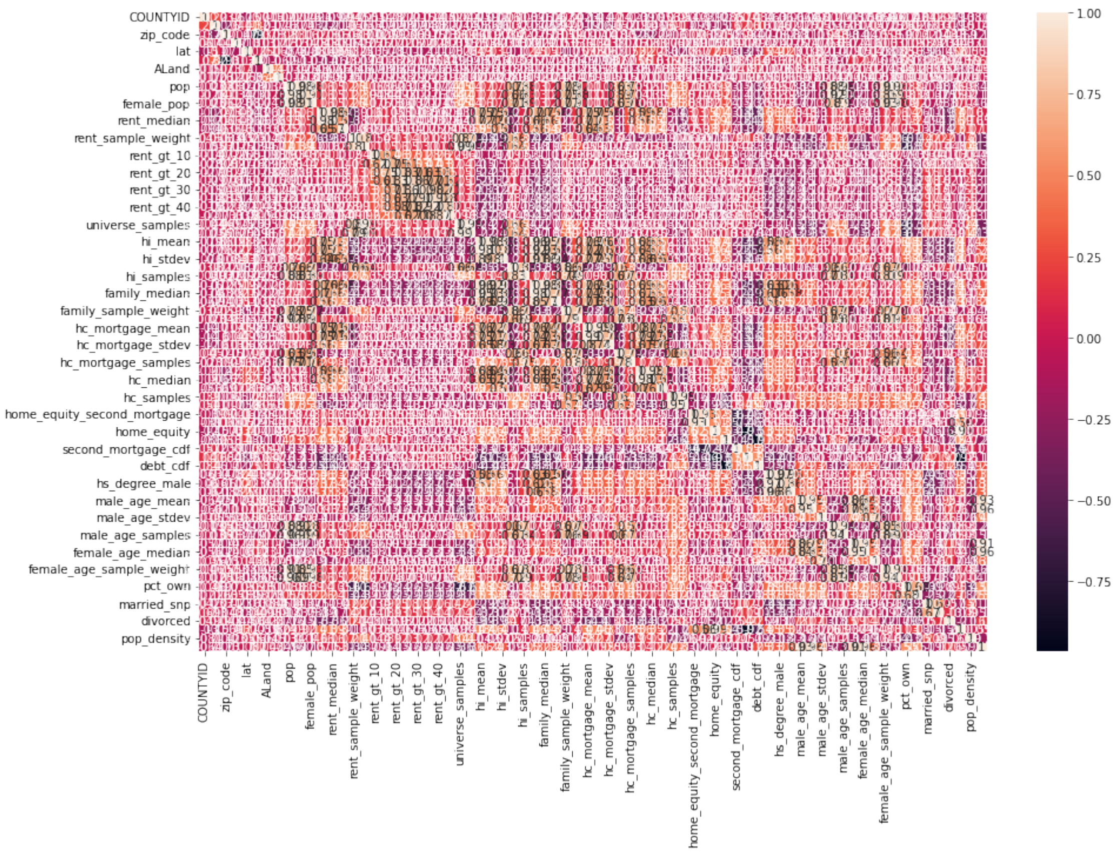
[ ]:
```
df_num.corr()
```

[ ]:
```
              COUNTYID   STATEID  zip_code  area_code       lat       lng  \
COUNTYID      1.000000  0.224549  0.036527   0.067171 -0.149272  0.070414
STATEID       0.224549  1.000000 -0.261465   0.043718  0.109934  0.319964
zip_code      0.036527 -0.261465  1.000000  -0.004681 -0.070775 -0.926708
area_code     0.067171  0.043718 -0.004681   1.000000 -0.125415 -0.013494
lat          -0.149272  0.109934 -0.070775  -0.125415  1.000000  0.025450
…                  …        …        …          …        …        …
separated     0.069059  0.030409 -0.048023   0.022543 -0.138048  0.049228
divorced      0.048850  0.018748  0.043310  -0.043722 -0.056018 -0.004321
bad_debt     -0.125892 -0.151007 -0.069348  -0.003658  0.208792 -0.005876
pop_density -0.080509 -0.013671 -0.119014  -0.030743  0.054513  0.066056
age_median   -0.063521 -0.017172 -0.126150  -0.017118  0.008246  0.104944

                 ALand    AWater       pop  male_pop  …  \
COUNTYID      0.015469  0.016550 -0.002662 -0.002615  …
STATEID      -0.017275 -0.026476 -0.036599 -0.040351  …
zip_code      0.072711  0.031679  0.083058  0.099959  …
area_code     0.016563  0.021711  0.031834  0.034387  …
lat           0.100498  0.067660 -0.078283 -0.072763  …
…                  …        …        …        …  …
separated    -0.005904 -0.001208 -0.083182 -0.074929  …
divorced      0.023381  0.007677 -0.160931 -0.146619  …
bad_debt     -0.079618 -0.024112  0.099489  0.092085  …
pop_density -0.044934 -0.013174  0.033740  0.020651  …
age_median    0.042532  0.004878 -0.162499 -0.166810  …

              female_age_sample_weight  female_age_samples   pct_own   married  \
COUNTYID                      0.004587           -0.001227 -0.004632 -0.021428
STATEID                      -0.025104           -0.028238  0.069314  0.025763
zip_code                      0.055497            0.059305 -0.069965  0.030217
area_code                     0.029857            0.031128  0.018877  0.057824
lat                          -0.080855           -0.087667  0.056487  0.035480
…                                  …                  …        …        …
separated                    -0.091913           -0.088709 -0.284877 -0.219686
divorced                     -0.198491           -0.169450 -0.095413 -0.267833
bad_debt                      0.078159            0.104039  0.134257  0.182985
pop_density                   0.046016            0.040268 -0.426353 -0.248678
age_median                   -0.246096           -0.153775  0.546692  0.495153
```

```
              married_snp  separated  divorced  bad_debt  pop_density  \
COUNTYID         0.041710   0.069059  0.048850 -0.125892    -0.080509
STATEID         -0.033283   0.030409  0.018748 -0.151007    -0.013671
zip_code         0.020541  -0.048023  0.043310 -0.069348    -0.119014
area_code        0.022687   0.022543 -0.043722 -0.003658    -0.030743
lat             -0.158657  -0.138048 -0.056018  0.208792     0.054513
…                     …          …         …         …           …
separated        0.668481   1.000000  0.133244 -0.151824     0.094859
divorced         0.057364   0.133244  1.000000 -0.210203    -0.155328
bad_debt        -0.151008  -0.151824 -0.210203  1.000000    -0.005871
pop_density      0.212778   0.094859 -0.155328 -0.005871     1.000000
age_median      -0.190105  -0.116763  0.164205  0.058892    -0.198546

              age_median
COUNTYID       -0.063521
STATEID        -0.017172
zip_code       -0.126150
area_code      -0.017118
lat             0.008246
…                   …
separated      -0.116763
divorced        0.164205
bad_debt        0.058892
pop_density    -0.198546
age_median      1.000000

[74 rows x 74 columns]
```

[ ]:
```python
plt.figure(figsize=(15,10))
sns.heatmap(df_num.corr(),annot=True)
```

[ ]:
```
<AxesSubplot:>
```



[ ]:
```
df_train.corr().nlargest(10,"hc_mortgage_mean")
```

[ ]:

	COUNTYID	STATEID	zip_code	area_code	lat	\
hc_mortgage_mean	-0.139581	-0.167274	-0.016521	0.042561	0.097747	
hc_mortgage_median	-0.137223	-0.163141	-0.014076	0.040420	0.098932	
hc_mortgage_stdev	-0.121160	-0.161088	-0.017648	0.037865	0.062863	
hc_mean	-0.090427	-0.014471	-0.216220	0.032167	0.217543	
hc_median	-0.090027	-0.006556	-0.218867	0.032809	0.216665	
hi_stdev	-0.076096	-0.102172	-0.008421	0.003285	0.107065	
hi_mean	-0.078694	-0.085679	0.001909	0.018253	0.128503	
family_mean	-0.075688	-0.071612	-0.024658	0.001865	0.151403	
rent_mean	-0.099668	-0.215943	0.073246	0.042648	-0.004272	
family_median	-0.073908	-0.062530	-0.027690	0.002106	0.150768	

	lng	ALand	AWater	pop	male_pop	…	\
hc_mortgage_mean	-0.097289	-0.056334	-0.009922	0.110659	0.106709	…	
hc_mortgage_median	-0.098047	-0.057950	-0.010905	0.106507	0.102745	…	
hc_mortgage_stdev	-0.081923	-0.015402	0.005098	0.082230	0.079537	…	
hc_mean	0.151952	-0.056723	-0.010573	0.051515	0.040595	…	
hc_median	0.157308	-0.058138	-0.010907	0.050546	0.039426	…	

```
hi_stdev          -0.047004 -0.018233  0.000892  0.126602  0.120234  …
hi_mean           -0.057359 -0.028435 -0.002166  0.166913  0.166467  …
family_mean       -0.027104 -0.027897 -0.002058  0.128173  0.125614  …
rent_mean         -0.168511 -0.067169 -0.009534  0.160590  0.156952  …
family_median     -0.022271 -0.029353 -0.002436  0.124272  0.121873  …

                    female_age_sample_weight  female_age_samples   pct_own  \
hc_mortgage_mean                    0.089454            0.111564  0.067828
hc_mortgage_median                  0.085296            0.107336  0.057242
hc_mortgage_stdev                   0.056719            0.082654  0.150366
hc_mean                             0.041283            0.061084  0.102150
hc_median                           0.041768            0.060374  0.089392
hi_stdev                            0.080518            0.128452  0.380186
hi_mean                             0.099221            0.162200  0.481066
family_mean                         0.081742            0.127229  0.450961
rent_mean                           0.127662            0.159766  0.140249
family_median                       0.078094            0.123292  0.451739

                     married  married_snp  separated  divorced  bad_debt  \
hc_mortgage_mean    0.222728    -0.082061  -0.178431 -0.403366  0.472699
hc_mortgage_median  0.207688    -0.074806  -0.170123 -0.397459  0.462500
hc_mortgage_stdev   0.273710    -0.112352  -0.180225 -0.296222  0.381657
hc_mean             0.199810    -0.116247  -0.167693 -0.336902  0.360709
hc_median           0.185114    -0.110327  -0.160633 -0.328496  0.345310
hi_stdev            0.444157    -0.253367  -0.282948 -0.343387  0.414195
hi_mean             0.530892    -0.291916  -0.316511 -0.390061  0.467399
family_mean         0.480095    -0.314925  -0.323433 -0.353274  0.455988
rent_mean           0.255671    -0.106256  -0.188108 -0.374508  0.412618
family_median       0.473053    -0.310826  -0.314345 -0.346997  0.442937

                    pop_density  age_median
hc_mortgage_mean       0.266100    0.114831
hc_mortgage_median     0.269361    0.095051
hc_mortgage_stdev      0.171223    0.252015
hc_mean                0.190739    0.142000
hc_median              0.188590    0.123160
hi_stdev               0.011956    0.295498
hi_mean               -0.041501    0.262170
family_mean           -0.040661    0.300215
rent_mean              0.156928    0.071445
family_median         -0.040476    0.280827

[10 rows x 74 columns]
```

[ ]:
```
pip install factor_analyzer
```

```
Defaulting to user installation because normal site-packages is not writeable
```

```
Requirement already satisfied: factor_analyzer in ./.local/lib/python3.7/site-
packages (0.4.1)
Requirement already satisfied: scipy in /usr/local/lib/python3.7/site-packages
(from factor_analyzer) (1.4.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.7/site-
packages (from factor_analyzer) (1.0.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/site-packages
(from factor_analyzer) (1.21.5)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/site-packages
(from factor_analyzer) (1.1.5)
Requirement already satisfied: pre-commit in ./.local/lib/python3.7/site-
packages (from factor_analyzer) (2.21.0)
Requirement already satisfied: python-dateutil>=2.7.3 in
/usr/local/lib/python3.7/site-packages (from pandas->factor_analyzer) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/site-
packages (from pandas->factor_analyzer) (2019.3)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.7/site-
packages (from pre-commit->factor_analyzer) (5.3.1)
Requirement already satisfied: nodeenv>=0.11.1 in ./.local/lib/python3.7/site-
packages (from pre-commit->factor_analyzer) (1.8.0)
Requirement already satisfied: cfgv>=2.0.0 in ./.local/lib/python3.7/site-
packages (from pre-commit->factor_analyzer) (3.3.1)
Requirement already satisfied: importlib-metadata in
/usr/local/lib/python3.7/site-packages (from pre-commit->factor_analyzer)
(1.6.0)
Requirement already satisfied: identify>=1.0.0 in ./.local/lib/python3.7/site-
packages (from pre-commit->factor_analyzer) (2.5.24)
Requirement already satisfied: virtualenv>=20.10.0 in
./.local/lib/python3.7/site-packages (from pre-commit->factor_analyzer)
(20.23.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/usr/local/lib/python3.7/site-packages (from scikit-learn->factor_analyzer)
(2.2.0)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/site-
packages (from scikit-learn->factor_analyzer) (0.14.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.7/site-
packages (from nodeenv>=0.11.1->pre-commit->factor_analyzer) (41.2.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-
packages (from python-dateutil>=2.7.3->pandas->factor_analyzer) (1.14.0)
Requirement already satisfied: platformdirs<4,>=3.2 in
./.local/lib/python3.7/site-packages (from virtualenv>=20.10.0->pre-
commit->factor_analyzer) (3.5.1)
Collecting filelock<4,>=3.11
  Using cached filelock-3.12.0-py3-none-any.whl (10 kB)
Collecting importlib-metadata
  Using cached importlib_metadata-6.6.0-py3-none-any.whl (22 kB)
Requirement already satisfied: distlib<1,>=0.3.6 in ./.local/lib/python3.7/site-
packages (from virtualenv>=20.10.0->pre-commit->factor_analyzer) (0.3.6)
```

```
Requirement already satisfied: typing-extensions>=3.6.4 in
/usr/local/lib/python3.7/site-packages (from importlib-metadata->pre-
commit->factor_analyzer) (4.0.1)
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/site-
packages (from importlib-metadata->pre-commit->factor_analyzer) (3.1.0)
Collecting typing-extensions>=3.6.4
  Using cached typing_extensions-4.5.0-py3-none-any.whl (27 kB)
Installing collected packages: typing-extensions, filelock, importlib-metadata
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
konoha 4.6.5 requires overrides<4.0.0,>=3.0.0, which is not installed.
flair 0.8.1 requires more-itertools~=8.8.0, but you have more-itertools 8.2.0
which is incompatible.
konoha 4.6.5 requires importlib-metadata<4.0.0,>=3.7.0, but you have importlib-
metadata 6.6.0 which is incompatible.
konoha 4.6.5 requires requests<3.0.0,>=2.25.1, but you have requests 2.23.0
which is incompatible.
Successfully installed filelock-3.12.0 importlib-metadata-6.6.0 typing-
extensions-4.5.0
WARNING: You are using pip version 22.0.3; however, version 23.1.2 is
available.
You should consider upgrading via the '/usr/local/bin/python3.7 -m pip install
--upgrade pip' command.
Note: you may need to restart the kernel to use updated packages.
```

[ ]:
```
from factor_analyzer import FactorAnalyzer
fa=FactorAnalyzer(n_factors=5)
fa.fit_transform(df_train.select_dtypes(exclude=('object','category')))
```

[ ]:
```
array([[-0.41205343,  0.51294274,  0.87903004, -1.11001903,  0.35041992],
       [-1.04824274, -0.50174344, -0.39507676,  0.081311  ,  0.32595819],
       [ 0.11209985,  1.26467376,  0.76773891, -0.47930207, -0.36363692],
       …,
       [-0.02669751, -0.75106047,  0.77972285, -1.39880081,  0.03865004],
       [ 2.53195117,  3.0676096 ,  1.45490888, -0.07337594, -1.50506532],
       [-0.1992642 ,  0.01415226, -1.23527594,  0.25760531, -0.04155054]])
```

[ ]:
```
# convert type column into numerical data
```

```
df_train.replace({'City':1,'Town':2,'CDP':3,'Village':4,'Borough':5,'Urban':
 ↪6},inplace=True)
```

[ ]:
```
df_train['type'].value_counts()
```

[ ]:
```
1    15237
2     3666
3     3658
4     3216
5     1226
6      318
Name: type, dtype: int64
```

[ ]:
```
df_test.replace({'City':1,'Town':2,'CDP':3,'Village':4,'Borough':5,'Urban':
 ↪6},inplace=True)
```

[ ]:
```
df_test['type'].value_counts()
```

[ ]:
```
1    6481
2    1634
3    1558
4    1356
5     509
6     171
Name: type, dtype: int64
```

[ ]:
```
input_cols=['COUNTYID',␣
 ↪'STATEID','type','zip_code','pop','family_mean','second_mortgage',
          ␣
 ↪'home_equity','debt','hs_degree','age_median','pct_own','married','separated',␣
 ↪'divorced']
```

[ ]:
```
x_train=df_train[input_cols]
```

[ ]:
```
x_train
```

[ ]:
```
        COUNTYID  STATEID  type  zip_code   pop   family_mean  \
UID
267822        53       36     1     13346  5230   67994.14790
246444       141       18     1     46616  2633   50670.10337
245683        63       18     1     46122  6881   95262.51431
279653       127       72     6       927  2700   56401.68133
247218       161       20     1     66502  5637   54053.42396
…            …       …    …         …     …           …
279212        43       72     6       769  1847   20889.14617
277856        91       42     5     19422  4155  118896.06830
233000        87        8     1     80653  2829   88878.57034
```

```
287425       439       48      2      76034   11542   167148.83770
265371         3       32      1      89123    3726    54886.07827

        second_mortgage  home_equity     debt  hs_degree  age_median  pct_own  \
UID
267822          0.02077      0.08919  0.52963    0.89288   44.666665  0.79046
246444          0.02222      0.04274  0.60855    0.90487   34.791665  0.52483
245683          0.00000      0.09512  0.73484    0.94288   41.833330  0.85331
279653          0.01086      0.01086  0.52714    0.91500   49.750000  0.65037
247218          0.05426      0.05426  0.51938    1.00000   22.000000  0.13046
…                  …           …      …        …          …        …
279212          0.00000      0.00000  0.11694    0.60155   40.916670  0.60422
277856          0.02112      0.19641  0.65364    0.95737   39.166665  0.68072
233000          0.02024      0.07857  0.58095    0.93555   44.166665  0.78508
287425          0.07550      0.12556  0.65722    0.98540   45.041670  0.93970
265371          0.01412      0.18362  0.65537    0.87370   31.166665  0.27912

        married  separated  divorced
UID
267822  0.57851    0.01240   0.08770
246444  0.34886    0.01426   0.09030
245683  0.64745    0.01607   0.10657
279653  0.47257    0.02021   0.10106
247218  0.12356    0.00000   0.03109
…          …          …        …
279212  0.24603    0.02249   0.14683
277856  0.61127    0.02473   0.04888
233000  0.70451    0.00520   0.07712
287425  0.75503    0.00915   0.05261
265371  0.34426    0.03005   0.13320

[27321 rows x 15 columns]
```

[ ]:
```python
y_train=df_train['hc_mortgage_mean']
```

[ ]:
```python
y_train
```

[ ]:
```
UID
267822    1414.80295
246444     864.41390
245683    1506.06758
279653    1175.28642
247218    1192.58759
              …
279212     770.11560
277856    2210.84055
233000    1671.07908
```

```
287425    3074.83088
265371    1455.42340
Name: hc_mortgage_mean, Length: 27321, dtype: float64
```

[ ]:
```
x_test=df_test[input_cols]
y_test=df_test['hc_mortgage_mean']
```

[ ]:
```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
```

[ ]:
```
x_train_scaled=sc.fit_transform(x_train)
```

[ ]:
```
x_test_scaled=sc.fit_transform(x_test)
```

[ ]:
```
#apply linear regression model
from sklearn.linear_model import LinearRegression
linear_reg=LinearRegression()
```

[ ]:
```
linear_reg.fit(x_train_scaled,y_train)
```

[ ]:
```
LinearRegression()
```

[ ]:
```
y_pred=linear_reg.predict(x_test_scaled)
```

[ ]:
```
y_pred
```

[ ]:
```
array([ 874.67481013, 1597.10903054, 1086.41351981, …, 1915.00495942,
       1505.10480889, 1151.68011643])
```

[ ]:
```
from sklearn.metrics import mean_squared_error,r2_score,accuracy_score
print('Mean Squared error',np.sqrt(mean_squared_error(y_test,y_pred)))
```

```
Mean Squared error 325.0919574748077
```

[ ]:
```
df_train["STATEID"].unique()
```

[ ]:
```
array([36, 18, 72, 20,  1, 48, 45,  6,  5, 24, 17, 19, 47, 32, 22,  8, 44,
       28, 34, 41,  4, 12, 55, 42, 37, 51, 26, 39, 40, 13, 16, 46, 27, 29,
       53, 56,  9, 54, 21, 25, 11, 15, 30,  2, 33, 49, 50, 31, 38, 35, 23,
       10])
```

[ ]:
```
for i in [20,1,45]:
    print('state id-->',i)
    x_train_nation=df_train[df_train['COUNTYID']==i][input_cols]
    y_train_nation=df_train[df_train['COUNTYID']==i]['hc_mortgage_mean']

    x_test_nation=df_test[df_test['COUNTYID']==i][input_cols]
```

```
    y_test_nation=df_test[df_test['COUNTYID']==i]['hc_mortgage_mean']

    x_train_scaled_nation=sc.fit_transform(x_train_nation)
    x_test_scaled_nation=sc.fit_transform(x_test_nation)

    linear_reg.fit(x_train_scaled_nation,y_train_nation)
    yprd=linear_reg.predict(x_test_scaled_nation)

    print('root Mean Squared error',np.
↪sqrt(mean_squared_error(y_test_nation,yprd)))
    print('R2 score',r2_score(y_test_nation,yprd))
```

```
state id--> 20
root Mean Squared error 307.9718899931471
R2 score 0.6046603766461811
state id--> 1
root Mean Squared error 307.7896199248688
R2 score 0.8104850042868166
state id--> 45
root Mean Squared error 225.62754461084364
R2 score 0.7888730697076223
```

[ ]: