Live Coding - Module 11

Rachel Holman

In [1]:

```
import numpy as np
import pandas as pd
import prince
from scipy import stats
import os
from ydata_profiling import ProfileReport
from matplotlib import pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:

```
ahs = pd.read_csv('ahs_cleaned-1.csv', na_values=[-6, "'-9'"])
```

profile = ProfileReport(ahs, title = 'American Housing Survey EDA', html = {'style': {'full_width': True}}, minimal = False) profile.to_notebook_iframe()

In [3]:

```
ahs['HINCP']
```

Out[3]:

```
0        257000.0
1        201000.0
2             NaN
3         66900.0
4         35000.0
           ...   
63180     74000.0
63181    207000.0
63182    158100.0
63183    130200.0
63184    120000.0
Name: HINCP, Length: 63185, dtype: float64
```

In [4]:

```
ahs['RODENT'].value_counts()
```

Out[4]:

```
No signs in the last 12 months             48821
Seen a few times in the last 12 months      4212
Seen monthly in the last 12 months           522
Seen daily in the last 12 months             474
Seen weekly in the last 12 months            426
Name: RODENT, dtype: int64
```

In [5]:

```
ahs.groupby("RODENT").agg({'HINCP':'mean'})
```

Out[5]:

	HINCP
RODENT	
No signs in the last 12 months	87738.246779
Seen a few times in the last 12 months	86156.387464
Seen daily in the last 12 months	51274.924051
Seen monthly in the last 12 months	82798.544061
Seen weekly in the last 12 months	64086.826291

In [6]:

```
stats.f_oneway(ahs.query("RODENT=='No signs in the last 12 months'").HINCP.drop
              ahs.query("RODENT=='Seen a few times in the last 12 months'").HIN
              ahs.query("RODENT=='Seen daily in the last 12 months'").HINCP.dro
              ahs.query("RODENT=='Seen monthly in the last 12 months'").HINCP.d
              ahs.query("RODENT=='Seen weekly in the last 12 months'").HINCP.dr
```

Out[6]:

```
F_onewayResult(statistic=21.68467615110672, pvalue=6.703833330074091e-18)
```

In [7]:

```
ahs['YRBUILT'].value_counts()
```

Out[7]:

```
1970    9313
1980    9072
2000    8883
1990    7863
1960    6860
1950    6330
1919    3594
1940    3001
1920    2494
1930    1699
2010     846
2017     503
2016     489
2015     486
2014     444
2013     340
2012     320
2018     269
2011     251
2019     128
Name: YRBUILT, dtype: int64
```

In [8]:

```
ahs['MARKETVAL'].describe()
```

Out[8]:

```
count    3.839000e+04
mean     3.762769e+05
std      5.537866e+05
min      1.000000e+03
25%      1.404465e+05
50%      2.552730e+05
75%      4.359682e+05
max      9.999998e+06
Name: MARKETVAL, dtype: float64
```

In [9]:

```
ahs[['MARKETVAL', 'YRBUILT']].corr()
```

Out[9]:

	MARKETVAL	YRBUILT
MARKETVAL	1.00000	-0.00403
YRBUILT	-0.00403	1.00000

In [10]:

```
ahs2 = ahs[['MARKETVAL', 'YRBUILT']].dropna()
stats.pearsonr(ahs2['MARKETVAL'], ahs2['YRBUILT'])
```

Out[10]:

```
PearsonRResult(statistic=-0.004029500232993559, pvalue=0.4298243664117904)
```

In [11]:

```
ahs.columns
```

Out[11]:

```
Index(['Unnamed: 0', 'DIVISION', 'TENURE', 'YRBUILT', 'UNITSIZE', 'HSHLDTYPE',
       'HHRACE', 'HHSEX', 'HINCP', 'TOTHCAMT', 'MARKETVAL', 'MAINTAMT',
       'FUSEBLOW', 'SEWBREAK', 'ROACH', 'RODENT', 'NOWIRE', 'PLUGS', 'COLD',
       'NOTOIL', 'NOWAT', 'FLOORHOLE', 'FNDCRUMB', 'PAINTPEEL', 'ROOFHOLE',
       'ROOFSAG', 'ROOFSHIN', 'WALLCRACK', 'WALLSIDE', 'WALLSLOPE', 'WINBOAR
D',
       'WINBROKE', 'LEAKI', 'MOLDBATH'],
      dtype='object')
```

In [12]:

```
broken = ahs[['FUSEBLOW', 'SEWBREAK', 'ROACH', 'RODENT', 'NOWIRE', 'PLUGS', 'CO
        'NOTOIL', 'NOWAT', 'FLOORHOLE', 'FNDCRUMB', 'PAINTPEEL', 'ROOFHOLE',
        'ROOFSAG', 'ROOFSHIN', 'WALLCRACK', 'WALLSIDE', 'WALLSLOPE', 'WINBOARD',
        'WINBROKE', 'LEAKI', 'MOLDBATH']].dropna()
```

In [13]:

```
MCA = prince.MCA(n_components=2)
MCA = MCA.fit(broken)
```

In [14]:

```
pd.set_option('display.max_rows', 100)
MCA.column_coordinates(broken).sort_values(1)
```

Out[14]:

	0	1
WALLSLOPE_Broken	4.859631	-3.112322
ROOFSAG_Broken	3.902633	-2.395702
ROOFHOLE_Broken	4.146965	-2.057354
ROOFSHIN_Broken	2.383521	-1.497986
WALLSIDE_Broken	3.081901	-1.304344
WINBOARD_Broken	3.453411	-0.900215
FNDCRUMB_Broken	1.821282	-0.490842
FLOORHOLE_Broken	4.284035	-0.411352
WINBROKE_Broken	2.177871	-0.299583
LEAKI_Not broken	-0.080362	-0.090882
FUSEBLOW_No fuses / breakers blown in the last 3 months	-0.071363	-0.073826
RODENT_No signs in the last 12 months	-0.115874	-0.066227
COLD_Not broken	-0.083472	-0.065144
NOTOIL_Not broken	-0.024595	-0.052225
NOWAT_Not broken	-0.022932	-0.046835
SEWBREAK_No breakdowns in the last 3 months	-0.024639	-0.045126
PAINTPEEL_Broken	3.404749	-0.039555
ROACH_No signs in the last 12 months	-0.080402	-0.025697
MOLDBATH_Not broken	-0.032766	-0.015399
WALLCRACK_Not broken	-0.109388	-0.008917
PLUGS_Not broken	-0.018019	-0.004301
NOWIRE_Not broken	-0.017618	-0.004205
PAINTPEEL_Not broken	-0.059399	0.000690
FLOORHOLE_Not broken	-0.042145	0.004047
WINBOARD_Not broken	-0.031694	0.008262
WINBROKE_Not broken	-0.076222	0.010485
FNDCRUMB_Not broken	-0.092293	0.024873
ROOFHOLE_Not broken	-0.054295	0.026936
WALLSLOPE_Not broken	-0.046273	0.029635
WALLSIDE_Not broken	-0.072108	0.030518
ROOFSAG_Not broken	-0.060365	0.037056
ROOFSHIN_Not broken	-0.076180	0.047877
ROACH_Seen a few times in the last 12 months	0.343749	0.069489
NOWIRE_Broken	0.685447	0.163621
WALLCRACK_Broken	2.236701	0.182337

	0	1
RODENT_Seen monthly in the last 12 months	1.356893	0.223186
PLUGS_Broken	1.027688	0.245321
ROACH_Seen monthly in the last 12 months	0.917792	0.310836
RODENT_Seen a few times in the last 12 months	0.580349	0.388932
ROACH_Seen daily in the last 12 months	2.536539	0.468289
FUSEBLOW_2 fuses / breakers blown in the last 3 months	0.752378	0.614884
FUSEBLOW_1 fuse / breaker blown in the last 3 months	0.660546	0.632649
ROACH_Seen weekly in the last 12 months	1.212483	0.926285
COLD_Broken	1.363422	1.064051
RODENT_Seen weekly in the last 12 months	1.927000	1.175182
LEAKI_Broken	1.118381	1.264789
MOLDBATH_Broken	3.340400	1.569872
FUSEBLOW_3 fuses / breakers blown in the last 3 months	1.296569	1.682641
RODENT_Seen daily in the last 12 months	3.184849	1.834898
FUSEBLOW_4 or more fuses / breakers blown in the last 3 months	1.584176	1.930694
SEWBREAK_Sewage system broke down in the last 3 months, but never for 6 hours or more	0.679875	1.976740
SEWBREAK_Two breakdowns in the last 3 months for 6 hours or more	3.190900	2.130127
NOWAT_Broken	1.089729	2.225630
SEWBREAK_Four or more breakdowns in last three months for 6 hours or more	5.443718	2.839466
NOTOIL_Broken	2.186931	4.643638
SEWBREAK_One breakdown in the last 3 months for 6 hours or more	1.532887	5.369539
SEWBREAK_Three breakdowns in the last 3 months for 6 hours or more	4.994402	8.163568

In [15]:
```
indices = MCA.row_coordinates(broken).reset_index()
indices
```

Out[15]:

	index	0	1
0	0	-0.159315	-0.052107
1	1	-0.159315	-0.052107
2	3	-0.073491	0.033082
3	6	-0.072648	-0.020696
4	8	-0.159315	-0.052107
...	...	...	...
35512	63180	-0.159315	-0.052107
35513	63181	-0.107030	-0.034292
35514	63182	-0.159315	-0.052107
35515	63183	-0.159315	-0.052107
35516	63184	-0.159315	-0.052107

35517 rows × 3 columns

In [16]:

```
indices = indices.rename({0:'brokenness',
                          1:'structure_infrastructure'},
                        axis=1)
```

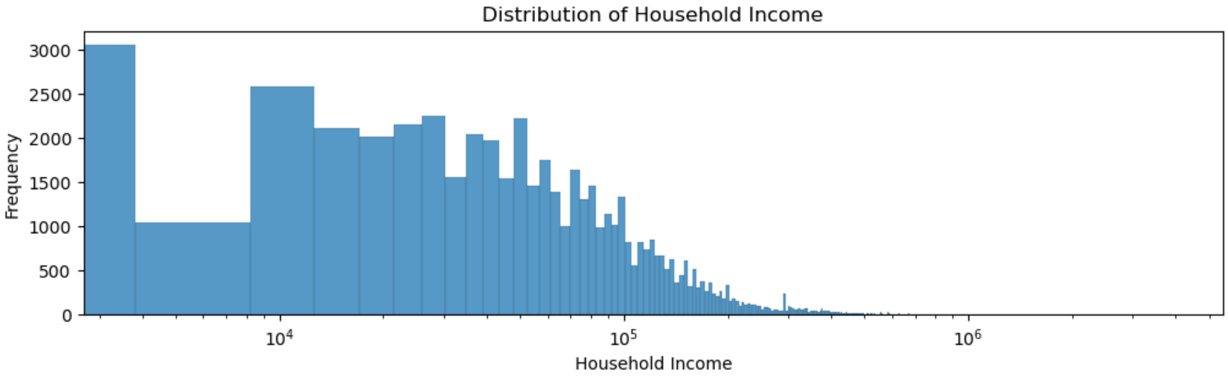
In [17]:

```
ahs_merge = pd.merge(ahs.reset_index(), indices,
                    how = 'outer',
                    on = 'index',
                    validate = 'one_to_one')
```

The distributions of income, race, ownership, and housing expenses

In [23]:

```
plt.figure(figsize = (12,3))
sns.histplot(x='HINCP', data=ahs_merge)
plt.xlabel("Household Income")
plt.ylabel("Frequency")
plt.title("Distribution of Household Income")
plt.xscale("log")
```



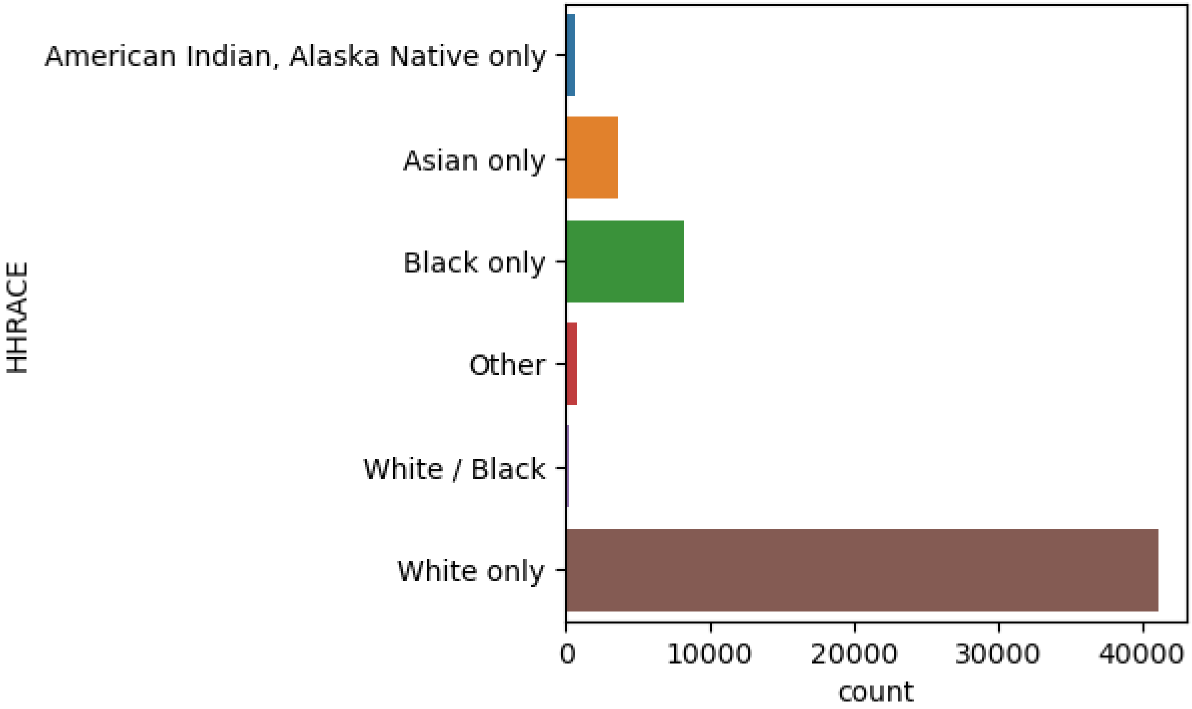
In [29]:

```
ahs_merge['HHRACE'].value_counts()
ahs_bar = ahs_merge.groupby('HHRACE').size().reset_index().rename({0:'count'},

plt.figure(figsize=(4,4))
sns.barplot(y='HHRACE', x='count', data=ahs_bar)
```

Out[29]:

```
<Axes: xlabel='count', ylabel='HHRACE'>
```



In [30]:

```
ahs_merge['TENURE'].value_counts()
```

Out[30]:

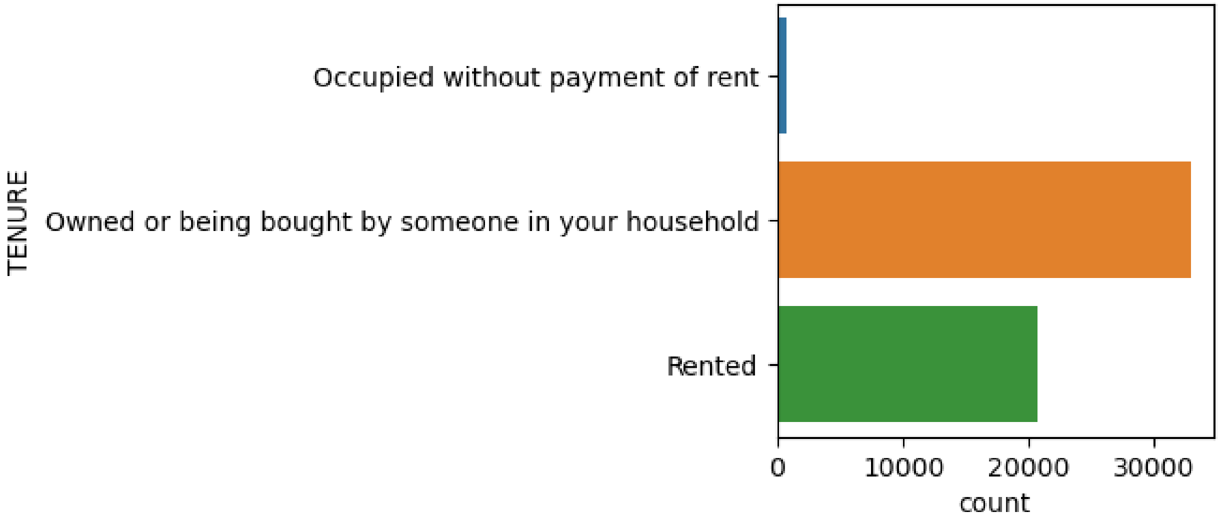
```
Owned or being bought by someone in your household    32972
Rented                                                20743
Occupied without payment of rent                        740
Name: TENURE, dtype: int64
```

In [32]:

```
ahs_bar = ahs_merge.groupby('TENURE').size().reset_index().rename({0:'count'},
plt.figure(figsize=(3,3))
sns.barplot(y='TENURE', x='count', data=ahs_bar)
```

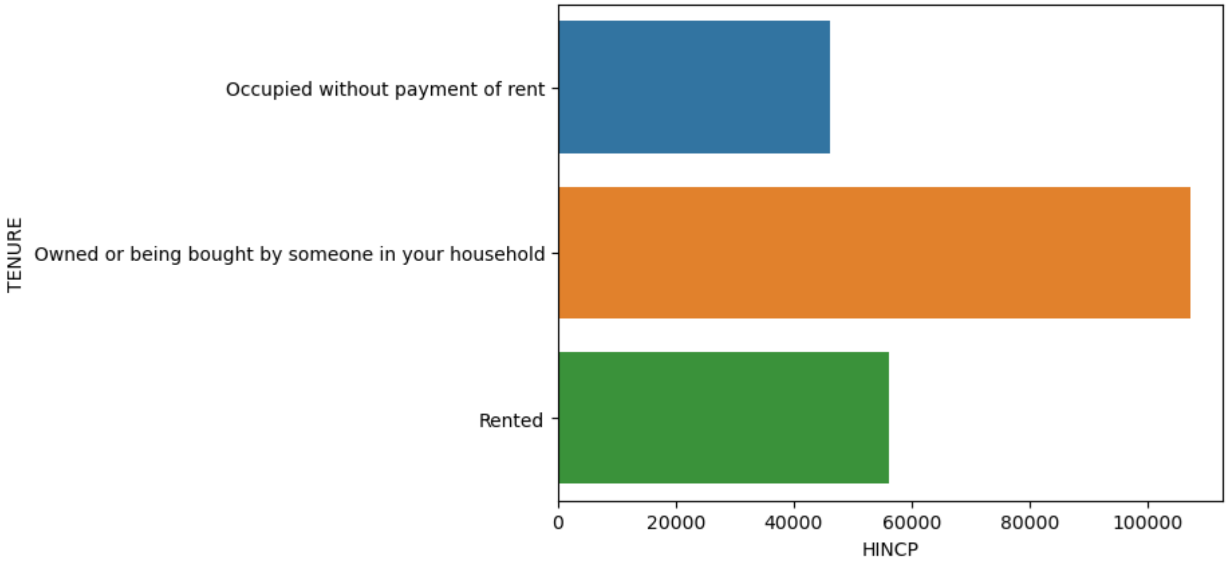
Out[32]:

```
<Axes: xlabel='count', ylabel='TENURE'>
```



In [33]:
```
ahs_bar = ahs_merge.groupby('TENURE').agg({'HINCP':'mean'}).reset_index()
sns.barplot(y='TENURE', x='HINCP', data=ahs_bar)
```

Out[33]:
```
<Axes: xlabel='HINCP', ylabel='TENURE'>
```

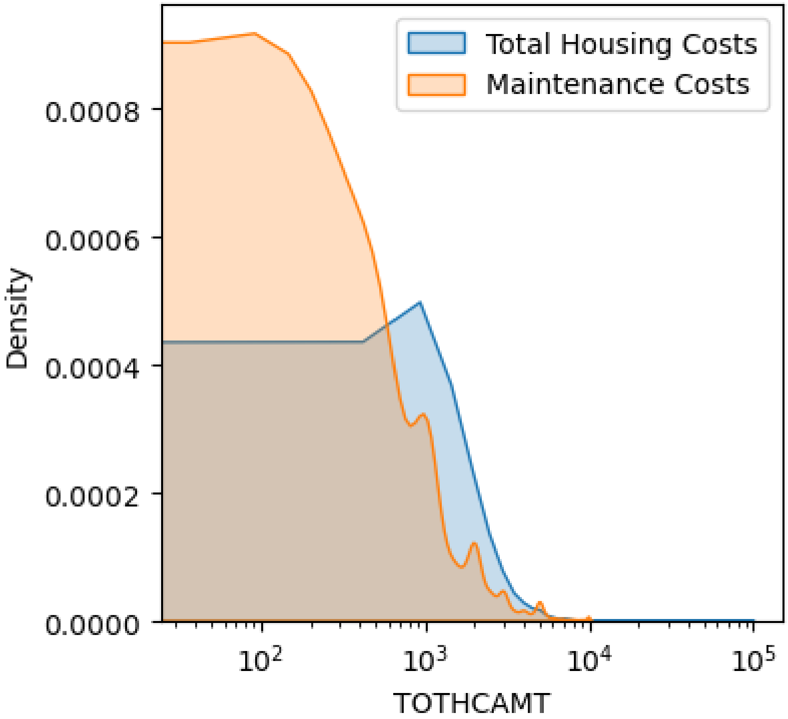


In [34]:
```
ahs_merge['TOTHCAMT'].describe()
```

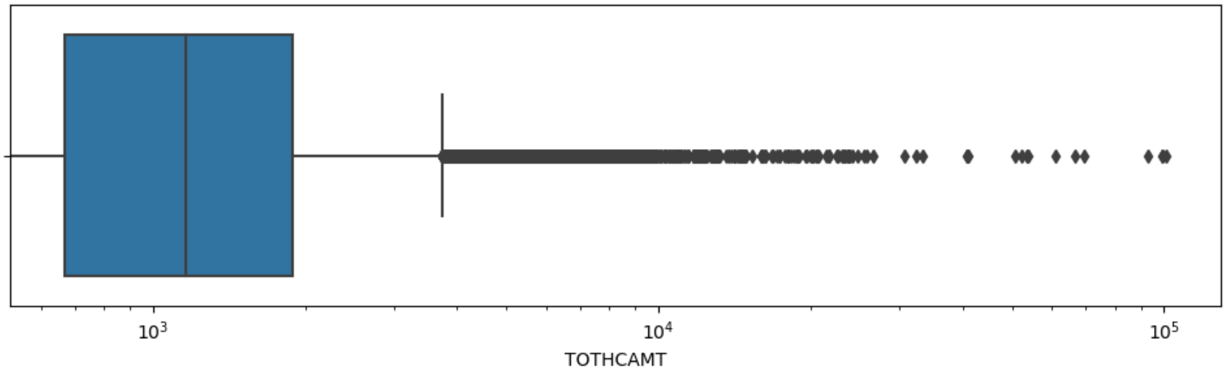
Out[34]:
```
count     54455.000000
mean       1517.628739
std        1783.335753
min           0.000000
25%         670.000000
50%        1164.000000
75%        1892.500000
max      100700.000000
Name: TOTHCAMT, dtype: float64
```

In [37]:
```
plt.figure(figsize = (4,4))
sns.kdeplot(x='TOTHCAMT', data=ahs_merge, fill=True, label='Total Housing Costs
sns.kdeplot(x='MAINTAMT', data=ahs_merge, fill=True, label='Maintenance Costs')
plt.xscale('log')
plt.legend()
```

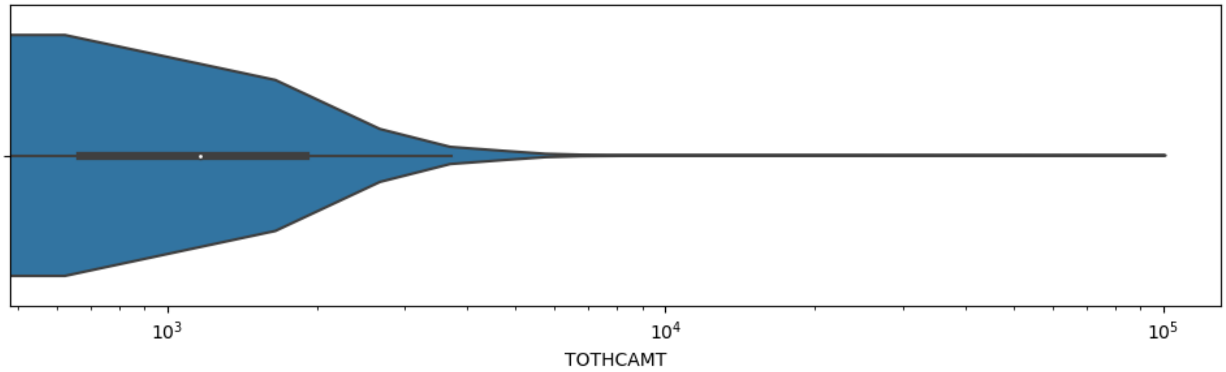
```
Out[37]: <matplotlib.legend.Legend at 0x2be249630>
```



```
In [40]: plt.figure(figsize=(12,3))
         sns.boxplot(x='TOTHCAMT', data=ahs_merge)
         plt.xscale('log')
```

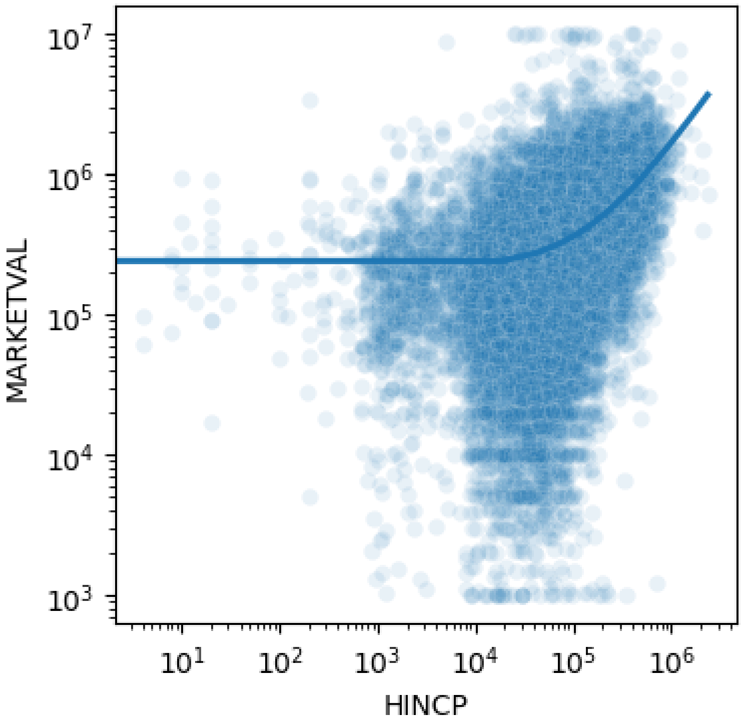


```
In [41]: plt.figure(figsize=(12,3))
         sns.violinplot(x='TOTHCAMT', data=ahs_merge)
         plt.xscale('log')
```



The relationship between these four features with each other, as well as the two indices of home disrepair that we built last week

In [46]:
```
ahs_merge[['HINCP','MARKETVAL']].corr()
plt.figure(figsize=(4,4))
sns.scatterplot(x='HINCP', y='MARKETVAL', data=ahs_merge, alpha=.1)
sns.regplot(x='HINCP', y='MARKETVAL', scatter=False, data=ahs_merge)
plt.xscale('log')
plt.yscale('log')
```



The time progression of average house price against year built

In [50]:
```
ahs_line = ahs_merge.groupby('YRBUILT').agg({'MARKETVAL':'mean'}).reset_index()
ahs_line
```

Out[50]:

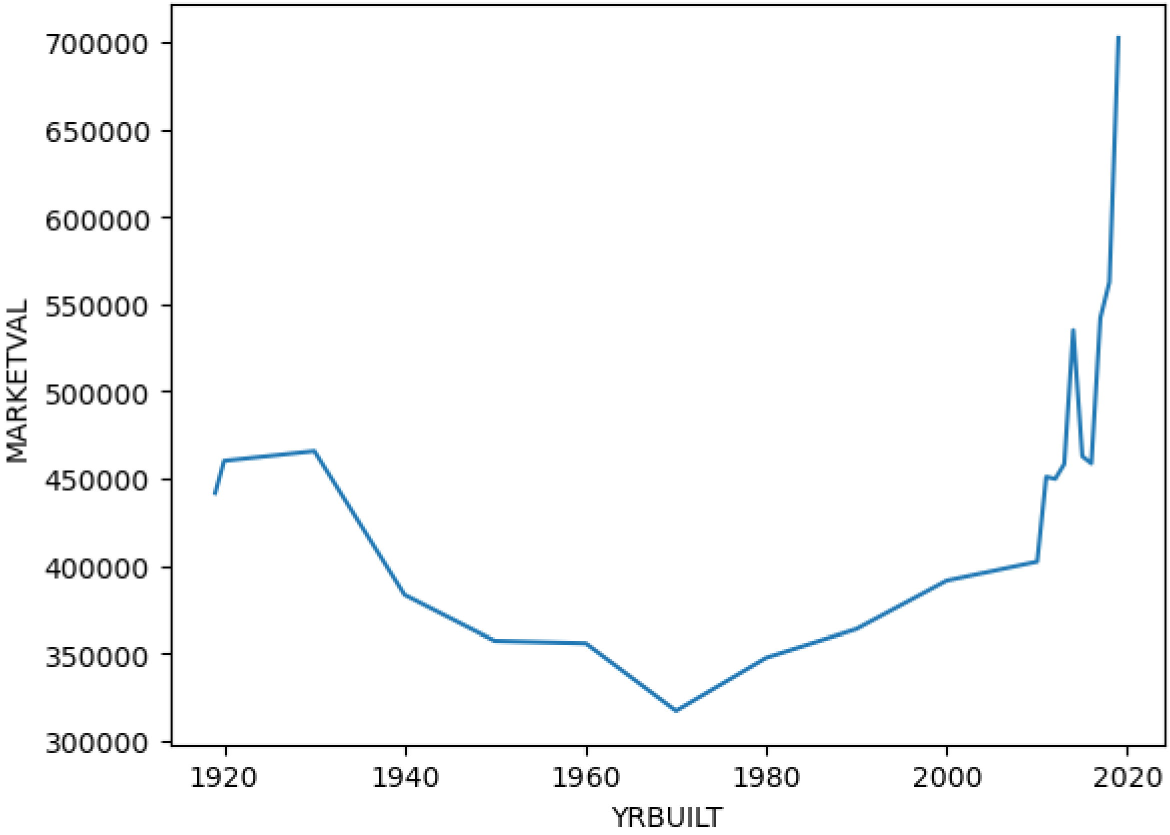
	YRBUILT	MARKETVAL
0	1919	441787.054923
1	1920	460242.827079
2	1930	465825.451481
3	1940	383461.994423
4	1950	357163.403623
5	1960	355834.658272
6	1970	317178.414418
7	1980	347564.607041
8	1990	363917.258788
9	2000	391545.044761
10	2010	402451.034810
11	2011	451071.786982
12	2012	449889.661836
13	2013	458420.516588
14	2014	535170.154122
15	2015	462851.737762
16	2016	458754.904605
17	2017	542170.035088
18	2018	562692.710526
19	2019	702470.655914

In [51]:

```
sns.lineplot(x='YRBUILT', y='MARKETVAL', data=ahs_line)
```

Out[51]:

```
<Axes: xlabel='YRBUILT', ylabel='MARKETVAL'>
```



Graph matrices in which each cell contains a graph that is specific to a census division or ownership status

In [52]:

```
ahs_facet = ahs_merge.assign(own = (ahs_merge['TENURE']=='Owned or being bought
ahs_facet = ahs_facet.groupby(['DIVISION', 'HHRACE']).agg({'own':'mean'}).reset
ahs_facet
```

Out[52]:

	DIVISION	HHRACE	own
0	East North Central	American Indian, Alaska Native only	0.347826
1	East North Central	Asian only	0.629496
2	East North Central	Black only	0.379967
3	East North Central	Other	0.484211
4	East North Central	White / Black	0.333333
5	East North Central	White only	0.728166
6	East South Central	American Indian, Alaska Native only	0.545455
7	East South Central	Asian only	0.640000
8	East South Central	Black only	0.332123
9	East South Central	Other	0.388889
10	East South Central	White / Black	0.200000
11	East South Central	White only	0.683044
12	Middle Atlantic	American Indian, Alaska Native only	0.500000
13	Middle Atlantic	Asian only	0.490683
14	Middle Atlantic	Black only	0.309598
15	Middle Atlantic	Other	0.422222
16	Middle Atlantic	White / Black	0.315789
17	Middle Atlantic	White only	0.632123
18	Mountain	American Indian, Alaska Native only	0.418919
19	Mountain	Asian only	0.640845
20	Mountain	Black only	0.323864
21	Mountain	Other	0.576271
22	Mountain	White / Black	0.263158
23	Mountain	White only	0.665655
24	New England	American Indian, Alaska Native only	0.428571
25	New England	Asian only	0.496732
26	New England	Black only	0.313364
27	New England	Other	0.357143
28	New England	White / Black	0.187500
29	New England	White only	0.636017
30	Pacific	American Indian, Alaska Native only	0.427083
31	Pacific	Asian only	0.573589
32	Pacific	Black only	0.355975
33	Pacific	Other	0.526132
34	Pacific	White / Black	0.314286

	DIVISION	HHRACE	own
35	Pacific	White only	0.580234
36	South Atlantic	American Indian, Alaska Native only	0.524752
37	South Atlantic	Asian only	0.653571
38	South Atlantic	Black only	0.453227
39	South Atlantic	Other	0.589744
40	South Atlantic	White / Black	0.394737
41	South Atlantic	White only	0.676737
42	West North Central	American Indian, Alaska Native only	0.409091
43	West North Central	Asian only	0.612245
44	West North Central	Black only	0.224852
45	West North Central	Other	0.652174
46	West North Central	White / Black	0.600000
47	West North Central	White only	0.697686
48	West South Central	American Indian, Alaska Native only	0.550725
49	West South Central	Asian only	0.642229
50	West South Central	Black only	0.386050
51	West South Central	Other	0.481013
52	West South Central	White / Black	0.411765
53	West South Central	White only	0.655397

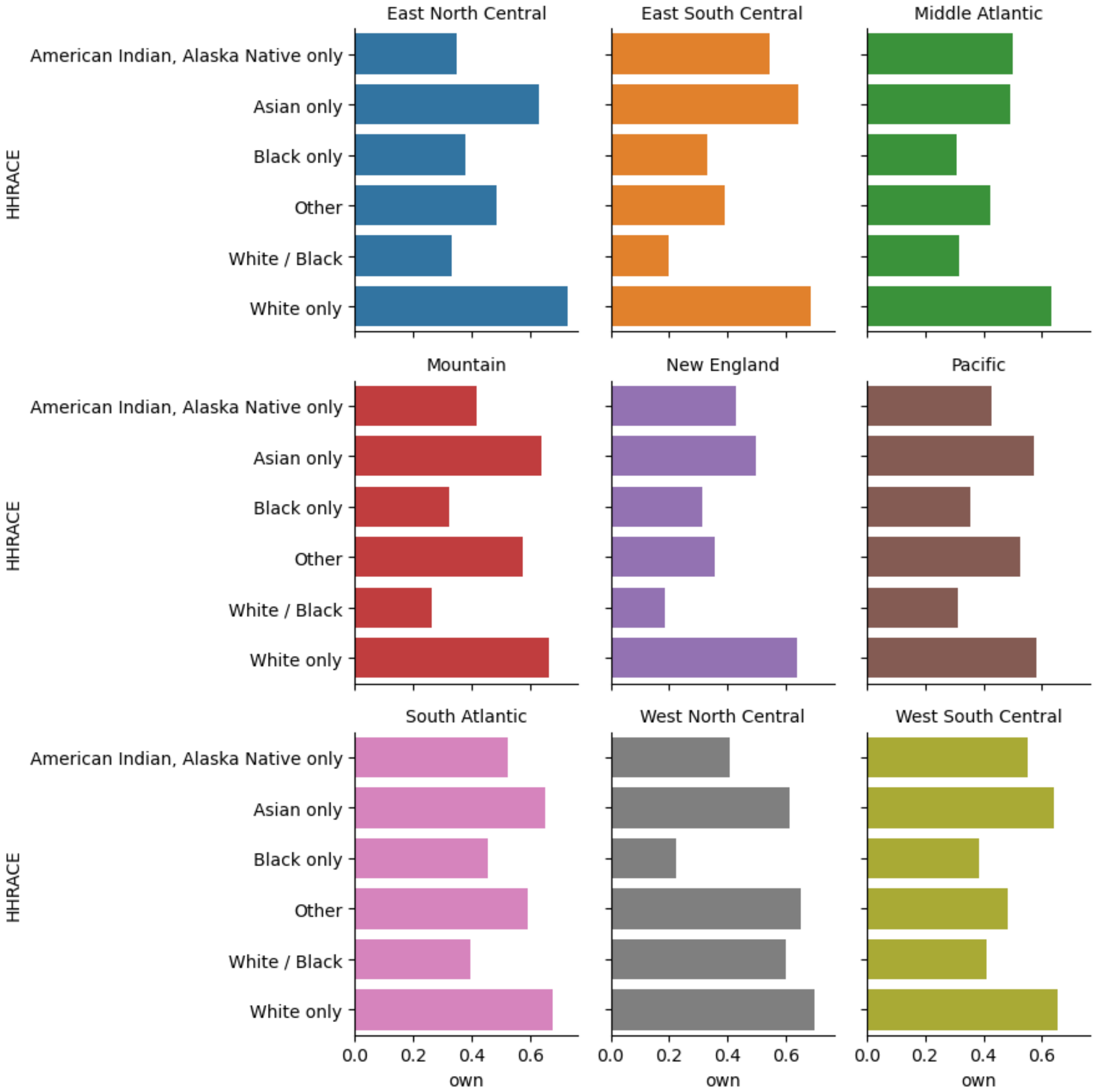
In [59]:

```
g = sns.FacetGrid(ahs_facet, col='DIVISION', col_wrap = 3, hue='DIVISION')
g.map(sns.barplot, 'own', 'HHRACE')
g.set_titles('{col_name}')
```

```
/Users/rachelholman/anaconda3/lib/python3.10/site-packages/seaborn/axisgrid.p
y:712: UserWarning: Using the barplot function without specifying `order` is l
ikely to produce an incorrect plot.
  warnings.warn(warning)
```

Out[59]:

```
<seaborn.axisgrid.FacetGrid at 0x2d8496b00>
```


East North Central
East South Central
Middle Atlantic
Mountain
New England
Pacific
South Atlantic
West North Central
West South Central
HHRACE
American Indian, Alaska Native only
Asian only
Black only
Other
White / Black
White only
0.0
0.2
0.4
0.6
own

```
In [ ]:
```