In [1]:

```
import pandas as pd
import numpy as np
import scipy.stats as sps
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn as skl
from sklearn import preprocessing
from sklearn.decomposition import PCA
%matplotlib inline
```

In [2]:

```
%%time
fa_dir = '/Users/stevecoggeshall/Documents/Teaching/Fraud Analytics/2018 USC fra
ud class'
property_data = pd.read_csv(fa_dir + '/data/NY property/NY property 1 million.cs
v', index_col=0)
```

```
CPU times: user 8.69 s, sys: 982 ms, total: 9.68 s
Wall time: 10.6 s
```

In [3]:

```
property_data.dtypes
```

```
Out[3]:

BBLE          object
BLOCK          int64
LOT            int64
EASEMENT      object
OWNER         object
BLDGCL        object
TAXCLASS      object
LTFRONT        int64
LTDEPTH        int64
STORIES      float64
FULLVAL        int64
AVLAND         int64
AVTOT          int64
EXLAND         int64
EXTOT          int64
EXCD1        float64
STADDR        object
ZIP          float64
EXMPTCL       object
BLDFRONT       int64
BLDDEPTH       int64
AVLAND2      float64
AVTOT2       float64
EXLAND2      float64
EXTOT2       float64
EXCD2        float64
PERIOD        object
YEAR          object
VALTYPE       object
dtype: object
```

Calculate means for AVTOT, AVLAND, FULLVAL by taxclass, avoiding the records with zeros

```
In [4]:
```

```
property_data.head().transpose()
```

```
Out[4]:
```

RECORD	**1**	**2**	**3**	**4**	**5**
BBLE	3046020035	5046820019	3074790028	4027980132	1006950027E
BLOCK	4602	4682	7479	2798	695
LOT	35	19	28	132	27
EASEMENT	NaN	NaN	NaN	NaN	E
OWNER	DESMOND CAMPBELL	CINISOMO MARIO	GANGICHIODO DONALD	DCAS	CONRAIL

	CAMPBELL	MARIO	DONALD		
BLDGCL	B1	A5	V0	V0	U6
TAXCLASS	1	1	1B	1B	3
LTFRONT	18	25	16	21	0
LTDEPTH	100	100	19	75	0
STORIES	2	3	NaN	NaN	NaN
FULLVAL	407000	415000	128000	112613	0
AVLAND	12337	13301	81	1940	0
AVTOT	19537	21312	81	1940	0
EXLAND	1620	1620	0	0	0
EXTOT	1620	1620	0	0	0
EXCD1	1017	1017	NaN	NaN	NaN
STADDR	140 EAST 49 STREET	537 AMHERST AVENUE	COYLE STREET	MAZEAU STREET	WEST 23 STREET
ZIP	11203	10306	NaN	NaN	NaN
EXMPTCL	X7	NaN	NaN	NaN	NaN
BLDFRONT	18	14	0	0	0
BLDDEPTH	36	51	0	0	0
AVLAND2	NaN	NaN	NaN	NaN	NaN
AVTOT2	NaN	NaN	NaN	NaN	NaN
EXLAND2	NaN	NaN	NaN	NaN	NaN
EXTOT2	NaN	NaN	NaN	NaN	NaN
EXCD2	NaN	NaN	NaN	NaN	NaN
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR

In [5]:

```
numrecords = len(property_data)
```

In [6]:

```
%%time
temp =property_data[property_data['FULLVAL']!=0]
mean_fullval = temp.groupby('TAXCLASS')['FULLVAL'].mean()
print(mean_fullval)
```

```
TAXCLASS
1       570486
1A      337564
1B      548322
1C      761535
1D    22336137
2       799812
2A      864085
2B     1253078
2C      772879
3       111276
4      3254843
Name: FULLVAL, dtype: int64
CPU times: user 286 ms, sys: 126 ms, total: 413 ms
Wall time: 429 ms
```

In [7]:

```
%%time
property_data['AVLAND'].replace('NaN',0)
temp_avland = property_data[property_data['AVLAND']!=0]
mean_avland = temp_avland.groupby('TAXCLASS')['AVLAND'].mean()
print(mean_avland)
```

```
TAXCLASS
1      14833.974370
1A      2244.771009
1B     14746.500023
1C      8225.658898
1D    709303.793103
2      90830.637240
2A     31364.009044
2B     54443.666619
2C     25639.066531
3      43368.352941
4     606593.741300
Name: AVLAND, dtype: float64
CPU times: user 313 ms, sys: 142 ms, total: 455 ms
Wall time: 609 ms
```

In [8]:

```
%%time
property_data['AVTOT'].replace('NaN',0)
temp_avtot = property_data[property_data['AVTOT']!=0]
mean_avtot = temp_avland.groupby('TAXCLASS')['AVTOT'].mean()
print(mean_avtot)
```

```
TAXCLASS
1     2.489733e+04
1A    1.442187e+04
1B    1.474966e+04
1C    2.898447e+04
1D    1.166866e+06
2     3.599188e+05
2A    7.961781e+04
2B    1.785866e+05
2C    1.170552e+05
3     5.007444e+04
4     1.508998e+06
Name: AVTOT, dtype: float64
CPU times: user 302 ms, sys: 142 ms, total: 444 ms
Wall time: 573 ms
```

In [9]:

```
%%time
temp_test = property_data[property_data['AVTOT']==0]
```

```
CPU times: user 22.5 ms, sys: 3.7 ms, total: 26.2 ms
Wall time: 27.9 ms
```

In [10]:

```
%%time
temp_test.head().transpose()
```

```
CPU times: user 1.04 ms, sys: 152 µs, total: 1.2 ms
Wall time: 1.09 ms
```

Out[10]:

RECORD	**5**	**230**	**414**	**435**	**493**
BBLE	1006950027E	4006037501	1007167503	3011047502	3056837503
BLOCK	695	603	716	1104	5683
LOT	27	7501	7503	7502	7503
EASEMENT	E	NaN	NaN	NaN	NaN
OWNER	CONRAIL	NaN	NaN	NaN	NaN
BLDGCL	U6	R0	R0	R0	R0
TAXCLASS	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

TAXCLASS	3	2	2	2	2
LTFRONT	0	100	66	25	40
LTDEPTH	0	80	92	100	100
STORIES	NaN	3	12	3	2.5
FULLVAL	0	0	0	0	0
AVLAND	0	0	0	0	0
AVTOT	0	0	0	0	0
EXLAND	0	0	0	0	0
EXTOT	0	0	0	0	0
EXCD1	NaN	NaN	NaN	NaN	NaN
STADDR	WEST 23 STREET	35-12 31 STREET	447 WEST 18 STREET	394 15 STREET	1219 56 STREET
ZIP	NaN	11106	10011	11215	11219
EXMPTCL	NaN	NaN	NaN	NaN	NaN
BLDFRONT	0	100	0	0	40
BLDDEPTH	0	80	0	0	50
AVLAND2	NaN	NaN	NaN	NaN	NaN
AVTOT2	NaN	NaN	NaN	NaN	NaN
EXLAND2	NaN	NaN	NaN	NaN	NaN
EXTOT2	NaN	NaN	NaN	NaN	NaN
EXCD2	NaN	NaN	NaN	NaN	NaN
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR

Substituting decent values for AVTOT, AVLAND, FULLVAL from averages by taxclass

In [11]:

```
%%time
for index in mean_fullval.index:
    property_data.loc[(property_data['FULLVAL']==0) & (property_data['TAXCLASS']
==index),'FULLVAL']=mean_fullval[index]
    property_data.loc[(property_data['AVLAND']==0) & (property_data['TAXCLASS']=
=index),'AVLAND']=mean_avland[index]
    property_data.loc[(property_data['AVTOT']==0) & (property_data['TAXCLASS']==
index),'AVTOT']=mean_avtot[index]
```

```
CPU times: user 2.43 s, sys: 411 ms, total: 2.85 s
Wall time: 3.01 s
```

In [12]:

```
property_data.head().transpose()
```

Out[12]:

RECORD	1	2	3	4	5
BBLE	3046020035	5046820019	3074790028	4027980132	1006950027E
BLOCK	4602	4682	7479	2798	695
LOT	35	19	28	132	27
EASEMENT	NaN	NaN	NaN	NaN	E
OWNER	DESMOND CAMPBELL	CINISOMO MARIO	GANGICHIODO DONALD	DCAS	CONRAIL
BLDGCL	B1	A5	V0	V0	U6
TAXCLASS	1	1	1B	1B	3
LTFRONT	18	25	16	21	0
LTDEPTH	100	100	19	75	0
STORIES	2	3	NaN	NaN	NaN
FULLVAL	407000	415000	128000	112613	111276
AVLAND	12337	13301	81	1940	43368.4
AVTOT	19537	21312	81	1940	50074.4
EXLAND	1620	1620	0	0	0
EXTOT	1620	1620	0	0	0
EXCD1	1017	1017	NaN	NaN	NaN
STADDR	140 EAST 49 STREET	537 AMHERST AVENUE	COYLE STREET	MAZEAU STREET	WEST 23 STREET

ZIP	11203	10306	NaN	NaN	NaN
EXMPTCL	X7	NaN	NaN	NaN	NaN
BLDFRONT	18	14	0	0	0
BLDDEPTH	36	51	0	0	0
AVLAND2	NaN	NaN	NaN	NaN	NaN
AVTOT2	NaN	NaN	NaN	NaN	NaN
EXLAND2	NaN	NaN	NaN	NaN	NaN
EXTOT2	NaN	NaN	NaN	NaN	NaN
EXCD2	NaN	NaN	NaN	NaN	NaN
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR

Fill in missing STORIES

In [13]:

```
temp = property_data[property_data['STORIES'].isnull()]
len(temp)
```

Out[13]:

```
52142
```

In [14]:

```
temp['TAXCLASS'].value_counts()
```

Out[14]:

```
1B    22191
4     20888
3      4543
2      3434
1       879
2C      138
2B       34
2A       30
1A        5
Name: TAXCLASS, dtype: int64
```

```
In [15]:
```

```
property_data['TAXCLASS'].value_counts()
```

```
Out[15]:
```

```
1     643774
2     188592
4     102281
2A     40558
1B     22193
1A     20899
2B     13962
2C     10795
3       4546
1C       946
1D        29
Name: TAXCLASS, dtype: int64
```

```
In [16]:
```

```
%%time
mean_stories = property_data.groupby('TAXCLASS')['STORIES'].mean()
print(mean_stories)
```

```
TAXCLASS
1      2.115100
1A     1.671647
1B     4.000000
1C     3.052748
1D     1.068966
2     16.096540
2A     2.844833
2B     4.004782
2C     4.745097
3      1.333333
4      5.474805
Name: STORIES, dtype: float64
CPU times: user 62.7 ms, sys: 18.4 ms, total: 81.1 ms
Wall time: 81.4 ms
```

```
In [17]:
```

```
temp.head().transpose()
```

```
Out[17]:
```

RECORD	3	4	5	19	28
BBLE	3074790028	4027980132	1006950027E	3039330053	5008600054
BLOCK	7479	2798	695	3933	860
LOT	28	132	27	53	54
EASEMENT	NaN	NaN	E	NaN	NaN

EASEMENT	NaN	NaN	E	NaN	NaN
OWNER	GANGICHIODO DONALD	DCAS	CONRAIL	SRI DURGA MANDIR INC	ALCORN, DAVID
BLDGCL	V0	V0	U6	V1	V0
TAXCLASS	1B	1B	3	4	1B
LTFRONT	16	21	0	17	25
LTDEPTH	19	75	0	70	100
STORIES	NaN	NaN	NaN	NaN	NaN
FULLVAL	128000	112613	111276	95200	135000
AVLAND	81	1940	43368.4	42840	1392
AVTOT	81	1940	50074.4	42840	1392
EXLAND	0	0	0	0	0
EXTOT	0	0	0	0	0
EXCD1	NaN	NaN	NaN	NaN	NaN
STADDR	COYLE STREET	MAZEAU STREET	WEST 23 STREET	2799 FULTON STREET	84 REAR RIDGE AVENUE
ZIP	NaN	NaN	NaN	11207	10304
EXMPTCL	NaN	NaN	NaN	NaN	NaN
BLDFRONT	0	0	0	0	0
BLDDEPTH	0	0	0	0	0
AVLAND2	NaN	NaN	NaN	22167	NaN
AVTOT2	NaN	NaN	NaN	22167	NaN
EXLAND2	NaN	NaN	NaN	NaN	NaN
EXTOT2	NaN	NaN	NaN	NaN	NaN
EXCD2	NaN	NaN	NaN	NaN	NaN
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR

In [18]:

```
len(property_data[property_data["STORIES"] == 0])
```

Out[18]:

```
0
```

In [19]:

```
%%time
property_data['STORIES']=property_data['STORIES'].fillna(value =0)
for index in mean_stories.index:
    property_data.loc[(property_data['STORIES'] == 0) & (property_data['TAXCLASS
']==index),'STORIES']=mean_stories[index]
```

```
CPU times: user 731 ms, sys: 86.9 ms, total: 818 ms
Wall time: 846 ms
```

In [20]:

```
property_data.head().transpose()
```

Out[20]:

RECORD	1	2	3	4	5
BBLE	3046020035	5046820019	3074790028	4027980132	1006950027E
BLOCK	4602	4682	7479	2798	695
LOT	35	19	28	132	27
EASEMENT	NaN	NaN	NaN	NaN	E
OWNER	DESMOND CAMPBELL	CINISOMO MARIO	GANGICHIODO DONALD	DCAS	CONRAIL
BLDGCL	B1	A5	V0	V0	U6
TAXCLASS	1	1	1B	1B	3
LTFRONT	18	25	16	21	0
LTDEPTH	100	100	19	75	0
STORIES	2	3	4	4	1.33333
FULLVAL	407000	415000	128000	112613	111276
AVLAND	12337	13301	81	1940	43368.4
AVTOT	19537	21312	81	1940	50074.4
EXLAND	1620	1620	0	0	0
EXTOT	1620	1620	0	0	0
EXCD1	1017	1017	NaN	NaN	NaN

STADDR	140 EAST 49 STREET	537 AMHERST AVENUE	COYLE STREET	MAZEAU STREET	WEST 23 STREET
ZIP	11203	10306	NaN	NaN	NaN
EXMPTCL	X7	NaN	NaN	NaN	NaN
BLDFRONT	18	14	0	0	0
BLDDEPTH	36	51	0	0	0
AVLAND2	NaN	NaN	NaN	NaN	NaN
AVTOT2	NaN	NaN	NaN	NaN	NaN
EXLAND2	NaN	NaN	NaN	NaN	NaN
EXTOT2	NaN	NaN	NaN	NaN	NaN
EXCD2	NaN	NaN	NaN	NaN	NaN
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR

Fill in LTFRONT, LTDEPTH, BLDDEPTH, BLDFRONT with averages by TAXCLASS

In [21]:

```
%%time
# as these 4 values do not have NAs, we just need to replace 0s.

# calculate groupwise average (1st replace 0 by NAs so they are not counted in c
alculating mean)

property_data.loc[property_data['LTFRONT']==0,'LTFRONT']=np.nan
property_data.loc[property_data['LTDEPTH']==0,'LTDEPTH']=np.nan
property_data.loc[property_data['BLDFRONT']==0,'BLDFRONT']=np.nan
property_data.loc[property_data['BLDDEPTH']==0,'BLDDEPTH']=np.nan

#calculate mean now (mean function ignores NAs but not 0s hence we converted 0 t
o NA)
mean_LTFRONT=property_data.groupby(property_data['TAXCLASS'])['LTFRONT'].mean()
mean_LTDEPTH=property_data.groupby(property_data['TAXCLASS'])['LTDEPTH'].mean()
mean_BLDFRONT=property_data.groupby(property_data['TAXCLASS'])['BLDFRONT'].mean(
)
mean_BLDDEPTH=property_data.groupby(property_data['TAXCLASS'])['BLDDEPTH'].mean(
)

#update values
for index in mean_LTFRONT.index:
    property_data.loc[(property_data['LTFRONT'].isnull()) & (property_data['TAXC
LASS']==index),'LTFRONT']=mean_LTFRONT[index]
    property_data.loc[(property_data['LTDEPTH'].isnull()) & (property_data['TAXC
LASS']==index),'LTDEPTH']=mean_LTDEPTH[index]
    property_data.loc[(property_data['BLDFRONT'].isnull()) & (property_data['TAX
CLASS']==index),'BLDFRONT']=mean_BLDFRONT[index]
    property_data.loc[(property_data['BLDDEPTH'].isnull()) & (property_data['TAX
CLASS']==index),'BLDDEPTH']=mean_BLDDEPTH[index]
```

```
CPU times: user 3.63 s, sys: 775 ms, total: 4.4 s
Wall time: 4.53 s
```

In [22]:

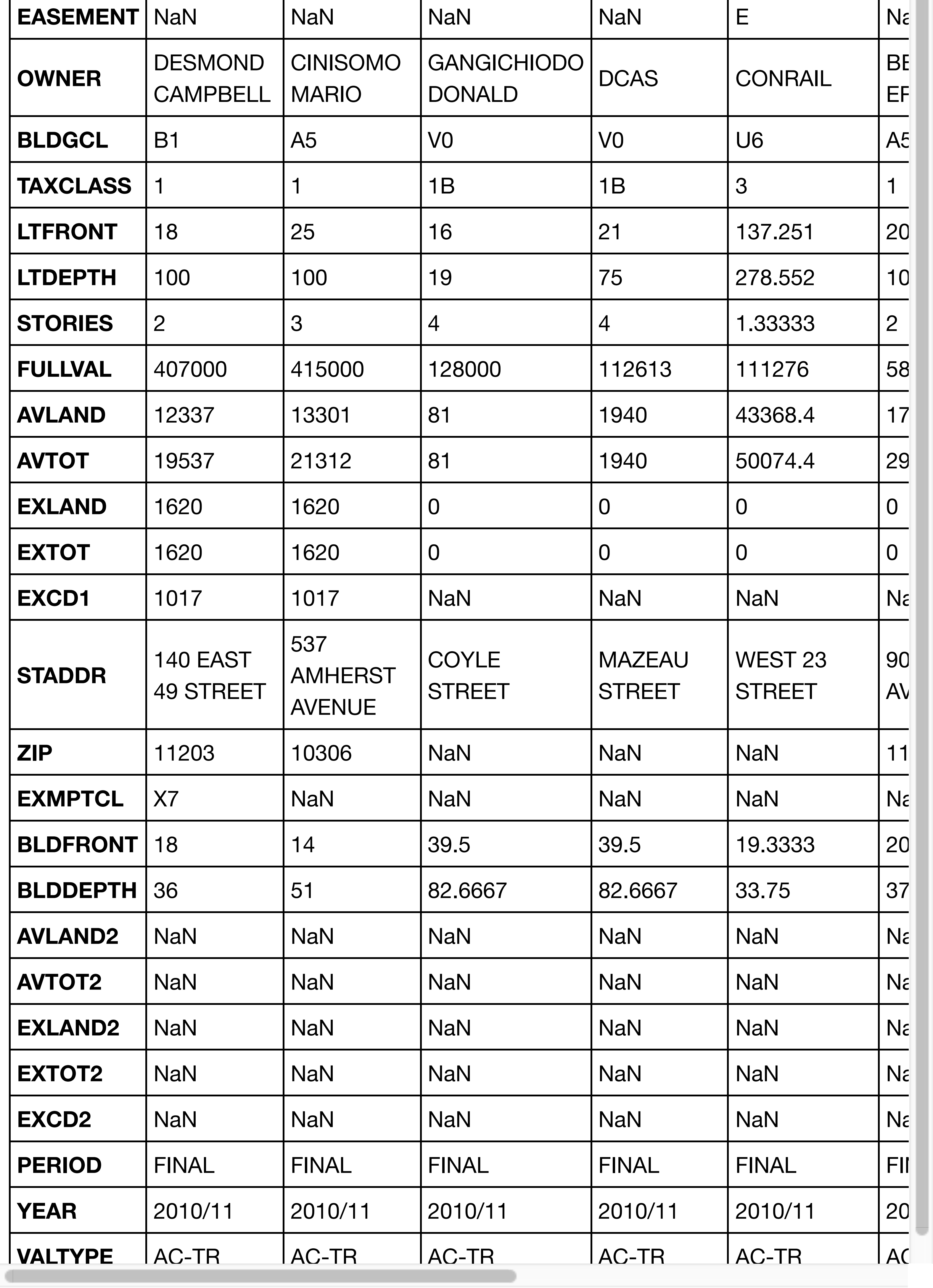
```
mydata = property_data
```

In [23]:

```
mydata.head(10).transpose()
```

Out[23]:

RECORD	1	2	3	4	5	6
BBLE	3046020035	5046820019	3074790028	4027980132	1006950027E	40
BLOCK	4602	4682	7479	2798	695	31
LOT	35	19	28	132	27	7



EASEMENT	NaN	NaN	NaN	NaN	E	Na
OWNER	DESMOND CAMPBELL	CINISOMO MARIO	GANGICHIODO DONALD	DCAS	CONRAIL	BE EF
BLDGCL	B1	A5	V0	V0	U6	A5
TAXCLASS	1	1	1B	1B	3	1
LTFRONT	18	25	16	21	137.251	20
LTDEPTH	100	100	19	75	278.552	10
STORIES	2	3	4	4	1.33333	2
FULLVAL	407000	415000	128000	112613	111276	58
AVLAND	12337	13301	81	1940	43368.4	17
AVTOT	19537	21312	81	1940	50074.4	29
EXLAND	1620	1620	0	0	0	0
EXTOT	1620	1620	0	0	0	0
EXCD1	1017	1017	NaN	NaN	NaN	Na
STADDR	140 EAST 49 STREET	537 AMHERST AVENUE	COYLE STREET	MAZEAU STREET	WEST 23 STREET	90 AV
ZIP	11203	10306	NaN	NaN	NaN	11
EXMPTCL	X7	NaN	NaN	NaN	NaN	Na
BLDFRONT	18	14	39.5	39.5	19.3333	20
BLDDEPTH	36	51	82.6667	82.6667	33.75	37
AVLAND2	NaN	NaN	NaN	NaN	NaN	Na
AVTOT2	NaN	NaN	NaN	NaN	NaN	Na
EXLAND2	NaN	NaN	NaN	NaN	NaN	Na
EXTOT2	NaN	NaN	NaN	NaN	NaN	Na
EXCD2	NaN	NaN	NaN	NaN	NaN	Na
PERIOD	FINAL	FINAL	FINAL	FINAL	FINAL	FII
YEAR	2010/11	2010/11	2010/11	2010/11	2010/11	20
VALTYPE	AC-TR	AC-TR	AC-TR	AC-TR	AC-TR	AC

In [24]:

```
mydata['borough'] = mydata['BBLE'].astype(str).str[0]
mydata['borough'] = mydata['borough'].astype(int)
mydata['borough'].value_counts()
```

Out[24]:

```
4    358046
3    323243
1    146221
5    113780
2    107285
Name: borough, dtype: int64
```

In [25]:

```
del mydata['YEAR']
del mydata['PERIOD']
del mydata['VALTYPE']
```

In [26]:

```
mydata['zip3'] = np.ones(numrecords)
mydata['lotarea'] = np.ones(numrecords)
mydata['bldarea'] = np.ones(numrecords)
mydata['bldvol'] = np.ones(numrecords)
mydata['fullval_la'] = np.ones(numrecords)
mydata['avland_la'] = np.ones(numrecords)
mydata['avtot_la'] = np.ones(numrecords)
mydata['fullval_ba'] = np.ones(numrecords)
mydata['avland_ba'] = np.ones(numrecords)
mydata['avtot_ba'] = np.ones(numrecords)
mydata['fullval_bv'] = np.ones(numrecords)
mydata['avland_bv'] = np.ones(numrecords)
mydata['avtot_bv'] = np.ones(numrecords)
```

In [27]:

```
mydata.shape
```

Out[27]:

```
(1048575, 40)
```

In [28]:

```
mydata['ZIP'].fillna(value = 0, inplace = True)
mydata['zip3'] = mydata['ZIP'] / 100
mydata['zip3'] = mydata['zip3'].astype(int)
```

In [29]:

```
%%time
mydata['lotarea'] = mydata['LTFRONT']*mydata['LTDEPTH']
mydata['bldarea'] = mydata['BLDFRONT']*mydata['BLDDEPTH']
mydata['bldvol'] = mydata['bldarea']*mydata['STORIES']
mydata['fullval_la'] = mydata['FULLVAL']/mydata['lotarea']
mydata['fullval_ba'] = mydata['FULLVAL']/mydata['bldarea']
mydata['fullval_bv'] = mydata['FULLVAL']/mydata['bldvol']
mydata['avland_la'] = mydata['AVLAND']/mydata['lotarea']
mydata['avland_ba'] = mydata['AVLAND']/mydata['bldarea']
mydata['avland_bv'] = mydata['AVLAND']/mydata['bldvol']
mydata['avtot_la'] = mydata['AVTOT']/mydata['lotarea']
mydata['avtot_ba'] = mydata['AVTOT']/mydata['bldarea']
mydata['avtot_bv'] = mydata['AVTOT']/mydata['bldvol']
```

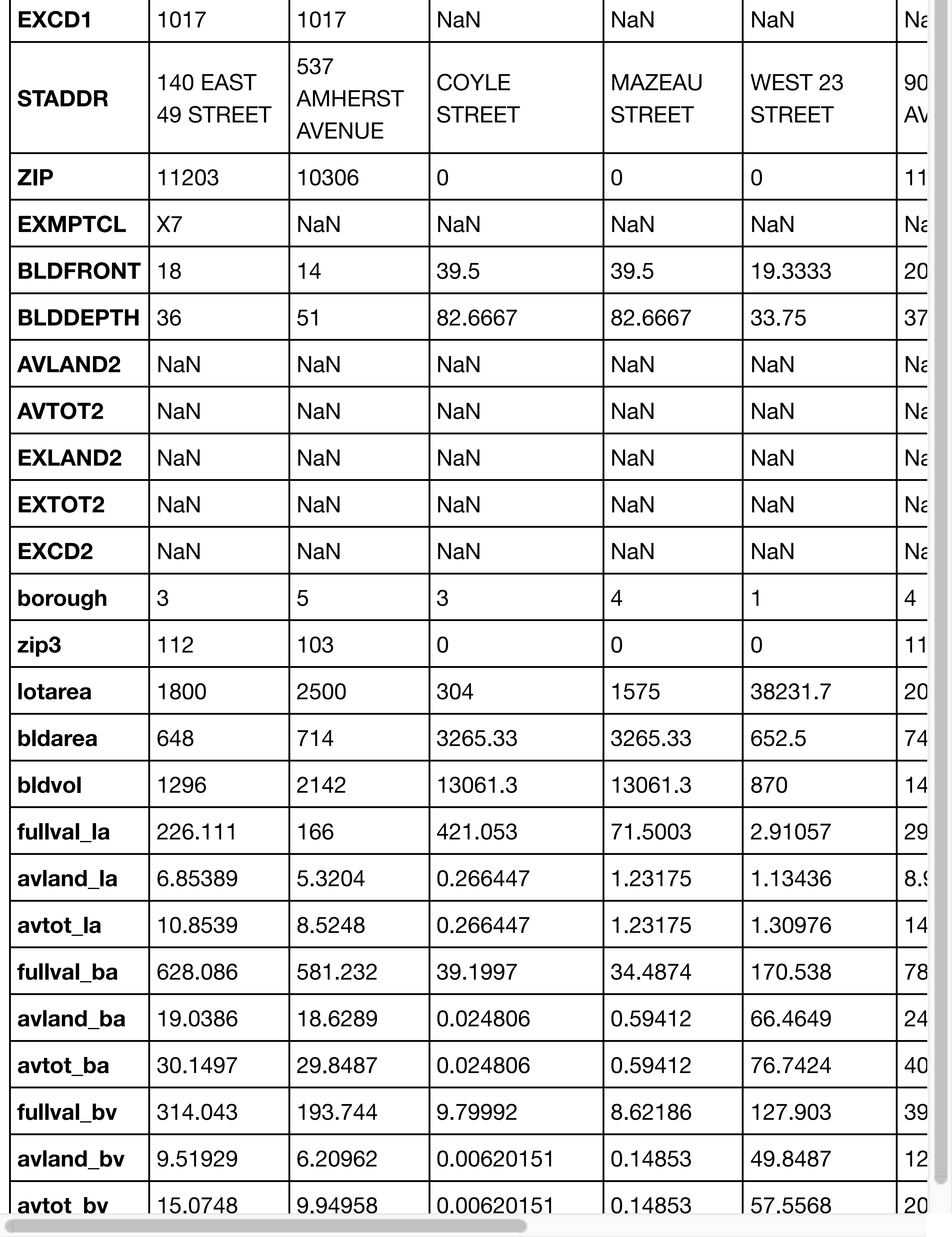
```
CPU times: user 158 ms, sys: 39.9 ms, total: 198 ms
Wall time: 135 ms
```

In [30]:

```
mydata.head(10).transpose()
```

Out[30]:

RECORD	1	2	3	4	5	6
BBLE	3046020035	5046820019	3074790028	4027980132	1006950027E	40
BLOCK	4602	4682	7479	2798	695	31
LOT	35	19	28	132	27	7
EASEMENT	NaN	NaN	NaN	NaN	E	Na
OWNER	DESMOND CAMPBELL	CINISOMO MARIO	GANGICHIODO DONALD	DCAS	CONRAIL	BE EF
BLDGCL	B1	A5	V0	V0	U6	A5
TAXCLASS	1	1	1B	1B	3	1
LTFRONT	18	25	16	21	137.251	20
LTDEPTH	100	100	19	75	278.552	10
STORIES	2	3	4	4	1.33333	2
FULLVAL	407000	415000	128000	112613	111276	58
AVLAND	12337	13301	81	1940	43368.4	17
AVTOT	19537	21312	81	1940	50074.4	29
EXLAND	1620	1620	0	0	0	0
EXTOT	1620	1620	0	0	0	0



EXCD1	1017	1017	NaN	NaN	NaN	Na
STADDR	140 EAST 49 STREET	537 AMHERST AVENUE	COYLE STREET	MAZEAU STREET	WEST 23 STREET	90 AV
ZIP	11203	10306	0	0	0	11
EXMPTCL	X7	NaN	NaN	NaN	NaN	Na
BLDFRONT	18	14	39.5	39.5	19.3333	20
BLDDEPTH	36	51	82.6667	82.6667	33.75	37
AVLAND2	NaN	NaN	NaN	NaN	NaN	Na
AVTOT2	NaN	NaN	NaN	NaN	NaN	Na
EXLAND2	NaN	NaN	NaN	NaN	NaN	Na
EXTOT2	NaN	NaN	NaN	NaN	NaN	Na
EXCD2	NaN	NaN	NaN	NaN	NaN	Na
borough	3	5	3	4	1	4
zip3	112	103	0	0	0	11
lotarea	1800	2500	304	1575	38231.7	20
bldarea	648	714	3265.33	3265.33	652.5	74
bldvol	1296	2142	13061.3	13061.3	870	14
fullval_la	226.111	166	421.053	71.5003	2.91057	29
avland_la	6.85389	5.3204	0.266447	1.23175	1.13436	8.
avtot_la	10.8539	8.5248	0.266447	1.23175	1.30976	14
fullval_ba	628.086	581.232	39.1997	34.4874	170.538	78
avland_ba	19.0386	18.6289	0.024806	0.59412	66.4649	24
avtot_ba	30.1497	29.8487	0.024806	0.59412	76.7424	40
fullval_bv	314.043	193.744	9.79992	8.62186	127.903	39
avland_bv	9.51929	6.20962	0.00620151	0.14853	49.8487	12
avtot_bv	15.0748	9.94958	0.00620151	0.14853	57.5568	20

In [31]:

```
mydata['TAXCLASS'].value_counts()
```

Out[31]:

```
1     643774
2     188592
4     102281
2A     40558
1B     22193
1A     20899
2B     13962
2C     10795
3       4546
1C       946
1D        29
Name: TAXCLASS, dtype: int64
```

In [32]:

```
%%time
zip3_means = mydata.groupby('zip3').mean()
```

```
CPU times: user 657 ms, sys: 531 ms, total: 1.19 s
Wall time: 1.22 s
```

In [33]:

```
%%time
zip5_means = mydata.groupby('ZIP').mean()
```

```
CPU times: user 431 ms, sys: 186 ms, total: 617 ms
Wall time: 632 ms
```

In [34]:

```
%%time
taxclass_means = mydata.groupby('TAXCLASS').mean()
```

```
CPU times: user 394 ms, sys: 150 ms, total: 544 ms
Wall time: 562 ms
```

In [35]:

```
%%time
borough_means = mydata.groupby('borough').mean()
```

```
CPU times: user 355 ms, sys: 147 ms, total: 502 ms
Wall time: 514 ms
```

In [36]:

```
borough_means.head(100)
```

Out[36]:

	BLOCK	LOT	LTFRONT	LTDEPTH	STORIES	FULLVAL
borough						
1	1101.490969	1111.573454	96.836600	126.788328	18.316241	2.388563e+06
2	4104.400764	503.720753	57.077353	110.305495	3.567672	6.431948e+05
3	4604.815919	265.551471	40.122141	103.947460	3.176697	7.072416e+05
4	7098.641599	174.028616	45.264387	106.106570	2.709955	6.440634e+05
5	2690.140332	205.173721	49.978920	109.265224	2.235807	4.879099e+05

5 rows × 32 columns

In [37]:

```
%%time
temp = mydata._get_numeric_data()
all_means = temp.mean()
```

```
CPU times: user 206 ms, sys: 151 ms, total: 357 ms
Wall time: 369 ms
```

In [38]:

```
all_means.head(100)
```

Out[38]:

```
BLOCK           4708.867421
LOT              370.092395
LTFRONT           52.591015
LTDEPTH          109.097356
STORIES            5.066400
FULLVAL       889772.221986
AVLAND         87386.536055
AVTOT         234748.371681
EXLAND         36811.788682
EXTOT          92543.814625
EXCD1           1604.500100
ZIP            10660.456077
BLDFRONT          38.855784
BLDDEPTH          60.426103
AVLAND2       246365.484475
AVTOT2        716078.713584
EXLAND2       351802.210545
EXTOT2        658114.779009
EXCD2           1371.659098
borough            3.177268
zip3             106.285080
lotarea         8075.638102
bldarea         3451.799745
bldvol         44161.087128
fullval_la       216.172994
avland_la         11.418543
avtot_la          28.118754
fullval_ba       573.123950
avland_ba         34.059076
avtot_ba          64.568788
fullval_bv       260.301444
avland_bv         15.946743
avtot_bv          25.615051
dtype: float64
```

In [39]:

```
zip3_means.head()
```

Out[39]:

	BLOCK	LOT	LTFRONT	LTDEPTH	STORIES	FULLVAL	AVL
zip3							
0	5135.068182	173.580551	88.636684	149.987283	4.270667	1.477644e+06	416
100	1108.540211	1040.787025	95.821529	126.151300	17.881549	2.447597e+06	342
101	1537.278118	1117.493877	97.194645	127.234899	24.501999	1.525579e+06	169
102	16.104406	3837.846758	115.473579	132.150804	27.760781	1.319686e+06	135
103	2690.460010	213.841540	47.425346	106.001898	2.097504	4.703097e+05	282

5 rows × 32 columns

In [40]:

```
%%time
zip5_means.loc[0] = all_means
zip3_means.loc[0] = all_means
```

```
CPU times: user 3.23 ms, sys: 1.14 ms, total: 4.36 ms
Wall time: 3.91 ms
```

In [41]:

```
zip3_means.head()
```

Out[41]:

	BLOCK	LOT	LTFRONT	LTDEPTH	STORIES	FULLVAL	A
zip3							
0	4708.867421	370.092395	52.591015	109.097356	5.066400	8.897722e+05	8
100	1108.540211	1040.787025	95.821529	126.151300	17.881549	2.447597e+06	3
101	1537.278118	1117.493877	97.194645	127.234899	24.501999	1.525579e+06	1
102	16.104406	3837.846758	115.473579	132.150804	27.760781	1.319686e+06	1
103	2690.460010	213.841540	47.425346	106.001898	2.097504	4.703097e+05	2

5 rows × 32 columns

Now the missing data has been reasonably filled in. Calculate the variables.

In [42]:

```
%%time
consolidated_means_dict = {
    k: {
        c: mydata[c].to_dict()
        for c in mydata.columns.values
    } for k, mydata in zip(
        ['zip3_means', 'zip5_means', 'taxclass_means', 'borough_means'],
        [zip3_means, zip5_means, taxclass_means, borough_means]
    )
}

# consolidated_means_dict['all_means'] = all_means.to_dict()

def calc_vars(row_data):
    izip5 = row_data['ZIP']
    izip3 = row_data['zip3']
    itc = row_data['TAXCLASS']
    ibo = row_data['borough']

    row_vars = pd.Series()

    row_vars['fv_la_z3'] = row_data['fullval_la']/consolidated_means_dict['zip3_means']['fullval_la'][izip3]
    row_vars['vl_la_z3'] = row_data['avland_la']/consolidated_means_dict['zip3_means']['avland_la'][izip3]
    row_vars['vt_la_z3'] = row_data['avtot_la']/consolidated_means_dict['zip3_means']['avtot_la'][izip3]
    row_vars['fv_la_z5'] = row_data['fullval_la']/consolidated_means_dict['zip5_means']['fullval_la'][izip5]
    row_vars['vl_la_z5'] = row_data['avland_la']/consolidated_means_dict['zip5_means']['avland_la'][izip5]
    row_vars['vt_la_z5'] = row_data['avtot_la']/consolidated_means_dict['zip5_means']['avtot_la'][izip5]
    row_vars['fv_la_tc'] = row_data['fullval_la']/consolidated_means_dict['taxclass_means']['fullval_la'][itc]
    row_vars['vl_la_tc'] = row_data['avland_la']/consolidated_means_dict['taxclass_means']['avland_la'][itc]
    row_vars['vt_la_tc'] = row_data['avtot_la']/consolidated_means_dict['taxclass_means']['avtot_la'][itc]
    row_vars['fv_la_bo'] = row_data['fullval_la']/consolidated_means_dict['borough_means']['fullval_la'][ibo]
    row_vars['vl_la_bo'] = row_data['avland_la']/consolidated_means_dict['borough_means']['avland_la'][ibo]
    row_vars['vt_la_bo'] = row_data['avtot_la']/consolidated_means_dict['borough_means']['avtot_la'][ibo]
    row_vars['fv_la_none'] = row_data['fullval_la']
    row_vars['vl_la_none'] = row_data['avland_la']
    row_vars['vl_la_none'] = row_data['avland_la']
    row_vars['vt_la_none'] = row_data['avtot_la']

    row_vars['fv_ba_z3'] = row_data['fullval_ba']/consolidated_means_dict['zip3_
means']['fullval_ba'] [izip3]
    row_vars['vl_ba_z3'] = row_data['avland_ba']/consolidated_means_dict['zip3_m
eans']['avland_ba'][izip3]
    row_vars['vt_ba_z3'] = row_data['avtot_ba']/consolidated_means_dict['zip3_me
ans']['avtot_ba'][izip3]
    row_vars['fv_ba_z5'] = row_data['fullval_ba']/consolidated_means_dict['zip5_
means']['fullval_ba'][izip5]
    row_vars['vl_ba_z5'] = row_data['avland_ba']/consolidated_means_dict['zip5_m
eans']['avland_ba'][izip5]
    row_vars['vt_ba_z5'] = row_data['avtot_ba']/consolidated_means_dict['zip5_me
ans']['avtot_ba'][izip5]
    row_vars['fv_ba_tc'] = row_data['fullval_ba']/consolidated_means_dict['taxcl
ass_means']['fullval_ba'][itc]
    row_vars['vl_ba_tc'] = row_data['avland_ba']/consolidated_means_dict['taxcla
ss_means']['avland_ba'][itc]
    row_vars['vt_ba_tc'] = row_data['avtot_ba']/consolidated_means_dict['taxclas
s_means']['avtot_ba'][itc]
    row_vars['fv_ba_bo'] = row_data['fullval_ba']/consolidated_means_dict['borou
gh_means']['fullval_ba'][ibo]
    row_vars['vl_ba_bo'] = row_data['avland_ba']/consolidated_means_dict['boroug
h_means']['avland_ba'][ibo]
    row_vars['vt_ba_bo'] = row_data['avtot_ba']/consolidated_means_dict['borough
_means']['avtot_ba'][ibo]
    row_vars['fv_ba_none'] = row_data['fullval_ba']
    row_vars['vl_ba_none'] = row_data['avland_ba']
    row_vars['vt_ba_none'] = row_data['avtot_ba']

    row_vars['fv_bv_z3'] = row_data['fullval_bv']/consolidated_means_dict['zip3_
means']['fullval_bv'][izip3]
    row_vars['vl_bv_z3'] = row_data['avland_bv']/consolidated_means_dict['zip3_m
eans']['avland_bv'][izip3]
    row_vars['vt_bv_z3'] = row_data['avtot_bv']/consolidated_means_dict['zip3_me
ans']['avtot_bv'][izip3]
    row_vars['fv_bv_z5'] = row_data['fullval_bv']/consolidated_means_dict['zip5_
means']['fullval_bv'][izip5]
    row_vars['vl_bv_z5'] = row_data['avland_bv']/consolidated_means_dict['zip5_m
eans']['avland_bv'][izip5]
    row_vars['vt_bv_z5'] = row_data['avtot_bv']/consolidated_means_dict['zip5_me
ans']['avtot_bv'][izip5]
    row_vars['fv_bv_tc'] = row_data['fullval_bv']/consolidated_means_dict['taxcl
ass_means']['fullval_bv'][itc]
    row_vars['vl_bv_tc'] = row_data['avland_bv']/consolidated_means_dict['taxcla
ss_means']['avland_bv'][itc]
    row_vars['vt_bv_tc'] = row_data['avtot_bv']/consolidated_means_dict['taxclas
s_means']['avtot_bv'][itc]
    row_vars['fv_bv_bo'] = row_data['fullval_bv']/consolidated_means_dict['borou
gh_means']['fullval_bv'][ibo]
    row_vars['vl_bv_bo'] = row_data['avland_bv']/consolidated_means_dict['boroug
h_means']['avland_bv'][ibo]
    row_vars['vt_bv_bo'] = row_data['avtot_bv']/consolidated_means_dict['borough
_means']['avtot_bv'][ibo]

    row_vars['fv_bv_none'] = row_data['fullval_bv']
    row_vars['vl_bv_none'] = row_data['avland_bv']
    row_vars['vt_bv_none'] = row_data['avtot_bv']

    row_vars['fv_none_z3'] = row_data['FULLVAL']/consolidated_means_dict['zip3_means']['FULLVAL'] [izip3]
    row_vars['vl_none_z3'] = row_data['AVLAND']/consolidated_means_dict['zip3_means']['AVLAND'][izip3]
    row_vars['vt_none_z3'] = row_data['AVTOT']/consolidated_means_dict['zip3_means']['AVTOT'][izip3]
    row_vars['fv_none_z5'] = row_data['FULLVAL']/consolidated_means_dict['zip5_means']['FULLVAL'][izip5]
    row_vars['vl_none_z5'] = row_data['AVLAND']/consolidated_means_dict['zip5_means']['AVLAND'][izip5]
    row_vars['vt_none_z5'] = row_data['AVTOT']/consolidated_means_dict['zip5_means']['AVTOT'][izip5]
    row_vars['fv_none_tc'] = row_data['FULLVAL']/consolidated_means_dict['taxclass_means']['FULLVAL'][itc]
    row_vars['vl_none_tc'] = row_data['AVLAND']/consolidated_means_dict['taxclass_means']['AVLAND'][itc]
    row_vars['vt_none_tc'] = row_data['AVTOT']/consolidated_means_dict['taxclass_means']['AVTOT'][itc]
    row_vars['fv_none_bo'] = row_data['FULLVAL']/consolidated_means_dict['borough_means']['FULLVAL'][ibo]
    row_vars['vl_none_bo'] = row_data['AVLAND']/consolidated_means_dict['borough_means']['AVLAND'][ibo]
    row_vars['vt_none_bo'] = row_data['AVTOT']/consolidated_means_dict['borough_means']['AVTOT'][ibo]
    row_vars['fv_none_none'] = row_data['FULLVAL']
    row_vars['vl_none_none'] = row_data['AVLAND']
    row_vars['vt_none_none'] = row_data['AVTOT']

    return row_vars

myvars = mydata.apply(calc_vars, axis=1)
```

```
CPU times: user 11h 40min 10s, sys: 11min 15s, total: 11h 51min 25s
Wall time: 12h 5min 59s
```

In [43]:

```
myvars.shape
```

Out[43]:

```
(1048575, 60)
```

In [44]:

```
%%time
myvars_zscale = (myvars - myvars.mean()) / myvars.std()
```

```
CPU times: user 3.35 s, sys: 2.31 s, total: 5.67 s
Wall time: 4.75 s
```

In [45]:

```
fa_dir = '/Users/stevecoggeshall/Documents/Teaching/Fraud Analytics'
myvars_zscale.to_csv(fa_dir + '/2018 USC fraud class/data/NY property/NY propert
y vars 1 million zscale.csv')
```

In [ ]:

```

```