Diversion Demographics

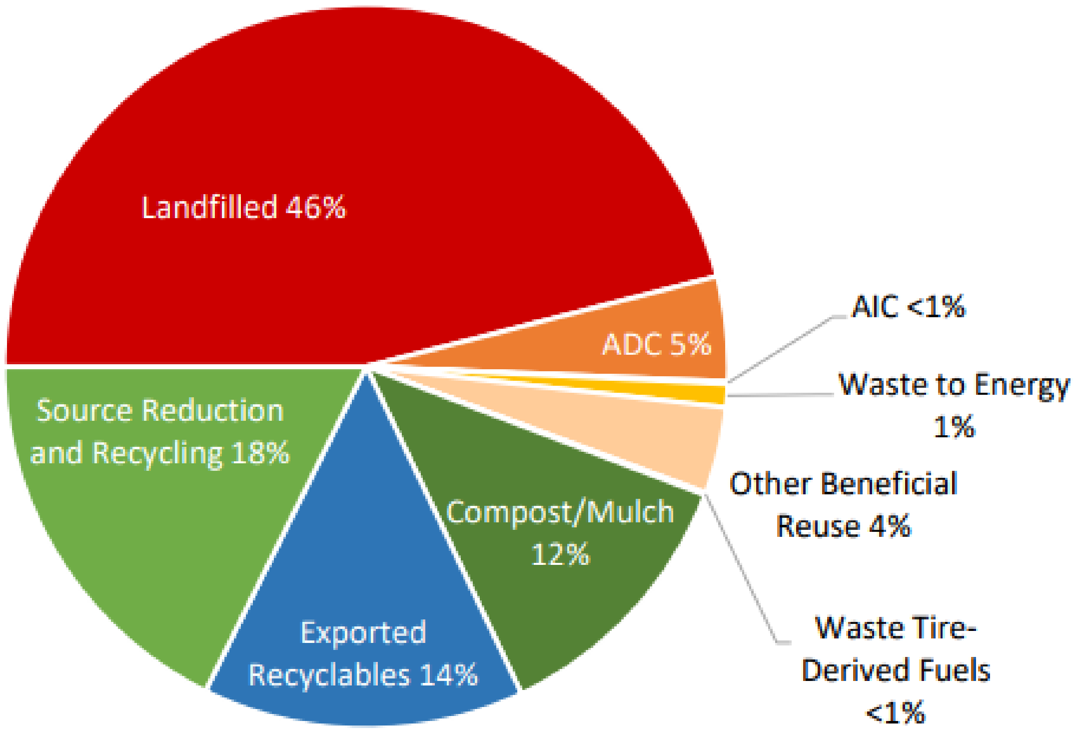
A quick glimpse of how neighborhood characteristics explain behaviors towards recycling and composting

The Problem

La Puenta Hills

The Problem

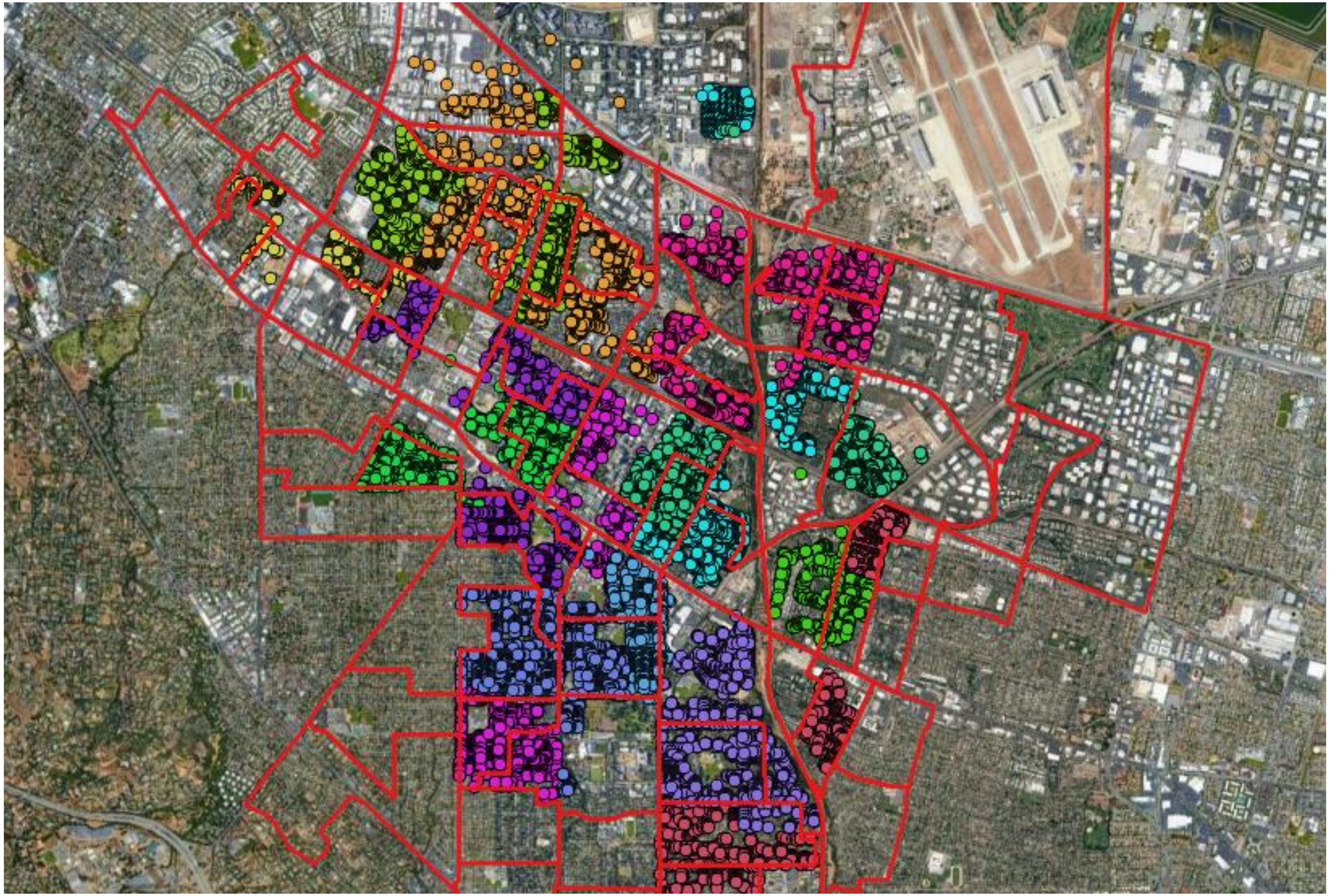
- We Are In A Garbage Crisis
- 75% resource reduction by 2020



The Solution?

SF: 80% diversion rate after introducing compost bin and price incentives for recycling

Mountain View



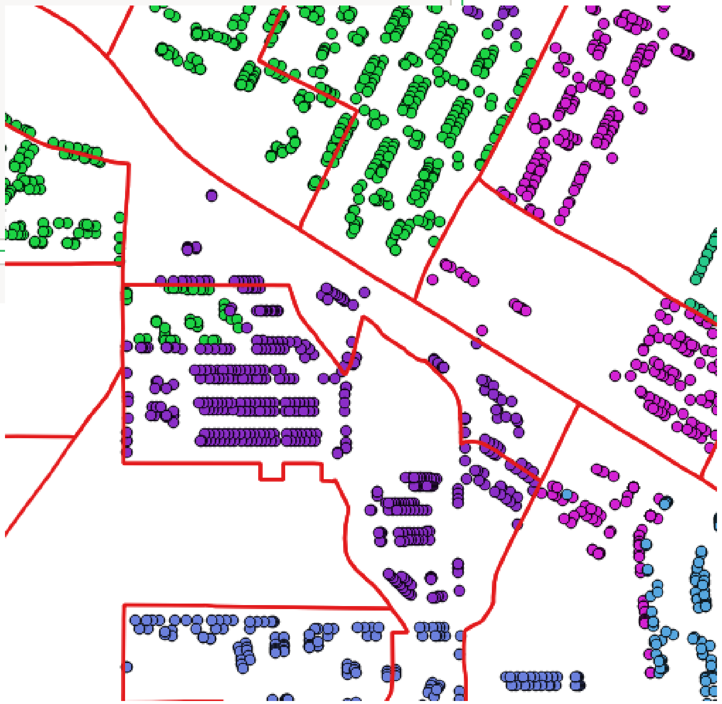
Calculated Diversion Rate: 51%

Methodology

- Collect census data and join it to census block groups
- Overlay census block groups with customer stops and route territories
- Apply recorded weights from routes to block groups based on percentage of stops and route within that block group
- Calculate Diversion Rate (p_NLF = (GRO-G)/GRO)
- Determine if census data is explaining variance in diversion rates amongst block groups

y: Fitting routes to blockgroups

```
[9]: # Group by BG ID and route to get the count of rount numbers in a block group

bgRoute = pd.read_csv(r"C:\Data\Waste_Intervention\Mountain_View\AllComs.csv", low_memory=False)
#create series that gets number of route stops in a BG, include route to do a join later as we will be using the route count
bg = bgRoute.groupby(['GEOID','Z1COMM','Route']).size().reset_index(name='count')

# mkae a dataframe getting the count of stops per commodity in each BG
bgCom = bgRoute.groupby(['GEOID','Z1COMM']).size().reset_index(name='count')

# create a dateframe that gets a count of the route that can be joined back to the BG table
rt = bgRoute.groupby(['Route', 'Z1COMM']).size().reset_index(name='count')

bgRt = pd.merge(bg, rt, on = 'Route')
bgRtCom = pd.merge(bgRt, bgCom, left_on=(['GEOID','Z1COMM_x']), right_on=(['GEOID', 'Z1COMM']))

[10]: bgRtCom.head()
```



y: Use % of route stops to assign weight contribution to a given blockgroup

Out[10]:

	GEOID	Z1COMM_x	Route	count_x	Z1COMM_y	count_y	Z1COMM	count
0	60855046011	G	303X	4	G	622	G	367
1	60855046011	G	404X	363	G	817	G	367
2	60855092023	G	303X	148	G	622	G	152
3	60855092023	G	305X	4	G	725	G	152
4	60855093031	G	303X	9	G	622	G	162

In [11]:
```
bgRtCom['perRt'] = bgRtCom['count_x'] / bgRtCom['count_y']
```

In [12]:
```
bgRtCom.head()
```

Out[12]:

	GEOID	Z1COMM_x	Route	count_x	Z1COMM_y	count_y	Z1COMM	count	perRt
0	60855046011	G	303X	4	G	622	G	367	0.006431
1	60855046011	G	404X	363	G	817	G	367	0.444308
2	60855092023	G	303X	148	G	622	G	152	0.237942
3	60855092023	G	305X	4	G	725	G	152	0.005517
4	60855093031	G	303X	9	G	622	G	162	0.014469

y: Final

	GEOID	Z1COMM_x_x	bgTons_x	Z1COMM_x_y	bgTons_y	Z1COMM_x	bgTons	totalWaste	nonLF	per_NLF
0	60855046011	G	3.417532	O	2.715642	R	1.419062	7.552236	4.134705	0.547481
1	60855091051	G	6.038411	O	3.142073	R	3.062613	12.243097	6.204686	0.506791
2	60855091052	G	2.820326	O	2.109063	R	1.794390	6.723779	3.903453	0.580545
3	60855091053	G	3.001942	O	1.826215	R	1.100159	5.928316	2.926374	0.493627
4	60855091081	G	1.776370	O	1.170674	R	0.488146	3.435189	1.658820	0.482890

```
x = pd.read_csv(r"C:\Data\Waste Intervention\Mountain View\census merged 3.csv")
```

Who is below .508?

X: Data Dictionary

Id2	Blockgroup ID
med_hh_inc	median husehold income
med_age	median age
p_vacant	percent vacant
avg_hh_size	average household size
p_renters	percent renters
p_r1000_plus	percent of blockgroup paying $1000 or more in monthly rent
p_ccasian	percent of caucasians in population
p_phh_ba	percent of population with a bacelor's degree
p_phh_stem	percent of population with a STEM related degree
p_phh_biz	percent of population with business major degrees
p_phh_ed	percent of population with education major degrees
p_phh_hum	percent of population with humanities major degrees
p_nonfam_hh	percent of non-family households
p_apt	percent of apartments in housing stock

X: features

```
print df.head(2)
```

```
           Id2  med_age
0  60855001001     27.2
1  60855001002     33.7
           Id2  White  AfAm  NatAm  Asian  Hawaiin  Other  Inter
0  60855001001    310     0      0    112        0    500     10
1  60855001002    397    18     12    727        0    717     65
           Id2  Total_HH  FamHH  NonFM  NoFamAloneHH
0  60855001001       286    174    112            86
1  60855001002       664    436    228           142
           Id2  HH1  HH2  HH3  HH4  HH5  HH6  HH7
0  60855001001   86   67   69   16    0    0   48
1  60855001002  142  135  157  211   19    0    0
           Id2  MedHHInc
0  60855001001   59118.0
1  60855001002  110714.0
           Id2  Vacant
0  60855001001       0
1  60855001002       0
           Id2  TotalHH  OwnerHH  RenterHH
0  60855001001      888      290       598
1  60855001002     1883      607      1276
           Id2  AvgHHSize  AvOwnHHsize  AvRenHHsize
0  60855001001       3.10         3.02         3.15
1  60855001002       2.84         2.21         3.28
```

X: Final

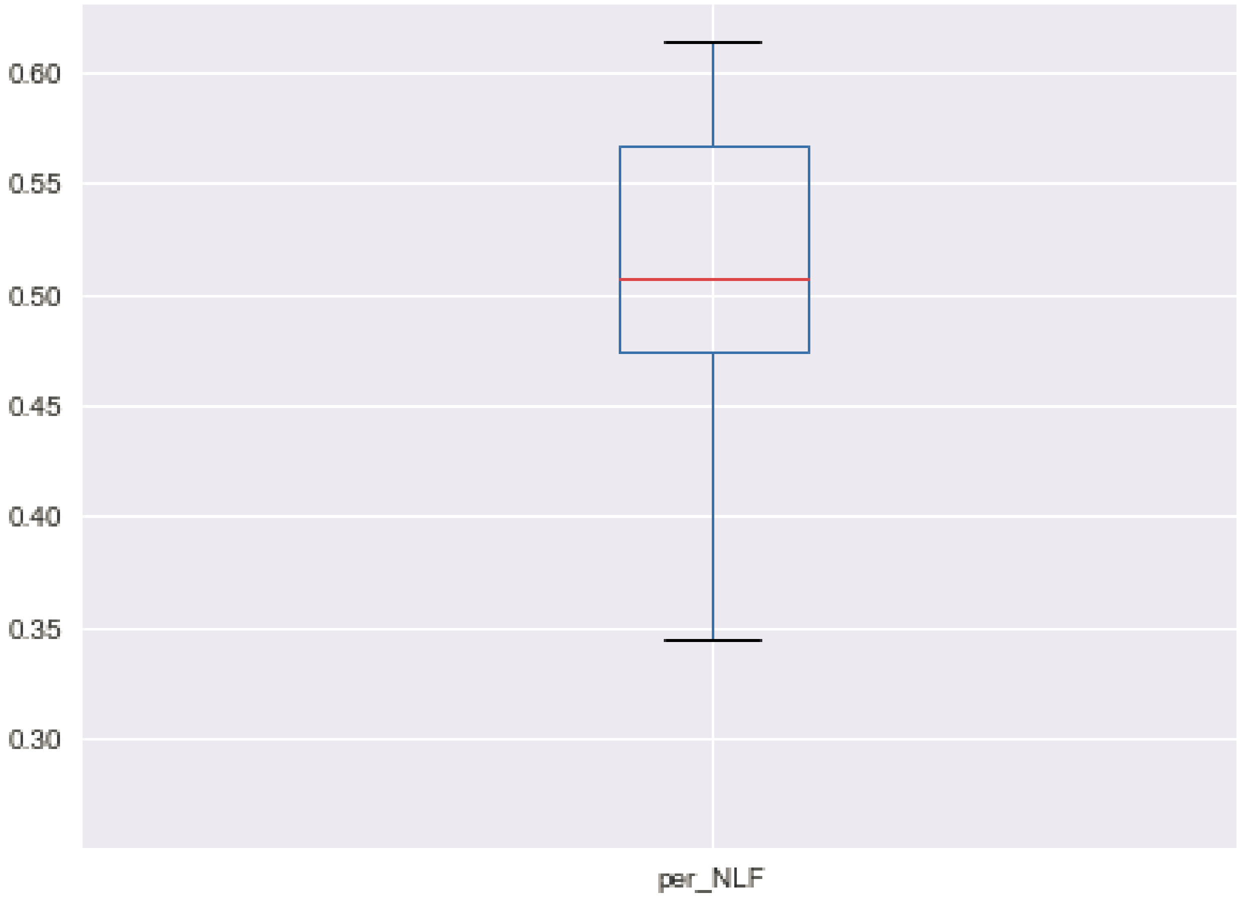
```
x.head()
```

_hh_inc	med_age	p_vacant	avg_hh_size	p_renters	p_r1000_plus	p_ccasian	p_phh_ba	p_phh_stem	p_phh_biz	p_phh_ed	p_phh_hum	p_nonfam_hh	p_
59118.0	27.2	0.000000	3.10	0.673423	0.611888	0.332618	0.671329	0.307692	0.132867	0.000000	0.230769	0.391608	0.293
110714.0	33.7	0.000000	2.84	0.677642	0.468373	0.205062	0.862952	0.554217	0.147590	0.021084	0.140060	0.343373	0.218
62730.0	26.9	0.000000	4.67	0.618702	0.577540	0.279008	0.641711	0.370766	0.149733	0.030303	0.090909	0.351159	0.229
85648.0	34.1	0.100679	2.64	0.674658	0.533937	0.323202	0.803167	0.437783	0.138009	0.011312	0.216063	0.420814	0.489
103265.0	30.6	0.028517	2.71	0.570927	0.509506	0.507653	0.994297	0.610266	0.057034	0.022814	0.304183	0.513308	0.281

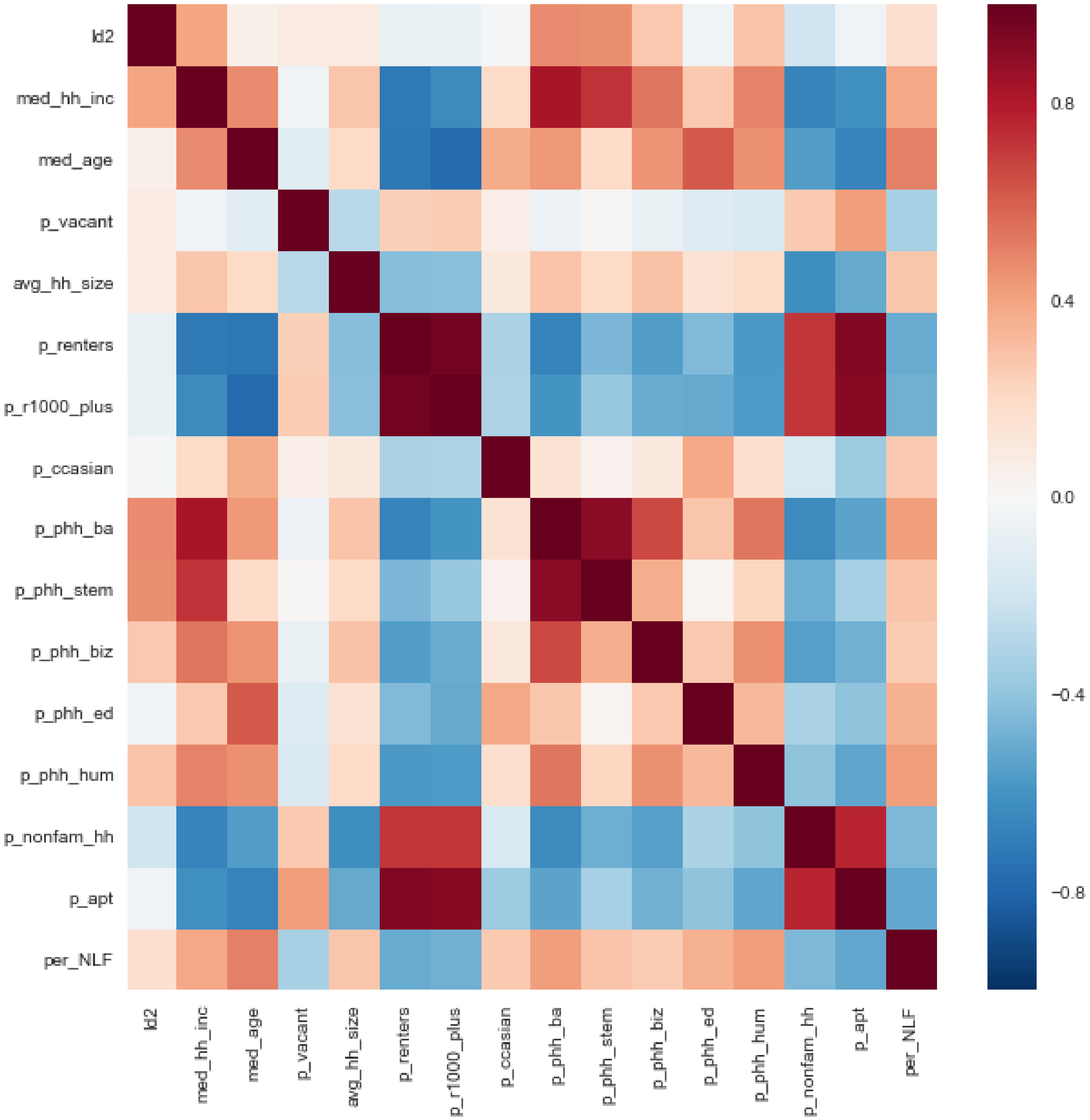
count 56.000000 mean 0.509190 std 0.070802 min 0.268017 25% 0.474059 50% 0.506763 75% 0.566571 max 0.613565

Exploratory Analysis

count 56.000000
mean 0.509190
std 0.070802
min 0.268017
25% 0.474059
50% 0.506763
75% 0.566571
max 0.613565



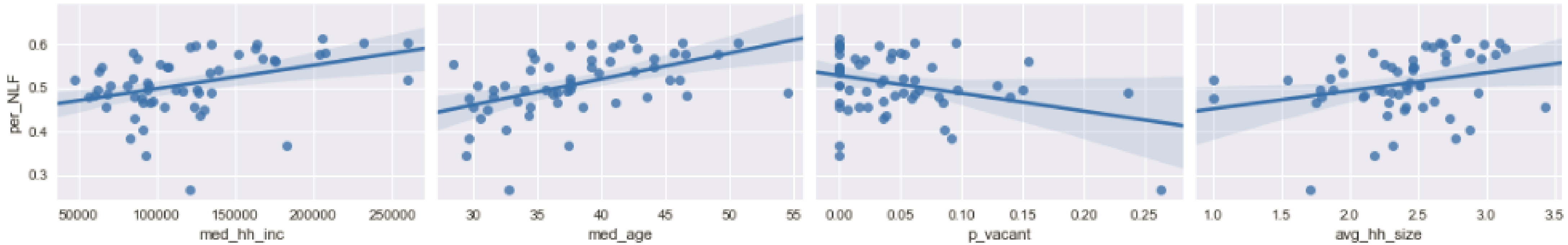
Exploratory Analysis: Heat Map



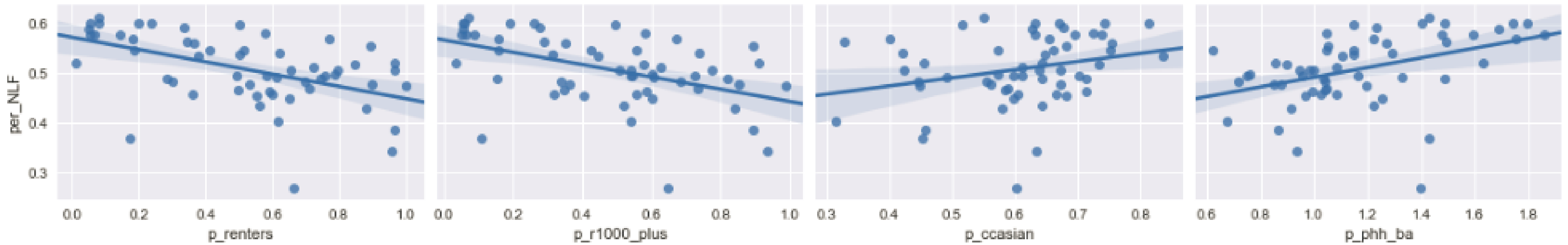
Regression Analysis

```
R-Squared: 0.477963521181
Intercept: 0.185102394096
('med_hh_inc', -1.9147997945004355e-07)
('med_age', 0.0046061213611979356)
('p_vacant', -0.26772753022466556)
('avg_hh_size', 0.014898735889548643)
('p_renters', -0.098761377285677063)
('p_r1000_plus', 0.15208720266927717)
('p_ccasian', 0.063468874373172807)
('p_phh_ba', 18925908.334525019)
('p_phh_stem', -18925908.26390072)
('p_phh_biz', -18925908.443293739)
('p_phh_ed', -18925908.162769377)
('p_phh_hum', -18925908.04051315)
('p_nonfam_hh', -0.016407628543674946)
('p_apt', -0.01725218445062637З)
```

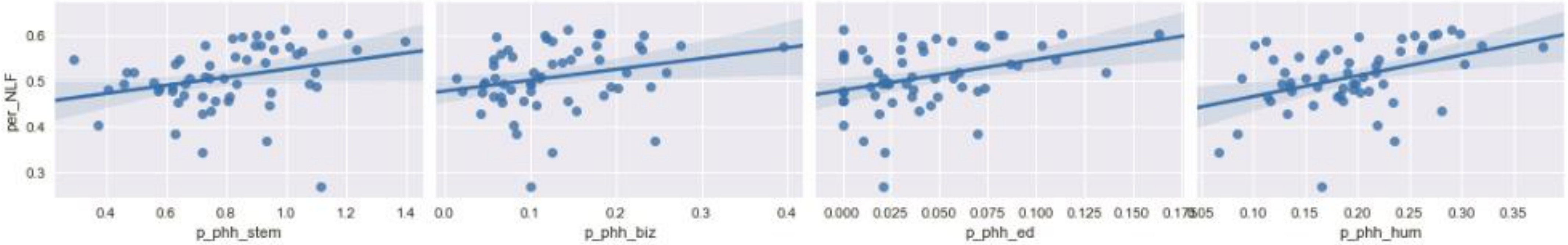
More exploratory analysis



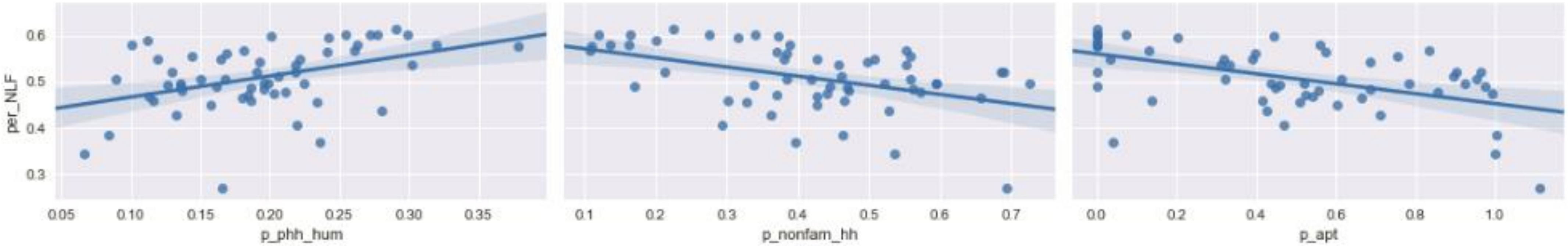
In [104]:
```
g = sns.pairplot(df, x_vars=['p_renters', 'p_r1000_plus', 'p_ccasian', 'p_phh_ba'], y_vars='per_NLF',  size=15, aspect=0.7, kind=

g.fig.set_size_inches(15,2)
```



More exploratory analysis




```python
g = sns.pairplot(df, x_vars=[ 'p_phh_hum', 'p_nonfam_hh', 'p_apt'], y_vars='per_NLF',  size=15, aspect=0.7, kind='reg')

g.fig.set_size_inches(15,2)
```



Random Forest Regressor

```
In [144]: from sklearn.ensemble import RandomForestRegressor

          cls = RandomForestRegressor(n_estimators=50)
          cls.fit(X,y)

          print cls.score(X,y)
          print cross_val_score(cls, X, y, scoring='r2')

          0.923654403012
          [-0.20153946 -0.22140289 -2.66111633]
```

Conclusion – Limitations – Future considerations

- Features explain somewhere between 47% to 92% of the variance in diversion rates across residential blockgroups in Mountain View CA
- Cross validation scoring did not yield desirable results. Will need to do diagnostics on data validity to improve model performance
- More data points (blockgroups) could help improve model and allow for more robust analyses
- Would be interesting to start building a predictive model to identify waste bin contaminators

Questions?