In [215]:
```
%load_ext sql
```

```
The sql extension is already loaded. To reload it, use:
  %reload_ext sql
```

In [216]:
```
%sql mysql://prod:nerd@52.2.153.189/rental_nerd
```

Out[216]:
```
u'Connected: prod@rental_nerd'
```

In [217]:
```
result = %sql (SELECT \
properties.id as "property_id", \
property_transaction_logs.id as "transaction_log_id", \
properties.*, \
property_transaction_logs.* \
FROM \
properties, \
property_transactions, \
property_transaction_logs \
WHERE \
properties.id = property_transactions.property_id AND \
property_transactions.property_transaction_log_id = property_transaction_logs.id AND \
property_transactions.transaction_type = 'rental')

data = result.DataFrame()
```

```
560 rows affected.
```

In [218]:
```
result.csv(filename="SQLdump.csv")
```

Out[218]: CSV results (./files/SQLdump.csv)

In [219]:
```
# imports
import pandas as pd
import matplotlib.pyplot as plt
# follow the usual sklearn pattern: import, instantiate, fit
from sklearn.linear_model import LinearRegression
import numpy as np

# this allows plots to appear directly in the notebook
%matplotlib inline

# read data into a DataFrame
data.head()
```

Out[219]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | id | price | tra |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 1 | 1 | 567 Vallejo Street #PH500 | San Francisco (North Beach) | 3 | 3 | 2081 | climbsf_renting | http://www.climbsf.com/for-rent/567-vallejo-st... | ... | 1 | 12000 | op |
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 2 | 3950 | op |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 3 | 5400 | op |
| **3** | 4 | 4 | 4 | 333 Fremont Street #705 | San Francisco (South Beach) | 1 | 1 | 0 | climbsf_renting | http://www.climbsf.com/for-rent/333-fremont-st... | ... | 4 | 3600 | op |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 5 | 3975 | op |

5 rows × 26 columns

In [220]:
```
import datetime

Date_final = [0.1] * len(data)

for x in range(0,len(data)):
    data
    if data["date_closed"][x] is not None :
        # print " row: "+ `x` + ": using date_rented"
        # data.ix['Date_final',x]
        Date_final[x] = data["date_closed"][x]

    elif data["date_listed"][x] is not None :
        # print " row: "+ `x` + ": using date_listed"
        Date_final[x] = data["date_listed"][x]
    else:
        Date_final[x] = data["date_closed"][2]
        print " row: "+ `x` + ": we are screwed"



data['Date'] = pd.to_datetime(Date_final)

data.head()
```

Out[220]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | price | transa |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 1 | 1 | 567 Vallejo Street #PH500 | San Francisco (North Beach) | 3 | 3 | 2081 | climbsf_renting | http://www.climbsf.com/for-rent/567-vallejo-st... | ... | 12000 | open |
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 3950 | open |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 5400 | open |
| **3** | 4 | 4 | 4 | 333 Fremont Street #705 | San Francisco (South Beach) | 1 | 1 | 0 | climbsf_renting | http://www.climbsf.com/for-rent/333-fremont-st... | ... | 3600 | open |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 3975 | open |

5 rows × 27 columns

In [221]:
```
# create neighborhoods from lat/long coordinates
import fiona
import shapely as shapely
from shapely.geometry import asShape
```

In [227]:
```
shaped_neighborhood = ['None'] * len(data)

with fiona.open('data/Planning_Neighborhoods_4326/planning_hoods_4326.shp') as fiona_collection:
    for hood in fiona_collection:
        print "checking for listings in: " + hood["properties"]["neighborho"]
        # Use Shapely to create the polygon
        shape = asShape( hood['geometry'] )

        for row in range(0,len(data)):
            point = shapely.geometry.Point([data['longitude'][row], data['latitude'][row]]) # longitude, latitude

            if shaped_neighborhood[row] != 'None':
                continue

            if shape.contains(point):
                #print `row` + ": Found " + data.address[row] + " in hood " + hood["properties"]["nbrhood"]
                shaped_neighborhood[row] = hood["properties"]["neighborho"]

data['shaped_neighborhood'] = shaped_neighborhood
data.head()
```

```
checking for listings in: Seacliff
checking for listings in: Haight Ashbury
checking for listings in: Outer Mission
checking for listings in: Russian Hill
checking for listings in: Noe Valley
checking for listings in: Inner Sunset
checking for listings in: Downtown/Civic Center
checking for listings in: Diamond Heights
checking for listings in: Treasure Island/YBI
checking for listings in: Lakeshore
checking for listings in: Outer Richmond
checking for listings in: Crocker Amazon
checking for listings in: Excelsior
checking for listings in: Parkside
checking for listings in: Financial District
checking for listings in: Ocean View
checking for listings in: Mission
checking for listings in: West of Twin Peaks
checking for listings in: Inner Richmond
checking for listings in: Marina
checking for listings in: Bayview
checking for listings in: Visitacion Valley
checking for listings in: Pacific Heights
checking for listings in: Presidio
checking for listings in: Nob Hill
checking for listings in: Outer Sunset
checking for listings in: Western Addition
checking for listings in: Golden Gate Park
checking for listings in: Presidio Heights
checking for listings in: South of Market
checking for listings in: Glen Park
checking for listings in: Potrero Hill
checking for listings in: Castro/Upper Market
checking for listings in: Twin Peaks
checking for listings in: Bernal Heights
checking for listings in: Chinatown
checking for listings in: North Beach
```

Out[227]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | transaction_s |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 1 | 1 | 567 Vallejo Street #PH500 | San Francisco (North Beach) | 3 | 3 | 2081 | climbsf_renting | http://www.climbsf.com/for-rent/567-vallejo-st... | ... | open |
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | open |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | open |
| **3** | 4 | 4 | 4 | 333 Fremont Street #705 | San Francisco (South Beach) | 1 | 1 | 0 | climbsf_renting | http://www.climbsf.com/for-rent/333-fremont-st... | ... | open |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | open |

5 rows × 28 columns

In [228]:
```
# filter out any outliers, defined as rent >$10k or >2,500 sq ft, or not in SF

print "Entries before filter: " + `len(data)`
data = data[(data.shaped_neighborhood != 'None') & (data.sqft <= 2500) & (data.price <= 8000) & (data.price != 0) & (data.
bedrooms <= 4) & (data.bathrooms <= 3) & (data.sqft != 0)]

# filter out listings over one month old


print "Entries after filter: " + `len(data)`
```

```
Entries before filter: 560
Entries after filter: 304
```

In [229]:
```
# create year dummy variables (because date isn't very intuitive variable)
data["Year"] = pd.DatetimeIndex(data["Date"]).to_period('Y')

# create dummy variables using get_dummies, then exclude the first dummy column
year_dummies = pd.get_dummies(data.Year, prefix='Year').iloc[:, :-1]

# print out baseline neighborhood
base_area = pd.get_dummies(data.shaped_neighborhood, prefix='neighborhood').iloc[:, 0:1].columns[0]
print('Base neighborhood: %s' % base_area)

# create dummy variables using get_dummies, then exclude the first dummy column
area_dummies = pd.get_dummies(data.shaped_neighborhood, prefix='neighborhood').iloc[:, 1:]

# concatenate the dummy variable columns onto the original DataFrame (axis=0 means rows, axis=1 means columns)
data = pd.concat([data, area_dummies, year_dummies], axis=1)

data.head()
```

```
Base neighborhood: neighborhood_Bayview
```

Out[229]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | neighborho Hill |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 0 |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 0 |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 0 |
| **7** | 8 | 8 | 8 | 1160 Mission Street #1112 | San Francisco (SOMA) | 1 | 1 | 664 | climbsf_renting | http://www.climbsf.com/for-rent/1160-mission-s... | ... | 0 |
| **11** | 12 | 12 | 12 | 655 26th Avenue | San Francisco (Central Richmond) | 2 | 1 | 1300 | climbsf_renting | http://www.climbsf.com/for-rent/655-26th-ave/ | ... | 0 |

5 rows × 62 columns

In [230]:
```
# FACTORING BY YEAR AND NEIGHBORHOOD
# Thesis: Neighborhoods influence valuations as a multiplier, rather than a constant.
# a square foot in SOMA is worth more than a square foot in Portrero by X%
# New model will look like this:
#      Price = B_1 x (SOMA Coeff * Year Coeff * Sqft) + intercept
#      $3,900 = B_1 x (1.20% * 1.15% * 2,023 sqft) + intercept
# where B_1 represents the price per square foot in base year and base neighborhood
# I will ignore intercepts for now FIXME
# calculate the coefficients for the following matrix and save them for later regressions
#                  SOMA    Mission    Portrero    Intercept
#  Price/SQFT      $1.23    $0.59       $0.88        $_.__

# create Price per square foot

price_per_foot = data.price / data.sqft
price_per_foot.name = 'price_per_foot'
data = pd.concat([data, price_per_foot], axis=1)

data.head()
```

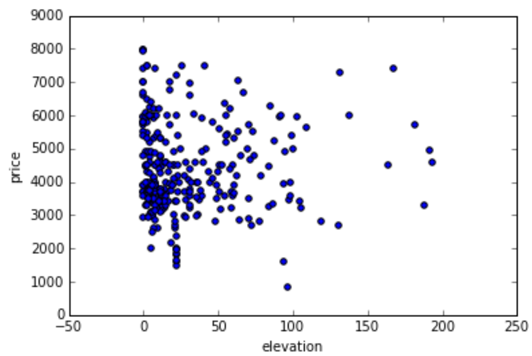
Out[230]:

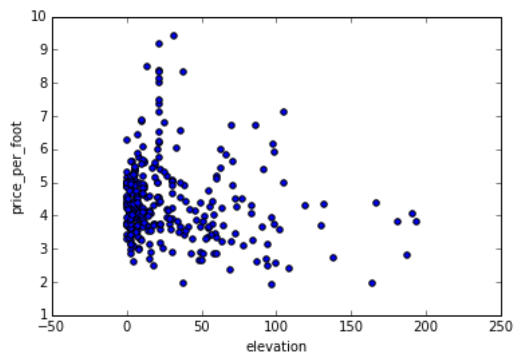
| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | neighborho of Market |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 0 |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 0 |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 1 |
| **7** | 8 | 8 | 8 | 1160 Mission Street #1112 | San Francisco (SOMA) | 1 | 1 | 664 | climbsf_renting | http://www.climbsf.com/for-rent/1160-mission-s... | ... | 1 |
| **11** | 12 | 12 | 12 | 655 26th Avenue | San Francisco (Central Richmond) | 2 | 1 | 1300 | climbsf_renting | http://www.climbsf.com/for-rent/655-26th-ave/ | ... | 0 |

5 rows × 63 columns

In [231]:
```
# visualize the relationship between the features and the response using scatterplots
data.plot(kind='scatter', x='elevation', y='price')
data.plot(kind='scatter', x='elevation', y='price_per_foot')
```

Out[231]: <matplotlib.axes._subplots.AxesSubplot at 0x110e87890>




In [232]:
```
class ListTable(list):
    """ Overridden list class which takes a 2-dimensional list of
        the form [[1,2,3],[4,5,6]], and renders an HTML Table in
        IPython Notebook. """

    def _repr_html_(self):
        html = ["<table>"]
        for row in self:
            html.append("<tr>")

            for col in row:
                html.append("<td>{0}</td>".format(col))

            html.append("</tr>")
        html.append("</table>")
        return ''.join(html)
```

In [233]:
```
feature_cols = area_dummies.columns

X = data[feature_cols]
y = data.price_per_foot

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))

# print raw results
print("Base area is %s: $%.2f" % (base_area, lm.intercept_))

zip(feature_cols,lm.coef_)

table = ListTable()

dtype = [('Neighborhood', 'S100'), ('$ per square', float)]

# round to pennies
round_coef = map(round,lm.coef_,[2]*len(lm.coef_))
x = np.array(zip(feature_cols, round_coef),dtype=dtype)
x.T
x = np.sort(x,axis=0,order='$ per square')

table.append(['Neighborhood','$ per square (+/-)'])
for i in x:
    table.append(i)

table
```

```
Residual sum of squares: 1.04
Variance score: 0.29
Base area is neighborhood_Bayview: $3.15
```

Out[233]:

| Neighborhood | $ per square (+/-) |
|---|---|
| neighborhood_Visitacion Valley | -0.58 |
| neighborhood_Glen Park | -0.42 |
| neighborhood_Bernal Heights | -0.36 |
| neighborhood_Diamond Heights | -0.34 |
| neighborhood_Lakeshore | -0.28 |
| neighborhood_Excelsior | -0.21 |
| neighborhood_Ocean View | 0.06 |
| neighborhood_Outer Richmond | 0.13 |
| neighborhood_Haight Ashbury | 0.14 |
| neighborhood_Parkside | 0.37 |
| neighborhood_West of Twin Peaks | 0.38 |
| neighborhood_Outer Sunset | 0.4 |
| neighborhood_Potrero Hill | 1.02 |
| neighborhood_Noe Valley | 1.07 |
| neighborhood_Marina | 1.12 |
| neighborhood_South of Market | 1.13 |
| neighborhood_Inner Richmond | 1.24 |
| neighborhood_Downtown/Civic Center | 1.26 |
| neighborhood_North Beach | 1.27 |
| neighborhood_Nob Hill | 1.32 |
| neighborhood_Pacific Heights | 1.52 |
| neighborhood_Mission | 1.62 |
| neighborhood_Financial District | 1.65 |
| neighborhood_Castro/Upper Market | 1.68 |
| neighborhood_Inner Sunset | 1.94 |
| neighborhood_Russian Hill | 2.48 |
| neighborhood_Western Addition | 2.53 |
| neighborhood_Chinatown | 5.19 |

In [234]:
```
full_price = [lm.intercept_] * len(lm.coef_)
full_price += lm.coef_

area_price_per_foot = dict(zip(feature_cols,full_price))
area_price_per_foot[base_area] = lm.intercept_

dtype = [('Neighborhood', 'S100'), ('$ per sqft', float)]

# round to pennies
round_coef = map(round,full_price,[2]*len(full_price))
x = np.array(zip(feature_cols, full_price),dtype=dtype)
x.T
x = np.sort(x,axis=0,order='$ per sqft')

table = ListTable()

table.append(['Neighborhood','$ per sqft'])
for i in x:
    table.append(i)

table
```

Out[234]:

| Neighborhood | $ per sqft |
|---|---|
| neighborhood_Visitacion Valley | 2.56314257913 |
| neighborhood_Glen Park | 2.72727272727 |
| neighborhood_Bernal Heights | 2.78200061463 |
| neighborhood_Diamond Heights | 2.8085106383 |
| neighborhood_Lakeshore | 2.86666666667 |
| neighborhood_Excelsior | 2.93253968254 |
| neighborhood_Ocean View | 3.21056547619 |
| neighborhood_Outer Richmond | 3.27613474488 |
| neighborhood_Haight Ashbury | 3.28587304169 |
| neighborhood_Parkside | 3.51137487636 |
| neighborhood_West of Twin Peaks | 3.53093930792 |
| neighborhood_Outer Sunset | 3.54200988468 |
| neighborhood_Potrero Hill | 4.16518162307 |
| neighborhood_Noe Valley | 4.2183338668 |
| neighborhood_Marina | 4.27103404056 |
| neighborhood_South of Market | 4.27811516862 |
| neighborhood_Inner Richmond | 4.38912151969 |
| neighborhood_Downtown/Civic Center | 4.40392841058 |
| neighborhood_North Beach | 4.42100898552 |
| neighborhood_Nob Hill | 4.46503589805 |
| neighborhood_Pacific Heights | 4.66270219935 |
| neighborhood_Mission | 4.76198374942 |
| neighborhood_Financial District | 4.79539277753 |
| neighborhood_Castro/Upper Market | 4.8298805149 |
| neighborhood_Inner Sunset | 5.08241758242 |
| neighborhood_Russian Hill | 5.62111377504 |
| neighborhood_Western Addition | 5.67123941357 |
| neighborhood_Chinatown | 8.33333333333 |

In [235]:
```
# calculate the multipliers for each neighborhood relative to base area
# SOMA_mult = SOMA_per_foot / Base_per_foot

area_mults = [lm.intercept_] * len(lm.coef_)
area_mults = full_price / area_mults - [1]*len(lm.coef_)


dtype = [('Neighborhood', 'S100'), ('Multiplier', float)]

# round to pennies
round_coef = map(round,area_mults,[2]*len(area_mults))
x = np.array(zip(feature_cols, area_mults),dtype=dtype)
x.T
x = np.sort(x,axis=0,order='Multiplier')

table = ListTable()

table.append(['Neighborhood','Multiplier'])
table.append([base_area,0])
for i in x:
    table.append(i)

table
```

Out[235]:

| Neighborhood | Multiplier |
|---|---|
| neighborhood_Bayview | 0 |
| neighborhood_Visitacion Valley | -0.185286214946 |
| neighborhood_Glen Park | -0.133116236059 |
| neighborhood_Bernal Heights | -0.115720573165 |
| neighborhood_Diamond Heights | -0.107294166495 |
| neighborhood_Lakeshore | -0.08880884368 |
| neighborhood_Excelsior | -0.0678706194346 |
| neighborhood_Ocean View | 0.0205019309387 |
| neighborhood_Outer Richmond | 0.0413436069017 |
| neighborhood_Haight Ashbury | 0.04443899641 |
| neighborhood_Parkside | 0.11611641879 |
| neighborhood_West of Twin Peaks | 0.122335117748 |
| neighborhood_Outer Sunset | 0.125853982273 |
| neighborhood_Potrero Hill | 0.323933718397 |
| neighborhood_Noe Valley | 0.340828551337 |
| neighborhood_Marina | 0.357579690501 |
| neighborhood_South of Market | 0.359830479313 |
| neighborhood_Inner Richmond | 0.395114667242 |
| neighborhood_Downtown/Civic Center | 0.399821146787 |
| neighborhood_North Beach | 0.405250333588 |
| neighborhood_Nob Hill | 0.419244612659 |
| neighborhood_Pacific Heights | 0.482074305325 |
| neighborhood_Mission | 0.513631678725 |
| neighborhood_Financial District | 0.524250984873 |
| neighborhood_Castro/Upper Market | 0.535213166719 |
| neighborhood_Inner Sunset | 0.61548393738 |
| neighborhood_Russian Hill | 0.786712497844 |
| neighborhood_Western Addition | 0.802645301983 |
| neighborhood_Chinatown | 1.64881150093 |

In [236]:
```
# calculate the adjusted Sqft (Sqft * Area_mult) for the dataset and add it as a new column to data

# for each property, multiplier is sum of array [area_dummies] x [area_mults]

t = data[area_dummies.columns] * area_mults
t = t.T.sum()

t.name = 'area_multiplier'
t = t + 1
data = pd.concat([data, t], axis=1)

adj_sqft = data.sqft * t
adj_sqft.name = 'area_adj_sqft'
data = pd.concat([data, adj_sqft], axis=1)

data.head()
```

Out[236]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | neighborho of Twin Pea |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 0 |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 0 |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 0 |
| **7** | 8 | 8 | 8 | 1160 Mission Street #1112 | San Francisco (SOMA) | 1 | 1 | 664 | climbsf_renting | http://www.climbsf.com/for-rent/1160-mission-s... | ... | 0 |
| **11** | 12 | 12 | 12 | 655 26th Avenue | San Francisco (Central Richmond) | 2 | 1 | 1300 | climbsf_renting | http://www.climbsf.com/for-rent/655-26th-ave/ | ... | 0 |

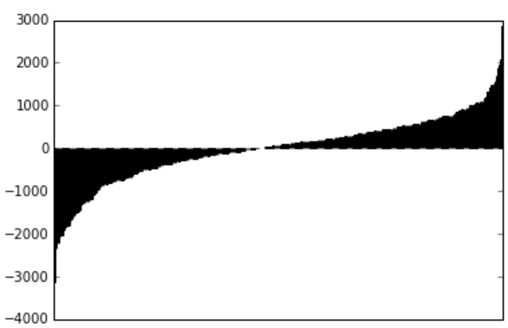
5 rows × 65 columns

In [237]:

```
# run the regression based on area_adj_sqft rather than sqft

# create X and y
feature_cols = [data.area_adj_sqft.name]

X = data[feature_cols]
y = data.price

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
print("Intercept: %.2f" % lm.intercept_)

# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))
zip(feature_cols, lm.coef_)

# calculate predictions for the data set and plot errors
predictions = lm.predict(X)
errors = predictions-y
errors.name = 'Error'

# visualize the relationship between the features and the response using scatterplots
errors.sort()
errors.plot(kind='bar').get_xaxis().set_ticks([])
```

```
Intercept: 1616.99
Residual sum of squares: 611848.95
Variance score: 0.66
```

Out[237]: []



In [238]:
```
feature_cols = year_dummies.columns

X = data[feature_cols]
y = data.price_per_foot

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))

# print raw results
print lm.intercept_

zip(feature_cols,lm.coef_)
```

```
Residual sum of squares: 1.32
Variance score: 0.11
4.79633313103
```

Out[238]:
```
[(u'Year_1969', -1.9410031817959141),
 (u'Year_2011', -0.62577229712902971),
 (u'Year_2012', -1.3700383103173595),
 (u'Year_2013', -0.98855679368271021),
 (u'Year_2014', -0.60179239513716498)]
```

In [239]:
```
full_price = [lm.intercept_] * len(lm.coef_)
full_price += lm.coef_

year_price_per_foot = dict(zip(feature_cols,full_price))
year_price_per_foot[base_area] = lm.intercept_

print year_price_per_foot
```

```
{u'Year_1969': 2.8553299492385782, u'Year_2012': 3.4262948207171329, u'Year_2013': 3.8077763373517821, u'neighborhood_Bayv
iew': 4.7963331310344923, u'Year_2011': 4.1705608339054629, u'Year_2014': 4.194540735897327}
```

In [240]:
```
# calculate the multipliers for each year relative to base year
# 2014_mult = 2014_per_foot / 2015_per_foot

year_mults = [lm.intercept_] * len(lm.coef_)
year_mults = full_price / year_mults - [1]*len(lm.coef_)

zip(feature_cols, year_mults)
```

Out[240]:
```
[(u'Year_1969', -0.40468481416287083),
 (u'Year_2011', -0.13046889780861826),
 (u'Year_2012', -0.28564285942787804),
 (u'Year_2013', -0.2061067833854765),
 (u'Year_2014', -0.12546926551937987)]
```

In [241]:
```
# calculate the adjusted Sqft (Sqft * Year_mult) for the dataset and add it as a new column to data

# for each property, multiplier is sum of array [year_dummies] x [year_mults]

t = data[year_dummies.columns] * year_mults
t = t.T.sum()

t.name = 'year_multiplier'
t = t + 1
data = pd.concat([data, t], axis=1)

year_adj_sqft = data.area_adj_sqft * t
year_adj_sqft.name = 'adj_sqft'
data = pd.concat([data, year_adj_sqft], axis=1)

data.head()
```

Out[241]:

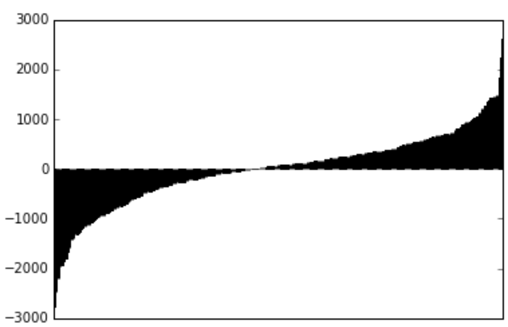
| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | Year_1969 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 0 |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 0 |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 0 |
| **7** | 8 | 8 | 8 | 1160 Mission Street #1112 | San Francisco (SOMA) | 1 | 1 | 664 | climbsf_renting | http://www.climbsf.com/for-rent/1160-mission-s... | ... | 0 |
| **11** | 12 | 12 | 12 | 655 26th Avenue | San Francisco (Central Richmond) | 2 | 1 | 1300 | climbsf_renting | http://www.climbsf.com/for-rent/655-26th-ave/ | ... | 0 |

5 rows × 67 columns

In [242]:
```
# run the regression based on year_and_area_adj_sqft rather than area_adj_sqft

# create X and y
feature_cols = ['adj_sqft']

X = data[feature_cols]
y = data.price

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
print lm.intercept_
# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))
print zip(feature_cols, lm.coef_)

# calculate predictions for the data set and plot errors
predictions = lm.predict(X)
errors = predictions-y
errors.name = 'Error'

# visualize the relationship between the features and the response using scatterplots
errors.sort(inplace=True)
errors.plot(kind='bar').get_xaxis().set_ticks([])

errors.tail(10)
```

```
1507.8595447
Residual sum of squares: 552478.52
Variance score: 0.69
[('adj_sqft', 2.1307024315775638)]
```

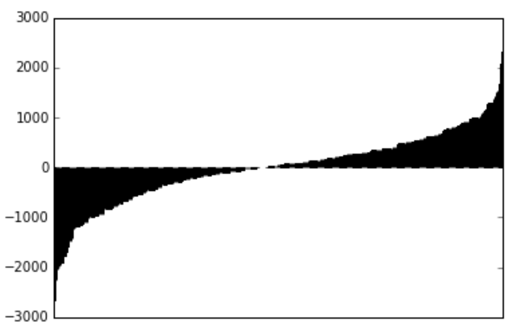
Out[242]:
```
202    1384.734919
194    1417.817265
318    1441.690199
313    1457.964199
108    1463.663204
294    1476.202741
236    1481.893854
455    1827.549514
60     2224.440816
523    2608.505581
Name: Error, dtype: float64
```



In [243]:
```
# create X and y
feature_cols = ['adj_sqft', 'bedrooms', 'bathrooms']

X = data[feature_cols]
y = data.price

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
print("Intercept: %.2f" % lm.intercept_)
# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))
print zip(feature_cols, lm.coef_)

# calculate predictions for the data set and plot errors
predictions = lm.predict(X)
errors = predictions-y
errors.name = 'Error'

# visualize the relationship between the features and the response using scatterplots
errors.sort()
errors.plot(kind='bar').get_xaxis().set_ticks([])
```

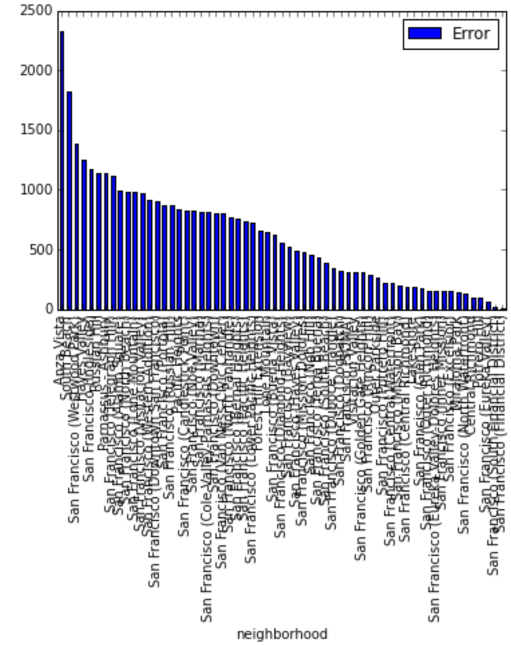
```
Intercept: 1269.34
Residual sum of squares: 509648.11
Variance score: 0.71
[('adj_sqft', 1.7729741412548015), ('bedrooms', 155.64516216444514), ('bathrooms', 323.83523244860413)]
```

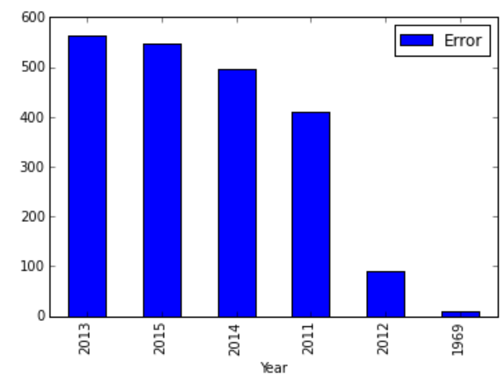
Out[243]: []

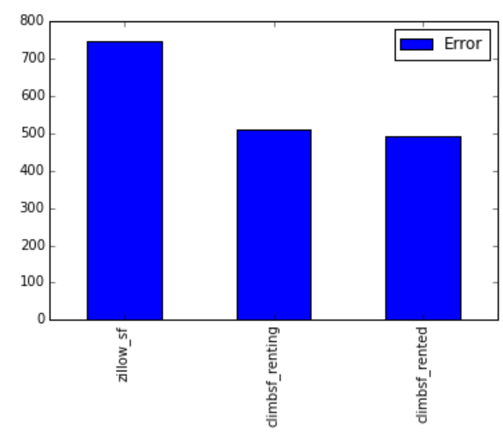


In [244]:
```
# show errors by neighborhood to see if there are any neighborhoods with funky differences

hooderrors = data[['neighborhood']]

errors = predictions-y
errors.name = 'Error'

hooderrors = pd.concat([hooderrors,errors.abs()],axis=1)

hood_group = hooderrors.groupby('neighborhood')

import numpy
def median(lst):
    return numpy.median(numpy.array(lst))

error_avg = hood_group.median()
error_avg.sort(columns='Error',ascending=False).plot(kind='bar')

# show errors by year to see if there are any years with funky differences

yearerrors = data[['Year']]

yearerrors = pd.concat([yearerrors,errors.abs()],axis=1)

year_group = yearerrors.groupby('Year')
error_avg = year_group.mean()
error_avg.sort(columns='Error',ascending=False).plot(kind='bar')

# show errors by source to see if there are any sources have noisy data

srcerrors = data[['source']]

srcerrors = pd.concat([srcerrors,errors.abs()],axis=1)

src_group = srcerrors.groupby('source')
error_avg = src_group.mean()
error_avg.sort(columns='Error',ascending=False).plot(kind='bar')
```

```
Out[244]: <matplotlib.axes._subplots.AxesSubplot at 0x111d8ac50>
```



In [245]:

```
import csv

table = ListTable()

dtype = [('Effect', 'S100'), ('Coefficient', float)]

# round to pennies
round_coef = map(round,lm.coef_,[6]*len(lm.coef_))
x = np.array(zip(feature_cols, round_coef),dtype=dtype)
x.T
print zip(feature_cols, lm.coef_)
#x = np.sort(x,axis=0,order='Coefficient')

with open('model_features_v1.csv', 'wb') as csvfile:
    modelwriter = csv.writer(csvfile, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)

    header = ['Effect','Coefficient']
    table.append(header)
    modelwriter.writerow(header)
    for i in x:
        table.append(i)
        modelwriter.writerow(i)


    table.append(['base_rent', lm.intercept_])


    modelwriter.writerow(['base_rent',lm.intercept_])

table
```

```
[('adj_sqft', 1.7729741412548015), ('bedrooms', 155.64516216444514), ('bathrooms', 323.83523244860413)]
```

Out[245]:

| Effect | Coefficient |
|---|---|
| adj_sqft | 1.772974 |
| bedrooms | 155.645162 |
| bathrooms | 323.835232 |
| base_rent | 1269.34366491 |

In [246]:
```
table = ListTable()

dtype = [('Effect', 'S100'), ('Coefficient', float)]

# round to pennies
round_coef = map(round,(area_mults + [1]*len(area_mults)),[6]*len(area_mults))
x = np.array(zip(area_dummies.columns, round_coef),dtype=dtype)
x.T
x = np.sort(x,axis=0,order='Coefficient')

with open('model_hoods_v1.csv', 'wb') as csvfile:
    hoodwriter = csv.writer(csvfile, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)

    header = ['Neighborhood','Multiplier']
    table.append(header)
    hoodwriter.writerow(header)

    for i in x:
        table.append(i)
        hoodwriter.writerow(i)

    lastrow = [base_area, 1]
    table.append(lastrow)
    hoodwriter.writerow(lastrow)


table
```

Out[246]:

| Neighborhood | Multiplier |
|---|---|
| neighborhood_Visitacion Valley | 0.814714 |
| neighborhood_Glen Park | 0.866884 |
| neighborhood_Bernal Heights | 0.884279 |
| neighborhood_Diamond Heights | 0.892706 |
| neighborhood_Lakeshore | 0.911191 |
| neighborhood_Excelsior | 0.932129 |
| neighborhood_Ocean View | 1.020502 |
| neighborhood_Outer Richmond | 1.041344 |
| neighborhood_Haight Ashbury | 1.044439 |
| neighborhood_Parkside | 1.116116 |
| neighborhood_West of Twin Peaks | 1.122335 |
| neighborhood_Outer Sunset | 1.125854 |
| neighborhood_Potrero Hill | 1.323934 |
| neighborhood_Noe Valley | 1.340829 |
| neighborhood_Marina | 1.35758 |
| neighborhood_South of Market | 1.35983 |
| neighborhood_Inner Richmond | 1.395115 |
| neighborhood_Downtown/Civic Center | 1.399821 |
| neighborhood_North Beach | 1.40525 |
| neighborhood_Nob Hill | 1.419245 |
| neighborhood_Pacific Heights | 1.482074 |
| neighborhood_Mission | 1.513632 |
| neighborhood_Financial District | 1.524251 |
| neighborhood_Castro/Upper Market | 1.535213 |
| neighborhood_Inner Sunset | 1.615484 |
| neighborhood_Russian Hill | 1.786712 |
| neighborhood_Western Addition | 1.802645 |
| neighborhood_Chinatown | 2.648812 |
| neighborhood_Bayview | 1 |

In [247]:
```
# show negative errors meaning we expected rents to be higher

error = predictions-y
error.name = 'error'

data = pd.concat([data,error,pd.DataFrame(predictions,columns=['predicted_price'])],axis=1)

data.head()
```

Out[247]:

| | property_id | transaction_log_id | id | address | neighborhood | bedrooms | bathrooms | sqft | source | origin_url | ... | Year_2012 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| **1** | 2 | 2 | 2 | 252 Granada Avenue | San Francisco (Ingleside) | 2 | 2 | 1600 | climbsf_renting | http://www.climbsf.com/for-rent/252-granada-ave/ | ... | 0 |
| **2** | 3 | 3 | 3 | 460 Valley Street | San Francisco (Noe Valley) | 2 | 2 | 1446 | climbsf_renting | http://www.climbsf.com/for-rent/460-valley-st/ | ... | 0 |
| **3** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| **4** | 5 | 5 | 5 | 420 Mission Bay Boulevard North #121 | San Francisco (Mission Bay) | 1 | 1 | 980 | climbsf_renting | http://www.climbsf.com/for-rent/420-mission-ba... | ... | 0 |

5 rows × 69 columns

In [248]:
```
# filter out overshoot error
overshoot = data[(data.error <= -500)]
columns = data.columns - ['error','latitude', 'longitude', 'address', 'origin_url','price','neighborhood']
overshoot = data.drop(columns,1)
overshoot.sort('error',ascending=True,inplace=True)
overshoot.head(30)
```

Out[248]:

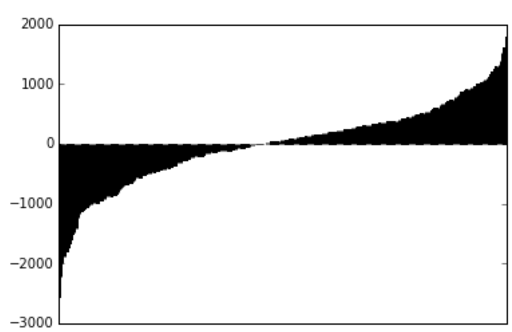
| | address | neighborhood | origin_url | latitude | longitude | price | error |
|---|---|---|---|---|---|---|---|
| **546** | 301 Main St UNIT 35A, San Francisco, CA 94105 | South Beach | http://www.zillow.com/homedetails/301-Main-St-... | 37.7894 | -122.391 | 7950 | -2664.618204 |
| **233** | 338 Spear Street #39E | San Francisco (South Beach) | http://www.climbsf.com/for-rent/338-spear-st-39e/ | 37.7894 | -122.391 | 7975 | -2255.490601 |
| **517** | 20th St San Francisco, CA 94110 | None | http://www.zillow.com/homedetails/20th-St-San-... | 37.7588 | -122.416 | 6200 | -2035.909097 |
| **158** | 338 Spear Street #39A | San Francisco (South Beach) | http://www.climbsf.com/for-rent/338-spear-st-39a/ | 37.7894 | -122.391 | 6700 | -1983.453547 |
| **273** | 301 Mission Street #29F | San Francisco (South Beach) | http://www.climbsf.com/for-rent/301-mission-st... | 37.7905 | -122.396 | 7975 | -1962.920623 |
| **89** | 88 King Street #904 | San Francisco (South Beach) | http://www.climbsf.com/for-rent/88-king-st-904/ | 37.7807 | -122.389 | 6250 | -1895.205653 |
| **434** | 748 Bay St, San Francisco, CA 94109 | Russian Hill | http://www.zillow.com/homedetails/748-Bay-St-S... | 37.8049 | -122.419 | 7500 | -1771.282008 |
| **299** | 401 Harrison Street #3803 | San Francisco (Rincon Hill) | http://www.climbsf.com/for-rent/401-harrison-s... | 37.7864 | -122.392 | 7225 | -1768.441160 |
| **232** | 2560 Vallejo Street | San Francisco (Pacific Heights) | http://www.climbsf.com/for-rent/2560-vallejo-st/ | 37.7950 | -122.439 | 7050 | -1678.668048 |
| **381** | 301 Main Street #35F | San Francisco (South Beach) | http://www.climbsf.com/for-rent/301-main-st-35f/ | 37.7894 | -122.391 | 7000 | -1582.016268 |
| **525** | 20th St San Francisco, CA 94114 | None | http://www.zillow.com/homedetails/20th-St-San-... | 37.7578 | -122.432 | 5700 | -1481.921519 |
| **382** | 480 Mission Bay Boulevard North #PH1606 | San Francisco (Mission Bay) | http://www.climbsf.com/for-rent/480-mission-ba... | 37.7731 | -122.393 | 7500 | -1472.047366 |
| **283** | 234 Grand View Avenue | San Francisco (Noe Valley) | http://www.climbsf.com/for-rent/234-grand-view... | 37.7545 | -122.441 | 7300 | -1419.202673 |
| **411** | Vallejo St San Francisco, CA 94133 | None | http://www.zillow.com/homedetails/Vallejo- | 37.7985 | -122.410 | 4000 | -1229.429094 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | St-S... | | | | |
| **293** | 425 1st Street #3402 | San Francisco (Rincon Hill) | http://www.climbsf.com/for-rent/425-1st-st-3402/ | 37.7858 | -122.392 | 6600 | -1215.769488 |
| **203** | 461 2nd St. #557T | San Francisco (South Beach) | http://www.climbsf.com/for-rent/461-2nd-st-557t/ | 37.7838 | -122.394 | 6750 | -1180.673838 |
| **459** | Tehama St San Francisco, CA 94103 | None | http://www.zillow.com/homedetails/Tehama-St-Sa... | 37.7793 | -122.407 | 6000 | -1167.299698 |
| **457** | Lombard St San Francisco, CA 94133 | None | http://www.zillow.com/homedetails/Lombard-St-S... | 37.8021 | -122.419 | 6700 | -1152.566463 |
| **119** | 1837 Jefferson Street | San Francisco (Marina) | http://www.climbsf.com/for-rent/1837-jefferson... | 37.8045 | -122.443 | 6200 | -1140.531105 |
| **357** | 296 Francisco Street | San Francisco (Telegraph Hill) | http://www.climbsf.com/for-rent/296-francisco-st/ | 37.8053 | -122.410 | 5475 | -1111.447417 |
| **316** | 35 Dolores Street #410 | San Francisco (Mission Dolores) | http://www.climbsf.com/for-rent/35-dolores-st-... | 37.7686 | -122.427 | 6050 | -1099.802300 |
| **204** | 1839 Jefferson Street | San Francisco (Marina) | http://www.climbsf.com/for-rent/1839-jefferson... | 37.8048 | -122.443 | 6400 | -1098.461283 |
| **134** | 301 Main Street #25E | San Francisco (South Beach) | http://www.climbsf.com/for-rent/301-main-st-25e/ | 37.7894 | -122.391 | 5800 | -1083.730601 |
| **282** | 301 Main Street #5C | San Francisco (South Beach) | http://www.climbsf.com/for-rent/301-main-st-5c/ | 37.7894 | -122.391 | 7000 | -1018.663679 |
| **405** | Vallejo St San Francisco, CA 94123 | None | http://www.zillow.com/homedetails/Vallejo-St-S... | 37.7952 | -122.435 | 4200 | -998.230614 |
| **109** | 229 Brannan Street #12J | San Francisco (South Beach) | http://www.climbsf.com/for-rent/229-brannan-st... | 37.7826 | -122.390 | 5950 | -994.199509 |
| **113** | 301 Mission Street #701 | San Francisco (SOMA) | http://www.climbsf.com/for-rent/301-mission-st... | 37.7905 | -122.396 | 7400 | -977.315886 |
| **430** | 501 Beale St, San Francisco, CA 94105 | South Beach | http://www.zillow.com/homedetails/501-Beale-St... | 37.7863 | -122.389 | 6000 | -975.000185 |
| **174** | 301 Main Street #14F | San Francisco (South Beach) | http://www.climbsf.com/for-rent/301-main-st-14f/ | 37.7894 | -122.391 | 5950 | -965.958102 |
| **72** | 235 Berry Street #107 | San Francisco (Mission Bay) | http://www.climbsf.com/for-rent/235-berry-st-1... | 37.7749 | -122.394 | 7500 | -964.150538 |

In [249]:
```
data = data[(data.sqft <= 2500) & (data.price <= 8000) & (data.price != 0) & (data.bedrooms <= 4) & (data.bathrooms <= 3)
& (data.sqft != 0)]



# add squared square footage to the table
squared = data.adj_sqft ** 2
squared.name = 'sqft_squared'

squared_beds = data.bedrooms ** 2
squared_beds.name = 'beds_squared'

data = pd.concat([data, squared, squared_beds], axis=1)
#data = pd.concat([data, squared_beds], axis=1)



# create X and y
feature_cols = ['adj_sqft', 'bedrooms', 'bathrooms','sqft_squared','beds_squared']

X = data[feature_cols]
y = data.price

# instantiate, fit
lm = LinearRegression()
lm.fit(X, y)

# print coefficients
print("Intercept: %.2f" % lm.intercept_)
# The mean square error
print("Residual sum of squares: %.2f"
      % np.mean((lm.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Variance score: %.2f' % lm.score(X, y))
print zip(feature_cols, lm.coef_)

# calculate predictions for the data set and plot errors
predictions = lm.predict(X)
errors = predictions-y
errors.name = 'Error'

# visualize the relationship between the features and the response using scatterplots
errors.sort()
errors.plot(kind='bar').get_xaxis().set_ticks([])
```

```
Intercept: 247.37
Residual sum of squares: 474109.65
Variance score: 0.73
[('adj_sqft', 3.6996192523678673), ('bedrooms', -157.61331809822042), ('bathrooms', 316.48288907402383), ('sqft_squared',
-0.00059060553655249949), ('beds_squared', 53.514005384693448)]
```

Out[249]: []



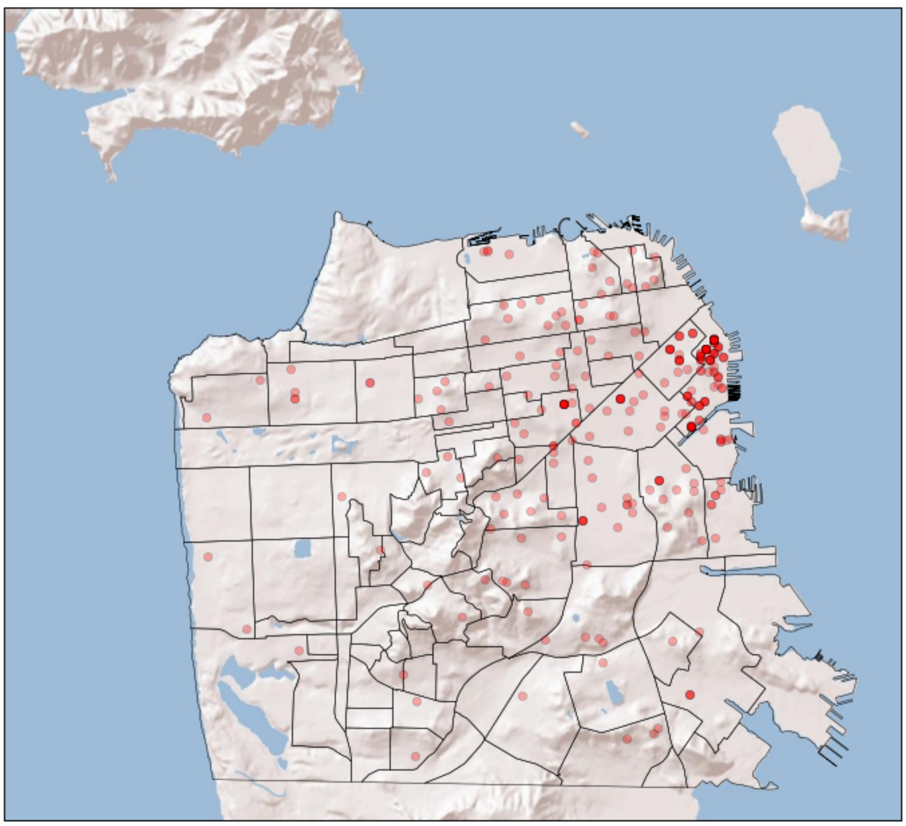
In [250]:
```
import statsmodels.formula.api as sm
result = sm.ols(formula="price ~ adj_sqft + bedrooms + bathrooms + elevation", data=data).fit()
print result.params
print result.summary()
```

```
Intercept    1355.629789
adj_sqft        1.756130
bedrooms      223.018916
bathrooms     277.936411
elevation      -3.520630
dtype: float64
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.721
Model:                            OLS   Adj. R-squared:                  0.717
Method:                 Least Squares   F-statistic:                     192.5
Date:                Sun, 16 Aug 2015   Prob (F-statistic):           2.69e-81
Time:                        13:17:50   Log-Likelihood:                -2416.8
No. Observations:                 303   AIC:                             4844.
Df Residuals:                     298   BIC:                             4862.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept   1355.6298    131.263     10.328      0.000      1097.309  1613.950
adj_sqft       1.7561      0.109     16.059      0.000         1.541     1.971
bedrooms     223.0189     75.983      2.935      0.004        73.488   372.550
bathrooms    277.9364     99.297      2.799      0.005        82.524   473.349
elevation     -3.5206      1.182     -2.979      0.003        -5.846    -1.195
==============================================================================
Omnibus:                       18.909   Durbin-Watson:                   1.769
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               22.344
Skew:                           0.532   Prob(JB):                     1.41e-05
Kurtosis:                       3.800   Cond. No.                     5.15e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.15e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

In [251]:
```
from mpl_toolkits.basemap import Basemap
import fiona
```

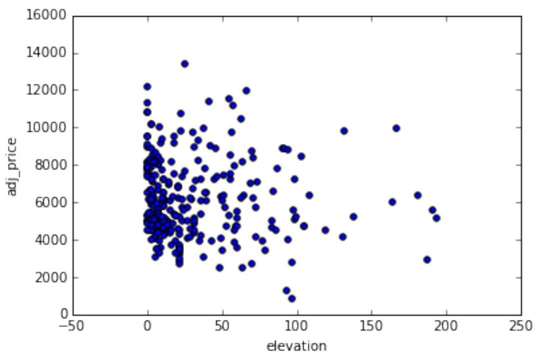
In [252]:
```
plt.figure(figsize=(12,12))


# Create the Basemap
event_map = Basemap(projection='merc',
                    resolution='h', epsg=2227,
                    lat_0 = 37.7, lon_0=-122.4, # Map center
                    llcrnrlon=-122.55, llcrnrlat=37.7, # Lower left corner
                    urcrnrlon=-122.35, urcrnrlat=37.85) # Upper right corner

# Draw important features
event_map.arcgisimage(service='World_Shaded_Relief', xpixels = 1500, verbose= True)

# add neighborhoods
event_map.readshapefile(
    'data/Realtor_Neighborhoods_4326/hoods_4326', 'SF', color='black', zorder=2)

# create array storing lats and longs
listing_coords = zip(data.latitude,data.longitude)

# Draw the points on the map:
for longitude, latitude in listing_coords:
    x, y = event_map(latitude, longitude) # Convert lat, long to y,x
    event_map.plot(x,y, 'ro', alpha=0.3)
```

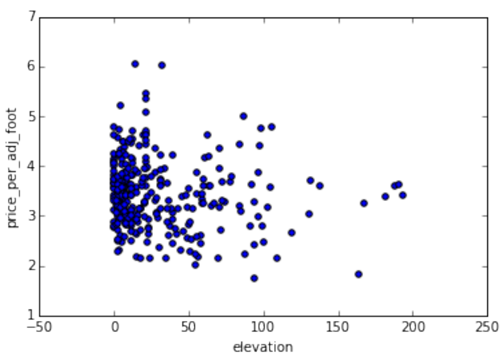
```
http://server.arcgisonline.com/ArcGIS/rest/services/World_Shaded_Relief/MapServer/export?bbox=5968621.97922,2083843.65958,
6027551.68158,2137245.61137&bboxSR=2227&imageSR=2227&size=1500,1359&dpi=96&format=png32&f=image
```



In [253]:
```
price_per_adj_foot = data['price'] / data['adj_sqft']
price_per_adj_foot.name = 'price_per_adj_foot'
adj_price = data['price'] * data['area_multiplier']
adj_price.name = 'adj_price'
data = pd.concat([data, price_per_adj_foot, adj_price], axis=1)

# visualize the relationship between the features and the response using scatterplots
data.plot(kind='scatter', x='elevation', y='adj_price')
data.plot(kind='scatter', x='elevation', y='price_per_adj_foot')
```

Out[253]: <matplotlib.axes._subplots.AxesSubplot at 0x10d822b90>




In [ ]: