Juhi_Srivastava_HW_10-17

October 17, 2016

```
In [1]: import numpy as np
        import pandas as pd
        import lxml as sd
        import html5lib as hlib
        import seaborn as sa
        import matplotlib.pyplot as plt
        %matplotlib inline
```

```
In [2]: df = pd.read_csv("/Users/juhi/data/Newark.csv", thousands=',')
        df = df.dropna()
        column_names = {'Class Desc': 'Status'}
        df = (df.rename(columns=column_names))

        df['StatusValue'] = df['Status']
        df.loc[df['Status'] == 'Vacant', 'StatusValue'] = '1'
        df.loc[df['Status'] == 'Abandoned', 'StatusValue'] = '2'

        df["StatusValue"] = df["StatusValue"].astype(int)

        df.head()
```

```
Out[2]:    Vital House Number Vital Street Name   Block   Lot   Latitude  Longitude
        9                 451     WASHINGTON ST   115.0  17.0  40.729412 -74.179463
        13                 35          GOBLE ST  1166.0   8.0  40.720251 -74.173605
        15               1011          BROAD ST   118.0   3.0  40.728728 -74.176753
        17                161          EMMET ST  1180.0  21.0  40.719006 -74.177085
        28            503-505     WASHINGTON ST   121.0  48.0  40.727758 -74.180254

           Most Recent Inspection  Status                          Owner Name  \
        9                 5/15/15  Vacant            FD WASHINGTON REALTY LLC
        13                7/18/16  Vacant                CASTILO, MAURA CHICA
        15                 9/3/15  Vacant   LIP LIFE REALTY INC C/O BYUNG JUN
        17                5/24/16  Vacant  EVERGREEN PROPERTY SOLUTIONS, LLC.
        28                 8/6/15  Vacant                                CAPC

                   Owner Address        City, State  Zipcode  NetValue  StatusValue
        9      156 WASHINGTON ST         NEWARK, NJ   7102.0  106000.0            1
        13           35 GOBLE ST         NEWARK, NJ   7105.0  106800.0            1
```

```
15         15 LIBERTY ST   LITTLE FERRY, NJ   7643.0  165500.0            1
17            PO BOX 278     LIVINGSTON, NJ   7105.0  161000.0            1
28  108 CHURCH ST 3RD FL  NEW BRUNSWICK, NJ   8816.0  116800.0            1
```

```
In [3]: df2 = pd.read_csv("/Users/juhi/data/income.csv")
        df2 = df2.dropna()
        df2['Mean'] = df2['Mean'].str.replace(',', '')
        column_names = {'Zip': 'Zipcode', 'Mean': 'AvgSalary'}
        df2 = (df2.rename(columns=column_names))
        df2 = df2.drop('Pop',1)

        df2.head()
```

```
Out[3]:    Zipcode AvgSalary
        0     1001     66688
        1     1002     75063
        2     1003     35121
        3     1005     82442
        4     1007     85802
```

```
In [4]: df3 = pd.merge(df, df2, on='Zipcode', how="inner")
        df3 = df3.sort_values(by="AvgSalary", ascending=True)

        df3.loc[df3['Status'] == 'Vacant', 'Abandoned'] = '0'
        df3.loc[df3['Status'] == 'Abandoned', 'Abandoned'] = '1'

        df3["Abandoned"] = df3["Abandoned"].astype(int)
        df3["Zipcode"] = df3["Zipcode"].astype(int)
        df3["AvgSalary"] = df3["AvgSalary"].astype(int)

        df3.head()
```

```
Out[4]:     Vital House Number Vital Street Name   Block   Lot   Latitude  Longitud
        800                  9         WARREN ST    64.0  23.0  40.739274 -74.17246
        788            485-487         S 18TH ST   325.0  16.0  40.739947 -74.20891
        361                337         S 19TH ST  1794.0  25.0  40.745446 -74.20751
        43             503-505     WASHINGTON ST   121.0  48.0  40.727758 -74.18025
        375                 24       FLEMING AVE  2017.0  33.0  40.730977 -74.14733

            Most Recent Inspection     Status                     Owner Name  \
        800                5/15/15     Vacant  MRW REALTY LLC C/O M. WEBER
        788                9/23/15  Abandoned         GELT PROPERTIES LLC,
        361                9/26/16     Vacant    CONTEMPORARY VISIONS, LLC
        43                  8/6/15     Vacant                         CAPC
        375                7/11/16     Vacant        24 F L MANAGEMENT LL,

                         Owner Address            City, State  Zipcode  NetValue  \
        800              71 GARWOOD RD          FAIR LAWN, NJ     7410  347500.0
        788  2755 PHILMONT AVE STE 130  HUNTINGDON VALLEY, PA    19006   46600.0
```

```
361         623 EAGLE ROCK AVE        WEST ORANGE, NJ     7052  177800.0
43        108 CHURCH ST 3RD FL      NEW BRUNSWICK, NJ     8816  116800.0
375                119 35TH ST          UNION CITY,NJ    75024  131300.0

     StatusValue  AvgSalary  Abandoned
800            1     108521          0
788            2     114065          1
361            1     117284          0
43             1     117473          0
375            1     117938          0
```

```
In [5]: df4 = df3[['Zipcode','AvgSalary','Abandoned','Latitude', 'Longitude', 'NetV
        df4.head()
```

```
Out[5]:      Zipcode  AvgSalary  Abandoned   Latitude  Longitude  NetValue
        800     7410     108521          0  40.739274 -74.172463  347500.0
        788    19006     114065          1  40.739947 -74.208918   46600.0
        361     7052     117284          0  40.745446 -74.207515  177800.0
        43      8816     117473          0  40.727758 -74.180254  116800.0
        375    75024     117938          0  40.730977 -74.147337  131300.0
```

```
In [6]: from collections import Counter

        df5 = df4.groupby(['Zipcode'])['Abandoned'].sum()

        df5 = df5.to_frame().reset_index()
        df5.columns = ['Zipcode', 'Abandoned']

        full_sub_df = pd.merge(df4, df5, on='Zipcode')
        full_sub_df= full_sub_df [['Zipcode','Abandoned_y', 'Longitude','Latitude',
        column_names = {'Abandoned_y': 'Abandoned#'}
        full_sub_df = (full_sub_df.rename(columns=column_names))
        full_sub_df
```

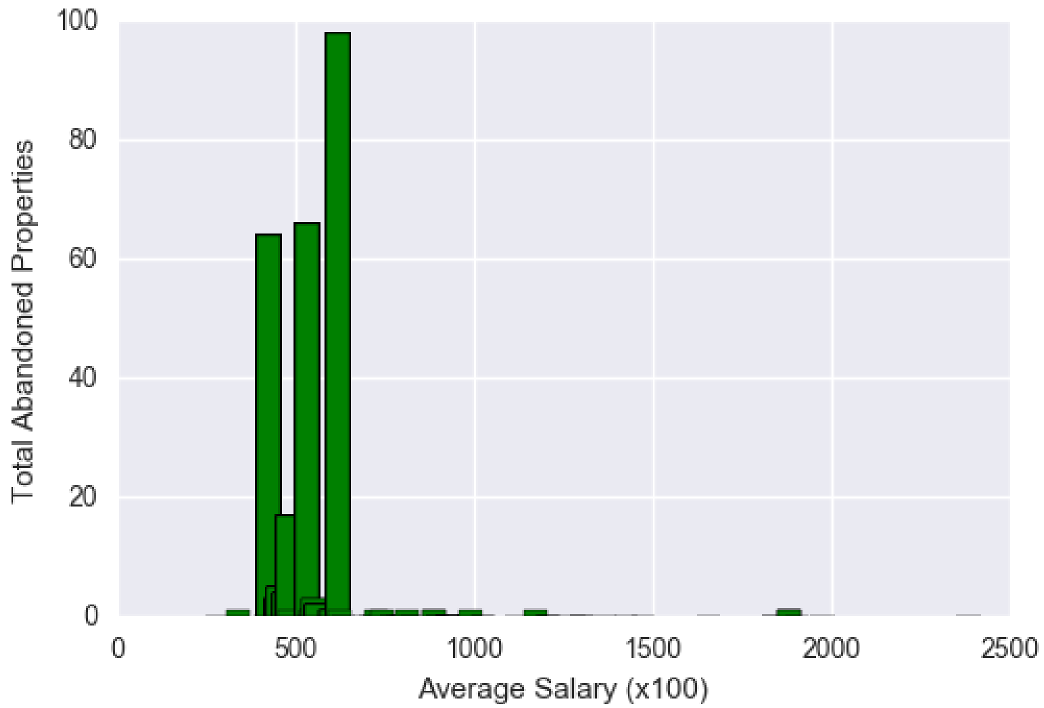
```
Out[6]:      Zipcode  Abandoned#  Longitude   Latitude  AvgSalary
        0       7410           0 -74.172463  40.739274     108521
        1      19006           1 -74.208918  40.739947     114065
        2       7052           0 -74.207515  40.745446     117284
        3       8816           0 -74.180254  40.727758     117473
        4      75024           0 -74.147337  40.730977     117938
        5       7652           0 -74.202112  40.718564     124464
        6       7042           0 -74.208074  40.745291     127038
        7       7042           0 -74.197173  40.737446     127038
        8      10019           0 -74.171198  40.736266     133175
        9       7006           0 -74.171359  40.742304     139504
        10      7960           0 -74.180682  40.746531     143929
        11     33156           0 -74.206753  40.745741     162451
        12     10128           0 -74.195789  40.717244     180473
        13     11598           0 -74.173243  40.741958     180670
```

```
14      7670           0 -74.139120  40.730934     184764
15      7039           1 -74.206925  40.727180     184778
16      7039           1 -74.196945  40.738039     184778
17      7039           1 -74.171533  40.735808     184778
18     11530           0 -74.209226  40.707053     187098
19     10023           0 -74.175665  40.745359     194366
20      7417           0 -74.161577  40.774433     235259
21      7114           0 -74.204467  40.690555      25040
22      7501           1 -74.229608  40.749470      30516
23     19104           0 -74.211277  40.731239      38468
24      7108          64 -74.203466  40.724923      39145
25      7108          64 -74.202597  40.725361      39145
26      7108          64 -74.200904  40.725384      39145
27      7108          64 -74.201371  40.724645      39145
28      7108          64 -74.201436  40.724490      39145
29      7108          64 -74.203344  40.725212      39145
..       ...         ...        ...        ...        ...
776     7205           0 -74.207504  40.746249      68893
777     7205           0 -74.201357  40.715209      68893
778     7203           1 -74.197280  40.742587      69366
779     7062           0 -74.195176  40.746845      70074
780    32713           1 -74.168080  40.759797      70776
781     7643           0 -74.176753  40.728728      73771
782     7643           0 -74.188746  40.746487      73771
783    30054           0 -74.206932  40.716252      75106
784     8879           0 -74.187442  40.721903      76378
785    11422           0 -74.198226  40.752073      76689
786    10310           0 -74.187464  40.721862      77596
787     7204           1 -74.164370  40.776417      77751
788     8723           0 -74.165571  40.783729      78541
789    11423           0 -74.206831  40.740599      79561
790    11428           1 -74.197086  40.751102      85684
791    11428           1 -74.209151  40.737768      85684
792     7508           0 -74.161388  40.770299      86820
793     7508           0 -74.205235  40.723346      86820
794     7083           0 -74.187405  40.758556      89152
795     7083           0 -74.209094  40.726738      89152
796     7083           0 -74.196899  40.749535      89152
797     7083           0 -74.175436  40.745095      89152
798     7083           0 -74.209853  40.718336      89152
799     7663           0 -74.208584  40.738230      92553
800    19103           0 -74.207853  40.745543      92863
801     7024           0 -74.171787  40.739816      95417
802     7024           0 -74.172335  40.740575      95417
803    33146           1 -74.208435  40.741204      95773
804     7302           0 -74.208969  40.746298      98752
805     8873           0 -74.169718  40.749335      99431
```

```
        [806 rows x 5 columns]
```

```
In [7]: width = 65

        plt.bar(full_sub_df['AvgSalary']/100, (full_sub_df['Abandoned#']), width, c
        plt.ylabel('Total Abandoned Properties')
        plt.xlabel('Average Salary (x100)')
```

```
Out[7]: <matplotlib.text.Text at 0x10f7236a0>
```



```
In [8]: full_sub_df['Abandoned#'].describe()
```

```
Out[8]: count    806.000000
        mean      60.214640
        std       33.827951
        min        0.000000
        25%       64.000000
        50%       64.000000
        75%       98.000000
        max       98.000000
        Name: Abandoned#, dtype: float64
```

```
In [9]: full_sub_df['AvgSalary'].describe()
```

```
Out[9]: count       806.000000
        mean      52742.736973
        std       20437.993278
        min       25040.000000
        25%       39145.000000
        50%       49988.000000
        75%       58633.000000
        max      235259.000000
        Name: AvgSalary, dtype: float64
```

```
In [10]: from bokeh.io import output_file, show
         from bokeh.models import GeoJSONDataSource
         from bokeh.plotting import figure
         from bokeh.sampledata.sample_geojson import geojson
         from bokeh.io import output_file, show
         from bokeh.models import (
             GMapPlot, GMapOptions, ColumnDataSource, Circle, DataRange1d, PanTool,
         )

         geo_source = GeoJSONDataSource(geojson=geojson)

         source = ColumnDataSource(ColumnDataSource.from_df(data=full_sub_df))
         print(source)
```

```
ColumnDataSource, ViewModel:ColumnDataSource, ref _id: df6fd10f-0a06-415c-91cc-a38
```

```
In [11]: api_key="AIzaSyCm19igMhPstn_HQatZurkitBwyOP6nZ1w"
         map_options = GMapOptions(lat=40.73, lng=-74.16, map_type="roadmap", zoom=

         plot = GMapPlot(
             x_range=DataRange1d(), y_range=DataRange1d(), map_options=map_options,
         )
         plot.title.text="Newark Income to Property Abandonment"
         circle = Circle(x="Longitude", y="Latitude", size=10, fill_color="green",
         plot.add_glyph(source, circle)

         plot.add_tools(PanTool(), WheelZoomTool(), BoxSelectTool())
         output_file("gmap_plot.html")
         show(plot)
```

```
In [18]: #BONUS QUESTION
         from geopy.geocoders import Nominatim
         full_sub_df["house_zipcode"] = full_sub_df.apply(lambda x: Nominatim().rev
         full_sub_df.head()
```

```
Out[18]:    Zipcode  Abandoned#  Longitude   Latitude  AvgSalary house_zipcode
         0     7410           0 -74.172463  40.739274     108521         07102
         1    19006           1 -74.208918  40.739947     114065         07103
```

```
        2     7052           0 -74.207515  40.745446     117284         07103
        3     8816           0 -74.180254  40.727758     117473         07102
        4    75024           0 -74.147337  40.730977     117938         07105
```

```
In [19]: full_sub_df.to_csv("With_house_zip.csv")
```

```
In [48]: temp_house_income = pd.read_csv("/Users/juhi/data/income.csv")
         temp_house_income = temp_house_income.dropna()
         temp_house_income['Mean'] = temp_house_income['Mean'].str.replace(',', '')
         column_names = {'Zip': 'house_zipcode', 'Mean': 'House_Salary'}
         temp_house_income = (temp_house_income.rename(columns=column_names))
         temp_house_income = temp_house_income.drop('Pop',1)
         temp_house_income['house_zipcode'] = temp_house_income.apply(lambda x: st
         temp_house_income.head()
```

```
Out[48]:   house_zipcode House_Salary
        0          01001        66688
        1          01002        75063
        2          01003        35121
        3          01005        82442
        4          01007        85802
```

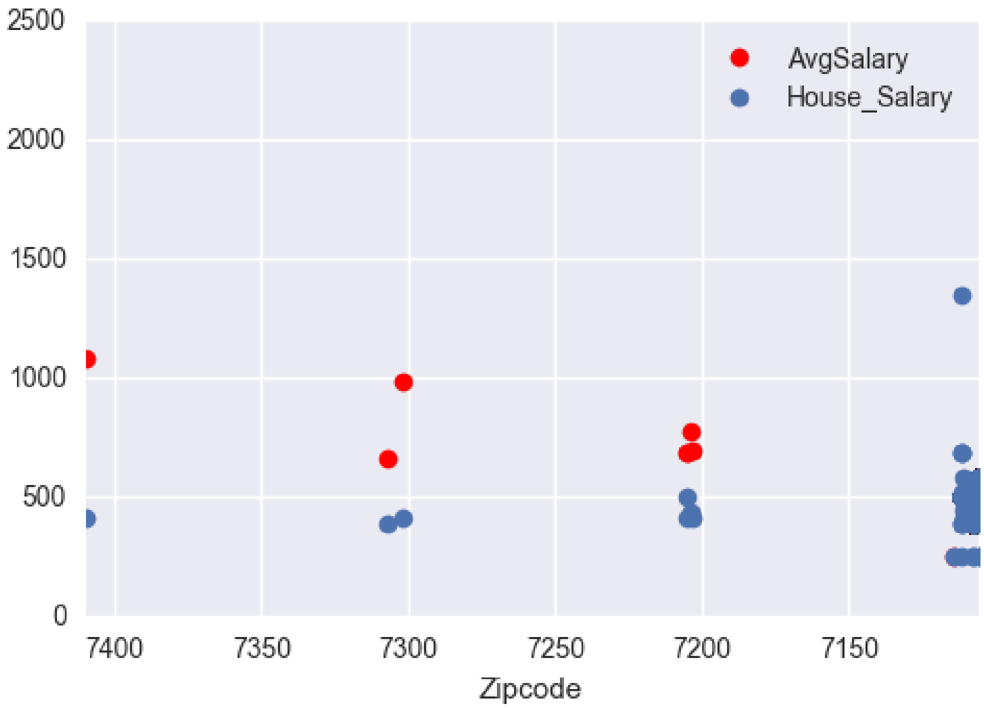
```
In [81]: bonus_df = pd.merge(full_sub_df, temp_house_income, on='house_zipcode', ho
         bonus_df["House_Salary"] = bonus_df["House_Salary"].astype(int)/100
         bonus_df["AvgSalary"] = bonus_df["AvgSalary"].astype(int)/100
         bonus_df.head()
```

```
Out[81]:    Zipcode  Abandoned#  Longitude   Latitude  AvgSalary house_zipcode  \
        0     7410           0 -74.172463  40.739274    1085.21         07102
        1     8816           0 -74.180254  40.727758    1174.73         07102
        2    10019           0 -74.171198  40.736266    1331.75         07102
        3     7006           0 -74.171359  40.742304    1395.04         07102
        4    11598           0 -74.173243  40.741958    1806.70         07102

           House_Salary
        0        412.33
        1        412.33
        2        412.33
        3        412.33
        4        412.33
```

```
In [82]: bonus_df1 = bonus_df[['Zipcode','AvgSalary','House_Salary']]
         bonus_df1.head()
```

```
Out[82]:    Zipcode  AvgSalary  House_Salary
        0     7410    1085.21        412.33
        1     8816    1174.73        412.33
        2    10019    1331.75        412.33
        3     7006    1395.04        412.33
        4    11598    1806.70        412.33
```

```
In [84]: bonus_df1.plot(x="Zipcode", style=['ro','o'])

Out[84]: <matplotlib.axes._subplots.AxesSubplot at 0x10339aba8>
```



```
In [ ]:
```