Visualizing Earnings Based on College Majors

May 24, 2018

0.0.1 Introduction

```
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
        % matplotlib inline
        recent_grads = pd.read_csv("./databank/recent-grads.csv")
        print(recent_grads.iloc[0])
        print(recent_grads.head())
        print(recent_grads.tail())
        recent_grads.describe()
```

```
Rank                                        1
Major_code                               2419
Major                   PETROLEUM ENGINEERING
Total                                    2339
Men                                      2057
Women                                     282
Major_category                    Engineering
ShareWomen                           0.120564
Sample_size                                36
Employed                                 1976
Full_time                                1849
Part_time                                 270
Full_time_year_round                     1207
Unemployed                                 37
Unemployment_rate                   0.0183805
Median                                 110000
P25th                                   95000
P75th                                  125000
College_jobs                             1534
Non_college_jobs                          364
Low_wage_jobs                             193
Name: 0, dtype: object
   Rank  Major_code                                      Major    Total  \
0     1        2419                      PETROLEUM ENGINEERING   2339.0
1     2        2416             MINING AND MINERAL ENGINEERING    756.0
2     3        2415                  METALLURGICAL ENGINEERING    856.0
3     4        2417  NAVAL ARCHITECTURE AND MARINE ENGINEERING   1258.0
```

```
4     5        2405                       CHEMICAL ENGINEERING  32260.0

       Men    Women Major_category  ShareWomen  Sample_size  Employed  \
0   2057.0    282.0    Engineering    0.120564           36      1976
1    679.0     77.0    Engineering    0.101852            7       640
2    725.0    131.0    Engineering    0.153037            3       648
3   1123.0    135.0    Engineering    0.107313           16       758
4  21239.0  11021.0    Engineering    0.341631          289     25694

       ...        Part_time  Full_time_year_round  Unemployed  \
0      ...              270                  1207          37
1      ...              170                   388          85
2      ...              133                   340          16
3      ...              150                   692          40
4      ...             5180                 16697        1672

   Unemployment_rate  Median  P25th   P75th  College_jobs  Non_college_jobs  \
0           0.018381  110000  95000  125000          1534               364
1           0.117241   75000  55000   90000           350               257
2           0.024096   73000  50000  105000           456               176
3           0.050125   70000  43000   80000           529               102
4           0.061098   65000  50000   75000         18314              4440

   Low_wage_jobs
0            193
1             50
2              0
3              0
4            972

[5 rows x 21 columns]
     Rank  Major_code                   Major   Total     Men   Women  \
168   169        3609                 ZOOLOGY  8409.0  3050.0  5359.0
169   170        5201  EDUCATIONAL PSYCHOLOGY  2854.0   522.0  2332.0
170   171        5202     CLINICAL PSYCHOLOGY  2838.0   568.0  2270.0
171   172        5203   COUNSELING PSYCHOLOGY  4626.0   931.0  3695.0
172   173        3501         LIBRARY SCIENCE  1098.0   134.0   964.0

               Major_category  ShareWomen  Sample_size  Employed  \
168    Biology & Life Science    0.637293           47      6259
169  Psychology & Social Work    0.817099            7      2125
170  Psychology & Social Work    0.799859           13      2101
171  Psychology & Social Work    0.798746           21      3777
172                 Education    0.877960            2       742

         ...        Part_time  Full_time_year_round  Unemployed  \
168      ...             2190                  3602         304
169      ...              572                  1211         148
```

```
170      ...              648                  1293         368
171      ...              965                  2738         214
172      ...              237                   410          87

     Unemployment_rate  Median  P25th  P75th  College_jobs  Non_college_jobs  \
168           0.046320   26000  20000  39000          2771              2947
169           0.065112   25000  24000  34000          1488               615
170           0.149048   25000  25000  40000           986               870
171           0.053621   23400  19200  26000          2403              1245
172           0.104946   22000  20000  22000           288               338

     Low_wage_jobs
168            743
169             82
170            622
171            308
172            192

[5 rows x 21 columns]
```

```
Out[1]:         Rank  Major_code          Total            Men          Women  \
        count  173.000000  173.000000     172.000000     172.000000     172.000000
        mean    87.000000  3879.815029   39370.081395   16723.406977   22646.674419
        std     50.084928  1687.753140   63483.491009   28122.433474   41057.330740
        min      1.000000  1100.000000     124.000000     119.000000       0.000000
        25%     44.000000  2403.000000    4549.750000    2177.500000    1778.250000
        50%     87.000000  3608.000000   15104.000000    5434.000000    8386.500000
        75%    130.000000  5503.000000   38909.750000   14631.000000   22553.750000
        max    173.000000  6403.000000  393735.000000  173809.000000  307087.000000

               ShareWomen  Sample_size       Employed      Full_time      Part_time  \
        count  172.000000   173.000000     173.000000     173.000000     173.000000
        mean     0.522223   356.080925   31192.763006   26029.306358    8832.398844
        std      0.231205   618.361022   50675.002241   42869.655092   14648.179473
        min      0.000000     2.000000       0.000000     111.000000       0.000000
        25%      0.336026    39.000000    3608.000000    3154.000000    1030.000000
        50%      0.534024   130.000000   11797.000000   10048.000000    3299.000000
        75%      0.703299   338.000000   31433.000000   25147.000000    9948.000000
        max      0.968954  4212.000000  307933.000000  251540.000000  115172.000000

               Full_time_year_round   Unemployed  Unemployment_rate         Median  \
        count            173.000000   173.000000         173.000000     173.000000
        mean           19694.427746  2416.329480           0.068191   40151.445087
        std            33160.941514  4112.803148           0.030331   11470.181802
        min              111.000000     0.000000           0.000000   22000.000000
        25%             2453.000000   304.000000           0.050306   33000.000000
        50%             7413.000000   893.000000           0.067961   36000.000000
```

```
75%           16891.000000   2393.000000           0.087557   45000.000000
max          199897.000000  28169.000000           0.177226  110000.000000

              P25th          P75th   College_jobs  Non_college_jobs  \
count    173.000000     173.000000     173.000000        173.000000
mean   29501.445087   51494.219653   12322.635838      13284.497110
std     9166.005235   14906.279740   21299.868863      23789.655363
min    18500.000000   22000.000000       0.000000          0.000000
25%    24000.000000   42000.000000    1675.000000       1591.000000
50%    27000.000000   47000.000000    4390.000000       4595.000000
75%    33000.000000   60000.000000   14444.000000      11783.000000
max    95000.000000  125000.000000  151643.000000     148395.000000

       Low_wage_jobs
count     173.000000
mean     3859.017341
std      6944.998579
min         0.000000
25%       340.000000
50%      1231.000000
75%      3466.000000
max     48207.000000
```

0.0.2 Getting Familiar and Cleaning the Data Set

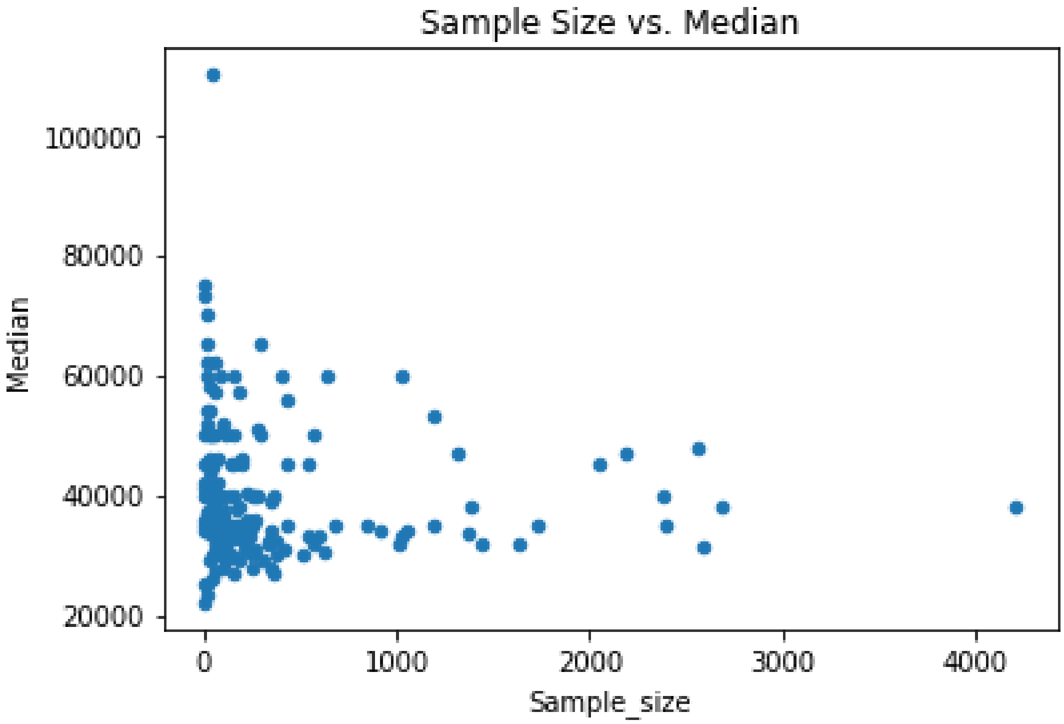
```
In [2]: raw_data_count = recent_grads.shape[0]
        recent_grads = recent_grads.dropna()
        clear_data_count = recent_grads.shape[0]
        print(raw_data_count)
        print(clear_data_count)
```

```
173
172
```

0.0.3 Generating Scatter Plots

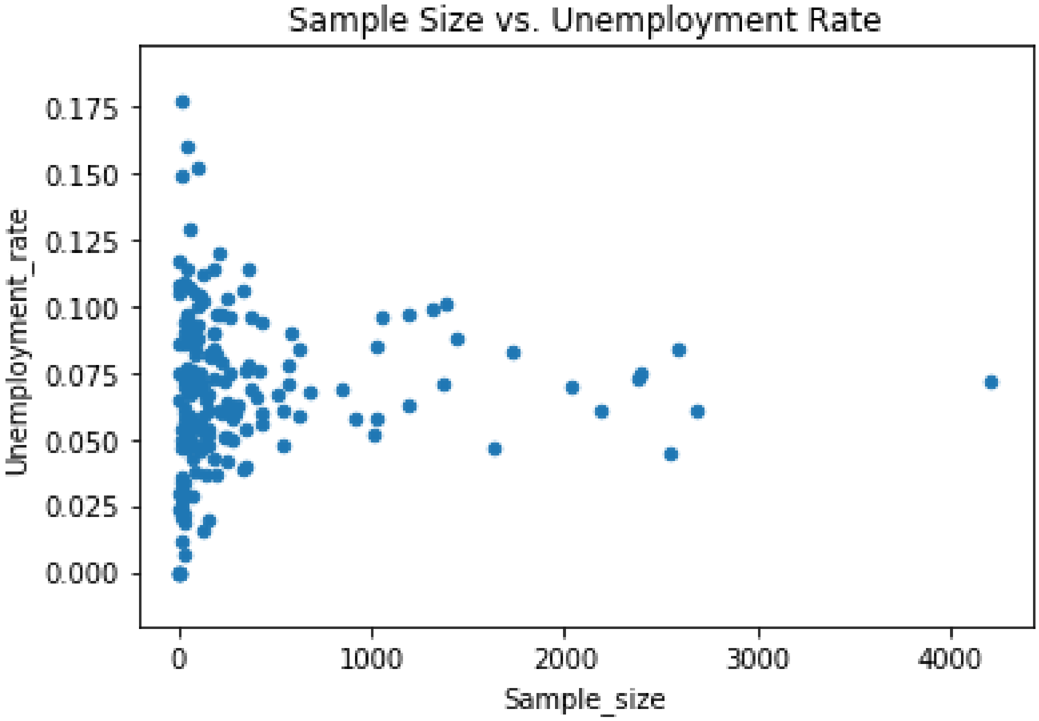
```
In [3]: recent_grads.plot(x='Sample_size', y='Median', kind='scatter', title='Sample Size vs.
```

```
Out[3]: <matplotlib.axes._subplots.AxesSubplot at 0x9503400>
```



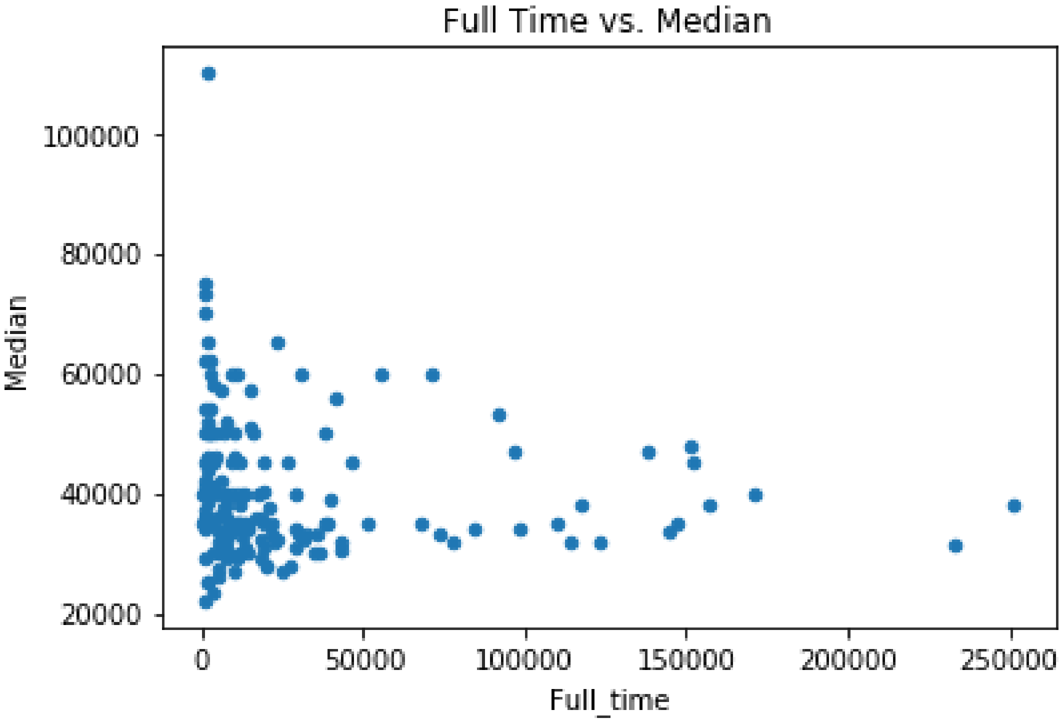
```
In [4]: recent_grads.plot(x='Sample_size', y='Unemployment_rate', kind='scatter', title='Sample
```

```
Out[4]: <matplotlib.axes._subplots.AxesSubplot at 0x5729128>
```



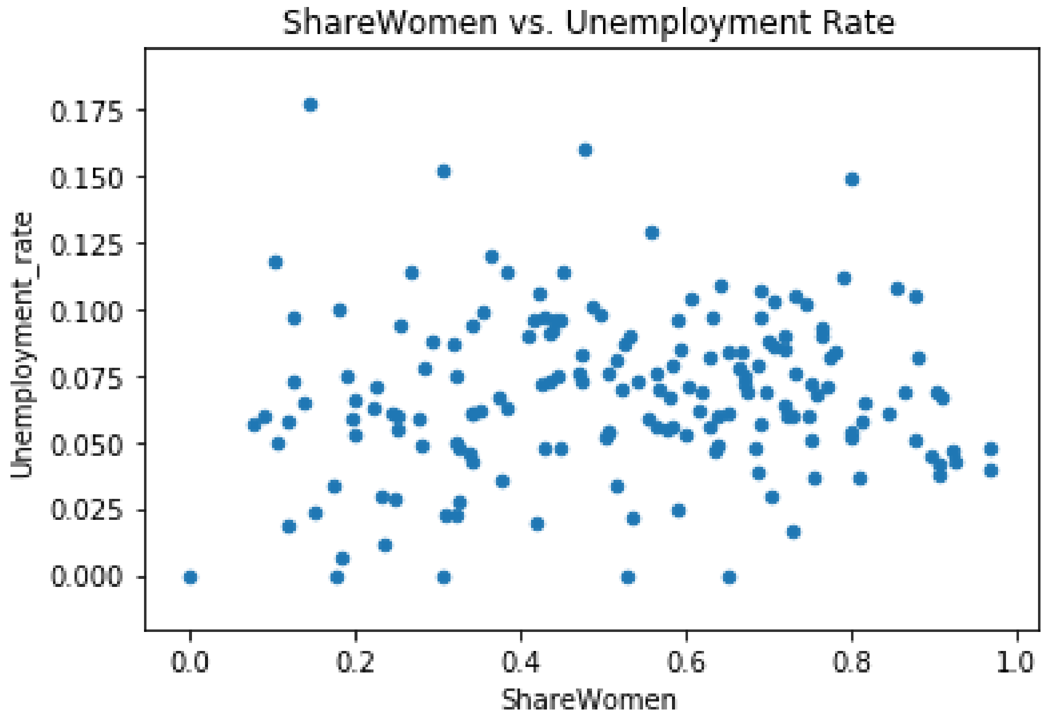
```
In [5]: recent_grads.plot(x='Full_time', y='Median', kind='scatter', title='Full Time vs. Media
```

```
Out[5]: <matplotlib.axes._subplots.AxesSubplot at 0x9506908>
```

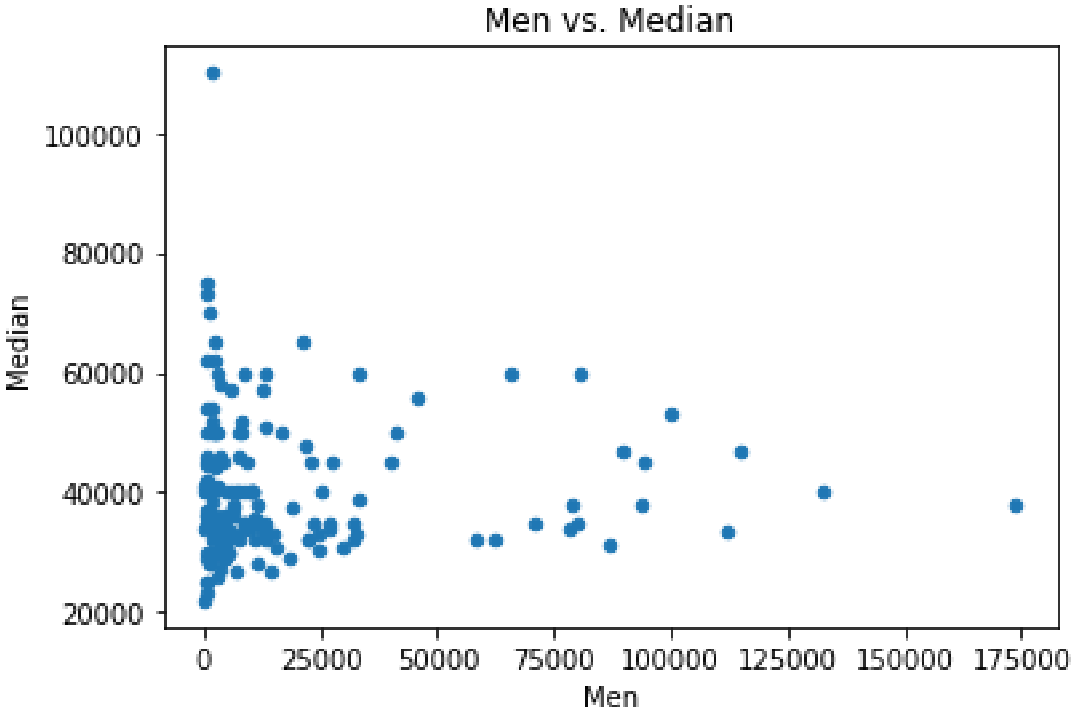


```
In [6]: recent_grads.plot(x='ShareWomen', y='Unemployment_rate', kind='scatter', title='ShareWo
```

```
Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x97928d0>
```

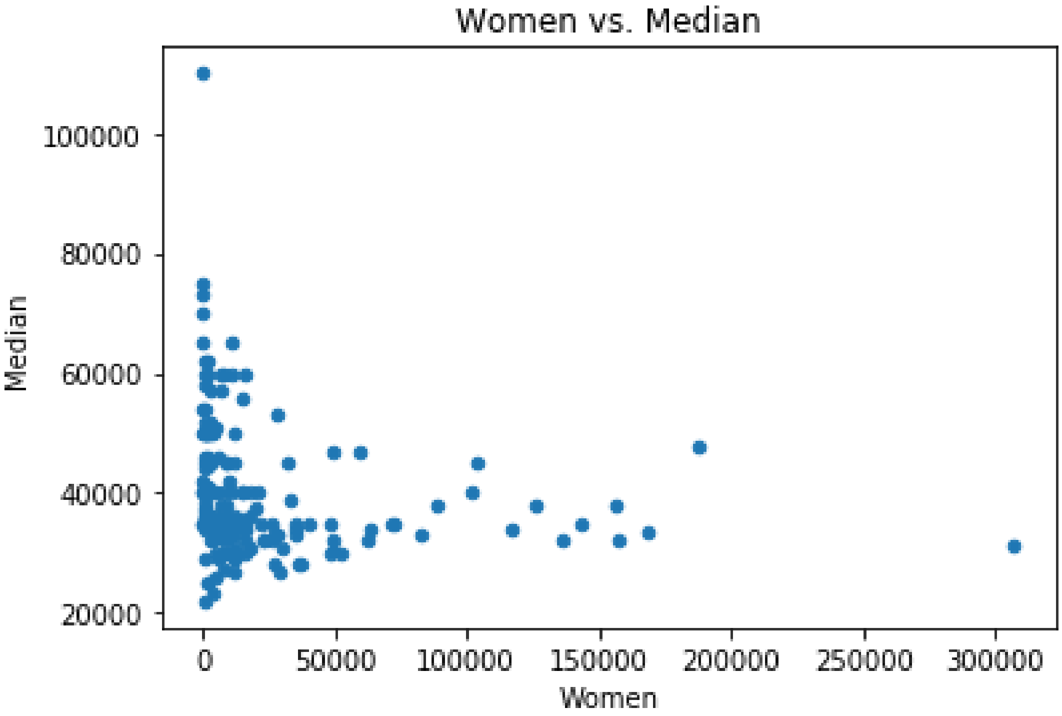


```
In [7]: recent_grads.plot(x='Men', y='Median', kind='scatter', title='Men vs. Median')

Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x97b71d0>
```



```
In [8]: recent_grads.plot(x='Women', y='Median', kind='scatter', title='Women vs. Median')
```

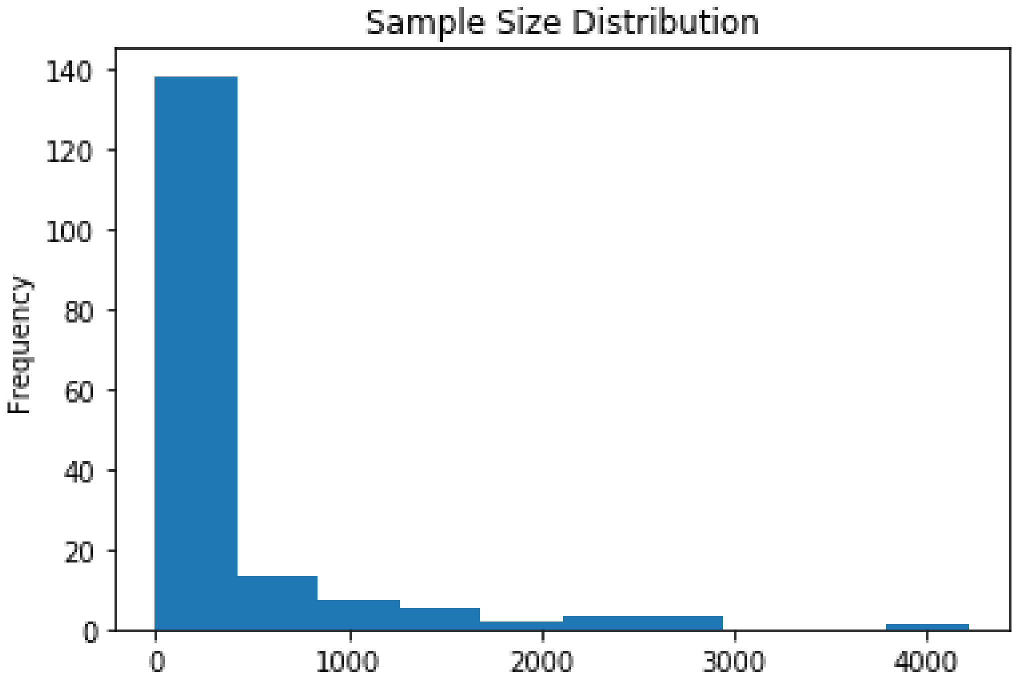
```
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x983bf28>
```



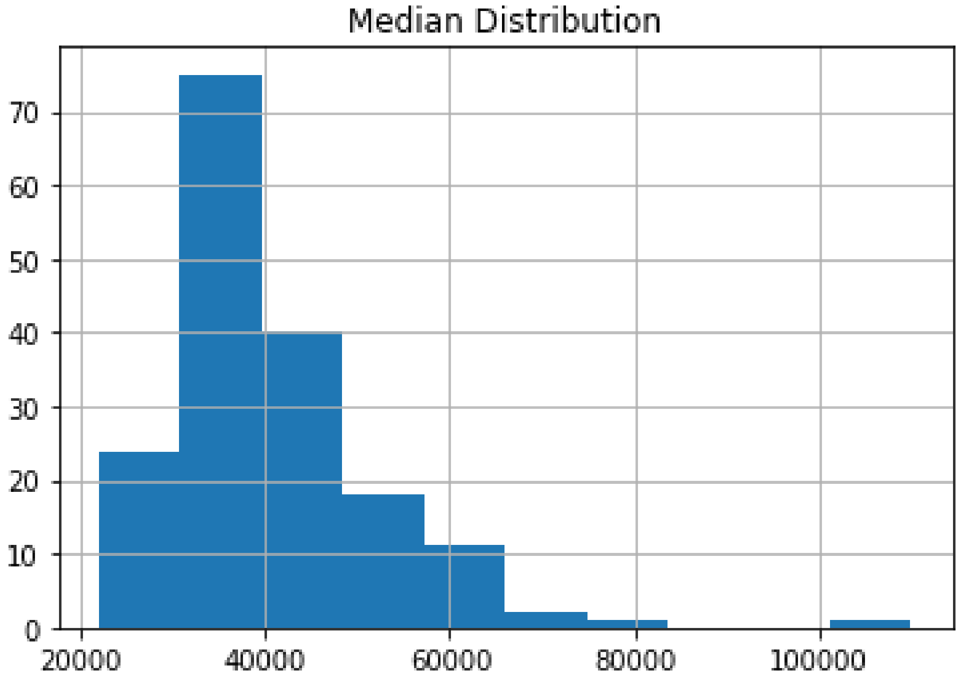
0.0.4 Generating Histograms

```
In [9]: # Generate histogram to explore distributions
        recent_grads["Sample_size"].plot(kind='hist', title="Sample Size Distribution")
```

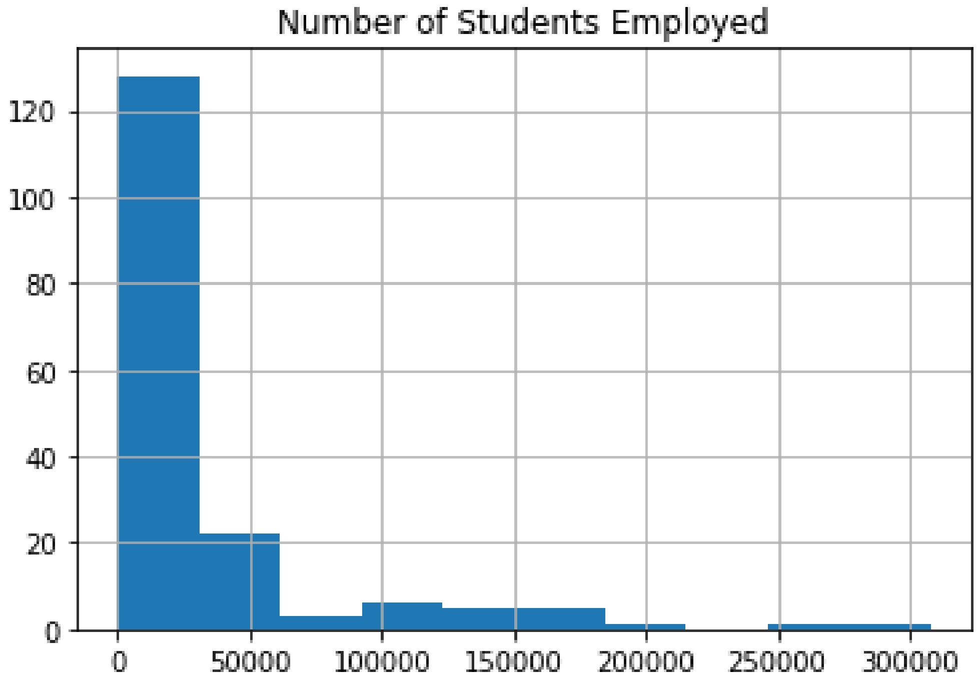
```
Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0xa8414e0>
```



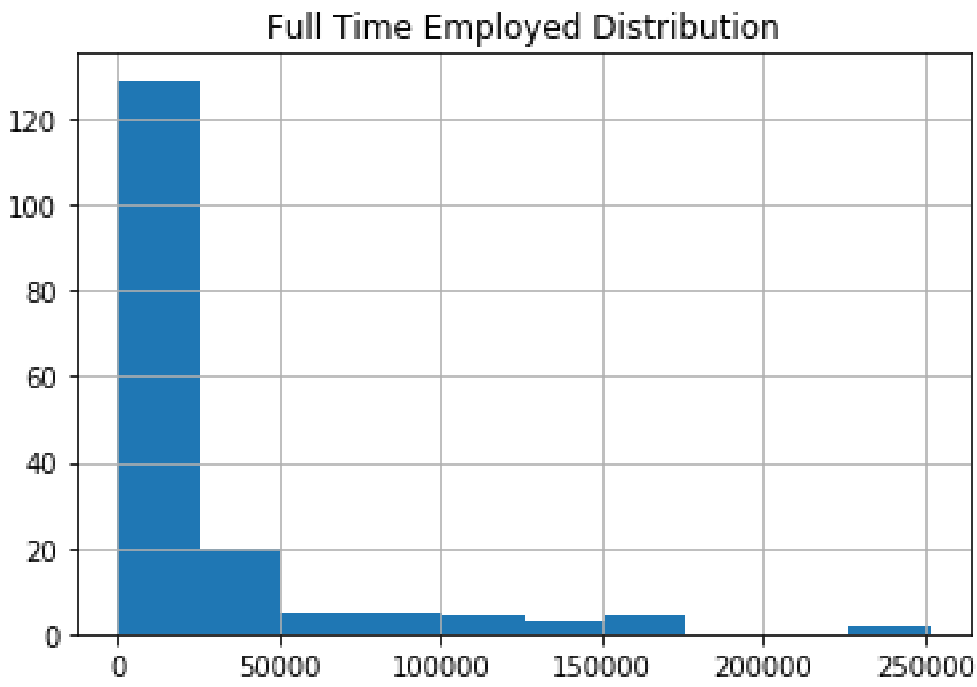
```
In [10]: recent_grads["Median"].hist()
         plt.title("Median Distribution")

Out[10]: Text(0.5,1,'Median Distribution')
```



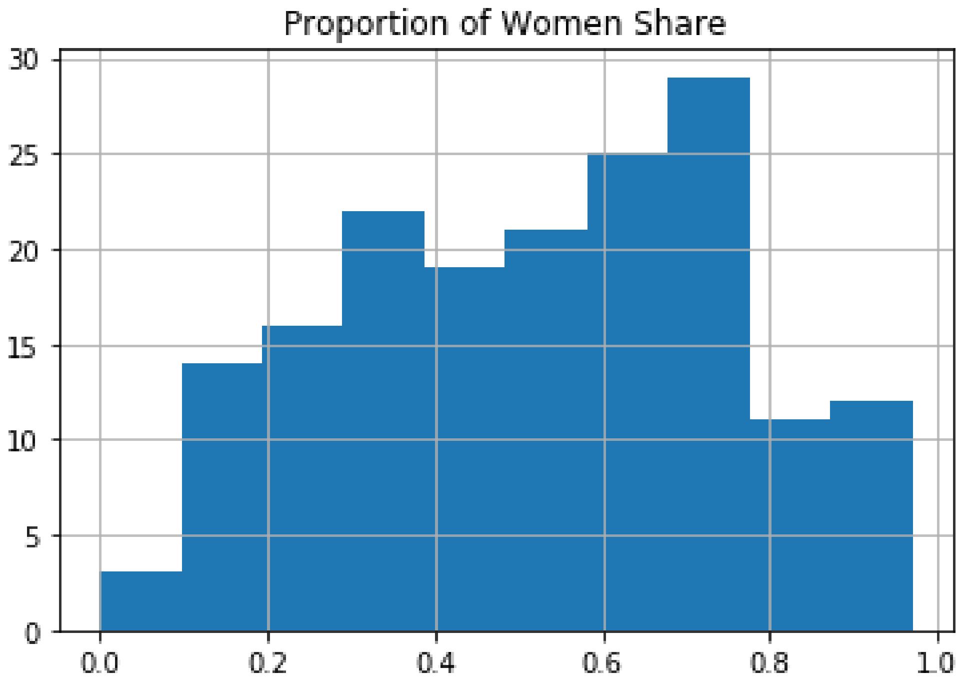
```
In [11]: recent_grads["Employed"].hist()
         plt.title("Number of Students Employed")

Out[11]: Text(0.5,1,'Number of Students Employed')
```



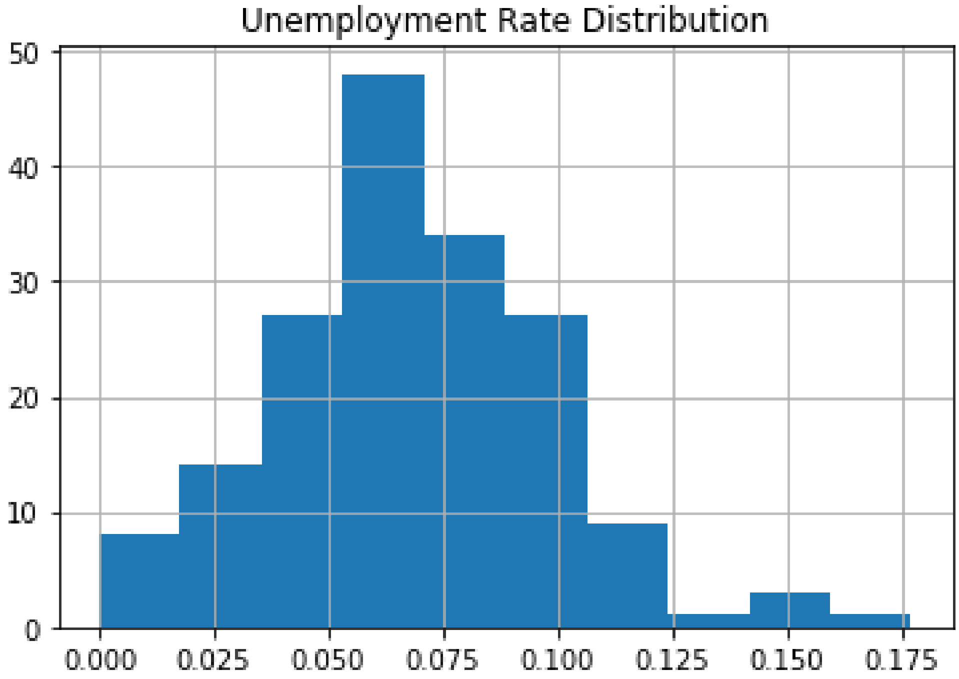
```
In [12]: recent_grads["Full_time"].hist()
         plt.title("Full Time Employed Distribution")

Out[12]: Text(0.5,1,'Full Time Employed Distribution')
```



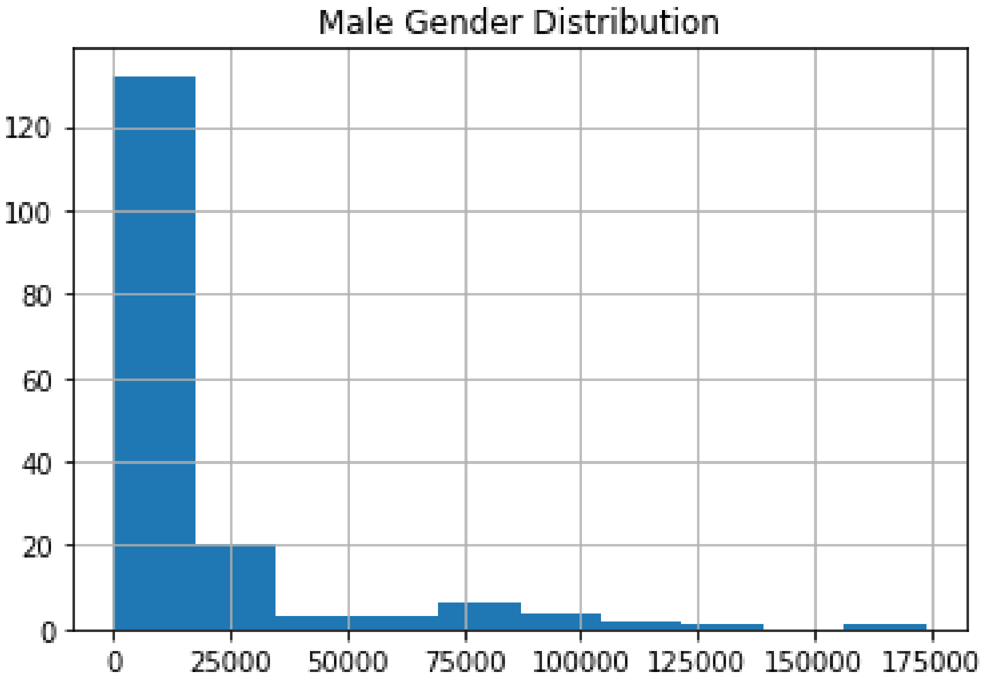
```
In [13]: recent_grads["ShareWomen"].hist()
         plt.title("Proportion of Women Share")

Out[13]: Text(0.5,1,'Proportion of Women Share')
```



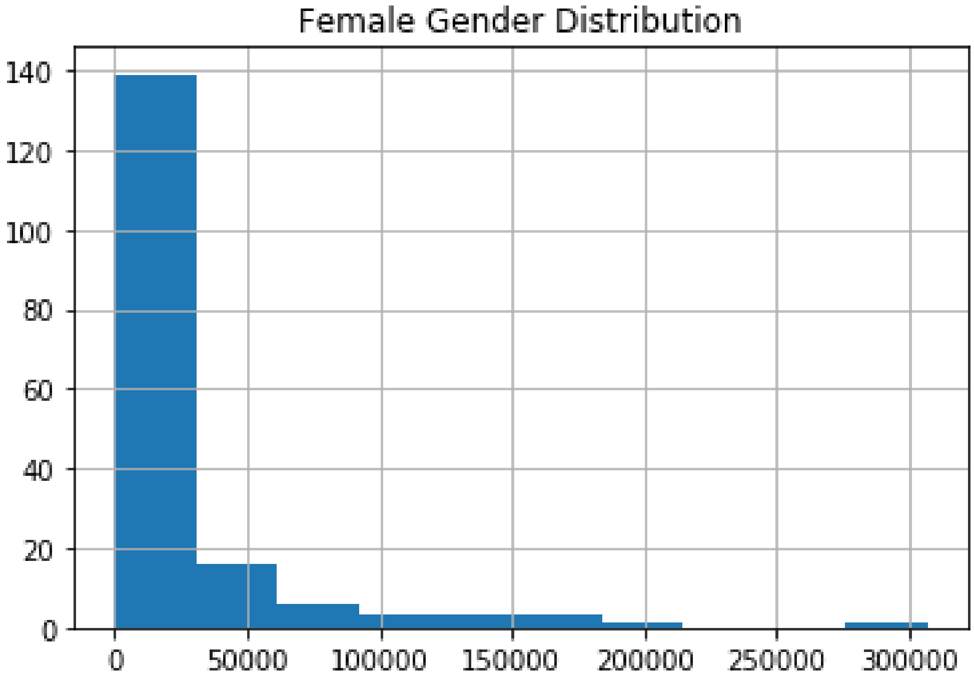
```
In [14]: recent_grads["Unemployment_rate"].hist()
         plt.title("Unemployment Rate Distribution")

Out[14]: Text(0.5,1,'Unemployment Rate Distribution')
```



```
In [15]: recent_grads["Men"].hist()
         plt.title("Male Gender Distribution")

Out[15]: Text(0.5,1,'Male Gender Distribution')
```



```
In [16]: recent_grads["Women"].hist()
         plt.title("Female Gender Distribution")

Out[16]: Text(0.5,1,'Female Gender Distribution')
```

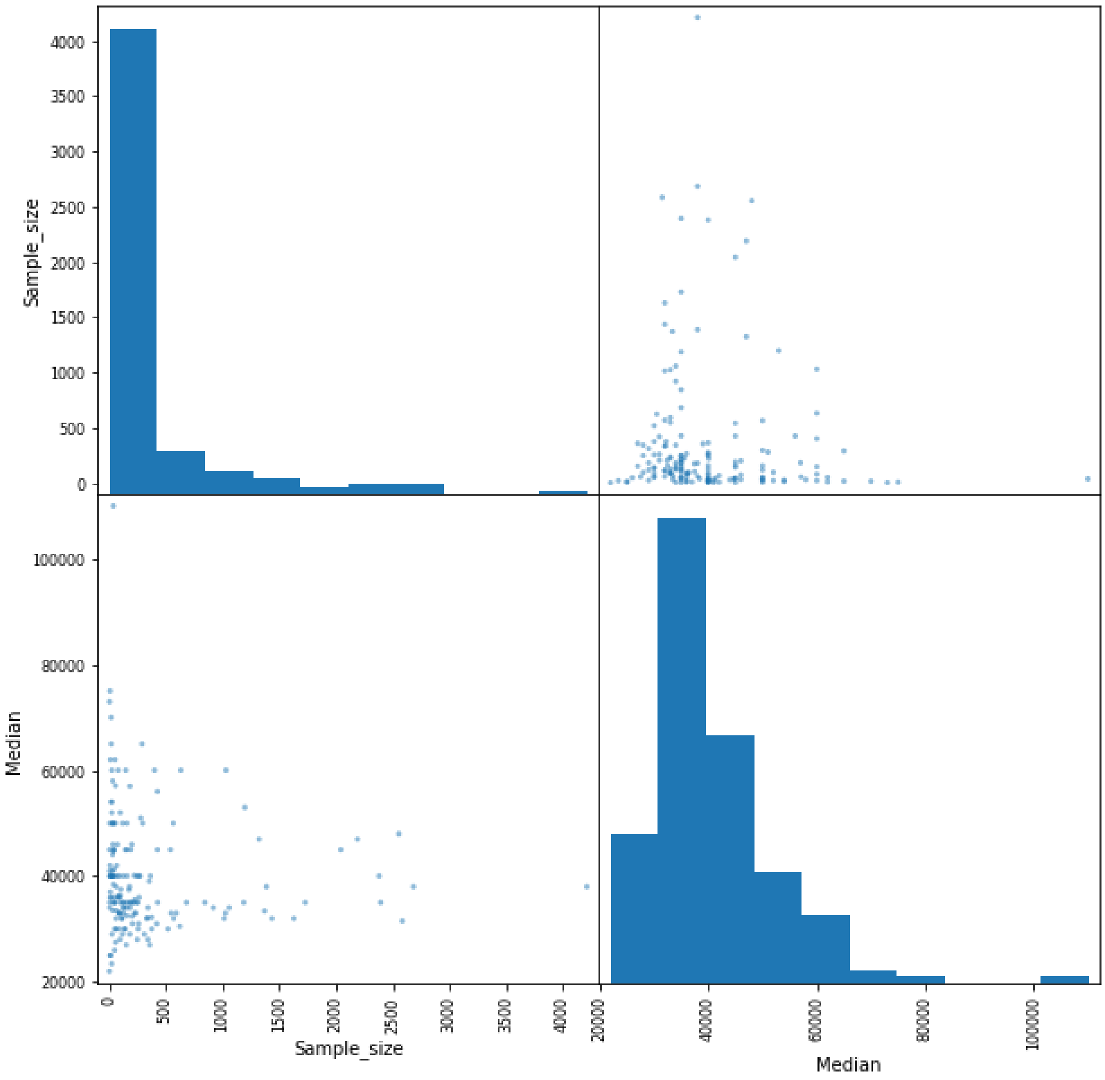


Prepare Scatter Matrix Plot for Data Statistics & Unemployment Rate

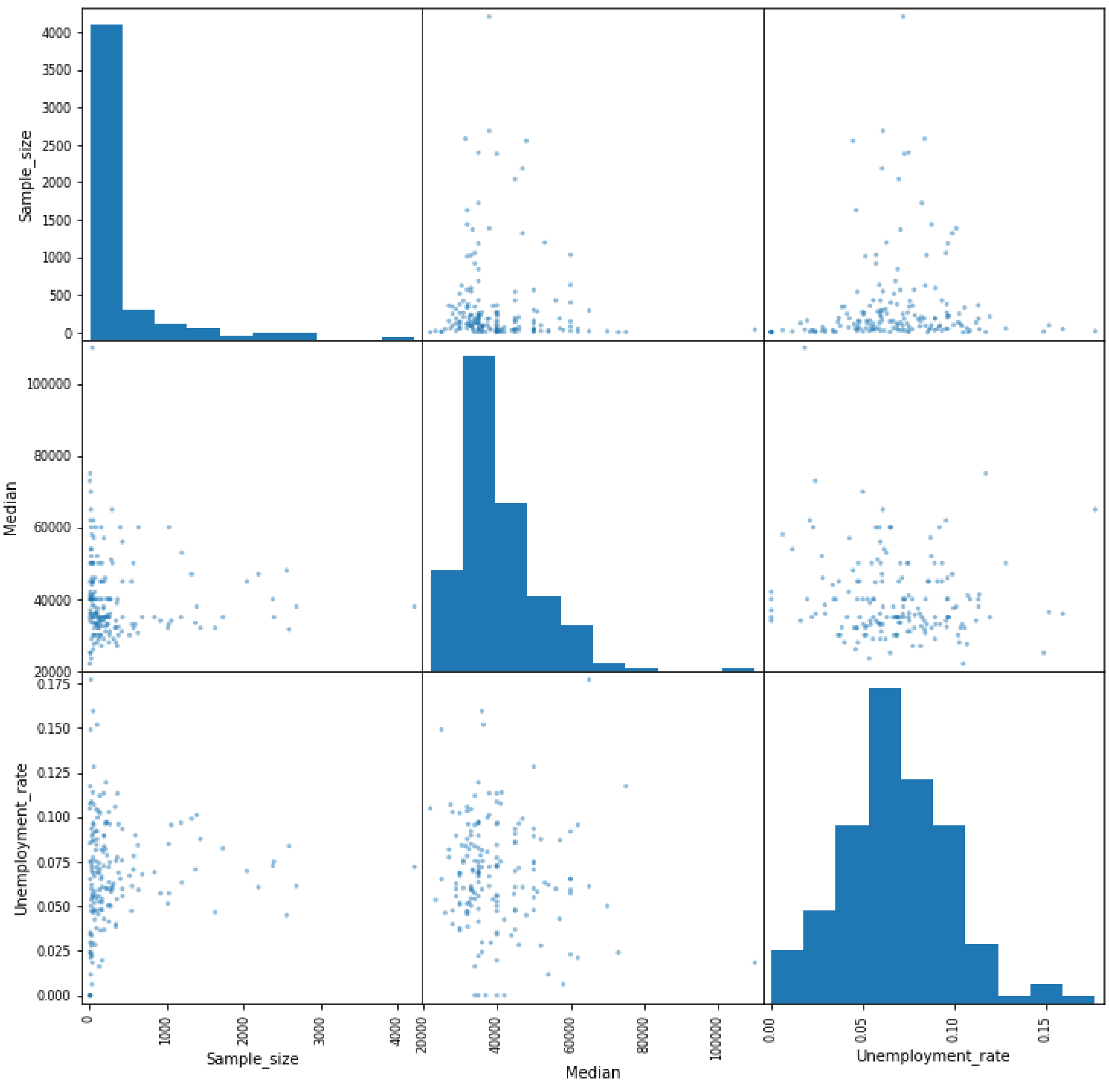
```
In [17]: # Working with Pandas' Scatter Matrix Plot
         from pandas.tools.plotting import scatter_matrix
         scatter_matrix(recent_grads[['Sample_size', 'Median']], figsize=(10,10))
```

```
C:\Users\Yogi_Ashwast\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: FutureWarning: 'panc
  This is separate from the ipykernel package so we can avoid doing imports until
```

```
Out[17]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000000000A9BC940>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000ACD4160>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x000000000AD10160>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AD4A160>]],
               dtype=object)
```

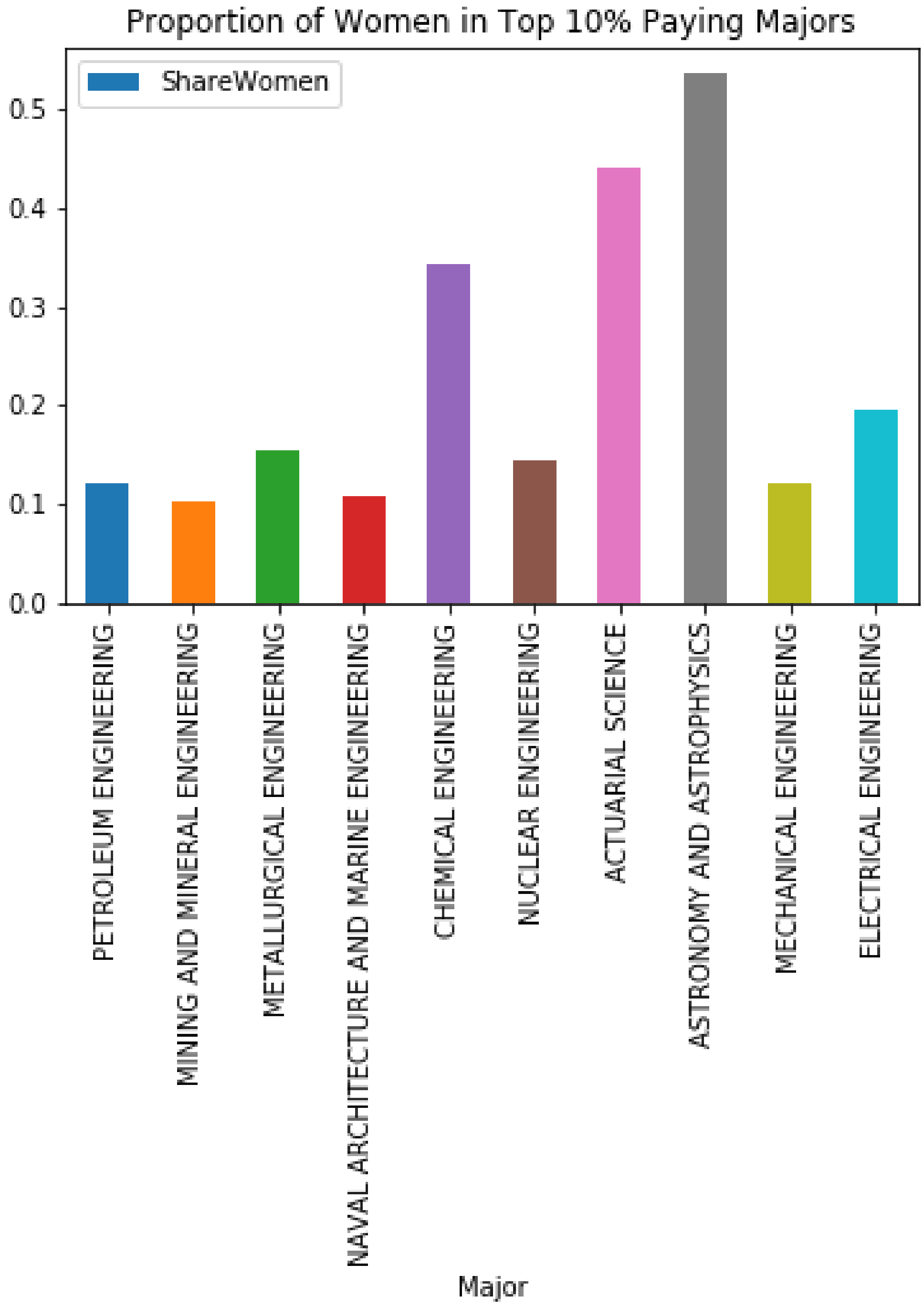


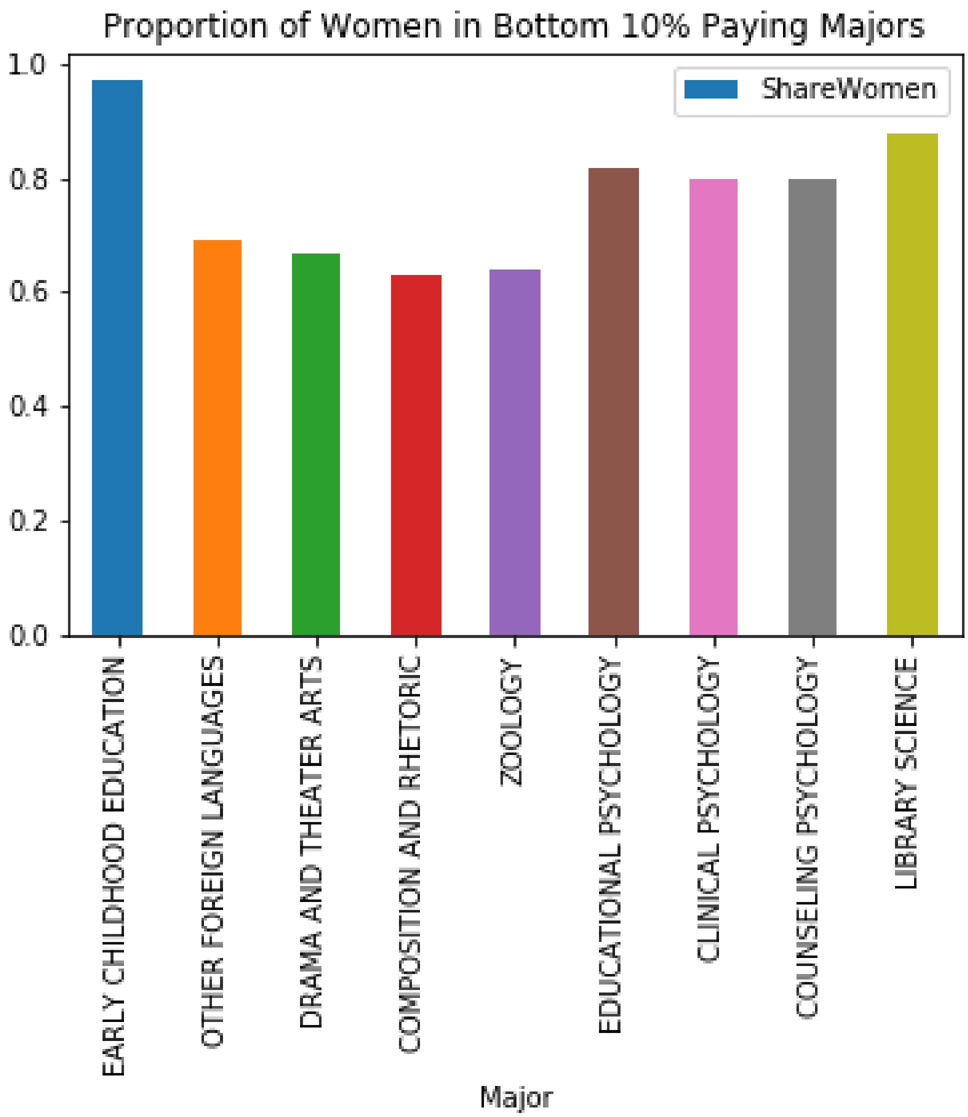
```
In [18]: scatter_matrix(recent_grads[['Sample_size','Median','Unemployment_rate']], figsize=(1

C:\Users\Yogi_Ashwast\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: FutureWarning: 'pan
  """Entry point for launching an IPython kernel.


Out[18]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000000000ADCA860>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000B070400>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AE69470>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x000000000AE87748>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AEDD470>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AEDD4A8>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x000000000AF3EE80>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AF84390>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x000000000AFBD390>]],
               dtype=object)
```



0.0.5 Plot Women Proportions in Different Majors

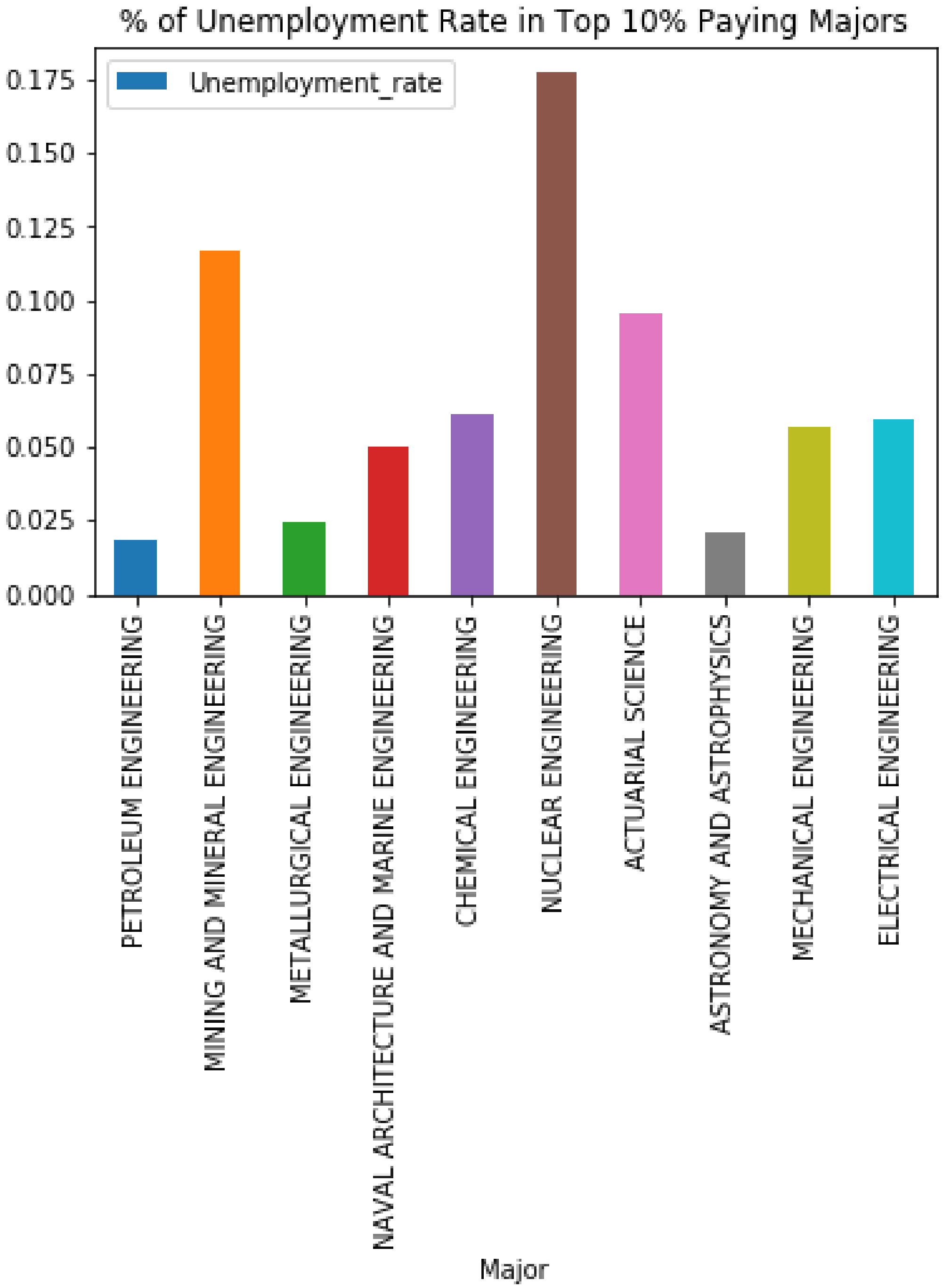
```
In [19]: # Percent of Women from "ShareWomen" corresponds to top & bottom 10% paying majors
         recent_grads[:10].plot.bar(x='Major', y='ShareWomen')
         plt.title("Proportion of Women in Top 10% Paying Majors")
         plt.show()

         recent_grads[163:].plot.bar(x='Major', y='ShareWomen')
         plt.title("Proportion of Women in Bottom 10% Paying Majors")
         plt.show()
```

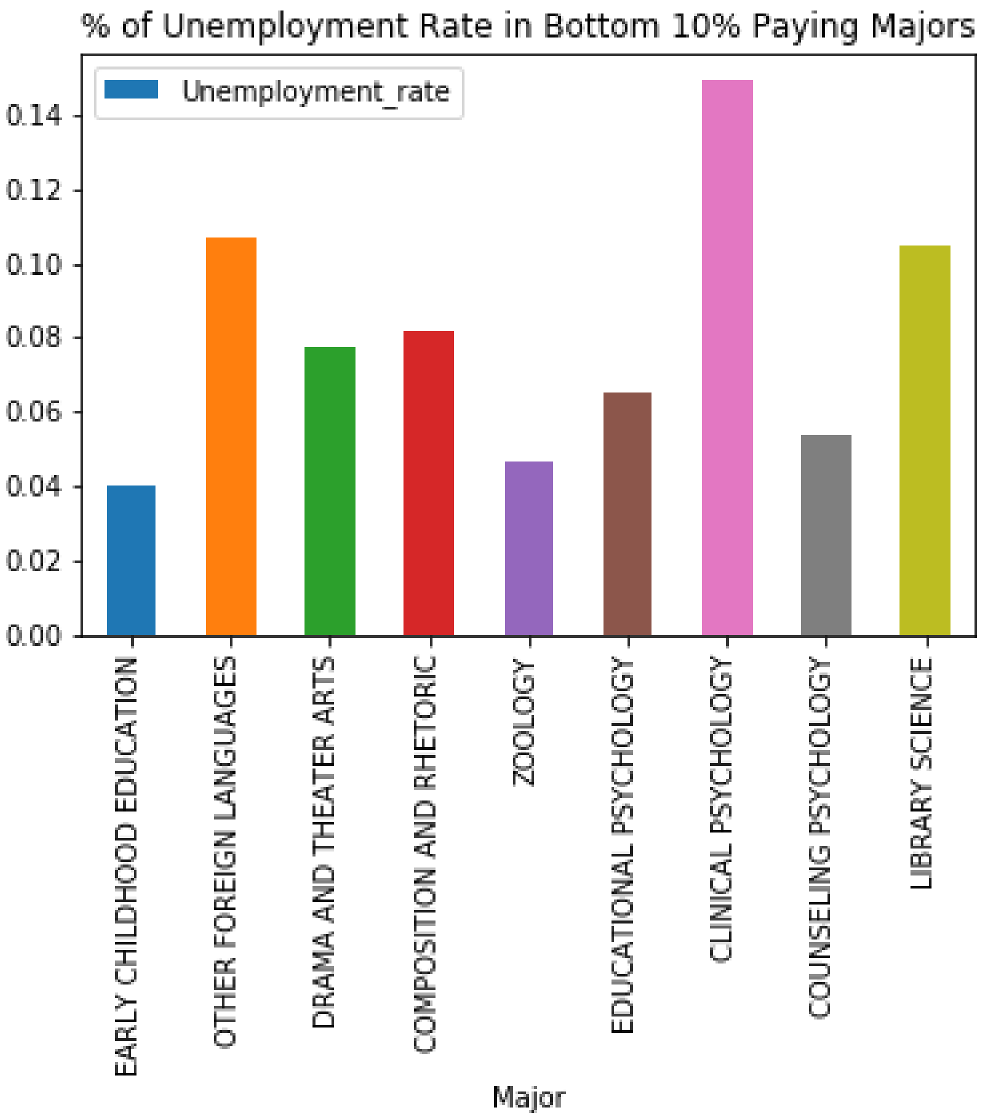

Proportion of Women in Top 10% Paying Majors
ShareWomen
0.5
0.4
0.3
0.2
0.1
0.0
PETROLEUM ENGINEERING
MINING AND MINERAL ENGINEERING
METALLURGICAL ENGINEERING
NAVAL ARCHITECTURE AND MARINE ENGINEERING
CHEMICAL ENGINEERING
NUCLEAR ENGINEERING
ACTUARIAL SCIENCE
ASTRONOMY AND ASTROPHYSICS
MECHANICAL ENGINEERING
ELECTRICAL ENGINEERING
Major



0.0.6 Plot Unemployment Rate in Different Majors

```
In [20]: # Percent of Unemployment Rate corresponds to top & bottom 10% paying majors
         recent_grads[:10].plot.bar(x='Major', y='Unemployment_rate')
         plt.title("% of Unemployment Rate in Top 10% Paying Majors")
         plt.show()

         recent_grads[163:].plot.bar(x='Major', y='Unemployment_rate')
         plt.title("% of Unemployment Rate in Bottom 10% Paying Majors")
         plt.show()
```


% of Unemployment Rate in Top 10% Paying Majors
Unemployment_rate
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
PETROLEUM ENGINEERING
MINING AND MINERAL ENGINEERING
METALLURGICAL ENGINEERING
NAVAL ARCHITECTURE AND MARINE ENGINEERING
CHEMICAL ENGINEERING
NUCLEAR ENGINEERING
ACTUARIAL SCIENCE
ASTRONOMY AND ASTROPHYSICS
MECHANICAL ENGINEERING
ELECTRICAL ENGINEERING
Major



0.1 Find Out Gender Participation for Major Categories

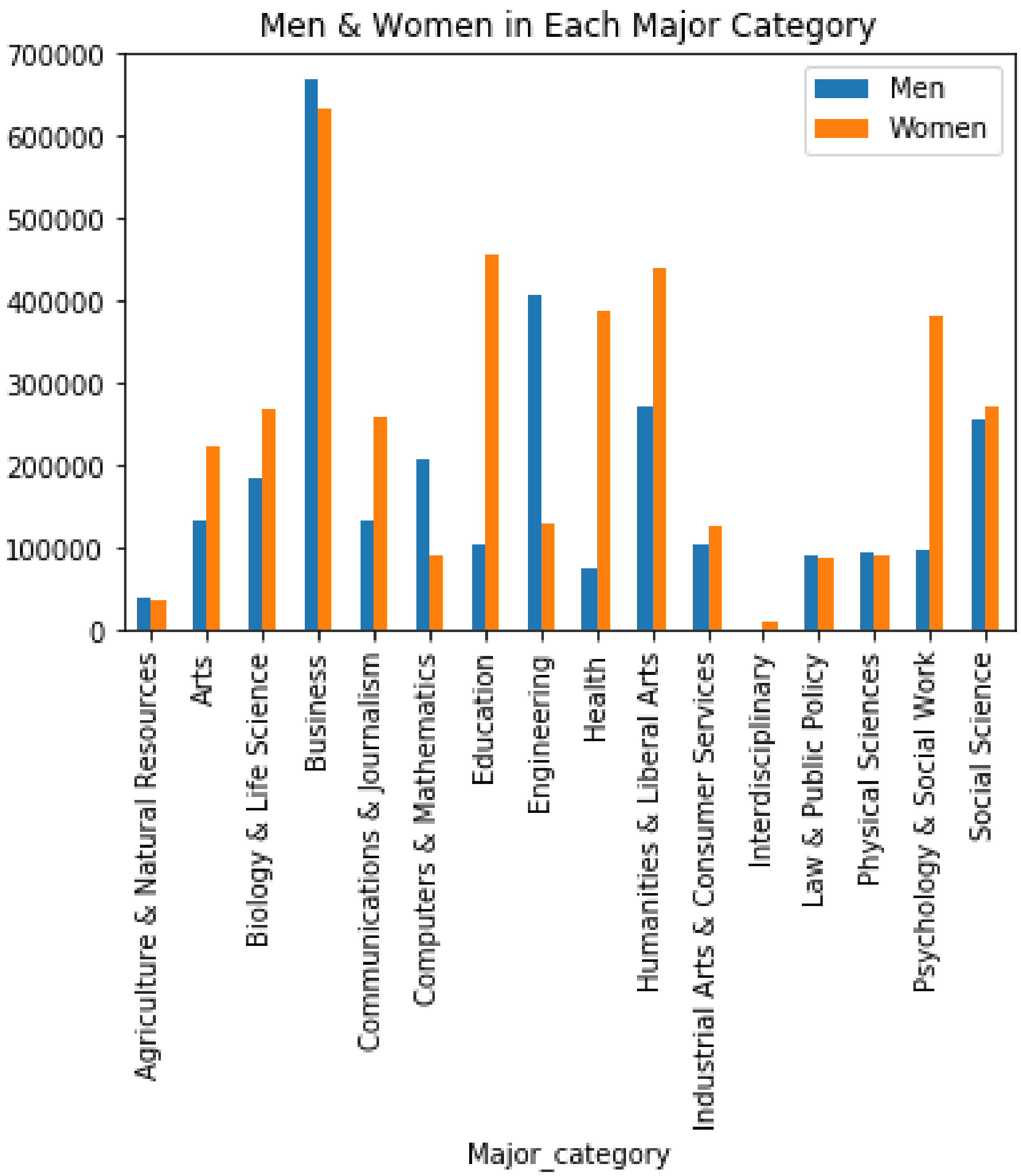
```
In [21]: from pandas import *

         cols = ["Major_category", "Men", "Women"]
         gender_major = recent_grads[cols]
         gender_major_cat = gender_major.groupby("Major_category", as_index=True).sum()
         print(gender_major_cat)
```

```
                                          Men     Women
Major_category
Agriculture & Natural Resources       40357.0   35263.0
Arts                                 134390.0  222740.0
```

```
Biology & Life Science                 184919.0  268943.0
Business                               667852.0  634524.0
Communications & Journalism            131921.0  260680.0
Computers & Mathematics                208725.0   90283.0
Education                              103526.0  455603.0
Engineering                            408307.0  129276.0
Health                                  75517.0  387713.0
Humanities & Liberal Arts              272846.0  440622.0
Industrial Arts & Consumer Services    103781.0  126011.0
Interdisciplinary                        2817.0    9479.0
Law & Public Policy                     91129.0   87978.0
Physical Sciences                       95390.0   90089.0
Psychology & Social Work                98115.0  382892.0
Social Science                         256834.0  273132.0
```

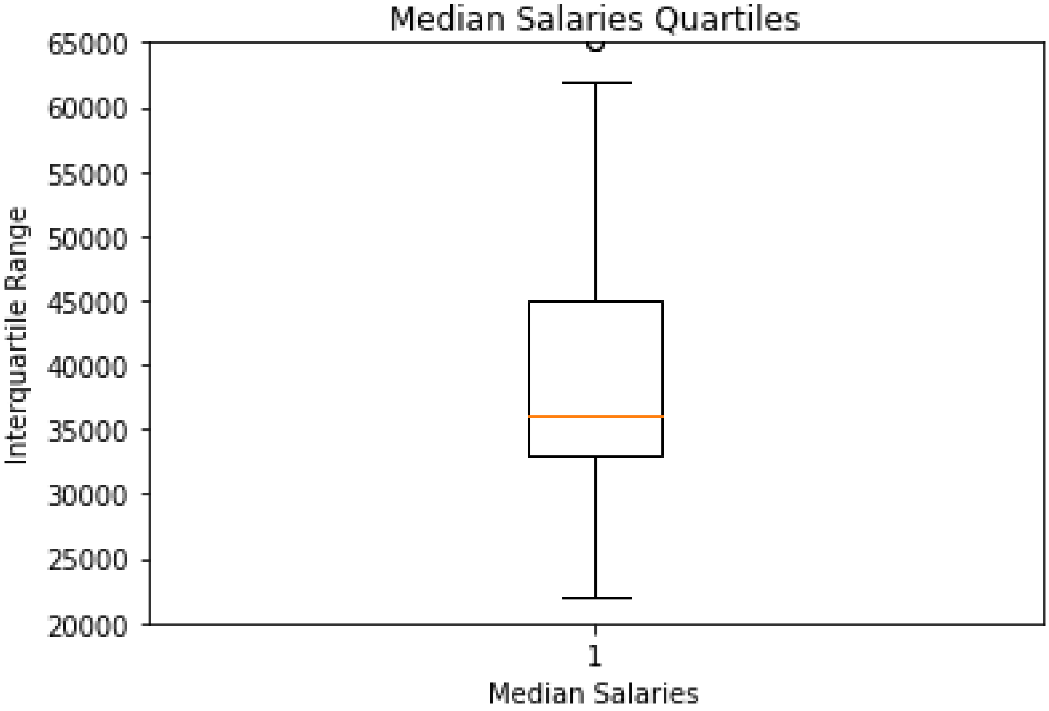
0.2 Plot Number of Men & Women in Each Major Category

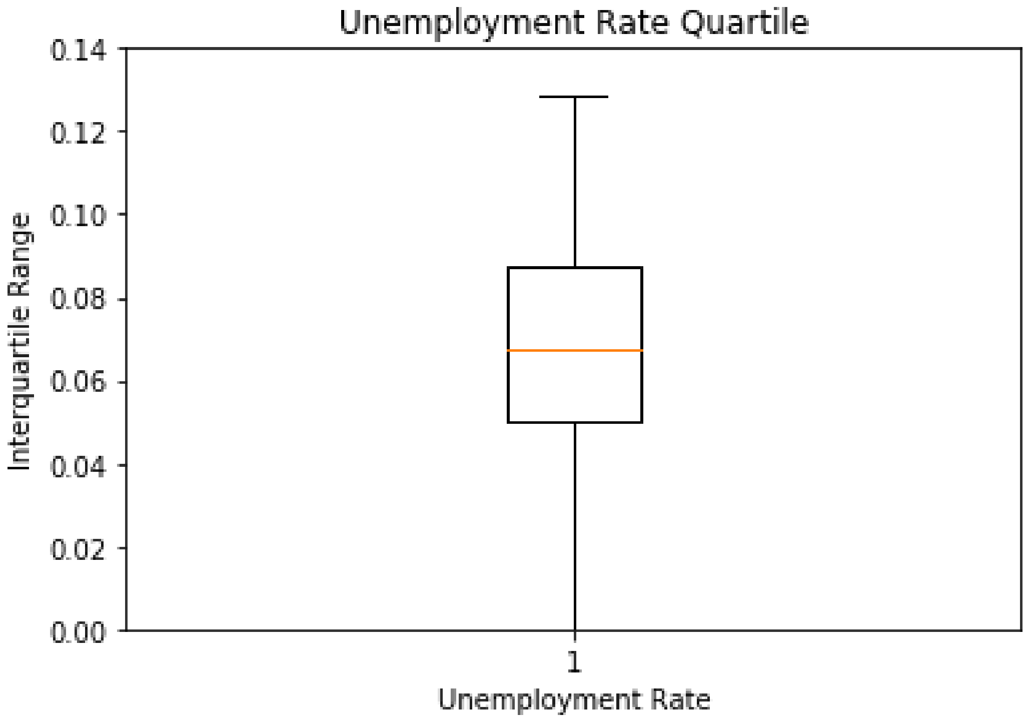
```
In [22]: gender_table = pivot_table(gender_major, values=['Men', 'Women'], index='Major_catego
         gender_table.plot(kind='bar')
         plt.title("Men & Women in Each Major Category")
         plt.show()
         print()
```



0.2.1 Plot Showing Distribution of Median Salaries & Unemployment Rate

```
In [23]: # GEt a Box Plot for Median Salaries
         fig, ax = plt.subplots()
         ax.boxplot(recent_grads['Median'].values)
         ax.set_xlabel("Median Salaries")
         ax.set_ylabel("Interquartile Range")
         ax.set_title("Median Salaries Quartiles")
         ax.set_ylim(20000, 65000)
         plt.show()
         # Get a Box Plot for Unemployment Rate
         fig, ax = plt.subplots()
         ax.boxplot(recent_grads['Unemployment_rate'].values)
         ax.set_xlabel("Unemployment Rate")
         ax.set_ylabel("Interquartile Range")
         ax.set_title("Unemployment Rate Quartile")
         ax.set_ylim(0, 0.14)
         plt.show()
```





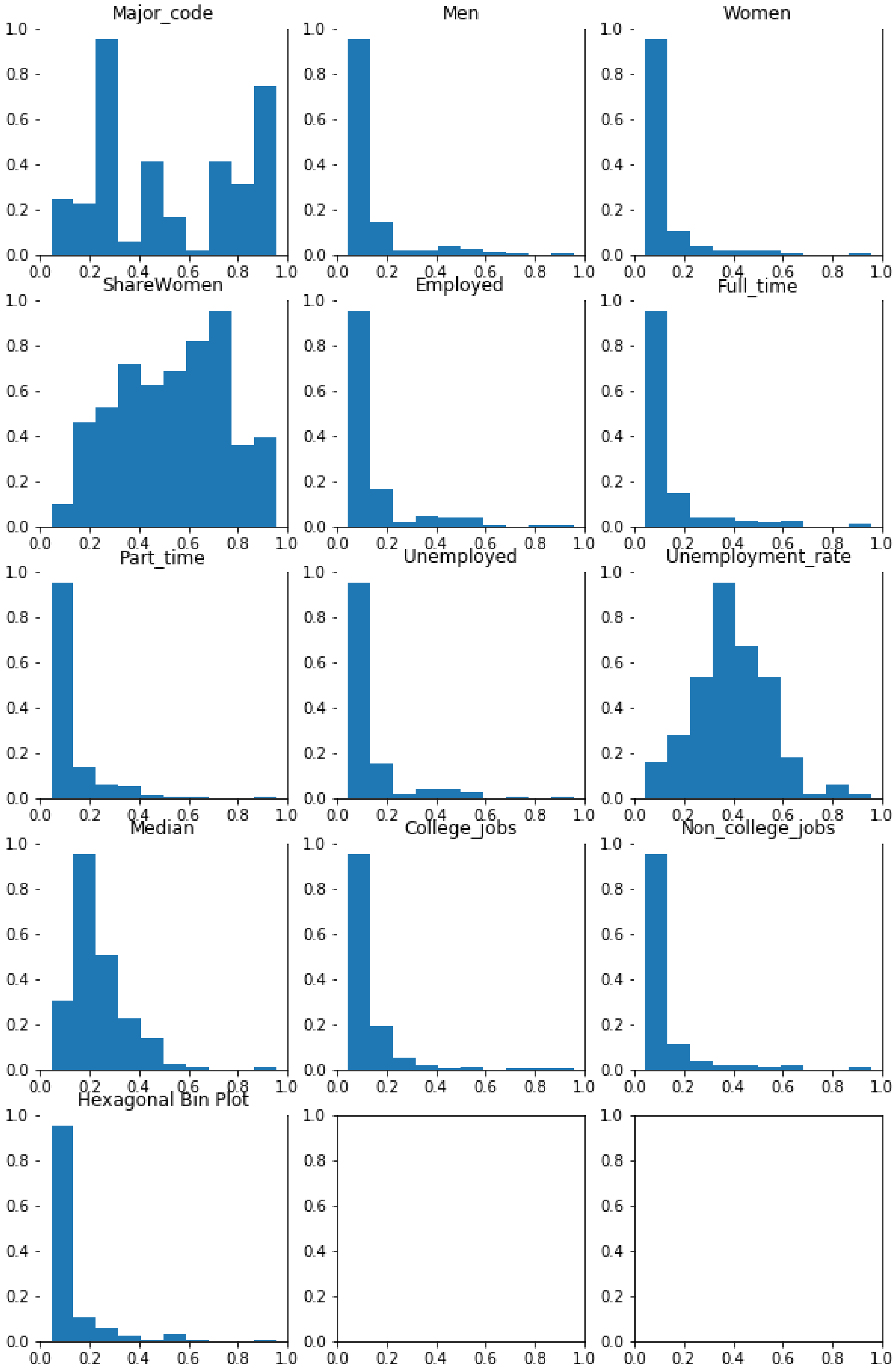
0.2.2 Generate Hexagonal Bin Plot to Visualize Data Density

```
In [24]: # Segregating dataframe columns that carry numeric values
         numerical_recent_grads = recent_grads.select_dtypes(include=['int64', 'float64'])
         print(numerical_recent_grads.dtypes)
         print(numerical_recent_grads.columns)
```

```
Rank                      int64
Major_code                int64
Total                   float64
Men                     float64
Women                   float64
ShareWomen              float64
Sample_size               int64
Employed                  int64
Full_time                 int64
Part_time                 int64
Full_time_year_round      int64
Unemployed                int64
Unemployment_rate       float64
Median                    int64
P25th                     int64
P75th                     int64
College_jobs              int64
```

```
Non_college_jobs          int64
Low_wage_jobs             int64
dtype: object
Index(['Rank', 'Major_code', 'Total', 'Men', 'Women', 'ShareWomen',
       'Sample_size', 'Employed', 'Full_time', 'Part_time',
       'Full_time_year_round', 'Unemployed', 'Unemployment_rate', 'Median',
       'P25th', 'P75th', 'College_jobs', 'Non_college_jobs', 'Low_wage_jobs'],
      dtype='object')
```

```
In [25]: cols = ['Major_code', 'Men', 'Women', 'ShareWomen', 'Employed', 'Full_time', 'Part_ti
                 'Unemployment_rate', 'Median', 'College_jobs', 'Non_college_jobs', 'Low_wage_

         if (len(cols)%3 != 0):
             row_num = len(cols)//3 + 1
         else:
             row_num = len(cols)//3

         fig, ax = plt.subplots(row_num, 3, figsize=(10,16))

         j=0
         for i in range(1, len(cols)*3, 3):
             sbp = int((i-1)/3)
             ax = fig.add_subplot(row_num, 3, sbp+1)
             X = (numerical_recent_grads[str(cols[j])].values)
             ax.hist(X)

             ax.set_title(cols[j])
             for keys, spine in ax.spines.items():
                 spine.set_visible(False)
             ax.tick_params(axis='both', left='off', top='off', right='off', \
                            bottom='off', labelleft='off', labeltop='off', \
                            labelright='off', labelbottom='off')
             j += 1

         plt.title("Hexagonal Bin Plot")
         plt.show()
```


Major_code
Men
Women
ShareWomen
Employed
Full_time
Part_time
Unemployed
Unemployment_rate
Median
College_jobs
Non_college_jobs
Hexagonal Bin Plot
0.0
0.2
0.4
0.6
0.8
1.0