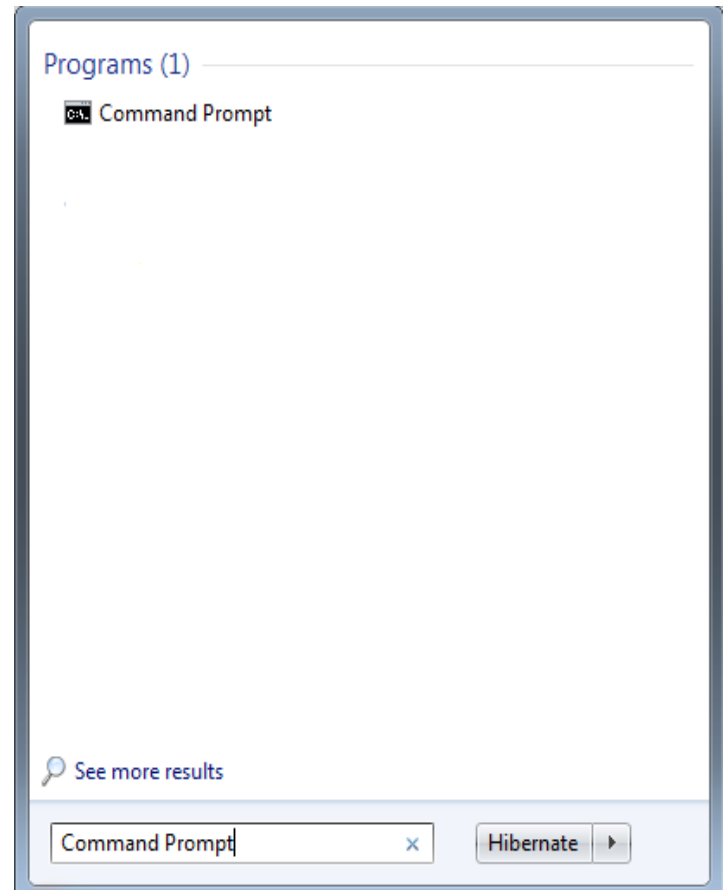
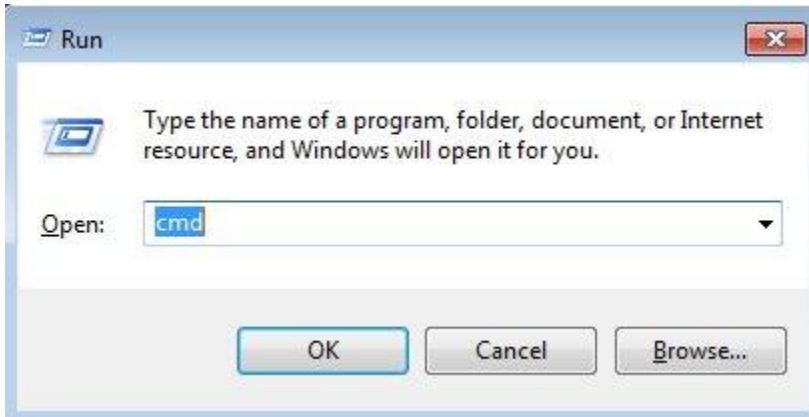


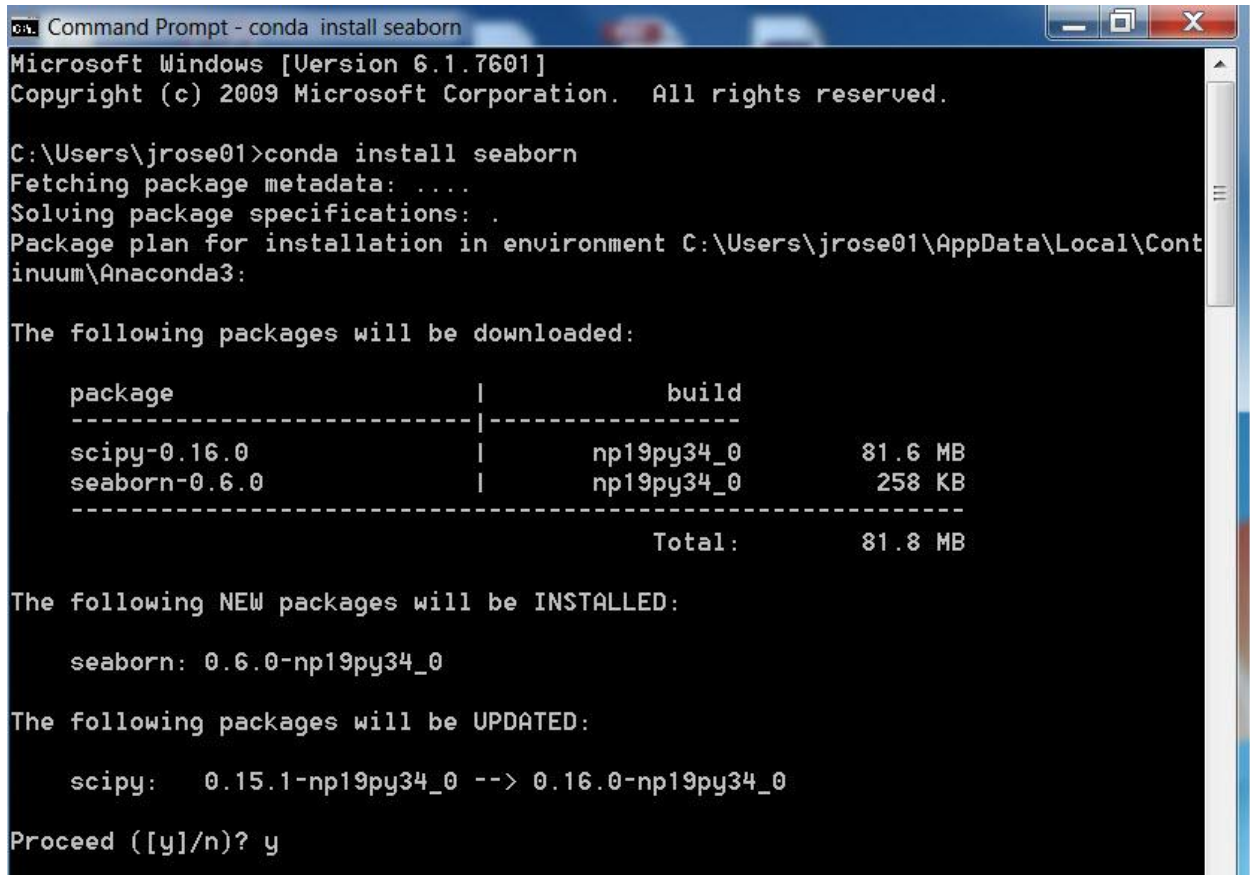
Using Python Seaborn package for graphing.

First, you need to download the Seaborn package because it is not one of the packages included with Anaconda.

1. Open a command prompt
2. On a Mac, type “Terminal” into Finder or on Windows type “Command Prompt” into the Start Menu search box and click the program that shows up.



2. In the Command Prompt window type “conda install seaborn”, then type “y” after the Proceed ([y]/n)? line.



```
Command Prompt - conda install seaborn
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\jrose01>conda install seaborn
Fetching package metadata: ....
Solving package specifications: .
Package plan for installation in environment C:\Users\jrose01\AppData\Local\Continuum\Anaconda3:

The following packages will be downloaded:

package | build | size
-----|-----|-----
scipy-0.16.0 | np19py34_0 | 81.6 MB
seaborn-0.6.0 | np19py34_0 | 258 KB
-----|-----|-----
Total: | | 81.8 MB

The following NEW packages will be INSTALLED:

seaborn: 0.6.0-np19py34_0

The following packages will be UPDATED:

scipy: 0.15.1-np19py34_0 --> 0.16.0-np19py34_0

Proceed ([y]/n)? y
```

As with any package, you will need to import the seaborn package before you can use it.

```
import seaborn
```

You will also need to import the matplotlib.pyplot library because the seaborn package is dependent on this package to create graphs. Because the name of this package is so long, We'll give it a nickname “plt”, which can be used in place of the full package name when we write code using this package.

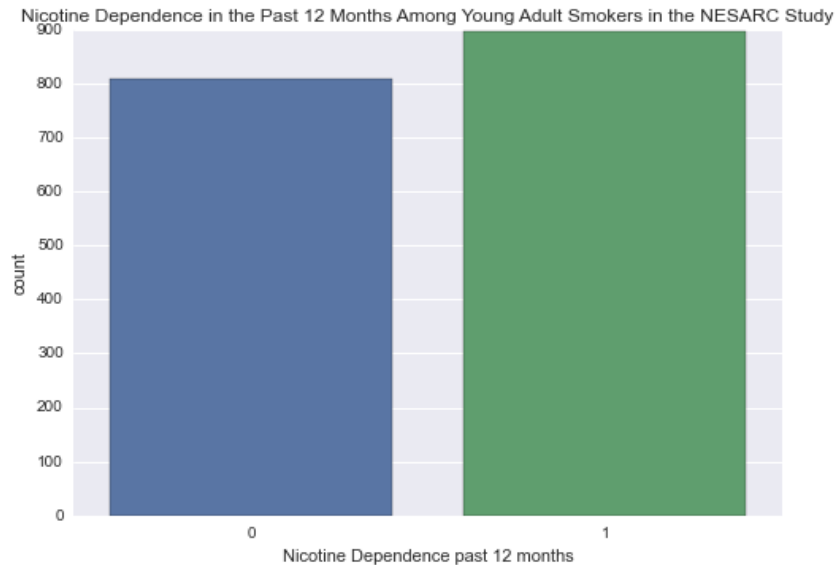
```
import matplotlib.pyplot as plt
```

Note: In order for categorical explanatory variables to be ordered properly on the horizontal axis, you should convert your categorical explanatory variables (which are often formatted as numeric variables) into a format that Python recognizes as categorical. For example:

```
# change format from integer to categorical
sub2["ND"] = sub2["TAB12MDX"].astype('category')
```

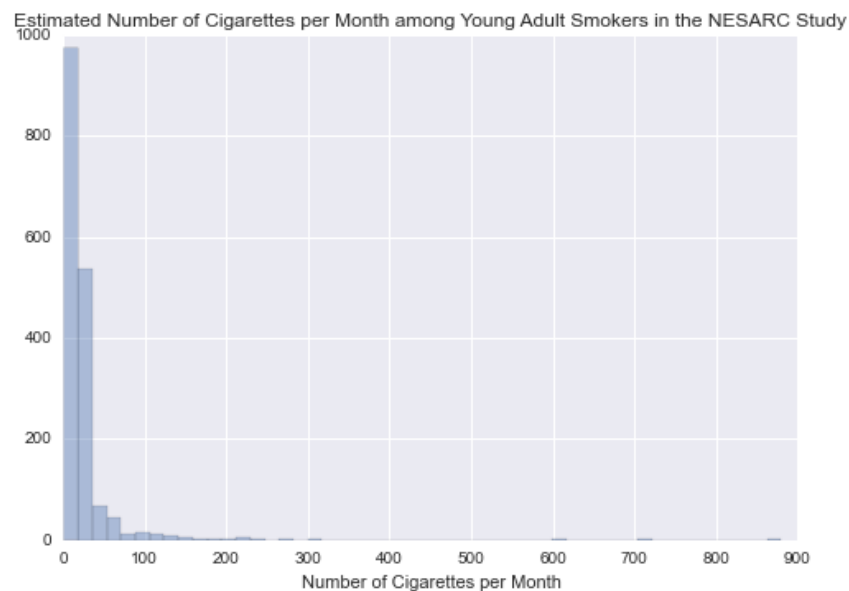
Seaborn code for a univariate bar plot:

```
# univariate bar plot
seaborn.countplot(x="ND", data=sub2)
plt.xlabel('Nicotine Dependence past 12 months')
plt.title('Nicotine Dependence in the Past 12 Months Among Young Adult Smokers in the NESARC Study')
```



Seaborn code for a histogram:

```
seaborn.distplot(sub2["NUMCIGMO_EST"].dropna(), kde=False);
plt.xlabel('Number of Cigarettes per Month')
plt.title('Estimated Number of Cigarettes per Month among Young Adult Smokers in the NESARC Study')
```



Extra:

you can rename categorical variable values for graphing if original values

are not informative

first change the variable format to categorical if you haven't already done so

```
sub2['PACKCATEGORY'] = sub2['PACKCATEGORY'].astype('category')
```

second create a new variable (PACKCAT) that has the new variable value labels

```
sub2['PACKCAT'] = sub2['PACKCATEGORY'].cat.rename_categories(["1-5", "6-10", "11-20", "21-30",  
"More than 30"])
```

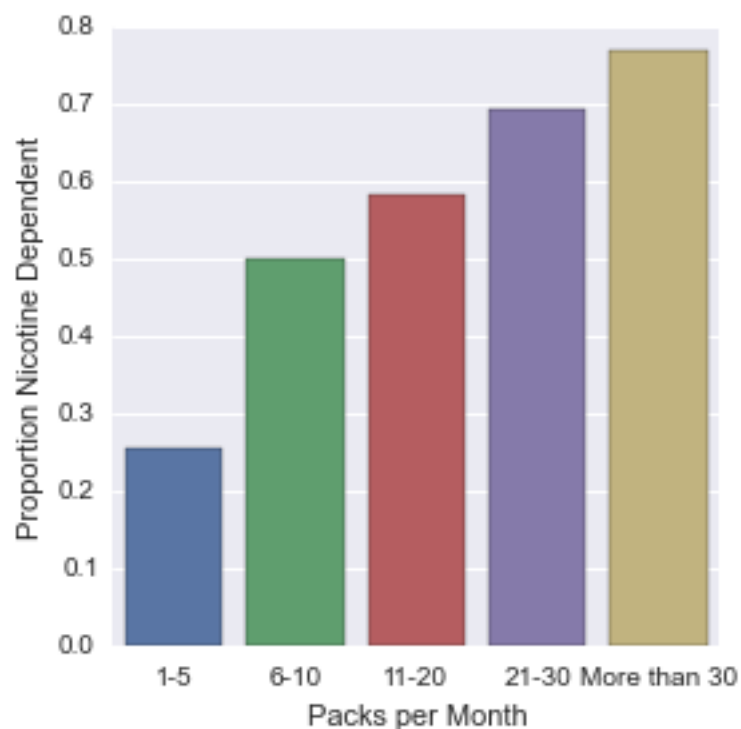
Seaborn code for a bivariate bar plot:

bivariate bar graph

```
seaborn.factorplot(x="PACKCAT", y="TAB12MDX", data=sub2, kind="bar", ci=None)
```

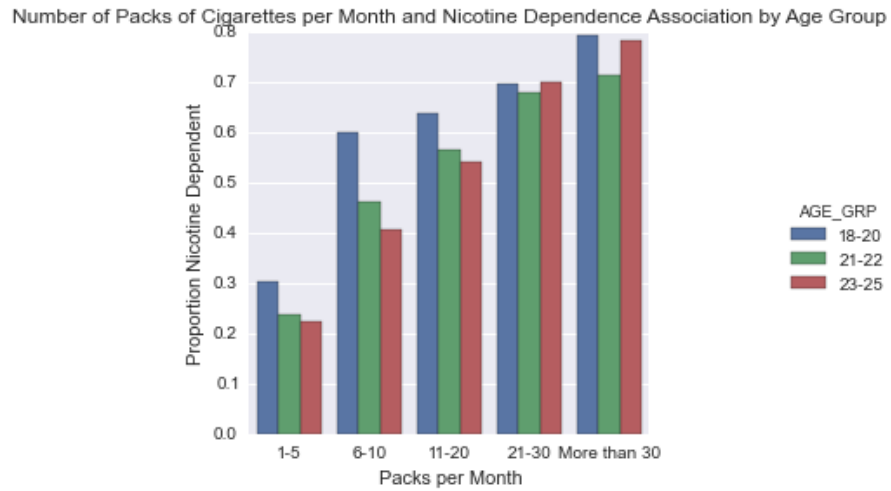
```
plt.xlabel('Packs per Month')
```

```
plt.ylabel('Proportion Nicotine Dependent')
```



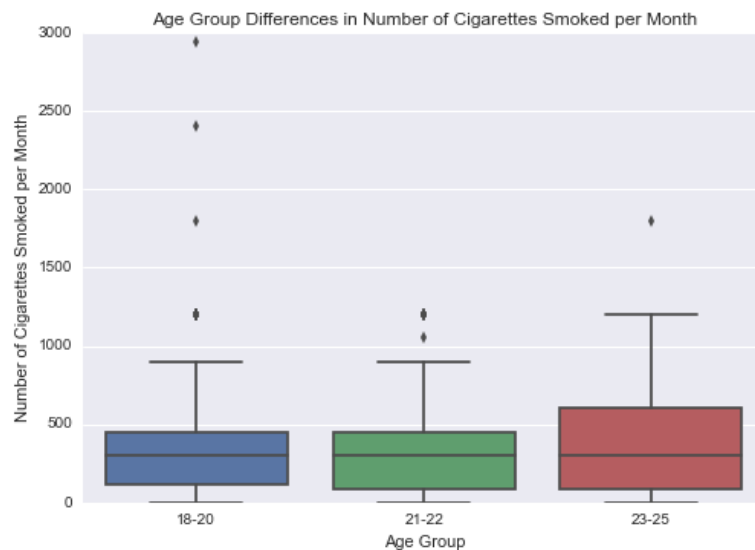
Seaborn code for multivariate bar graph:

```
# multivariate bar graph
seaborn.factorplot(x="PACKCAT", y="TAB12MDX", hue="AGE_GRP", data=sub2, kind="bar", ci=None)
plt.xlabel('Packs per Month')
plt.ylabel('Proportion Nicotine Dependent')
plt.title('Number of Packs of Cigarettes per Month and Nicotine Dependence Association by Age Group')
```



Seaborn code for boxplot:

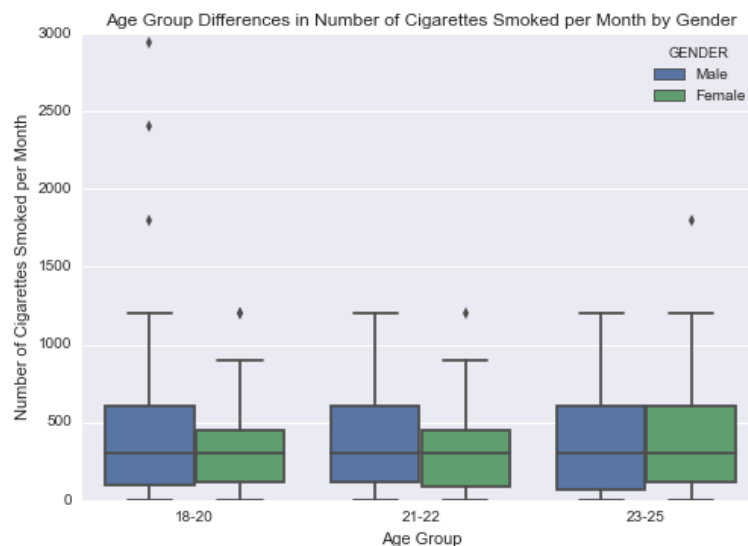
```
# boxplot
seaborn.boxplot(x="AGE_GRP", y="NUMCIGMO_EST", data=sub2)
plt.xlabel('Age Group')
plt.ylabel('Number of Cigarettes Smoked per Month')
plt.title('Age Group Differences in Number of Cigarettes Smoked per Month')
```



Seaborn code for multivariate boxplot:

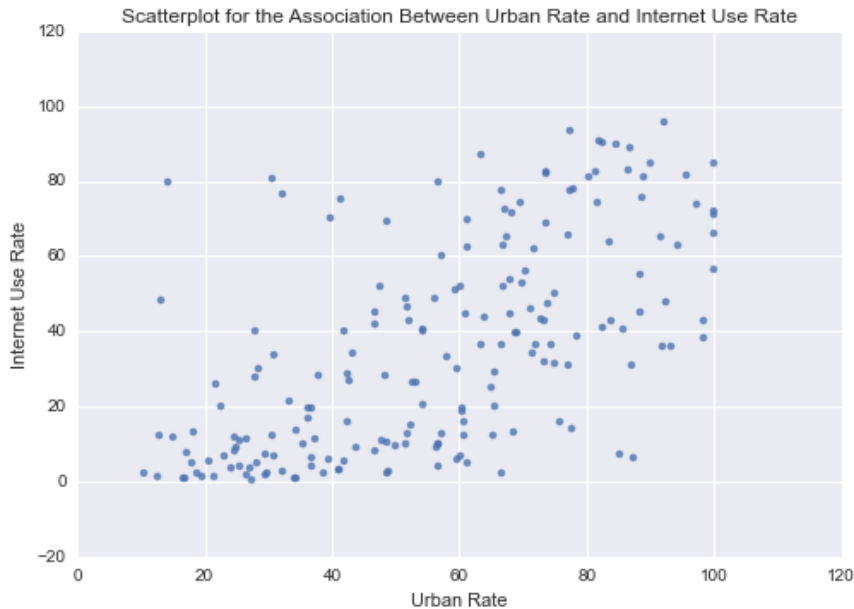
```
sub2['SEX'] = sub2['SEX'].astype('category')
sub2['GENDER'] = sub2['SEX'].cat.rename_categories(["Male", "Female"])
```

```
# multivariate boxplot
seaborn.boxplot(x="AGE_GRP", y="NUMCIGMO_EST", hue="GENDER", data=sub2)
plt.xlabel('Age Group')
plt.ylabel('Number of Cigarettes Smoked per Month')
plt.title('Age Group Differences in Number of Cigarettes Smoked per Month by Gender')
```



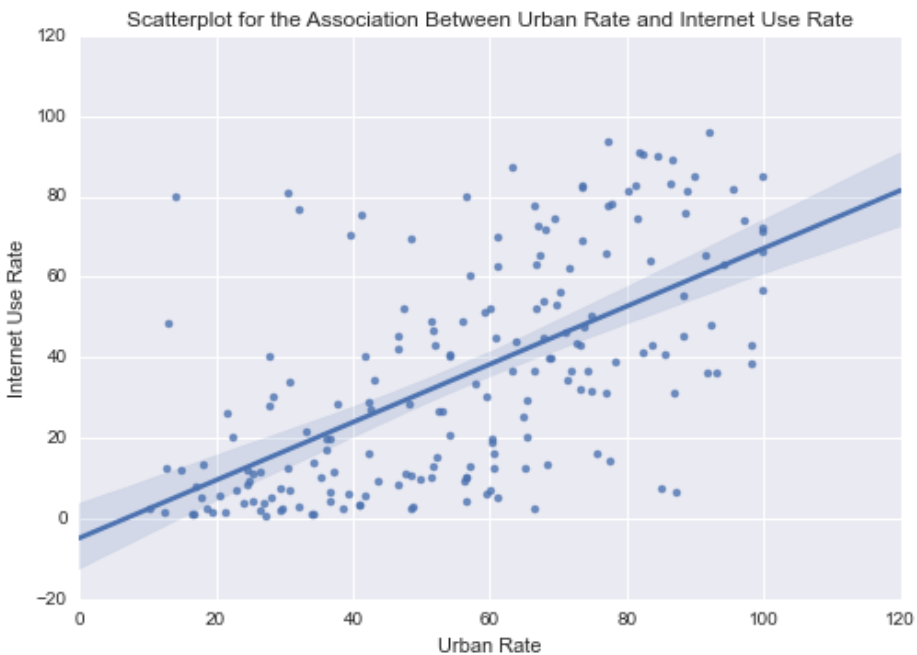
Seaborn code for a basic scatterplot:

```
scat1 = sns.regplot(x="urbanrate", y="internetuserate", scatter=True, fit_reg=False, data=data)
plt.xlabel('Urban Rate')
plt.ylabel('Internet Use Rate')
plt.title('Scatterplot for the Association Between Urban Rate and Internet Use Rate')
```



Ad

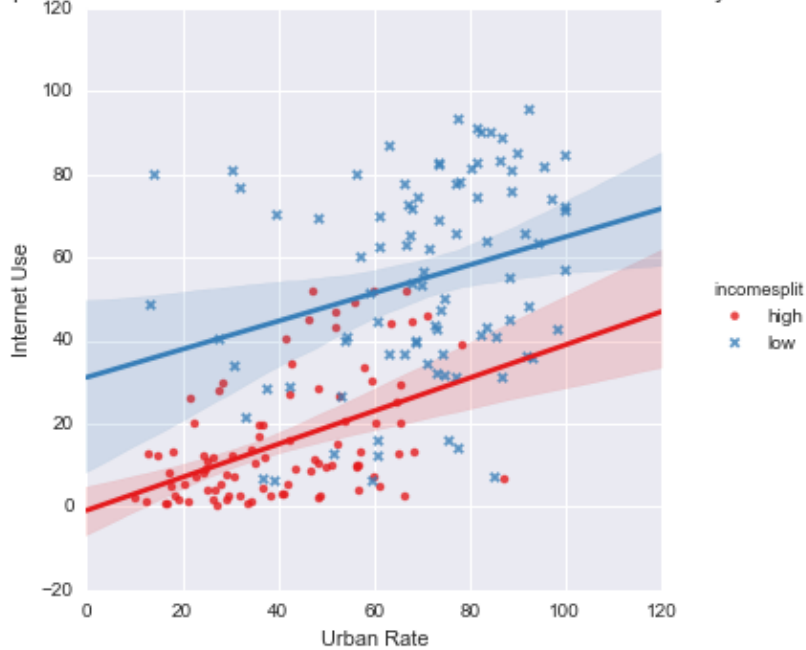
```
scat1 = sns.regplot(x="urbanrate", y="internetuserate", data=data)
plt.xlabel('Urban Rate')
plt.ylabel('Internet Use Rate')
plt.title('Scatterplot for the Association Between Urban Rate and Internet Use Rate')
```



Graph an interaction:

```
sns.lmplot(x="urbanrate", y="internetuserate", hue="incomesplit", data=data2,  
           markers=["o", "x"], palette="Set1");  
plt.xlabel('Urban Rate')  
plt.ylabel('Internet Use')  
plt.title('Scatterplot for the Association Between Urban Rate and Internet Use Rate by Income')
```

Scatterplot for the Association Between Urban Rate and Internet Use Rate by Income



Here's what you get if you change palette="Set1" to palette="Set2"

Scatterplot for the Association Between Urban Rate and Internet Use Rate by Income

