It looks like ell\_percent correlates with sat\_score more strongly, because the scatterplot is more linear. However, there's still the cluster of schools that have very high ell\_percent values and low sat\_score values. This cluster represents the same group of international high schools we investigated earlier.

In order to explore this relationship, we'll want to map out ell\_percent by school district. The map will show us which areas of the city have a lot of English language learners.

We learned how to use the [Basemap](http://matplotlib.org/basemap/" \t "_blank) package to create maps in the [Visualizing Geographic Data mission](https://www.dataquest.io/mission/224/visualizing-geographic-data). The Basemap package enables us to create high-quality maps, plot points over them, and then draw coastlines and other features.

We extracted the coordinates for all of the schools earlier, and stored them in the lat and lon columns. The coordinates will enable us to plot all of the schools on a map of New York City.

We can set up the map with this code:



from mpl\_toolkits.basemap import Basemap

m = Basemap(

   projection='merc',

   llcrnrlat=40.496044,

   urcrnrlat=40.915256,

   llcrnrlon=-74.255735,

   urcrnrlon=-73.700272,

   resolution='i'

)

​

m.drawmapboundary(fill\_color='#85A6D9')

m.drawcoastlines(color='#6D5F47', linewidth=.4)

m.drawrivers(color='#6D5F47', linewidth=.4)

This code snippet will create a map that centers on New York City (llcrnrlat, urcrnrlat, llcrnrlon, and urcrnrlondefine the corners of the geographic area the map depicts). It will also draw coastlines and rivers accordingly.

Now all we need to do is convert our lat and lon coordinates to x and y coordinates so we can plot them on top of the map. This will show us where all of the schools in our data set are located.

As you may recall, in order to plot coordinates using Basemap, we need to:

* Convert the pandas series containing the latitude and longitude coordinates to lists using the [pandas.Series.tolist()](http://pandas.pydata.org/pandas-docs/stable/generated/pandas.Series.tolist.html" \t "_blank)method.
* Make a scatterplot using the longitudes and latitudes with the [scatter()](http://matplotlib.org/basemap/api/basemap_api.html#mpl_toolkits.basemap.Basemap.scatter) method on the Basemap object.
* Show the plot using the [pyplot.show()](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.show" \t "_blank) method.

We also need to make sure we pass a few keyword arguments to the scatter() method:

* s - Determines the size of the point that represents each school on the map.
* zorder - Determines where the method draws the points (that represent schools) on the z axis. In other words, it determines the order of the layers on the map. If we set zorder to 2, the method will draw the points on top of the continents, which is where we want them.
* latlon - A Boolean value that specifies whether we're passing in latitude and longitude coordinates instead of x and y plot coordinates.

Instructions

* Set up the map using the code snippet you saw above -- the one that creates a map, then draws rivers, coastlines, and boundaries.
* Convert the lon column of combined to a list, and assign it to the longitudes variable.
* Convert the lat column of combined to a list, and assign it to the latitudes variable.
* Call the [Basemap.scatter()](http://matplotlib.org/basemap/api/basemap_api.html" \l "mpl_toolkits.basemap.Basemap.scatter" \t "_blank) method on m, and pass in longitudes and latitudes as arguments.
  + Make sure to pass in longitudes and latitudes in the correct order.
  + Pass in the keyword argument s=20 to increase the size of the points in the scatterplot.
  + Pass in the keyword argument zorder=2 to plot the points on top of the rest of the map. Otherwise the method will draw the points underneath the land.
  + Pass in the keyword argument latlon=True to indicate that we're passing in latitude and longitude coordinates, rather than axis coordinates.
* Show the plot using the [pyplot.show()](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.show" \t "_blank) method.

Answer

import pandas as pd

import matplotlib.pyplot as plt

from mpl\_toolkits.basemap import Basemap

m = Basemap(

projection='merc',

llcrnrlat=40.496044,

urcrnrlat=40.915256,

llcrnrlon=-74.255735,

urcrnrlon=-73.700272,

resolution='i'

)

m.drawmapboundary(fill\_color='#85A6D9')

m.drawcoastlines(color='#6D5F47', linewidth=.4)

m.drawrivers(color='#6D5F47', linewidth=.4)

longitudes = combined["lon"].tolist()

latitudes = combined["lat"].tolist()

m.scatter(longitudes, latitudes, s = 20, zorder = 2, latlon = True)

plt.show()

