Analyzing variables in drought deaths

1: Introduction

For this project, I chose to use indicator data from Gapminder. Specifically, I am using the datasets on people affected and killed by droughts between 1970 and 2008. I've downloaded these as Excel files, which I then saved as CSVs.

The questions I'm interested in answering with these datasets are:

```
-how has the global amount of deaths from drought changed over time?

    -does it correlate with how many people have been *affected* by
drought?

-How do other indicators relate to people dying from drought?

-how do effects from drought differ between countries? Between regio
ns?
```

2: Importing libraries and initial drought data

The first thing I am doing is importing the libraries necessary to read CSVs and to use Pandas dataframes. I'm starting off with two CSV files, pulled from Gapminder's Excel spreadsheets for people affected and killed by drought, by country, from 1970 to 2008.

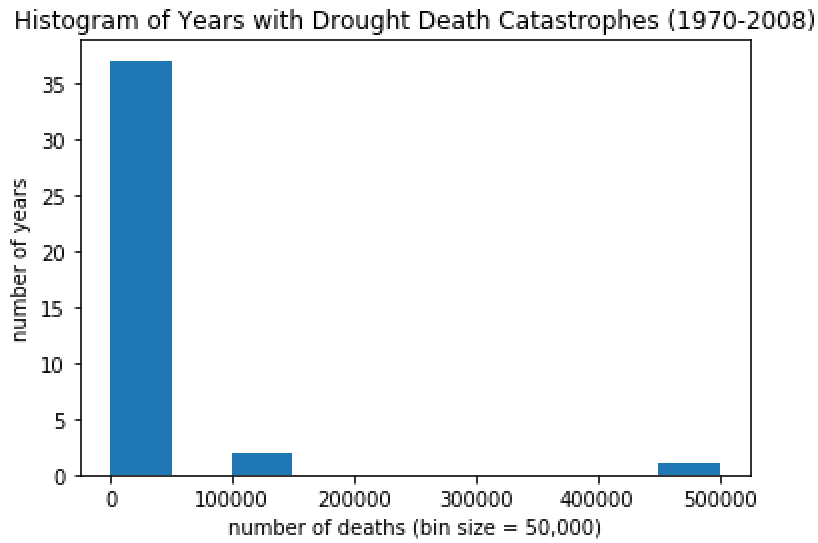
In [1]:
```
import unicodecsv #import csv reader
import pandas as pd #import pandas for data frames
import numpy as np #import numpy for numerical functions
import matplotlib.pyplot as plt #import plotting libraries
%matplotlib inline

drought_death_filename = '/Users/thinkpad/Downloads/indicator_drought_kill
ed.csv'
drought_affected_filename = '/Users/thinkpad/Downloads/indicator_drought_a
ffected.csv'
```

Next, I am going to load the two CSV files into Pandas dataframes.

In [2]:
```
#loading data from csv files into dataframes
deaths_df = pd.read_csv(drought_death_filename, index_col='Country')
affected_df = pd.read_csv(drought_affected_filename, index_col='Country')
```

I want to look at how big a global problem drought has been during the period covered in the data sample. Is drought a constant killer in the background, or do we occasionally have catastrophic droughts that kill many people? I am going to make a histogram of which years saw deaths totaling more or less than 100,000. That means that I am going to add up all the deaths across different countries by year, turning the dataframe into a series.

In [77]:
```
#Adds up data from all countries, so that the data
#becomes a series describing drought deaths, worldwide, by year
series_deaths_by_year = deaths_df.sum(axis='index')
plt.hist(series_deaths_by_year, bins=[0, 50000, 100000, 150000, 200000, 25
0000, 300000, 350000, 400000, 450000, 500000])
plt.xticks()
plt.title('Histogram of Years with Drought Death Catastrophes (1970-2008)'
)
plt.xlabel('number of deaths (bin size = 50,000)')
plt.ylabel('number of years')
plt.show()
```



It looks as though there are fewer than five years that saw more than 100,000 deaths from drought. That makes it seem like large drought catastrophes are rare. If drought death was constantly happening on a large scale, i would expect this histogram to be much more evenly distributed.

And to get a more precise view of the numbers, I'm going to print the series.

In [42]:
```
actual_deaths_by_year = series_deaths_by_year[series_deaths_by_year>0]
#this removes zero values to save space
print actual_deaths_by_year.sort_values(ascending=False)
```

```
1983    450020.0
1973    119000.0
1981    100500.0
1991      2000.0
1988      1600.0
1987      1317.0
1997       732.0
2002       579.0
1999       382.0
1982       280.0
```

```
1989       237.0
1984       230.0
2005       161.0
2006       134.0
2001       108.0
1986        84.0
2004        80.0
1978        63.0
2000        59.0
1998        20.0
1979        18.0
2003         9.0
2008         4.0
dtype: float64
```

I now know that the three years with more than 100,000 deaths from drought are 1973, 1981, and 1983. The year in the data set with the next-most droughts, 1991, has only ~2000 deaths, a mere fraction of 100,000.

I want to look at the numbers of people affected and killed by drought by year globally. My naive hypothesis is that the years with the most people affected by drought, will *also* be the years with the most deaths: 1971, 1983, and 1985.

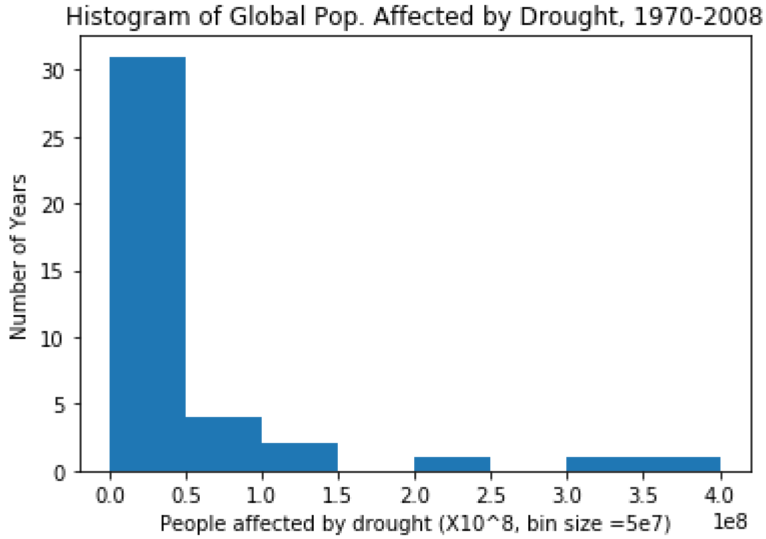
I am going to start by summing the "affected" dataframe by year, just like I did with the "death" dataframe. Then, I am going to make a histogram of the data. I want to see how the distribution of people affected by drought compares with the distribution of those killed.

In [78]:

```
#Adds up data from all countries, so that the data
#becomes a series of number of people affected by drought, worldwide, by y
ear
series_affected_by_year=affected_df.sum(axis='index')

#creates histogram of affected by drought data
plt.hist(series_affected_by_year, bins = [0, 50000000, 100000000, 15000000
0, 200000000, 250000000, 300000000, 350000000, 400000000])
plt.title("Histogram of Global Pop. Affected by Drought, 1970-2008")
plt.ylabel("Number of Years")
plt.xlabel("People affected by drought (X10^8, bin size =5e7)")
```

Out[78]:

```
<matplotlib.text.Text at 0x10103438>
```

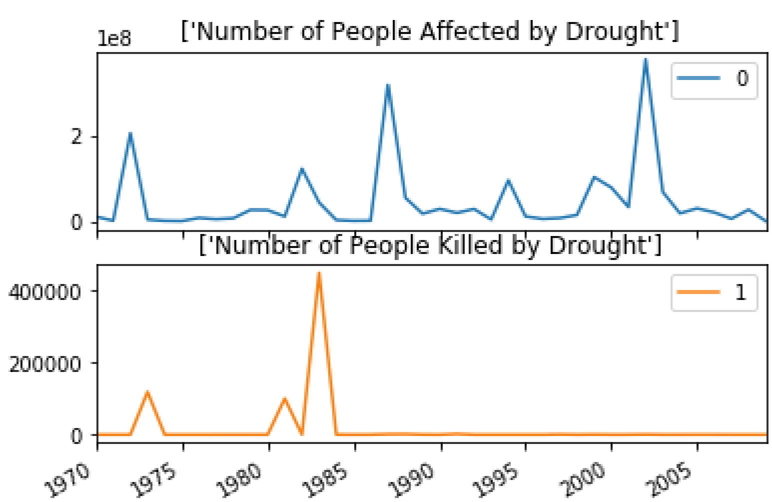


This looks a lot like the distribution of data in the histogram used for deaths by year, but there are some differences. First of all, many, many more people are affected by drought, than killed by it. The bin sizes here are three orders of magnitude, or 1e3 times as big. Additionally, the distribution is a little smoother that the distribution for deaths.

Now I will plot the "affected" and "killed" on the same plot so that I can get a good sense of their correlation.

In [43]:
```
#The next two lines will plot the data on the same graph
affected_v_death_df=pd.concat([series_affected_by_year, series_deaths_by_y
ear], axis=1)

affected_v_death_df.plot(subplots=True, sharex=True, sharey=False, title =
 [["Number of People Affected by Drought"],["Number of People Killed by Dr
ought"]])
#print affected_v_death_df
```

Out[43]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x000000000E3A55C0
>,
       <matplotlib.axes._subplots.AxesSubplot object at 0x000000000EA26F60
>], dtype=object)
```



This interests me because I see about 3 spikes in deaths from drought on this plot, one in 1973, one at around 1981, and one at 1983. From 1983 to 2008, it appears that deaths from drought were close to zero. But we can see from the above plot, that many more people were affected total by drought in the late 1980s and early 2000s than during the 70s and early 80s when most of these deaths occurred.

I have some hypotheses about why so many fewer people who were affected by drought, ended up dying from it, after 1985.

The first question I want to answer is: I have heard that global extreme poverty has been decreasing drastically in recent years. I wonder if people affected by drought are more likely to die if they are in extreme poverty. I would expect that, if true, a reduction in global poverty might be followed by a reduction in deaths from drought.

3: The poverty and drought relationship

To do that, I am going to add two more Gapminder data sets, poverty headcount as percent of population, and population total, both tallied by country.

We don't know how the poverty data set might differ from our drought data sets. It may include, or lack, some years or countries included in the drought data. So, our process is going to include some data reformatting and cleaning to make sure we can make comparisons between these data sets.

In [6]:
```
poverty_headcount_filename = '/Users/thinkpad/Downloads/poverty.csv'
#creates filename for the poverty csv
poverty_df = pd.read_csv(poverty_headcount_filename, index_col="Country")
#loads poverty headcount data into dataframe
poverty_df = poverty_df/100 #convert percentage into ratio
print poverty_df[0:10]
```

```
             1974  1975  1976  1977  1978  1979  1980    1981  1982  1983  \
Country                                                                     
Afghanistan   NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Albania       NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Algeria       NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Angola        NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Argentina     NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Armenia       NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Australia     NaN   NaN   NaN   NaN   NaN   NaN   NaN  0.0067   NaN   NaN   
Austria       NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
Azerbaijan    NaN   NaN   NaN   NaN   NaN   NaN   NaN     NaN   NaN   NaN   
```

```
Bahamas         NaN    NaN    NaN    NaN    NaN    NaN    NaN      NaN    NaN    NaN


               ...     2004     2005     2006     2007     2008     2009     2010
 \
Country        ...

Afghanistan    ...      NaN      NaN      NaN      NaN      NaN      NaN      NaN

Albania        ...   0.0053   0.0044      NaN      NaN   0.0020      NaN      NaN

Algeria        ...      NaN      NaN      NaN      NaN      NaN      NaN      NaN

Angola         ...      NaN      NaN      NaN      NaN      NaN   0.4337      NaN

Argentina      ...   0.0629   0.0462   0.0376   0.0301   0.0274   0.0268   0.0173

Armenia        ...   0.0760   0.0422   0.0317   0.0337   0.0141   0.0156   0.0250

Australia      ...      NaN      NaN      NaN      NaN      NaN      NaN      NaN

Austria        ...   0.0034      NaN      NaN      NaN      NaN      NaN      NaN

Azerbaijan     ...   0.0000   0.0000      NaN      NaN   0.0031      NaN      NaN

Bahamas        ...      NaN      NaN      NaN      NaN      NaN      NaN      NaN


                 2011     2012   2013
Country
Afghanistan       NaN      NaN    NaN
Albania           NaN   0.0046    NaN
Algeria           NaN      NaN    NaN
Angola            NaN      NaN    NaN
Argentina      0.0141      NaN    NaN
Armenia        0.0245   0.0175    NaN
Australia         NaN      NaN    NaN
Austria           NaN      NaN    NaN
Azerbaijan        NaN      NaN    NaN
Bahamas           NaN      NaN    NaN

[10 rows x 40 columns]
```

The "NaN" values seem to indicate to me that data wasn't available for certain years, rather than lack of poverty. Nonetheless, I am going to fill in the "NaN" values so that the data can be plotted.

It's worth mentioning that while it's more convenient to change NaNs in this dataframe to zeroes, to make the data types compatible, this might yield a false impression that extreme poverty does not exist in the countries and years with NaN values. In reality, it probably reflects a lack of information, not a lack of poverty.

In [7]:
```
poverty_df=poverty_df.fillna(0.0)
```

Now, since extreme poverty by country is in ratio form, if we are going to compare poverty and drought deaths indicators, we need to multiply the poverty ratio by the population to get numbers of people in poverty by country.

To do that, we need to first load population data into a dataframe, and print a few lines, to make sure it is formatted to have the same indices, columns, and data types as the drought and poverty dataframes.

In [8]:
```
population_filename = '/Users/thinkpad/Downloads/population.csv'
#filename for global population by country
population_df = pd.read_csv(population_filename, index_col="Country")
#puts global population file in a dataframe
print population_df[0:10]
```

```
                            1800     1810     1820     1830     1840     1
850  \
Country

Abkhazia                     NaN      NaN      NaN      NaN      NaN
NaN
Afghanistan              3280000  3280000  3323519  3448982  3625022  3810
047
Akrotiri and Dhekelia        NaN      NaN      NaN      NaN      NaN
NaN
Albania                  410,445   423591   438671   457234   478227   506
889
Algeria                2,503,218  2595056  2713079  2880355  3082721  3299
305
American Samoa             8,170     8156     8142     8128     8114     7
958
Andorra                     2654     2654     2700     2835     3026     3
230
Angola                   1567028  1567028  1597530  1686390  1813100  1949
329
Anguilla                    2025     2025     2064     2177     2338     2
511
Antigua and Barbuda        37000    37000    37000    37000    37000    37
000

                           1860     1870     1880     1890       ...      \
Country                                                          ...
Abkhazia                    NaN      NaN      NaN      NaN       ...
Afghanistan             3973968  4169690  4419695  4710171       ...
Akrotiri and Dhekelia       NaN      NaN      NaN      NaN       ...
Albania                  552800   610036   672544   741688       ...
Algeria                 3536468  3811028  4143163  4525691       ...
American Samoa             7564     7057     6582     6139       ...
Andorra                    3436     3654     3885     4131       ...
Angola                  2110747  2285417  2473597  2677047       ...
Anguilla                   2693     2888     3097     3320       ...
Antigua and Barbuda       36532    35546    35222    36286       ...

                      Unamed: 82 Unnamed: 83 Unnamed: 84 Unnamed: 85  \
Country
Abkhazia                     NaN         NaN         NaN         NaN
Afghanistan                  NaN         NaN         NaN         NaN
Akrotiri and Dhekelia        NaN         NaN         NaN         NaN
```

```
Albania                          NaN          NaN          NaN          NaN
Algeria                          NaN          NaN          NaN          NaN
American Samoa                   NaN          NaN          NaN          NaN
Andorra                          NaN          NaN          NaN          NaN
Angola                           NaN          NaN          NaN          NaN
Anguilla                         NaN          NaN          NaN          NaN
Antigua and Barbuda              NaN          NaN          NaN          NaN

                      Unnamed: 86 Unnamed: 87 Unnamed: 88 Unnamed: 89  \
Country
Abkhazia                         NaN          NaN          NaN          NaN
Afghanistan                      NaN          NaN          NaN          NaN
Akrotiri and Dhekelia            NaN          NaN          NaN          NaN
Albania                          NaN          NaN          NaN          NaN
Algeria                          NaN          NaN          NaN          NaN
American Samoa                   NaN          NaN          NaN          NaN
Andorra                          NaN          NaN          NaN          NaN
Angola                           NaN          NaN          NaN          NaN
Anguilla                         NaN          NaN          NaN          NaN
Antigua and Barbuda              NaN          NaN          NaN          NaN

                      Unnamed: 90 Unnamed: 91
Country
Abkhazia                         NaN          NaN
Afghanistan                      NaN          NaN
Akrotiri and Dhekelia            NaN          NaN
Albania                          NaN          NaN
Algeria                          NaN          NaN
American Samoa                   NaN          NaN
Andorra                          NaN          NaN
Angola                           NaN          NaN
Anguilla                         NaN          NaN
Antigua and Barbuda              NaN          NaN

[10 rows x 91 columns]
```

By printing out the first few lines, we can already see a few problems:

```
        - the population dataframe has many countries, like Abkhazia,
   that
         are not represented in the drought deaths data
        - the population data frame has some large numbers with commas
   ,
        which means the numbers are strings, but we need them to be
        integers or floating values
        -there are many years represented in the population data that
        are not represented in the drought data
        -There are a lot of NaN values
```

So, we're going to need to fix those things about the population dataframe before we can use it.

In [9]:
```
cleaned_pop_df = population_df[population_df.index.isin(deaths_df.index)]
#gets rid of indices that aren't in the drought deaths dataframe
cleaned_pop_df =cleaned_pop_df.drop(['1800', '1810', '1820', '1830', '1840
```

```
', '1850', '1860', '1870',
       '1880', '1890', '1900', '1910', '1920', '1930', '1940', '1950',
       '1951', '1952', '1953', '1954', '1955', '1956', '1957', '1958',
       '1959', '1960', '1961', '1962', '1963', '1964', '1965', '1966',
       '1967', '1968', '1969', '2009', '2010', '2011', '2012', '2013', '20
14', '2015'], axis=1)
#removes columns from the dataframe that aren't in deaths_df, manually
cleaned_pop_df.drop([col for col in cleaned_pop_df.columns if "Unnamed" in
 col], axis=1, inplace=True)
#removed unnamed columns in population dataframe
def fix_data(entry): #Removes commas from string values, turns them to num
bers
    if isinstance(entry, str) == True: #tests whether value is a string
        entry = entry.replace(",","") #removes commas
    return float(entry) #returns value as a float regardless of original f
ormat
cleaned_pop_df = cleaned_pop_df.applymap(fix_data)
#new cleaned population df
deaths_df.fillna(0.0) #fill NaN values
cleaned_pop_df.fillna(0.0) #fillNaN values
print cleaned_pop_df[0:10] #print first 10 lines to check the data
```

```
                          1970        1971        1972        1973  \
Country
Afghanistan         11121097.0  11412821.0  11716896.0  12022514.0
Albania              2150602.0   2202040.0   2253842.0   2305999.0
Algeria             14550033.0  14960111.0  15377095.0  15804428.0
Angola               6300969.0   6437645.0   6587647.0   6750215.0
Antigua and Barbuda    65369.0     66338.0     67205.0     67972.0
Argentina           23973062.0  24366442.0  24782950.0  25213388.0
Armenia              2518408.0   2580894.0   2643464.0   2705584.0
Australia           12904760.0  13150591.0  13364238.0  13552190.0
Azerbaijan           5178160.0   5287272.0   5393176.0   5496061.0
Bangladesh          65048701.0  66417450.0  67578486.0  68658472.0

                          1974        1975        1976        1977  \
Country
Afghanistan         12315553.0  12582954.0  12831361.0  13056499.0
Albania              2358467.0   2411229.0   2464338.0   2517869.0
Algeria             16247113.0  16709098.0  17190236.0  17690184.0
Angola               6923749.0   7107334.0   7299508.0   7501320.0
Antigua and Barbuda    68655.0     69253.0     69782.0     70223.0
Argentina           25644505.0  26066975.0  26477153.0  26878567.0
Armenia              2766495.0   2825650.0   2882831.0   2938181.0
Australia           13725400.0  13892674.0  14054956.0  14211657.0
Azerbaijan           5596160.0   5693796.0   5789050.0   5882395.0
Bangladesh          69837960.0  71247153.0  72930206.0  74848466.0

                          1978        1979     ...            1999  \
Country                                        ...
Afghanistan         13222547.0  13283279.0     ...      19038420.0
Albania              2571845.0   2626290.0     ...       3114851.0
Algeria             18212331.0  18760761.0     ...      30766551.0
Angola               7717139.0   7952882.0     ...      14601983.0
Antigua and Barbuda    70508.0     70553.0     ...         76041.0
Argentina           27277742.0  27684530.0     ...      36648054.0
Armenia              2991954.0   3044564.0     ...       3093820.0
```

```
Australia            14368543.0  14532401.0     ...        18906936.0
Azerbaijan            5975045.0   6068531.0     ...         8047997.0
Bangladesh           76948378.0  79141947.0     ...       128746273.0

                            2000         2001         2002         2003  \
Country
Afghanistan          19701940.0   20531160.0   21487079.0   22507368.0
Albania               3121965.0    3124093.0    3123112.0    3117045.0
Algeria              31183658.0   31590320.0   31990387.0   32394886.0
Angola               15058638.0   15562791.0   16109696.0   16691395.0
Antigua and Barbuda     77648.0      78972.0      80030.0      80904.0
Argentina            37057453.0   37471535.0   37889443.0   38309475.0
Armenia               3076098.0    3060036.0    3047249.0    3036420.0
Australia            19107251.0   19308681.0   19514385.0   19735255.0
Azerbaijan            8117742.0    8195648.0    8280599.0    8371536.0
Bangladesh          131280739.0  133776064.0  136228456.0  138600174.0

                            2004         2005         2006         2007  \
Country
Afghanistan          23499850.0   24399948.0   25183615.0   25877544.0
Albania               3103758.0    3082172.0    3050741.0    3010849.0
Algeria              32817225.0   33267887.0   33749328.0   34261971.0
Angola               17295500.0   17912942.0   18541467.0   19183907.0
Antigua and Barbuda     81718.0      82565.0      83467.0      84397.0
Argentina            38728778.0   39145491.0   39558750.0   39969903.0
Armenia               3025982.0    3014917.0    3002161.0    2988117.0
Australia            19985475.0   20274282.0   20606228.0   20975949.0
Azerbaijan            8466304.0    8563398.0    8662137.0    8763359.0
Bangladesh          140843786.0  142929979.0  144839238.0  146592687.0

                            2008
Country
Afghanistan          26528741.0
Albania               2968026.0
Algeria              34811059.0
Angola               19842251.0
Antigua and Barbuda     85350.0
Argentina            40381860.0
Armenia               2975029.0
Australia            21370348.0
Azerbaijan            8868713.0
Bangladesh          148252473.0

[10 rows x 39 columns]
```

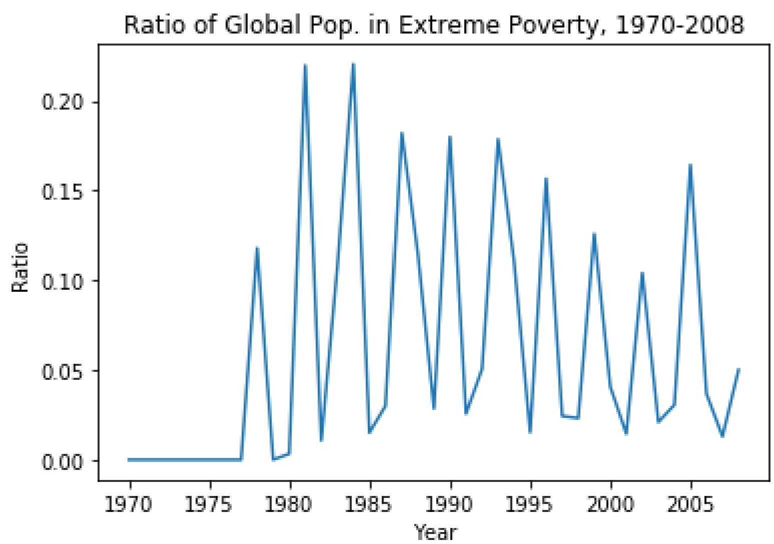
So far, so good! Now we can create a series of the ratio of drought deaths over global population, by year. Then we can plot that against the ratio of people living in extreme poverty.

My hypothesis is that there will be some correlation between decrease in global poverty, and the decrease in deaths from drought.

In [85]:
```
ratio_drought = deaths_df/cleaned_pop_df
#make data frame of fraction of popu;lation killed by drought
mean_drought_by_year = ratio_drought.mean(axis='index')
#ratio of global population that died from drought, by year
```

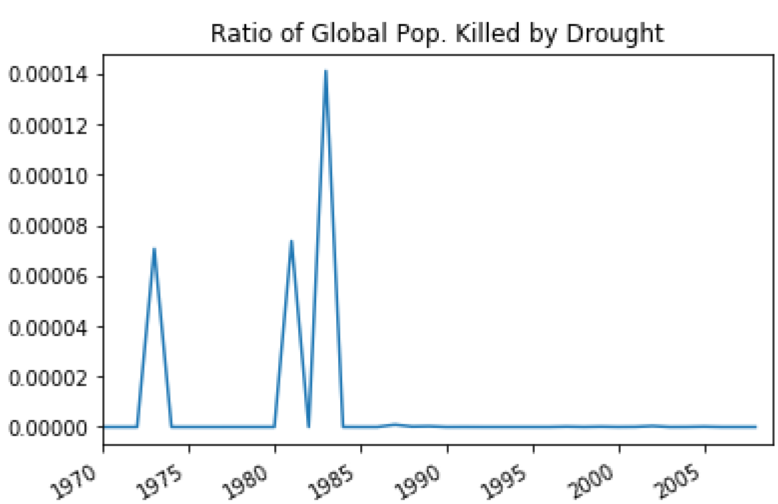
```
poverty_numbers = poverty_df*cleaned_pop_df
#total headcount of people in extreme poverty
poverty_numbers=poverty_numbers.fillna(0.0)
#full NaN values
pop_by_year = cleaned_pop_df.sum(axis='index')
#global population by year
poverty_ratio_series = (poverty_numbers.sum(axis='index')/pop_by_year)
#ratio of global population in poverty



plt.plot(poverty_ratio_series)
plt.title('Ratio of Global Pop. in Extreme Poverty, 1970-2008')
plt.xlabel('Year')
plt.ylabel('Ratio')
```

Out[85]: <matplotlib.text.Text at 0x11338898>



In [88]:
```
mean_drought_by_year.plot(subplots=True,title=["Ratio of Global Pop. Kille
d by Drought"])
```

Out[88]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000117CA828
>], dtype=object)
```



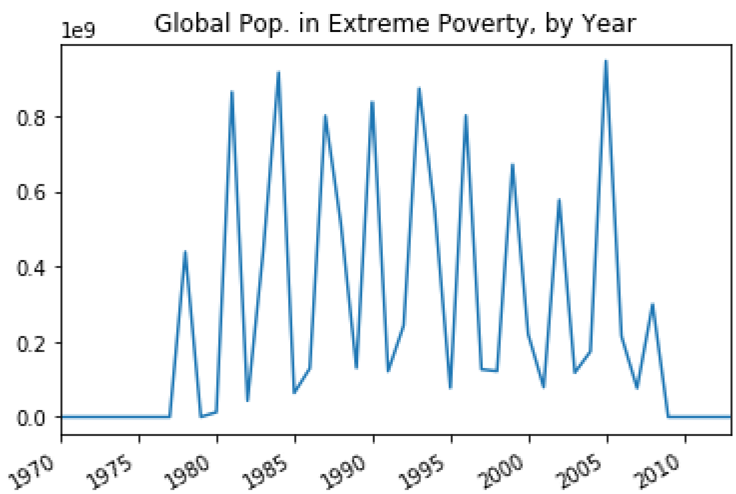
I do see a slight downward trend in extreme poverty from 1984 to 2008--aside from a big upward spike in 2005!

Additionally, the two years with the greatest reported ratio of the population in poverty, immediately precede the two years when the greatest ratio of the global population was killed by drought--this seems to bolster the hypothesis that poverty is a factor in deaths from drought.

I wonder if the trend looks different if we look at overall numbers, rather than ratios.

In [91]:
```
(poverty_numbers.sum(axis='index')).plot(subplots=True, title=['Global Pop
. in Extreme Poverty, by Year'])
```

Out[91]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000106D8E48
>], dtype=object)
```



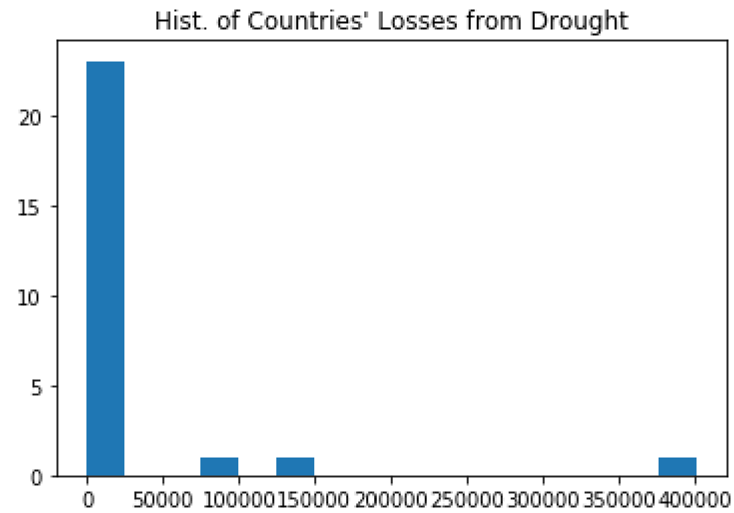
It looks about the same. It seems plausible that reduction in global poverty could be a factor in reduction in drought deaths, but the trend seems a little weak. Next, I am going to examine drought deaths by country, to try and tease out some other factors.

4: Examining drought death by country

I want to start out this part of the analysis the same way I started out teasing out trends while examining drought death by year. The following cell will sum drought deaths by country, so we will how many people were lost to drought in each country from 1970 to 2008.

In [113]:
```
series_deaths_by_country = deaths_df.sum(axis = 'columns')
#sum deaths by country
actual_deaths_by_country = series_deaths_by_country[series_deaths_by_count
ry>0]
#removes zero values to save space
plt.hist(actual_deaths_by_country, bins=16)
plt.title('Hist. of Countries\' Losses from Drought')
#plots histogram of countries' losses
```

Out[113]:
```
<matplotlib.text.Text at 0x1445cc88>
```



Here we can see that even when countries see people die from drought, the numbers are not especially high. However, three countries have had catastrophic losses of more than 50,000 lives.

Next up, I am going to examine which countries those were.

In [12]:
```
actual_deaths_by_country.sort_values(ascending=False, inplace=True)
#sort countries from most deaths to least deaths
print actual_deaths_by_country #print countries with most drought deaths
```

```
Country
Ethiopia            400367.0
Sudan               150000.0
Mozambique          100068.0
Somalia              19623.0
China                 3534.0
Indonesia             1329.0
Swaziland              500.0
Malawi                 500.0
India                  320.0
Rwanda                 237.0
Madagascar             200.0
Kenya                  196.0
Uganda                 194.0
Pakistan               143.0
Burundi                126.0
Papua New Guinea        60.0
Angola                  58.0
Guatemala               41.0
Afghanistan             37.0
Brazil                  20.0
Bangladesh              18.0
Guinea                  12.0
Paraguay                12.0
Algeria                 12.0
Philippines              8.0
Moldova                  2.0
dtype: float64
```

Next, I'm going to look at which countries had the most people *affected* by drought.

In [13]:
```
series_affected_by_country = affected_df.sum(axis='columns')
#sum drought effects by country
series_affected_by_country.sort_values(ascending=False, inplace=True)
#sort countries by most to least affected
print series_affected_by_country[0:10]
#prints 10 most affected
```

```
Country
India          961175320.0
China          366417534.0
Ethiopia        52136567.0
Brazil          47750020.0
Iran            37000000.0
Kenya           36490196.0
Bangladesh      25002018.0
Thailand        23500000.0
Sudan           23360000.0
Malawi          19679202.0
dtype: float64
```

One thing I am noticing here is that of the 10 countries with the greatest drought deaths, seven of them are located in Africa. The remaining 3 are located in Asia.

On the other hand, of the three countries with the most *deaths* from drought, only two are in the top 10 most affected, and neither is at the top. Ethiopia is number 3, and Sudan is number 9. Mozambique doesn't break the top 10 at all.

I also want to look at those spikes in drought deaths. According to the plot above, the three biggest years were 1973, 1981 and 1983. But before I do that, I am going to write a quick function to write indicator-by-year to a series.

In [14]:
```
def year_series(year, df):
    #define function with inputs year and dataframe
    inner_series = df[year]
    #creates series from column of dataframe
    inner_series = inner_series[inner_series>0]
    #removes zero values from series
    return inner_series.sort_values(ascending=False)
    #returns series in descending order
```

Now we can start analyzing the three big years for drought deaths. 1973 will be first.

In [15]:
```
print "1973 Drought Deaths by Country"
deaths_73=year_series('1973', deaths_df)
print deaths_73
```

```
1973 Drought Deaths by Country
Country
Ethiopia    100000.0
Somalia      19000.0
Name: 1973, dtype: float64
```

In [16]:
```
print "1973 People Affected by Drought by Country"
```

```
affected_73 = year_series('1973', affected_df)
print affected_73
#prints population affected by drought in 1973, by country, in descending order
```

```
1973 People Affected by Drought by Country
Country
Ethiopia    3100000.0
Somalia      249000.0
Name: 1973, dtype: float64
```

It looks as though Ethiopia and Somalia alone were responsible for all recorded deaths from drought in 1973. Ethiopia and Somalia border one another, so I'm curious as to whether a particular climate event caused this drought. They were also the only two countries to have anybody affected by drought!

Next, I am going to check out which countries were responsible for the deaths in the 1981 drought.

In [17]:
```
print "1981 Drought Deaths by Country" #print title
deaths_81 = year_series('1981', deaths_df)
print deaths_81
```

```
1981 Drought Deaths by Country
Country
Mozambique    100000.0
Swaziland        500.0
Name: 1981, dtype: float64
```

In [18]:
```
print "1981 People Affected by Drought by Country" #prints title
affected_81 = year_series('1981', affected_df)
print affected_81
```

```
1981 People Affected by Drought by Country
Country
Mozambique    4850000.0
Nigeria       3000000.0
Botswana      1037300.0
Madagascar    1000000.0
Zimbabwe       700000.0
Australia       80000.0
Angola          80000.0
Swaziland         500.0
Name: 1981, dtype: float64
```

It looks as though Mozambique and Swaziland are responsible for all the 1981 deaths. What's interesting, though, is that Nigeria, Botswana, Madagascar, Zimbabwe, Australia and Angola all had tens of thousands affected by drought in 1981, and yet according to Gapminder's records, nobody in those countries died from drought during those years. On the other hand, according to these numbers, all 500 of the people who were affected by drought in Swaziland, died! I'm going to put that in my back pocket, because the differences between the countries that can weather drought without casualties, and those that can't, may be important.

And what about 1983, the year with the most deaths from drought?

In [19]:
```
print "1983 Drought Deaths by Country" #print title
deaths_83 = year_series('1983', deaths_df)
print deaths_83
```

```
1983 Drought Deaths by Country
Country
Ethiopia    300000.0
Sudan       150000.0
Brazil          20.0
Name: 1983, dtype: float64
```

In [20]:
```
print "1983 People Affected by Drought by Country" #print title
affected_83 = year_series('1983', affected_df)
print affected_83
```

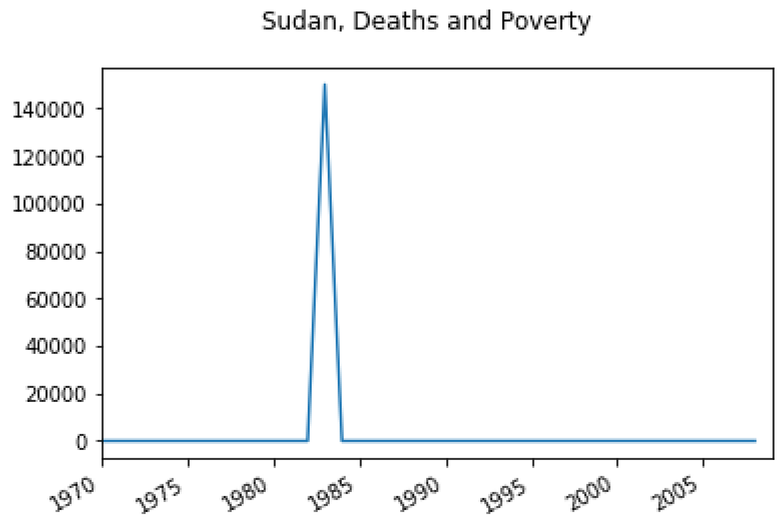
```
1983 People Affected by Drought by Country
Country
Brazil                     20000020.0
Sudan                       8550000.0
Ethiopia                    8050000.0
Bolivia                     3083049.0
Philippines                 1691060.0
Kenya                        600000.0
Lesotho                      500000.0
Congo, Dem. Rep.             300000.0
Sao Tome and Principe         93000.0
Panama                        81000.0
Djibouti                      80000.0
Antigua and Barbuda           75000.0
Fiji                          31000.0
Name: 1983, dtype: float64
```

Ethiopia and Sudan top this list. Ethiopia was also the country to see the most deaths in the 1973 drought. However, again we see that there isn't a straightforward causal relationship between the number of people affected by drought, and the number of people killed by it. in 1983, Brazil saw the most people affected by drought -- 20 million, more than twice Sudan's 8.6 million. And yet, Brazil saw only 20 deaths from drought in 1982, while Sudan saw 150,000.

Out of curiosity, I am going to plot Sudan and Ethiopia's drought deaths timelines alongside their poverty timelines.

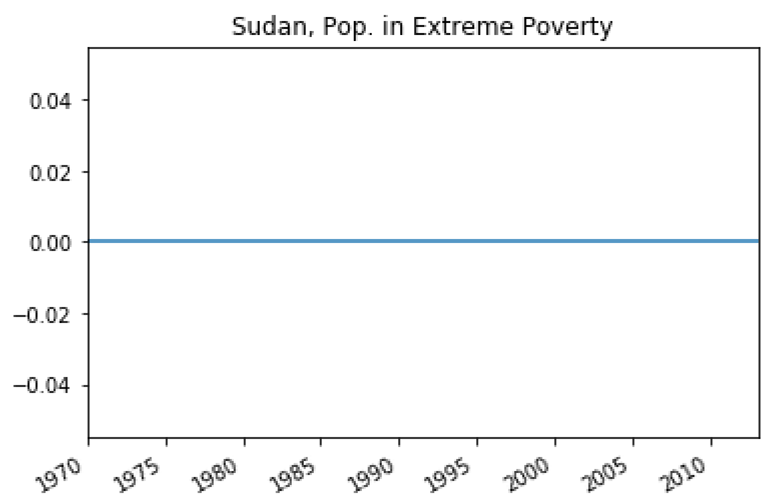
In [95]:
```
deaths_df.loc["Sudan"].plot(subplots = True, title = "Sudan, Drought Death
s")
```

Out[95]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000123DB6A0
>], dtype=object)
```



In [98]:
```
poverty_numbers.loc['Sudan'].plot(subplots=True, title=['Sudan, Pop. in Ex
treme Poverty'])
```

Out[98]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000128BEEF0
>], dtype=object)
```

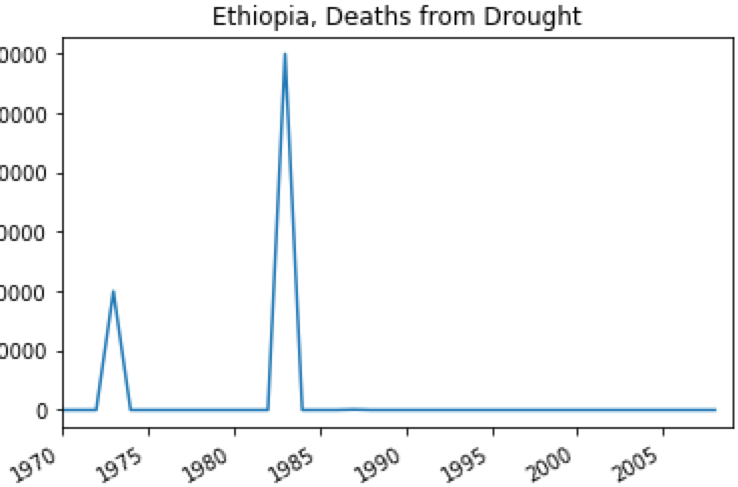


This looks like a weak point in the poverty data--according to the United Nations Development Program, [Sudan has an extreme poverty rate of 50%](). Yet, the data set from Gapminder shows a near-zero ratio of the population in poverty for the last 40 years.

This may be due to issues in collection of data, or something else, but we do now know that we should take any conclusions drawn from the Gapminder poverty dataset with an extra grain of salt.

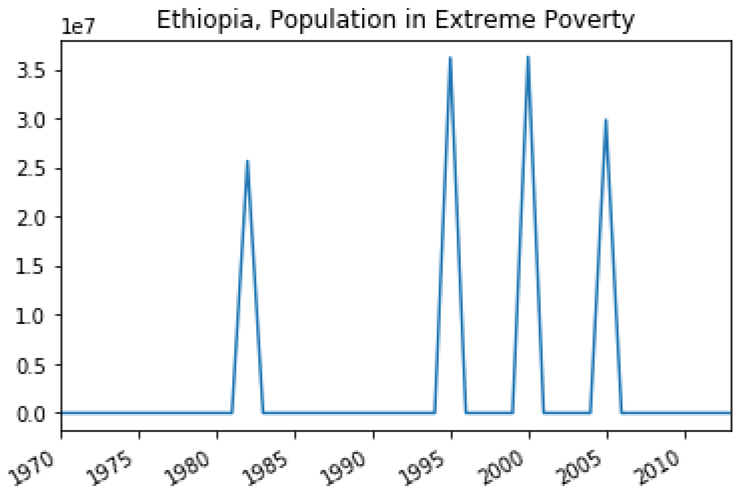
In [99]:
```
deaths_df.loc['Ethiopia'].plot(subplots=True, title=['Ethiopia, Deaths fro
m Drought'])
```

Out[99]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000128F80F0
>], dtype=object)
```



In [100]:
```
poverty_numbers.loc['Ethiopia'].plot(subplots=True, title=['Ethiopia, Population in Extreme Poverty'])
```

Out[100]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000000012B95550
>], dtype=object)
```



While there appears to be a big poverty spike in 1982 that may correspond to the 1983 drought deaths, there doesn't appear to be a big overall trend between drough deaths in Ethiopia and Sudan, and number of people in poverty. This might of course be due to the poor quality of the poverty data, which started off with many NaN values.

However, I am curious about one more variable--how does foreign aid received, relate to extreme poverty and drought death globally?

5: Aid received

In the next cell, I am going to import the aid received csv, and print it out to check it out.

In [23]:
```
aid_filename = '/Users/thinkpad/Downloads/indicator_aid_received.csv'
aid_df = pd.read_csv(aid_filename, index_col='Country')
print aid_df[0:10]
```

```
                            1960       1961       1962       1963       19
64  \
Country

Abkhazia                     NaN        NaN        NaN        NaN        N
aN
Afghanistan             1.776437   3.516253   1.683492   3.573637   4.4076
78
Akrotiri and Dhekelia        NaN        NaN        NaN        NaN        N
aN
Albania                      NaN        NaN        NaN        NaN        N
aN
Algeria                32.875009  39.785973  35.471498  24.683813  19.5488
51
American Samoa               NaN        NaN        NaN        NaN        N
aN
Andorra                      NaN        NaN        NaN        NaN        N
aN
Angola                 -0.010074   4.659959        NaN   0.005723        N
aN
Anguilla                     NaN        NaN        NaN        NaN        N
aN
Antigua and Barbuda          NaN        NaN        NaN        NaN        N
aN

                            1965       1966      1967      1968      1969
 \
Country

Abkhazia                     NaN        NaN       NaN       NaN       NaN

Afghanistan             5.041137   4.608462  3.636801  2.562435  2.333649

Akrotiri and Dhekelia        NaN        NaN       NaN       NaN       NaN

Albania                      NaN        NaN       NaN       NaN       NaN

Algeria                12.264529  10.377884  8.561455  9.239531  9.838129

American Samoa               NaN        NaN       NaN       NaN       NaN

Andorra                      NaN        NaN       NaN       NaN       NaN

Angola                  0.204377   0.517732  3.236464  0.001750 -0.018916

Anguilla                     NaN        NaN       NaN       NaN       NaN

Antigua and Barbuda          NaN        NaN       NaN       NaN       NaN


                          ...          2001        2002        2003  \
Country                   ...
Abkhazia                  ...           NaN         NaN         NaN
Afghanistan               ...     15.370768   47.687783   56.401679
Akrotiri and Dhekelia     ...           NaN         NaN         NaN
Albania                   ...     87.259349   99.495821  114.108181
Algeria                   ...      6.412711    6.041630    7.461428
```

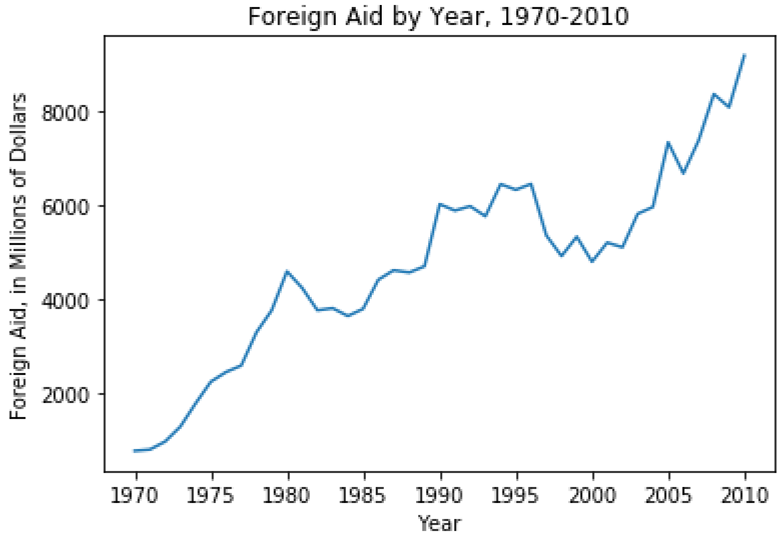
```
American Samoa          ...        NaN        NaN        NaN
Andorra                 ...        NaN        NaN        NaN
Angola                  ...  19.650070  27.805025  32.018925
Anguilla                ...        NaN        NaN        NaN
Antigua and Barbuda     ... 107.875955 167.314643  75.383034

                             2004        2005        2006        2007  \
Country
Abkhazia                      NaN         NaN         NaN         NaN
Afghanistan             79.518324   94.887932   96.309269  157.005494
Akrotiri and Dhekelia         NaN         NaN         NaN         NaN
Albania                 95.988270  101.578713  101.938569   96.997632
Algeria                  9.760450   10.538959    7.184665   11.629003
American Samoa                NaN         NaN         NaN         NaN
Andorra                       NaN         NaN         NaN         NaN
Angola                  71.719434   25.142184    9.612962   14.132086
Anguilla                      NaN         NaN         NaN         NaN
Antigua and Barbuda     19.676960   92.830926   38.610039   85.878862

                              2008        2009        2010
Country
Abkhazia                       NaN         NaN         NaN
Afghanistan             149.920708  186.470442  186.894497
Akrotiri and Dhekelia          NaN         NaN         NaN
Albania                 114.185686  111.804250  106.326405
Algeria                   9.441435    9.116980    5.591486
American Samoa                 NaN         NaN         NaN
Andorra                        NaN         NaN         NaN
Angola                   20.446875   12.864916   12.484598
Anguilla                       NaN         NaN         NaN
Antigua and Barbuda     100.712469   64.235439  214.970127

[10 rows x 51 columns]
```

Looks like we need to fill some NaN values, remove some countries, and remove some years. We'll do that in the next cell. Then we'll total aid by year, and plot it in matplotlib.

In [108]:

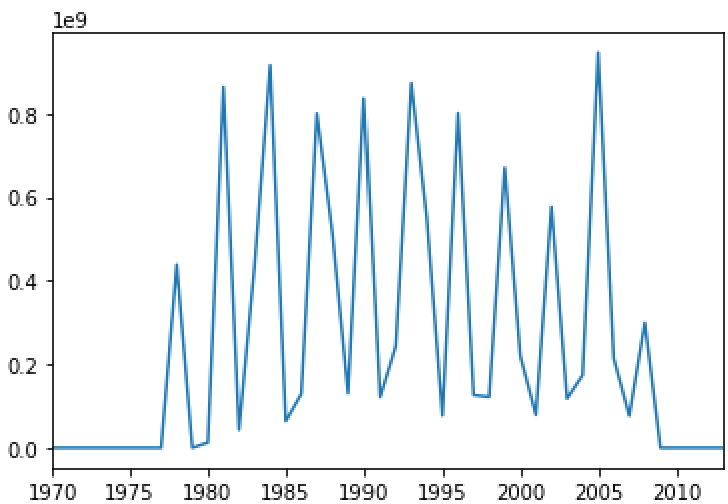
```
cleaned_aid_df = aid_df[aid_df.index.isin(deaths_df.index)]
#remove years that aren't shared in drought dataframe
cleaned_aid_df =cleaned_aid_df.drop(['1960', '1961', '1962', '1963', '1964
', '1965', '1966',
       '1967', '1968', '1969'], axis=1)
#remove irrelevant years
cleaned_aid_df=cleaned_aid_df.fillna(0.0)
#fill NaN values
sum_aid = cleaned_aid_df.sum(axis='index')
#sum foreign aid by year
plt.plot(sum_aid)
plt.title('Foreign Aid by Year, 1970-2010')
plt.xlabel('Year')
plt.ylabel('Foreign Aid, in Millions of Dollars')
```

Out[108]: <matplotlib.text.Text at 0x13981320>



In [25]:
```
poverty_numbers.sum(axis='index').plot()
```

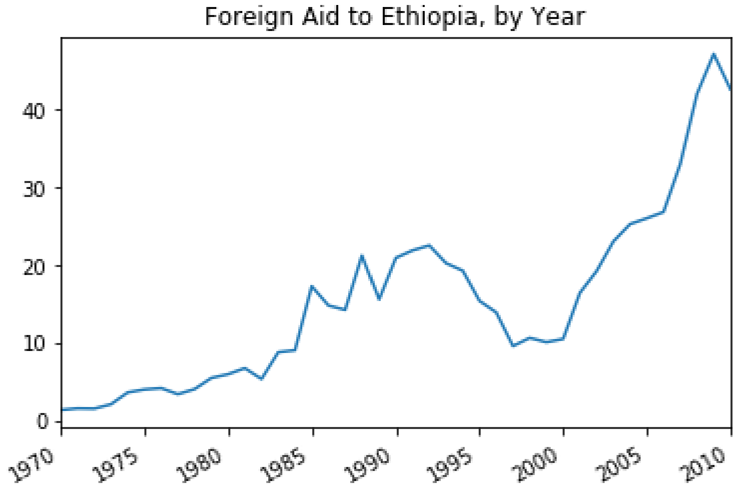
Out[25]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xbe2bc50>
```



I don't see a clear trend, but again, perhaps we can see if there's a relationship between drought, aid, and poverty for one country. For this purpose, I am returning to the Ethiopia data.

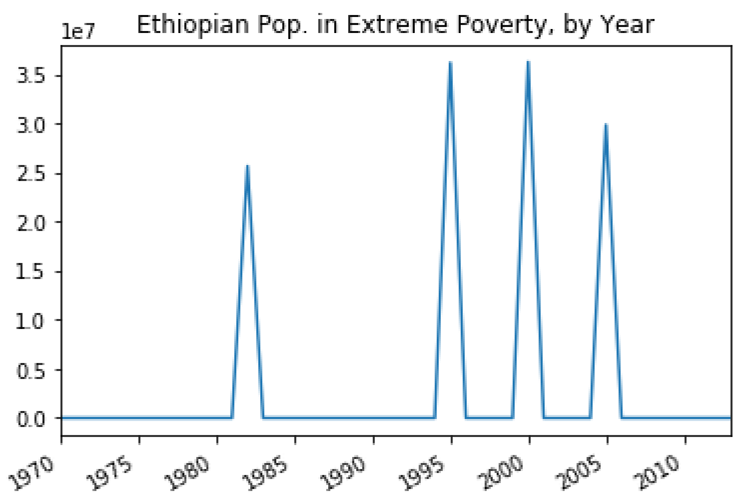
In [111]:
```
cleaned_aid_df.loc['Ethiopia'].plot(subplots=True, title=['Foreign Aid to
Ethiopia, by Year'])
```

Out[111]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000000013F9BE48
>], dtype=object)
```



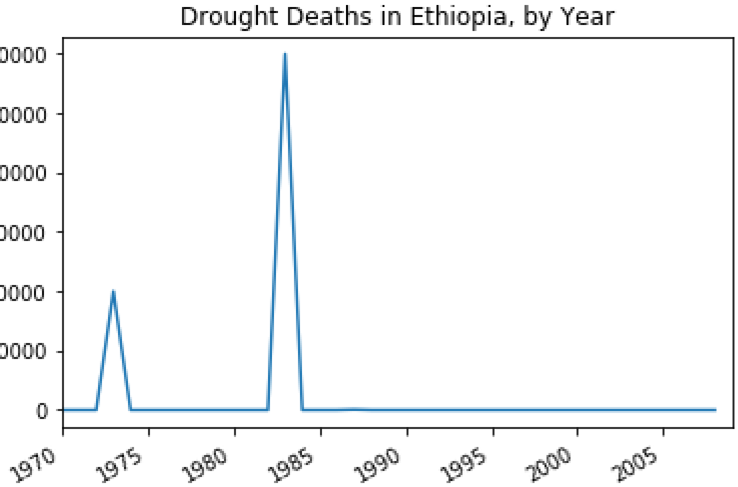
In [92]:
```
poverty_numbers.loc['Ethiopia'].plot(subplots=True,title=['Ethiopian Pop.
in Extreme Poverty, by Year'])
```

Out[92]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x00000000118F4E48
>], dtype=object)
```



In [110]:
```
deaths_df.loc['Ethiopia'].plot(subplots=True, title=['Drought Deaths in Et
hiopia, by Year'])
```

Out[110]:
```
array([<matplotlib.axes._subplots.AxesSubplot object at 0x0000000013D38E80
>], dtype=object)
```



It looks as though deaths from drought in Ethiopia spiked in 1983, one year after a spike in extreme poverty. It also looks as though a downward trend in foreign aid dollars to Ethiopia might correspond to another spike in poverty, in 1995. However, the data here is spotty enough that I would not draw causal conclusions without obtaining more information.

6: Conclusions

What we have learned from this data is:

-For some reason, fewer people died from large droughts that occurred after the mid-1980s. Changes in global poverty may be related.

-East African countries have been hit the hardest by drought during the sample period

-There is not a clear relationship between drought survival, or poverty reduction, and foreign aid dollars, at least when looked at in a specific country, Ethiopia.

Some of the limitations of the data set:

-It may be inaccurate. When I was examining extreme poverty in Sudan, I discovered that according to the Gapminder data, there is either no data on Sudanese poverty, or the measured poverty rate is zero. That directly conflicts with what the United Nations says about poverty in Sudan.

-Relatedly: the information is incomplete. For both foreign aid and global extreme poverty, there were many years for which there was no data, and the dataframe had NaN values for these cells. I filled the NaNs with zeros so that i could use mathematical operations on the dataframes, but this is not necessarily the best solution, as not having data is very different from poverty disappearing.