

## 0.1 Software Carpentry Demo - June 7, 2012

Note: IPython is running here in `--pylab` mode so I didn't have to import matplotlib.

```
%config InlineBackend.figure_format = 'svg'
```

```
import numpy as np
import scipy.stats as spst
import scipy.optimize as spop
import scipy.interpolate as spint
```

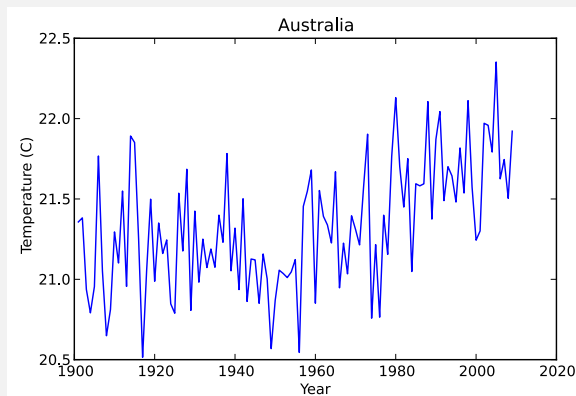
Load the data and take an initial look.

<http://docs.scipy.org/doc/numpy/reference/generated/numpy.loadtxt.html>

```
year, aus, can = np.loadtxt('temperatures.txt', delimiter=',',
                           skiprows=1, unpack=True)
```

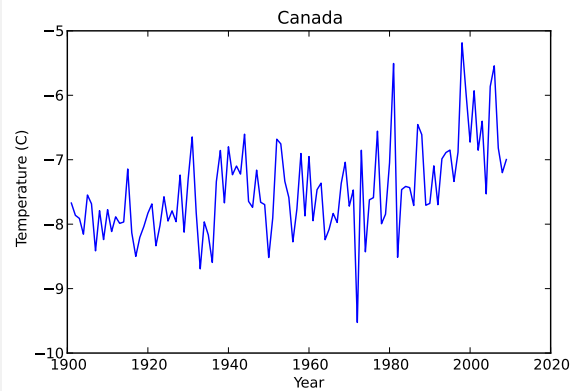
```
plot(year, aus)
title('Australia')
xlabel('Year')
ylabel('Temperature (C)')
```

<matplotlib.text.Text at 0x1059d7cd0>



```
plot(year, can)
title('Canada')
xlabel('Year')
ylabel('Temperature (C)')
```

<matplotlib.text.Text at 0x105dabd90>



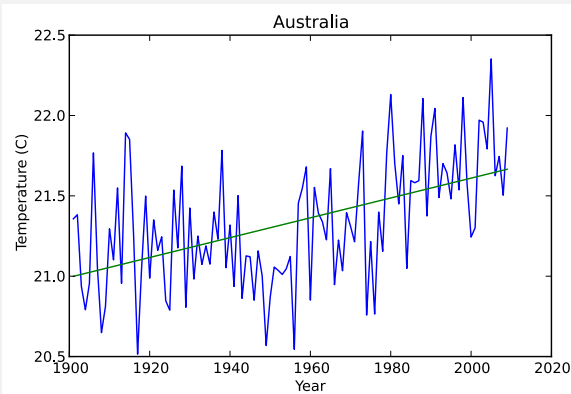
Fit a line to the Australia temperatures using `scipy.stats.linregress`:  
<http://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.linregress.html>

```
aus_line = spst.linregress(year, aus)
```

```
aus_line_pts = aus_line[0] * year + aus_line[1]
```

```
plot(year, aus)
plot(year, aus_line_pts)
title('Australia')
xlabel('Year')
ylabel('Temperature (C)')
```

<matplotlib.text.Text at 0x105f20690>



Fit a line to the Canada data using `scipy.optimize.curve_fit`:  
[http://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve\\_fit.html](http://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html)

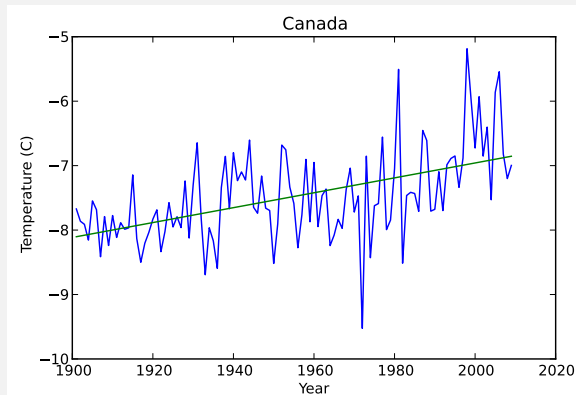
```
def linear_func(year, slope, intercept):
    return slope * year + intercept
```

```
can_line, can_cov = spop.curve_fit(linear_func, year, can)
```

```
can_line_pts = linear_func(year, can_line[0], can_line[1])
```

```
plot(year, can)
plot(year, can_line_pts)
title('Canada')
xlabel('Year')
ylabel('Temperature (C)')
```

<matplotlib.text.Text at 0x105f69f50>



```
spst.pearsonr(year, can)
```

(0.51195782671512846, 1.2713276823278399e-08)

Fit a spline to the Australia data using `scipy.interpolate.UnivariateSpline`:  
<http://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.UnivariateSpline.html>

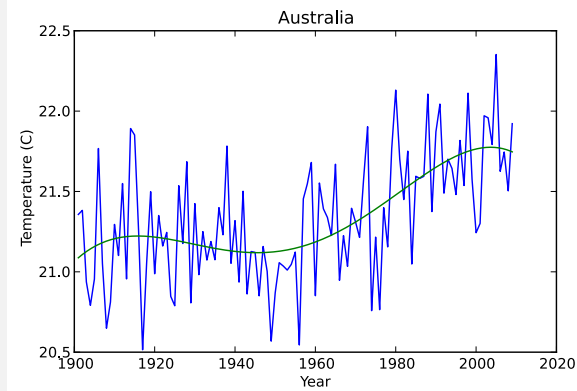
```
spline = spint.UnivariateSpline(year, aus, k=5)
```

```
x = np.linspace(year.min(), year.max(), 1000)
```

```
y = spline(x)
```

```
plot(year, aus)
plot(x, y)
title('Australia')
xlabel('Year')
ylabel('Temperature (C)')
```

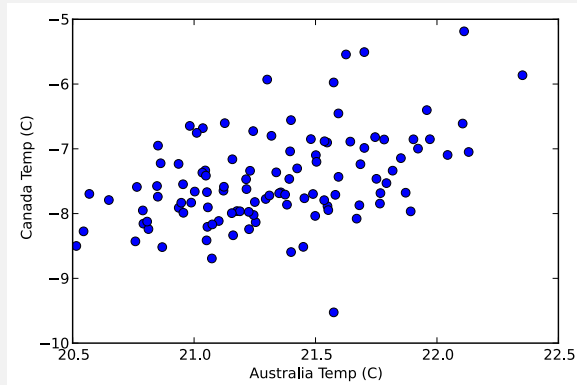
<matplotlib.text.Text at 0x105fb2c50>



Plot the temperatures against each other and fit a line.

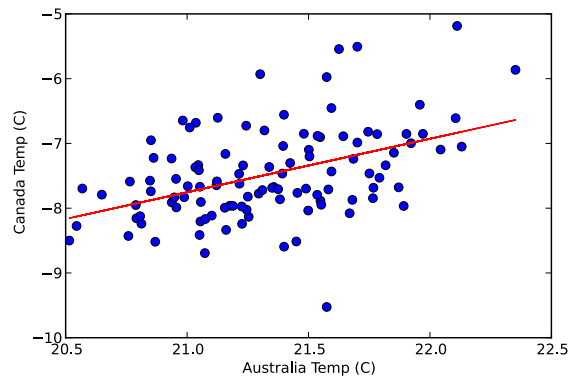
```
plot(aus, can, 'bo')
xlabel('Australia Temp (C)')
ylabel('Canada Temp (C)')
```

<matplotlib.text.Text at 0x105ffe390>



```
new_line = spst.linregress(aus, can)
new_line_pts = new_line[0] * aus + new_line[1]
plot(aus, can, 'bo')
plot(aus, new_line_pts, 'r-')
xlabel('Australia Temp (C)')
ylabel('Canada Temp (C)')
```

<matplotlib.text.Text at 0x106341790>



```
new_line
```

```
(0.8255661193335796,  
-25.090352153252795,  
0.44948681311947275,  
9.4409823433233215e-07,  
0.15861125998986189)
```

Plot the temperatures as variations about their respective best fit lines.

```
aus_sub = aus - aus_line_pts
```

```
can_sub = can - can_line_pts
```

```
plot(year, aus_sub, label='Aus')  
plot(year, can_sub, label='Can')  
xlabel('Year')  
ylabel('Temperature Variation (C)')  
legend(loc='lower left')
```

```
<matplotlib.legend.Legend at 0x1063a7950>
```

