

Interpolating Data

Created using Maple 14.01

Jake Bobowski

```
> restart;
with(StringTools) :
with(plots) :
with(Statistics) :
FormatTime("%m-%d-%Y, %H:%M");
"11-13-2013, 19:10" (1)
```

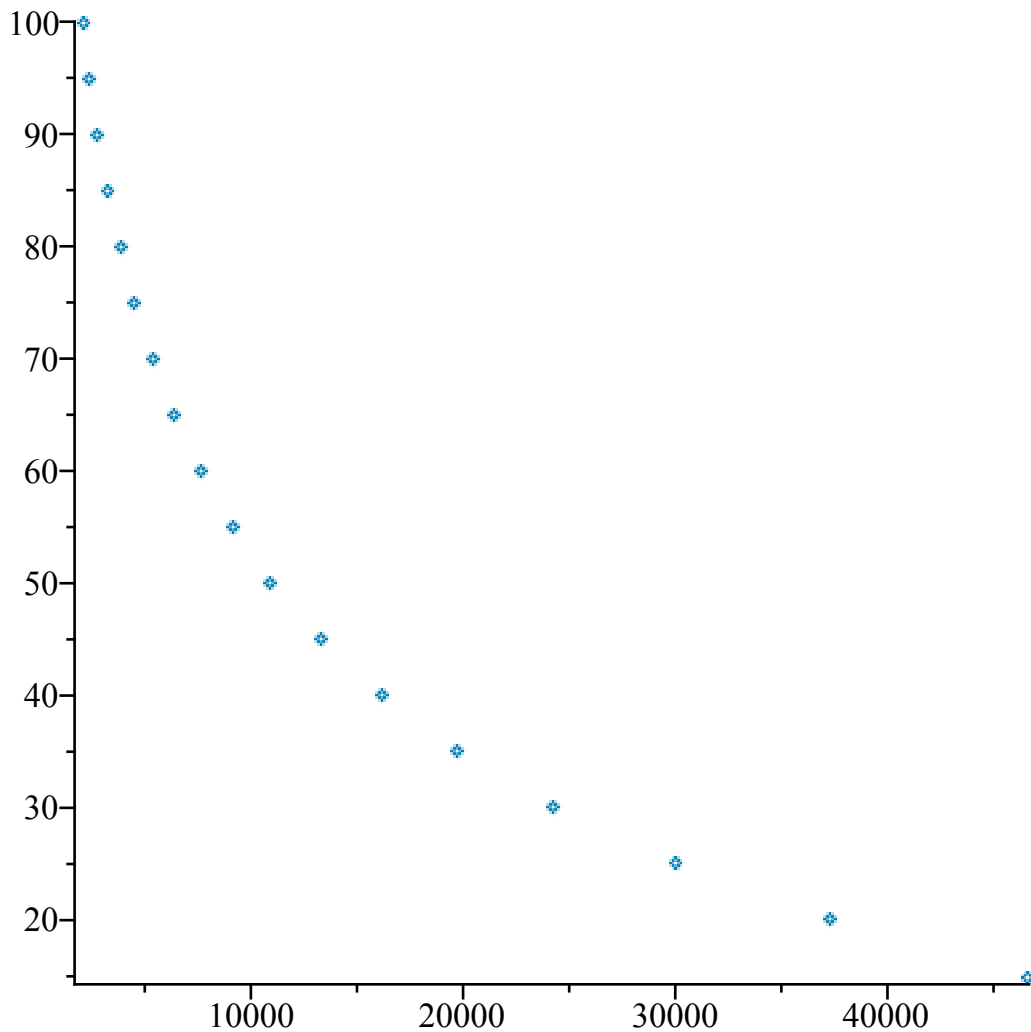
In this Maple tutorial, we will show how to interpolate calibration data. The calibration data can then be used to convert a measured quantity (say a voltage or resistance) to a desired physical quantity (like temperature, for example). In this example, a measured resistance from a thermistor will be converted to temperature using the known calibration data.

The first step is to import and plot the calibration data.

```
> calData
:= readdata("G:\UBCO\2013-2014\people page\2013\Maple\thermistor calibration
data.dat", [float, float]) :
```

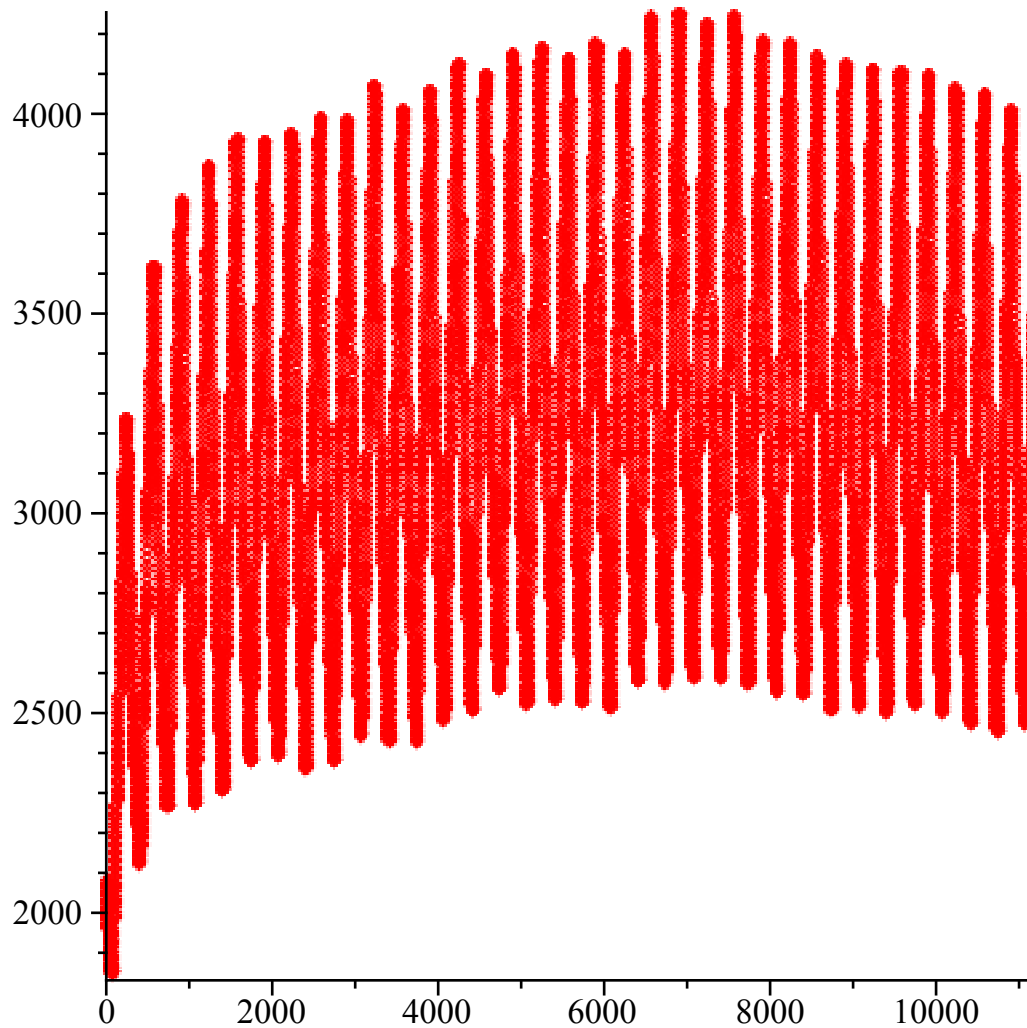
For this set of calibration data, the first column corresponds to resistance and the second column corresponds to temperature.

```
> temp := [seq(calData[i, 2], i = 1 .. nops(calData)) ] :
resistance := [seq(calData[i, 1], i = 1 .. nops(calData)) ] :
calPlot := ScatterPlot(resistance, temp) :
display(calPlot);
```



Now import and plot some example resistance data measured as a function of time. If you've taken PHYS 232, this is an example set of thermal waves data.

```
> Data05cm
:= readdata("G:\UBCO\2013-2014\people
page\2013\Maple\thermalWaves-12092010-00-x05cm f003Hz V80V.dat", [float,
float]) :
Data05cmR := [seq(Data05cm[i, 2], i = 1 ..nops(Data05cm))]:
Data05cmTime := [seq(Data05cm[i, 1], i = 1 ..nops(Data05cm))]:
Data05cmPlot := ScatterPlot(Data05cmTime, Data05cmR, color = red) :
display(Data05cmPlot);
```



Here, we now use the Maple function *ArrayInterpolation*(...) to use the calibration data to convert the resistance measurements to temperature. Notice from the calibration data that large resistance corresponds to low temperature. *ArrayInterpolation* requires three arguments, the first two are the resistance and temperature from the calibration data and the third is the resistance measurements to be converted to temperature. First, note that *ArrayInterpolation* requires the package *CurveFitting* to be loaded.

```
> with(CurveFitting) :
> Data05cmTemp := ArrayInterpolation(resistance, temp, Data05cmR) :
  Data05cmTempPlot := ScatterPlot(Data05cmTime, Data05cmTemp, color = blue) :
  display(Data05cmTempPlot);
```

