Eyeballing Parameters

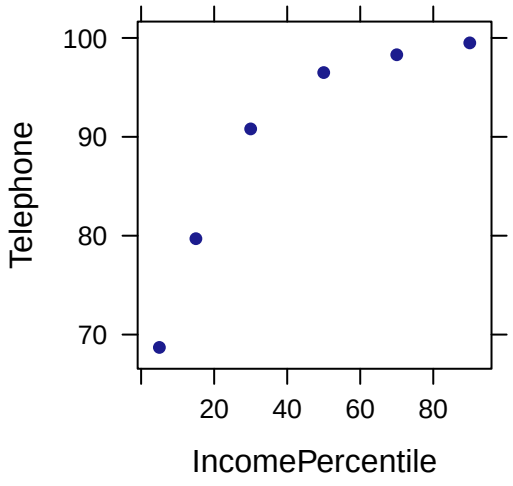
Daniel Kaplan

January 4, 2013

For each graph, choose an appropriate function form to represent the pattern shown by the data and estimate, by eye, the parameters that will fit the data.

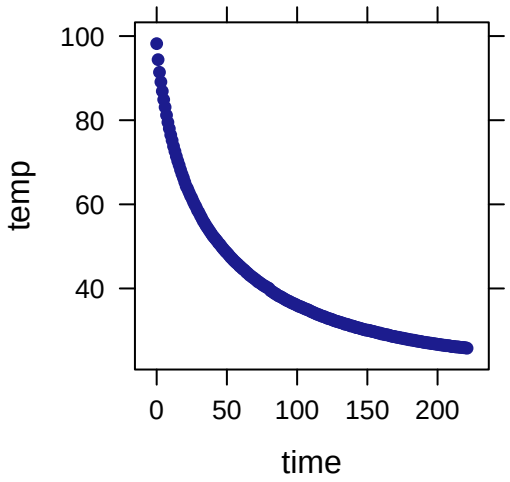
Telephone The fraction of US households with a telephone, as a function of family income. (Source: Susan E. Mayer (1997) *What money can't buy: Family income and children's life chances* Harvard Univ. Press p. 102.)

```
inc = fetchData("Income-Housing.csv")
plotPoints(Telephone~IncomePercentile,data=inc)
```



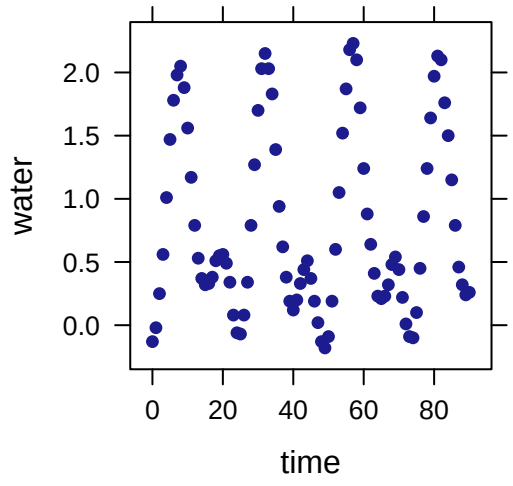
Cooling Water Stan Wagon's measurements of the temperature of cooling water. The water was boiled and poured into an aluminum pot. Time in seconds, temperature in degrees C. (For similar data: Stan Wagon and Robert Portmann, "How quickly does hot water cool?", *Mathematica in Education and Research* **10**:3 (July 2005) 1-9.)

```
Cooling = fetchData("stan-data.csv")
plotPoints( temp ~ time, data=Cooling)
```



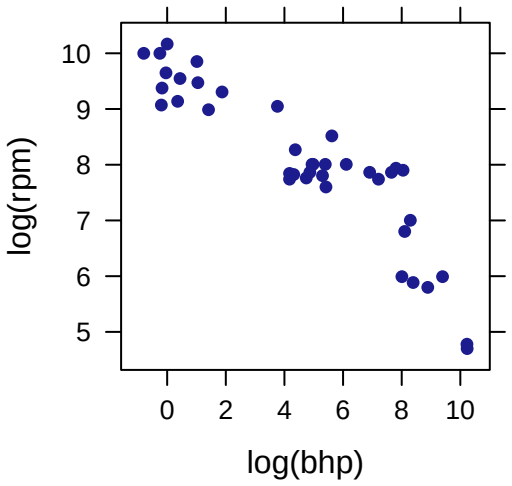
Tides Tidal measurements in Pearl Harbor, Hawaii. (Source: Andrew Beveridge)

```
Hawaii = fetchData("hawaii.csv")
plotPoints(water ~ time, data=Hawaii)
```



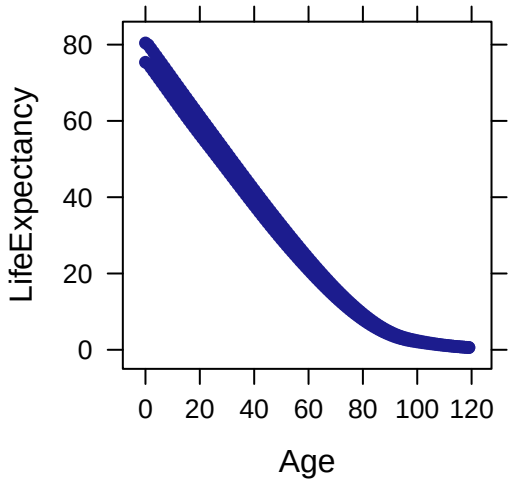
Internal Combustion Engines Data about the size, power, and other characteristics of internal combustion engines. (Source: Thomas McMahon and John Tyler Bonner (1983) *On Size and Life*)

```
Engines = fetchData("engines.csv")
Engines = transform(Engines, rpm=RPM, bhp=BHP)
plotPoints(log(rpm)~log(bhp), data=Engines)
```



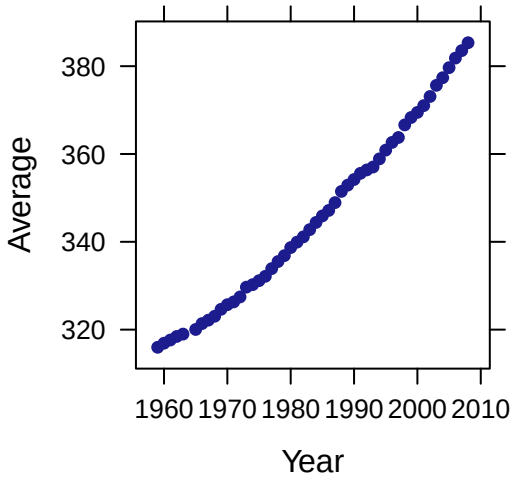
Life Expectancy Residual life expectancy versus age in the US. US Census Bureau.

```
LE = fetchData("LifeExpectancy.csv")
plotPoints(LifeExpectancy ~ Age, data=LE)
```



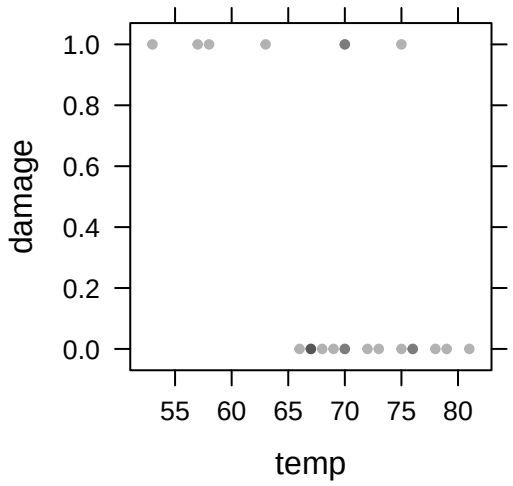
Carbon Dioxide Keeling's record of atmospheric carbon dioxide from Mauna Loa, 1958-2008. `http://cdiac.ornl.gov/trends/co2/sio-mlo.html`

```
CO2 = fetchData("maunaloa-CO2.csv")
CO2 = subset(CO2, Average > 200)
plotPoints( Average ~ Year, data=CO2)
```



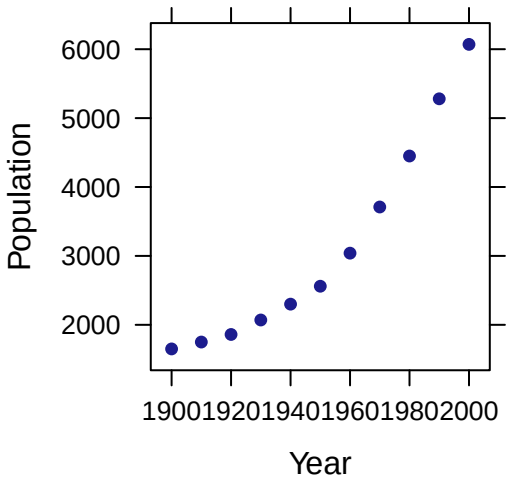
Space Shuttle O-ring Failures The number of O-ring failures versus temperature at the launch site for each Space Shuttle launch up to the 1986 Challenger accident.

```
oring = fetchData("oring-damage.csv")
plotPoints(damage ~ temp, data=oring,
           pch=20, col=rgb(0,0,0,.3))
```



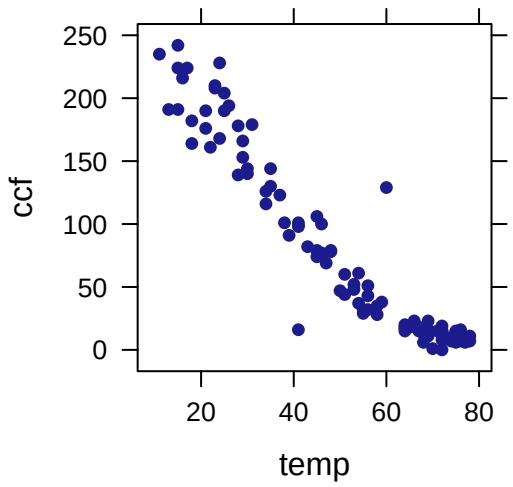
Interpolating World Population Data from a problem in Stewart, giving world population versus year.(Source: Stewart, **Calculus: Concepts and Contexts** 2/e p. 38.

```
worldPop = fetchData("PREP-Stewart-World-Population.csv")
plotPoints(Population ~ Year, data=worldPop )
```



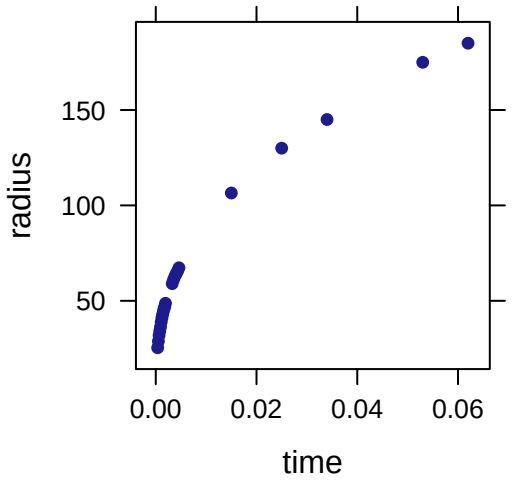
Utility Use Data Month-by-month utility bill data from a single-family house in Saint Paul, Minnesota.

```
utils = fetchData("utilities.csv")
plotPoints(ccf ~ temp, data=utils)
```



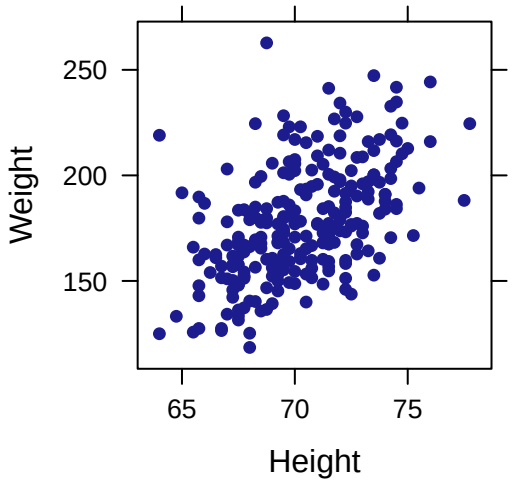
Trinity Test Fireball The radius of the fireball versus time measured at the first atomic bomb test, Trinity, in Alamogordo, New Mexico, on July 16, 1945.

```
blast = fetchData("blastdata.csv")
plotPoints(radius ~ time, data=blast)
```



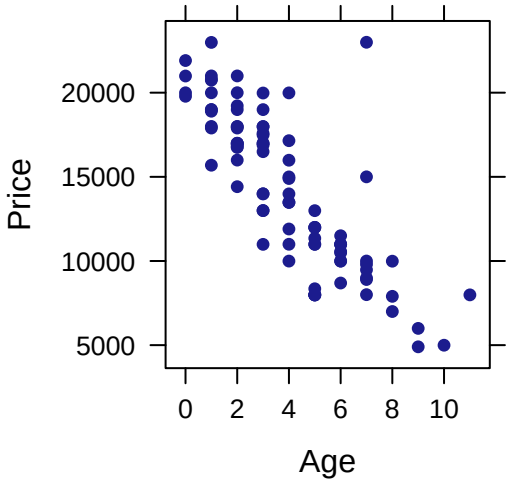
Body Fat Measurements Body circumference measurements for 252 men, along with estimates of the percentage of body fat determined by underwater weighing. (Source: StatLib `http://lib.stat.cmu.edu/datasets/bodyfat`. See also a discussion by Victor Addona `http://www.mosaic-web.org/go/MCAST/materials/Sept-10-2010/Addona_MCAST_Sept_10.pdf`.)

```
body = fetchData("BodyFat.csv")
body = subset(body, Height>60 & Weight<300 )
plotPoints(Weight ~ Height, data=body)
```



Used Car Prices Data on used Honda Accords collected by a student group in Macalester's Math 155 class. (Contact: Daniel Kaplan, Macalester College)

```
cars = fetchData("used-hondas.csv")
plotPoints(Price ~ Age, data=cars)
```



Kepler's Observations of Mars Kepler's measurements of the position of Mars relative to the sun. (Data source: McLaughlin, Michael P. (1999) "A Tutorial on Mathematical Modelling" `http://www.causascientia.org/math_stat/Tutorial.pdf` p. 21-23.) The 'Time' variable is measured as a Julian date; 0 is Greenwich noon, on 1 January 4713 BC. Radii are in astronomical units (AU) — the mean distance from the Earth to the Sun.

```
kdata = fetchData("kepler-mars.csv")
plotPoints( kepler.radius ~ kepler.angle, data=kdata, xlab="Angle",ylab="Radius" )
kdata = transform(kdata, x=kepler.radius*cos(kepler.angle))
kdata = transform(kdata, y=kepler.radius*sin(kepler.angle))
plotPoints(y~x, data=kdata,col="red")
```