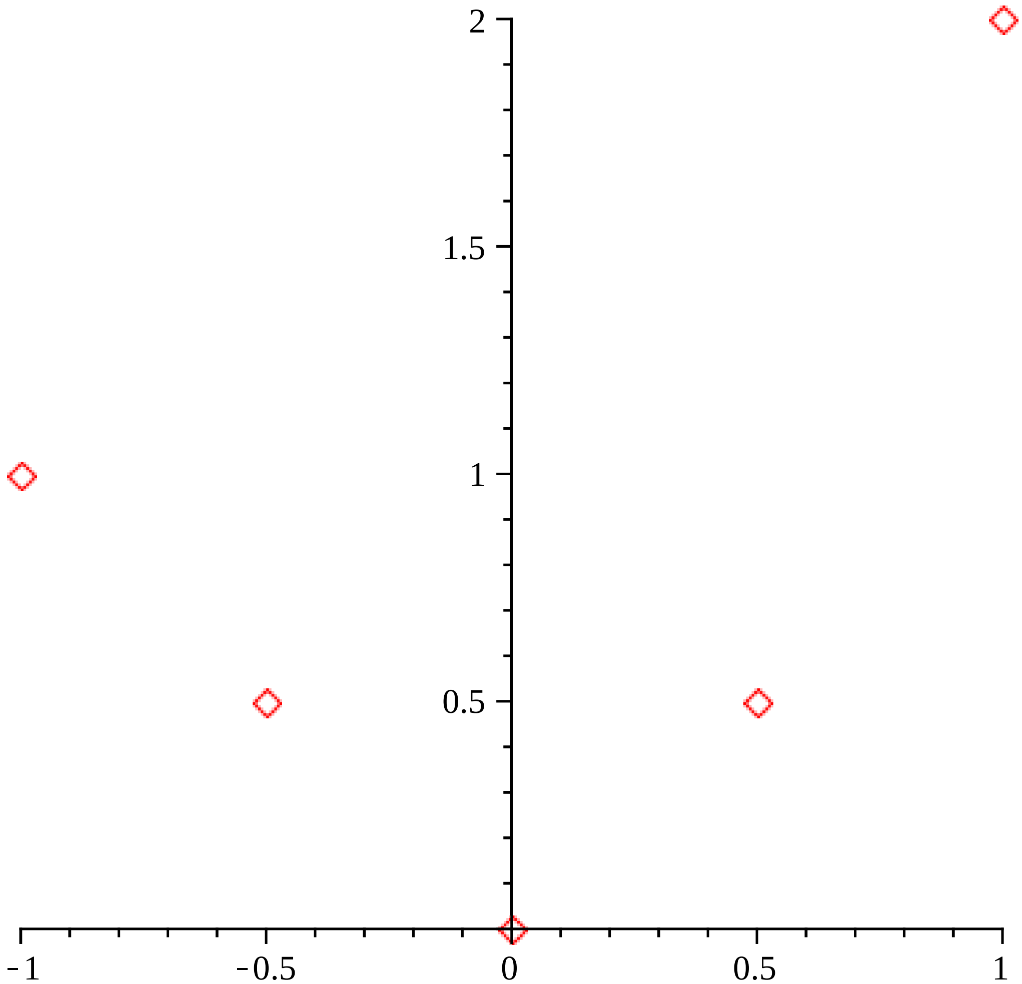
Check the link:

http://www.cse.uiuc.edu/heath/scicomp/notes/chap03.pdf

for a presentation on Linear Least Squares problem

```
> t := Vector([-1.0,-0.5,0,.5,1.0]):
  y := Vector([1.0,.5,0,.5,2.0]):
```

```
> plot( [seq([t[k],y[k]],k=1..5)],style=point,symbolsize=20);
```



```
> A := Matrix(5,3):
  b := Vector(5):
```

```
> for k from 1 to 5 do
     A[k,1] := 1.0:
     A[k,2] := t[k]:
```

```
    A[k,3] := t[k]^2:
    b[k] := y[k]:
end do:
> A, b;
  bb := b:
```

$$\begin{bmatrix} 1.0 & -1.0 & 1.00 \\ 1.0 & -0.5 & 0.25 \\ 1.0 & 0 & 0 \\ 1.0 & 0.5 & 0.25 \\ 1.0 & 1.0 & 1.00 \end{bmatrix}, \begin{bmatrix} 1.0 \\ 0.5 \\ 0 \\ 0.5 \\ 2.0 \end{bmatrix} \quad (1)$$

```
> x := LinearAlgebra:-LeastSquares(A,b);
```

$$x := \begin{bmatrix} 0.0857142857142855208 \\ 0.400000000000000188 \\ 1.42857142857142860 \end{bmatrix} \quad (2)$$

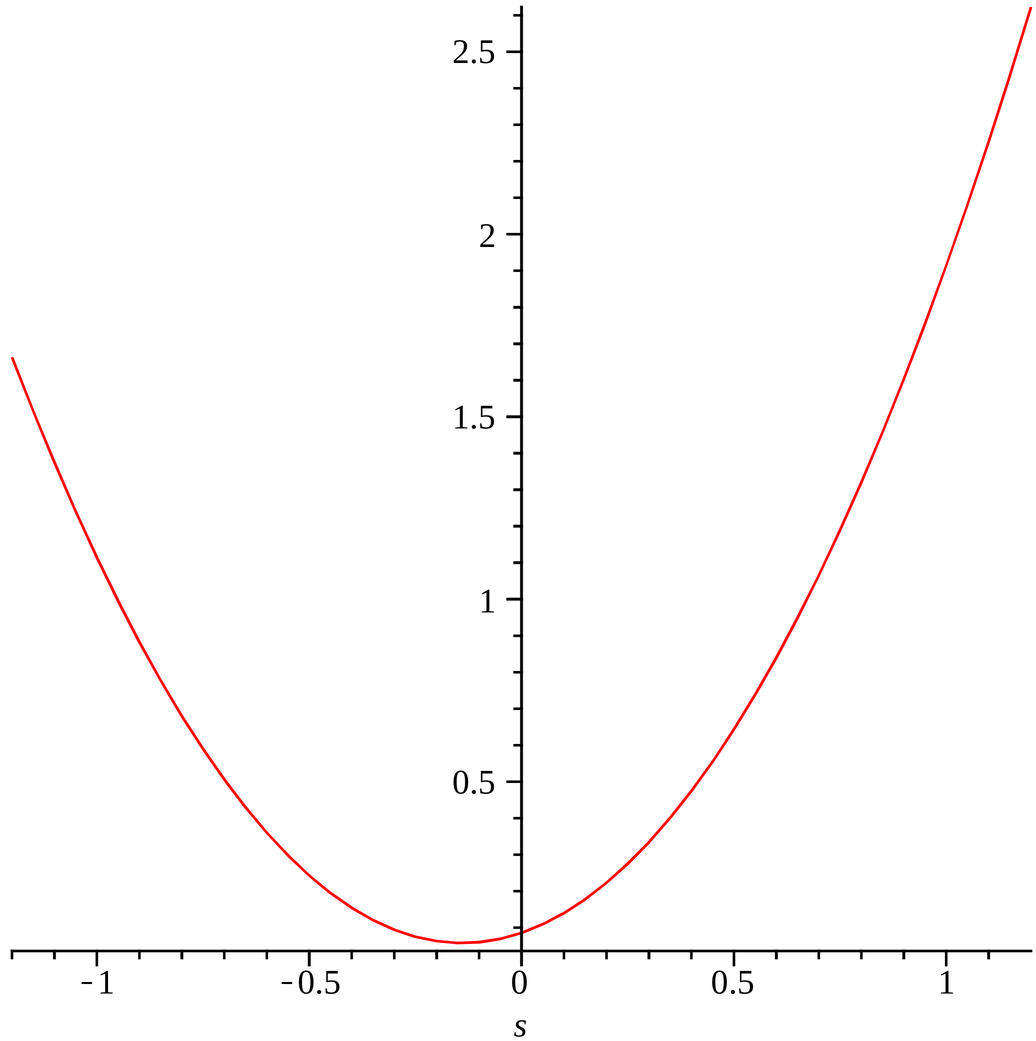
```
> a, b, c := x[1], x[2], x[3];
```

$$a, b, c := 0.0857142857142855208, 0.400000000000000188, 1.42857142857142860 \quad (3)$$

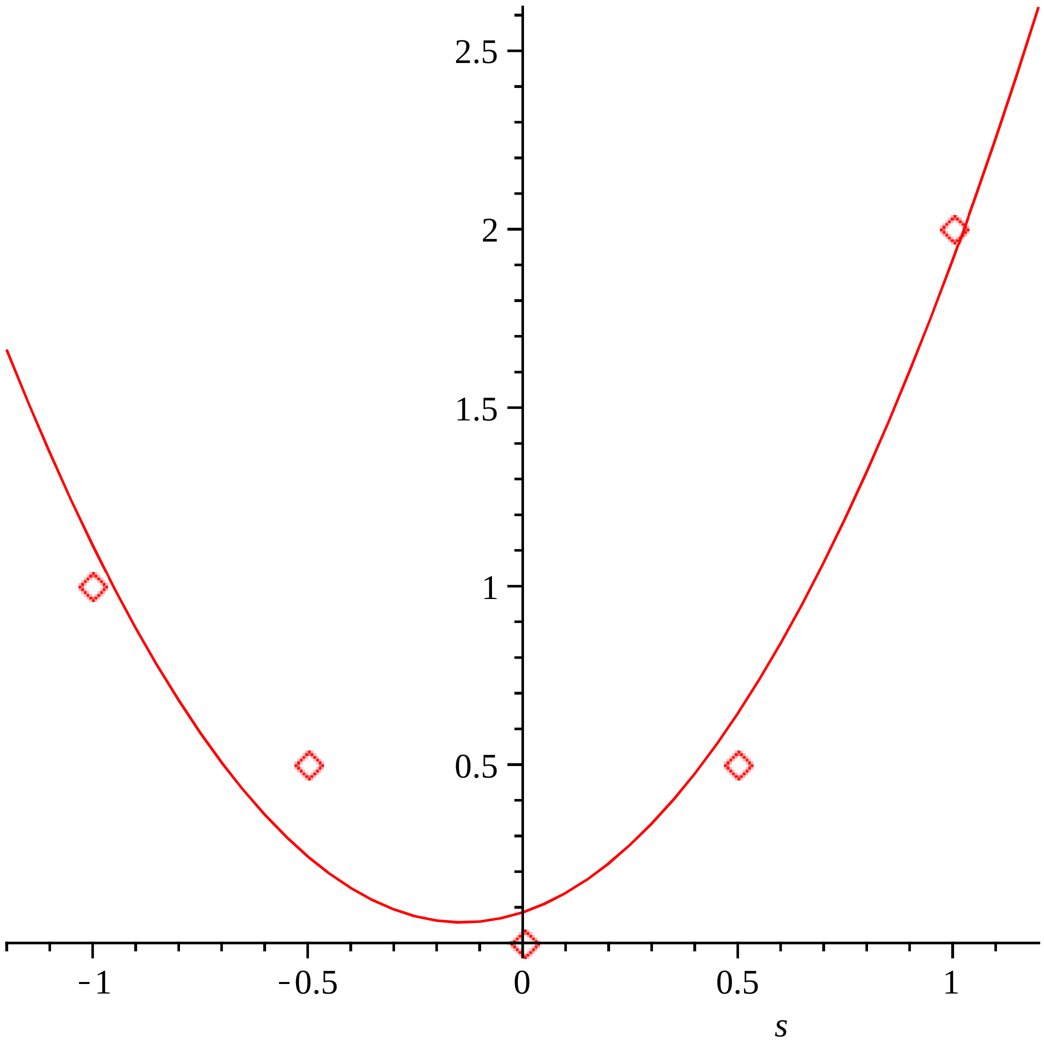
```
> f := t -> a + b*t + c*t^2;
```

$$f := t \rightarrow a + b\,t + c\,t^2 \quad (4)$$

```
> plot(f(s), s = -1.2..1.2);
```




```
> plot1 := plot(f(s), s = -1.2..1.2):
  plot2 := plot( [seq([t[k],y[k]],k=1..5)],style=point,symbolsize=
  20):
> plots[display]({plot1,plot2});
```




```
> b := bb:
  B := LinearAlgebra:-Transpose(A).A:
  d := LinearAlgebra:-Transpose(A).b:
  B, d;
```

$$\begin{bmatrix} 5. & 0. & 2.50000000000000000 \\ 0. & 2.50000000000000000 & 0. \\ 2.50000000000000000 & 0. & 2.12500000000000000 \end{bmatrix}, \begin{bmatrix} 4. \\ 1. \\ 3.25000000000000000 \end{bmatrix} \quad (5)$$

```
> u := LinearAlgebra:-LinearSolve(B,d);
```

$$u := \begin{bmatrix} 0.0857142857142857012 \\ 0.400000000000000022 \\ 1.42857142857142860 \end{bmatrix} \quad (6)$$

```
> u, x;
```

$$\begin{bmatrix} 0.08571428571428857012 \\ 0.400000000000000022 \\ 1.42857142857142860 \end{bmatrix}, \begin{bmatrix} 0.08571428571428855208 \\ 0.400000000000000188 \\ 1.42857142857142860 \end{bmatrix} \quad \mathbf{(7)}$$

Using the QR decomposition:

```
> UseHardwareFloats:= false:
  Digits := 6:
> (Q,R) := LinearAlgebra:-QRDecomposition(A);
```

$$Q, R := \begin{bmatrix} -0.44722 & -0.632458 & 0.534525 \\ -0.447217 & -0.316231 & -0.267260 \\ -0.447217 & -0.000001 & -0.534515 \\ -0.447217 & 0.316225 & -0.267257 \\ -0.447217 & 0.632454 & 0.534520 \end{bmatrix}, \begin{bmatrix} -2.23606 & 0. & -1.11804 \\ 0. & 1.58115 & -0.000004 \\ 0. & 0. & 0.935422 \end{bmatrix} \quad \mathbf{(8)}$$

```
>  d:= LinearAlgebra:-Transpose(Q).b;
```

$$d := \begin{bmatrix} -1.78887 \\ 0.632448 \\ 1.33631 \end{bmatrix} \quad \mathbf{(9)}$$

```
> v := LinearAlgebra:-BackwardSubstitute(R,d);
```

$$v := \begin{bmatrix} 0.0857222 \\ 0.399996 \\ 1.42856 \end{bmatrix} \quad \mathbf{(10)}$$

```
> x, u, v;
```

$$\begin{bmatrix} 0.08571428571428855208 \\ 0.400000000000000188 \\ 1.42857142857142860 \end{bmatrix}, \begin{bmatrix} 0.08571428571428857012 \\ 0.400000000000000022 \\ 1.42857142857142860 \end{bmatrix}, \begin{bmatrix} 0.0857222 \\ 0.399996 \\ 1.42856 \end{bmatrix} \quad \mathbf{(11)}$$

U.S. advertising expenditures in millions

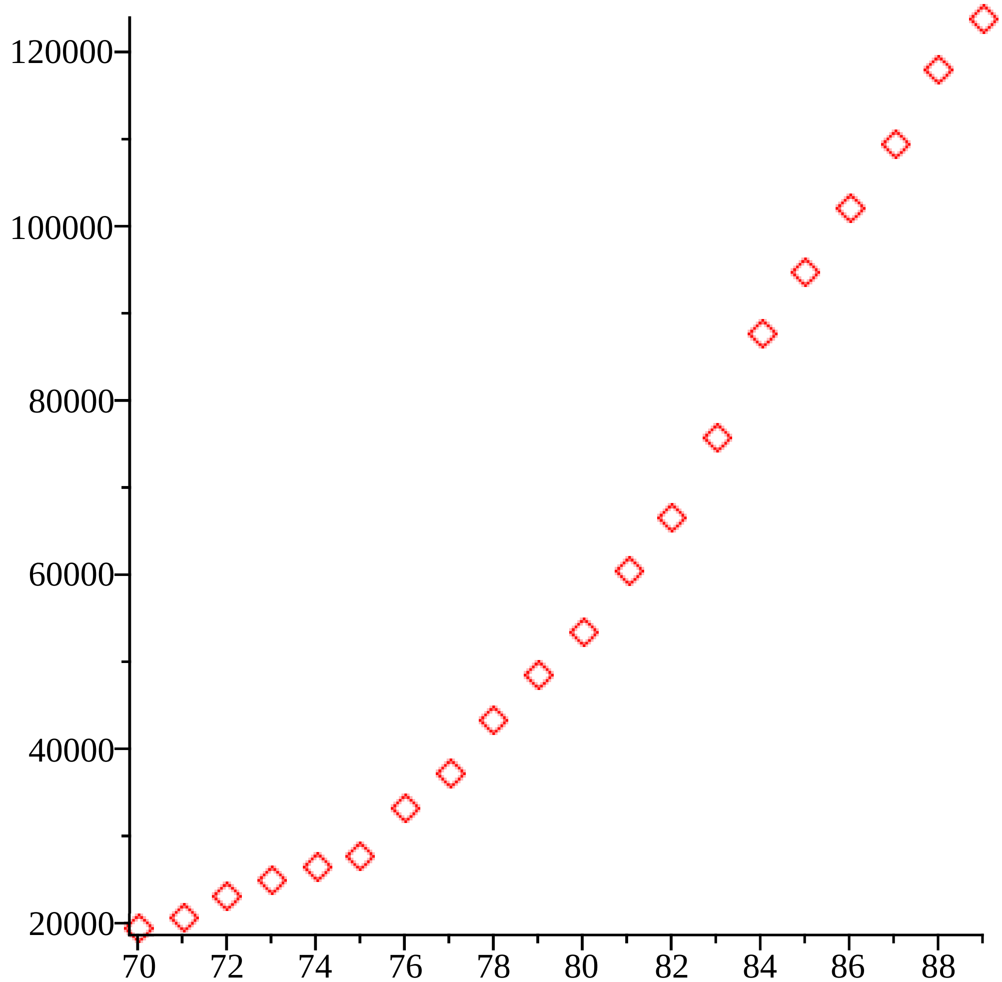
```
> restart:
> x := Vector([seq(k,k=70..89)]):
> y := Vector([19550,20700,23210,24980,26620,27900, 33300, 37400,
```

```
43330, 48780,
              53550,60430,66580,75850,87820,94750,102140,109650,
118050,123930]):
```

```
> x[1..10],y[1..10],"    ",x[11..20],y[11..20];
```

$$\begin{bmatrix}70\\71\\72\\73\\74\\75\\76\\77\\78\\79\end{bmatrix}, \begin{bmatrix}19550\\20700\\23210\\24980\\26620\\27900\\33300\\37400\\43330\\48780\end{bmatrix}, \text{"   "}, \begin{bmatrix}80\\81\\82\\83\\84\\85\\86\\87\\88\\89\end{bmatrix}, \begin{bmatrix}53550\\60430\\66580\\75850\\87820\\94750\\102140\\109650\\118050\\123930\end{bmatrix} \quad \textbf{(12)}$$

```
> plot( [seq([x[k],y[k]],k=1..20)], style=point,symbolsize=20  );
```


120000
100000
80000
60000
40000
20000
70
72
74
76
78
80
82
84
86
88

>

How about $y = a \cdot e^{bx}$

$$\ln y = \ln a + \ln e^{bx}$$

$$\ln y = \ln a + bx \qquad (c = \ln a)$$

use data

x	x_1	x_2	$\cdots$	x_{20}
y	y_1	y_2	$\cdots$	y_{20}

$$\begin{cases} c + bx_1 = \ln y_1 \\ c + bx_2 = \ln y_2 \\ \quad \vdots \\ c + bx_{20} = \ln y_{20} \end{cases}$$

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & \vdots \\ \vdots & \vdots \\ 1 & x_{19} \\ 1 & x_{20} \end{bmatrix} \begin{bmatrix} c \\ b \end{bmatrix} = \begin{bmatrix} \ln y_1 \\ \ln y_2 \\ \vdots \\ \ln y_{20} \end{bmatrix}$$

20×2 system

```
> A := Matrix(20,2):
  d := Vector(20):
> for k from 1 to 20 do  # shift-enter
     A[k,1] := 1.0:
     A[k,2] := x[k]:
     d[k] := evalf(ln(y[k])):
  end do:
```

```
> A[1..10,1..2], d[1..10];
```

$$\begin{bmatrix} 1.0 & 70 \\ 1.0 & 71 \\ 1.0 & 72 \\ 1.0 & 73 \\ 1.0 & 74 \\ 1.0 & 75 \\ 1.0 & 76 \\ 1.0 & 77 \\ 1.0 & 78 \\ 1.0 & 79 \end{bmatrix}, \begin{bmatrix} 9.88073 \\ 9.93789 \\ 10.0523 \\ 10.1258 \\ 10.1894 \\ 10.2364 \\ 10.4133 \\ 10.5294 \\ 10.6766 \\ 10.7951 \end{bmatrix} \quad (13)$$

```
> A[11..20], d[11..20];
```

$$\begin{bmatrix} 1.0 & 80 \\ 1.0 & 81 \\ 1.0 & 82 \\ 1.0 & 83 \\ 1.0 & 84 \\ 1.0 & 85 \\ 1.0 & 86 \\ 1.0 & 87 \\ 1.0 & 88 \\ 1.0 & 89 \end{bmatrix}, \begin{bmatrix} 10.8884 \\ 11.0092 \\ 11.1062 \\ 11.2365 \\ 11.3830 \\ 11.4590 \\ 11.5341 \\ 11.6050 \\ 11.6789 \\ 11.7275 \end{bmatrix} \quad (14)$$

```
> (Q,R) := LinearAlgebra:-QRDecomposition(A);
```

$$Q, R := \begin{bmatrix} \textit{20 x 2 Matrix} \\ \textit{Data Type: sfloat} \\ \textit{Storage: rectangular} \\ \textit{Order: Fortran\_order} \end{bmatrix}, \begin{bmatrix} -4.47214 & -355.537 \\ 0. & 25.7869 \end{bmatrix} \quad (15)$$

```
> g := LinearAlgebra:-Transpose(Q).d;
```

$$g := \begin{bmatrix} -48.4030 \\ 2.72040 \end{bmatrix} \quad (16)$$

```
> w := LinearAlgebra:-LinearSolve(R,g);
```

$$w := \begin{bmatrix} 2.43633 \\ 0.105495 \end{bmatrix} \quad (17)$$

```
> a := exp(w[1]);  b := w[2];
```

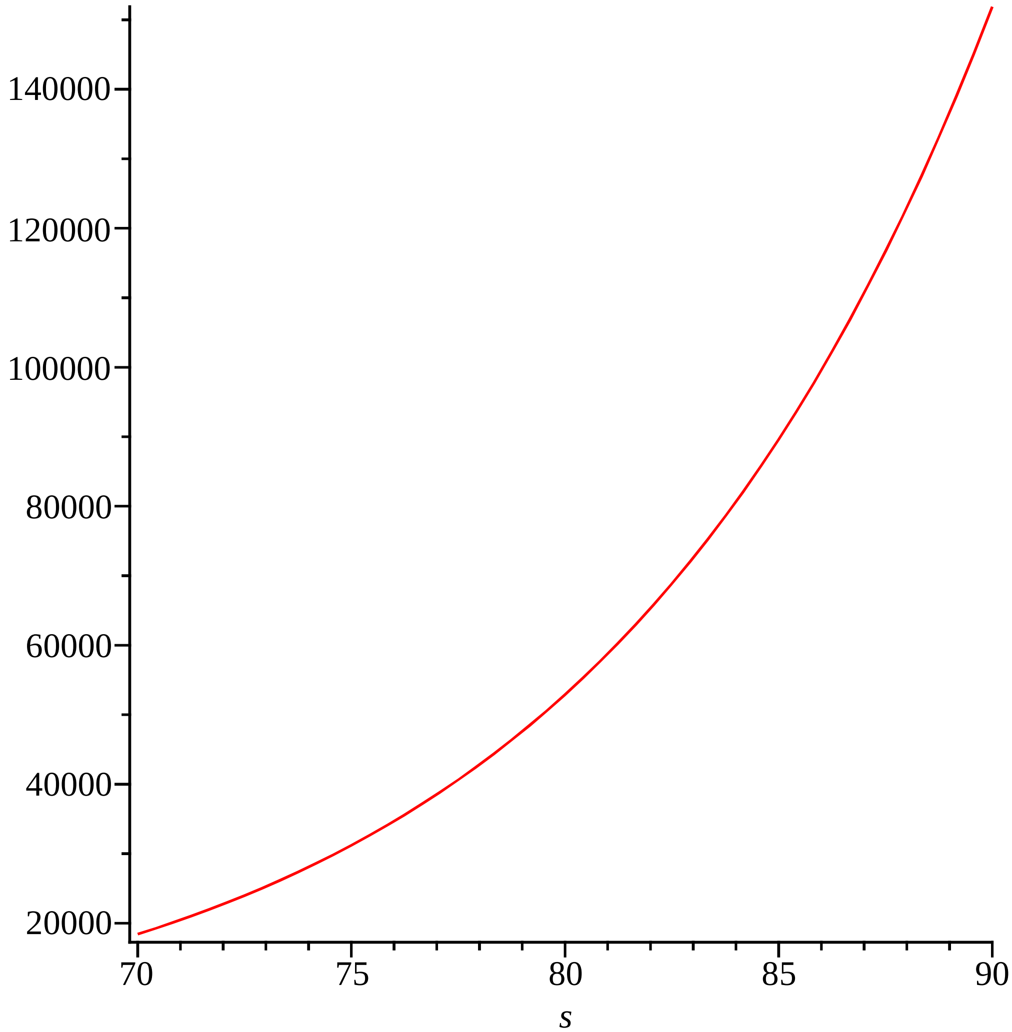
$$a := 11.4310$$

$$b := 0.105495 \qquad (18)$$

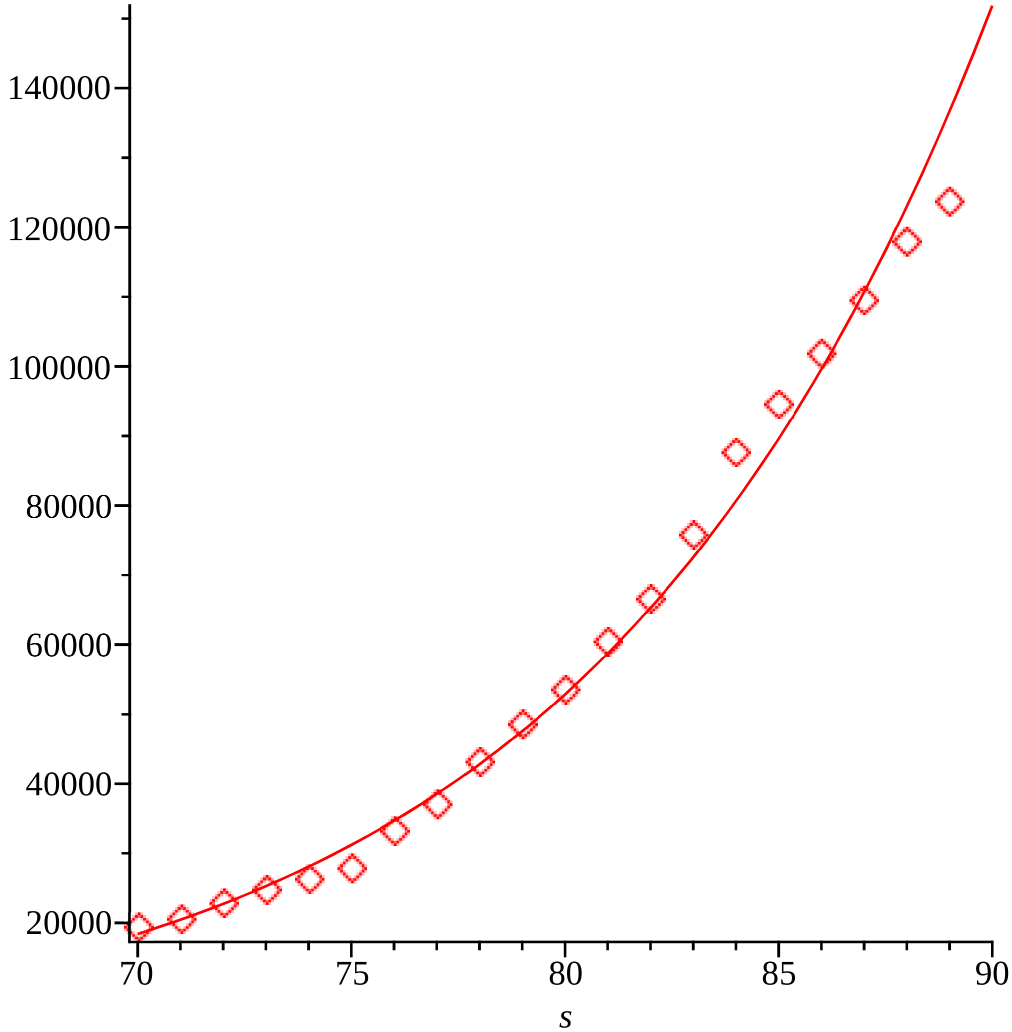
```
> f := t -> a*exp(b*t);
```

$$f := t \rightarrow a\, e^{b\, t} \qquad (19)$$

```
> plot( f(s),s=70..90 );
```




```
> plot1 := plot( f(s),s=70..90 ):
  plot2 := plot( [seq([x[k],y[k]],k=1..20)], style=point,
  symbolsize=20  ):
  plots[display]({plot1,plot2});
```



>

Solve the US ad expenditure problem using

(1) quadratic model
(2) power model $y = a \cdot x^b$ (need to take natural logorithm as well)

put the functions and data into the same graphs

Homework is due April 21, Tuesday (no class on April 16)