CHAPTER 13

13.1 The data can be tabulated as

i	y	$(y_i - \bar{y})^2$
1	8.8	0.725904
2	9.4	0.063504
3	10	0.121104
4	9.8	0.021904
5	10.1	0.200704
6	9.5	0.023104
7	10.1	0.200704
8	10.4	0.559504
9	9.5	0.023104
10	9.5	0.023104
11	9.8	0.021904
12	9.2	0.204304
13	7.9	3.069504
14	8.9	0.565504
15	9.6	0.002704
16	9.4	0.063504
17	11.3	2.715904
18	10.4	0.559504
19	8.8	0.725904
20	10.2	0.300304
21	10	0.121104
22	9.4	0.063504
23	9.8	0.021904
24	10.6	0.898704
25	8.9	0.565504
Σ	**241.3**	**11.8624**

(a) $\bar{y} = \dfrac{241.3}{25} = 9.652$

(b) 9.6

(c) There are three values that occur most frequently: 9.4, 9.5 and 9.8.

(d) range = maximum – minimum = 11.3 – 7.9 = 3.4

(e) $s_y = \sqrt{\dfrac{11.8624}{25-1}} = 0.703041$

(f) $s_y^2 = 0.703041^2 = 0.494267$

(g) $\text{c.v.} = \dfrac{0.703041}{9.652} \times 100\% = 7.28\%$

Here is how the problem would be answered using MATLAB’s built-in functions:

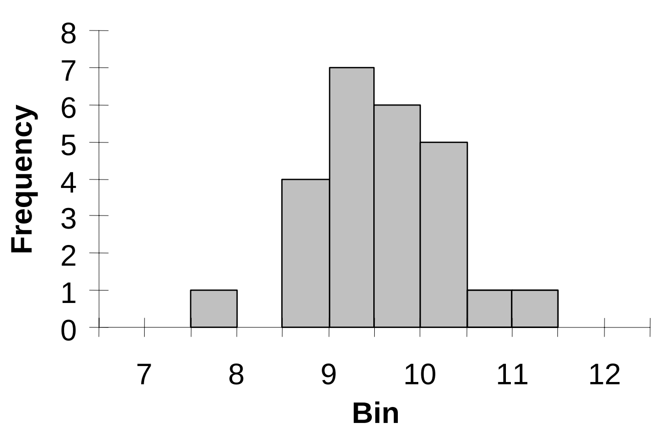
```
>> y=[8.8 9.5 9.8 9.4 10 9.4 10.1 9.2 11.3 9.4 10 10.4 7.9 10.4 9.8 9.8
9.5 8.9 8.8 10.6 10.1 9.5 9.6 10.2 8.9];
```

```
>> m=mean(y)
m =
    9.6520
>> median(y)
ans =
    9.6000
>> mode(y)
ans =
    9.4000
>> range=max(y)-min(y)
range =
    3.4000
>> s=std(y)
s =
    0.7030
>> var(y)
ans =
    0.4943
>> cv=s/m
cv =
    0.0728
```

13.2 The data can be sorted and then grouped. We assume that if a number falls on the border between bins, it is placed in the lower bin.

lower	upper	Frequency
7.5	8	1
8	8.5	0
8.5	9	4
9	9.5	7
9.5	10	6
10	10.5	5
10.5	11	1
11	11.5	1

The histogram can then be constructed as



13.3 The data can be tabulated as

I	y	$(y_i - \bar{y})^2$
1	28.65	0.390625
2	28.65	0.390625
3	27.65	0.140625
4	29.25	1.500625
5	26.55	2.175625
6	29.65	2.640625
7	28.45	0.180625
8	27.65	0.140625
9	26.65	1.890625
10	27.85	0.030625
11	28.65	0.390625
12	28.65	0.390625
13	27.65	0.140625
14	27.05	0.950625
15	28.45	0.180625
16	27.65	0.140625
17	27.35	0.455625
18	28.25	0.050625
19	31.65	13.14063
20	28.55	0.275625
21	28.35	0.105625
22	28.85	0.680625
23	26.35	2.805625
24	27.65	0.140625
25	26.85	1.380625
26	26.75	1.625625
27	27.75	0.075625
28	27.25	0.600625
Σ	**784.7**	**33.0125**

(a) $\bar{y} = \dfrac{784.7}{28} = 28.025$

(b) 27.8

(c) 27.65

(d) range = maximum – minimum = 31.65 – 27.65 = 5.3

(e) $s_y = \sqrt{\dfrac{33.0125}{28-1}} = 1.105751$

(f) $s_y^2 = 1.105751^2 = 1.222685$

(g) $\text{c.v.} = \dfrac{1.105751}{28.025} \times 100\% = 3.95\%$

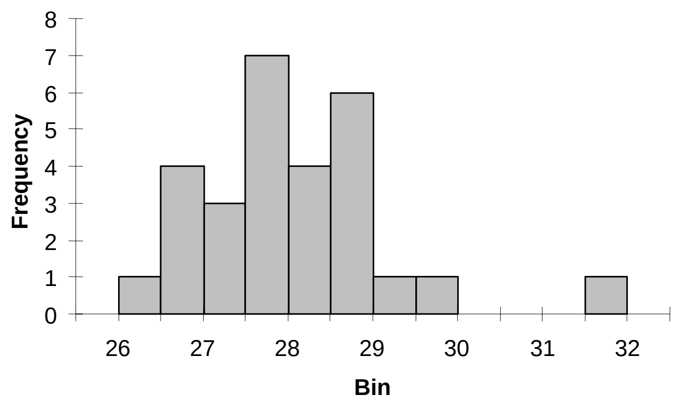
Here is how the problem would be answered using MATLAB's built-in functions:

```
>> y=[28.65 28.65 27.65 29.25 26.55 29.65 28.45 27.65 26.65 27.85 28.65
28.65 27.65 27.05 28.45 27.65 27.35 28.25 31.65 28.55 28.35 28.85 26.35
27.65 26.85 26.75 27.75 27.25];
>> m=mean(y)
m =
   28.0250
>> median(y)
ans =
   27.8000
>> mode(y)
ans =
   27.6500
>> range=max(y)-min(y)
range =
    5.3000
>> s=std(y)
s =
    1.1058
>> var(y)
ans =
    1.2227
>> cv=s/m
cv =
    0.0395
```

(h) The data can be sorted and grouped.

Lower	Upper	Frequency
26	26.5	1
26.5	27	4
27	27.5	3
27.5	28	7
28	28.5	4
28.5	29	6
29	29.5	1
29.5	30	1
30	30.5	0
30.5	31	0
31	31.5	0
31.5	32	1

The histogram can then be constructed as



(i) 68% of the readings should fall between $\bar{y} - s_y$ and $\bar{y} + s_y$. That is, between 28.025 – 1.10575096 = 26.919249 and 28.025 + 1.10575096 = 29.130751. Twenty values fall between these bounds which is equal to 20/28 = 71.4% of the values which is not that far from 68%.

13.4 The sum of the squares of the residuals for this case can be written as

$$S_r = \sum_{i=1}^{n} (y_i - a_1 x_i)^2$$

The partial derivative of this function with respect to the single parameter a_1 can be determined as

$$\frac{\partial S_r}{\partial a_1} = -2\sum\left[(y_i - a_1 x_i)x_i\right]$$

Setting the derivative equal to zero and evaluating the summations gives

$$0 = \sum y_i x_i - a_1 \sum x_i^2$$

which can be solved for

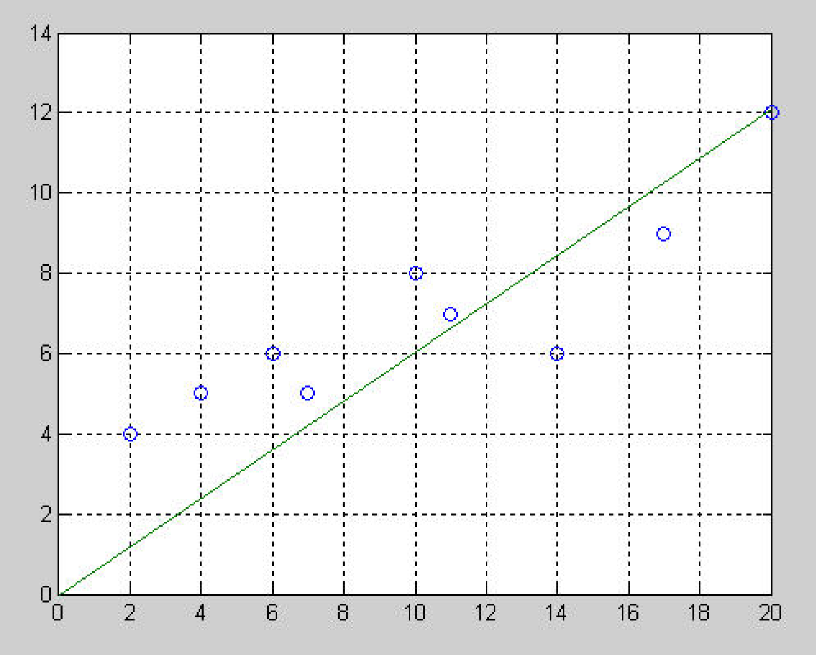
$$a_1 = \frac{\sum y_i x_i}{\sum x_i^2}$$

So the slope that minimizes the sum of the squares of the residuals for a straight line with a zero intercept is merely the ratio of the sum of the dependent variables (y) times the sum of the independent variables (x) over the sum of the independent variables squared (x^2).

The following M-file determines the best-fit slope and plots the resulting line,

```
function [a] = linregrzero(x,y)
% linregrzero: linear regression curve fitting with zero intercept
%   [a, r2] = linregr(x,y): Least squares fit of straight
%             line to data with zero intercept
% input:
```

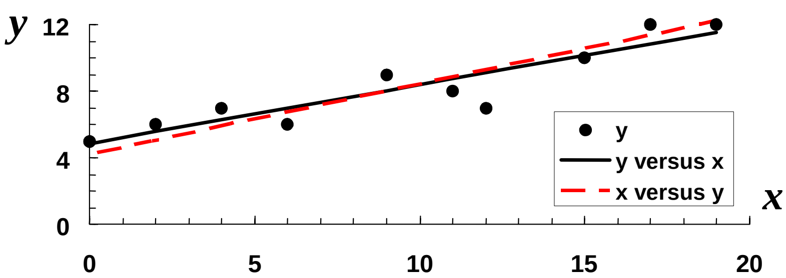
```
%   x = independent variable
%   y = dependent variable
% output:
%   a = slope

n = length(x);
if length(y)~=n, error('x and y must be same length'); end
x = x(:); y = y(:);    % convert to column vectors
sx2 = sum(x.*x); sxy = sum(x.*y);
a = sxy/sx2;
% create plot of data and best fit line
xp = linspace(0,max(x),2);
yp = a*xp;
plot(x,y,'o',xp,yp)
grid on

>> x=[2 4 6 7 10 11 14 17 20];
>> y=[4 5 6 5 8 7 6 9 12];
>> a1=linregrzero(x,y)

a1 =
    0.6053
```



13.5 The results can be summarized as

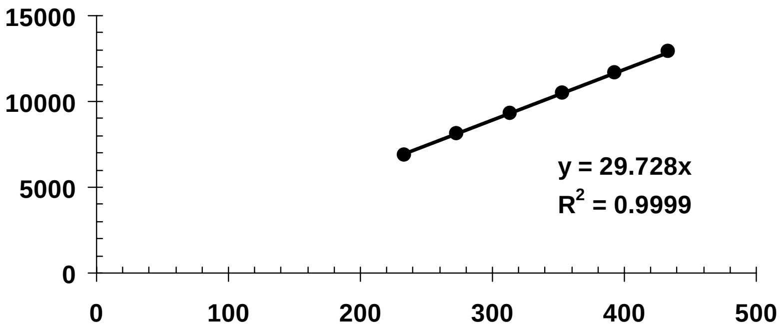
	y versus x	x versus y
Best fit equation	$y = 4.851535 + 0.35247x$	$x = -9.96763 + 2.374101y$
Standard error	1.06501	2.764026
Correlation coefficient	0.914767	0.914767

We can also plot both lines on the same graph



Thus, the “best” fit lines and the standard errors differ. This makes sense because different errors are being minimized depending on our choice of the dependent (ordinate) and independent (abscissa) variables. In contrast, the correlation coefficients are identical since the same amount of uncertainty is explained regardless of how the points are plotted.

13.6 Linear regression with a zero intercept gives [note that T(K) = T(°C) + 273.15].



Thus, the fit is

$$p = 29.728T$$

Using the ideal gas law

$$R = \left(\frac{p}{T}\right)\frac{V}{n}$$

For our fit

$$\frac{p}{T} = 29.728$$

For nitrogen,

$$n = \frac{1\,\text{kg}}{28\,\text{g/mole}}$$

Therefore,

$$R = 29.728\left(\frac{10}{10^3/28}\right) = 8.324$$

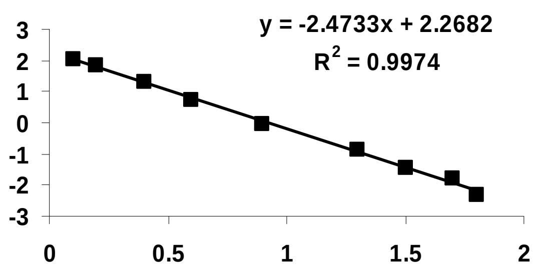
This is close to the standard value of 8.314 J/gmole.

13.7 The function can be linearized by dividing it by x and taking the natural logarithm to yield

$$\ln(y/x) = \ln\alpha_4 + \beta_4 x$$

Therefore, if the model holds, a plot of ln(y/x) versus x should yield a straight line with an intercept of $\ln\alpha_4$ and a slope of β_4.

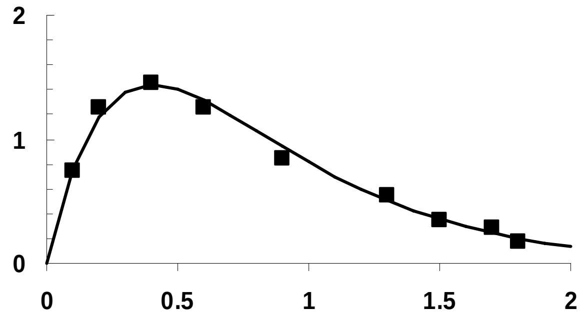
x	*y*	ln(*y*/*x*)
0.1	0.75	2.014903
0.2	1.25	1.832581
0.4	1.45	1.287854
0.6	1.25	0.733969
0.9	0.85	-0.05716
1.3	0.55	-0.8602
1.5	0.35	-1.45529
1.7	0.28	-1.80359
1.8	0.18	-2.30259



Therefore, $\beta_4 = -2.4733$ and $\alpha_4 = e^{2.2682} = 9.661786$, and the fit is

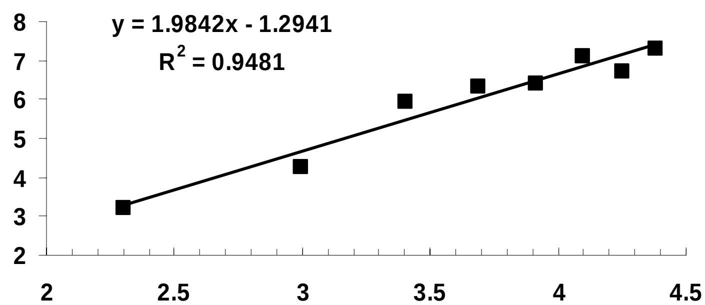
$$y = 9.661786xe^{-2.4733x}$$

This equation can be plotted together with the data:



13.8 The data can be transformed, plotted and fit with a straight line

v, m/s	*F*, N	ln *v*	ln *F*
10	25	2.302585	3.218876
20	70	2.995732	4.248495
30	380	3.401197	5.940171
40	550	3.688879	6.309918
50	610	3.912023	6.413459
60	1220	4.094345	7.106606
70	830	4.248495	6.721426
80	1450	4.382027	7.279319



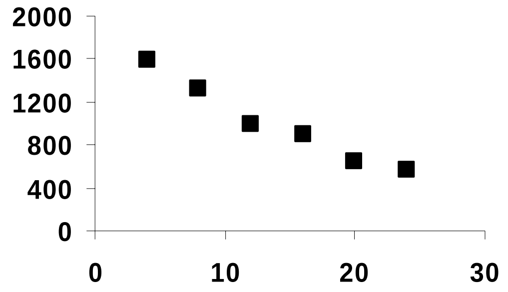
The least-squares fit is

$$\ln y = 1.9842 \ln x - 1.2941$$

The exponent is 1.9842 and the leading coefficient is $e^{-1.2941}$ = 0.274137. Therefore, the result is the same as when we used common or base-10 logarithms:

$$y = 0.274137 x^{1.9842}$$

13.9 (*a*) The data can be plotted

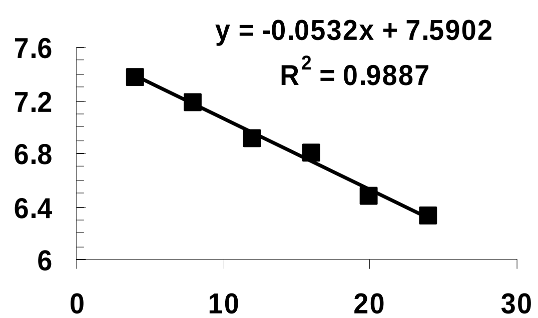


The plot indicates that the data is somewhat curvilinear. An exponential model (i.e., a semi-log plot) is the best choice to linearize the data. This conclusion is based on

- A power model does not result in a linear plot
- Bacterial decay is known to follow an exponential model
- The exponential model by definition will not produce negative values.

The exponential fit can be determined as

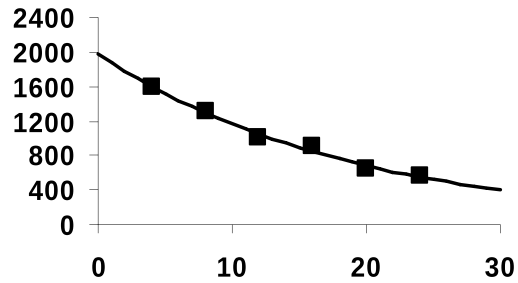
t (hrs)	*c* (CFU/100 mL)	ln *c*
4	1590	7.371489
8	1320	7.185387
12	1000	6.907755
16	900	6.802395
20	650	6.476972
24	560	6.327937



Therefore, the coefficient of the exponent (β_1) is –0.0532 and the lead coefficient (α_1) is $e^{7.5902}$ = 1978.63, and the fit is

$$c = 1978.63e^{-0.0532t}$$

Consequently the concentration at t = 0 is 1978.63 CFU/100 ml. Here is a plot of the fit along with the original data:



(***b***) The time at which the concentration will reach 200 CFU/100 mL can be computed as

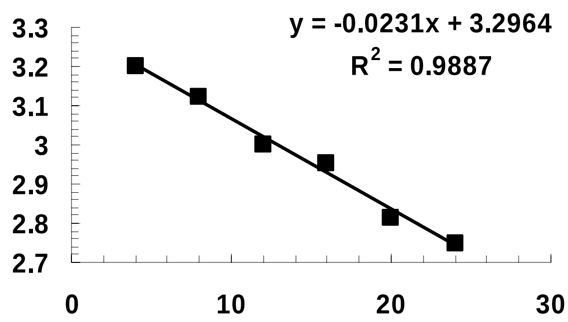
$$200 = 1978.63e^{-0.0532t}$$

$$\ln\left(\frac{200}{1978.63}\right) = -0.0532t$$

$$t = \frac{\ln\left(\frac{200}{1978.63}\right)}{-0.0532} = 43.08\ \text{d}$$

13.10 (a) The exponential fit can be determined with the base-10 logarithm as

t (hrs)	*c* (CFU/100 mL)	log *c*
4	1590	3.201397
8	1320	3.120574
12	1000	3
16	900	2.954243
20	650	2.812913
24	560	2.748188



Therefore, the coefficient of the exponent (β_5) is –0.0231 and the lead coefficient (α_5) is $10^{3.2964}$ = 1978.63, and the fit is

$c = 1978.63(10)^{-0.0231t}$

Consequently the concentration at t = 0 is 1978.63 CFU/100 ml.

(b) The time at which the concentration will reach 200 CFU/100 mL can be computed as

$$200 = 1978.63(10)^{-0.0231t}$$

$$\log_{10}\left(\frac{200}{1978.63}\right) = -0.0231t$$

$$t = \frac{\log_{10}\left(\frac{200}{1978.63}\right)}{-0.0231} = 43.08 \text{ d}$$

Thus, the results are identical to those obtained with the base-*e* model.

The relationship between β_1 and β_5 can be developed as in

$$e^{-\beta_1 t} = 10^{-\beta_5 t}$$

Take the natural log of this equation to yield

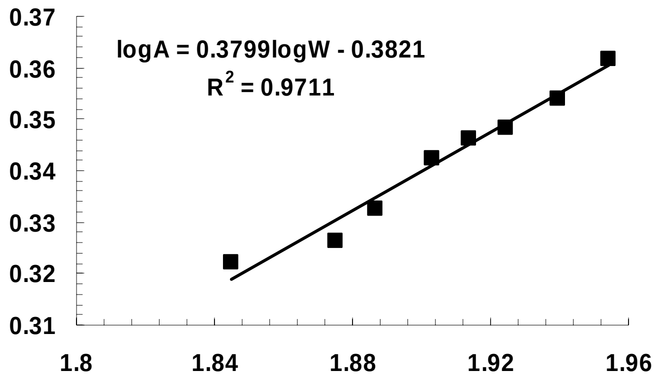
$$-\beta_1 t = -\beta_5 t \ln 10$$

or

$$\beta_1 = 2.302585\beta_5$$

13.11 The power fit can be determined as

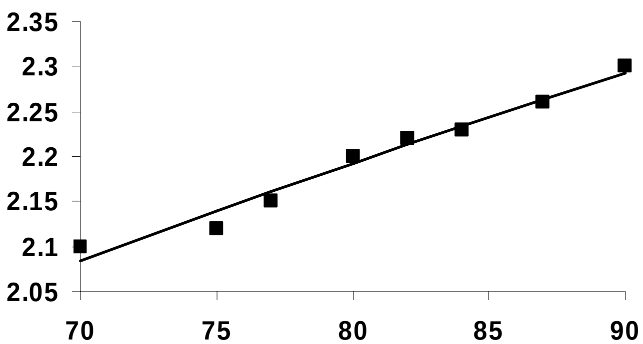
W (kg)	*A* (m^2)	log W	log A
70	2.1	1.845098	0.322219
75	2.12	1.875061	0.326336
77	2.15	1.886491	0.332438
80	2.2	1.90309	0.342423
82	2.22	1.913814	0.346353
84	2.23	1.924279	0.348305
87	2.26	1.939519	0.354108
90	2.3	1.954243	0.361728



Therefore, the power is b = 0.3799 and the lead coefficient is $a = 10^{-0.3821} = 0.4149$, and the fit is

$$A = 0.4149W^{0.3799}$$

Here is a plot of the fit along with the original data:



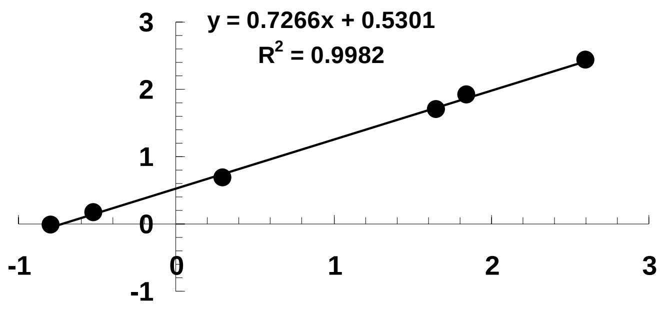
The value of the surface area for a 95-kg person can be estimated as

$$A = 0.4149(95)^{0.3799} = 2.34\ \mathrm{m}^2$$

13.12 A power fit can be determined by taking the common logarithm of the data,

Animal	Mass (kg)	Metabolism (watts)	log(Mass)	log(Met)
Cow	400	270	2.6021	2.4314
Human	70	82	1.8451	1.9138
Sheep	45	50	1.6532	1.6990
Hen	2	4.8	0.3010	0.6812
Rat	0.3	1.45	-0.5229	0.1614
Dove	0.16	0.97	-0.7959	-0.0132

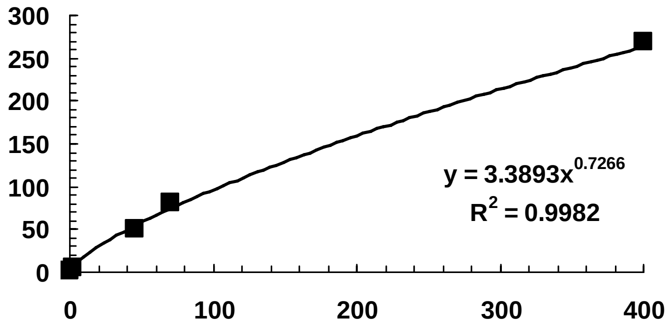
Linear regression gives



Therefore, the power is b = 0.7266 and the lead coefficient is $a = 10^{0.5301}$ = 3.389, and the fit is

$$\text{Metabolism} = 3.389\text{Mass}^{0.7266}$$

Here is a plot of the fit along with the original data:



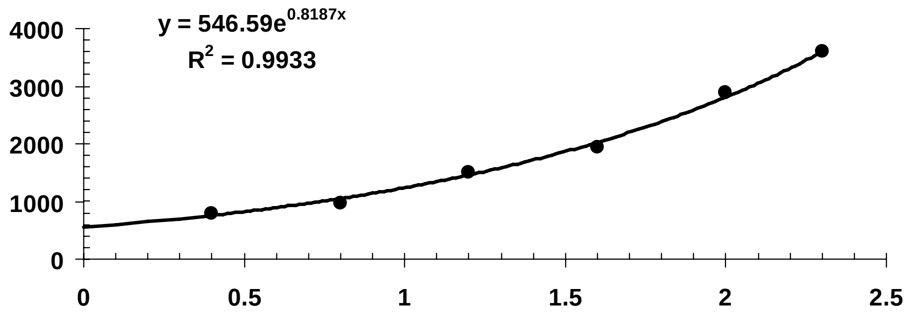
13.13 We regress ln(y) versus x to give

$$\ln y = 6.303701 + 0.818651x$$

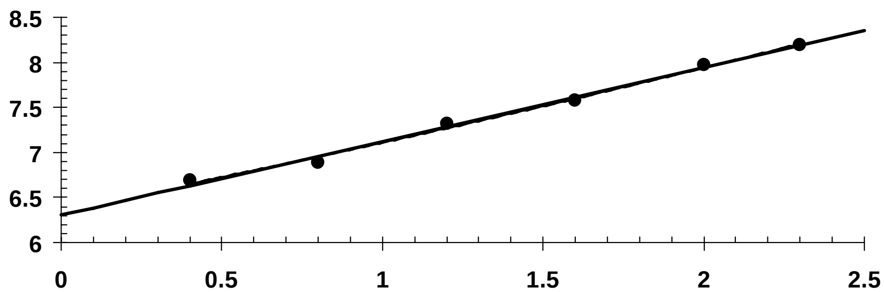
Therefore, $\alpha_1 = e^{6.303701}$ = 546.5909 and β_1 = 0.818651, and the exponential model is

$$y = 546.5909e^{0.818651x}$$

The model and the data can be plotted as



A semi-log plot can be developed by plotting the natural log versus x. As expected, both the data and the best-fit line are linear when plotted in this way.



13.14 The equation can be linearized by inverting it to yield

$$\frac{1}{k} = \frac{c_s}{k_{\max}} \frac{1}{c^2} + \frac{1}{k_{\max}}$$

Consequently, a plot of $1/k$ versus $1/c$ should yield a straight line with an intercept of $1/k_{\max}$ and a slope of $c_s/k_{\max}$

***c*, mg/L**	***k*, /d**	**1/*c*^{2}**	**1/*k***	**1/*c*^{2}×1/*k***	**(1/*c*^{2})^2**
0.5	1.1	4.000000	0.909091	3.636364	16.000000
0.8	2.4	1.562500	0.416667	0.651042	2.441406
1.5	5.3	0.444444	0.188679	0.083857	0.197531
2.5	7.6	0.160000	0.131579	0.021053	0.025600
4	8.9	0.062500	0.112360	0.007022	0.003906
	Sum →	**6.229444**	**1.758375**	**4.399338**	**18.66844**

The slope and the intercept can be computed as

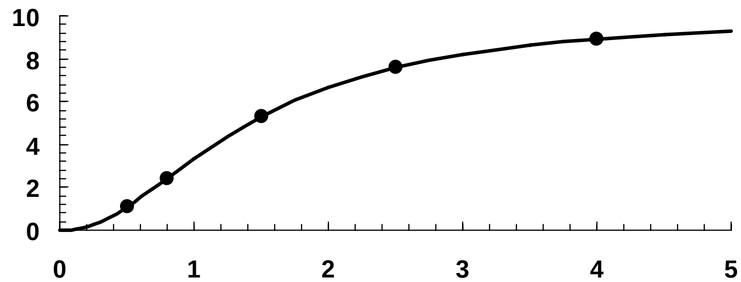
$$a_1 = \frac{5(4.399338) - 6.229444(1.758375)}{5(18.66844) - (6.229444)^2} = 0.202489$$

$$a_0 = \frac{1.758375}{5} - 0.202489 \frac{6.229444}{5} = 0.099396$$

Therefore, k_{max} = 1/0.099396 = 10.06074 and c_s = 10.06074(0.202489) = 2.037189, and the fit is

$$k = \frac{10.06074c^2}{2.037189 + c^2}$$

This equation can be plotted together with the data:



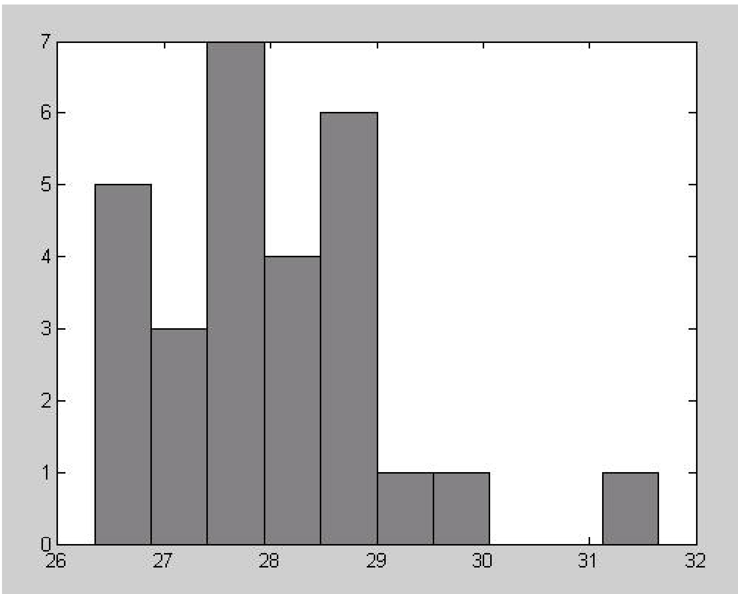
The equation can be used to compute

$$k = \frac{10.06074(2)^2}{2.037189 + (2)^2} = 6.666$$

13.15
```
function stats(x)
% stats: simple descriptive statistics and histogram
%   stats(x): computes simple descriptive statistics
%             and generates a histogram for the values
%             in a vector
% input:
%   x = vector of values
% output:
%   the function does not return any values, but displays the
%   following statistics: number of values, mean, median, mode,
%   range, standard deviation, variance,
%           and coefficient of variation
if length(x)<=1, error('Vector must hold at least 2 values'); end
fprintf('number = %5d\n', length(x))
fprintf('mean = %8.4g\n', mean(x))
fprintf('median = %8.4g\n', median(x))
fprintf('mode = %8.4g\n', mode(x))
fprintf('range = %8.4g\n', max(x)-min(x))
fprintf('standard deviation = %8.4g\n', std(x))
fprintf('variance = %8.4g\n', var(x))
fprintf('coefficient of variation = %8.4g\n', std(x)/mean(x))
hist(x)
```

Application to Prob. 13.3:

```
>> y=[28.65 28.65 27.65 29.25 26.55 29.65 28.45 27.65 26.65 27.85 28.65
28.65 27.65 27.05 28.45 27.65 27.35 28.25 31.65 28.55 28.35 28.85 26.35
27.65 26.85 26.75 27.75 27.25];
>> stats(y)

number =    28
mean =    28.02
median =     27.8
mode =    27.65
range =      5.3
standard deviation =    1.106
variance =    1.223
coefficient of variation =  0.03946
```



13.16

```
function [a, r2, syx] = linregr2(x,y)
% [a, r2, syx] = linregr(x,y):
%   Least squares fit of a straight line to data
%   by solving the normal equations.
% input:
%   x = independent variable
%   y = dependent variable
% output:
%   a = vector of slope, a(1), and intercept, a(2)
%   r2 = coefficient of determination
%   syx = standard error of the estimate

n = length(x);
if length(y)~=n, error('x and y must be same length'); end
x = x(:); y = y(:);    % convert to column vectors
sx = sum(x); sy = sum(y);
sx2 = sum(x.*x); sxy = sum(x.*y); sy2 = sum(y.*y);
a(1) = (n*sxy-sx*sy)/(n*sx2-sx^2);
a(2) = sy/n-a(1)*sx/n;
```

```
syx=sqrt(sum((y-a(1)*x-a(2)).^2)/(n-2));
r2 = ((n*sxy-sx*sy)/sqrt(n*sx2-sx^2)/sqrt(n*sy2-sy^2))^2;
% create plot of data and best fit line
xp = linspace(min(x),max(x),2);
yp = a(1)*xp+a(2);
res=a(1)*x+a(2)-y;
subplot(2,1,1);plot(x,y,'o',xp,yp)
xlabel('x'),ylabel('y'),title('Linear Least-Squares Fit')
subplot(2,1,2);plot(x,res,'o',x,res)
xlabel('x'),ylabel('residual'),title('Residual Plot')
grid on
```

Test application to data from Examples 13.2 and 13.3,

```
>> x=[10 20 30 40 50 60 70 80];
>> y=[25 70 380 550 610 1220 830 1450];
>> [a, r2, syx] = linregr2(x,y)

a =
   19.4702 -234.2857

r2 =
    0.8805

syx =
  189.7885
```

13.17

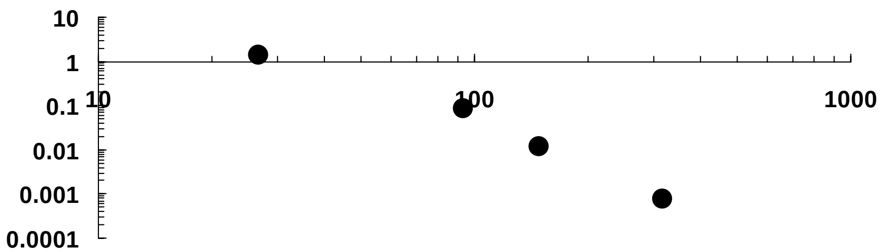
```
function [b, r2] = powerfit(x,y)
% linregr: linear regression curve fitting
%   [b, r2] = powerfit(x,y): Least squares fit of straight
%            line to log-transformed data
% input:
%   x = independent variable
%   y = dependent variable
% output:
%   b = vector of coefficient, b(1), and exponent, b(2)
%   r2 = coefficient of determination

n = length(x);
if length(y)~=n, error('x and y must be same length'); end
x = x(:); y = y(:);
xl=log10(x);yl=log10(y);
sx = sum(xl); sy = sum(yl);
sx2 = sum(xl.*xl); sxy = sum(xl.*yl); sy2 = sum(yl.*yl);
% compute slope and intercept
a(1) = (n*sxy-sx*sy)/(n*sx2-sx^2);
a(2) = sy/n-a(1)*sx/n;
% compute coefficient and exponent
b(1) = 10^a(2);
b(2) = a(1);
St=sum((mean(y)-y).^2)
ypred = b(1)*x.^b(2);
Sr=sum((ypred-y).^2)
r2 = (St - Sr)/St;
```

```
% create plots of data and best fit line
xp = linspace(min(x),max(x));
yp = b(1)*xp.^b(2);
subplot(2,1,1);plot(x,y,'o',xp,yp)
plot(x,y,'o',xp,yp)
xlabel('x'),ylabel('y'),title('Fit (untransformed)')
subplot(2,1,2);loglog(x,y,'o',xp,yp)
xlabel('log(x)'),ylabel('log(y)'),title('Fit (transformed)')
grid on
```

Application to Prob. 13.12.

```
>> mass=[400 70 45 2 0.3 0.16];
>> metabolism=[270 82 50 4.8 1.45 0.97];
>> [b,r2]=powerfit(mass,metabolism)

b =
    3.3893    0.7266
r2 =
    0.9978
```

13.18 A log-log plot of μ versus T suggests a linear relationship.



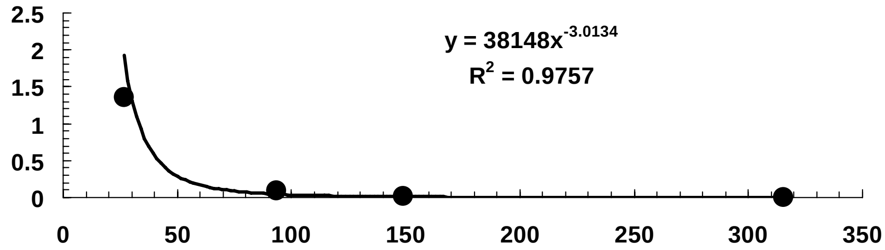
We regress $\log_{10}\mu$ versus $\log_{10}T$ to give

$$\log_{10} \mu = 4.581471 - 3.01338 \log_{10} T \qquad (r^2 = 0.975703)$$

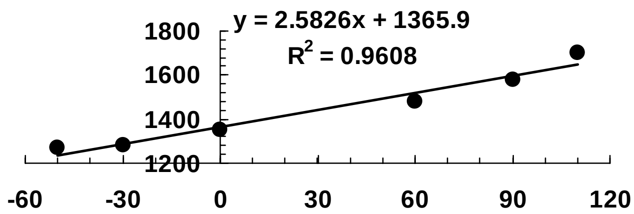
Therefore, $\alpha_2 = 10^{4.581471} = 38{,}147.94$ and $\beta_2 = -3.01338$, and the power model is

$$\mu = 38{,}147.94 T^{-3.01338}$$

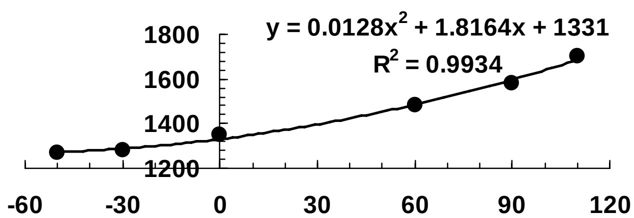
The model and the data can be plotted on untransformed scales as



13.19 We can first try a linear fit

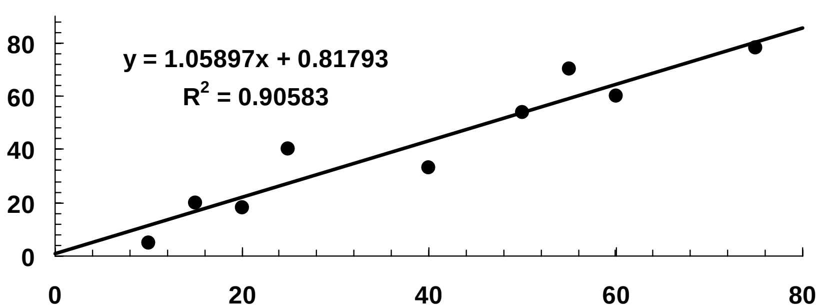


As shown, the fit line is somewhat lacking. Therefore, we can use polynomial regression to fit a parabola



This fit seems adequate in that it captures the general trend of the data. Note that a slightly better fit can be attained with a cubic polynomial, but the improvement is marginal.

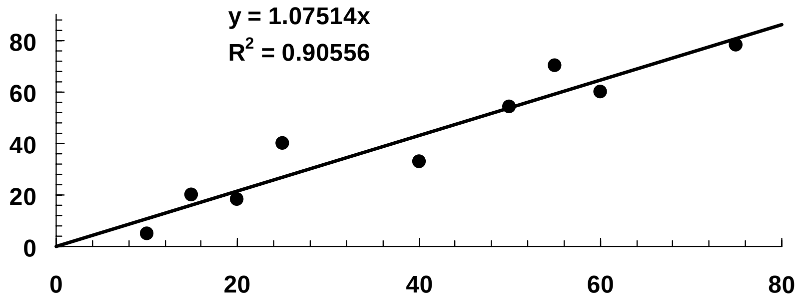
13.20 (a) The linear fit is



The tensile strength at t = 32 can be computed as

$y = 1.05897(32) + 0.81793 = 34.7048913$

(b) A straight line with zero intercept can be fit as



For this case, the tensile strength at t = 32 can be computed as

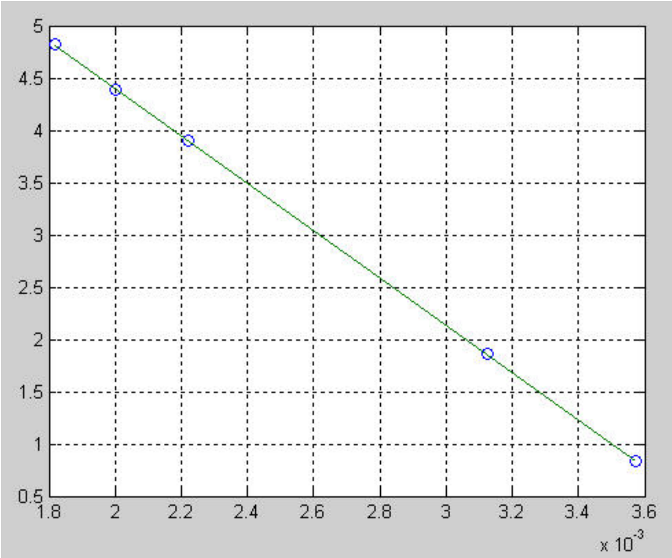
$y = 1.07514(32) = 34.40452$

13.21 The equation can be linearized,

$$\ln\left(\frac{-dA/dt}{A}\right) = \ln k_{01} - \frac{E_1}{RT}$$

This indicates that a plot of ln(–(dA/dt)/A) versus 1/T should have an intercept of ln k_{01} and a slope of E_1/R.

```
>> dAdt=[460 960 2485 1600 1245];
>> A=[200 150 50 20 10];
>> T=[280 320 450 500 550];
>> y=log(dAdt./A);
>> TI=1./T;
>> [a,r2]=linregr(TI,y)

a =
  1.0e+003 *
   -2.2683    0.0089
r2 =
    0.9999
```




```
>> k01=exp(a(2))

k01 =
  7.6205e+003
>> E1=-a(1)*0.00198
E1 =
    4.4913
```

Therefore, k_{01} = 7,620.5 and E_1 = 4.4913.

13.22 The standard errors can be computed via Eq. 17.9

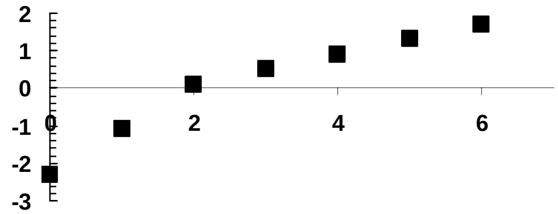
$$s_{y/x} = \sqrt{\frac{S_r}{n-p}}$$

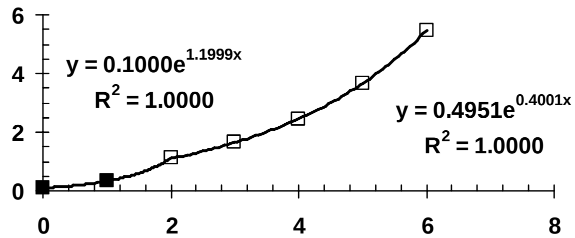
n = 15

	Model A	Model B	Model C
S_r	135	105	100
Number of model parameters fit (p)	2	3	5
$s_{y/x}$	3.222517	2.95804	3.162278

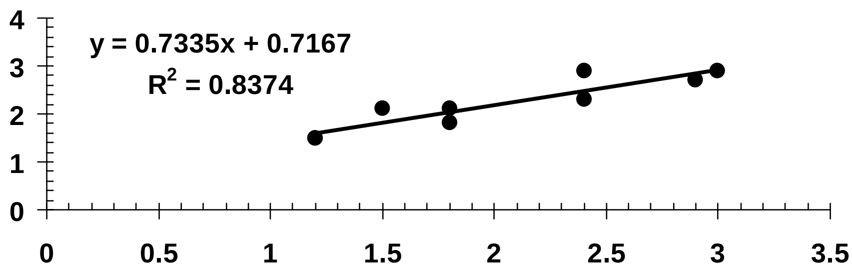
Thus, Model B seems best because its standard error is lower.

13.23 A plot of the natural log of cells versus time indicates two straight lines with a sharp break at 2. Each range can be fit separately with the exponential model as shown in the second plot.





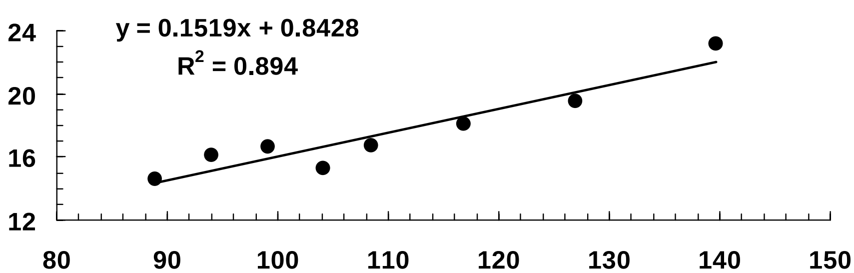
13.24 (a) and (b) Simple linear regression can be applied to yield the following fit



(c) The minimum lane width corresponding to a bike-car distance of 2 m can be computed as

$$y = 0.7335(2) + 0.7167 = 2.1837 \text{ m}$$

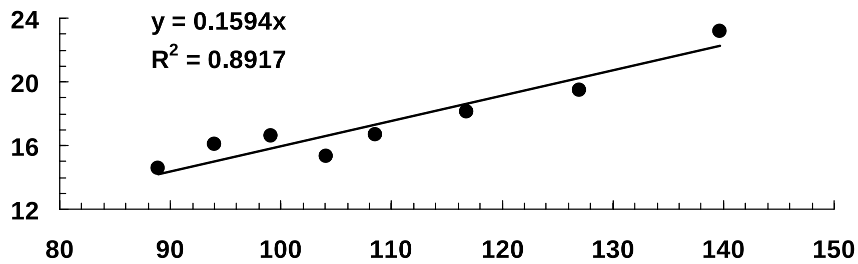
13.25 (a) and (b) Simple linear regression can be applied to yield the following fit



(c) The flow corresponding to the precipitation of 120 cm can be computed as

$Q = 0.1519(120) + 0.8428 = 19.067$

(d) We can redo the regression, but with a zero intercept



Thus, the model is

$$Q = 0.1594P$$

where Q = flow and P = precipitation. Now, if there are no water losses, the maximum flow, Q_m, that could occur for a level of precipitation should be equal to the product of the annual precipitation and the drainage area. This is expressed by the following equation.

$$Q_m = A(\text{km}^2)P\left(\frac{\text{cm}}{\text{yr}}\right)$$

For an area of 1100 km^2 and applying conversions so that the flow has units of m^3/s

$$Q_m = 1{,}100\ \text{km}^2 P\left(\frac{\text{cm}}{\text{yr}}\right)\frac{10^6\,\text{m}^2}{\text{km}^2}\frac{1\,\text{m}}{100\,\text{cm}}\frac{\text{d}}{86{,}400\,\text{s}}\frac{\text{yr}}{365\,\text{d}}$$

Collecting terms gives

$$Q_m = 0.348808P$$

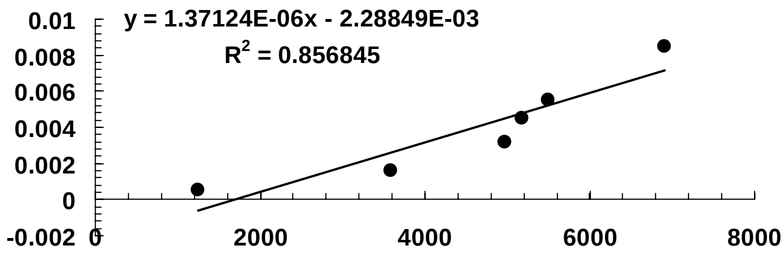
Using the slope from the linear regression with zero intercept, we can compute the fraction of the total flow that is lost to evaporation and other consumptive uses can be computed as

$$F = \frac{0.348808 - 0.1594}{0.348808} = 0.543$$

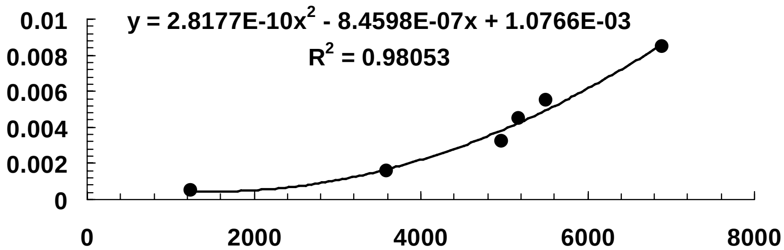
13.26 First, we can determine the stress

$$\sigma = \frac{25000}{10.65} = 2{,}347.418$$

We can then try to fit the data to obtain a mathematical relationship between strain and stress. First, we can try linear regression:



This is not a particularly good fit as the r^2 is relatively low. We therefore try a best-fit parabola,



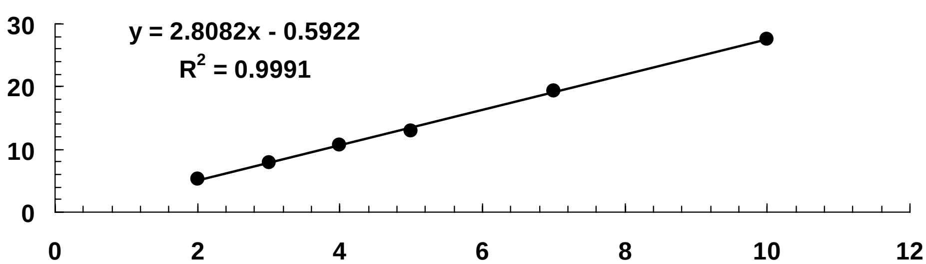
We can use this model to compute the strain as

$$\varepsilon = 2.8177\times10^{-10}(2347.4178)^2 - 8.4598\times10^{-7}(2347.4178) + 1.0766\times10^{-3} = 6.4341\times10^{-4}$$

The deflection can be computed as

$$\Delta L = 6.4341\times10^{-4}(9) = 0.0057907 \text{ m}$$

13.27 (a) The linear fit is

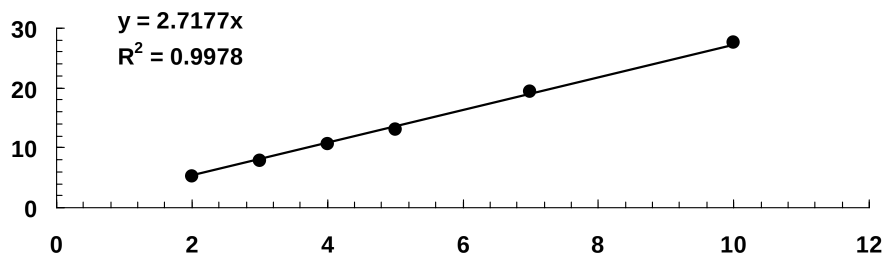


The current for a voltage of 3.5 V can be computed as

$$y = 2.8082(3.5) - 0.5922 = 9.2364$$

Both the graph and the r^2 indicate that the fit is good.

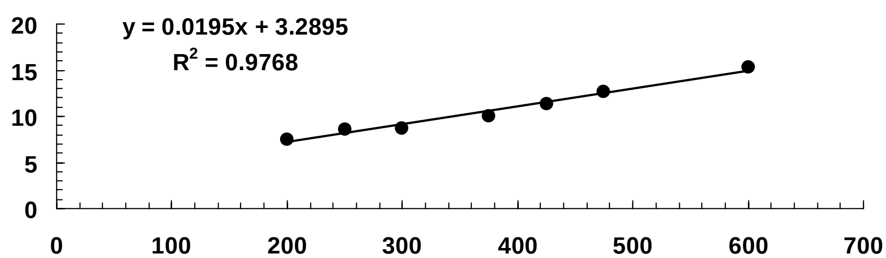
(b) A straight line with zero intercept can be fit as



For this case, the current at V = 3.5 can be computed as

$$y = 2.7177(3.5) = 9.512$$

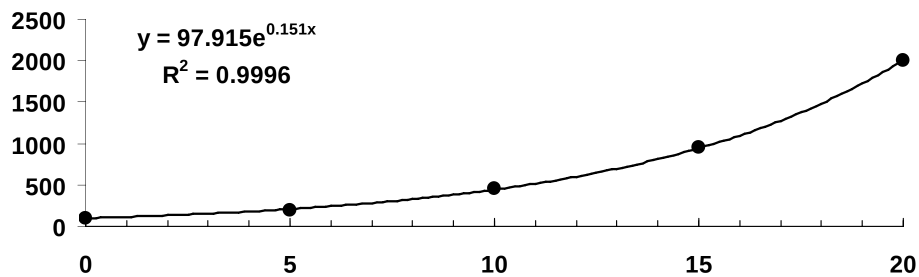
13.28 Linear regression yields



The percent elongation for a temperature of 400 can be computed as

$$\%\,\text{elongation} = 0.0195(400) + 3.2895 = 11.072$$

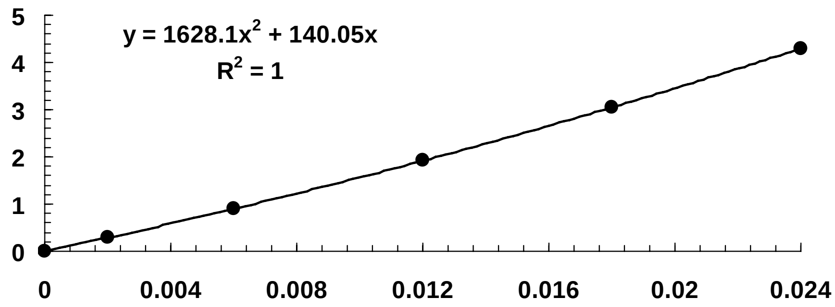
13.29 The fit of the exponential model is



The model can be used to predict the population 5 years in the future as

$$p = 97.91484e^{0.150992(25)} = 4268$$

13.30 We fit a number of curves to this data and obtained the best fit with a second-order polynomial with zero intercept



Therefore, the best-fit curve is

$$u = 1628.1y^2 + 140.05y$$

We can differentiate this function

$$\frac{du}{dy} = 3256.2y + 140.05$$

Therefore, the derivative at the surface is 140.05 and the shear stress can be computed as $1.8\times10^{-5}(140.05) = 0.002521$ N/m^2.

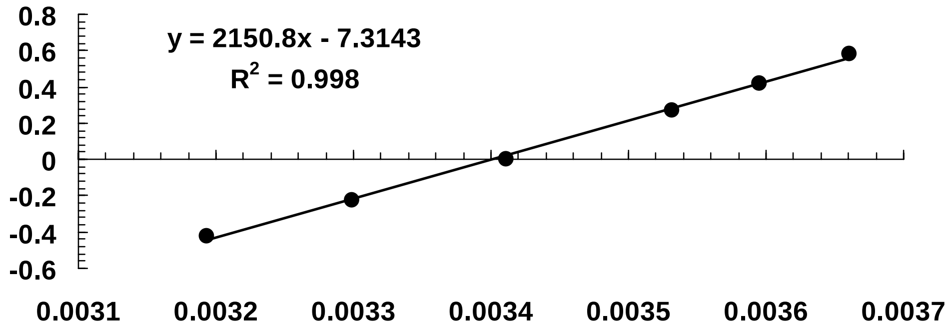
13.31 We can use transformations to linearize the model as

$$\ln\mu = \ln D + B\frac{1}{T_a}$$

Thus, we can plot the natural log of μ versus $1/T_a$ and use linear regression to determine the parameters. Here is the data showing the transformations.

T	μ	T_a	$1/T_a$	$\ln\mu$
0	1.787	273.15	0.003661	0.580538
5	1.519	278.15	0.003595	0.418052
10	1.307	283.15	0.003532	0.267734
20	1.002	293.15	0.003411	0.001998
30	0.7975	303.15	0.003299	-0.22627
40	0.6529	313.15	0.003193	-0.42633

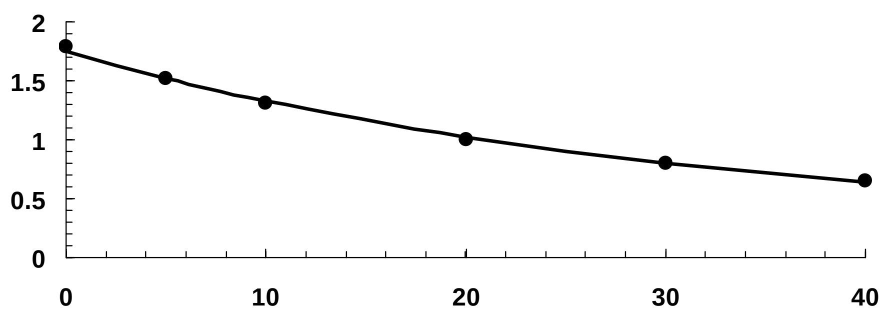
Here is the fit:



Thus, the parameters are estimated as $D = e^{-7.3143} = 6.65941\times10^{-4}$ and $B = 2150.8$, and the Andrade equation is

$$\mu = 6.65941\times10^{-4} e^{2150.8/Ta}$$

This equation can be plotted along with the data



Note that this model can also be fit with nonlinear regression. If this is done, the result is

$$\mu = 5.39872\times10^{-4} e^{2210.66/Ta}$$

Although it is difficult to discern graphically, this fit is slightly superior ($r^2 = 0.99816$) to that obtained with the transformed model ($r^2 = 0.99757$).

13.5

I	x_i	y_i	x_i^2	x_iy_i
1	0	9.8100	0	0
2	20000	9.7487	4.0E+08	194974
3	40000	9.6879	1.6E+09	387516
4	60000	9.6278	3.6E+09	577668
5	80000	9.5682	6.4E+09	765456
Σ	**200000**	**48.4426**	**1.2E+10**	**1925614**

$$a_1 = \frac{5(1{,}925{,}614) - 200{,}000(48.4426)}{5(1.2\times 10^{10}) - 200{,}000^2} = -3.0225\times 10^{-6}$$

$$a_0 = \frac{48.4426}{5} - 3.0225\times 10^{-6}\,\frac{200{,}000}{5} = 9.80942$$

Therefore, the line of best fit is (using the nomenclature of the problem)

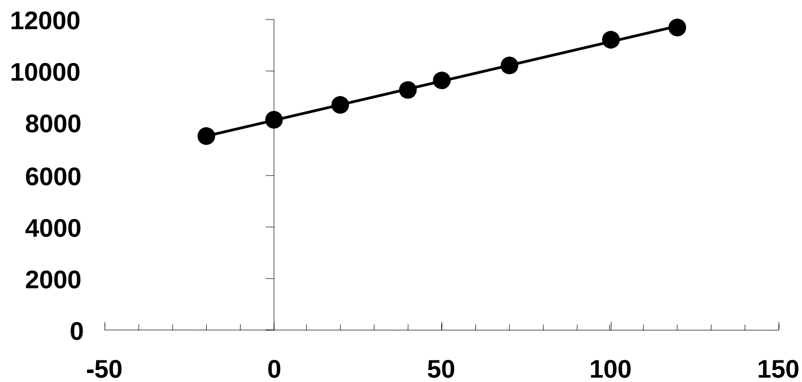
$$g = 9.80942 - 3.0225\times 10^{-6}\,y$$

The value at 55,000 m can therefore be computed as

$$g = 9.80942 - 3.0225\times 10^{-6}(55{,}000) = 9.6431825$$

13.6 Regression gives

$$p = 8100.47 + 30.3164T \qquad (r^2 = 0.999)$$



$$R = \left(\frac{p}{T}\right)\frac{V}{n}$$

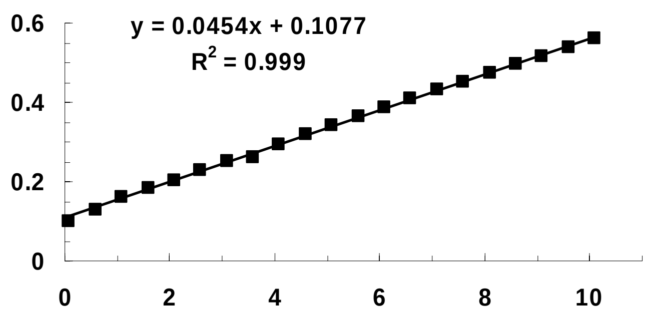
$$\frac{p}{T} = 30.3164$$

$$n = \frac{1\,\text{kg}}{28\,\text{g/mole}}$$

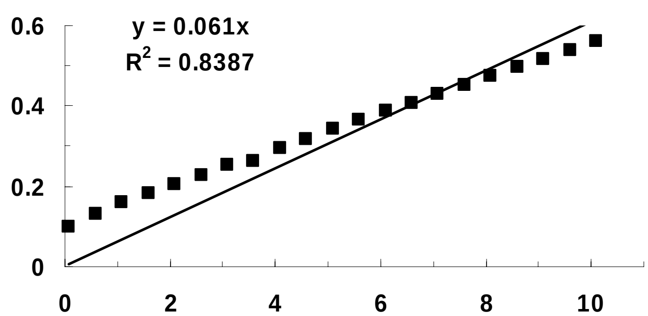
$$R = 30.3164\left(\frac{10}{10^3/28}\right) = 8.487$$

This is close to the standard value of 8.314 J/gmole.

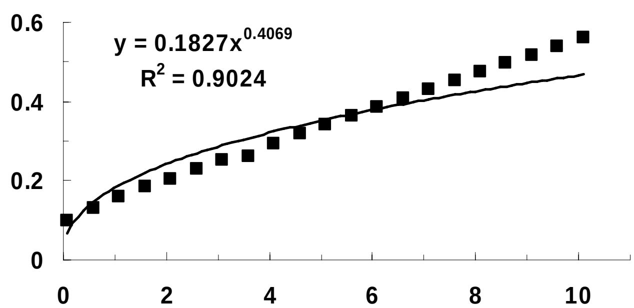
13.7 Linear regression gives



Forcing a zero intercept yields



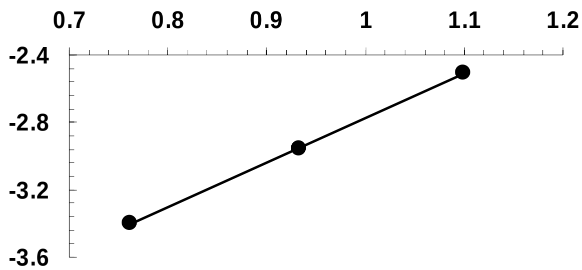
One alternative that would force a zero intercept is a power fit



However, this seems to represent a poor compromise since it misses the linear trend in the data. An alternative approach would to assume that the physically-unrealistic non-zero intercept is an artifact of the measurement method. Therefore, if the linear slope is valid, we might try $y = 0.0454x$.

13.14 Linear regression of the log transformed data yields

$$\log\dot{\varepsilon} = -5.41\log B + 2.6363\log\sigma \qquad (r^2 = 0.9997)$$



Therefore,

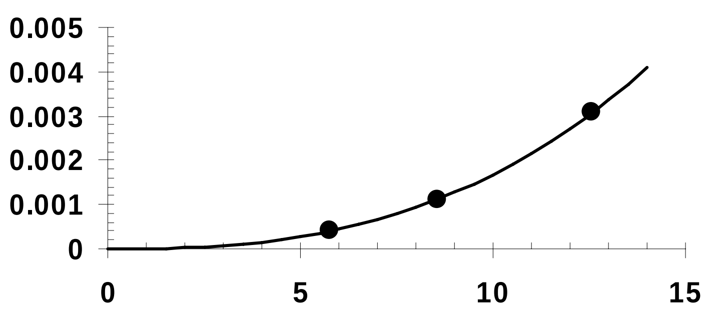
$$B = 10^{-5.41} = 3.88975\times10^{-6}$$

$$m = 2.6363$$

and the untransformed model is

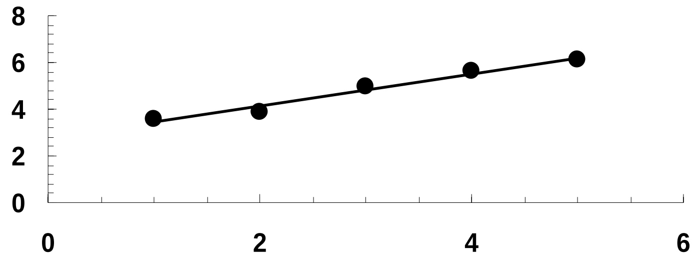
$$\dot{\varepsilon} = 3.88975\times10^{-6}\,\sigma^{2.6363}$$

A plot of the data and the model can be developed as



13.15 Linear regression of the data yields

$$\tau = 2.779 + 0.685\dot{\gamma} \qquad (r^2 = 0.977121)$$



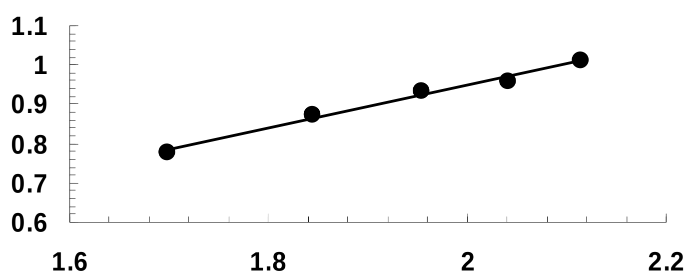
Therefore, $\mu = 0.685$ and $\tau_y = 2.779$ N/m^2.

13.16 The data can be transformed

strain	stress	log(strain)	log(stress)
50	5.99	1.69897	0.777427
70	7.45	1.845098	0.872156
90	8.56	1.954243	0.932474
110	9.09	2.041393	0.958564
130	10.25	2.113943	1.010724

Linear regression of the transformed data yields

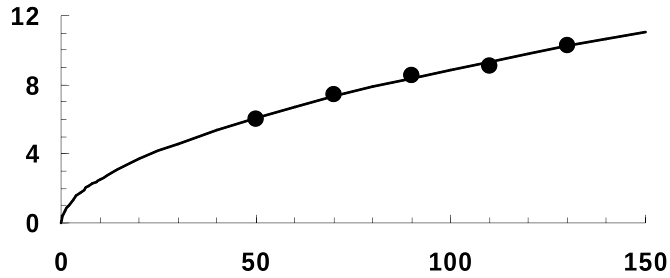
$\log \tau = -0.13808 + 0.54298 \log \dot{\gamma}$ $\qquad$ ($r^2 = 0.989118$)



Therefore, $\mu = 10^{-0.54298} =$ 0.72765 and $n = 0.54298$. The power model is therefore,

$\tau = 0.72765\dot{\gamma}^{0.54298}$

A plot of the power model along with the data can be created as


12
8
4
0
0
50
100
150