

Example 9.1

```
> restart:
```

```
> with (linalg):
```

Enter experimental data for the fraction of reactant remaining for all fifteen data points ($y_{exp,j} = 1,15$).

```
> yexp:=matrix(15,1,[0.980,0.983,0.955,0.979,0.993,0.626,0.544,
0.455,0.225,0.167,0.566,0.317,0.034,0.016,0.066]):
```

Enter values for the independent variables (time in hours and temperature in K) for all 15 data points in pairs ($t_j, temp_j, j=1,15$).

```
> x:=matrix(15,2,[0.1,100,0.2,100,0.3,100,0.4,100,0.5,100,0.05,
200,0.1,200,0.15,200,.2,200,0.25,200,0.02,300,0.04,300,0.06,300,
0.08,300,0.1,300]):
```

The first row in the x matrix is:

```
> x[1,1],x[1,2];
```

$0.1, 100$ (1)

Enter the model equation (y_{model}) for the fraction of reactant remaining as a function of time, temperature and parameter values θ_1 (Arrhenius constant, in hr^{-1}) and θ_2 (Activation energy divided by R, in K) .

```
> ymodel:=exp(-theta1*t*exp(-theta2/T));
```

$y_{model} := e^{-\theta_1 t e^{-\frac{\theta_2}{T}}}$ (2)

Use the y_{model} equation to predict the values of the dependent variable in y_{pred} .

```
> ypred:=matrix(15,1,[seq(eval(ymodel,{t=x[j,1],T=x[j,2]}),j=1.
.15)]):
```

Note that y_{pred} is a 15×1 matrix or vector, the first element of which is:

```
> ypred[1,1];
```

$e^{-0.1 \theta_1 e^{-\frac{1}{100} \theta_2}}$ (3)

Use Maple's Jacobian command in 'linalg' to find the elements of J (the sensitivity coefficients of the dependent variable to the parameters, θ_1 and θ_2). These are the derivatives of the dependent variable y with respect to the parameters evaluated at the experimental conditions

```
> J:=jacobian([seq(ypred[j,1],j=1..15)],[theta1,theta2]);
```

$$J := \begin{bmatrix} -0.1 e^{-\frac{1}{100} \varrho} e^{-0.1 \theta l e^{-\frac{1}{100} \varrho}} & 0.001000000000 \theta l e^{-\frac{1}{100} \varrho} e^{-0.1 \theta l e^{-\frac{1}{100} \varrho}} \\ -0.2 e^{-\frac{1}{100} \varrho} e^{-0.2 \theta l e^{-\frac{1}{100} \varrho}} & 0.002000000000 \theta l e^{-\frac{1}{100} \varrho} e^{-0.2 \theta l e^{-\frac{1}{100} \varrho}} \\ -0.3 e^{-\frac{1}{100} \varrho} e^{-0.3 \theta l e^{-\frac{1}{100} \varrho}} & 0.003000000000 \theta l e^{-\frac{1}{100} \varrho} e^{-0.3 \theta l e^{-\frac{1}{100} \varrho}} \\ -0.4 e^{-\frac{1}{100} \varrho} e^{-0.4 \theta l e^{-\frac{1}{100} \varrho}} & 0.004000000000 \theta l e^{-\frac{1}{100} \varrho} e^{-0.4 \theta l e^{-\frac{1}{100} \varrho}} \\ -0.5 e^{-\frac{1}{100} \varrho} e^{-0.5 \theta l e^{-\frac{1}{100} \varrho}} & 0.005000000000 \theta l e^{-\frac{1}{100} \varrho} e^{-0.5 \theta l e^{-\frac{1}{100} \varrho}} \\ -0.05 e^{-\frac{1}{200} \varrho} e^{-0.05 \theta l e^{-\frac{1}{200} \varrho}} & 0.0002500000000 \theta l e^{-\frac{1}{200} \varrho} e^{-0.05 \theta l e^{-\frac{1}{200} \varrho}} \\ -0.1 e^{-\frac{1}{200} \varrho} e^{-0.1 \theta l e^{-\frac{1}{200} \varrho}} & 0.0005000000000 \theta l e^{-\frac{1}{200} \varrho} e^{-0.1 \theta l e^{-\frac{1}{200} \varrho}} \\ -0.15 e^{-\frac{1}{200} \varrho} e^{-0.15 \theta l e^{-\frac{1}{200} \varrho}} & 0.0007500000000 \theta l e^{-\frac{1}{200} \varrho} e^{-0.15 \theta l e^{-\frac{1}{200} \varrho}} \\ -0.2 e^{-\frac{1}{200} \varrho} e^{-0.2 \theta l e^{-\frac{1}{200} \varrho}} & 0.001000000000 \theta l e^{-\frac{1}{200} \varrho} e^{-0.2 \theta l e^{-\frac{1}{200} \varrho}} \\ -0.25 e^{-\frac{1}{200} \varrho} e^{-0.25 \theta l e^{-\frac{1}{200} \varrho}} & 0.001250000000 \theta l e^{-\frac{1}{200} \varrho} e^{-0.25 \theta l e^{-\frac{1}{200} \varrho}} \\ -0.02 e^{-\frac{1}{300} \varrho} e^{-0.02 \theta l e^{-\frac{1}{300} \varrho}} & 0.00006666666667 \theta l e^{-\frac{1}{300} \varrho} e^{-0.02 \theta l e^{-\frac{1}{300} \varrho}} \\ -0.04 e^{-\frac{1}{300} \varrho} e^{-0.04 \theta l e^{-\frac{1}{300} \varrho}} & 0.0001333333333 \theta l e^{-\frac{1}{300} \varrho} e^{-0.04 \theta l e^{-\frac{1}{300} \varrho}} \\ -0.06 e^{-\frac{1}{300} \varrho} e^{-0.06 \theta l e^{-\frac{1}{300} \varrho}} & 0.0002000000000 \theta l e^{-\frac{1}{300} \varrho} e^{-0.06 \theta l e^{-\frac{1}{300} \varrho}} \\ -0.08 e^{-\frac{1}{300} \varrho} e^{-0.08 \theta l e^{-\frac{1}{300} \varrho}} & 0.0002666666667 \theta l e^{-\frac{1}{300} \varrho} e^{-0.08 \theta l e^{-\frac{1}{300} \varrho}} \\ -0.1 e^{-\frac{1}{300} \varrho} e^{-0.1 \theta l e^{-\frac{1}{300} \varrho}} & 0.0003333333333 \theta l e^{-\frac{1}{300} \varrho} e^{-0.1 \theta l e^{-\frac{1}{300} \varrho}} \end{bmatrix} \quad (4)$$

Note that J is a 15 x 2 matrix. The rows of J consist of two elements. The first element in the first row is the derivative of the model equation (ymodel) with respect to theta1 evaluated at the experimental conditions associated with the first data point (x[1,1] and x[1,2]). We use ypred to obtain and evaluate this element. The second element in the first row of J is the derivate of ymodel with respect to theta2 evaluated at the first data point. The elements in the first row of J are shown next.

```
> J[15,1];J[15,2];
```

$$-0.1 e^{-\frac{1}{300} \theta} e^{-0.1 \theta} e^{-\frac{1}{300} \theta} \\ 0.0003333333333 \theta e^{-\frac{1}{300} \theta} e^{-0.1 \theta} e^{-\frac{1}{300} \theta} \quad (5)$$

Next, determine the error between the measured value of y (yexp) and the predicted value of y (ypred). This is needed in the iteration process programmed below for determining the parameter values, theta1 and theta2, via the least squares equation.

```
> err:=evalm(yexp-ypred):
```

The variable named err is a 15 x 1 matrix the first element of which is:

```
> err[1,1];
```

$$0.980 - e^{-0.1 \theta} e^{-\frac{1}{100} \theta} \quad (6)$$

Initialize an iteration counter, n_iters.

```
> n_iters:=0;
```

$$n_iters := 0 \quad (7)$$

Initialize the parameter vector theta. Initial guesses can often be determined by considering only one of the data points. Another method is to try different initial guesses until the method yields the same converged values for theta1 and theta2.

```
> theta:=matrix(2,1,[500,1e3]);
```

$$\theta := \begin{bmatrix} 500 \\ 1000. \end{bmatrix} \quad (8)$$

Initialize the delta theta vector.

```
> dtheta:=evalm(theta) ;
```

$$dtheta := \begin{bmatrix} 500 \\ 1000. \end{bmatrix} \quad (9)$$

Write a 'do' loop that solves for delta theta, updates the parameter values, and uses a while statement to check for convergence or exceeding 100 iterations.

```
> while max(abs(dtheta[1,1]/theta[1,1]),abs(dtheta[2,1]/theta[2,1]
))> 1.0e-7 and n_iters <100 do Jeval:=eval(J,{theta1=theta[1,1],
theta2=theta[2,1]}):JevalT:=transpose(Jeval):dtheta:=evalm(in-
verse(JevalT*Jeval)*JevalT*eval(err,{theta1=theta[1,1],
theta2=theta[2,1]})):theta:=evalm(theta+dtheta); n_iters:=
n_iters+1; end do:
```

Error, missing operator or `;

Enter the number of iterations needed to obtain convergence.

```
> n_iters;
```

$$0 \quad (10)$$

Use the converged values of theta1 and theta2 to determine predicted values of y for plotting, call this plotypred:

```
> evalm(theta);
```

$$\begin{bmatrix} 500 \\ 1000. \end{bmatrix} \quad (11)$$

```
> plotypred:=eval(ymodel,{theta1=theta[1,1],theta2=theta[2,1]});
```

$$plotypred := e^{-500te - \frac{1000.}{T}} \quad (12)$$

Prepare the experimental data points for plotting.

```
> plotdata:=[seq([x[j,1],x[j,2],yexp[j,1]],j=1..15)];
```

```
plotdata := [[0.1, 100, 0.980], [0.2, 100, 0.983], [0.3, 100, 0.955], [0.4, 100, 0.979], [0.5, 100, 0.993], [0.05, 200, 0.626], [0.1, 200, 0.544], [0.15, 200, 0.455], [0.2, 200, 0.225], [0.25, 200, 0.167], [0.02, 300, 0.566], [0.04, 300, 0.317], [0.06, 300, 0.034], [0.08, 300, 0.016], [0.1, 300, 0.066]]
```

(13)

```
> plotdata:=[ [.1, 100, .980],[.2, 100, .983],[.3, 100,.955],[.4, 100,.979],[.5,100,.993],[0.5,200,.626],[.1,200,.544],[.15, 200,.455],[.2,200,.225],[.25,200,.167],[0.2,300,.566],[0.4,300,.317],[0.6,300,0.34],[0.8,300,0.16],[.1,300,0.66]];
```

```
plotdata := [[0.1, 100, 0.980], [0.2, 100, 0.983], [0.3, 100, 0.955], [0.4, 100, 0.979], [0.5, 100, 0.993], [0.5, 200, 0.626], [0.1, 200, 0.544], [0.15, 200, 0.455], [0.2, 200, 0.225], [0.25, 200, 0.167], [0.2, 300, 0.566], [0.4, 300, 0.317], [0.6, 300, 0.34], [0.8, 300, 0.16], [0.1, 300, 0.66]]
```

(14)

Specify the data (yexp) plot, P1, and the predicted values (ypred), P2.

```
> P1:=plot3d(plotdata, t=0..0.5, T=100..300, axes=boxed, style=point, symbol=box, symbolsize=14, color=black):
```

```
> P2:=plot3d(plotypred,t=0..0.5,T=100..300,axes=boxed,style=wireframe):
```

Display the plots for the data and the predicted values of y on the same plot as functions of time, t, and temperature, T.

```
> with(plots):
```

```
> display({P1,P2},title="Example 9.1",labels=["t(hr)","T(K)","y"], labeldirections=[horiz
```