```
t=[3.4935;4.2853;5.1374;5.8181;6.8632;8.1841];
x=[6;10.1333;14.2667;18.4000;22.5333;26.6667];
m = length(t);
n = 2;
G = [ ones(m,1) , x ];
sigma = 0.1;
I = eye(m);
W = sigma\I;
G_w = W*G;
t_w = W*t;
%The maximum likelihood estimates for the model parameters t_0 and s_2

m_mle = (G.'*W^2*G)\ G.'*W^2*t
```

```
m_mle = 2×1
    2.032337083707937
    0.220281403038290
```

1.

Given $\mathrm{Cov}(d) = \sigma^2 I$ and $W = \sigma^{-1} I$,

From;

$$C = \left(G_W^T G_w\right)^{-1} G_W^T \,\mathrm{Cov}(d_w)\, G_w \left(G_W^T G_w\right)^{-1}$$
$$= (G^T W^2 G)^{-1} (\mathrm{WG})^T \;\mathrm{Cov}(\mathrm{Wd})(\mathrm{WG})(G^T W^2 G)^{-1}$$
$$= (G^T W^2 G)^{-1} \left(G^T W^T\right) \; W\,\mathrm{Cov}(d)\, W^T \;(\mathrm{WG})\,(G^T W^2 G)^{-1} \qquad (1)$$

But $W = \sigma^{-1} I$, equation (1) then becomes;

$$C = (G^T \sigma^{-2} I\, G)^{-1} \left(G^T \sigma^{-1} I^T\right) \;\sigma^{-1} I\,(\sigma^2 I)\,\sigma^{-1} I^T \;(\sigma^{-1}\mathrm{IG})\,(G^T \sigma^{-2} I\, G)^{-1}$$
$$= \sigma^2 \;(G^T\; G)^{-1} G^T \;\sigma^{-2}\,(\sigma^2 I)\,\sigma^{-2}\, G \quad \sigma^2\,(G^T G)^{-1}$$
$$= \;\sigma^2 \;\;(G^T\; G)^{-1}\;\, G^T\, I\, G \;\;(G^T G)^{-1} \qquad (2)$$

But $(G^T\; G)^{-1}\;\, G^T\, I\, G \;\;(G^T G)^{-1} = (G^T\; G)^{-1}$,

Therefore $C = \sigma^2 \;(G^T\; G)^{-1}$

```
%format long
C = sigma^2 * inv(G.'*G)
```

```
C = 2×2
    0.010589647096896  -0.000546304924300
   -0.000546304924300   0.000033447240263
```

Discussion

From the covariance matrix above, the value at C(1,1) = 0.01059 is the variance of the least squares estimate for t_0, the value at at C(2,2) = 0.000033 is the variance of the least squares estimate for s_2 and the value at both points i.e. C(1,2) = C(2,1)= -0.000546 is the covariance between the least squares estimates for t_0 and s_2.

2.

```
%  95% parameter confidence intervals
[m_mle-1.96*sqrt(diag(C)) , m_mle+1.96*sqrt(diag(C))]
```

```
ans = 2×2
   1.830641302176854   2.234032865239020
   0.208946019577846   0.231616786498733
```

Discussion

The above 95% confidence intervals for the least squares estimates t_0 and $s_2$respectively mean that;

$t_0 \approx [1.8306, 2.2340]$ and that $s_2 \approx [0.2089, 0.2316]$. From question (1) above, we have that $\sqrt{C(1,1) * C(2,2)} = \sqrt{0.01059 * 0.000033} \neq 0$ therefore the 95% confidence interval doesnot capture the relationship between the model parameters t_0 and s_2.

3.

```
% Model parameter correlation matrix
sd=sqrt(diag(C));
rho = C./(sd *sd')
```

```
rho = 2×2
   1.000000000000000  -0.917939855336270
  -0.917939855336270   1.000000000000000
```

From above, the correlation between the parameters t_0 and s_2 is -0.9179 which is high and therefore the two model parameters are highly negatively statistically dependent meaning that an increase in t_0 leads to a decrease in s_2 . The ones on the main diagonal are understandable because the correlation of a parameter and itself is one.

4.

```
DELTA2 = chi2inv(0.95,2)
```

```
DELTA2 =
   5.991464547107981
```

The inequality for this ellipsoid is given by;

$(m_{\text{true}} - m_{\text{mle}})\, C^{-1}\, (m_{\text{true}} - m_{\text{mle}}) \leq \Delta^2$ Δ^2 is the 95^{th}percentile of a χ^2 distribution with 2 degrees of freedom.

$$\Rightarrow (m_{\text{true}} - m_{\text{mle}})\, C^{-1}\, (m_{\text{true}} - m_{\text{mle}}) \leq 5.991 \qquad (1)$$

The above inequality defines the ellipsoidal confidence region in the 2 dimensional Cartesian space where the centre of the ellipsoid gives the true values of the model parameters. And the confidence region captures the relatioship between the uncertainity and the parameters.

5.

```
%C_inv = Q*lambda*Q.'
%inv(C)
[Q,lambda]=eig(inv(C))
```

```
Q = 2×2
  -0.998670643915139   0.051545562196187
   0.051545562196187   0.998670643915139
lambda = 2×2
10^5 ×
   0.000941810770143                   0
                   0   1.904703184789859
```

Below are the benefits of using decomposition to define a confidence region for the parameter estimates;

(i). The eigenvectors given by the columns i.e. $Q_{.,i}$ in the matrix Q corresponding to the the two eigenvalues in matrix 'lambda' give the direction of the ellipsoid axes.

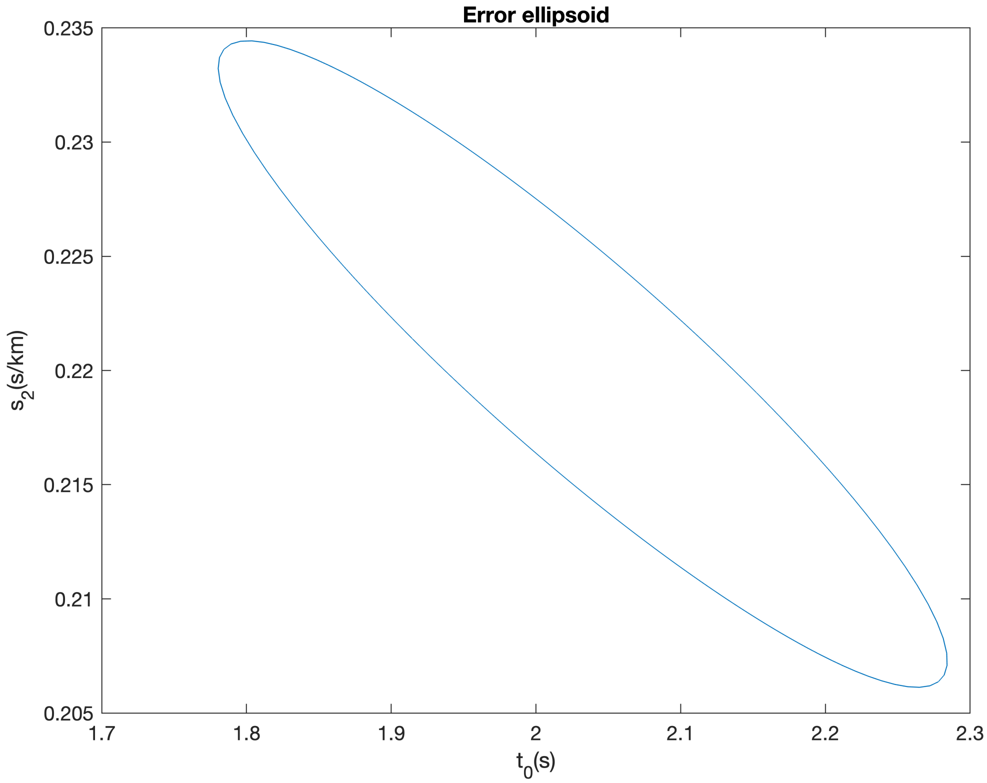
(ii). The eigenvalues seen in the main diagonal of the diagonal matrix 'lambda' above are used to find the lengths of the ellipsoid axes using the formular $\frac{\Delta}{\lambda_i}$ respectively where Δ^2 is the 95^{th}percentile of a χ^2 distribution with 2 degrees of freedom.

6.

```
DELTA2 = chi2inv(0.95,2)
```

```
DELTA2 =
   5.991464547107981
```

```
plot_ellipse(DELTA2,C,m_mle)
xlabel('t_0(s)');
ylabel('s_2(s/km)')
title('Error ellipsoid')
```



The error ellipsoid is highly statistically dependent and negatively inclined in the model space. This is becaue the correllation between the parameter estimates for t_0 and s_2 that is -0.9179 is approaching -1, hence the projection being needle-like with its long principal axis having a negative slope. The centre of the ellipsoid gives the true values of the model parameters t_0 and s_2.

```
function plot_ellipse(DELTA2,C,m_mle)
n=3000;
%construct a vector of n equally-spaced angles from (0,2*pi)
theta=linspace(0,2*pi,n)';
%corresponding unit vector
xhat=[cos(theta),sin(theta)];
Cinv=inv(C);
%preallocate output array
r=zeros(n,2);
for i=1:n
```

```
  %store each (x,y) pair on the confidence ellipse
  %in the corresponding row of r
  r(i,:)=sqrt(DELTA2/(xhat(i,:)*Cinv*xhat(i,:)'))*xhat(i,:);
end
%
% Plot the ellipse and set the axes.
%
plot(m_mle(1)+r(:,1), m_mle(2)+r(:,2));
axis equal

plot(m_mle(1)+r(:,1), m_mle(2)+r(:,2));
end
```