

```

> restart;
Maple calculations regarding example 3
=====

> for i from 1 to 20 do yc[i]:=a0+a1*x[i] end do;
> for i from 1 to 20 do s[i]:=s0+s1*abs(yc[i]) end do;
> lns:=add(ln(s[i]),i=1..20):
> lt:=add((y[i]-yc[i])^2/2/s[i]^2,i=1..20):

The probability density function 'Prob_ex3' is defined:
> Prob_ex3:=exp(-(20/2)*ln(2*Pi)-lns-lt):

The data points x[i] and y[i] are introduced:
> y[1]:=-6.65647123;y[2]:=-0.011729798;y[3]:=4.34198;y[4]:=4.56334;
y[5]:=8.37979;y[6]:=10.2627;y[7]:=10.22623;y[8]:=14.28611;y[9]:=13.79561;y[10]:=21.09856;y[11]:=17.50144;y[12]:=20.10634;y[13]:=23.32865;y[14]:=18.41583;y[15]:=36.10625;y[16]:=34.69508;y[17]:=33.13127;y[18]:=43.87842;y[19]:=34.06836;y[20]:=43.07314;


$$\begin{aligned} y_1 &:= -6.65647123 \\ y_2 &:= -0.011729798 \\ y_3 &:= 4.34198 \\ y_4 &:= 4.56334 \\ y_5 &:= 8.37979 \\ y_6 &:= 10.2627 \\ y_7 &:= 10.22623 \\ y_8 &:= 14.28611 \\ y_9 &:= 13.79561 \\ y_{10} &:= 21.09856 \\ y_{11} &:= 17.50144 \\ y_{12} &:= 20.10634 \\ y_{13} &:= 23.32865 \\ y_{14} &:= 18.41583 \\ y_{15} &:= 36.10625 \\ y_{16} &:= 34.69508 \\ y_{17} &:= 33.13127 \\ y_{18} &:= 43.87842 \\ y_{19} &:= 34.06836 \\ y_{20} &:= 43.07314 \end{aligned} \tag{1}$$


> x[1]:=-10;x[2]:=-5;x[3]:=0;x[4]:=5;x[5]:=10;x[6]:=15;x[7]:=20;x[8]:=25;x[9]:=30;x[10]:=35;x[11]:=40;x[12]:=45;x[13]:=50;x[14]:=55;x[15]:=60;x[16]:=65;x[17]:=70;x[18]:=75;x[19]:=80;x[20]:=85;
x_1 := -10

```

$$\begin{aligned}
x_2 &:= -5 \\
x_3 &:= 0 \\
x_4 &:= 5 \\
x_5 &:= 10 \\
x_6 &:= 15 \\
x_7 &:= 20 \\
x_8 &:= 25 \\
x_9 &:= 30 \\
x_{10} &:= 35 \\
x_{11} &:= 40 \\
x_{12} &:= 45 \\
x_{13} &:= 50 \\
x_{14} &:= 55 \\
x_{15} &:= 60 \\
x_{16} &:= 65 \\
x_{17} &:= 70 \\
x_{18} &:= 75 \\
x_{19} &:= 80 \\
x_{20} &:= 85
\end{aligned} \tag{2}$$

```

> with(LinearAlgebra):
> with(VectorCalculus):
[&x, `*`, `+`, `^`, ` `, <, >, <|>, About, AddCoordinates, ArcLength, BasisFormat, Binormal,
ConvertVector, CrossProduct, Curl, Curvature, D, Del, DirectionalDiff, Divergence,
DotProduct, Flux, GetCoordinateParameters, GetCoordinates, GetNames,
GetPVDescription, GetRootPoint, GetSpace, Gradient, Hessian, IsPositionVector,
IsRootedVector, IsVectorField, Jacobian, Laplacian, LineInt, MapToBasis,  $\nabla$ , Norm,
Normalize, PathInt, PlotPositionVector, PlotVector, PositionVector, PrincipalNormal,
RadiusOfCurvature, RootedVector, ScalarPotential, SetCoordinateParameters,
SetCoordinates, SpaceCurve, SurfaceInt, TNBFrame, TangentLine, TangentPlane,
TangentVector, Torsion, Vector, VectorField, VectorPotential, VectorSpace, Wronskian,
diff, eval, evalVF, int, limit, series]

```

```
> g1 := Gradient(Prob_ex3, [s0, s1, a0, a1]):
```

The gradient 'g1' is calculated, i.e. the partial derivatives 'delpdela0=partial(Prob\_ex3)/partial(a0)', 'delpdela1=partial(Prob\_ex3)/partial(a1)', 'delpdels0=partial(Prob\_ex3)/partial(s0)' and 'delpdels1=partial(Prob\_ex3)/partial(s1)' with respect to 'a0', 'a1', 's0' and 's1' are computed:

```
> delpdela1:=((-a1*x[4]-a0+y[4])*x[4]/(s0+s1*abs(a1*x[4]+a0))^2+(-a1*x[20]-a0+y[20])*x[20]/(s0+s1*abs(a1*x[20]+a0))^2+(-a1*x[9]-a0+y[9])*x[9]/(s0+s1*abs(a1*x[9]+a0))^2+(-a1*x[16]-a0+y[16])*x[16]/(s0+s1*abs(a1*x[16]+a0))^2+(-a1*x[13]-a0+y[13])*x[13]/(s0+s1*abs(a1*x[13]+a0)))
```

$$\begin{aligned}
& (a1*x[13]+a0) )^2 + (-a1*x[19]-a0+y[19])*x[19]/(s0+s1*abs(a1*x[19]+a0)) \\
& ^2 + (-a1*x[15]-a0+y[15])*x[15]/(s0+s1*abs(a1*x[15]+a0)) ^2 + (-a1*x[10]-a0+y[10])*x[10]/(s0+s1*abs(a1*x[10]+a0)) ^2 + (-a1*x[18]-a0+y[18])*x[18]/(s0+s1*abs(a1*x[18]+a0)) ^2 + (-a1*x[17]-a0+y[17])*x[17]/(s0+s1*abs(a1*x[17]+a0)) ^2 + (-a1*x[7]-a0+y[7])*x[7]/(s0+s1*abs(a1*x[7]+a0)) ^2 + (-a1*x[3]-a0+y[3])*x[3]/(s0+s1*abs(a1*x[3]+a0)) \\
& ^2 + (-a1*x[6]-a0+y[6])*x[6]/(s0+s1*abs(a1*x[6]+a0)) ^2 + (-a1*x[11]-a0+y[11])*x[11]/(s0+s1*abs(a1*x[11]+a0)) ^2 + (-a1*x[14]-a0+y[14])*x[14]/(s0+s1*abs(a1*x[14]+a0)) ^2 + (-a1*x[5]-a0+y[5])*x[5]/(s0+s1*abs(a1*x[5]+a0)) ^2 + (-a1*x[2]-a0+y[2])*x[2]/(s0+s1*abs(a1*x[2]+a0)) \\
& ^2 + (-a1*x[12]-a0+y[12])*x[12]/(s0+s1*abs(a1*x[12]+a0)) ^2 + (-a1*x[1]-a0+y[1])*x[1]/(s0+s1*abs(a1*x[1]+a0)) ^2 + (-a1*x[8]-a0+y[8])*x[8]/(s0+s1*abs(a1*x[8]+a0)) ^2 + (-a1*x[10]-a0+y[10])^2 *s1*x[10]*abs(1, a1*x[10]+a0)/(s0+s1*abs(a1*x[10]+a0)) \\
& ^3 + (-a1*x[11]-a0+y[11])^2 *s1*x[11]*abs(1, a1*x[11]+a0)/(s0+s1*abs(a1*x[11]+a0)) ^3 + (-a1*x[12]-a0+y[12])^2 *s1*x[12]*abs(1, a1*x[12]+a0)/(s0+s1*abs(a1*x[12]+a0)) ^3 + (-a1*x[13]-a0+y[13])^2 *s1*x[13]*abs(1, a1*x[13]+a0)/(s0+s1*abs(a1*x[13]+a0)) \\
& ^3 + (-a1*x[14]-a0+y[14])^2 *s1*x[14]*abs(1, a1*x[14]+a0)/(s0+s1*abs(a1*x[14]+a0)) ^3 + (-a1*x[15]-a0+y[15])^2 *s1*x[15]*abs(1, a1*x[15]+a0)/(s0+s1*abs(a1*x[15]+a0)) \\
& ^3 + (-a1*x[16]-a0+y[16])^2 *s1*x[16]*abs(1, a1*x[16]+a0)/(s0+s1*abs(a1*x[16]+a0)) ^3 + (-a1*x[17]-a0+y[17])^2 *s1*x[17]*abs(1, a1*x[17]+a0)/(s0+s1*abs(a1*x[17]+a0)) \\
& ^3 + (-a1*x[18]-a0+y[18])^2 *s1*x[18]*abs(1, a1*x[18]+a0)/(s0+s1*abs(a1*x[18]+a0)) ^3 + (-a1*x[19]-a0+y[19])^2 *s1*x[19]*abs(1, a1*x[19]+a0)/(s0+s1*abs(a1*x[19]+a0)) \\
& ^3 + (-a1*x[20]-a0+y[20])^2 *s1*x[20]*abs(1, a1*x[20]+a0)/(s0+s1*abs(a1*x[20]+a0)) ^3 + (-a1*x[1]-a0+y[1])^2 *s1*x[1]*abs(1, a1*x[1]+a0)/(s0+s1*abs(a1*x[1]+a0)) \\
& ^3 + (-a1*x[2]-a0+y[2])^2 *s1*x[2]*abs(1, a1*x[2]+a0)/(s0+s1*abs(a1*x[2]+a0)) ^3 + (-a1*x[3]-a0+y[3])^2 *s1*x[3]*abs(1, a1*x[3]+a0)/(s0+s1*abs(a1*x[3]+a0)) \\
& ^3 + (-a1*x[4]-a0+y[4])^2 *s1*x[4]*abs(1, a1*x[4]+a0)/(s0+s1*abs(a1*x[4]+a0)) ^3 + (-a1*x[5]-a0+y[5])^2 *s1*x[5]*abs(1, a1*x[5]+a0)/(s0+s1*abs(a1*x[5]+a0)) \\
& ^3 + (-a1*x[6]-a0+y[6])^2 *s1*x[6]*abs(1, a1*x[6]+a0)/(s0+s1*abs(a1*x[6]+a0)) ^3 + (-a1*x[7]-a0+y[7])^2 *s1*x[7]*abs(1, a1*x[7]+a0)/(s0+s1*abs(a1*x[7]+a0)) \\
& ^3 + (-a1*x[8]-a0+y[8])^2 *s1*x[8]*abs(1, a1*x[8]+a0)/(s0+s1*abs(a1*x[8]+a0)) ^3 + (-a1*x[9]-a0+y[9])^2 *s1*x[9]*abs(1, a1*x[9]+a0)/(s0+s1*abs(a1*x[9]+a0)) \\
& -s1*x[10]*abs(1, a1*x[10]+a0)/(s0+s1*abs(a1*x[10]+a0)) -s1*x[11]*abs(1, a1*x[11]+a0)/(s0+s1*abs(a1*x[11]+a0)) -s1*x[12]*abs(1, a1*x[12]+a0)/(s0+s1*abs(a1*x[12]+a0)) \\
& -s1*x[13]*abs(1, a1*x[13]+a0)/(s0+s1*abs(a1*x[13]+a0)) -s1*x[14]*abs(1, a1*x[14]+a0)/(s0+s1*abs(a1*x[14]+a0)) -s1*x[15]*abs(1, a1*x[15]+a0)/(s0+s1*abs(a1*x[15]+a0)) \\
& -s1*x[16]*abs(1, a1*x[16]+a0)/(s0+s1*abs(a1*x[16]+a0)) -s1*x[17]*abs(1, a1*x[17]+a0)/(s0+s1*abs(a1*x[17]+a0)) -s1*x[18]*abs(1, a1*x[18]+a0)/(s0+s1*abs(a1*x[18]+a0)) \\
& -s1*x[19]*abs(1, a1*x[19]+a0)/(s0+s1*abs(a1*x[19]+a0)) -s1*x[20]*abs(1, a1*x[20]+a0)/(s0+s1*abs(a1*x[20]+a0)) -s1*x[1]*abs(1, a1*x[1]+a0)/(s0+s1*abs(a1*x[1]+a0)) \\
& -s1*x[2]*abs(1, a1*x[2]+a0)/(s0+s1*abs(a1*x[2]+a0)) -s1*x[3]*abs(1, a1*x[3]+a0)/(s0+s1*abs(a1*x[3]+a0)) -s1*x[4]*abs(1, a1*x[4]+a0)/(s0+s1*abs(a1*x[4]+a0)) \\
& -s1*x[5]*abs(1, a1*x[5]+a0)/(s0+s1*abs(a1*x[5]+a0)) -s1*x[6]*abs(1, a1*x[6]+a0)/(s0+s1*abs(a1*x[6]+a0)) -s1*x[7]*abs(1, a1*x[7]+a0)/(s0+s1*abs(a1*x[7]+a0)) \\
& -s1*x[8]*abs(1, a1*x[8]+a0)/(s0+s1*abs(a1*x[8]+a0)) *exp(-10*ln(2*Pi)-ln(s0+s1*abs(a1*x[12]+a0))-ln(s0+s1*abs(a1*x[13]+a0))-ln(s0+s1*abs(a1*x[14]+a0))-ln(s0+s1*abs(a1*x[15]+a0))-ln(s0+s1*abs(a1*x[16]+a0)) \\
& -ln(s0+s1*abs(a1*x[17]+a0))-ln(s0+s1*abs(a1*x[18]+a0))-ln(s0+s1*abs(a1*x[19]+a0))-ln(s0+s1*abs(a1*x[20]+a0))-ln(s0+
\end{aligned}$$



```

s1*abs(a1*x[18]+a0))^2+(-a1*x[20]-a0+y[20])/ (s0+s1*abs(a1*x[20]+a0))^2+(-a1*x[19]-a0+y[19])/ (s0+s1*abs(a1*x[19]+a0))^2-s1*abs(1,a1*x[8]+a0)/(s0+s1*abs(a1*x[3]+a0))-s1*abs(1,a1*x[12]+a0)/(s0+s1*abs(a1*x[12]+a0))-s1*abs(1,a1*x[18]+a0)/(s0+s1*abs(a1*x[18]+a0))-s1*abs(1,a1*x[6]+a0)/(s0+s1*abs(a1*x[6]+a0))-s1*abs(1,a1*x[20]+a0)/(s0+s1*abs(a1*x[20]+a0))-s1*abs(1,a1*x[17]+a0)/(s0+s1*abs(a1*x[17]+a0))-s1*abs(1,a1*x[1]+a0)/(s0+s1*abs(a1*x[1]+a0))-s1*abs(1,a1*x[7]+a0)/(s0+s1*abs(a1*x[7]+a0))-s1*abs(1,a1*x[5]+a0)/(s0+s1*abs(a1*x[5]+a0))-s1*abs(1,a1*x[11]+a0)/(s0+s1*abs(a1*x[11]+a0))-s1*abs(1,a1*x[4]+a0)/(s0+s1*abs(a1*x[4]+a0))-s1*abs(1,a1*x[15]+a0)/(s0+s1*abs(a1*x[15]+a0))-s1*abs(1,a1*x[2]+a0)/(s0+s1*abs(a1*x[2]+a0))-s1*abs(1,a1*x[10]+a0)/(s0+s1*abs(a1*x[10]+a0))-s1*abs(1,a1*x[13]+a0)/(s0+s1*abs(a1*x[13]+a0))-s1*abs(1,a1*x[16]+a0)/(s0+s1*abs(a1*x[16]+a0))-s1*abs(1,a1*x[14]+a0)/(s0+s1*abs(a1*x[14]+a0))-s1*abs(1,a1*x[9]+a0)/(s0+s1*abs(a1*x[9]+a0))-s1*abs(1,a1*x[19]+a0)/(s0+s1*abs(a1*x[19]+a0)))*exp(-10*ln(2*Pi)-ln(s0+s1*abs(a1*x[12]+a0))-ln(s0+s1*abs(a1*x[13]+a0))-ln(s0+s1*abs(a1*x[14]+a0))-ln(s0+s1*abs(a1*x[15]+a0))-ln(s0+s1*abs(a1*x[16]+a0))-ln(s0+s1*abs(a1*x[17]+a0))-ln(s0+s1*abs(a1*x[18]+a0))-ln(s0+s1*abs(a1*x[19]+a0))-ln(s0+s1*abs(a1*x[20]+a0))-ln(s0+s1*abs(a1*x[1]+a0))-ln(s0+s1*abs(a1*x[2]+a0))-ln(s0+s1*abs(a1*x[3]+a0))-ln(s0+s1*abs(a1*x[4]+a0))-ln(s0+s1*abs(a1*x[5]+a0))-ln(s0+s1*abs(a1*x[6]+a0))-ln(s0+s1*abs(a1*x[7]+a0))-ln(s0+s1*abs(a1*x[8]+a0))-ln(s0+s1*abs(a1*x[9]+a0))-ln(s0+s1*abs(a1*x[10]+a0))-ln(s0+s1*abs(a1*x[11]+a0))-(-a1*x[20]-a0+y[20])^2/(2*(s0+s1*abs(a1*x[20]+a0))^2)-(-a1*x[19]-a0+y[19])^2/(2*(s0+s1*abs(a1*x[19]+a0))^2)-(-a1*x[18]-a0+y[18])^2/(2*(s0+s1*abs(a1*x[18]+a0))^2)-(-a1*x[17]-a0+y[17])^2/(2*(s0+s1*abs(a1*x[17]+a0))^2)-(-a1*x[16]-a0+y[16])^2/(2*(s0+s1*abs(a1*x[16]+a0))^2)-(-a1*x[15]-a0+y[15])^2/(2*(s0+s1*abs(a1*x[15]+a0))^2)-(-a1*x[14]-a0+y[14])^2/(2*(s0+s1*abs(a1*x[14]+a0))^2)-(-a1*x[13]-a0+y[13])^2/(2*(s0+s1*abs(a1*x[13]+a0))^2)-(-a1*x[12]-a0+y[12])^2/(2*(s0+s1*abs(a1*x[12]+a0))^2)-(-a1*x[11]-a0+y[11])^2/(2*(s0+s1*abs(a1*x[11]+a0))^2)-(-a1*x[10]-a0+y[10])^2/(2*(s0+s1*abs(a1*x[10]+a0))^2)-(-a1*x[9]-a0+y[9])^2/(2*(s0+s1*abs(a1*x[9]+a0))^2)-(-a1*x[8]-a0+y[8])^2/(2*(s0+s1*abs(a1*x[8]+a0))^2)-(-a1*x[7]-a0+y[7])^2/(2*(s0+s1*abs(a1*x[7]+a0))^2)-(-a1*x[6]-a0+y[6])^2/(2*(s0+s1*abs(a1*x[6]+a0))^2)-(-a1*x[5]-a0+y[5])^2/(2*(s0+s1*abs(a1*x[5]+a0))^2)-(-a1*x[4]-a0+y[4])^2/(2*(s0+s1*abs(a1*x[4]+a0))^2)-(-a1*x[3]-a0+y[3])^2/(2*(s0+s1*abs(a1*x[3]+a0))^2)-(-a1*x[2]-a0+y[2])^2/(2*(s0+s1*abs(a1*x[2]+a0))^2)-(-a1*x[1]-a0+y[1])^2/(2*(s0+s1*abs(a1*x[1]+a0))^2));
> delpdels0:=(-1/(s0+s1*abs(a1*x[4]+a0))-1/(s0+s1*abs(a1*x[3]+a0))-1/(s0+s1*abs(a1*x[2]+a0))-1/(s0+s1*abs(a1*x[1]+a0))-1/(s0+s1*abs(a1*x[20]+a0))-1/(s0+s1*abs(a1*x[19]+a0))-1/(s0+s1*abs(a1*x[18]+a0))-1/(s0+s1*abs(a1*x[17]+a0))-1/(s0+s1*abs(a1*x[16]+a0))-1/(s0+s1*abs(a1*x[15]+a0))-1/(s0+s1*abs(a1*x[14]+a0))-1/(s0+s1*abs(a1*x[13]+a0))-1/(s0+s1*abs(a1*x[12]+a0))-1/(s0+s1*abs(a1*x[11]+a0))-1/(s0+s1*abs(a1*x[10]+a0))-1/(s0+s1*abs(a1*x[9]+a0))-1/(s0+s1*abs(a1*x[8]+a0))-1/(s0+s1*abs(a1*x[7]+a0))-1/(s0+s1*abs(a1*x[6]+a0))-1/(s0+s1*abs(a1*x[5]+a0))+(-a1*x[19]-a0+y[19])^2/(s0+s1*abs(a1*x[19]+a0))^3+(-a1*x[20]-a0+y[20])^2/(s0+s1*abs(a1*x[20]+a0))^3+(-a1*x[7]-a0+y[7])^2/(s0+s1*abs(a1*x[7]+a0))^3+(-a1*x[8]-a0+y[8])^2/(s0+s1*abs(a1*x[8]+a0))^3+(-a1*x[9]-a0+y[9])^2/(s0+s1*abs(a1*x[9]+a0))^3+(-a1*x[10]-a0+y[10])^2/(s0+s1*abs(a1*x[10]+a0))^3+(-a1*x[11]-a0+y[11])^2/(s0+s1*abs(a1*x[11]+a0))^3+(-a1*x[12]-a0+y[12])^2/(s0+s1*abs(a1*x[12]+a0))^3;

```

```

a0+y[12])^2/(s0+s1*abs(a1*x[12]+a0))^3+(-a1*x[13]-a0+y[13])^2/
(s0+s1*abs(a1*x[13]+a0))^3+(-a1*x[14]-a0+y[14])^2/(s0+s1*abs(a1*x
[14]+a0))^3+(-a1*x[15]-a0+y[15])^2/(s0+s1*abs(a1*x[15]+a0))^3+(-
a1*x[16]-a0+y[16])^2/(s0+s1*abs(a1*x[16]+a0))^3+(-a1*x[17]-a0+y
[17])^2/(s0+s1*abs(a1*x[17]+a0))^3+(-a1*x[18]-a0+y[18])^2/(s0+s1*
abs(a1*x[18]+a0))^3+(-a1*x[1]-a0+y[1])^2/(s0+s1*abs(a1*x[1]+a0))
^3+(-a1*x[2]-a0+y[2])^2/(s0+s1*abs(a1*x[2]+a0))^3+(-a1*x[3]-a0+y
[3])^2/(s0+s1*abs(a1*x[3]+a0))^3+(-a1*x[4]-a0+y[4])^2/(s0+s1*abs
(a1*x[4]+a0))^3+(-a1*x[5]-a0+y[5])^2/(s0+s1*abs(a1*x[5]+a0))^3+(-
a1*x[6]-a0+y[6])^2/(s0+s1*abs(a1*x[6]+a0))^3)*exp(-10*ln(2*Pi)-ln
(s0+s1*abs(a1*x[12]+a0))-ln(s0+s1*abs(a1*x[13]+a0))-ln(s0+s1*abs
(a1*x[14]+a0))-ln(s0+s1*abs(a1*x[15]+a0))-ln(s0+s1*abs(a1*x[16]+
a0))-ln(s0+s1*abs(a1*x[17]+a0))-ln(s0+s1*abs(a1*x[18]+a0))-ln(s0+
s1*abs(a1*x[19]+a0))-ln(s0+s1*abs(a1*x[20]+a0))-ln(s0+s1*abs(a1*x
[1]+a0))-ln(s0+s1*abs(a1*x[2]+a0))-ln(s0+s1*abs(a1*x[3]+a0))-ln
(s0+s1*abs(a1*x[4]+a0))-ln(s0+s1*abs(a1*x[5]+a0))-ln(s0+s1*abs
(a1*x[6]+a0))-ln(s0+s1*abs(a1*x[7]+a0))-ln(s0+s1*abs(a1*x[8]+a0))
-ln(s0+s1*abs(a1*x[9]+a0))-ln(s0+s1*abs(a1*x[10]+a0))-ln(s0+s1*
abs(a1*x[11]+a0))-(-a1*x[20]-a0+y[20])^2/(2*(s0+s1*abs(a1*x[20]+
a0))^2)-(-a1*x[19]-a0+y[19])^2/(2*(s0+s1*abs(a1*x[19]+a0))^2)-(-
a1*x[18]-a0+y[18])^2/(2*(s0+s1*abs(a1*x[18]+a0))^2)-(-a1*x[17]-
a0+y[17])^2/(2*(s0+s1*abs(a1*x[17]+a0))^2)-(-a1*x[16]-a0+y[16])
^2/(2*(s0+s1*abs(a1*x[16]+a0))^2)-(-a1*x[15]-a0+y[15])^2/(2*(s0+
s1*abs(a1*x[15]+a0))^2)-(-a1*x[14]-a0+y[14])^2/(2*(s0+s1*abs(a1*x
[14]+a0))^2)-(-a1*x[13]-a0+y[13])^2/(2*(s0+s1*abs(a1*x[13]+a0)))
^2)-(-a1*x[12]-a0+y[12])^2/(2*(s0+s1*abs(a1*x[12]+a0))^2)-(-a1*x
[11]-a0+y[11])^2/(2*(s0+s1*abs(a1*x[11]+a0))^2)-(-a1*x[10]-a0+y
[10])^2/(2*(s0+s1*abs(a1*x[10]+a0))^2)-(-a1*x[9]-a0+y[9])^2/(2*
(s0+s1*abs(a1*x[9]+a0))^2)-(-a1*x[8]-a0+y[8])^2/(2*(s0+s1*abs(a1*
x[8]+a0))^2)-(-a1*x[7]-a0+y[7])^2/(2*(s0+s1*abs(a1*x[7]+a0))^2)-
(-a1*x[6]-a0+y[6])^2/(2*(s0+s1*abs(a1*x[6]+a0))^2)-(-a1*x[5]-a0+y
[5])^2/(2*(s0+s1*abs(a1*x[5]+a0))^2)-(-a1*x[4]-a0+y[4])^2/(2*(s0+
s1*abs(a1*x[4]+a0))^2)-(-a1*x[3]-a0+y[3])^2/(2*(s0+s1*abs(a1*x[3]
+a0))^2)-(-a1*x[2]-a0+y[2])^2/(2*(s0+s1*abs(a1*x[2]+a0))^2)-(-a1*x
[1]-a0+y[1])^2/(2*(s0+s1*abs(a1*x[1]+a0))^2)):

> delpdels1:=((-a1*x[3]-a0+y[3])^2*abs(a1*x[3]+a0)/(s0+s1*abs(a1*x
[3]+a0))^3+(-a1*x[16]-a0+y[16])^2*abs(a1*x[16]+a0)/(s0+s1*abs(a1*
x[16]+a0))^3+(-a1*x[18]-a0+y[18])^2*abs(a1*x[18]+a0)/(s0+s1*abs
(a1*x[18]+a0))^3+(-a1*x[6]-a0+y[6])^2*abs(a1*x[6]+a0)/(s0+s1*abs
(a1*x[6]+a0))^3+(-a1*x[11]-a0+y[11])^2*abs(a1*x[11]+a0)/(s0+s1*
abs(a1*x[11]+a0))^3+(-a1*x[4]-a0+y[4])^2*abs(a1*x[4]+a0)/(s0+s1*
abs(a1*x[4]+a0))^3+(-a1*x[7]-a0+y[7])^2*abs(a1*x[7]+a0)/(s0+s1*
abs(a1*x[7]+a0))^3+(-a1*x[13]-a0+y[13])^2*abs(a1*x[13]+a0)/(s0+
s1*abs(a1*x[13]+a0))^3+(-a1*x[15]-a0+y[15])^2*abs(a1*x[15]+a0)/
(s0+s1*abs(a1*x[15]+a0))^3+(-a1*x[8]-a0+y[8])^2*abs(a1*x[8]+a0)/
(s0+s1*abs(a1*x[8]+a0))^3+(-a1*x[10]-a0+y[10])^2*abs(a1*x[10]+a0)
/(s0+s1*abs(a1*x[10]+a0))^3+(-a1*x[20]-a0+y[20])^2*abs(a1*x[20]+
a0)/(s0+s1*abs(a1*x[20]+a0))^3+(-a1*x[12]-a0+y[12])^2*abs(a1*x
[12]+a0)/(s0+s1*abs(a1*x[12]+a0))^3+(-a1*x[17]-a0+y[17])^2*abs
(a1*x[17]+a0)/(s0+s1*abs(a1*x[17]+a0))^3+(-a1*x[14]-a0+y[14])^2*
abs(a1*x[14]+a0)/(s0+s1*abs(a1*x[14]+a0))^3+(-a1*x[5]-a0+y[5])^2*
abs(a1*x[5]+a0)/(s0+s1*abs(a1*x[5]+a0))^3+(-a1*x[19]-a0+y[19])^2*
abs(a1*x[19]+a0)/(s0+s1*abs(a1*x[19]+a0))^3+(-a1*x[1]-a0+y[1])^2*
abs(a1*x[1]+a0)/(s0+s1*abs(a1*x[1]+a0))^3+(-a1*x[2]-a0+y[2])^2*
abs(a1*x[2]+a0)/(s0+s1*abs(a1*x[2]+a0))^3+(-a1*x[9]-a0+y[9])^2*
abs(a1*x[9]+a0)/(s0+s1*abs(a1*x[9]+a0))^3-abs(a1*x[6]+a0)/(s0+s1*

```

```

abs(a1*x[6]+a0))-abs(a1*x[7]+a0)/(s0+s1*abs(a1*x[7]+a0))-abs(a1*x
[8]+a0)/(s0+s1*abs(a1*x[8]+a0))-abs(a1*x[9]+a0)/(s0+s1*abs(a1*x
[9]+a0))-abs(a1*x[10]+a0)/(s0+s1*abs(a1*x[10]+a0))-abs(a1*x[11]+
a0)/(s0+s1*abs(a1*x[11]+a0))-abs(a1*x[12]+a0)/(s0+s1*abs(a1*x[12]
+a0))-abs(a1*x[13]+a0)/(s0+s1*abs(a1*x[13]+a0))-abs(a1*x[14]+a0)/
(s0+s1*abs(a1*x[14]+a0))-abs(a1*x[15]+a0)/(s0+s1*abs(a1*x[15]+a0)
)-abs(a1*x[16]+a0)/(s0+s1*abs(a1*x[16]+a0))-abs(a1*x[17]+a0)/(s0+
s1*abs(a1*x[17]+a0))-abs(a1*x[18]+a0)/(s0+s1*abs(a1*x[18]+a0))-
abs(a1*x[19]+a0)/(s0+s1*abs(a1*x[19]+a0))-abs(a1*x[20]+a0)/(s0+
s1*abs(a1*x[20]+a0))-abs(a1*x[1]+a0)/(s0+s1*abs(a1*x[1]+a0))-abs
(a1*x[2]+a0)/(s0+s1*abs(a1*x[2]+a0))-abs(a1*x[3]+a0)/(s0+s1*abs
(a1*x[3]+a0))-abs(a1*x[4]+a0)/(s0+s1*abs(a1*x[4]+a0))-abs(a1*x[5]
+a0)/(s0+s1*abs(a1*x[5]+a0)))*exp(-10*ln(2*Pi)-ln(s0+s1*abs(a1*x
[12]+a0))-ln(s0+s1*abs(a1*x[13]+a0))-ln(s0+s1*abs(a1*x[14]+a0))-
ln(s0+s1*abs(a1*x[15]+a0))-ln(s0+s1*abs(a1*x[16]+a0))-ln(s0+s1*
abs(a1*x[17]+a0))-ln(s0+s1*abs(a1*x[18]+a0))-ln(s0+s1*abs(a1*x
[19]+a0))-ln(s0+s1*abs(a1*x[20]+a0))-ln(s0+s1*abs(a1*x[1]+a0))-ln
(s0+s1*abs(a1*x[2]+a0))-ln(s0+s1*abs(a1*x[3]+a0))-ln(s0+s1*abs
(a1*x[4]+a0))-ln(s0+s1*abs(a1*x[5]+a0))-ln(s0+s1*abs(a1*x[6]+a0))
-ln(s0+s1*abs(a1*x[7]+a0))-ln(s0+s1*abs(a1*x[8]+a0))-ln(s0+s1*abs
(a1*x[9]+a0))-ln(s0+s1*abs(a1*x[10]+a0))-ln(s0+s1*abs(a1*x[11]+
a0))-(-a1*x[20]-a0+y[20])^2/(2*(s0+s1*abs(a1*x[20]+a0))^2)-(-a1*x
[19]-a0+y[19])^2/(2*(s0+s1*abs(a1*x[19]+a0))^2)-(-a1*x[18]-a0+y
[18])^2/(2*(s0+s1*abs(a1*x[18]+a0))^2)-(-a1*x[17]-a0+y[17])^2/(2*
(s0+s1*abs(a1*x[17]+a0))^2)-(-a1*x[16]-a0+y[16])^2/(2*(s0+s1*abs
(a1*x[16]+a0))^2)-(-a1*x[15]-a0+y[15])^2/(2*(s0+s1*abs(a1*x[15]+
a0))^2)-(-a1*x[14]-a0+y[14])^2/(2*(s0+s1*abs(a1*x[14]+a0))^2)-(-
a1*x[13]-a0+y[13])^2/(2*(s0+s1*abs(a1*x[13]+a0))^2)-(-a1*x[12]-
a0+y[12])^2/(2*(s0+s1*abs(a1*x[12]+a0))^2)-(-a1*x[11]-a0+y[11])
^2/(2*(s0+s1*abs(a1*x[11]+a0))^2)-(-a1*x[10]-a0+y[10])^2/(2*(s0+
s1*abs(a1*x[10]+a0))^2)-(-a1*x[9]-a0+y[9])^2/(2*(s0+s1*abs(a1*x
[9]+a0))^2)-(-a1*x[8]-a0+y[8])^2/(2*(s0+s1*abs(a1*x[8]+a0))^2)-(-
a1*x[7]-a0+y[7])^2/(2*(s0+s1*abs(a1*x[7]+a0))^2)-(-a1*x[6]-a0+y
[6])^2/(2*(s0+s1*abs(a1*x[6]+a0))^2)-(-a1*x[5]-a0+y[5])^2/(2*(s0+
s1*abs(a1*x[5]+a0))^2)-(-a1*x[4]-a0+y[4])^2/(2*(s0+s1*abs(a1*x[4]
+a0))^2)-(-a1*x[3]-a0+y[3])^2/(2*(s0+s1*abs(a1*x[3]+a0))^2)-(-a1*
x[2]-a0+y[2])^2/(2*(s0+s1*abs(a1*x[2]+a0))^2)-(-a1*x[1]-a0+y[1])
^2/(2*(s0+s1*abs(a1*x[1]+a0))^2):

```

The partial derivatives are all set to zero, and the objective function  $Z_1$  (Eq. 18) is minimized providing the optimized parameters  $a0$ ,  $a1$ ,  $s0$  and  $s1$  (see also Table 1):

```

> f:=delpdels1=0:g:=delpdela1=0:u:=delpdels0=0:v:=delpdela0=0:
> fsolve({f,g,u,v}, {s0 = 1.5 .. 2, s1 = 0.07 .. 0.1, a0 = 1.5 ..
2, a1 = 0.4 .. 0.6});
{a0 = 1.767158847, a1 = 0.4681491315, s0 = 1.723975788, s1 = 0.08721863022} (4)

```

```

> Z:=-2*ln(Prob_ex3):
> hess:=Hessian(Z, [a0, a1, s0, s1]): 

> hess0:=subs(a0= 1.767158847,hess):
> hess1:=subs(a1= .4681491315,hess0):hess2:=subs(s0= 1.723975788,
hess1):hess3:=subs(s1= 0.8721863022e-1,hess2):

```

The Hessian matrix is calculated for the minimum of the objective function  $Z1$ :

```
> hess_matrix:=evalf(hess3);
```

$$hess\_matrix := \begin{bmatrix} 5.576189343 & 86.25003034 & 0.6919358702 & 4.007823943 \\ 86.25003034 & 4799.968285 & 27.54237605 & 240.2800389 \\ 0.6919358702 & 27.54237605 & 9.997150598 & 103.2304434 \\ 4.007823943 & 240.2800389 & 103.2304434 & 2529.680397 \end{bmatrix} \quad (5)$$

The covariance matrix (inverse of the Hessian matrix) is calculated:

```
> cov:=MatrixInverse(hess_matrix);
cov := [[ 0.248839419565509, -0.00443904009413467, -0.00911868397342866,
0.000399510520696810 ],
[ -0.00443904009413467, 0.000290928587417226, -0.000486589630174699,
-7.44195928457812 10-7 ],
[ -0.00911868397342866, -0.000486589630174698, 0.175199154424154,
-0.00708880954311573 ],
[ 0.000399510520696810, -7.44195928457812 10-7, -0.00708880954311573,
0.000684022614857737 ]]
```

```
> with(Statistics);
[AbsoluteDeviation, AgglomeratedPlot, AreaChart, AutoCorrelation, BarChart, Biplot,
Bootstrap, BoxPlot, BubblePlot, CDF, CGF, CentralMoment, CharacteristicFunction,
ChiSquareGoodnessOfFitTest, ChiSquareIndependenceTest, ChiSquareSuitableModelTest,
ColumnGraph, Correlation, CorrelationMatrix, Count, CountMissing, Covariance,
CovarianceMatrix, CrossCorrelation, Cumulant, CumulantGeneratingFunction,
CumulativeDistributionFunction, CumulativeProduct, CumulativeSum,
CumulativeSumChart, DataSummary, Decile, DensityPlot, DiscreteValueMap, Distribution,
ErrorPlot, EvaluateToFloat, Excise, ExpectedValue, ExponentialFit,
ExponentialSmoothing, FailureRate, FisherInformation, Fit, FivePointSummary,
FrequencyPlot, FrequencyTable, GeometricMean, GridPlot, HarmonicMean, HazardRate,
HeatMap, Histogram, HodgesLehmann, Information, InteractiveDataAnalysis,
InterquartileRange, InverseSurvivalFunction, Join, KernelDensity, KernelDensityPlot,
KernelDensitySample, Kurtosis, Likelihood, LikelihoodRatioStatistic, LineChart,
LinearFilter, LinearFit, LogLikelihood, LogarithmicFit, Lowess, MGF, MLE,
MakeProcedure, MaximumLikelihoodEstimate, Mean, MeanDeviation, Median,
MedianDeviation, MillsRatio, Mode, Moment, MomentGeneratingFunction,
MovingAverage, MovingMedian, MovingStatistic, NonlinearFit, NormalPlot,
OneSampleChiSquareTest, OneSampleTTest, OneSampleZTest, OneWayANOVA,
OrderByRank, OrderStatistic, PCA, PDF, ParetoChart, Percentile, PieChart, PointPlot,
PolynomialFit, PowerFit, PredictiveLeastSquares, PrincipalComponentAnalysis,
Probability, ProbabilityDensityFunction, ProbabilityFunction, ProbabilityPlot,
ProfileLikelihood, ProfileLogLikelihood, QuadraticMean, Quantile, QuantilePlot, Quartile,
RandomVariable, Range, Rank, Remove, RemoveInRange, RemoveNonNumeric,
RepeatedMedianEstimator, RousseeuwCrouxQn, RousseeuwCrouxSn, Sample, Scale,
ScatterPlot, ScatterPlot3D, Score, ScreePlot, Select, SelectInRange, SelectNonNumeric,
ShapiroWilkWTest, Shuffle, Skewness, Sort, Specialize, SplitByColumn, StandardDeviation,
```

*StandardError, StandardizedMoment, SunflowerPlot, Support, SurfacePlot, SurvivalFunction, SymmetryPlot, Tally, TallyInto, TreeMap, Trim, TrimmedMean, TwoSampleFTest, TwoSamplePairedTTest, TwoSampleTTest, TwoSampleZTest, Variance, Variation, VennDiagram, ViolinPlot, WeibullPlot, WeightedMovingAverage, Winsorize, WinsorizedMean]*

The correlation coefficients (Eq. 24) follow from the covariance matrix (compare Table 2):

```

> rho12:=cov(1,2)/sqrt(cov(1,1))/sqrt(cov(2,2)); rho13:=cov(1,3)
/sqrt(cov(1,1))/sqrt(cov(3,3)); rho14:=cov(1,4)/sqrt(cov(1,1))
/sqrt(cov(4,4));rho23:=cov(2,3)/sqrt(cov(2,2))/sqrt(cov(3,3));
rho24:=cov(2,4)/sqrt(cov(2,2))/sqrt(cov(4,4));rho34:=cov(3,4)
/sqrt(cov(3,3))/sqrt(cov(4,4));

```

$$\begin{aligned}\rho_{12} &:= -0.521718550137058 \\ \rho_{13} &:= -0.0436723510109006 \\ \rho_{14} &:= 0.0306219893043441 \\ \rho_{23} &:= -0.0681559298616130 \\ \rho_{24} &:= -0.00166824126790481\end{aligned}\tag{8}$$

The uncertainties of the parameters follow from the diagonal terms of the covariance matrix (compare Table 1):

```
> dela0:=sqrt(cov(1,1));
dela0 := 0.498838069482982
```

```
> dela1:=sqrt(cov(2,2));
          dela1 := 0.0170566288409294
```

```
> dels0:=sqrt(cov(3,3));
dels0 := 0.418567980648489
```

```
> dels1:=sqrt(cov(4,4));
dels1 := 0.0261538260080191
```

```
> with(LinearAlgebra);
[&x, Add, Adjoint, BackwardSubstitute, BandMatrix, Basis, BezoutMatrix, BidiagonalForm, (13)
```

*BilinearForm, CARE, CharacteristicMatrix, CharacteristicPolynomial, Column, ColumnDimension, ColumnOperation, ColumnSpace, CompanionMatrix, CompressedSparseForm, ConditionNumber, ConstantMatrix, ConstantVector, Copy, CreatePermutation, CrossProduct, DARE, DeleteColumn, DeleteRow, Determinant, Diagonal, DiagonalMatrix, Dimension, Dimensions, DotProduct, EigenConditionNumbers, Eigenvalues, Eigenvectors, Equal, ForwardSubstitute, FrobeniusForm, FromCompressedSparseForm, FromSplitForm, GaussianElimination, GenerateEquations, GenerateMatrix, Generic, GetResultDataType, GetResultShape, GivensRotationMatrix, GramSchmidt, HankelMatrix, HermiteForm, HermitianTranspose, HessenbergForm, HilbertMatrix, HouseholderMatrix, IdentityMatrix, IntersectionBasis, IsDefinite, IsOrthogonal, IsSimilar, IsUnitary, JordanBlockMatrix, JordanForm, KroneckerProduct, LA\_Main, LUDecomposition, LeastSquares, LinearSolve, LyapunovSolve, Map, Map2, MatrixAdd, MatrixExponential, MatrixFunction, MatrixInverse, MatrixMatrixMultiply, MatrixNorm, MatrixPower, MatrixScalarMultiply, MatrixVectorMultiply, MinimalPolynomial, Minor, Modular, Multiply, NoUserValue, Norm, Normalize, NullSpace, OuterProductMatrix, Permanent, Pivot, PopovForm, ProjectionMatrix,*

*QRDecomposition, RandomMatrix, RandomVector, Rank, RationalCanonicalForm,  
ReducedRowEchelonForm, Row, RowDimension, RowOperation, RowSpace, ScalarMatrix,  
ScalarMultiply, ScalarVector, SchurForm, SingularValues, SmithForm, SplitForm,  
StronglyConnectedBlocks, SubMatrix, SubVector, SumBasis, SylvesterMatrix,  
SylvesterSolve, ToeplitzMatrix, Trace, Transpose, TridiagonalForm, UnitVector,  
VandermondeMatrix, VectorAdd, VectorAngle, VectorMatrixMultiply, VectorNorm,  
VectorScalarMultiply, ZeroMatrix, ZeroVector, Zip ]*

> Determinant(hess\_matrix);  
2.777231249 10<sup>8</sup> (14)

> ConditionNumber(hess\_matrix);  
1354.464659 (15)