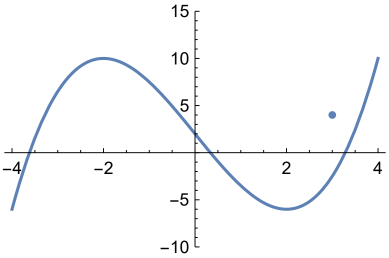
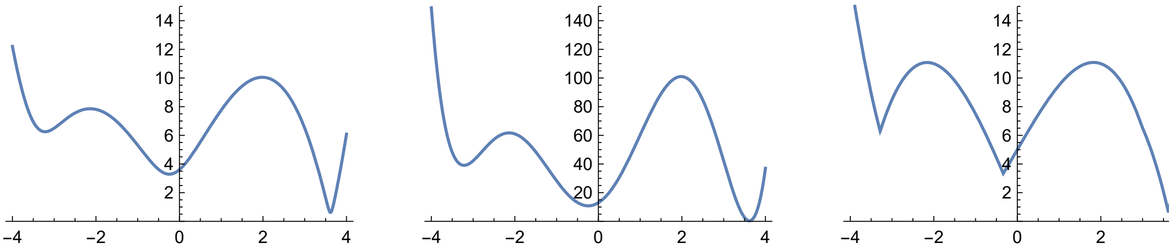
Vjezbe 2. Uvod i motivacija

Primjer 2. i 3. (Udaljenost tocke do parabole)

```
q[x_] := .5 x^3 - 6 x + 2;
a = - 4; b = 4;
{x0, y0} = {3 , 4};
slq = Plot[q[x], {x, a, b}, AxesOrigin → {0, 0}, PlotRange → {-10, 15}];
slp = ListPlot[{{x0, y0}}, PlotStyle -> AbsolutePointSize[4]];
sl1 = Show[slq, slp, ImageSize → 200]
```



```
deltaLS[x_] := (x - x0) ^2 + (q[x] - y0) ^2
sld2 = Plot[Sqrt[deltaLS[x]], {x, a, b}, PlotRange → {0, 15}, ImageSize → 200];
sldLS = Plot[deltaLS[x], {x, a, b}, PlotRange → {0, 150}, ImageSize → 200];
delta1[x_] := Abs[x - x0] + Abs[q[x] - y0]
sld1 = Plot[delta1[x], {x, a, b}, PlotRange → {0, 15}, ImageSize → 200];
GraphicsGrid[{{sld2, sldLS, sld1}}]
```



```
Clear[x]
NMinimize[deltaLS[x], x]
NMinimize[{deltaLS[x], 2 < x < 4}, x]
FindMinimum[deltaLS[x], {x, 3}]
```

```
{10.8186, {x → -0.243094}}

{0.382438, {x → 3.61676}}

{0.382438, {x → 3.61676}}
```

```
NMinimize[delta1[x], x]
FindMinimum[delta1[x], {x, 3}]
```

```
{3.33651, {x → -0.336509}}
```

FindMinimum::lstol :

The line search decreased the step size to within the tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the function. You may need more than MachinePrecision digits of working precision to meet these tolerances. >>

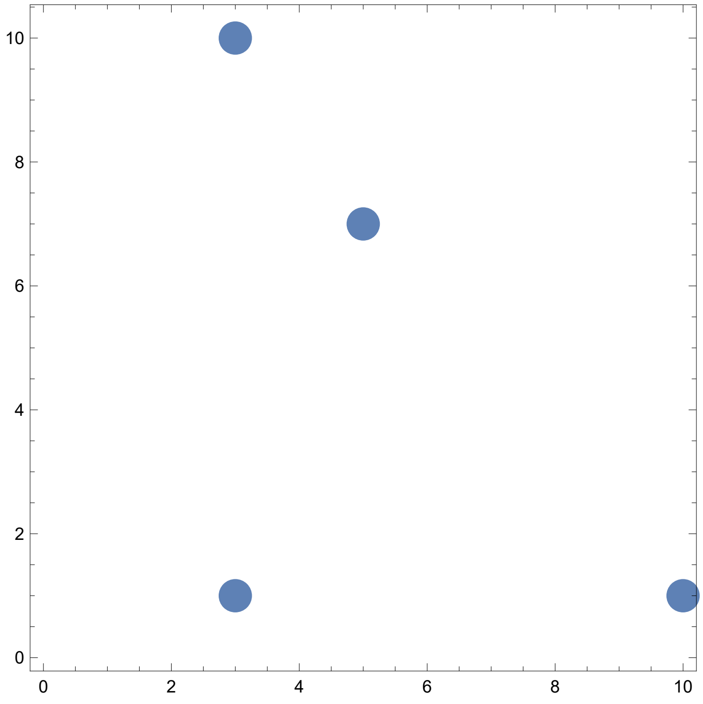
```
{0.620076, {x → 3.62008}}
```

Primjer 4. (Centroid)

Uzmimo da je n = 2, m broj točaka. Najprije ćemo generirati točke.

```
SeedRandom[18]
m = 4; p = 2;
tezine = ConstantArray[1, m];
tocke = RandomInteger[{0, 10}, {m, 2}]
sltocke = ListPlot[tocke, PlotStyle → {PointSize[0.05]},
  AspectRatio → 1, AxesOrigin → {0, 0}, Frame → True]
```

```
{{5, 7}, {3, 1}, {3, 10}, {10, 1}}
```



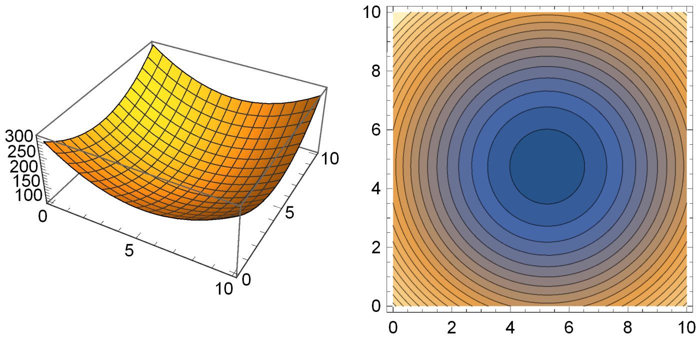
```
FLS[u_, v_] :=
  Sum[tezine[[i]] * ((u - tocke[[i]][[1]]) ^2 + (v - tocke[[i]][[2]]) ^2), {i, 1, m}];
minLS = NMinimize[FLS[u, v], {u, v}]
Options[NMinimize]
NMinimize[FLS[u, v], {u, v}, Method → "NelderMead"]
NMinimize[FLS[u, v], {u, v}, Method → "SimulatedAnnealing"]
GraphicsGrid[{{Plot3D[FLS[u, v], {u, 0, 10}, {v, 0, 10}],
   ContourPlot[FLS[u, v], {u, 0, 10}, {v, 0, 10}, Contours → 20]}}]
Show[{sltocke,
  Graphics[{PointSize[0.05], Red, Point[{u /. minLS[[2]], v /. minLS[[2]]}]}]}]
```

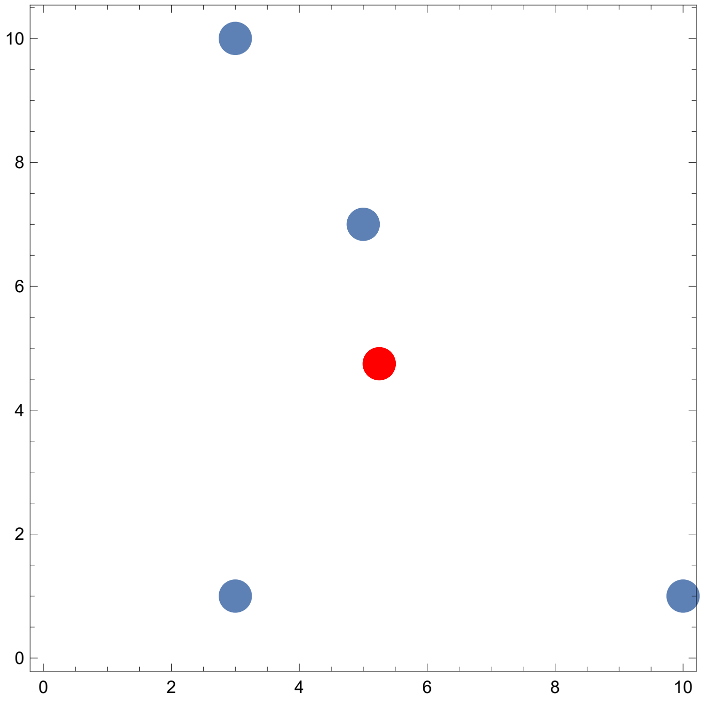
```
{93.5, {u → 5.25, v → 4.75}}
```

```
{AccuracyGoal → Automatic, EvaluationMonitor → None,
 MaxIterations → 100, Method → Automatic, PrecisionGoal → Automatic,
 StepMonitor → None, WorkingPrecision → MachinePrecision}
```

```
{93.5, {u → 5.25, v → 4.75}}
```

```
{93.5, {u → 5.25, v → 4.75}}
```



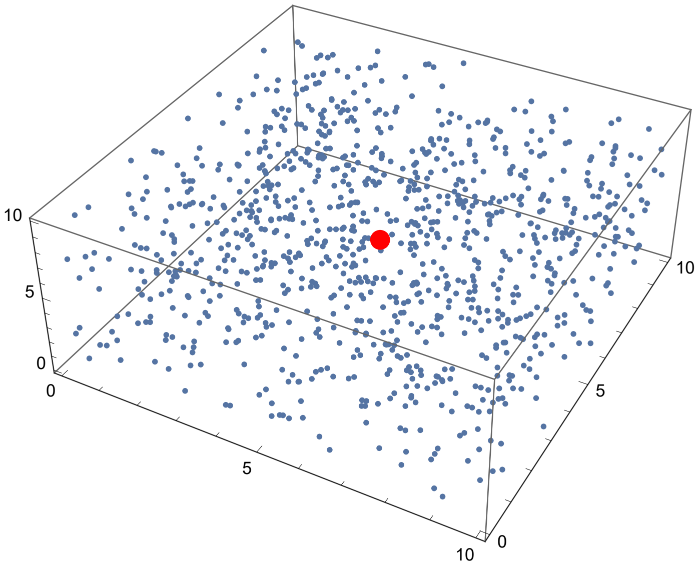


```
minLS
Mean[tocke[[All, 1]]] // N
Mean[tocke[[All, 2]]] // N
```

```
{93.5, {u → 5.25, v → 4.75}}

5.25

4.75
```

Primjer s više točaka:

```
SeedRandom[1];
tocke3D = RandomReal[{0, 10}, {1000, 3}];
sltocke3D = ListPointPlot3D[tocke3D];
centroid = Mean[tocke3D];
slcentroid = ListPointPlot3D[{centroid}, PlotStyle → {Red, PointSize[0.03]}];
Show[{sltocke3D, slcentroid}]
```



Primjer 5. (Medijan)

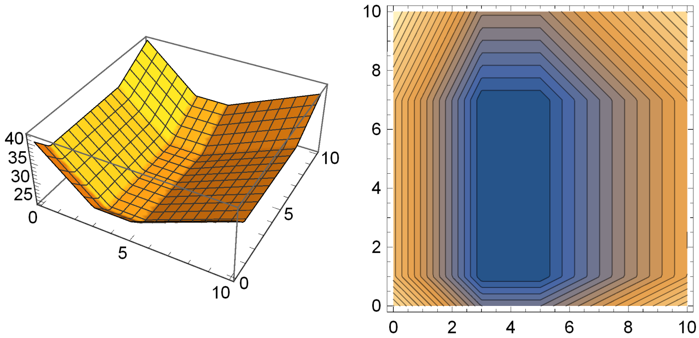
Tocke isto generiramo.

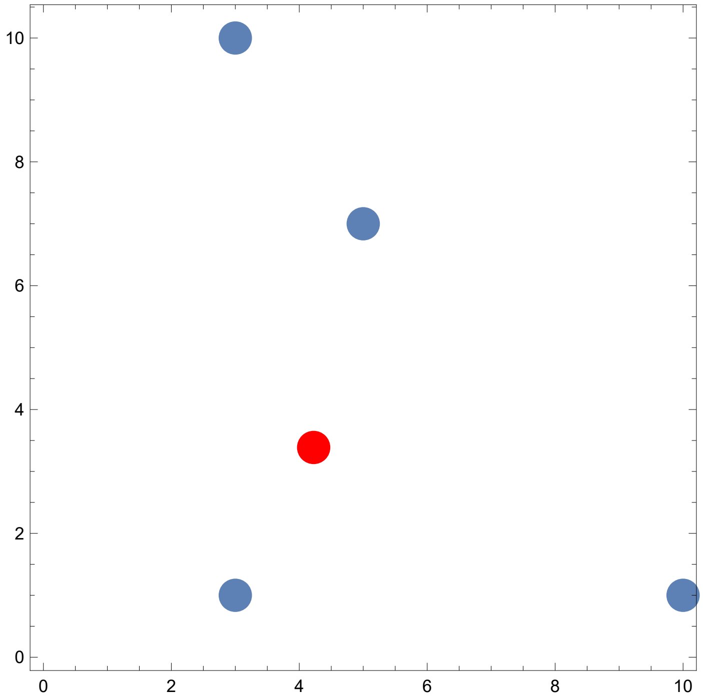
```
SeedRandom[18]
m = 4; p = 2;
tezine = ConstantArray[1, m];
tocke = RandomInteger[{0, 10}, {m, 2}]
ListPlot[tocke, PlotStyle → {PointSize[0.05]},
  AspectRatio → 1, AxesOrigin → {0, 0}, Frame → True];
```

```
{{5, 7}, {3, 1}, {3, 10}, {10, 1}}
```

```
F1[u_, v_] := Sum[
   tezine[[i]] * (Abs[u - tocke[[i]][[1]]] + Abs[v - tocke[[i]][[2]]]), {i, 1, m}];
min1 = NMinimize[F1[u, v], {u, v}]
GraphicsGrid[{{Plot3D[F1[u, v], {u, 0, 10}, {v, 0, 10}],
   ContourPlot[F1[u, v], {u, 0, 10}, {v, 0, 10}, Contours → 20]}}]
Show[{ListPlot[tocke, PlotStyle → {PointSize[0.05]},
   AspectRatio → 1, AxesOrigin → {0, 0}, Frame → True],
  Graphics[{PointSize[0.05], Red, Point[{u /. min1[[2]], v /. min1[[2]]}]}]}]
```

```
{24., {u → 4.22483, v → 3.38732}}
```



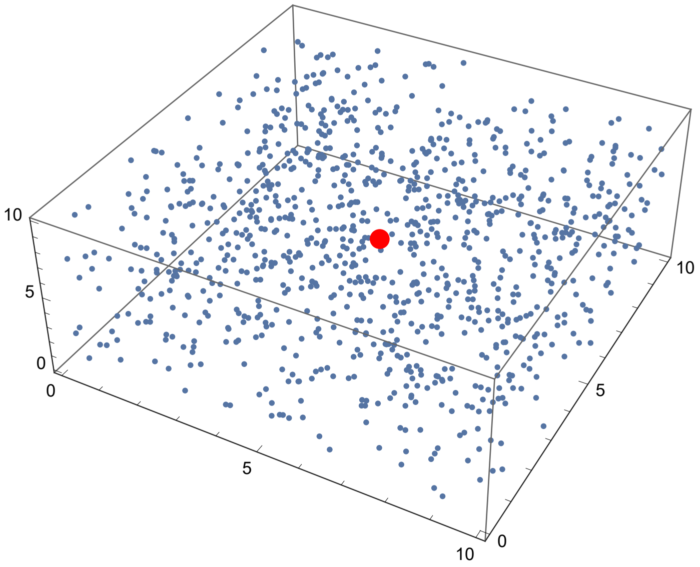


```
min1
Median[tocke[[All, 1]]] // N
Median[tocke[[All, 2]]] // N
Sort[tocke[[All, 1]]]
Sort[tocke[[All, 2]]]
```

```
{24., {u → 4.22483, v → 3.38732}}

4.

4.

{3, 3, 5, 10}

{1, 1, 7, 10}
```

Primjer s više točaka:

```
SeedRandom[1];
tocke3D = RandomReal[{0, 10}, {1000, 3}];
sltocke3D = ListPointPlot3D[tocke3D];
median = Median[tocke3D];
slmedian = ListPointPlot3D[{median}, PlotStyle → {Red, PointSize[0.03]}];
Show[{sltocke3D, slmedian}]
```



Primjer 6. (Geometrijski medijan)

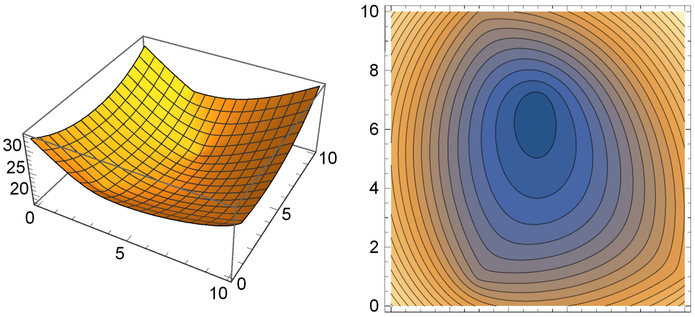
Tocke isto generiramo.

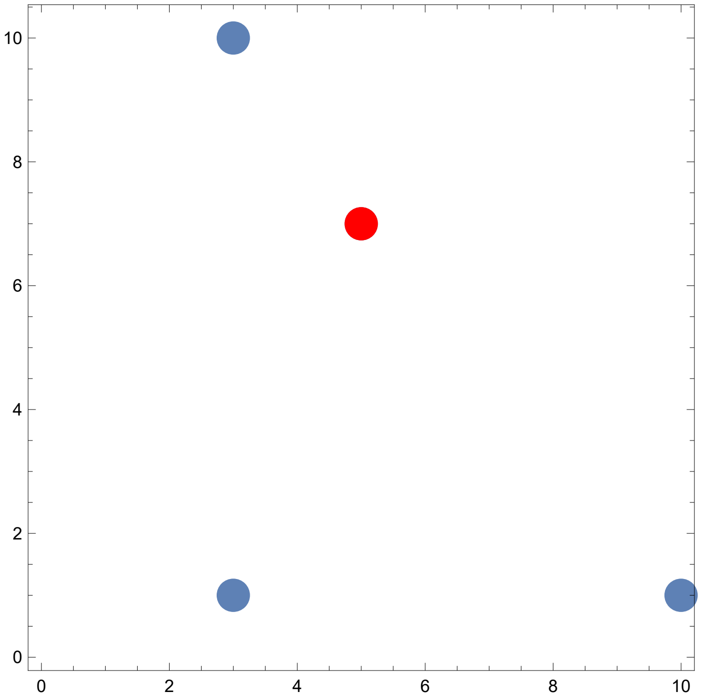
```
SeedRandom[18]
m = 4; p = 2;
tezine = ConstantArray[1, m];
tocke = RandomInteger[{0, 10}, {m, 2}]
ListPlot[tocke, PlotStyle → {PointSize[0.05]},
  AspectRatio → 1, AxesOrigin → {0, 0}, Frame → True];
```

```
{{5, 7}, {3, 1}, {3, 10}, {10, 1}}
```

```
F2[u_, v_] := Sum[tezine[[i]] *
     Sqrt[((u - tocke[[i]][[1]])^2 + (v - tocke[[i]][[2]])^2)], {i, 1, m}];
min2 = NMinimize[F2[u, v], {u, v}]
GraphicsGrid[{{Plot3D[F2[u, v], {u, 0, 10}, {v, 0, 10}],
    ContourPlot[F2[u, v], {u, 0, 10}, {v, 0, 10}, Contours → 20]}}]
Show[{ListPlot[tocke, PlotStyle → {PointSize[0.05]},
    AspectRatio → 1, AxesOrigin → {0, 0}, Frame → True],
  Graphics[{PointSize[0.05], Red, Point[{u /. min2[[2]], v /. min2[[2]]}]}]}]
```

```
{17.7404, {u → 5., v → 7.}}
```



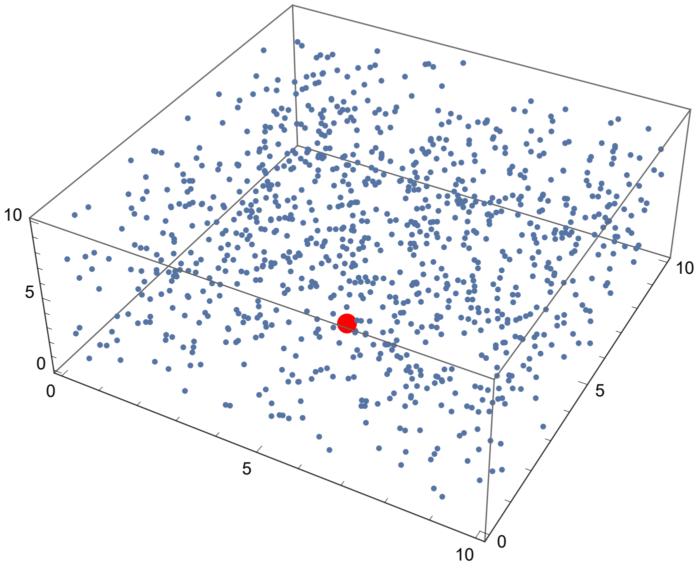


Ovdje ne znamo eksplicitni izraz.

Primjer s više točaka:

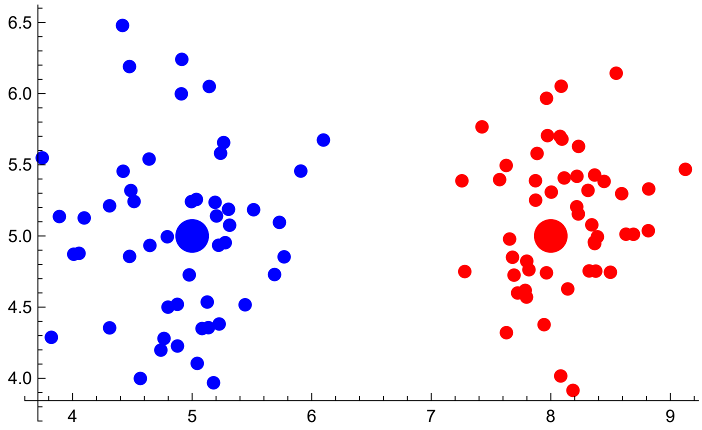
```
SeedRandom[1];
tocke3D = RandomReal[{0, 10}, {1000, 3}];
sltocke3D = ListPointPlot3D[tocke3D];
F2[u_, v_, z_] :=
  Sum[Sqrt[((u - tocke3D[[i]][[1]])^2 + (v - tocke3D[[i]][[2]])^2 +
       (z - tocke3D[[i]][[3]])^2)], {i, 1, m}];
min2 = NMinimize[F2[u, v, z], {u, v, z}]
geommedian = {u, v, z} /. min2[[2]];
slgeommedian =
  ListPointPlot3D[{geommedian}, PlotStyle → {Red, PointSize[0.03]}];
Show[{sltocke3D, slgeommedian}]
```

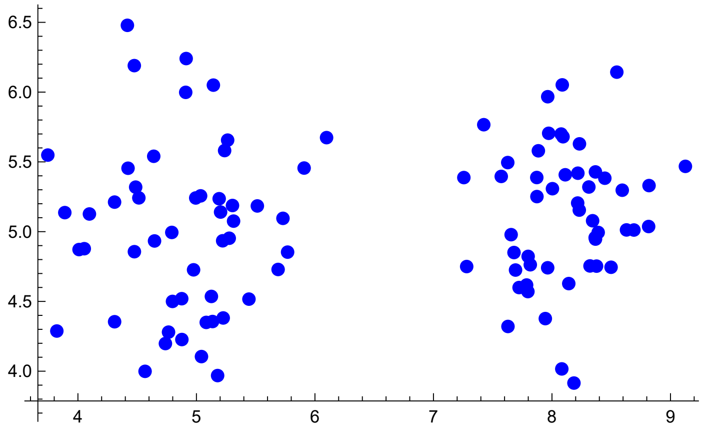
```
{13.6199, {u → 5.88707, v → 2.07715, z → 5.1061}}
```



Primjer 7. (Grupiranje podataka u k klastera)

```
SeedRandom[10];
g1 = Table[{5 + RandomVariate[NormalDistribution[0, 0.5]],
     5 + RandomVariate[NormalDistribution[0, 0.5]]}, {i, 50}];
g2 = Table[{8 + RandomVariate[NormalDistribution[0, 0.5]],
     5 + RandomVariate[NormalDistribution[0, 0.5]]}, {i, 50}];
Show[{ListPlot[g1, PlotStyle → {Blue, PointSize[0.02]}],
   ListPlot[g2, PlotStyle → {Red, PointSize[0.02]}],
   Graphics[{Blue, PointSize[0.05], Point[{5, 5}], Red,
     PointSize[0.05], Point[{8, 5}]}]}, PlotRange → All]
tocke = Join[g1, g2];
m = Length[tocke];
sltocke =
  Show[ListPlot[tocke, PlotStyle → {Blue, PointSize[0.02]}], PlotRange → All]
```





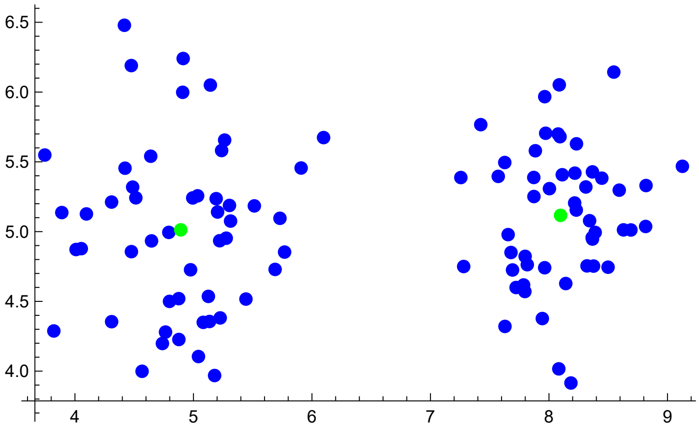
Treba na osnovu druge slike podatke grupirati u dvije skupine k = 2, dakle tražimo dva centra (4 varijable). Neka su tezine 1 a udaljenost d least squares.

```
F[x1_, y1_, x2_, y2_] :=
 Sum[Min[(x1 - tocke[[i]][[1]])^2 + (y1 - tocke[[i]][[2]])^2,
   (x2 - tocke[[i]][[1]])^2 + (y2 - tocke[[i]][[2]])^2], {i, 1, m}]
min1 = NMinimize[F[x1, y1, x2, y2], {x1, y1, x2, y2}]
(*potpuno pogrešna aproksimacija*)
min2 =
 NMinimize[{F[x1, y1, x2, y2], Min[tocke[[All, 1]]] ≤ x1 ≤ Max[tocke[[All, 1]]],
   Min[tocke[[All, 1]]] ≤ x2 <= Max[tocke[[All, 1]]],
   Min[tocke[[All, 2]]] ≤ y1 <= Max[tocke[[All, 2]]],
   Min[tocke[[All, 2]]] ≤ y2 <= Max[tocke[[All, 2]]]}, {x1, y1, x2, y2}]
FindMinimum[F[x1, y1, x2, y2], {x1, y1, x2, y2}] (*lokalni minimum*)
Show[{sltocke, ListPlot[{{x1, y1}, {x2, y2}} /. min2[[2]],
   PlotStyle → {Green, PointSize[0.02]}]}]
```

```
{308.988, {x1 → 6.4956, y1 → 5.0645, x2 → -4.50304, y2 → -1.31573}}

{51.9312, {x1 → 8.09805, y1 → 5.11644, x2 → 4.89314, y2 → 5.01256}}
```

```
FindMinimum::lstol :
  The line search decreased the step size to within the tolerance specified by AccuracyGoal and PrecisionGoal
      but was unable to find a sufficient decrease in the function. You may need more
      than MachinePrecision digits of working precision to meet these tolerances. >>
```

```
{308.988, {x1 → 6.49559, y1 → 5.0645, x2 → 6.49559, y2 → 5.0645}}
```

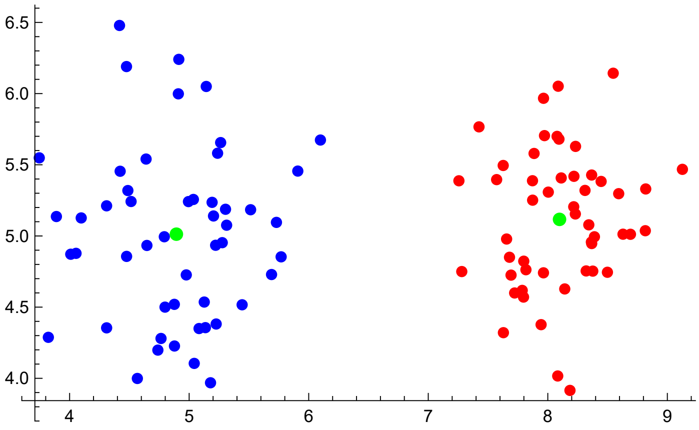


```
FindMinimum[F[x1, y1, x2, y2], {{x1, 5}, {y1, 5}, {x2, 8}, {y2, 5}}]
```

```
FindMinimum::lstol :
  The line search decreased the step size to within the tolerance specified by AccuracyGoal and PrecisionGoal
      but was unable to find a sufficient decrease in the function. You may need more
      than MachinePrecision digits of working precision to meet these tolerances. >>
```

```
{51.9312, {x1 → 4.89314, y1 → 5.01256, x2 → 8.09805, y2 → 5.11644}}
```

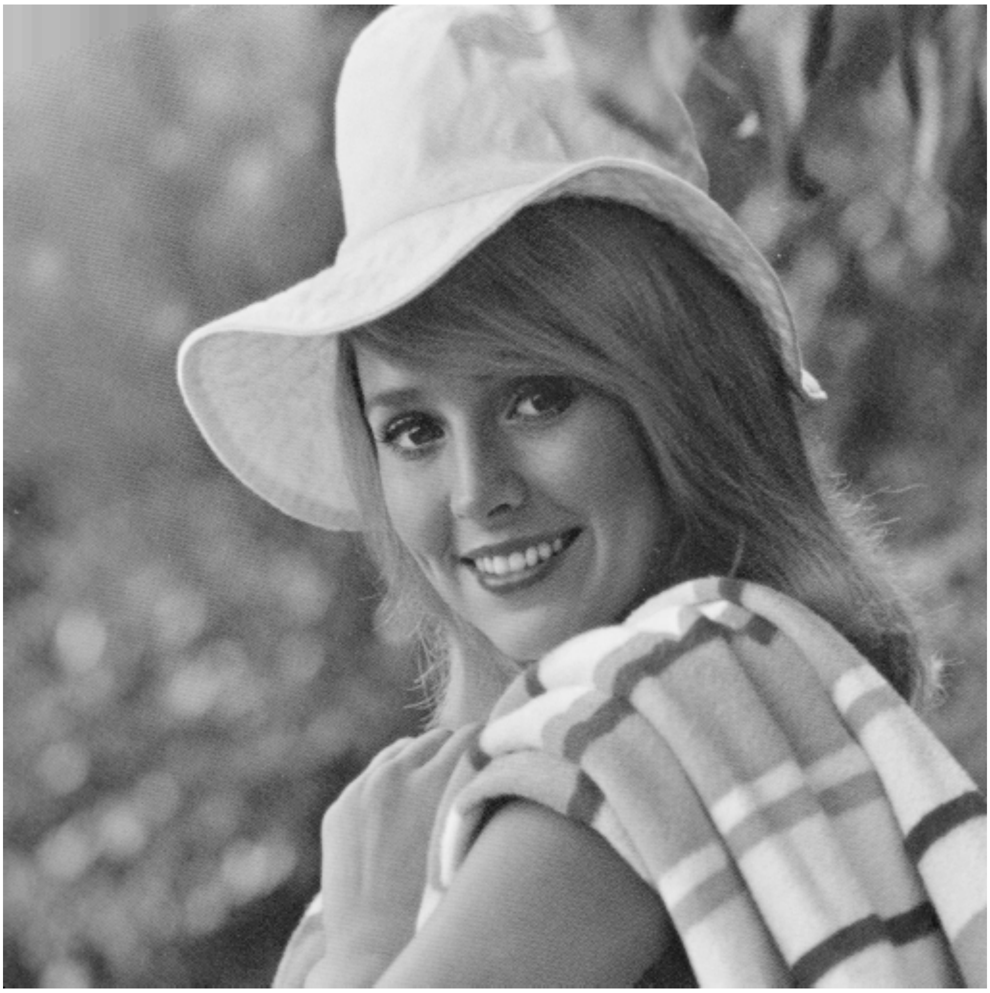
```
klasteri = FindClusters[tocke, 2];
Show[{ListPlot[klasteri[[1]], PlotStyle → Blue],
  ListPlot[klasteri[[2]], PlotStyle → Red],
  ListPlot[{Mean[klasteri[[1]]], Mean[klasteri[[2]]]},
   PlotStyle → {Green, PointSize[0.02]}]}, PlotRange → All]
Mean[klasteri[[1]]]
Mean[klasteri[[2]]]
```



```
{4.89314, 5.01256}

{8.09805, 5.11644}
```

Primjer 8. (Segmentacija crno-bijele slike)

```
slika = ExampleData[{"TestImage", "Elaine"}]
ImageDimensions[slika]
```



```
{512, 512}
```

Matrica dimenzije 512*512 u kojoj je svaki element nijansa sive boje odgovarajućeg piksela:

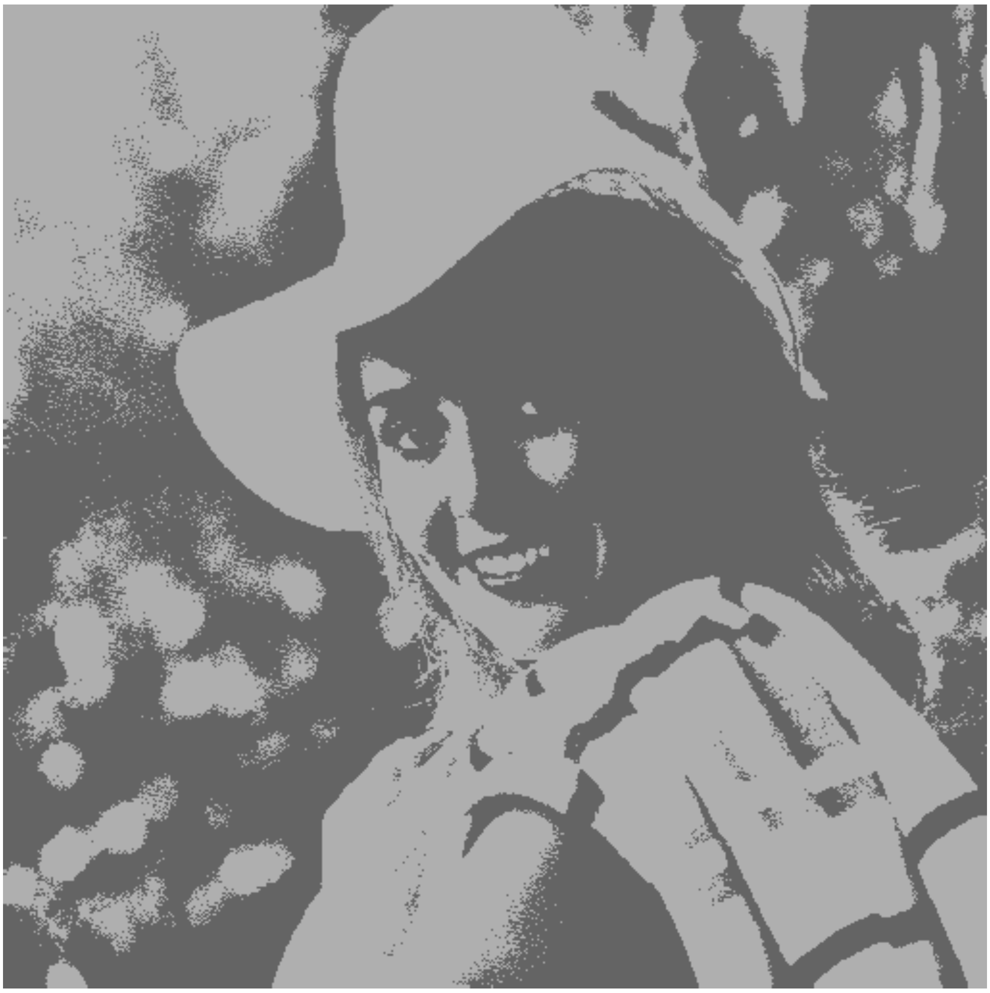
```
slikamat = ImageData[slika];
```

Segmentacija u k klastera

Nalazimo k=2 tipična tona sive boje za sliku, zatim svaki podatak zamijenimo odgovarajućim centrom i prikažemo sliku:

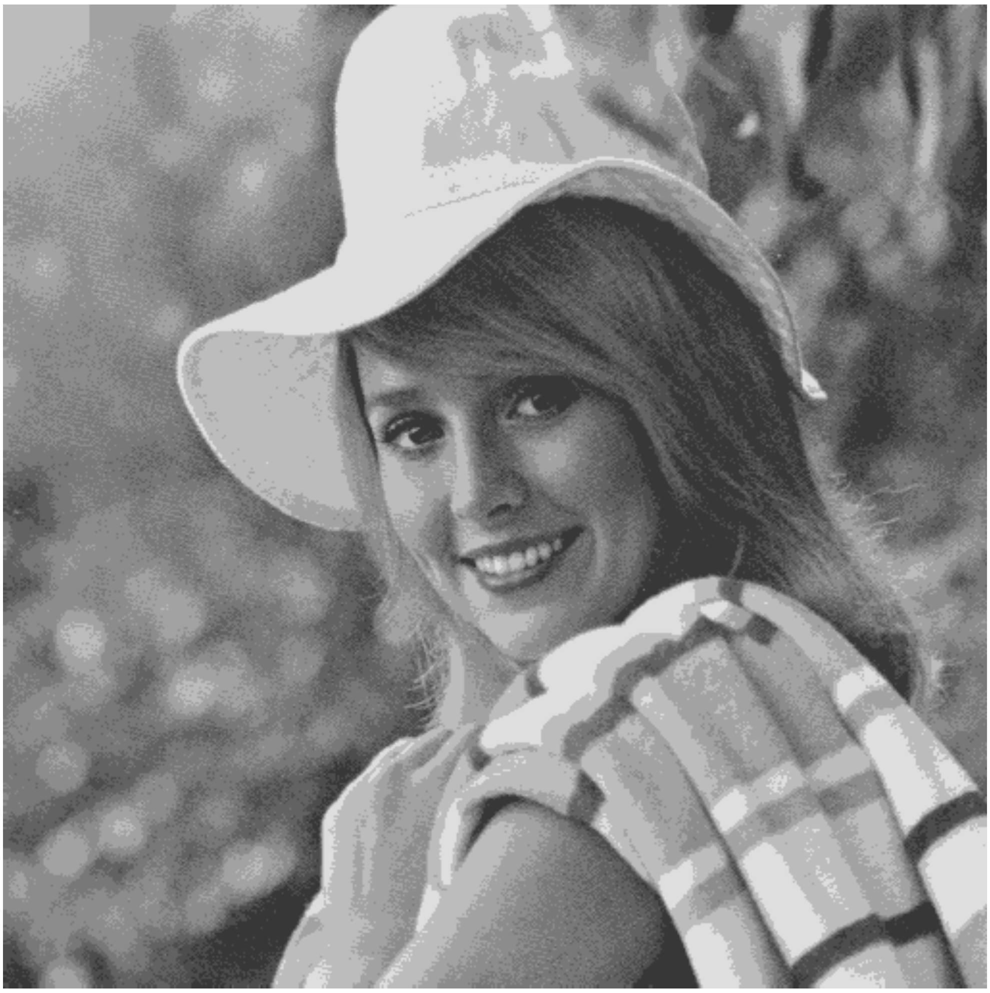
```
k = 2;
klasteri = ClusteringComponents[slikamat,
   k, 2, DistanceFunction → SquaredEuclideanDistance];
 (*2 oznacava na kojoj razini arraya trazi klastere*)
centri = Table[i -> Mean[Flatten[Pick[slikamat, klasteri, i]]], {i, 1, k}]
novaslika = Image[klasteri /. centri]
```

```
{1 → 0.687443, 2 → 0.392438}
```



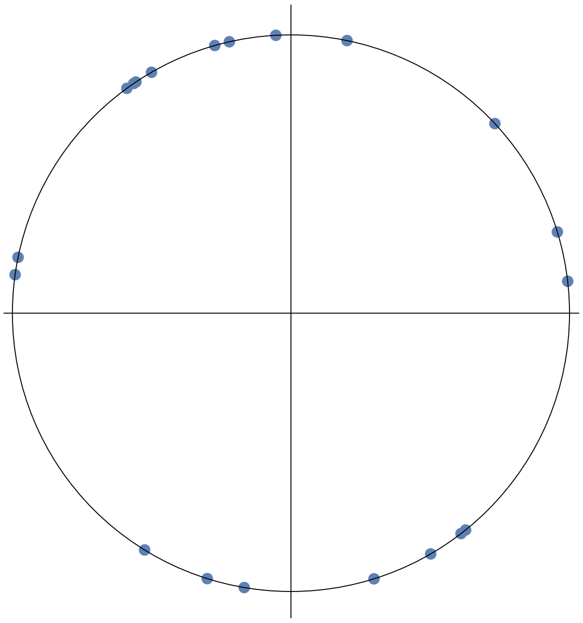
```
k = 8;
klasteri = ClusteringComponents[slikamat,
   k, 2, DistanceFunction → SquaredEuclideanDistance];
centri = Table[i -> Mean[Flatten[Pick[slikamat, klasteri, i]]], {i, 1, k}]
novaslika = Image[klasteri /. centri]
```

```
{1 → 0.735571, 2 → 0.63521, 3 → 0.548267, 4 → 0.46895,
 5 → 0.385511, 6 → 0.303582, 7 → 0.215294, 8 → 0.869886}
```



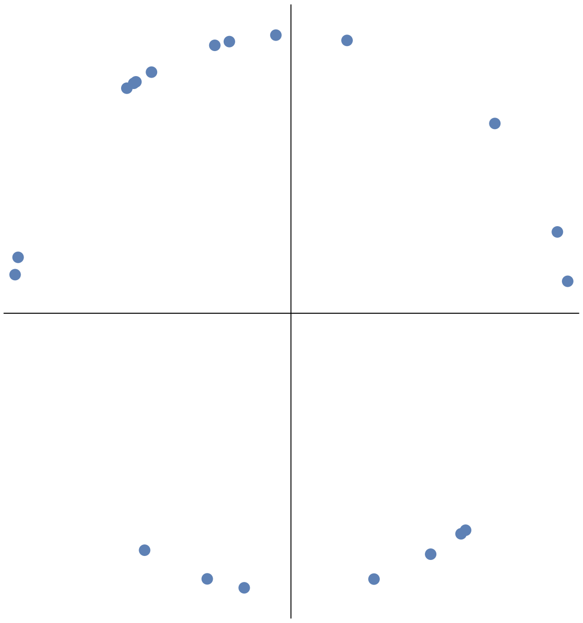
Primjer 10. (Reprezentant na jedinicnoj kruznici)

```
SeedRandom[11]
m = 20;
toc = RandomReal[{0, 2 Pi}, m];
tocke = Table[{toc[[i]], {Cos[toc[[i]]], Sin[toc[[i]]]}}, {i, m}];
sl0 = Graphics[Circle[{0, 0}, 1]];
sl1 = ListPlot[tocke[[All, 2]], PlotStyle → {PointSize[.02]},
   AspectRatio → Automatic, Axes → True, Ticks → None, ImageSize → 300];
Show[{sl1, sl0}]
```

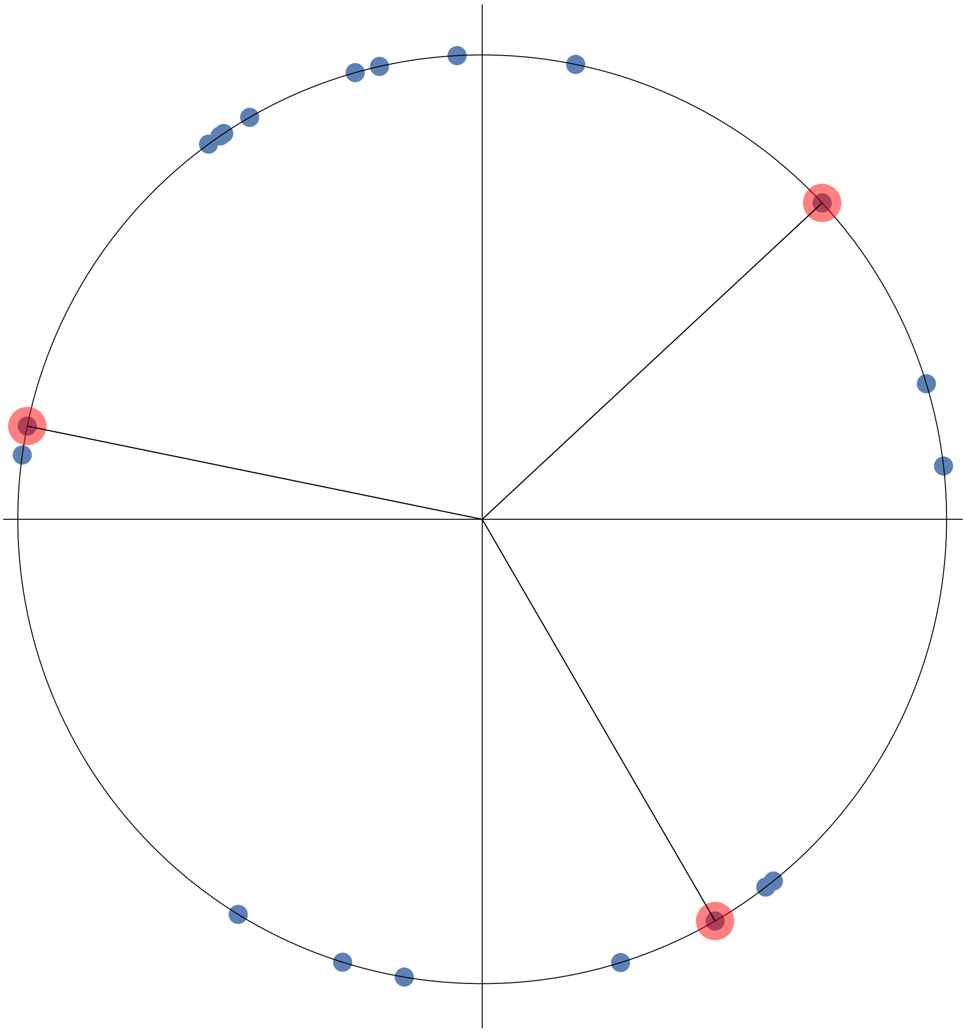


Kako izgleda udaljenost na kruznici?

```
sl1 = ListPlot[Table[Tooltip[tocke[[i, 2]], i], {i, 1, m}],
  PlotStyle → {PointSize[.02]}, AspectRatio → Automatic,
  Axes → True, Ticks → None, ImageSize → 300]
```

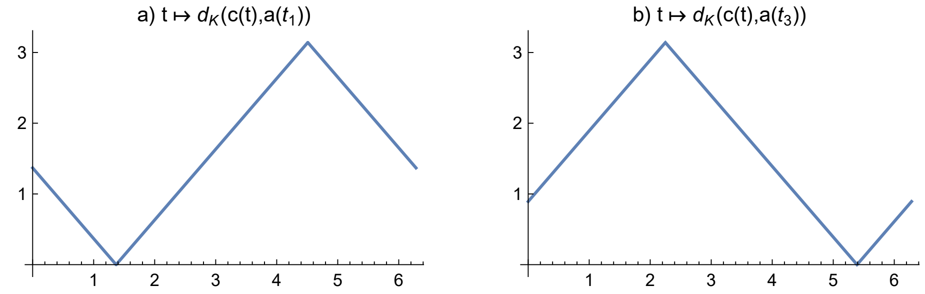


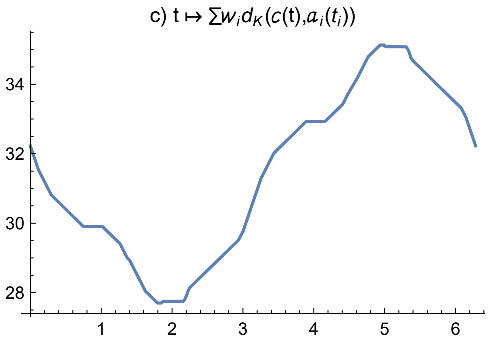
```
t0 = {0, 0};
t1 = tocke[[12, 2]];
t2 = tocke[[2, 2]];
t3 = tocke[[13, 2]];
sl2 = Graphics[{Red, PointSize[.04], Opacity[.5], Point[{t1, t2, t3}]}];
sl3 = Graphics[{Line[{{t0, t1}, {t0, t1}}],
    Line[{{t0, t2}, {t0, t2}}], Line[{{t0, t3}, {t0, t3}}]}];
Show[{sl1, sl0, sl2, sl3}, ImageSize → 500]
```



Udaljenost i minimizirajuca funkcija

```
f[t1_, t2_] := If[Abs[t2 - t1] ≤ Pi, Abs[t2 - t1], 2 Pi - Abs[t2 - t1]];
sl1 = Plot[f[t, toc[[1]]], {t, 0, 2 Pi}, Ticks → {Automatic, {1, 2, 3}},
   ImageSize → 180, PlotLabel -> "a)  t ↦ dK(c(t),a(t1))"];
sl2 = Plot[f[t, toc[[3]]], {t, 0, 2 Pi}, Ticks → {Automatic, {1, 2, 3}},
   ImageSize → 180, PlotLabel -> "b)  t ↦ dK(c(t),a(t3))"];
GraphicsGrid[{{sl1, sl2}}, ImageSize → 500]
F[t_] := Sum[f[t, toc[[i]]], {i, m}];
Plot[F[t], {t, 0, 2 Pi},
 PlotLabel -> "c)  t ↦ ∑wi dK(c(t),ai(ti))", ImageSize → 250]
```

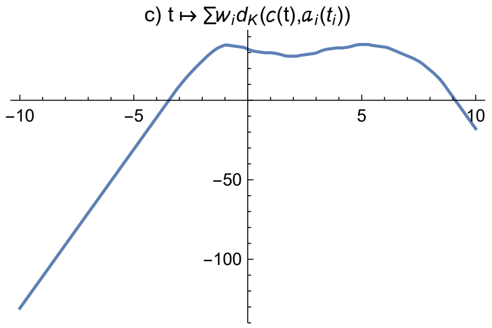




```
Clear[t]
NMinimize[F[t], t]
Plot[F[t], {t, -10, 10},
 PlotLabel -> "c)  t ↦ ∑w_i d_K (c(t),a_i(t_i))", ImageSize → 250]
NMinimize[{F[t], 0 ≤ t ≤ 2 Pi}, t]
FindMinimum[F[t], {t, 0, 2 Pi}]
Plot[F[t], {t, 1.7, 2.1},
 PlotLabel -> "c)  t ↦ ∑w_i d_K (c(t),a_i(t_i))", ImageSize → 250, PlotRange → All]
```

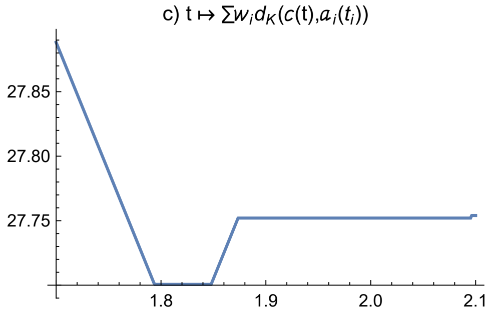
NMinimize::cvdiv : Failed to converge to a solution. The function may be unbounded. ≫

```
{-3.5442 × 10^107, {t → -1.7721 × 10^106}}
```



```
{27.7006, {t → 1.8046}}
```

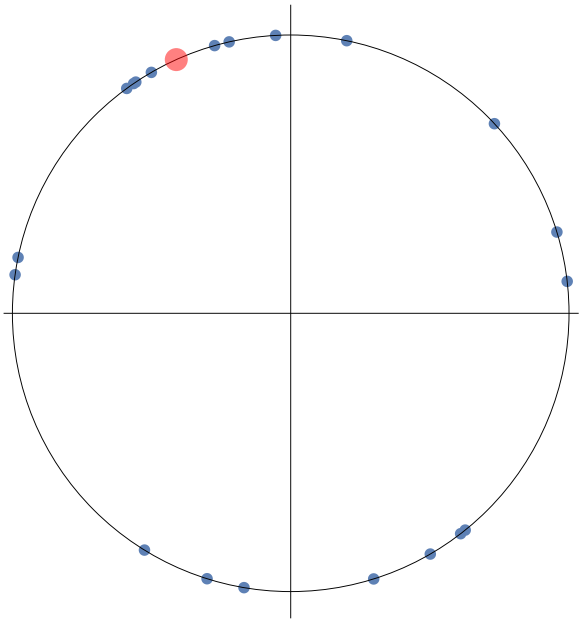
```
{27.752, {t → 1.99453}}
```



```
FindMinimum[F[t], {t, 0, 2 Pi}];
FindMinimum[F[t], {t, 0, 2 Pi}][[2]]
min = t /. FindMinimum[F[t], {t, 0, 2 Pi}][[2]]
slmin = ListPlot[{{Cos[min], Sin[min]}},
   PlotStyle → {Red, PointSize[.04], Opacity[.5]}];
(*ili slmin=Graphics[{Red,PointSize[.04],Opacity[.5],
    Point[{Cos[min],Sin[min]}]}];*)
sl0 = Graphics[Circle[{0, 0}, 1]];
sl1 = ListPlot[tocke[[All, 2]], PlotStyle → {PointSize[.02]},
   AspectRatio → Automatic, Axes → True, Ticks → None, ImageSize → 300];
Show[{sl1, sl0, slmin}]
```

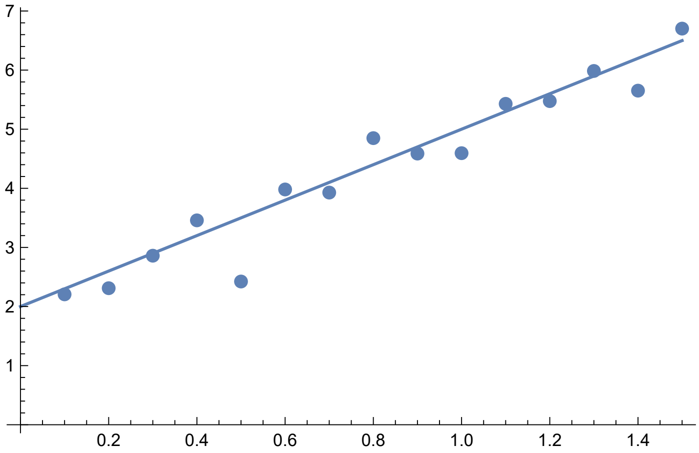
```
{t → 1.99453}
```

```
1.99453
```



Primjer 11. (Najbolji LS pravac)

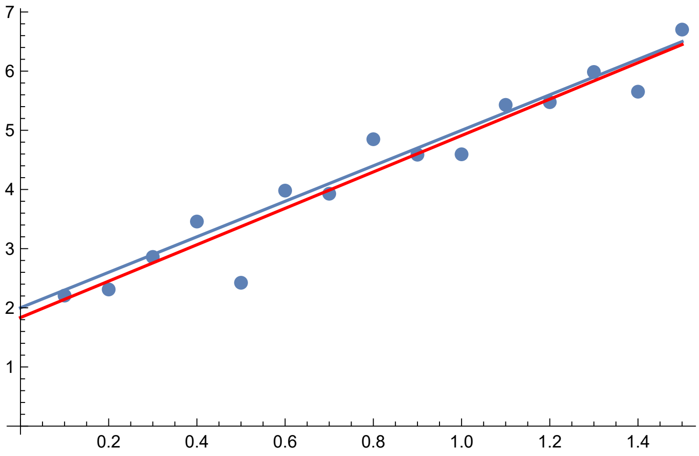
```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[NormalDistribution[0, 0.5]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[k, l]
min = NMinimize[Total[(y - k * x - l) ^2], {k, l}]
{k1, l1} = {k /. min[[2]], l /. min[[2]]}
k1 t + l1
slp1 = Plot[k1 t + l1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



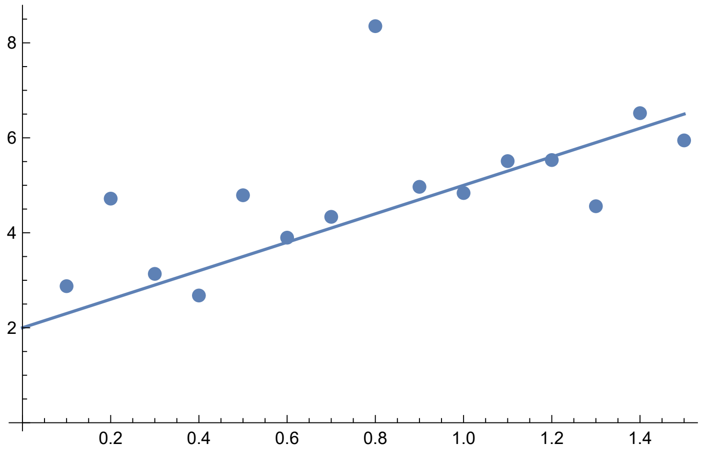
```
{1.9636, {k → 3.07621, l → 1.83528}}
```

```
{3.07621, 1.83528}
```

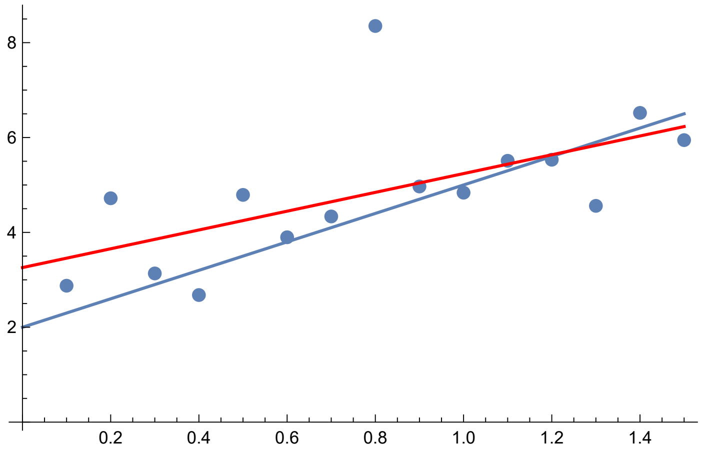
```
1.83528 + 3.07621 t
```



```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[StudentTDistribution[0, 0.5, 1]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[k, l]
min = NMinimize[Total[(y - k * x - l) ^2], {k, l}]
{k1, l1} = {k /. min[[2]], l /. min[[2]]}
k1 t + l1
slp1 = Plot[k1 t + l1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



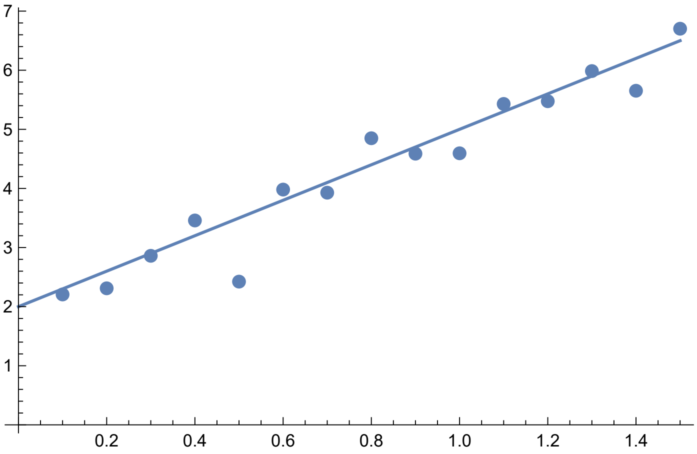
```
{18.9912, {k → 1.98176, l → 3.25839}}

{1.98176, 3.25839}

3.25839 + 1.98176 t
```



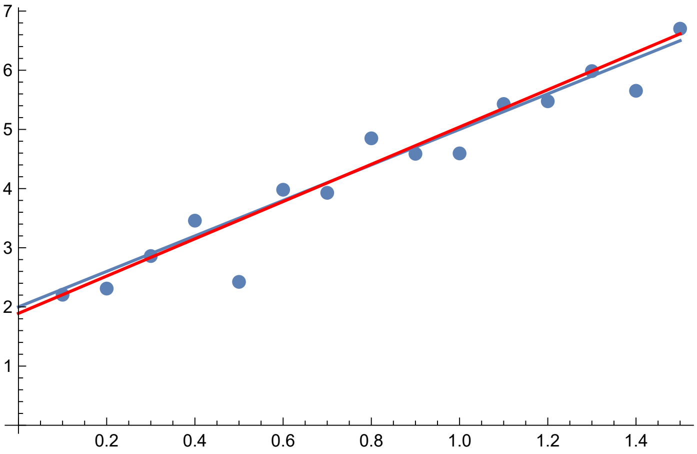
Primjer 12. (Najbolji LAD pravac)

Umjesto LS udaljenosti, promatramo l1 vertikalnu udaljenost tocki od pravca

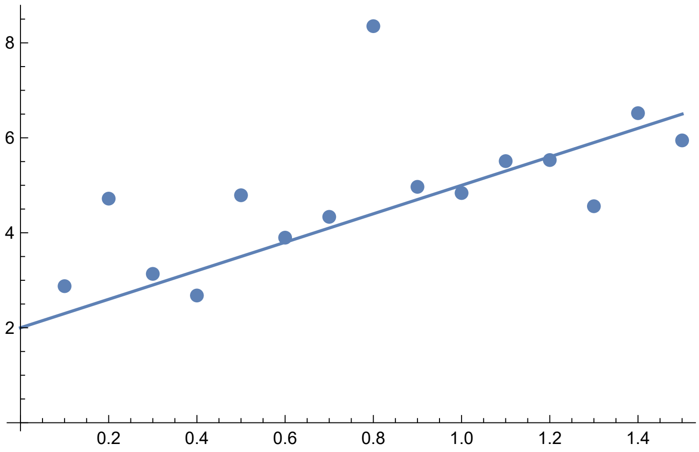
```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[NormalDistribution[0, 0.5]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[k, l]
min = NMinimize[Total[Abs[y - k * x - l]], {k, l}]
{k1, l1} = {k /. min[[2]], l /. min[[2]]}
k1 t + l1
slp1 = Plot[k1 t + l1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



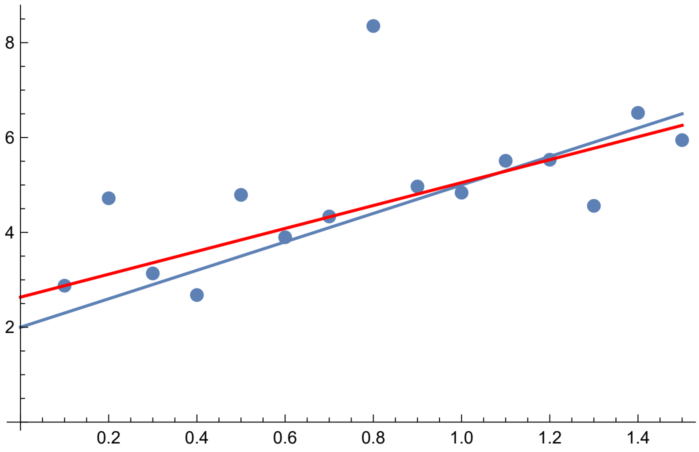
```
{3.97817, {k → 3.14963, l → 1.89121}}

{3.14963, 1.89121}

1.89121 + 3.14963 t
```



```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[StudentTDistribution[0, 0.5, 1]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[k, l]
min = NMinimize[Total[Abs[y - k * x - l]], {k, l}]
{k1, l1} = {k /. min[[2]], l /. min[[2]]}
k1 t + l1
slp1 = Plot[k1 t + l1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



```
{10.2988, {k → 2.41448, l → 2.63467}}

{2.41448, 2.63467}

2.63467 + 2.41448 t
```



Primjer 13. (Najbolji TLS pravac)

Kvadrat l2 udaljenost točke (x0, y0) do pravca y =- (a / b) x - c / b

```
Clear[x, x0, y0, a, b, c]
```

$$\text{Minimize}\left[(x-x0)^{\wedge}2+\left(-\frac{a}{b}x-\frac{c}{b}-y0\right)^{\wedge}2,\ x\right]$$

$$\left\{\begin{cases}\frac{c^2+2\,a\,c\,x0+a^2\,x0^2+2\,b\,c\,y0+2\,a\,b\,x0\,y0+b^2\,y0^2}{a^2+b^2} & b>0\ ||\ b<0\\ \infty & \text{True}\end{cases},\right.$$
$$\left.\left\{x\to\begin{cases}\text{Indeterminate} & !\,(b>0\ ||\ b<0)\\ \frac{-a\,c+b^2\,x0-a\,b\,y0}{a^2+b^2} & \text{True}\end{cases}\right\}\right\}$$

```
% // FullSimplify
```

$$\left\{\begin{cases}\frac{(c+a\,x0+b\,y0)^2}{a^2+b^2} & b\neq 0\\ \infty & \text{True}\end{cases},\ \left\{x\to\begin{cases}\text{Indeterminate} & b==0\\ \frac{b^2\,x0-a\,(c+b\,y0)}{a^2+b^2} & \text{True}\end{cases}\right\}\right\}$$

```
a = -1; b = 1; c = 2; x0 = 3; y0 = 4;
```

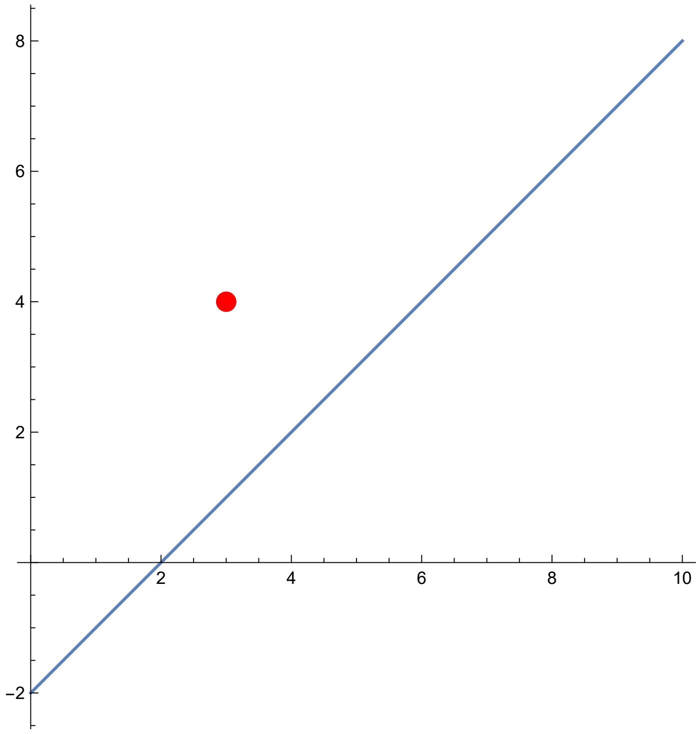
$$\text{Show}\left[\text{Plot}\left[-\frac{a}{b}x-\frac{c}{b},\ \{x,0,10\},\ \text{AxesOrigin}\to\{0,0\},\ \text{AspectRatio}\to\text{Automatic}\right],\right.$$
$$\left.\text{Graphics}[\{\text{Red},\ \text{PointSize}[.03],\ \text{Point}[\{x0,\ y0\}]\}]\right]$$

$$xx=\frac{b^2\,x0-a\,(c+b\,y0)}{a^2+b^2}$$

$$yy=-\frac{a}{b}xx-\frac{c}{b}$$

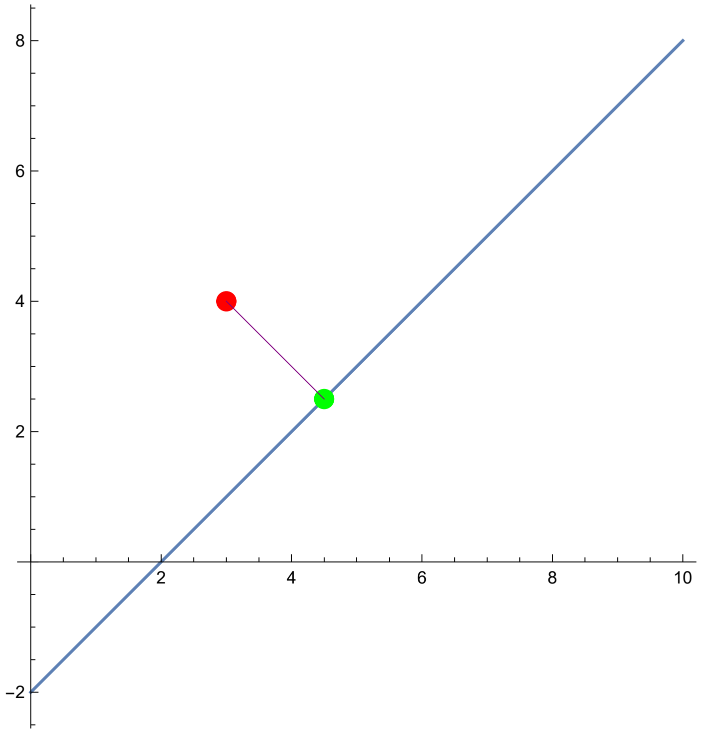
$$\text{Show}\left[\text{Plot}\left[-\frac{a}{b}x-\frac{c}{b},\ \{x,0,10\},\ \text{AxesOrigin}\to\{0,0\},\ \text{AspectRatio}\to\text{Automatic}\right],\right.$$
$$\text{Graphics}[\{\text{Red},\ \text{PointSize}[.03],\ \text{Point}[\{x0,\ y0\}]\}],$$
$$\text{Graphics}[\{\text{Green},\ \text{PointSize}[.03],\ \text{Point}[\{xx,\ yy\}]\}],$$
$$\left.\text{Graphics}[\{\text{Purple},\ \text{PointSize}[.03],\ \text{Line}[\{\{x0,\ y0\},\ \{xx,\ yy\}\}]\}]\right]$$

$$\frac{c^2+2\,a\,c\,x0+a^2\,x0^2+2\,b\,c\,y0+2\,a\,b\,x0\,y0+b^2\,y0^2}{a^2+b^2}$$



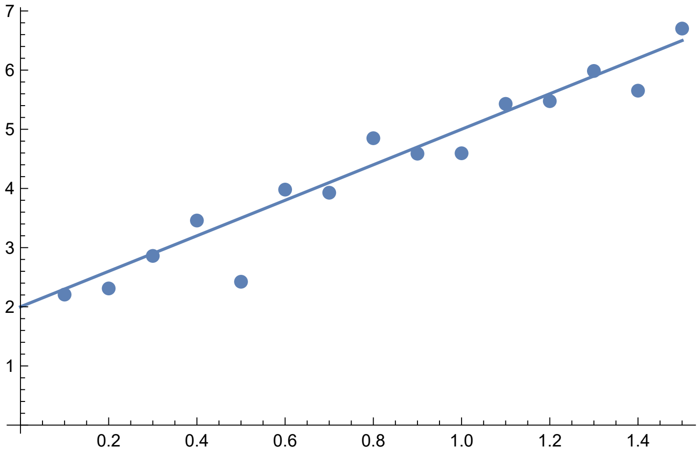
$\frac{9}{2}$

$\frac{5}{2}$

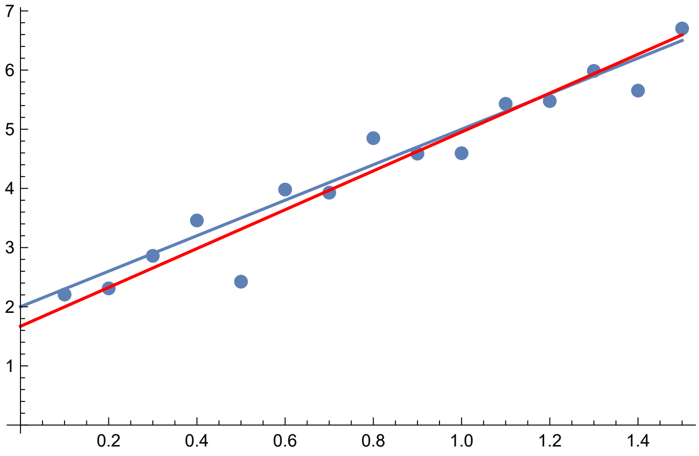


$\frac{9}{2}$

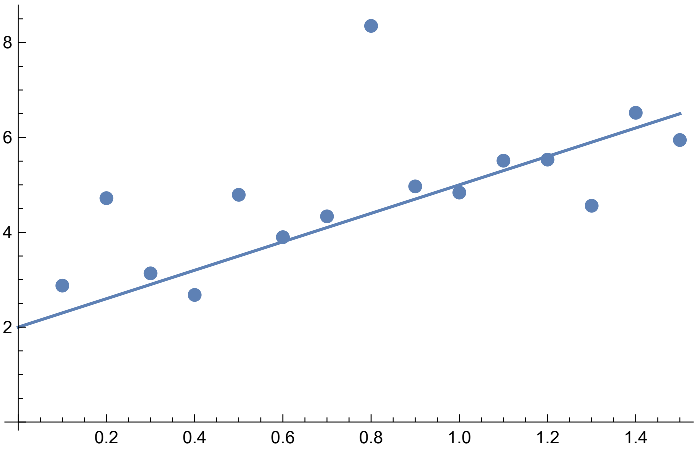
```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[NormalDistribution[0, 0.5]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[a, b, c]
min = NMinimize[Total[(a * x + b * y + c) ^2 / (a^2 + b^2)], {a, b, c}]
{a1, b1, c1} = {a /. min[[2]], b /. min[[2]], c /. min[[2]]}
(-a1 / b1) t - c1 / b1
slp1 = Plot[(-a1 / b1) t - c1 / b1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



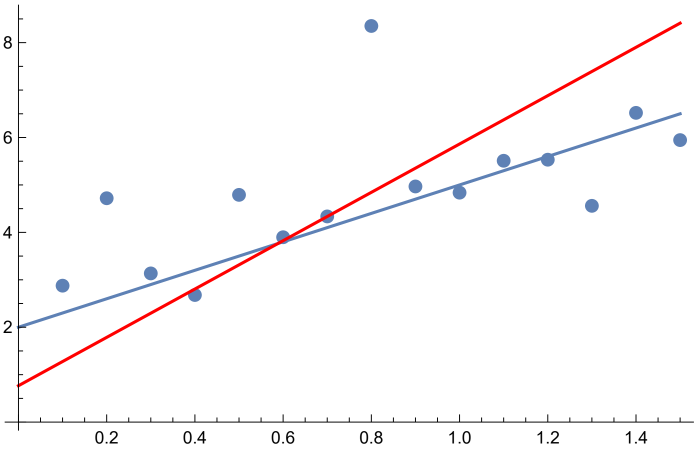
```
{0.176883, {a → 5.08633, b → -1.54899, c → 2.58578}}

{5.08633, -1.54899, 2.58578}

1.66933 + 3.28364 t
```



```
m = 15;
SeedRandom[7] ;
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[StudentTDistribution[0, 0.5, 1]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[a, b, c]
min = NMinimize[Total[(a * x + b * y + c) ^2 / (a^2 + b^2)], {a, b, c}]
{a1, b1, c1} = {a /. min[[2]], b /. min[[2]], c /. min[[2]]}
slp1 = Plot[(-a1 / b1) t - c1 / b1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



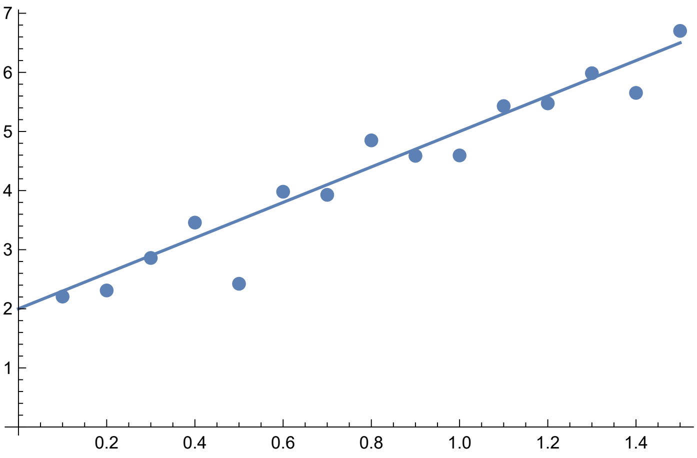
```
{1.7111, {a → 3.45188, b → -0.677384, c → 0.519613}}

{3.45188, -0.677384, 0.519613}
```



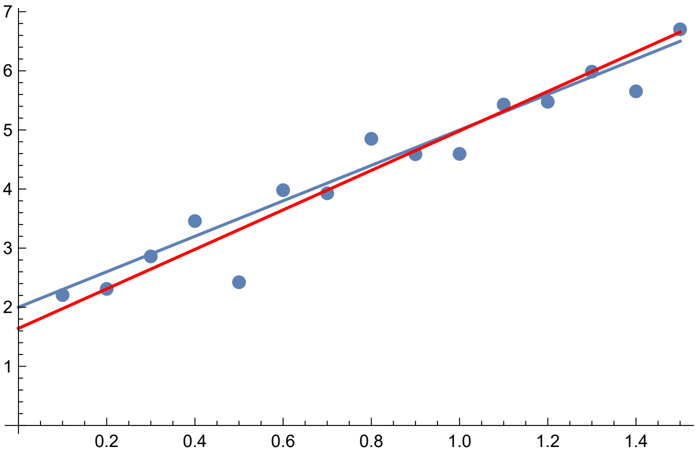
Primjer 14. (Najbolji OD pravac)

Sada promatramo l2 udaljenost tocki od pravca

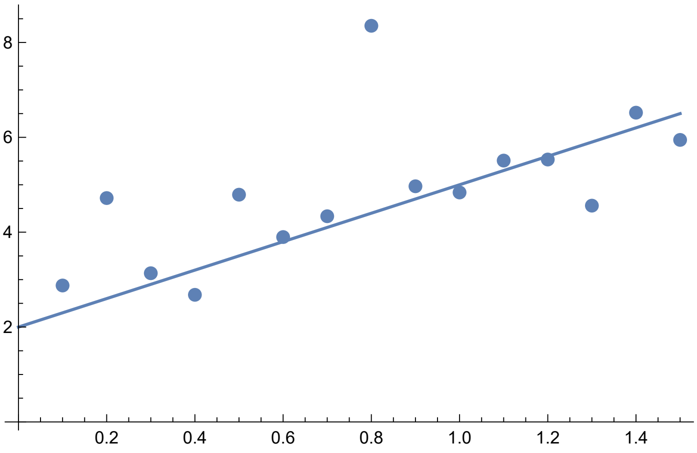
```
m = 15;
SeedRandom[7]
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[NormalDistribution[0, 0.5]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[a, b, c]
min = NMinimize[Total[Abs[a * x + b * y + c] / Sqrt[a^2 + b^2]], {a, b, c}]
{a1, b1, c1} = {a /. min[[2]], b /. min[[2]], c /. min[[2]]}
(-a1 / b1) t - c1 / b1
slp1 = Plot[(-a1 / b1) t - c1 / b1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



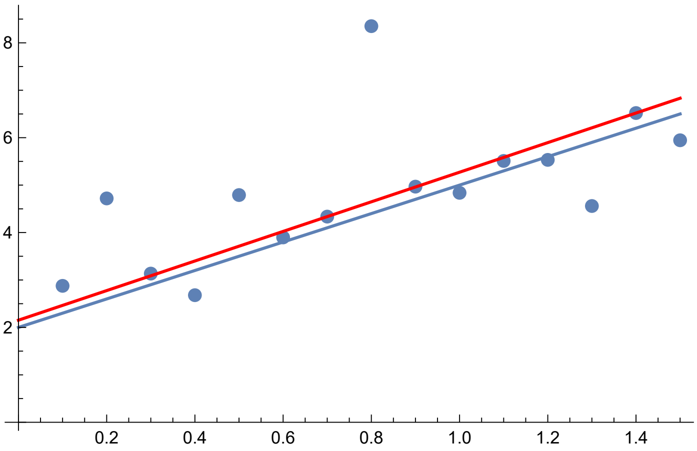
```
{1.20107, {a → 2.76268, b → -0.826832, c → 1.3577}}

{2.76268, -0.826832, 1.3577}

1.64205 + 3.34129 t
```



```
m = 15;
SeedRandom[7]
f[x_] := 3 x + 2
x = Table[i / 10., {i, m}];
y = f[x] + Table[RandomVariate[StudentTDistribution[0, 0.5, 1]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[a, b, c]
min = NMinimize[Total[Abs[a * x + b * y + c] / Sqrt[a^2 + b^2]], {a, b, c}]
{a1, b1, c1} = {a /. min[[2]], b /. min[[2]], c /. min[[2]]}
(-a1 / b1) t - c1 / b1
slp1 = Plot[(-a1 / b1) t - c1 / b1, {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



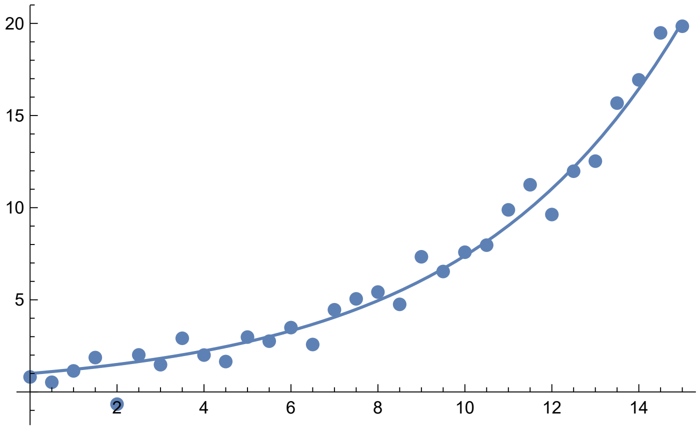
```
{3.49395, {a → 7.80544, b → -2.5022, c → 5.38589}}

{7.80544, -2.5022, 5.38589}

2.15246 + 3.11943 t
```



Primjer 15. (Procjena parametara eksponencijalne

funkcije)

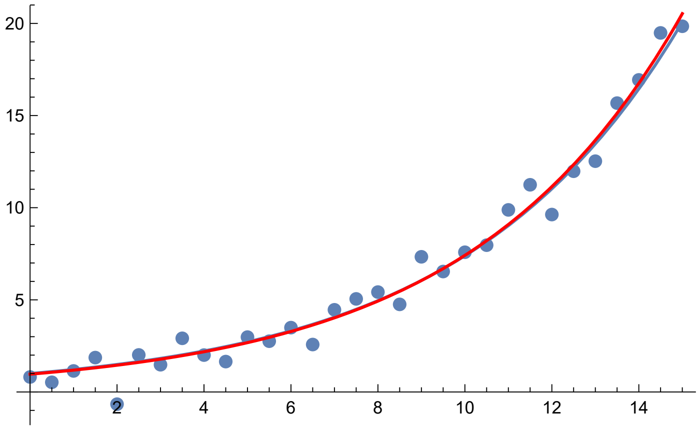
```
SeedRandom[7]
f[x_] := 1 * Exp[0.2 * x]
x = Range[0, 15, 0.5];
m = Length[x];
y = f[x] + Table[RandomVariate[NormalDistribution[0, 1]], {i, m}];
pod = Table[{x[[i]], y[[i]]}, {i, m}];
slpod = ListPlot[pod, PlotStyle → {PointSize[.02]}];
slp = Plot[f[t], {t, 0, x[[m]]}];
Show[slpod, slp]
Clear[b, c]
min = NMinimize[Total[(b * Exp[c * x] - y)^2], {b, c}]
{b1, c1} = {b /. min[[2]], c /. min[[2]]}
b1 * Exp[c1 * t]
slp1 = Plot[b1 * Exp[c1 * t], {t, 0, x[[m]]}, PlotStyle → {Red}];
sl2 = Show[slpod, slp, slp1]
```



```
{18.644, {b → 0.966184, c → 0.20375}}
```

```
{0.966184, 0.20375}
```

$0.966184\, e^{0.20375\, t}$


20
15
10
5
2
4
6
8
10
12
14