<u>Pienimm"an neli"osumman sovitus</u>

V2 19.3. 2002, V3 lokakuu -02

2013kevat/maple/

Aluksi luentoesimerkki

```
> restart
> with(LinearAlgebra):alias(Tr=Transpose):
> with(plots):
> xd:=[-1.3,-0.1,0.2,1.3]; yd:=[0.103,1.099,0.808,1.897];
```

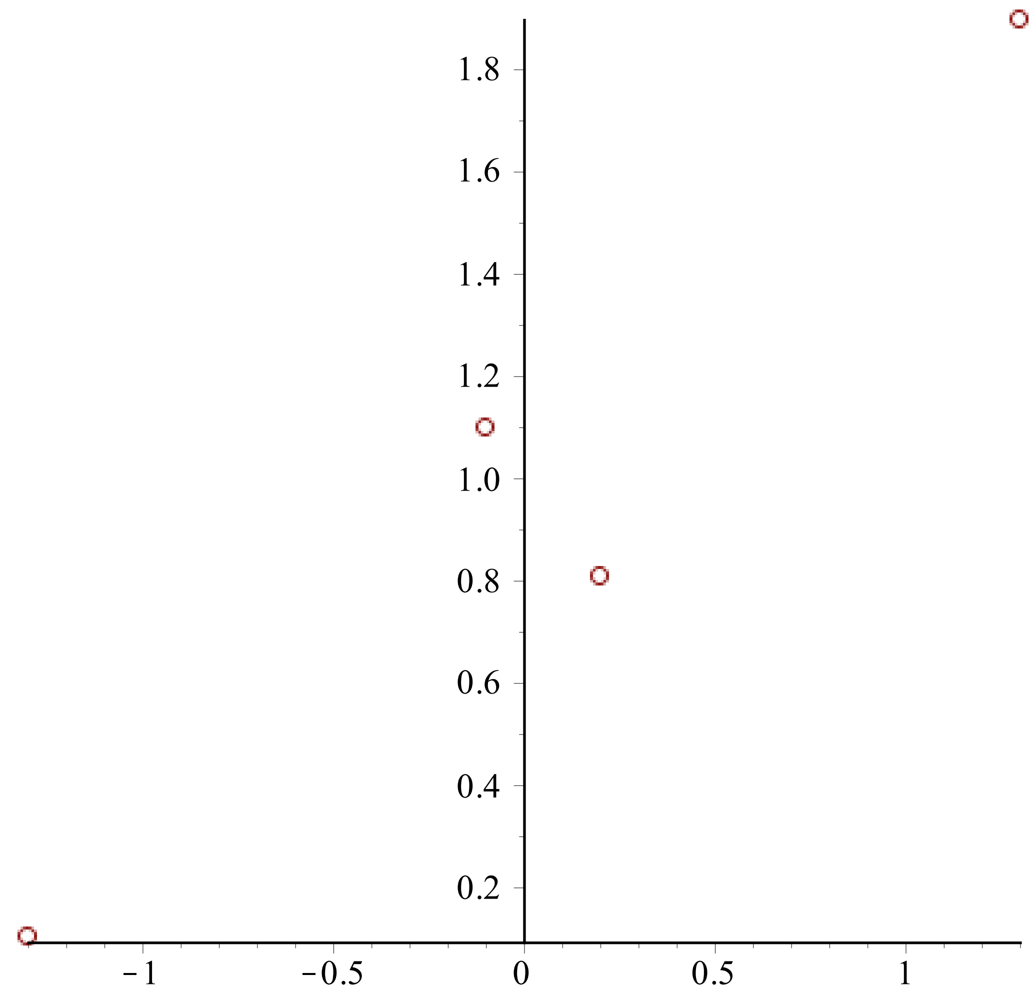
$$xd := [-1.3, -0.1, 0.2, 1.3]$$

$$yd := [0.103, 1.099, 0.808, 1.897] \quad \textbf{(1.1)}$$

```
> n:=nops(xd);
```

$$n := 4 \quad \textbf{(1.2)}$$

```
> plot([seq([xd[i],yd[i]],i=1..n)],style=point,symbol=circle,
  symbolsize=15);
```




```
> datakuva:=%:
> C:=<<1,1,1,1>|<op(xd)>>;
```

$$C := \begin{bmatrix} 1 & -1.3 \\ 1 & -0.1 \\ 1 & 0.2 \\ 1 & 1.3 \end{bmatrix} \tag{1.3}$$

```
> M:=Tr(C).C;
```

$$M := \begin{bmatrix} 4. & 0.0999999999999999 \\ 0.0999999999999999 & 3.43000000000000 \end{bmatrix} \tag{1.4}$$

```
> B:=Tr(C).Vector(yd);
```

$$B := \begin{bmatrix} 3.90700000000000 \\ 2.38390000000000 \end{bmatrix} \tag{1.5}$$

```
> ab:=LinearSolve(M,B);
```

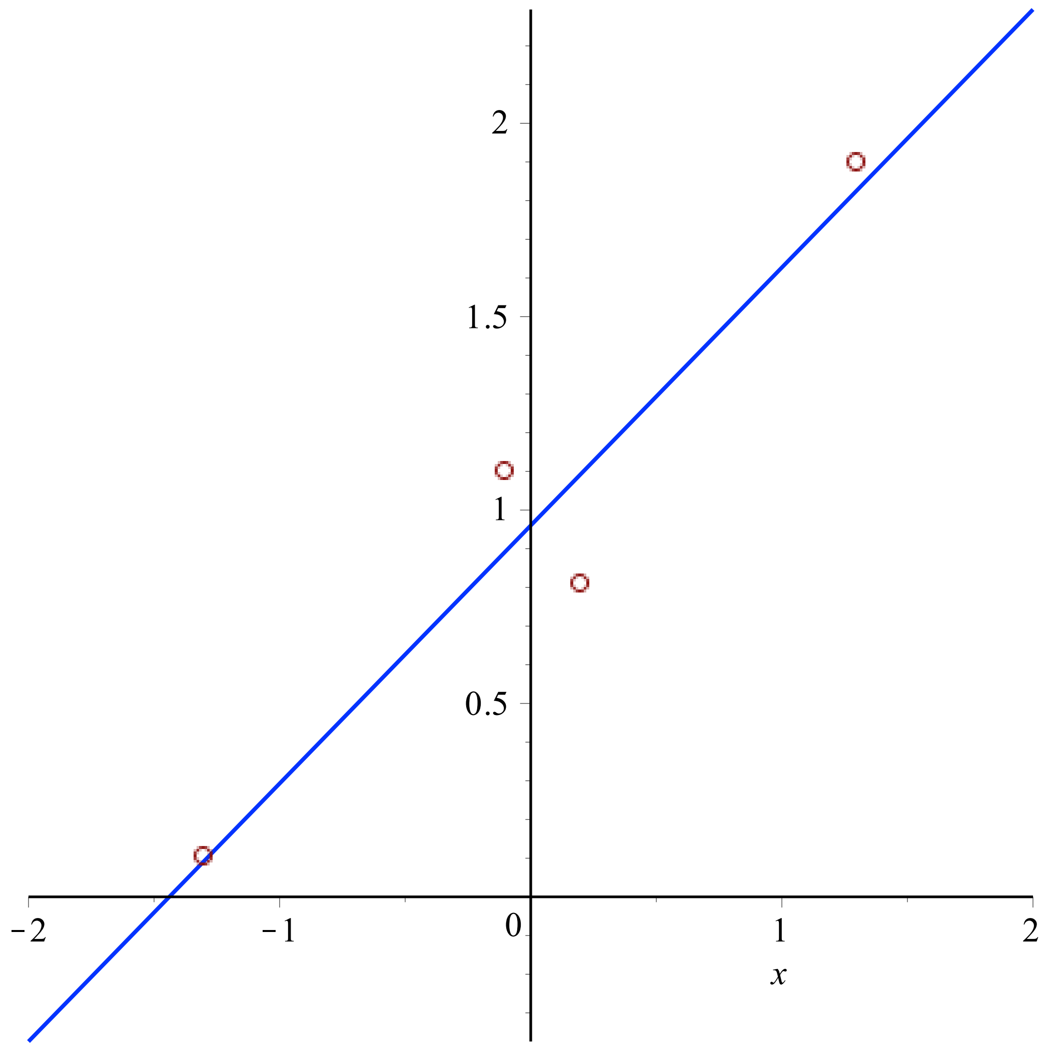
(1.6)

$$ab := \begin{bmatrix} 0.960074398249453 \\ 0.667024070021882 \end{bmatrix} \tag{1.6}$$

```
> suora:=ab[1]+ab[2]*x;
```

$$suora := 0.960074398249453 + 0.667024070021882\,x \tag{1.7}$$

```
> display(plot(suora,x=-2..2,color=blue),datakuva);
```



> *%restart; with(plots) :*

$$\%restart \tag{1}$$

Muistellaan aluksi interpolaatioteht.

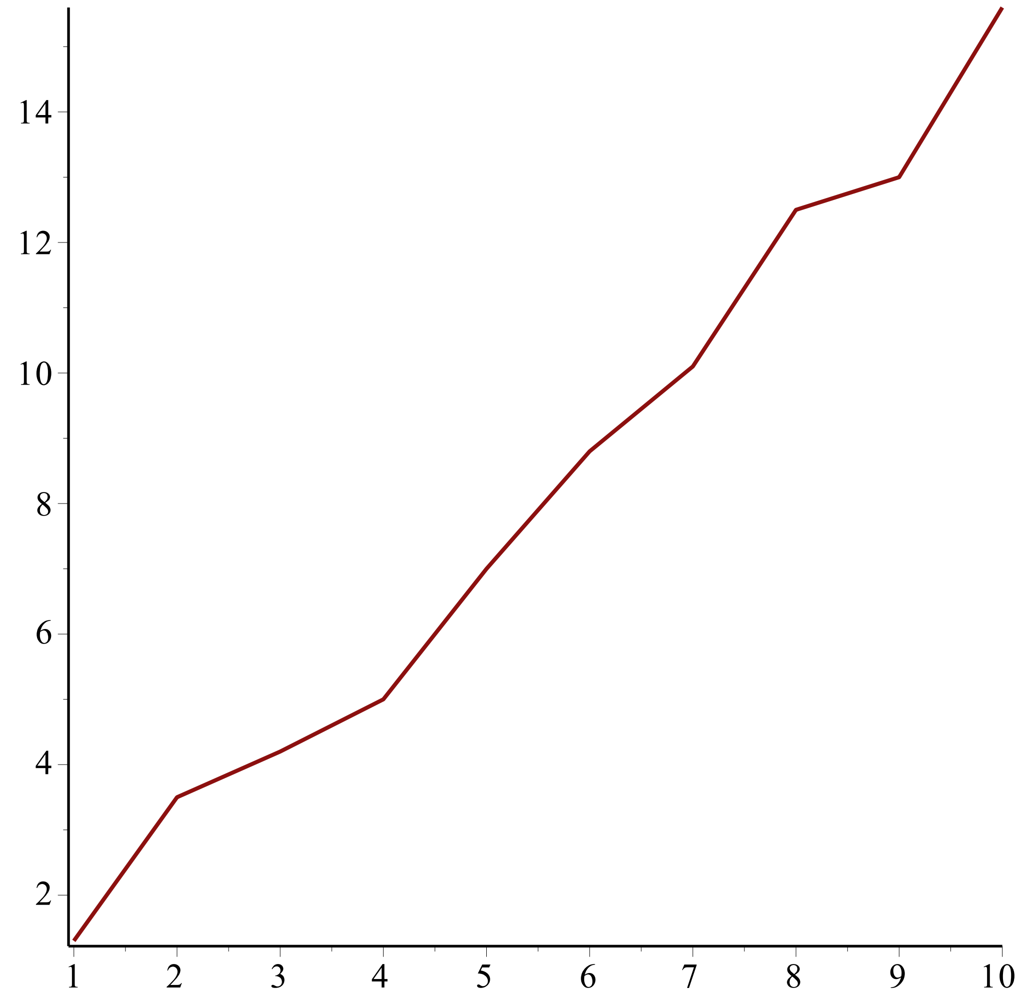
Esim:

```
> xd:=[$1..10];yd:=[1.3,3.5,4.2,5.0,7.0,8.8,10.1,12.5,13.0,15.6];
```

$$xd := [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$$

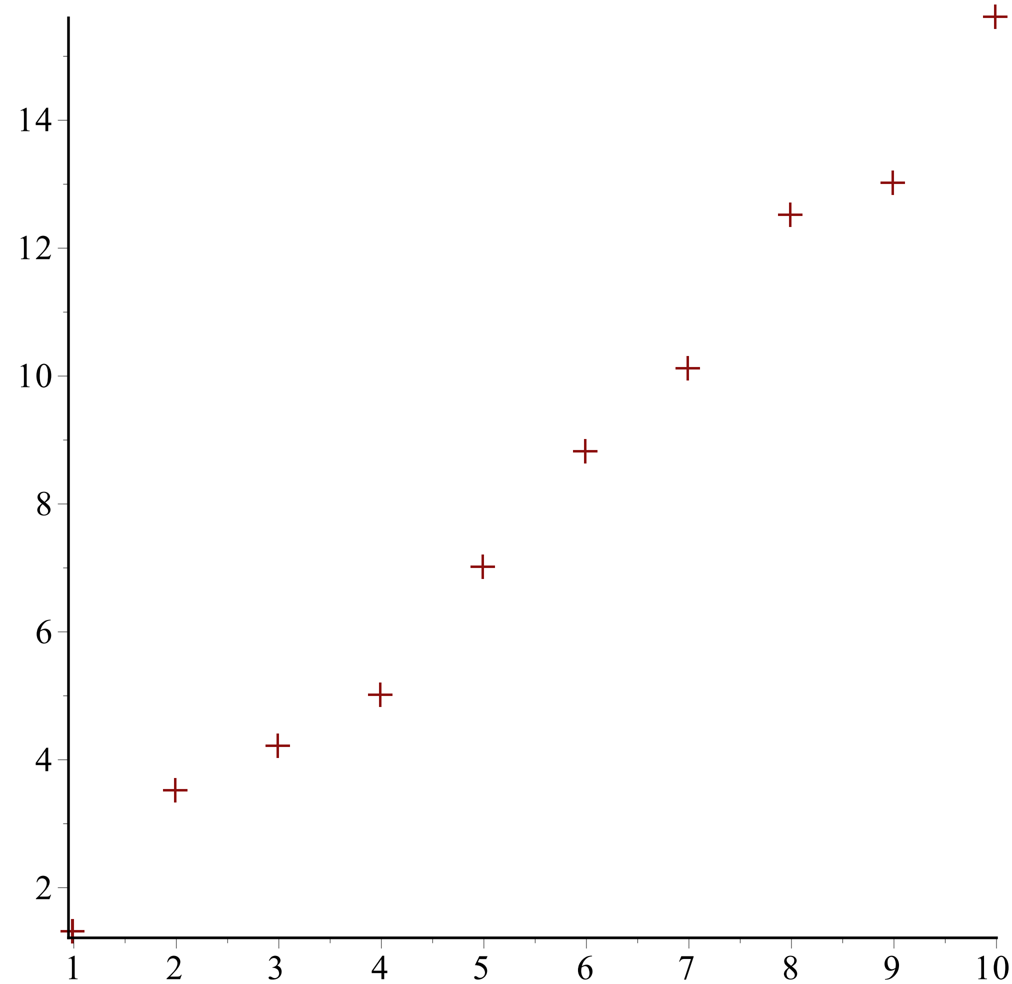
$$yd := [1.3, 3.5, 4.2, 5.0, 7.0, 8.8, 10.1, 12.5, 13.0, 15.6] \tag{2}$$

> $plot(xd, yd);$



> $datakuva := plot(xd, yd, style = point, symbol = cross, symbolsize = 18) :$

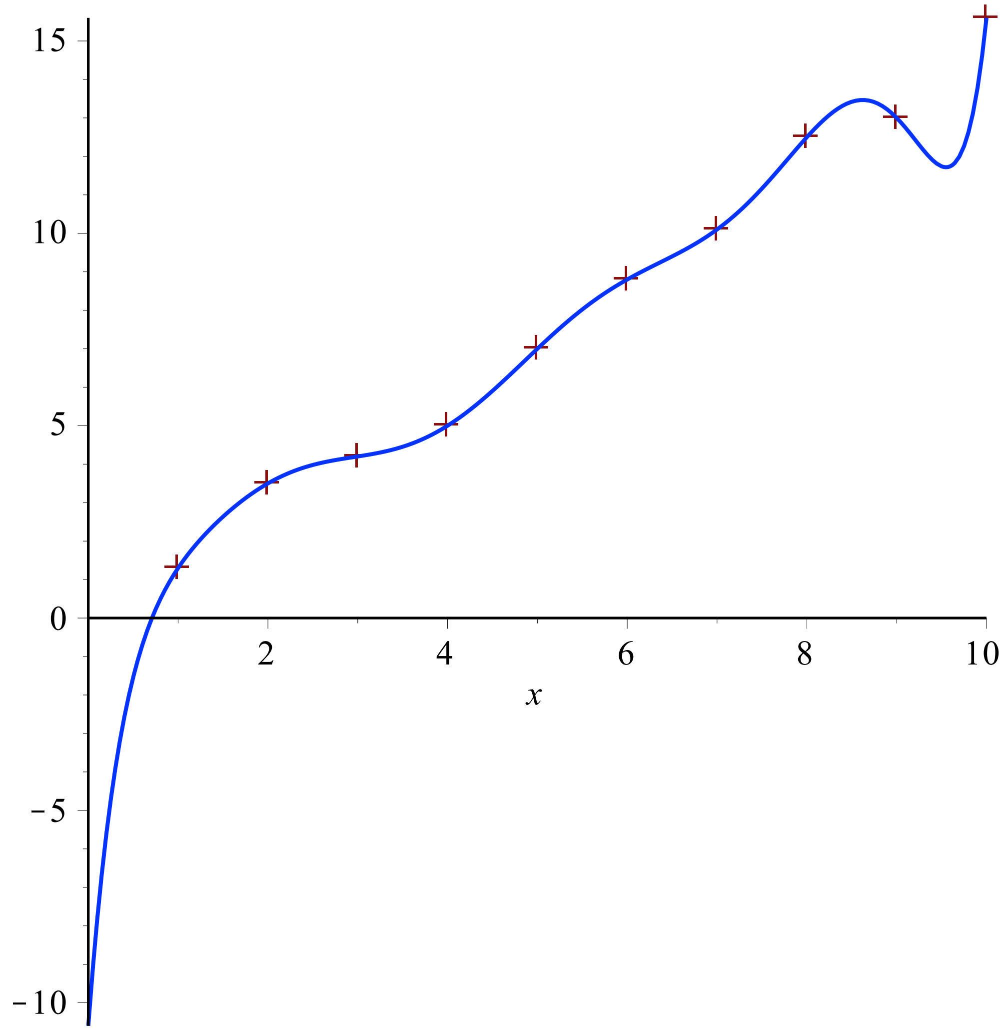
```
> plot([seq([xd[i],yd[i]],i=1..10)],style=point,symbol=cross,
  symbolsize=18); # Vaihtoehtoinen tapa: pisteiden lista.
```




```
> p:=interp(xd,yd,x); # Ei tarvitse ladata.
```

$$p := 0.00007164903227\,x^9 - 0.003127480266\,x^8 + 0.05718584874\,x^7 - 0.5698264133\,x^6 + 3.378032576\,x^5 - 12.26852504\,x^4 + 27.40284704\,x^3 - 37.40852517\,x^2 + 31.31186805\,x - 10.60000106 \quad \textbf{(3)}$$

```
> display(datakuva,plot(p,x=0..10,color=blue));
```



Interpolaatio pakottaa polynomin usein voimakkaaseen heilahteluun, jolloin "mets"a katoaa n"akyvist"a puilta".

Luovutaan vaatimuksesta, ett"a polynomin tulisi kulkea datapisteiden kautta. Valitaan alempiasteinen polynomi, joka sopivassa mieless"a approksimoi parhaiten annettua dataa.
Kriteeriksi valitaan usein funktiomallin (t"ass polynomi) ja datapisteiss"a laskettujen erotuksten neli"osumman minimointi.
T"at"a kautta menetelm"a liittyy funktioiden "a"ariarvoteemaan.

Polynomi-PNS-teht"av"ass"a on annettu datapisteit"a (x_i, y_i) (paljon) ja teht"av"an"a on sovittaa esim. polynomi

$$p(x) = b_0 + b_1 x + b_2 x^2 + b_3 x^3$$

annettuun dataan niin, ett"a $\sum_{i=1}^{n} \left(y_i - p\left(x_i\right)\right)^2$ minimoituu.
Yleens"a datapisteiden lkm =n on paljon suurempi kuin polynomin asteluku m (t"ass"a 3).
Jos A on matriisi , jonka alkiot ovat $a_{i,j} = x_i^j$, i=1..n, j=0..3, saadaan b-kertoimien m"a"ar"a"amiseksi ylim"a"ar"aytyv"a yht"al"osysteemi A b = y, miss"a siis A on n x 3 - matriisi.
PNS-ratkaisuun p"a"ast"a"an kertomalla t"am"a systeemi A:n transpoosilla, jolloin ratkaistavaksi tulee

systeemi
(1) (A^T)A b = A^T y (miss"a A^T tarkoittaa A:n transpoosia)
(T"am"a johdetaan joko osittaisderivoimalla tai ortogonaaliprojektioon perustuvalla geometrisluonteisella argumentilla.)

Jos tuntemattomia kertoimia on yht"a monta kuin datapisteit"a (polynomin asteluku = n-1), niin systeemill"a
A b = y on yksik"asitteinen ratkaisu, mik"ali x-pisteet ovat erillisi"a. T"all"oin on kyse interpolaatiopolynomista.

Yll"a oleva ratkaisu voidaan Maplella tehd"a rakentamalla matriisi A ja ratkaisemalla systeemi (1) .
Maplen **LinearAlgebra**-kirjastossa on my"os **LeastSquares**- funktio, jolle annetaan pelkk"a A -matriisi ja datapisteet, joka
siis ratkaisee ylim"a"ar"aytyv"an systeemin A b = y PNS-mieless"a.

Jotta valinta ei olisi liian helppoa, Maplessa on my"os kirjasto **CurveFitting** ja siell"a funktio LeastSquares , joka suorittaa
sovituksen suoraan datan perusteella, ja k"aytt"aj"all"a on valittavanaan eri menetelmi"a.

```
> restart:with(LinearAlgebra):
> #with(linalg):
> with(plots):#with(stats):
> alias(Tr=Transpose,Van=VandermondeMatrix):
```

Esim.

Sovita 2. asteen polynomi annettuun dataan

```
> xd:=[8,10,12,16,20,30,40,60,100];
  yd:=[0.88,1.22,1.64,2.72,3.96,7.66,11.96,21.56,43.16];
```

$$xd := [8, 10, 12, 16, 20, 30, 40, 60, 100]$$

$$yd := [0.88, 1.22, 1.64, 2.72, 3.96, 7.66, 11.96, 21.56, 43.16] \quad \textbf{(3.1)}$$

```
> n:=nops(xd);
```

$$n := 9 \quad \textbf{(3.2)}$$

```
> C:=Van(xd,n,3); # tai # Matrix((i,j)->xd[i]^(j-1),n,3);
```

$$C := \begin{bmatrix} 1 & 8 & 64 \\ 1 & 10 & 100 \\ 1 & 12 & 144 \\ 1 & 16 & 256 \\ 1 & 20 & 400 \\ 1 & 30 & 900 \\ 1 & 40 & 1600 \\ 1 & 60 & 3600 \\ 1 & 100 & 10000 \end{bmatrix} \quad \textbf{(3.3)}$$

```
> M:=Tr(C).C;
```

$$M := \begin{bmatrix} 9 & 296 & 17064 \\ 296 & 17064 & 1322336 \\ 17064 & 1322336 & 116590368 \end{bmatrix} \tag{3.4}$$

```
> b:=Tr(C).Vector(yd);
```

$$b := \begin{bmatrix} 94.7600000000000 \\ 6479.44000000000 \\ 5.37940800000000\ 10^{5} \end{bmatrix} \tag{3.5}$$

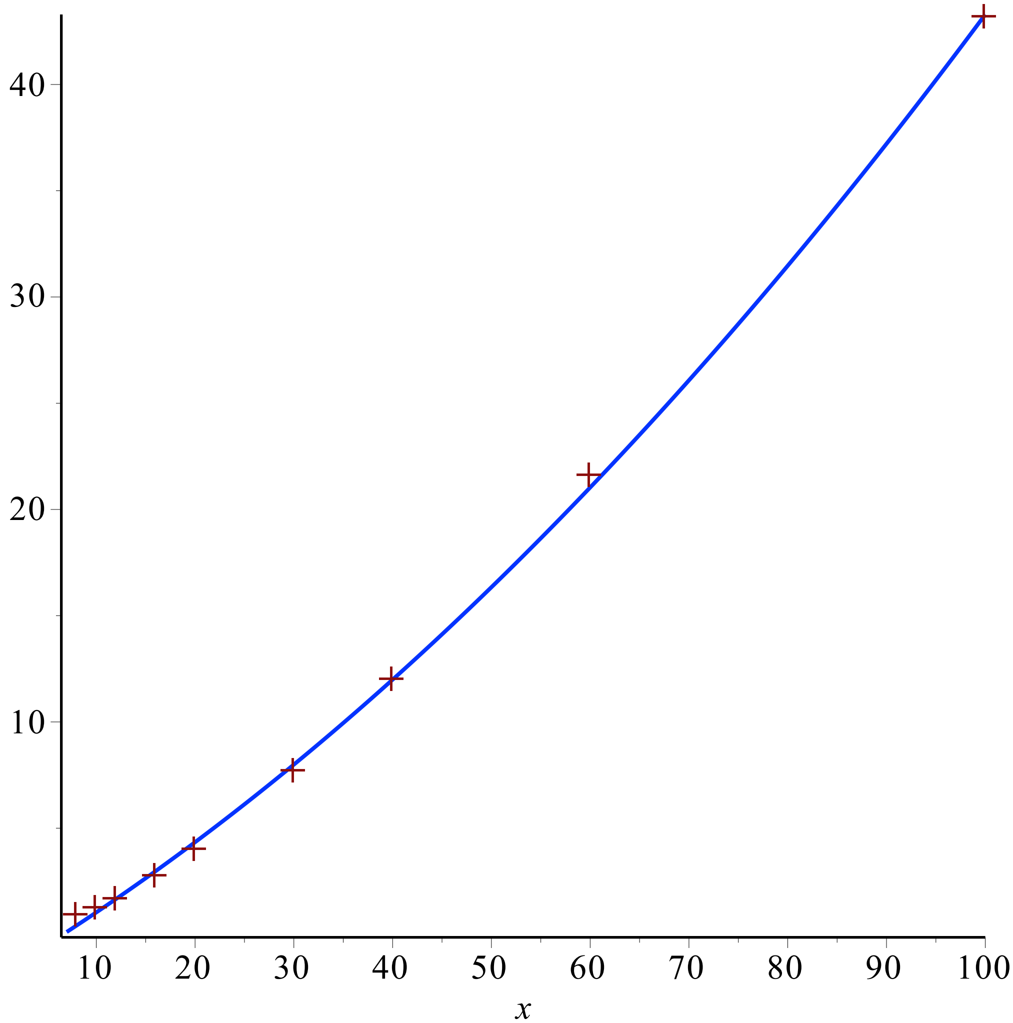
```
> a:=LinearSolve(M,b);
```

$$a := \begin{bmatrix} -1.91914925269908 \\ 0.278213536291725 \\ 0.00173940087505516 \end{bmatrix} \tag{3.6}$$

```
> p:=add(a[i]*x^(i-1),i=1..3);
```

$$p := -1.91914925269908 + 0.278213536291725\, x + 0.00173940087505516\, x^{2} \tag{3.7}$$

```
> datakuva:=plot([seq([xd[i],yd[i]],i=1..n)],style=point,symbol=
cross,symbolsize=18):
> display(plot(p,x=7..100,color=blue),datakuva);
```



Tarkistetaan **LinearAlgebra**- ja **CurveFitting**-pakkauksten LeastSquares-funktioilla.
Huomaa, ett"a samannimisten funktioiden tapauksessa on syyt"a k"aytt"a"a pitk"a"a muotoa, muuten toiminta riipuu kirjastojen
latausj"arjestyksest"a (onneksi menee virheeseen, eik"a laske jotain p"a"at"ont"a).

> *LinearAlgebra*[*LeastSquares*](*C*, *Vector*(*yd*));

$$\begin{bmatrix} -1.91914925269908 \\ 0.278213536291725 \\ 0.00173940087505516 \end{bmatrix} \qquad \textbf{(3.8)}$$

> *CurveFitting*[*LeastSquares*]$\left(xd, yd, x, curve = c0 \cdot x^2 + c1 \cdot x + c2\right)$

$$-1.91914925269908 + 0.278213536291725\, x + 0.00173940087505516\, x^2 \qquad \textbf{(3.9)}$$

>

H"airi"oalttius

> *alias*(*cond* = *ConditionNumber*)

$$Tr, Van, cond \qquad \textbf{(4.1)}$$

```
> n:=5: m:=n-2:x:=[$1..n];
```

$$x := [1, 2, 3, 4, 5] \quad (4.2)$$

```
> C:=Van(x,n,m);cond(Tr(C).C): evalf(%);
```

$$C := \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \\ 1 & 5 & 25 \end{bmatrix}$$

$$10575.60000 \quad (4.3)$$

```
> n:=6: m:=n-2:x:=[$1..n]:C:=Van(x,n,m):cond(Tr(C).C): evalf(%);
```

$$3.034650032\ 10^{6} \quad (4.4)$$

```
> n:=10: m:=n-2:x:=[$1..n]:C:=Van(x,n,m):cond(Tr(C).C): evalf(%);
```

$$1.786053076\ 10^{18} \quad (4.5)$$

H"airi"oluku cond mittaa sensitiivisyytt"a virheille (datan virheille ja py"oristysvirheille). Normaaliyht"al"oratkaisun varjopuoli on suuri h"ari"oalttius, jos teht"av"an koko on v"ah"ank"a"an suuri.

```
> n:=6: m:=2:x:=[$1..n]:C:=Van(x,n,m):cond(Tr(C).C): evalf(%);
```

$$119.4666667 \quad (4.6)$$

```
> n:=20: m:=2:x:=[$1..n]:C:=Van(x,n,m):cond(Tr(C).C): evalf(%);
```

$$713.2631579 \quad (4.7)$$

```
> n:=20: m:=3:x:=[$1..n]:C:=Van(x,n,m):cond(Tr(C).C): evalf(%);
```

$$5.124155474\ 10^{5} \quad (4.8)$$

Paremmat menetelm"at perustuvat ortogonalisointiin tai ns. singulaariarvohajoitelmaan.

Esim.

Sovita a) 2. asteen PNS-polynomi ja b) interpolaatiopolynomi

```
> P:=[[1.4,7400],[1.8,7500],[2.4,7600],[3,7500],[4,7200]];
```

$$P := [[1.4, 7400], [1.8, 7500], [2.4, 7600], [3, 7500], [4, 7200]] \quad (5.1)$$

```
> xd:=map(lis->lis[1],P);
  yd:=map(lis->lis[2],P);
```

$$xd := [1.4, 1.8, 2.4, 3, 4]$$

$$yd := [7400, 7500, 7600, 7500, 7200] \quad (5.2)$$

Jos tuo n"aytt"a"a oudolta, voitaisiin tehd"a funktiot `poimieka:=x->x[1]; poimitoka:= x->x[2];` ja suorittaa `xd:=map(poimieka,P);yd:=map(poimitoka,P);`

T"ass"a on k"atev"a ja yleisp"atev"a tapa muodostaa matriisi.

```
> C:=Matrix(5,3,(i,j)->xd[i]^(j-1));
```

$$C := \begin{bmatrix} 1.0 & 1.4 & 1.96 \\ 1.0 & 1.8 & 3.24 \\ 1.0 & 2.4 & 5.76 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix} \tag{5.3}$$

Tai, kuten edell"a, suoraan LinearAlgebra-kirjaston funktiolla (joka yll"a "aliasoitiinkin" nimelle Van).

```
> VandermondeMatrix(xd):%[1..5,1..3];  #
```

$$\begin{bmatrix} 1.0 & 1.4 & 1.96 \\ 1.0 & 1.8 & 3.24 \\ 1.0 & 2.4 & 5.76 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix} \tag{5.4}$$

K"aytet"a"an nyt sit"a keskimm"aist"a "harmaata laatikkoa"

```
> kertoimet:=LinearAlgebra[LeastSquares](C,Vector(yd));
```

$$kertoimet := \begin{bmatrix} 6639.59080643039 \\ 762.903547229810 \\ -156.021655373958 \end{bmatrix} \tag{5.5}$$

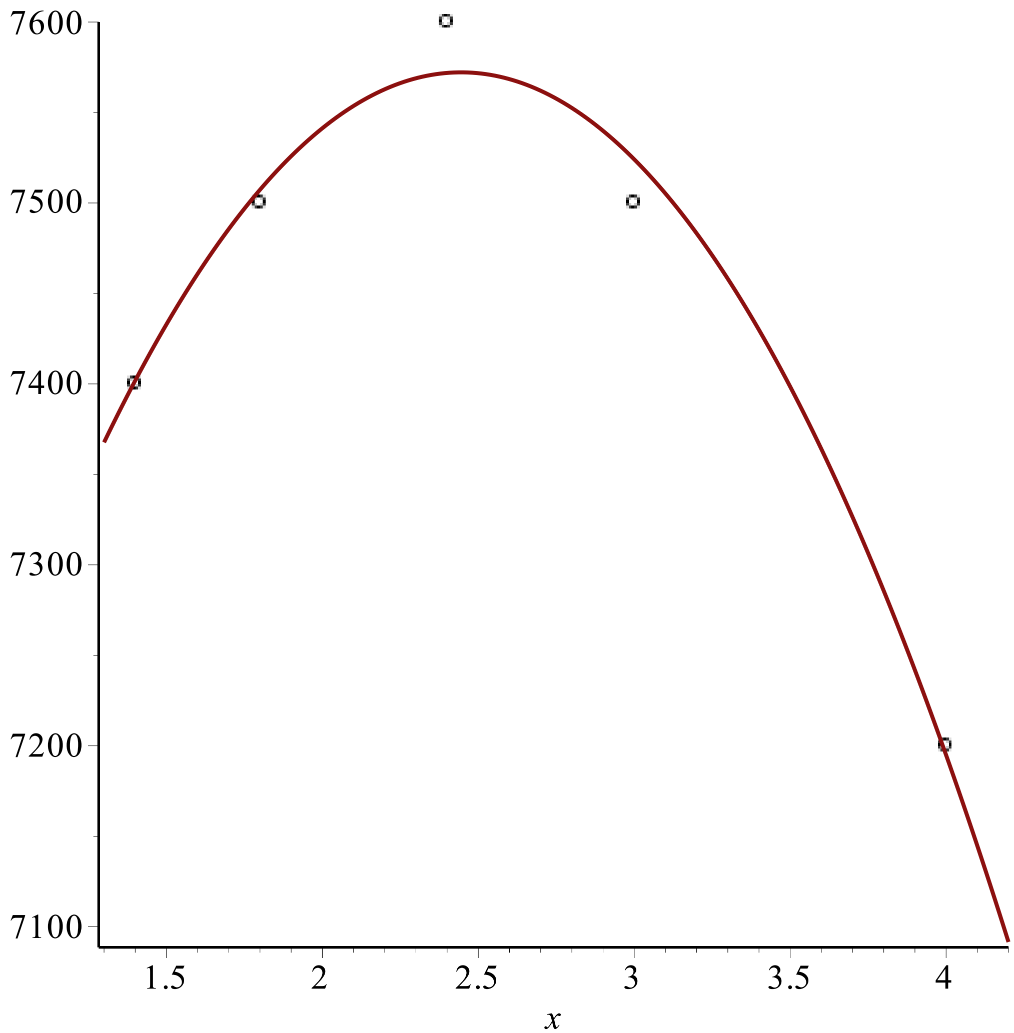
```
> x := 'x'
```

$$x := x \tag{5.6}$$

```
> poly2:=Tr(kertoimet).<1,x,x^2>;
```

$$poly2 := 6639.59080643039 + 762.903547229810\, x - 156.021655373958\, x^2 \tag{5.7}$$

```
> display(plot(P,style=point,symbol=circle,color=black),plot(poly2,
  x=1.3..4.2));
```

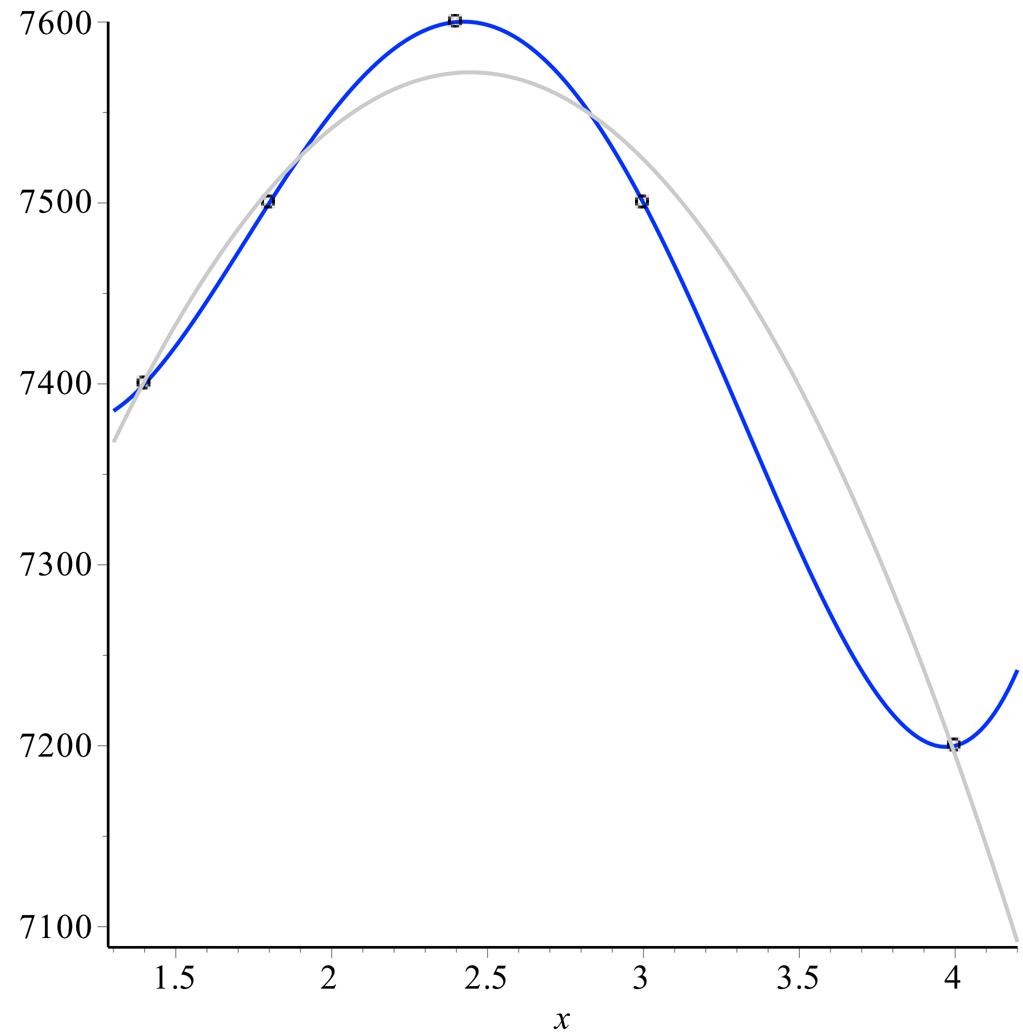


b)
Nyt haetaan interpolaatioplynomia. Sen asteluku = datapisteiden lkm -1 = 4. No siihen l"oytyy taas ihan
valmiskin funktio:
(Jos ratkaistaisiin LSQ-teht"av"an"a, matriisina olisi koko Vandermonde (transposilla kertominen voitaisiin "supistaa pois").)

```
> intpol:=interp(xd,yd,x);
```

$$intpol := 80.73523702\, x^4 - 815.8508161\, x^3 + 2777.073622\, x^2 - 3681.701634\, x + 9039.860141 \quad \textbf{(5.8)}$$

```
> display(plot(intpol,x=1.3..4.2,color=blue),plot(P,style=point,
  symbol=circle,color=black),plot(poly2,x=1.3..4.2,color=gray));
```



SVD:n ja QR-hajotelman k"aytt"o LSQ:ssa

Aiheesta jatkoty"oarkilla (ja jatkokursilla).

Hyv"a tiet"a"a: K"aytt"aj"a voi valita menetelm"an CurveFitting-pakkauksen LeastSquares-funktiossa.

Polynomin asteluvun nostaminen lis"a"a dramaattisesti h"airi"oaltiutta.