



13. előadás



Miről lesz szó?

- Függvények meghatározása

regresszió

példák a mérnöki gyakorlatból

szimbolikus regresszió

Függvények meghatározása

- alkalmazási példák
 - mért alakzatok geometriai jellemzőinek kiszámítása
 - kiegyenlítő felületek (mérnökgeodézia)
 - digitális felületmodellek felállítása
- rendelkezünk-e információval a függvény jellegéről?
 - igen: regresszió
 - nem: szimbolikus regresszió
- mely mennyiségeket tekintjük hibátlannak?

Regresszió

- a feladat ismert jellegű függvény ismeretlen paramétereinek a meghatározása
- lineáris-e a függvény a meghatározandó paraméterekre nézve?

igen (túlhatározott lin. egyenletrendszer)

részben (variable projection)

nem (Levenberg-Marquardt)

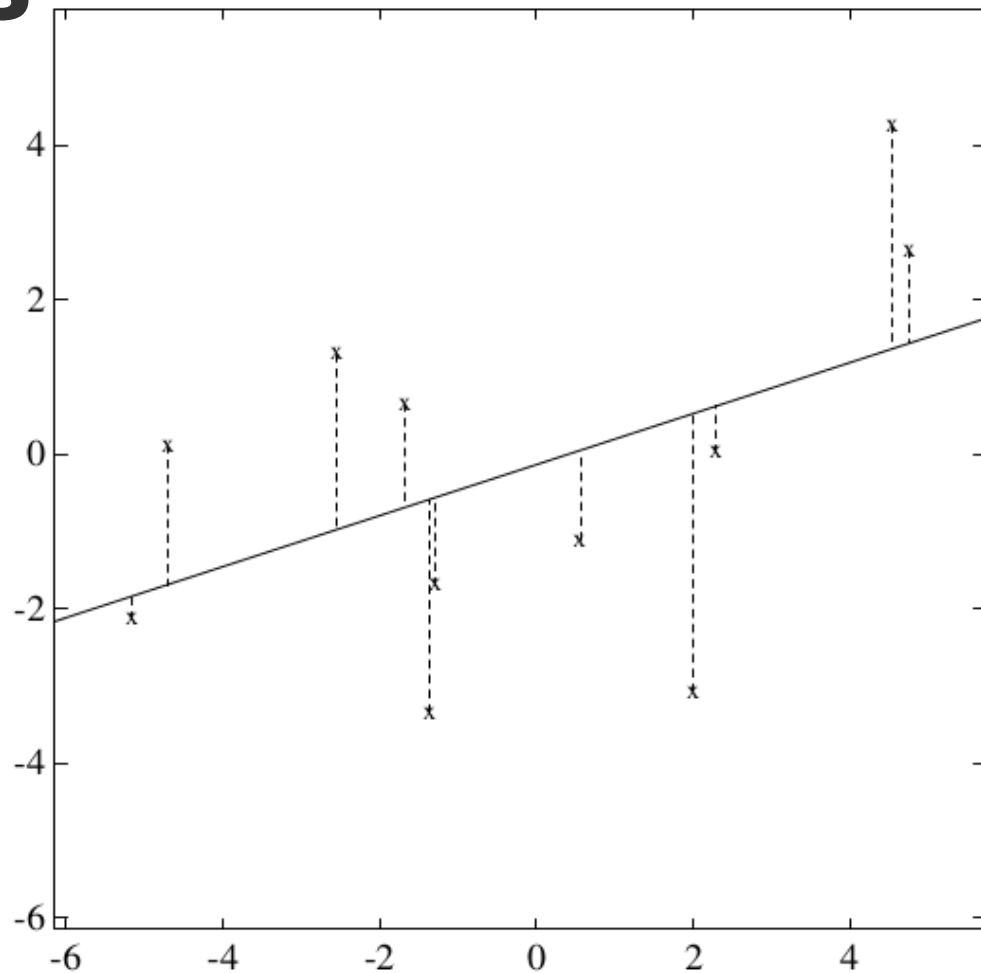


Regresszió

- egyenes illesztés
- egyenes sereg illesztés
- függvény meghatározás (láncgörbe illesztés)
- kiegyenlítő sík meghatározása
- kör, henger, gömb illesztés
- függvénySORBA fejtés

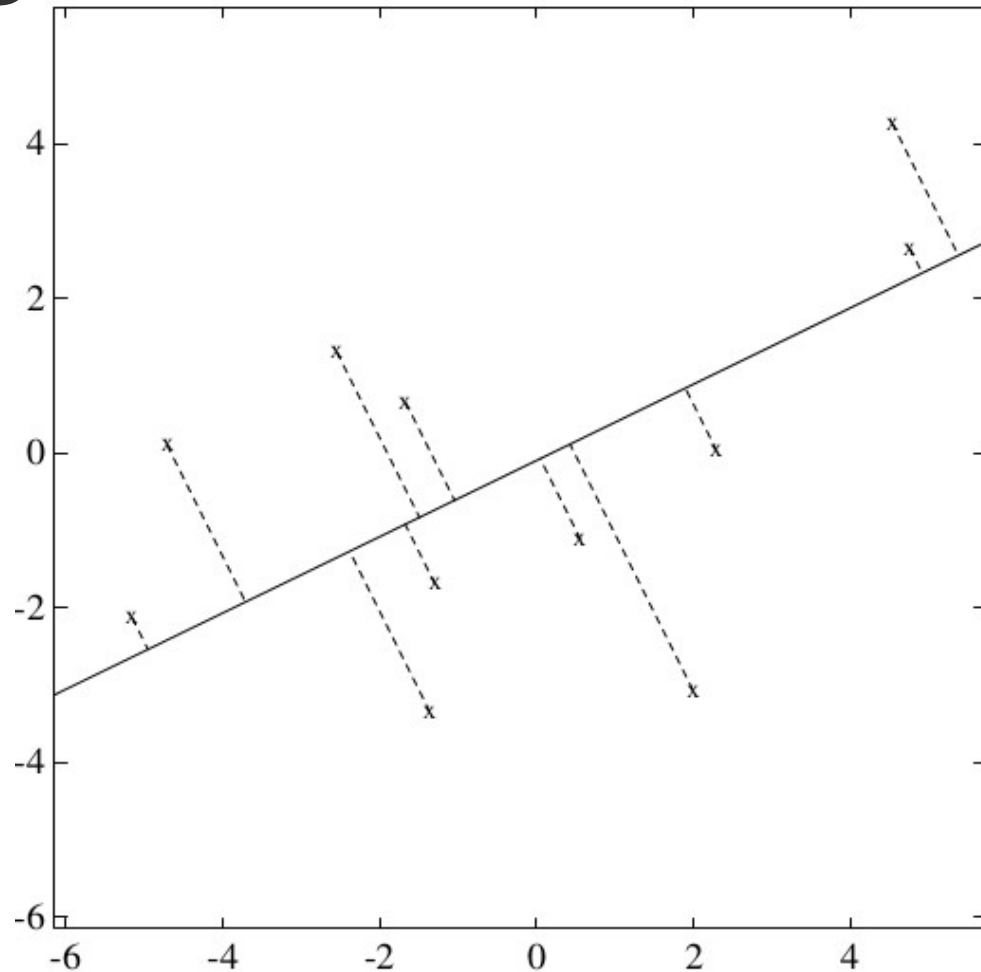
Egyenes illesztés

- egyszerű illesztés:
az x értékek
hibátlanok



Egyenes illesztés

- teljes illesztés:
az x értékek sem hibátlanok



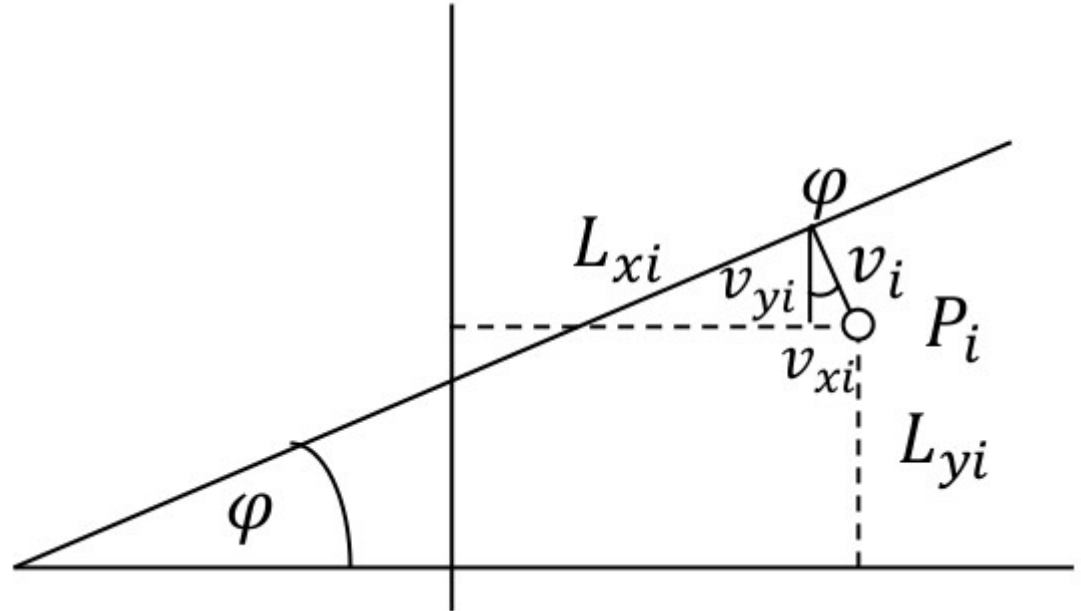
Egyenes illesztés

- Detrekői 11.3.3: kiegyenlítő egyenes
- mindkét koordináta hibával terhelt

$$y = m \cdot x + b$$

$$L_{yi} + v_{yi} = m(L_{xi} + v_{xi}) + b$$

$$v_i = (v_{xi}^2 + v_{yi}^2)^{1/2}$$



Egyenes illesztés

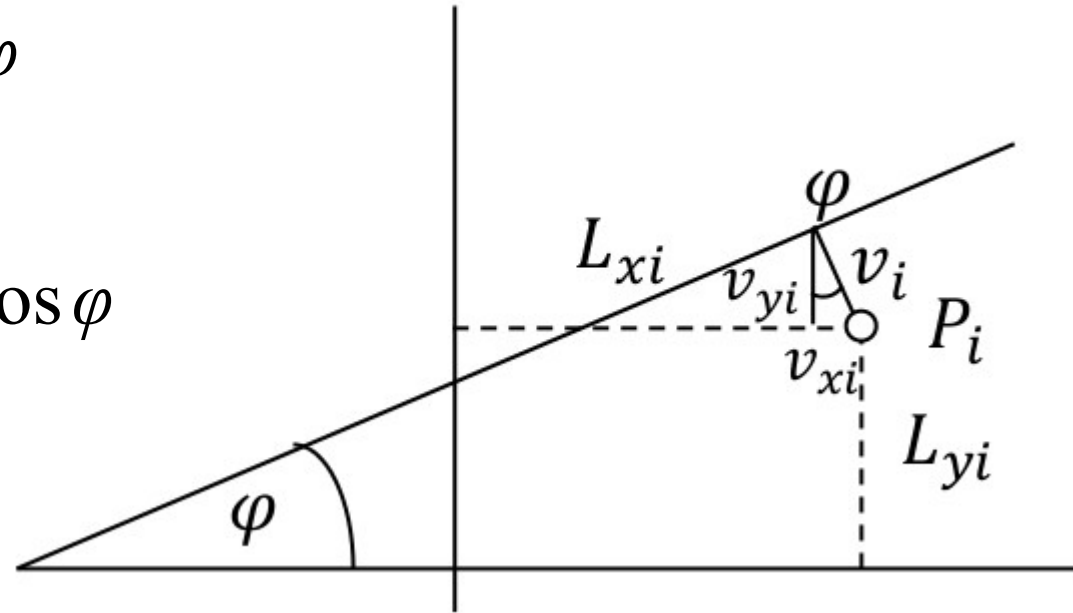
- Detrekői 11.3.3: kiegyenlítő egyenes
- mindkét koordináta hibával terhelt

$$y = m \cdot x + b \quad v_{xi} = -v_i \sin \varphi$$

$$v_{yi} = v_i \cos \varphi$$

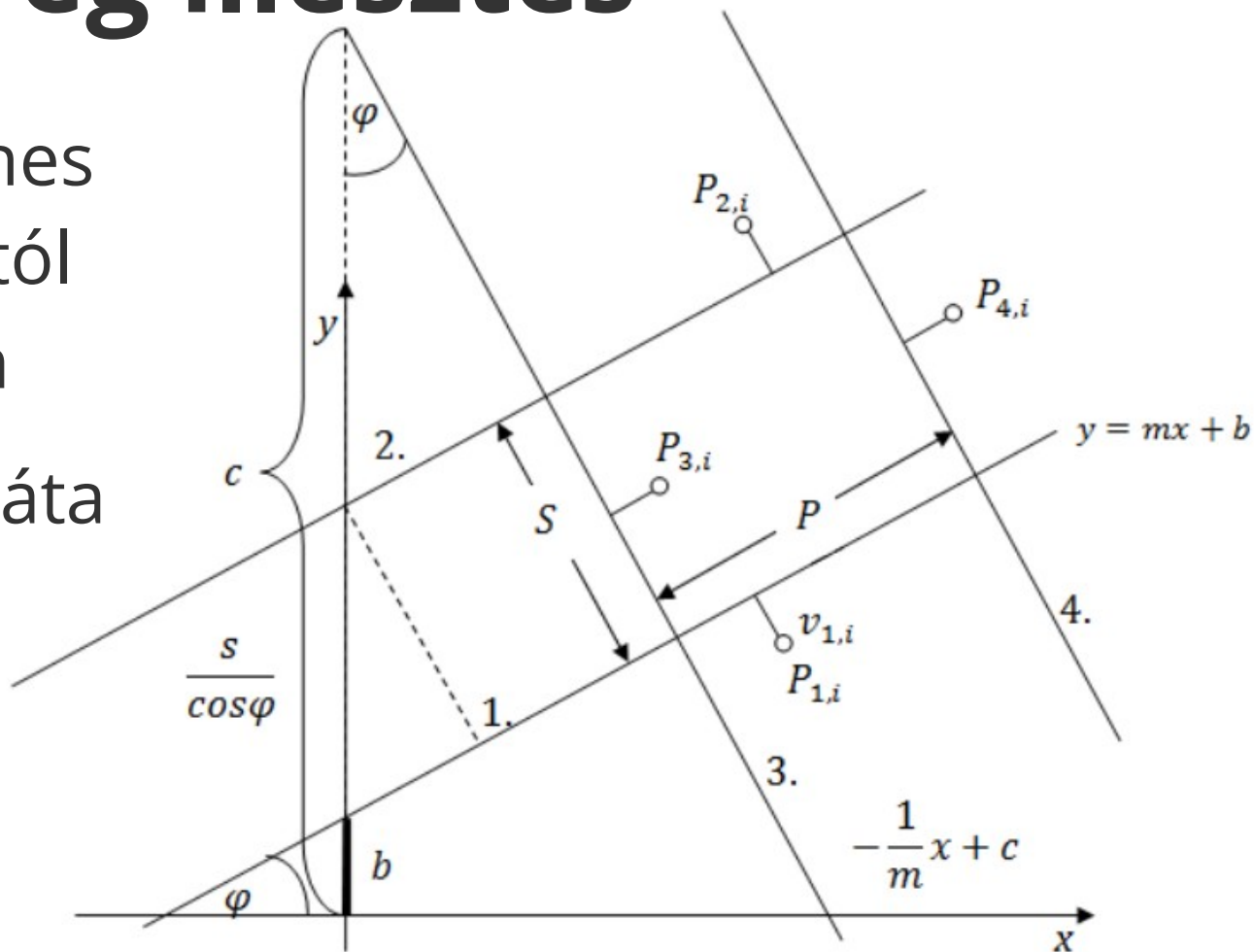
$$v_i = b \cos \varphi + L_{xi} \sin \varphi - L_{yi} \cos \varphi$$

$$\sum v_i^2 = \min$$



Egyenes sereg illesztés

- merőleges egyenes seregek egymástól p, s távolságban
- mindkét koordináta hibával terhelt



Egyenes sereg illesztés

- javítási egyenletek

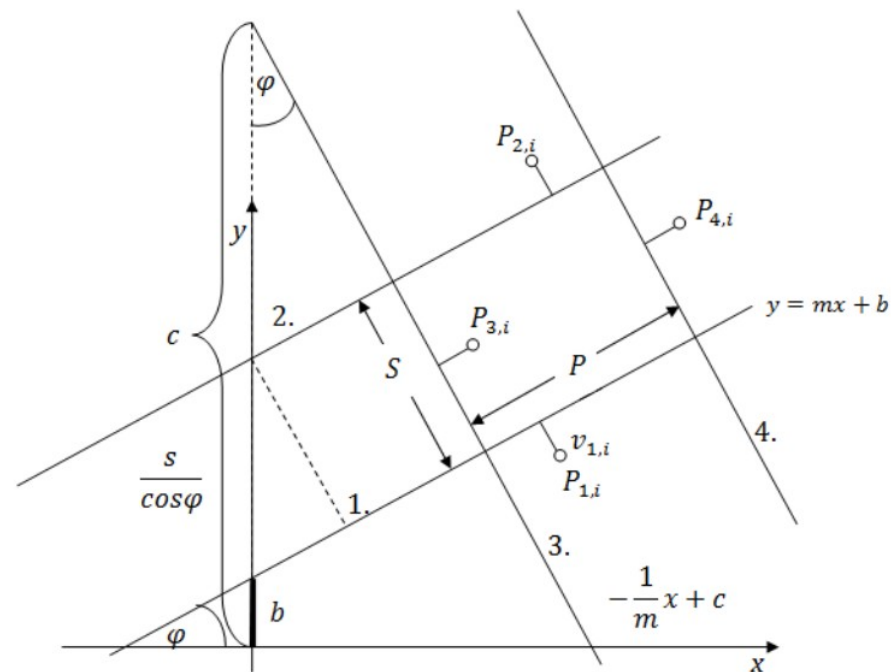
$$v_i = b \cos \varphi + L_{xi} \sin \varphi - L_{yi} \cos \varphi + s$$

$$v_i = c \sin \varphi - L_{xi} \cos \varphi - L_{yi} \sin \varphi + p$$

- alakmátrix, tisztatag vektor

$$A = \begin{bmatrix} -b_0 \cdot \sin \varphi_0 + L_{xi} \cdot \cos \varphi_0 + L_{yi} \cdot \sin \varphi_0 & \cos \varphi_0 & 0 \\ -b_0 \cdot \sin \varphi_0 + L_{xi} \cdot \cos \varphi_0 + L_{yi} \cdot \sin \varphi_0 & \cos \varphi_0 & 0 \\ \vdots & \vdots & \vdots \\ -c_0 \cdot \cos \varphi_0 + L_{xi} \cdot \sin \varphi_0 - L_{yi} \cdot \cos \varphi_0 & 0 & \sin \varphi_0 \\ -c_0 \cdot \cos \varphi_0 + L_{xi} \cdot \sin \varphi_0 - L_{yi} \cdot \cos \varphi_0 & 0 & \sin \varphi_0 \end{bmatrix}$$

$$l = \begin{bmatrix} -b_0 \cdot \cos \varphi_0 - L_{xi} \cdot \sin \varphi_0 + L_{yi} \cdot \cos \varphi_0 \\ -b_0 \cdot \cos \varphi_0 - L_{xi} \cdot \sin \varphi_0 + L_{yi} \cdot \cos \varphi_0 - s \\ \vdots \\ -c_0 \cdot \sin \varphi_0 - L_{xi} \cdot \cos \varphi_0 - L_{yi} \cdot \sin \varphi_0 \\ -c_0 \cdot \sin \varphi_0 - L_{xi} \cdot \cos \varphi_0 - L_{yi} \cdot \sin \varphi_0 - p \end{bmatrix}$$



$$x = (A^* \cdot A)^{-1} \cdot A^* \cdot l = \begin{bmatrix} \Delta \varphi \\ \Delta b \\ \Delta c \end{bmatrix}$$

$$v = A \cdot x - l$$

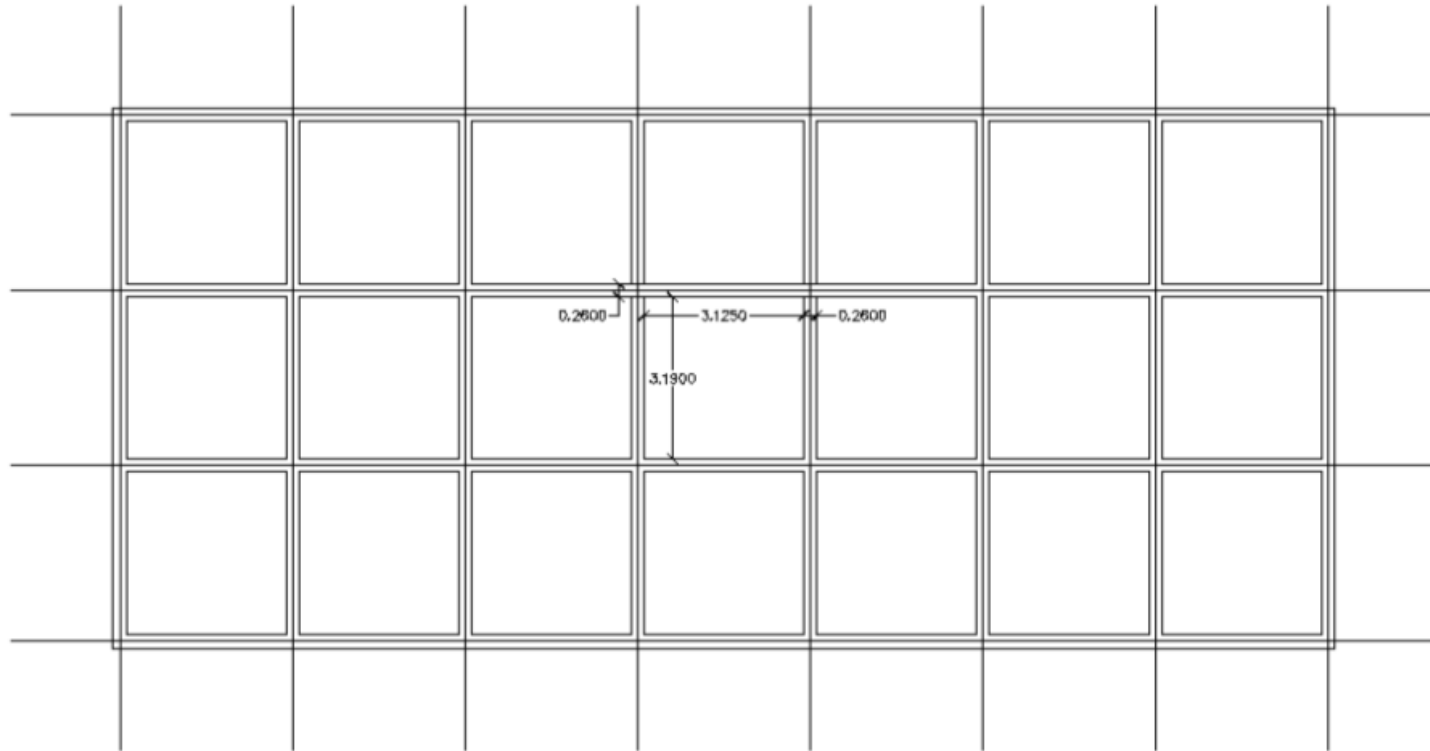
Egyenes sereg illesztés

- Kossuth téri látogatóközpont kazettás mennyezete (Vitányi A. TDK dolgozat, 2014)



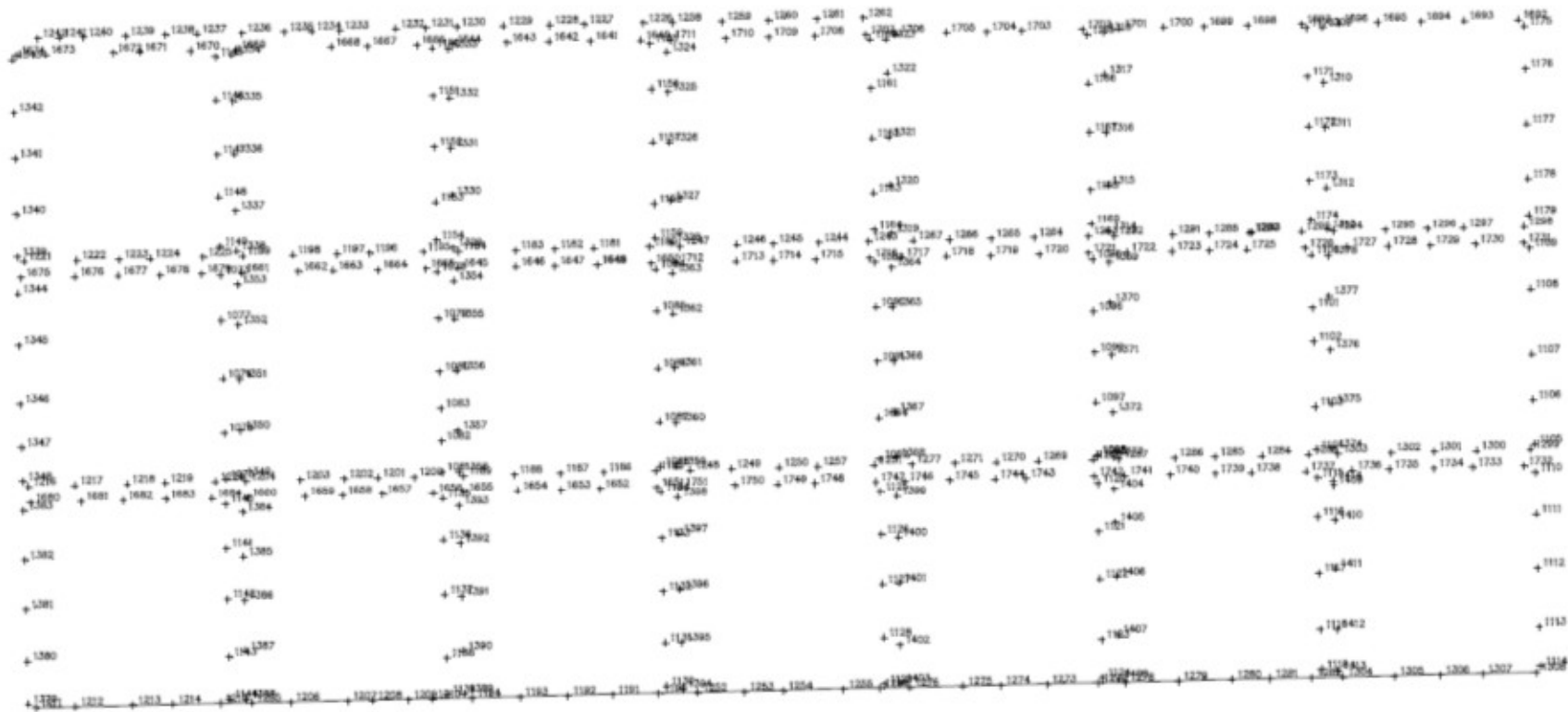
Egyenes sereg illesztés

- Tervezett geometria



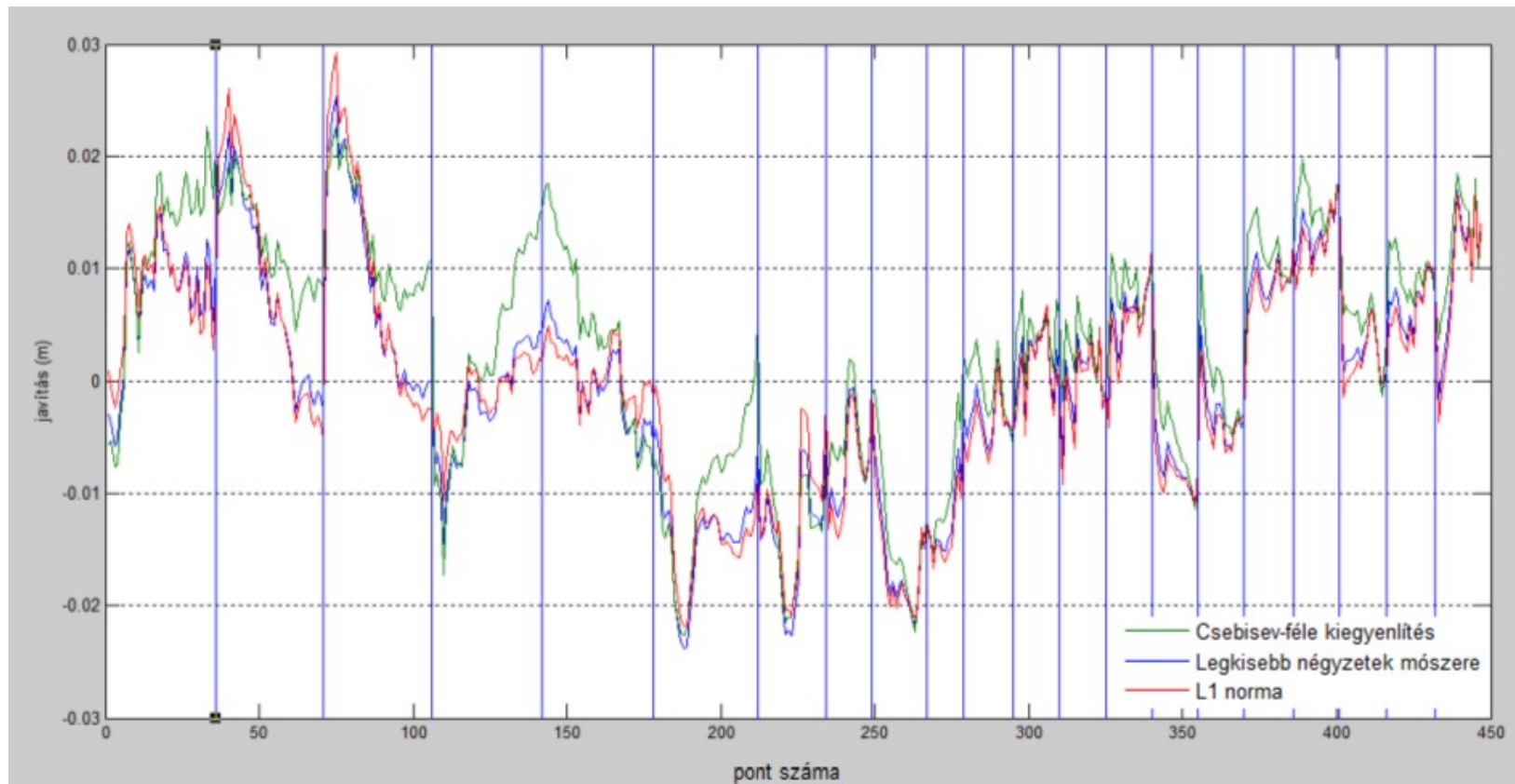
Egyenes sereg illesztés

- Mérőállomással mért pontok



Egyenes sereg illesztés

- L_1 , L_2 , L_∞ norma szerinti megoldások



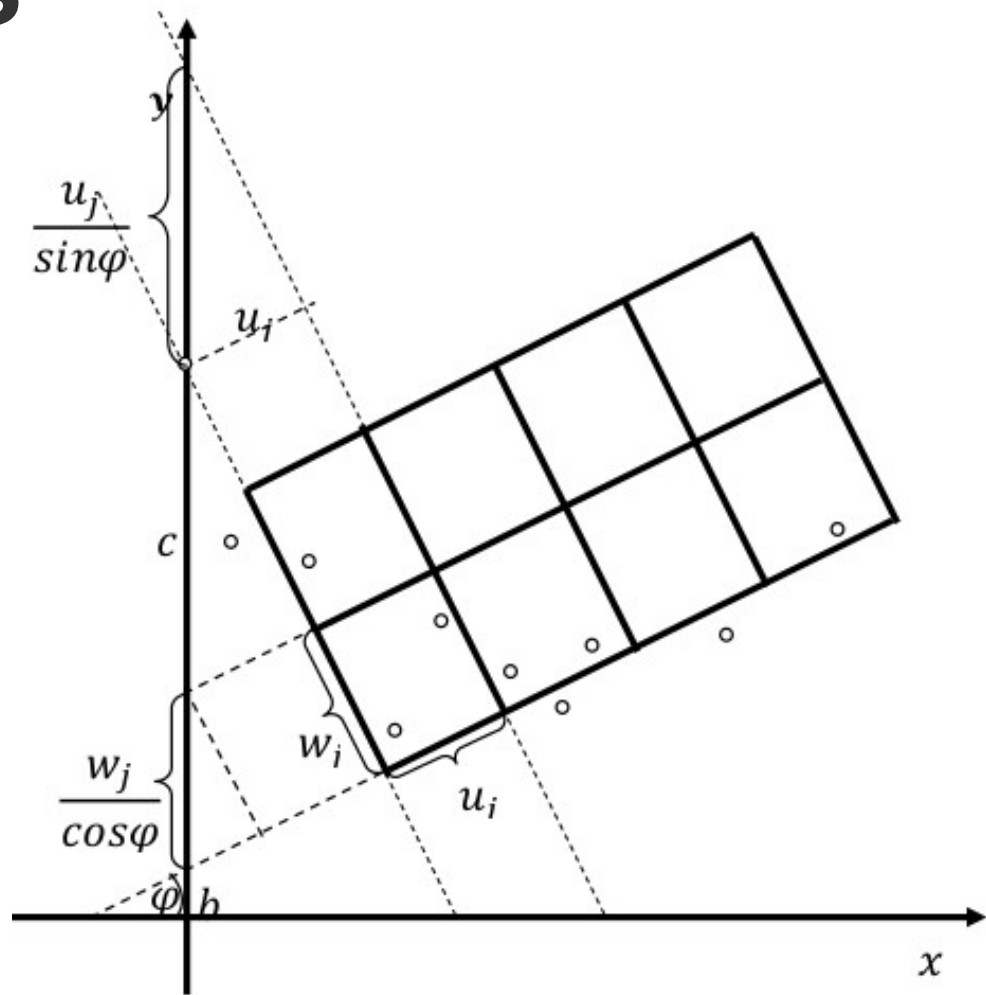
Egyenes sereg illesztés

- L_2 norma szerinti megoldás lineáris trendjei

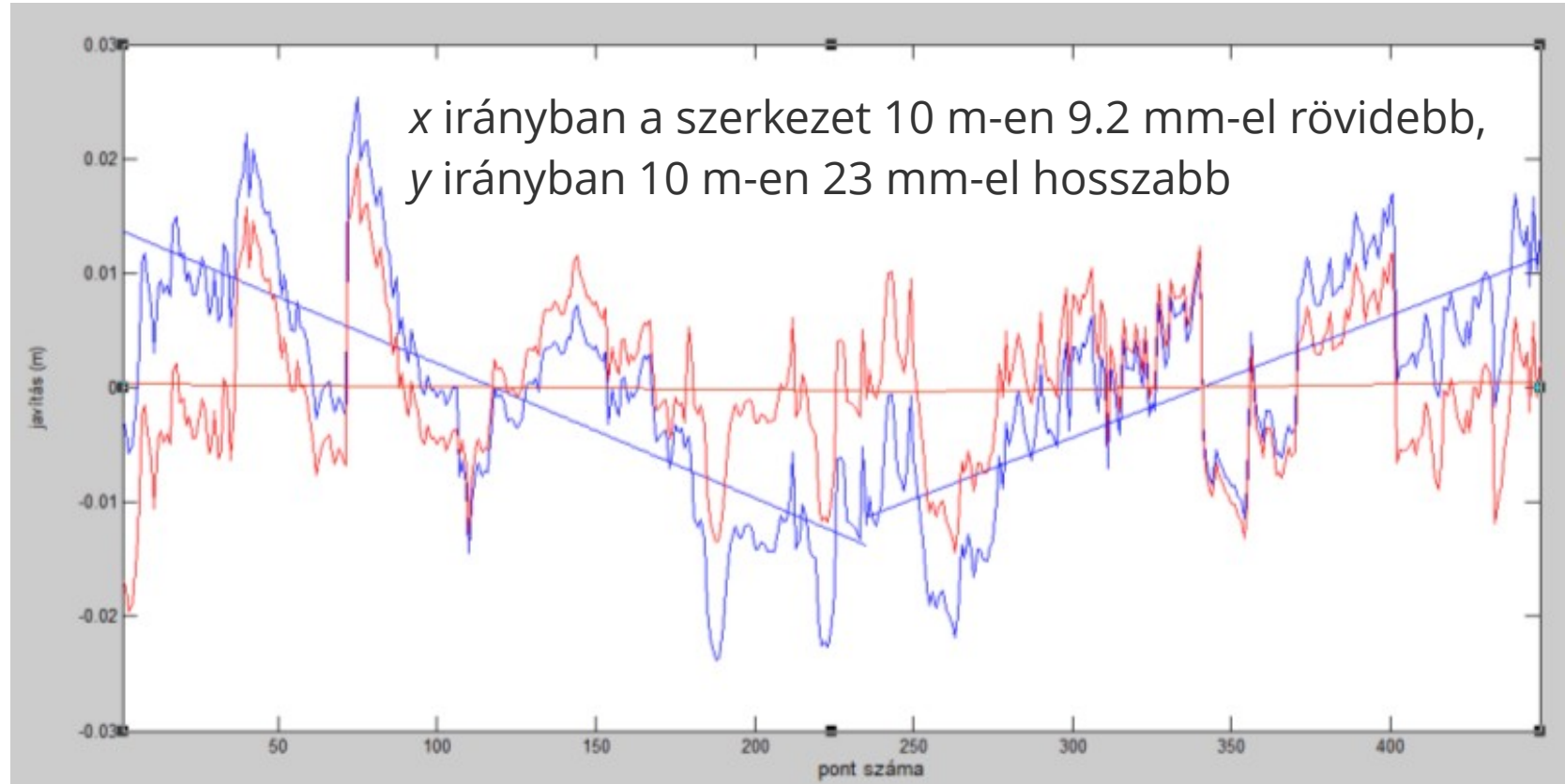


Egyenes sereg illesztés

- Kiegyenlítés két irányban eltérő méretarány tényezővel



- Kiegyenlítés két irányban eltérő méretarány tényezővel L_2 norma szerint



Függvény meghatározás

- Önsúlyukkal terhelt kötél jellegű befüggesztett szerkezetek alakja láncgörbe: z

$$z_0(x) = \frac{H_0}{\theta} \cdot \cosh\left(\frac{\theta}{H_0}(x + c_1)\right) + c_2$$

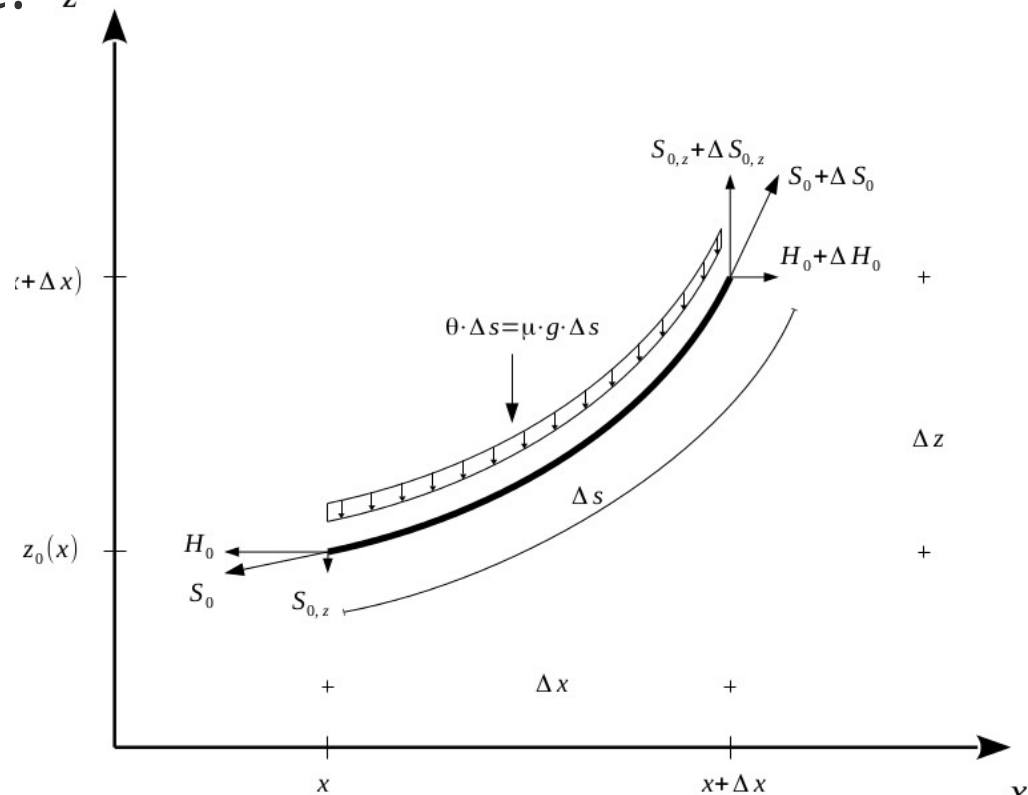
ahol $\theta = \mu \cdot g$

μ : a vezeték önsúlya

H_0 : vízszintes feszítőerő

meghatározandó:

$$H_0, \theta, c_1, c_2$$



Nagyfeszültségű távvezeték alakjának meghatározása

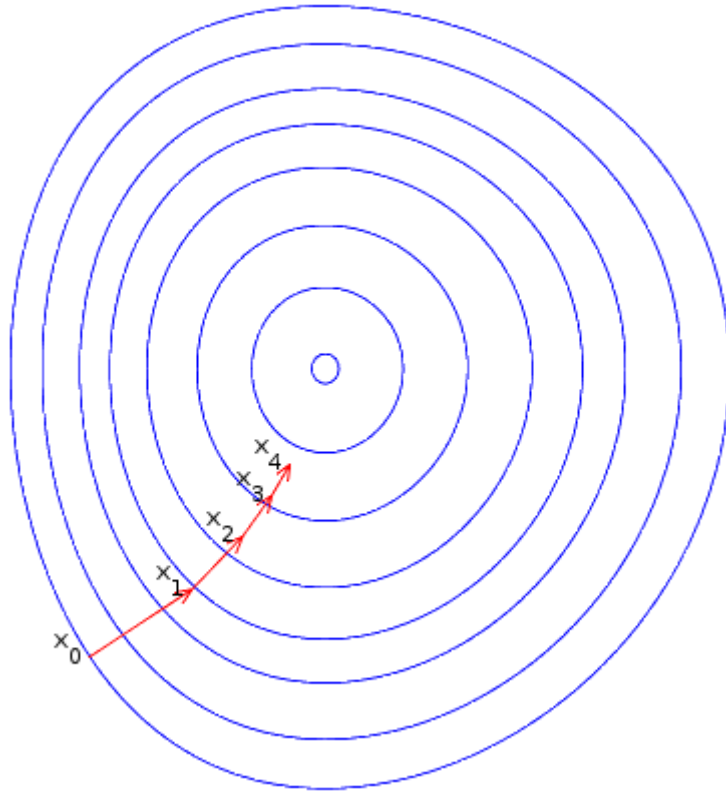
- Balogh Bálint (BSc diplomamunka, 2015)
- Leica TS15i robot mérőállomás
- Göd és Budafok



Nem lineáris regresszió számítási eljárása

- Matlab Curve Fitting Toolbox
a `fit` függvény a *Levenberg-Marquardt* eljárást alkalmazza a regresszióhoz
- Levenberg-Marquardt eljárás
nem lineáris LKN problémák megoldására
lokális minimum meghatározására alkalmas
Gauss-Newton és a legmeredekebb lejtő algoritmusok
csillapítási tényezőtől függő kombinációja
csillapítási tényező kezdetben nagy, azután csökken

Legmeredekebb lejtő algoritmus



$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma_n \cdot \nabla f(\mathbf{x}_n)$$

geometriai értelmezés:

minden iterációban az f felület \mathbf{x}_n pontban vett negatív gradiense ($-\nabla f$) irányában haladunk a minimum felé

Hátrány: a módszer cikcakkban haladva lassan konvergál hosszú keskeny „völgyekben”

Gauss-Newton módszer

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma_n \cdot [\mathbf{H}_f(\mathbf{x}_n)]^{-1} \nabla f(\mathbf{x}_n)$$

$\mathbf{H}_f(\mathbf{x})$: a Hesse – mátrix

geometriai értelmezés:

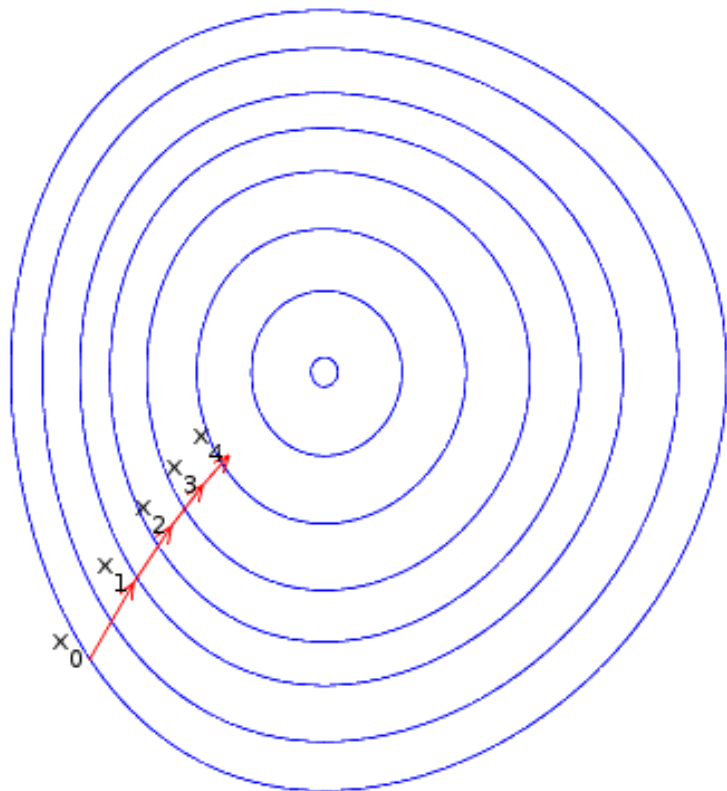
minden iterációban paraboloidot illesztünk az f felületre az \mathbf{x}_n pontban

a paraboloid gradiense és görbülete azonos az f gradiensével és görbületével

haladunk a paraboloid minimuma felé

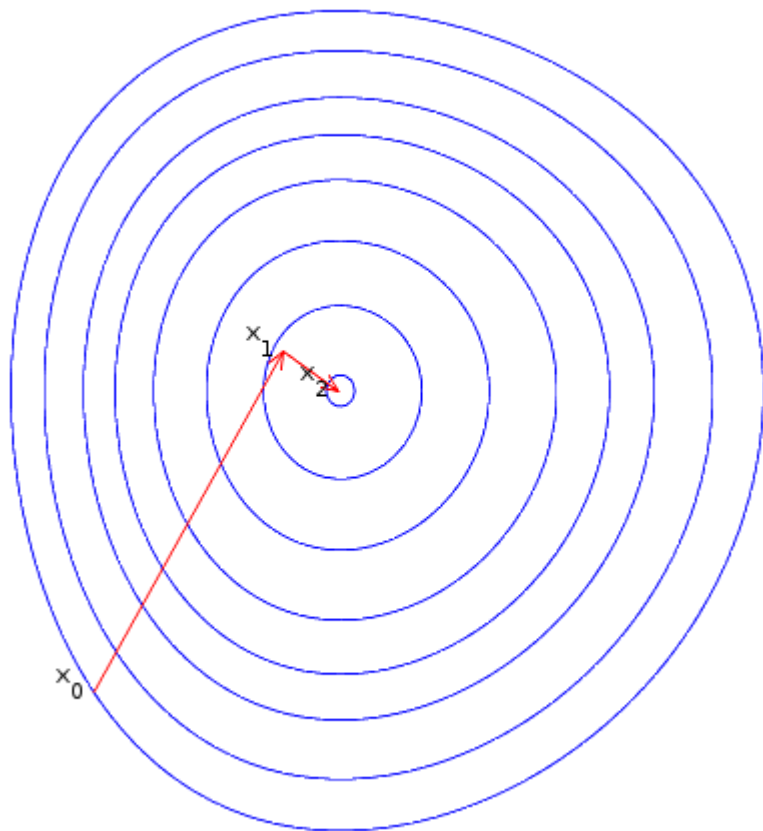
ha az f paraboloid, egy lépésben megkapjuk a minimumot

Hátrány: lassan vagy egyáltalán nem konvergál
rossz kezdőérték vagy rosszul kondicionált mátrix
esetében



Levenberg módszer

$$\mathbf{x}_{n+1} = \mathbf{x}_n - [\mathbf{H}_f(\mathbf{x}_n) + \lambda_n \cdot \mathbf{I}]^{-1} \nabla f(\mathbf{x}_n)$$



geometriai értelmezés:

legmeredekebb lejtő és gradiens módszer kombinációja

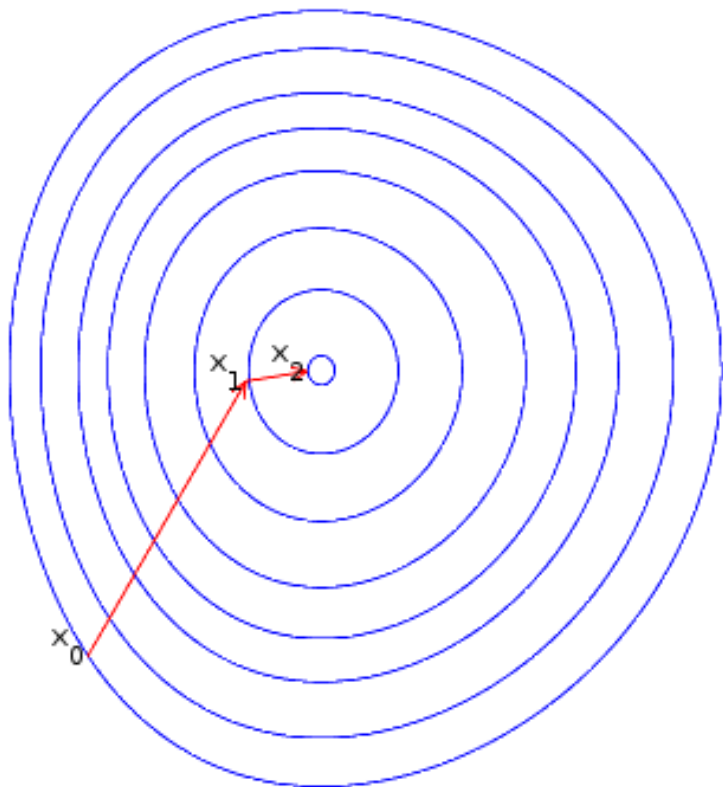
λ_n határozza meg, hogy milyen mértékben tekintjük a függvényt kvadratikushoz hasonlóknak

(ha λ_n zérus, teljesen kvadratikusnak vesszük és csak a Newton módszert alkalmazzuk; értéke az iteráció során változhat)

λ_n -t növeljük, ha felfelé léptünk, fordított esetben csökkentjük

Levenberg-Marquardt módszer

$$\mathbf{x}_{n+1} = \mathbf{x}_n - [\mathbf{H}_f(\mathbf{x}_n) + \lambda_n \cdot \text{diag}(\mathbf{H}_f(\mathbf{x}_n))]^{-1} \nabla f(\mathbf{x}_n)$$



geometriai értelmezés:

legmeredekebb lejtő és gradiens módszer kombinációja

a gradiens vektor összetevőit a görbület értéke szerint módosítjuk:

nagyobb lépés a kisebb görbület irányában

kisebb lépés a nagyobb görbület irányában

Heurisztikus, de a gyakorlatban jól bevált módszer

Levenberg-Marquardt eljárás görbeillesztéshez

- adott m adatpont (x_i, y_i) esetén határozzuk meg az $f(x, \boldsymbol{\beta})$ görbe $\boldsymbol{\beta}$ paramétereit (n db) a legkisebb négyzetes eltérések alapján:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^m [y_i - f(x_i, \boldsymbol{\beta})]^2$$

- iteráció, lépésenként változik a $\boldsymbol{\beta}$ vektor $\boldsymbol{\delta}$ -val (\boldsymbol{J} a parciális deriváltak $m \times n$ -es Jacobi mátrixa)

$$f(x, \boldsymbol{\beta} + \boldsymbol{\delta}) \approx f(x, \boldsymbol{\beta}) + \boldsymbol{J} \boldsymbol{\delta}$$

Levenberg-Marquardt eljárás

- megoldás a δ -ra:

$$(\mathbf{J}^T \mathbf{J}) \delta = \mathbf{J}^T [\mathbf{y} - f(\boldsymbol{\beta})]$$

- λ csillapítási tényezővel:

$$(\mathbf{J}^T \mathbf{J} + \lambda \mathbf{I}) \delta = \mathbf{J}^T [\mathbf{y} - f(\boldsymbol{\beta})]$$

- Marquardt javításával:

$$[\mathbf{J}^T \mathbf{J} + \lambda \text{diag}(\mathbf{J}^T \mathbf{J})] \delta = \mathbf{J}^T [\mathbf{y} - f(\boldsymbol{\beta})]$$

- (hasonló a Tyihonov regularizációhoz)
- ha $\lambda \approx 0$, Gauss-Newton iteráció, ha λ nagy, legmeredekebb lejtő módszere lesz

Tyihonov regularizáció

- LKN megoldást keressük \mathbf{x} -re:

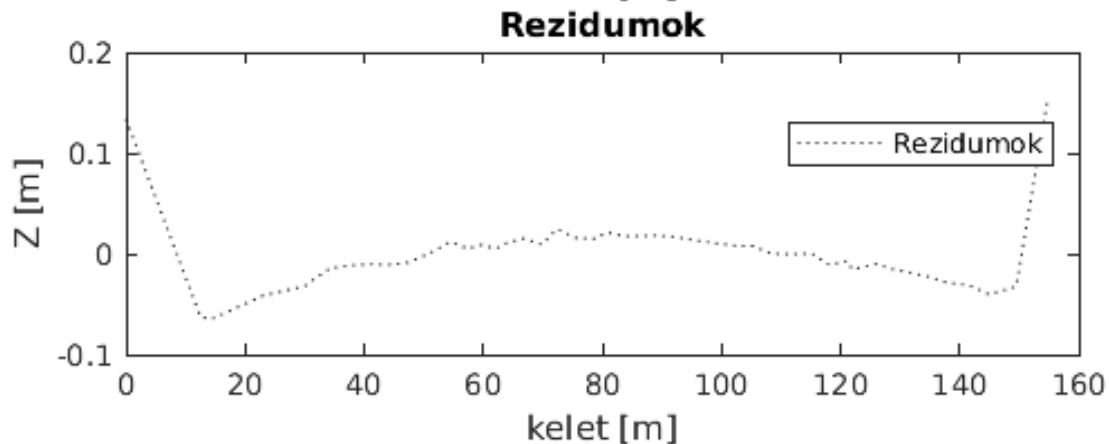
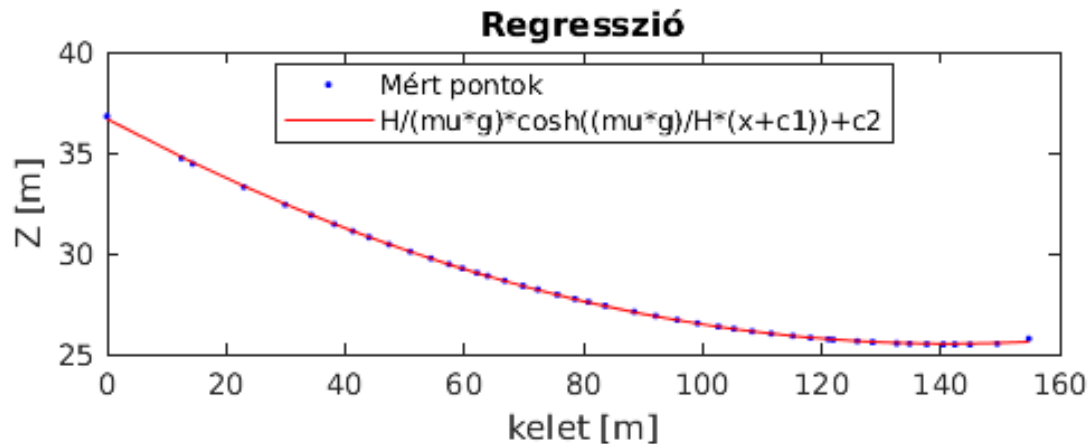
$$\mathbf{A}\mathbf{x} \approx \mathbf{b} \qquad \hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$$

- Ha \mathbf{A} simító művelet, vagyis \mathbf{b} simább \mathbf{x} -nél, akkor az inverz feladat növeli a zajt
- Megoldás: regularizáció a $\mathbf{\Gamma}$ Tyihonov mátrixszal

$$\hat{\mathbf{x}} = (\mathbf{A}^T \mathbf{A} + \mathbf{\Gamma}^T \mathbf{\Gamma})^{-1} \mathbf{A}^T \mathbf{b}$$

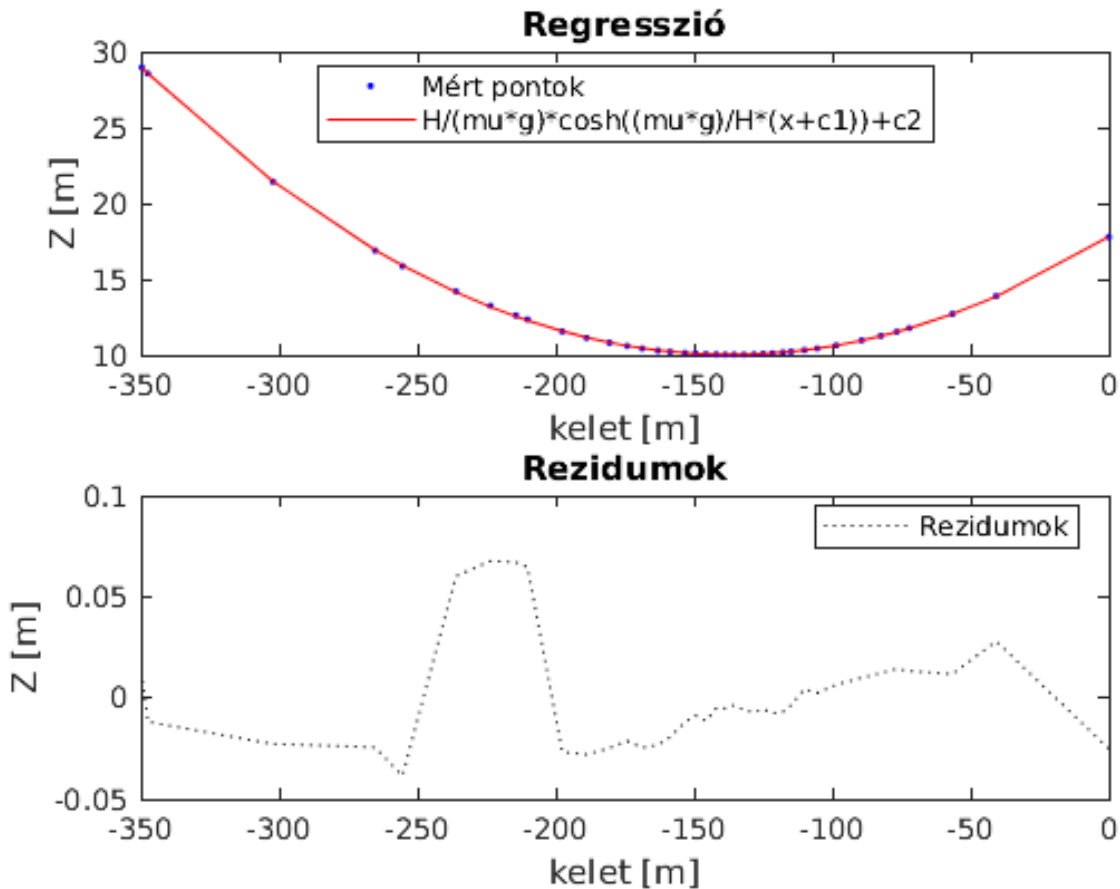
- A $\mathbf{\Gamma}$ Tyihonov mátrix gyakran skálázott egységmátrix, $\mathbf{\Gamma} = \alpha \cdot \mathbf{I}$, ahol α pozitív regularizációs (simító) paraméter

Budafoki távvezeték



```
Eredmények
-----Mérési fájl név-----
BUDAFOK_távvezeték.xlsx
-----Függvény-----
H/(mu*g)*cosh((mu*g)/H*(x+c1))+c2
-----Illesztés típusa-----
customnonlinear
-----Illesztési kategória-----
custom
-----Illesztési algoritmus-----
trust-region-reflective
-----Illesztési eljárás-----
NonlinearLeastSquares
-----Iterációk száma-----
18 db
-----Illesztés minősítése-----
SSE= 0.063891
R^2= 0.99983
adjR^2= 0.99982
RMSE= 0.039476
-----Paraméter értékek-----
H = 17.126_[kN]
c1=-141.729_[m]
c2= -876.69_[m]
g = 9.81_[kg*m/s^2]
mu= 1.935_[kg/m]
-----Egyéb adatok, jellemzők-----
df1 = 11.1552_[m]
df2 = 0.0947697_[m]
xmin= 141.729_[m]
fmin= 25.5176_[m]
l = 155.391_[m]
L = 154.807_[m]
l_L = 1.00377_-]
-----Feszítőerők-----
S_max = 17.34_[kN]
Sz_max=0.2482_[kN]
Si = -2.701_[kN]
Sj = 0.2482_[kN]
-----Húzófeszültségek-----
SzigmaS_max= 30.42_[N/mm^2]
SzigmaH = 30.05_[N/mm^2]
SzigmaZ_max=0.4355_[N/mm^2]
SzigmaSi = 30.42_[N/mm^2]
SzigmaSj = 30.05_[N/mm^2]
-----Végponti érintők-----
tan_i=-9 2 16_[DMS]
tan_j= 0 49 50_[DMS]
-----Illesztési hibák-----
maxEltérés = +15.4 cm
átlaEltérés = -0.0 cm
minEltérés = -6.5 cm
OK
```

Gödi távvezeték



Eredmények

Mérési fájl név-----
GÖD_dunaátvezető_2oszlopköz.xlsx

Függvény-----
 $H/(\mu \cdot g) \cdot \cosh\left(\frac{\mu \cdot g}{H} \cdot (x + c1)\right) + c2$

Illesztés típusa-----
customnonlinear

Illesztési kategória-----
custom

Illesztési algoritmus-----
trust-region-reflective

Illesztési eljárás-----
NonlinearLeastSquares

Iterációk száma-----
18 db

Illesztés minősítése-----
SSE= 0.026742
R²= 0.99997
adjR²= 0.99996
RMSE= 0.028045

Paraméter értékek-----
H = 12.2551_[kN]
c1= 136.623_[m]
c2=-1192.33_[m]
g = 9.81_[kg*m/s²]
mu = 1.039_[kg/m]

Egyéb adatok, jellemzők-----
df1 = 18.9171_[m]
df2 = 7.77049_[m]
xmin=-136.623_[m]
fmin= 10.0298_[m]
l = 362.698_[m]
L = 349.629_[m]
l_L = 1.03738_-]

Feszítőerők-----
S_max = 12.45_[kN]
S_z_max = 1.396_[kN]
S_i = 1.396_[kN]
S_j = -2.182_[kN]

Húzófeszültségek-----
SzigmaS_max=42.45_[N/mm²]
SzigmaH = 41.79_[N/mm²]
SzigmaZ_max=4.759_[N/mm²]
SzigmaSi = 42.06_[N/mm²]
SzigmaSj = 42.45_[N/mm²]

Végponti érintők-----
tan_i= 6 31 28_[DMS]
tan_j=-10 12 13_[DMS]

Illesztési hibák-----
maxEltérés = +6.8 cm
átlaEltérés = -0.0 cm
minEltérés = -3.9 cm

OK

Kiegyenlítő sík illesztése

- Lineáris LKN probléma

legjobban illeszkedő sík:

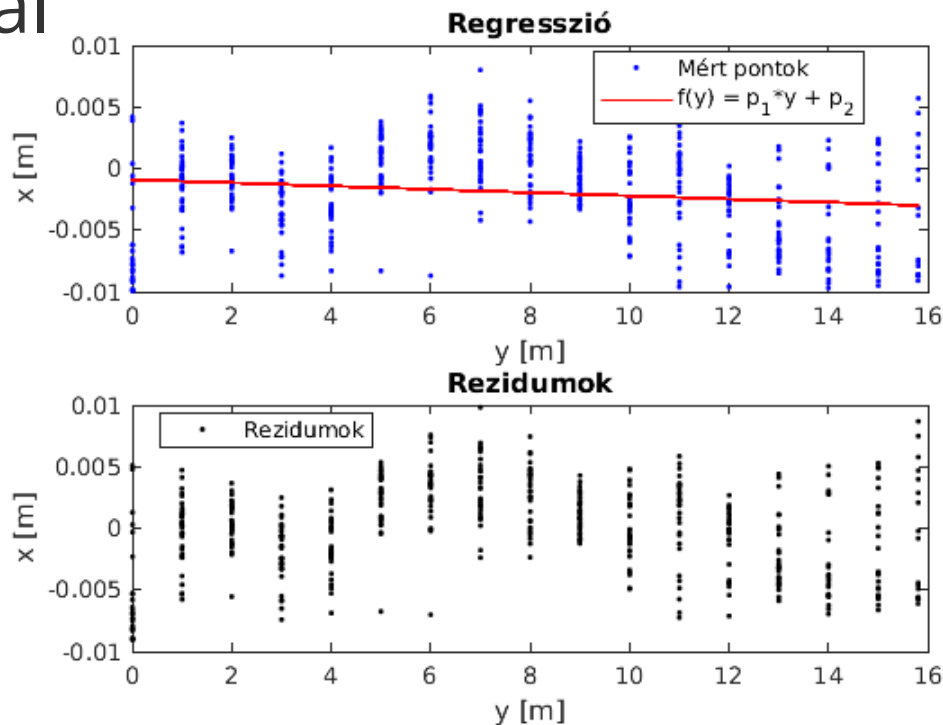
$$f(y, z) = p_{00} + p_{10}y + p_{01}z$$

függőleges sík illesztése esetén valójában csak egyenes illesztésre van szükség:

$$f(y) = p_1y + p_2$$

Függőleges sík illesztése

- MÜPA szkennelése robot mérőállomással



```
Eredmények
-----Mérési fájl név-----
MÜPA_rövidoldal_gyors_scann.xlsx
-----Függvény-----
p1*x + p2
-----Illesztés típusa-----
poly1
-----Illesztési kategória-----
library
-----Illesztési algoritmus-----
QR factorization and solve
-----Illesztési eljárás-----
LinearLeastSquares
-----Iterációk száma-----
1 db
-----Illesztés minősítése-----
SSE = 0.0063853
R^2 = 0.027173
adjR^2 = 0.025099
RMSE = 0.0036898
-----Paraméter értékek-----
p1=-0.000132256[-]
p2=-0.000889162[m]
-----Szóródó pontok-----
Összes mért pont = 527 db
Jó pontok = 471 db
Kivágó pontok = 56 db
Felső határ = 0.01 m
Alsó határ = -0.01 m
-----Sík Illesztési hibák-----
maxEltérés = +9.8 mm
átlaEltérés = -0.0 mm
minEltérés = -9.0 mm
OK
```


Gömb illesztése m pontra

- Mindegyik pontra illeszkedik az r sugarú gömb:

$$\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} - r = 0, \quad i = 1, \dots, m$$

ezért mindegyik pontra

$$\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} + r = 2r = c = \text{áll.}$$

- A két egyenletet összeszorozva

$$(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2 - r^2 = 0$$

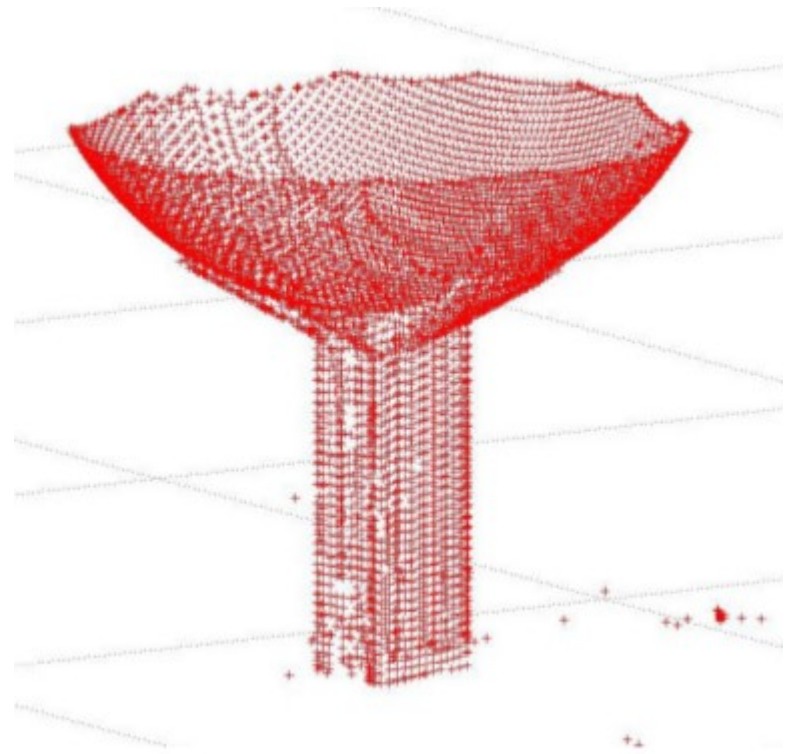
- Új α ismeretlent bevezetve r helyett túlhatározott *lineáris* egyenletrendszerrel kell megoldani:

$$\alpha = x_0^2 + y_0^2 + z_0^2 - r^2$$

$$x_i^2 + y_i^2 + z_i^2 - 2x_0x_i - 2y_0y_i - 2z_0z_i + \alpha = 0, \quad i = 1, \dots, m$$

Budafoki víztorony

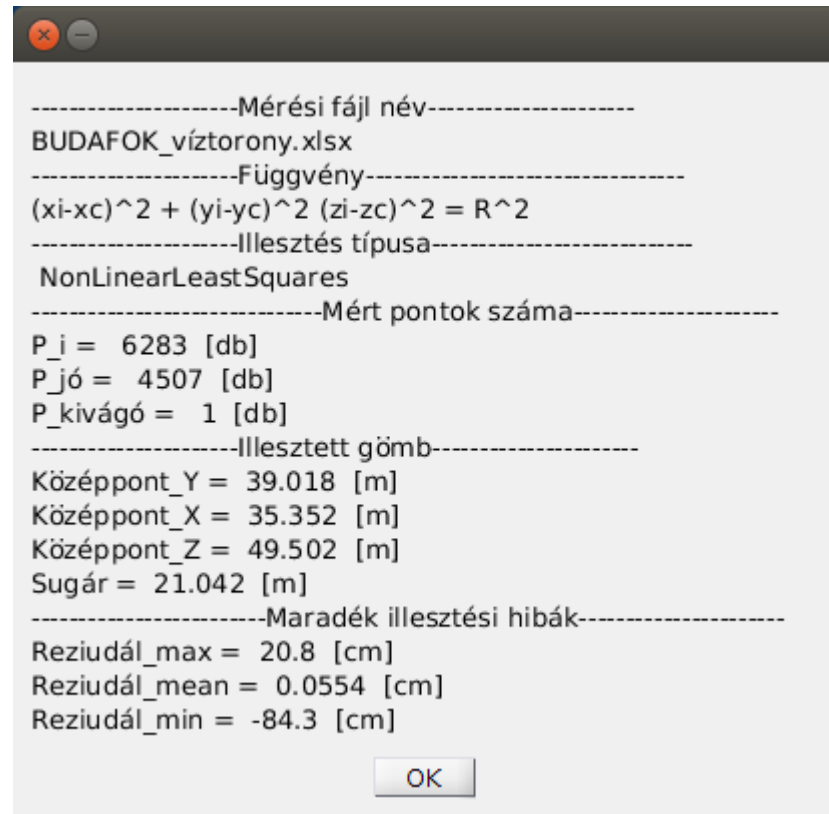
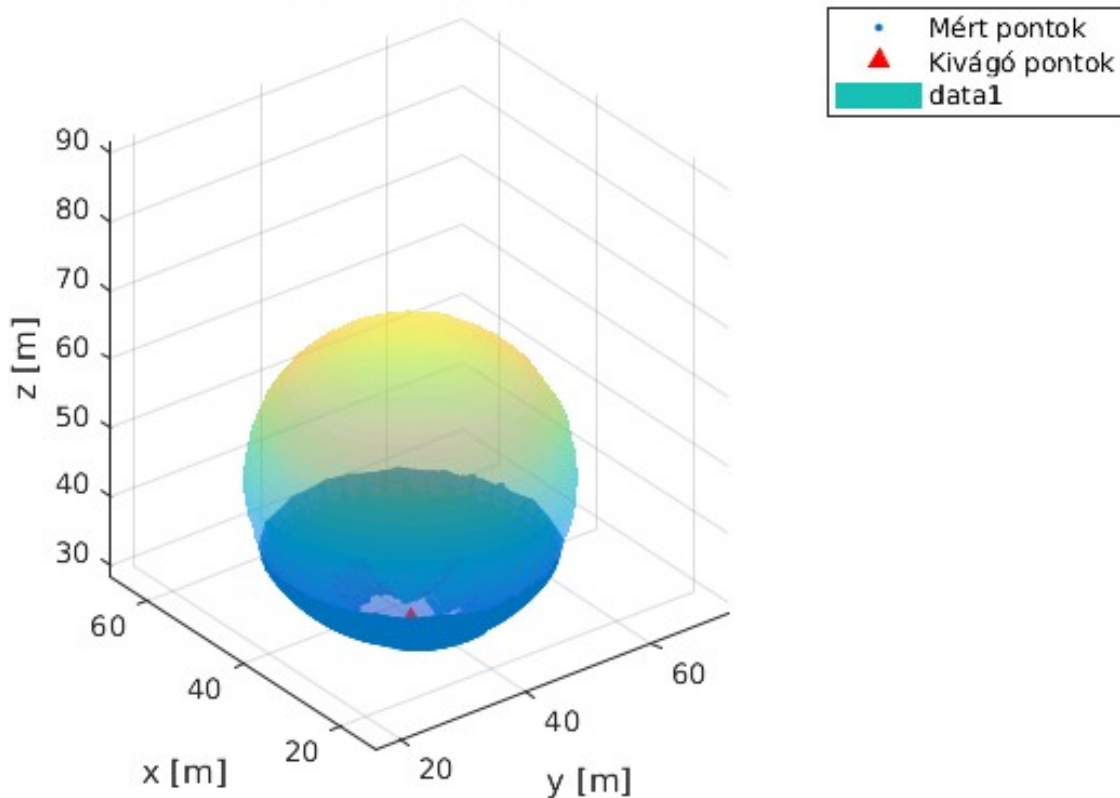
- Robot mérőállomással szkennelt pontfelhő



Budafoki víztorony

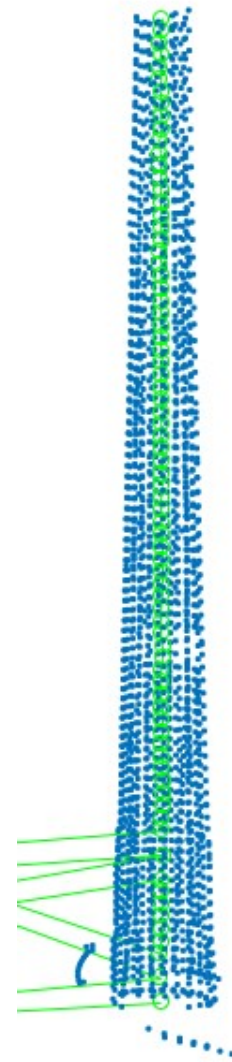
- gömb illesztése a tartályra

Gömb illesztés



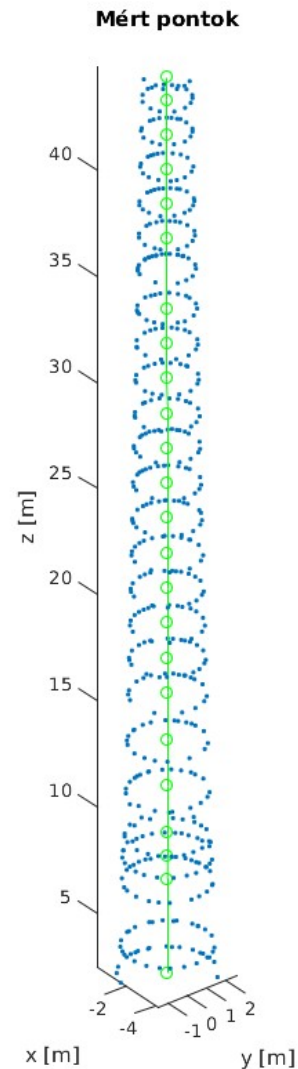
Henger tengelyének meghatározása

- Óbudai gyárkémény palástjának szkennelése



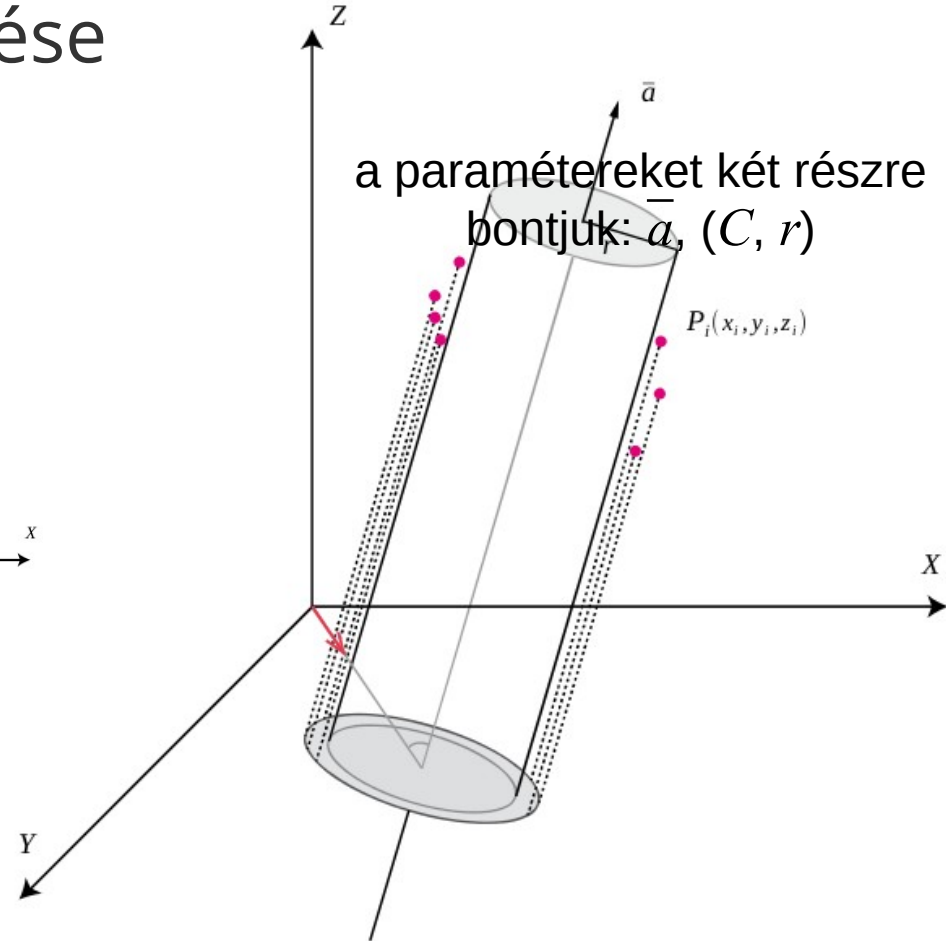
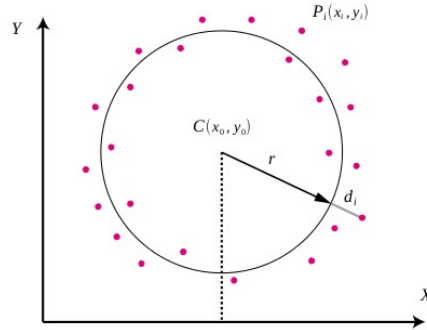
Henger tengelyének meghatározása

- körök illesztése a tengely meghatározásához



Henger palást meghatározása

- Csepeli víztározó szkennelése



Szimbolikus regresszió

- A szimbolikus regresszió olyan eljárás, amelynek során az adatokhoz legjobban illeszkedő modellt keresünk, amelynek a matematikai alakja *nem ismert*

matematikai *kifejezéseket* építünk fel

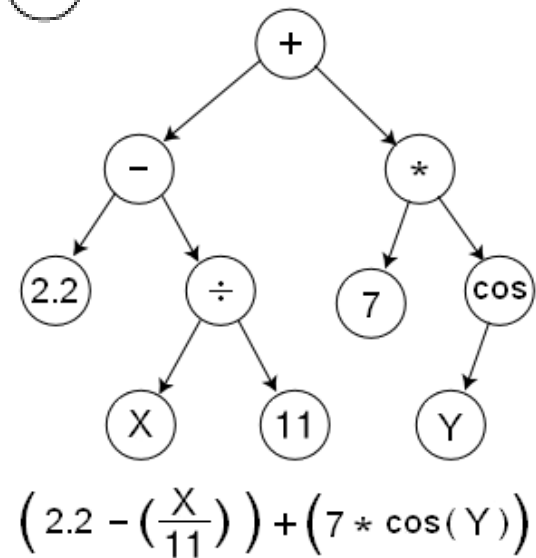
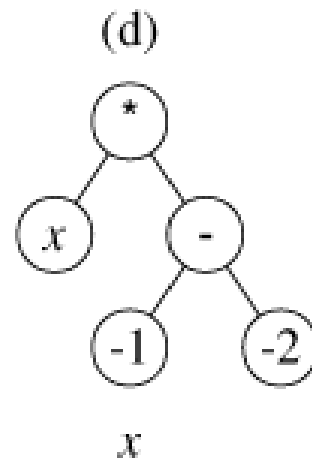
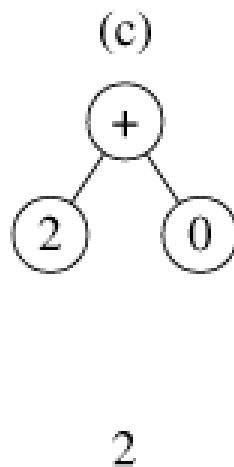
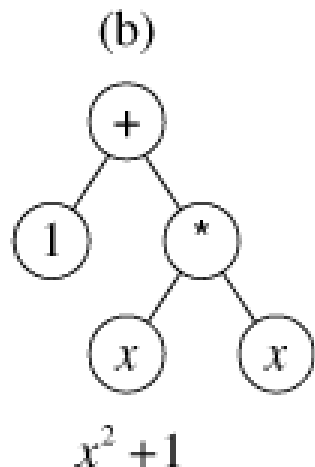
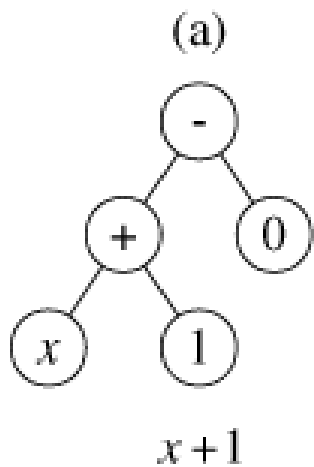
véletlenszerűen (genetikus algoritmus)

determinisztikusan (PGE)

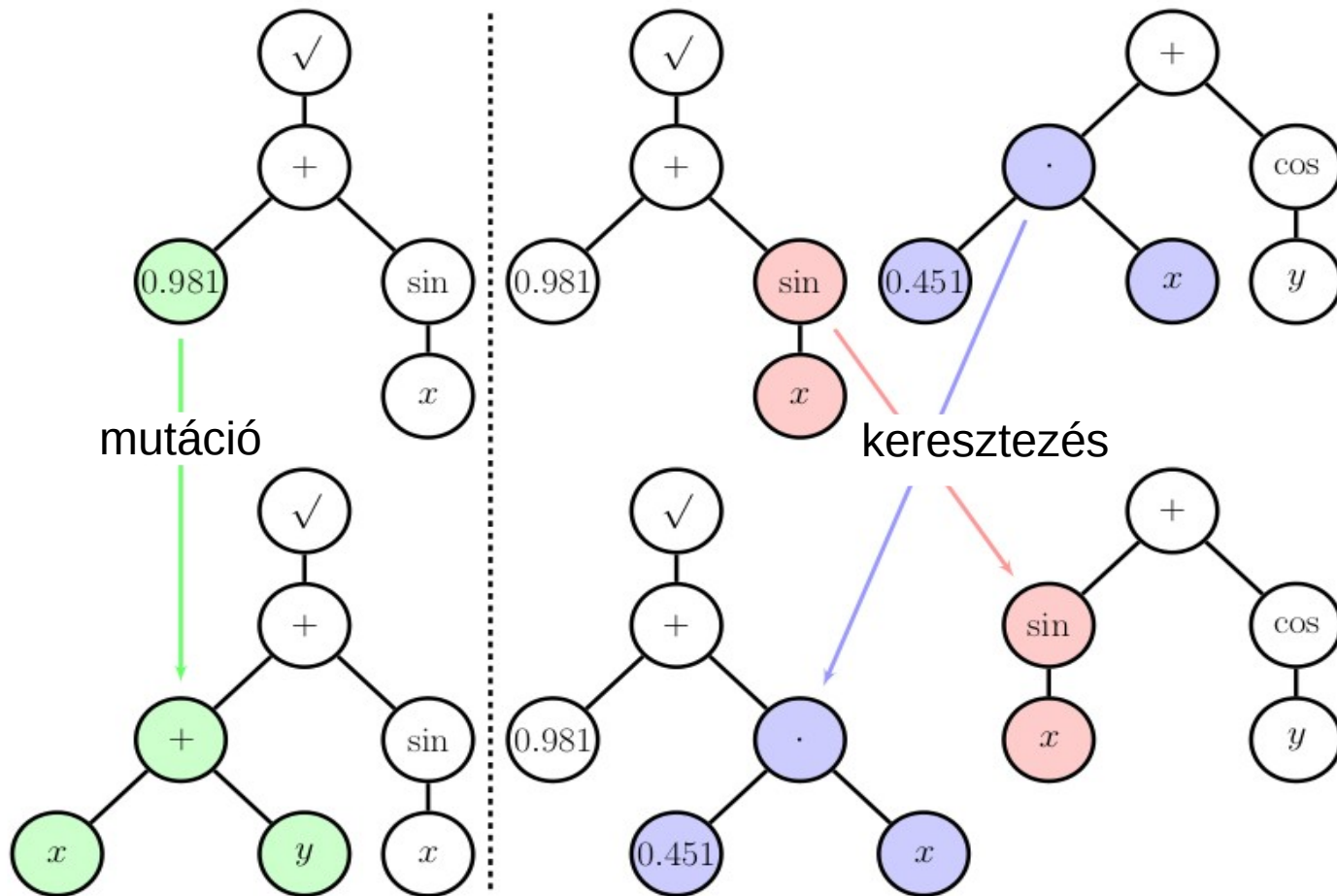
meghatározzuk a kifejezés *paramétereit*, hogy a legjobban illeszkedjen az adatokra

megtartjuk a legjobb modellt és paramétereit

Matematikai kifejezések fáí



Genetikus algoritmus

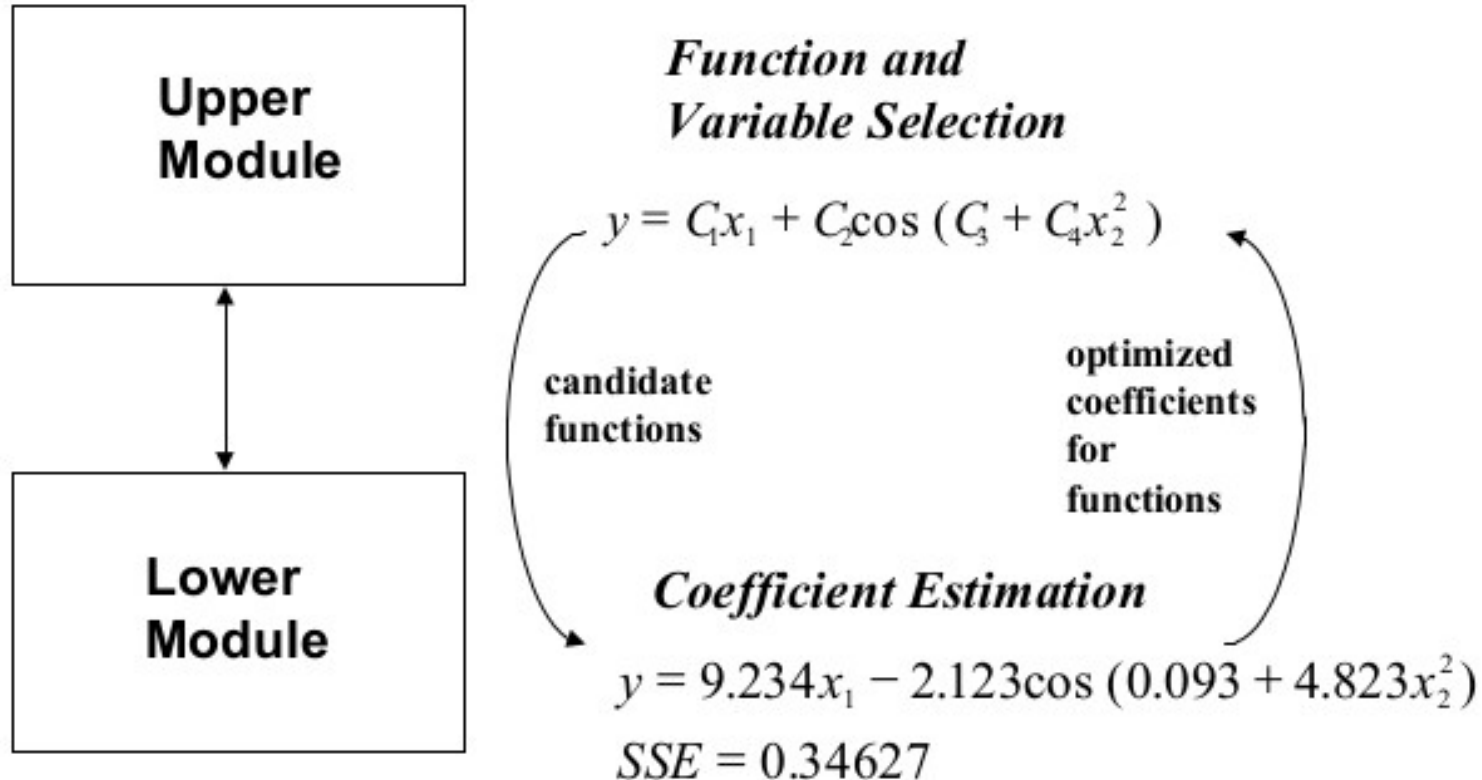


Hierarchikus genetikus algoritmus

- mindegyik kifejezés fa egy „kromoszóma” (egyed, amelyik „mutálódhat” és „kereszteződhet” egy másik „kromoszómával”)
- kell egy „rátermettség” függvény: ez megmondja, hogy egy kifejezés mennyire jól adja vissza az adatokat (pl. átlag négyzetes hiba)

egy második genetikus algoritmus határozza meg egy kifejezés fa optimális együtthatóit

Hierarchikus GA (Gulsen and Smith, 1998))



Példa

- Határozzuk meg szimbolikus regresszióval azt az $f(x, y, z)$ függvényt mely az alábbi értékeket adja:

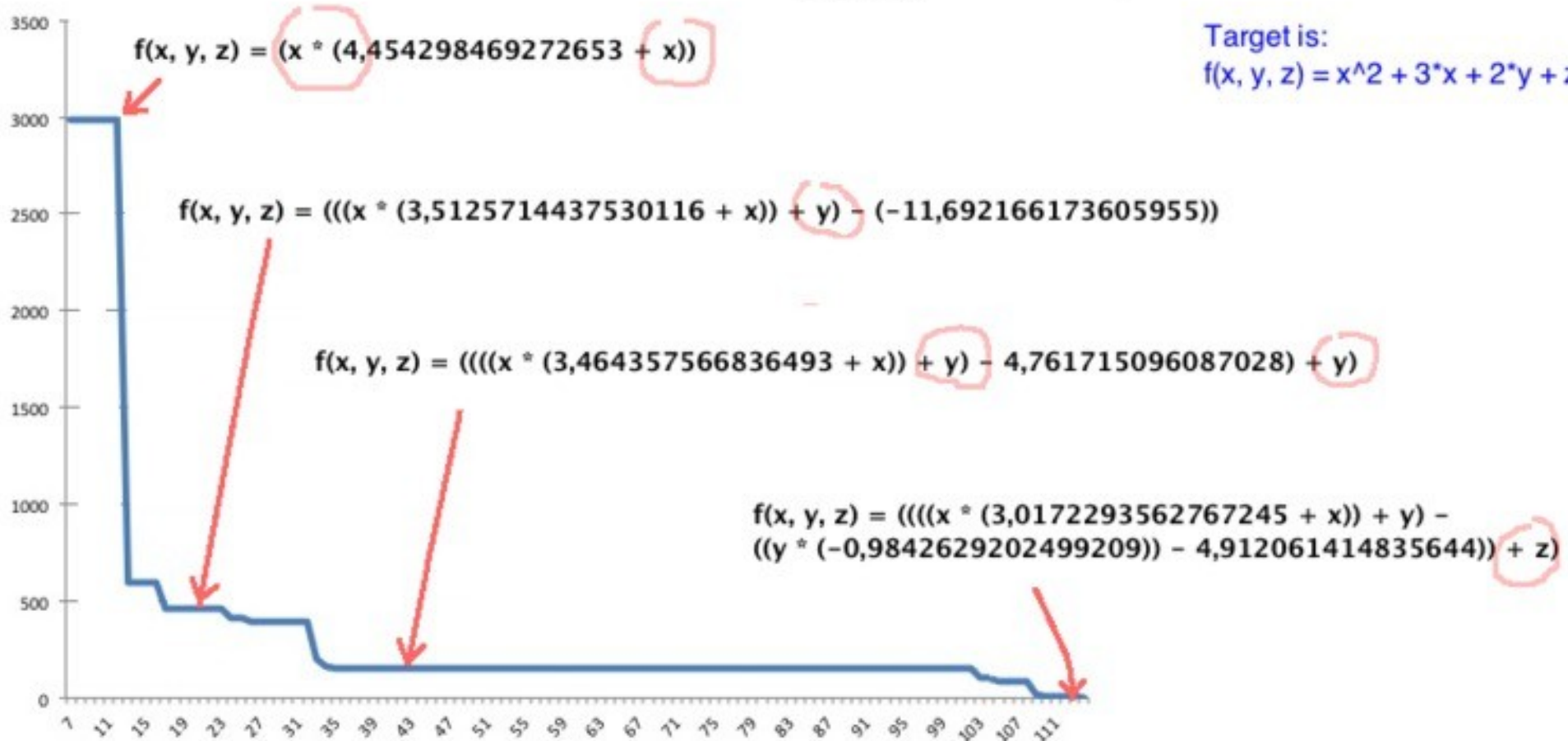
| x | y | z | f(x,y,z) |
|----|----|-----|----------|
| 26 | 35 | 1 | 830 |
| 8 | 24 | -11 | 130 |
| 20 | 1 | 10 | 477 |
| 33 | 11 | 2 | 1217 |
| 37 | 16 | 7 | 1524 |

- megengedett műveletek: ADD MUL SUB

Átlagos négyzetes hiba

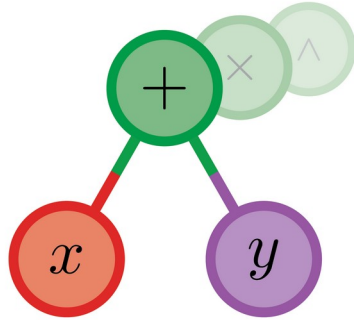
sum(Δ^2)

Target is:
 $f(x, y, z) = x^2 + 3x + 2y + z + 5$



Julia megoldás

SymbolicRegression



SymbolicRegression.jl

```
1 using Pkg
2 Pkg.add("SymbolicRegression")
3 using SymbolicRegression
4 X=[26.0 8 20 33 37;35 24 1 11 16;1 -11 10 2 7];
5 y=[830.0,130,477,1217,1524];
6 options=SymbolicRegression.Options(
7     binary_operators=(+,*,-),
8     npopulations=10
9 );
10 hall_of_fame = EquationSearch(
11     X, y, iterations=20, options=options,
12     parallelism=:multithreading
13 );
14 dominating = calculate_pareto_frontier(X, y, hall_of_fame,
15     options);
15 trees = [member.tree for member in dominating]
```

```
8-element Vector{Node{Float64}}:
```

```
835.60000000003572
```

```
(x1 * x1)
(x1 * (x1 + 4.455530850480829))
(((x1 + 3.878265703156426) * x1) + x2)
(((x1 + 3.458684008703402) * x1) + (x2 * 1.483593597989665))
((((x1 + 3.463089810228236) * x1) + x2) + (x2 * 0.7192117779508957))
(((x1 * (x1 - -2.999999999999992)) + (x3 + (x2 - -5.0000000000000219))) + x2)
((((((x2 - (x1 * -1.2697725658796626)) + (x1 * (x1 - -0.7302274341203455))) + (x3 + 4.9999999999998685)) + x1) + x2)
```

$$y + 1.26977 x + x^2 + 0.73023 x + z + 5 + x + y = x^2 + 3 x + 2 y + z + 5$$

Szakirodalom

- Detrekői (1991) Kiegyenlítő számítások, 11. fejezet
- Madsen K, Nielsen H B, Tingleff O (2004): Methods for Non-linear Least Squares Problems, Informatics and Mathematical Modelling, TU Denmark
- Gulsen M, Smith A E, Tate D M (1995): A genetic algorithm approach to curve fitting, Int. J. Prod. Res. 33(7), 1911-1923
- Worm T, Chiu K (2013): Prioritized Grammar Enumeration: Symbolic Regression by Dynamic Programming, Proc. GECCO '13, 1021-1028, ACM