Harjoitus 3: Regressiomallit (Matlab)

SCI-C0200 Fysiikan ja matematiikan menetelmien studio 2023

Harjoituksen aiheita

- Pienimmän neliösumman menetelmä mallin sovittamisessa
- Lineaarisen regressiomallin sovittaminen ja käyttö
  - Ennustamisessa
  - Mallin parametrien estimoimisessa

Oppimistavoitteet

- Ymmärrät PNS-sovitteen idean
- Osaat muodostaa lineaarisen mallin käyttäen Matlabin valmiita työkaluja
- Opit joitain regressiomallien sovelluskohteita

Regressioanalyysi

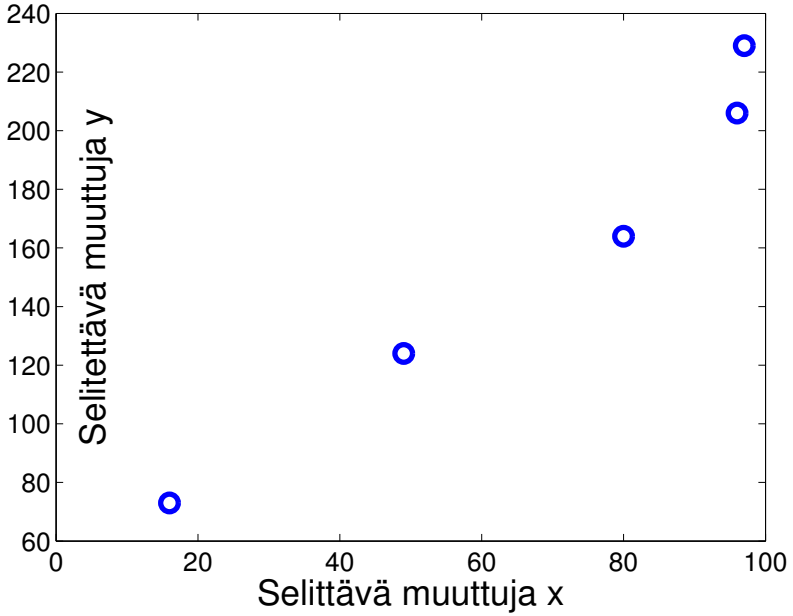
- Peruskysymys: Kuinka selitettävän muuttujan havaittu arvo riippuu selittävien muuttujien arvoista?
- Regressioanalyysissä selittävien ja selitettävän muuttujan arvojen riippuvuudelle rakennetaan regressiomalli, joka sisältää systemaattisen osan ja satunnaisosan.
- Systemaattinen osa kuvaa selittävien ja selitettävän välistä tilastollista riippuvuutta. Esimerkiksi iän $\in [25, 60]$ ja tulotason välillä voisi päteä tilastollinen riippuvuus: ”keskimäärin 5v ikäero tarkoittaa 4000€:n erotusta vuosiansioissa”
- Satunnaisosan avulla huomioidaan se, että systemaattisen osan kuvaama riippuvuussuhde ei aina päde. Esimerkiksi: ”5v ikäero tarkoittaa 4000€$\pm$3000€:n erotusta vuosiansioissa (95%:n luottamusvälillä)”.

Yhden selittäjän lineaarinen regressiomalli

Olkoon annettuna havaintodata (x_j, y_j), $j = 1, ..., n$, jossa

y_j = Selitettävän muuttujan havaittu arvo havaintoyksikössä j

x_j = Selittävän muuttujan havaittu arvo havaintoyksikössä j



Yhden selittäjän lineaarinen regressiomalli

- Regressioanalyysissä etsitään dataan parhaiten sopivaa regressiomallia tiettyjen mallien, kuten lineaaristen mallien, joukosta
- Yhden selittäjän lineaarinen regressiomalli on muotoa

$$y_j = \underbrace{\beta_0 + \beta_1 \cdot x_j}_{\text{Systemaattinen osa}} + \underbrace{\varepsilon_j}_{\text{Satunnaisosa}} \quad , j = 1, 2, \ldots, n$$

  β_0 = tuntematon regressiokerroin 'vakioselittäjälle' 1

  β_1 = tuntematon regressiokerroin selittäjälle x

  ε_j = Satunnainen virhetermi havaintoyksikössä j

- Tyypillinen toteutustapa on pienimmän neliösumman (PNS) menetelmä. Siinä etsitään parameterien β_0 ja β_1 arvot s.e. neliösumma $\sum_{j=1}^{n} \varepsilon_j^2$ minimoituu.

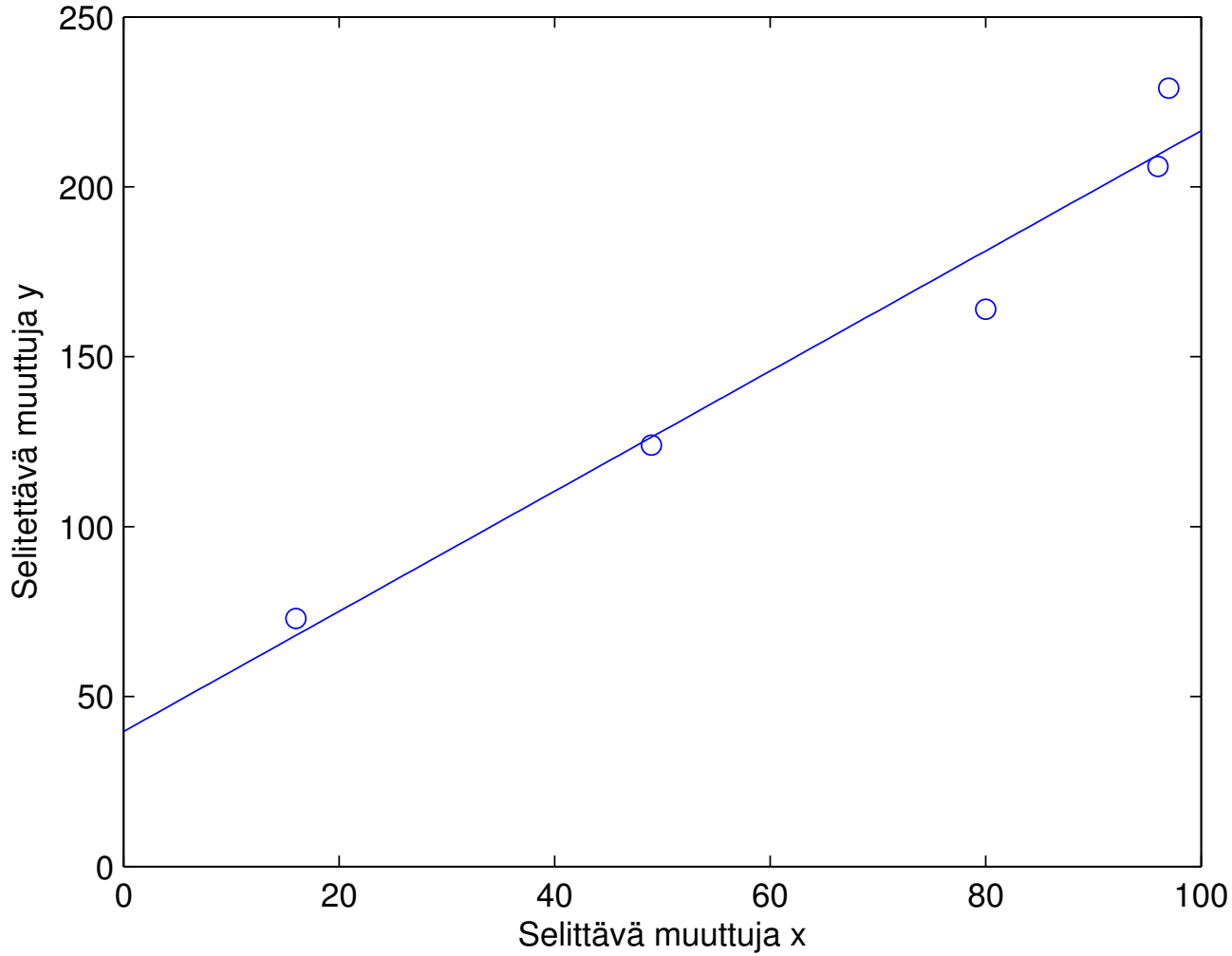
Yhden selittäjän lin. regressiomalli Matlabissa

```
>> x=[16,97,96,49,80]';
>> y=[73,229,206,124,164]';
>> model1=fitlm(x,y,'linear') %luo model1-muuttujaan lineaarisen mallin

Linear regression model:
    y ~ 1 + x1
Estimated Coefficients:
                   Estimate    SE         tStat     pValue
    (Intercept)    39.707       15.791    2.5145     0.086594
    x1             1.7676      0.21223     8.329    0.0036274

>>plot(x,y,'o'), hold on
>>t=[0,100]';
>>plot(t, predict(model1,t)) %predict laskee ennusteet pisteissä t
>>grid on, xlabel('Selittävä muuttuja x'), ylabel('Selitettävä muuttuja y
```

Yhden selittäjän lin. regressiomalli Matlabissa



fitlm-komennon käyttö edellisessä esimerkissä

- `fitlm` komento otti sisäänsä
  - `x`: pystyvektori selittäjän havaituista arvoista (jos useita selittäjiä niin `x` on matriisi)
  - `y`: pystyvektori selitettävän muuttujan havaituista arvoista
  - `'linear'` viittaa siihen, että luodaan lineaarinen malli (johon oletusarvoisesti kuuluu vakiotermi)
- `model1=fitlm(x,y,'linear')` loi model1 nimisen tiedoston, joka sisältää dataan sovitetun lineaarisen mallin
- Tulosteesta voi lukea selittäjille estimoidut regressiokertoimet
- Plottaamisessa käytetään komentoa `predict(model1,t)`, joka laskee mallin ennusteet `t`-vektorin pisteissä

Termejä

- Kun malli on sovitettu dataan, niin tuntemattomille parametereille β_0 ja β_1 saadaan estimaatit. Niitä merkitään symboleilla b_0 ja b_1.
- Mallin ennuste y

$$y = b_0 + b_1 x,$$

on regressiosuoran arvo mielivaltaisessa pisteessä x.

- Sovite

$$\hat{y}_j = b_0 + b_1 x_j,\ j = 1, \ldots, n$$

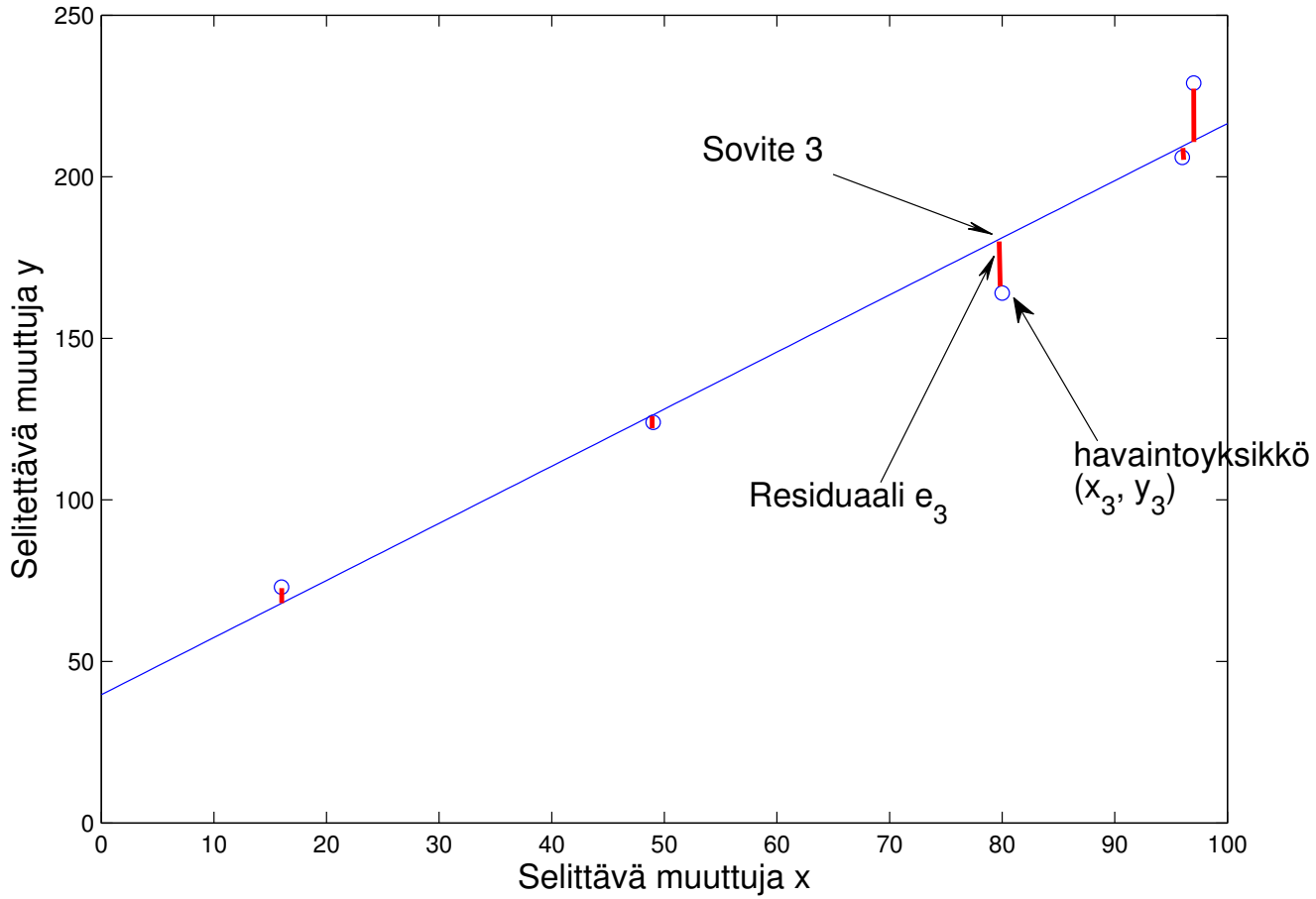
on regressiosuoran arvo havaintopisteessä x_j.

- Residuaali

$$e_j = y_j - \hat{y}_j,\ j = 1, 2, \ldots, n$$

on selitettävän muuttujan havaitun arvon y_j ja sovitteen $\hat{y}_j$ arvon erotus.

Sovitteet ja residuaalit esimerkissä



Lineaarisen regressioanalyysin oletuksista

- Lineaarinen regressiomalli olettaa, että selittävän ja selitettävän muuttujan välillä pätee lineaarinen tilastollinen riippuvuussuhde.
- Virhetermit ε_j tulkitaan satunnaismuuttujina, joista tehdään standardioletukset:

  (i) $\mathrm{E}(\varepsilon_j) = 0,\ j = 1, 2, \ldots, n.$

  (ii) $\mathrm{Var}(\varepsilon_j) = \sigma^2,\ j = 1, 2, \ldots, n.$

  (iii) $\mathrm{Cor}(\varepsilon_j, \varepsilon_l) = 0,\ j \neq l.$

  Tavallisesti myös: (iv) $\varepsilon_j \sim N(0, \sigma^2),\ j = 1, 2, \ldots, n.$
- Ennusteisiin tulee aina suhtautua harkiten. Mallin oletukset eivät välttämättä päde pisteessä tai ajanhetkessä, jolle ennustetta laaditaan.

PNS-menetelmä yleisemmin

- Yleisemmin PNS-menetelmä voidaan nähdä tapana sovittaa funktio $f(x;\beta)$ annettuun pistejoukkoon (x_j, y_j), $j = 1, ..., n$
- Funktion f muoto riippuu parametrin β arvosta. Funktion ei tarvitse olla lineaarinen
- PNS-menetelmässä etsitään parametrille β PNS-estimaatti b siten, että regressiomallin

$$y_j = f(x_j;\beta) + \varepsilon_j \, , \, j = 1, 2, \ldots, n$$

virhetermien ε_j neliöiden summa minimoituu. Kyseessä on siis rajoittamaton (=ei rajoitusehtoja) optimointitehtävä

$$\min_{\beta} \sum_{j=1}^{n} \varepsilon_j^2 \quad = \quad \min_{\beta} \sum_{j=1}^{n} (y_j - f(x_j;\beta))^2,$$

jonka päätösmuuttuja on β.

PNS-menetelmä yleisemmin, Matlab esimerkki

- `lsqcurvefit` sovittaa käyrän dataan käyttäen PNS-menetelmää
- Sovitetaan esimerkiksi annettuun dataan käyrää $y = \beta_0 + x^{\beta_1}$
- Ensin tulee määritellä funktio omaan tiedostoonsa. Tuntematon parametri tulee olla ensimmäisessä argumentissa. Toisessa x-suuntainen data.

```
function F = myfun(b,xdata)
F = b(1)+xdata.^b(2);
```

- Olkoon data seuraavissa vektoreissa:

```
xdata=[0 1 2 3 4 5 6 7 8 9 10];
ydata=[4.9 2.9 10.1 14.0 21.7 29.8 43.8 51.4 72.9 85.1 105.2];
```

- Sitten voidaankin käyttää sovitusalgoritmia:

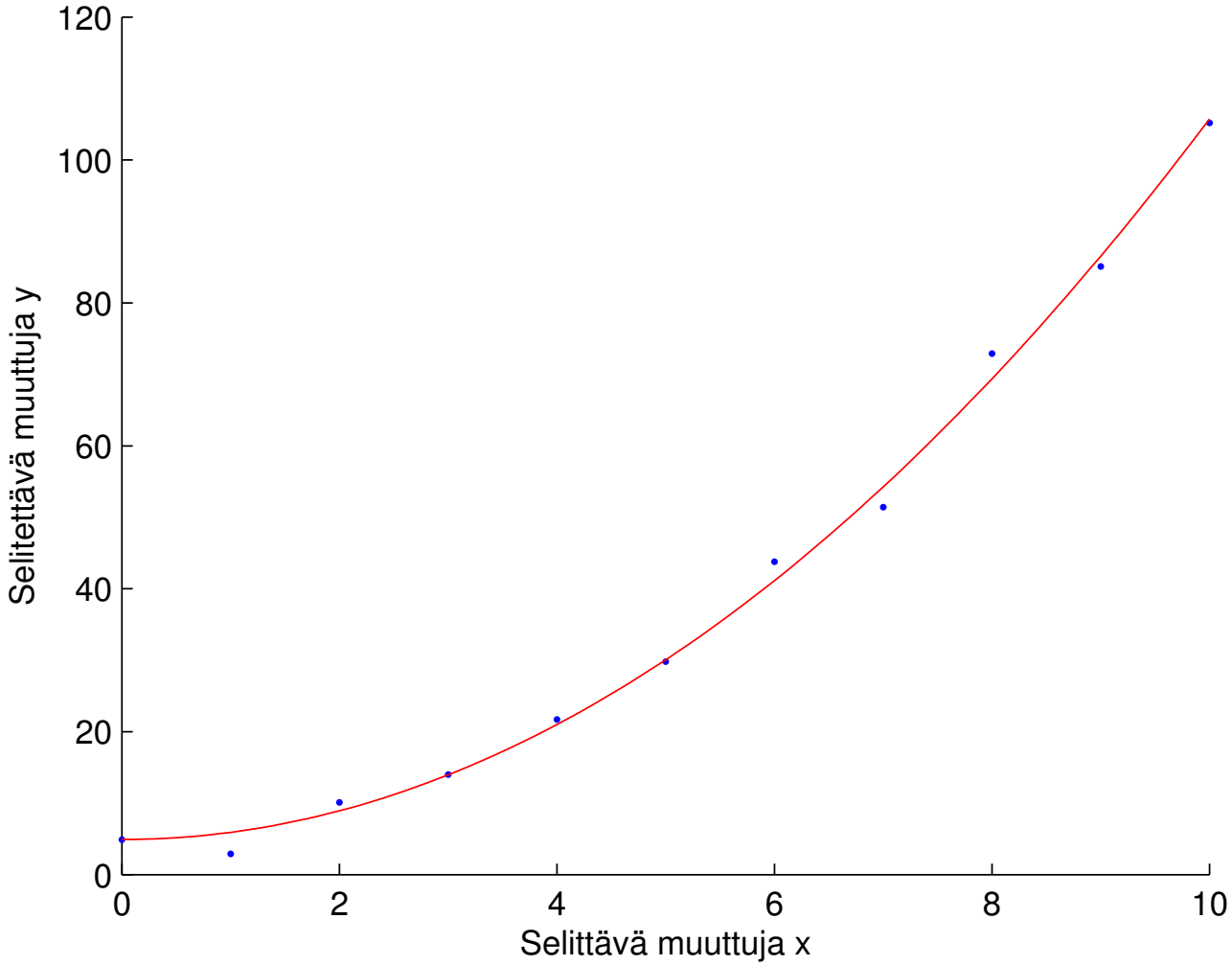
```
b_est=lsqcurvefit(@(b,xdata) myfun(b,xdata),[0;0],xdata,ydata)
```

PNS-menetelmä yleisemmin, Matlab esimerkki

- Sovitettava funktio annetaan ensimmäisessä argumentissa. Sen eteen tulee laittaa `@(b,xdata)`, jotta `lsqcurvefit` toimii. Toisessa argumentissa annetaan alkuarvaus tuntemattomien parametrien arvoille. (Vinkki: Ongelmatilanteessa ks. help lsqcurvefit.)
- Kuva voidaan piirtää vaikkapa näin:

```
figure
hold on
plot(xdata,ydata,'.')
xlabel('Selittävä muuttuja x')
ylabel('Selitettävä muuttuja y')
%piirretään funktio b_est(1)+xdata.^b_est(2);
%käytetään x_akselilla tiheämpää väliä niin tulee siistimpi kuva
x_akseli=[0:0.1:10];
plot(x_akseli, myfun(b_est, x_akseli),'r')
```

PNS-menetelmä yleisemmin, Matlab esimerkki



Tehtävä A: Lineaarinen energiankulutusmalli

- Tehtävänäsi on muodostaa lineaarinen regressiomalli, jossa selittäjänä on kotitalouden varallisuus ja selitettävänä on kotitalouden kuluttama energia
- Tehtävässä käytetään Matlabin `fitlm` ja `predict` komentoja
- Tehtävässä käytettävä data löytyy seuraavalta sivulta ja lisäohjeita sen jälkeiseltä sivulta

Tehtävä A: Lineaarinen energiankulutusmalli

Kotitalouden varallisuus	Käytetty energia
20.0	1.8
30.5	3.0
40.0	4.8
55.1	5.0
60.3	6.5
74.9	7.0
88.4	9.0
95.2	9.1

Tehtävä A: Lineaarinen energiankulutusmalli

1. Tuo tehtävänannon data Matlabiin, tallenna ne pystyvektoreihin `varallisuus` ja `energia`.
2. Plottaa tehtävänannon data, luo lineaarinen malli ja piirrä sen ennusteet välillä $[10, 120]$ samaan kuvaan datan kanssa. Katso esimerkkiä luento-osuuden esimerkistä.

Tehtävä A: Luottamusvälien laskeminen ja visualisointi

- Regressiosuoran mukainen ennuste pätee keskimäärin. Usein on kuitenkin tarpeellista tietää kuinka tarkka ennuste on.

3. Laske ennusteille 64% luottamusvälit (=välit joiden sisään uusi havainto osuisi 64% todennäköisyydellä).

- Ohjeita seuraavalla sivulla.

- Olkoon `x_akseli=[10:1:120]'` ja

  `model=fitlm(varallisuus,energia,'linear')`

- Syntaksilla `[ypred,yci]=predict(model,x_akseli,'alpha',p,`

  `'Prediction','observation')`

  - `ypred`:iin tallentuu ennusteet, jotka vastaavat selittäjän arvoja $\{10, 11, ..., 120\}$.
  - `yci`:n tallentuu kunkin ennusteen luottamusvälin ala- ja ylärajat. Rajat ovat satunnaismuuttujia ja estimoitava parametri sisältyy rajojen välille todennäköisyydellä $(1 - p) \cdot 100\%$.
  - Kohta `...,'alpha',p,...` kertoo predictille luottamusvälin leveyden
  - Kohta `...,'Prediction', 'observation',...` kertoo, että luottamusvälit lasketaan yksittäisille ennusteille, ei ennusteiden odotusarvoille

Tehtävä A: Luottamusvälien laskeminen ja visualisointi

- Piirrä kuva, jossa näkyy
  - punaisina rakseina tehtävänannon data,
  - estimoidun regressiomallin ennusteet välillä $[10, 120]$ sekä
  - katkoviivalla ennusteiden 64% luottamusvälit

  Muista laittaa akseleille selitteet. Nimeä lisäksi kuva ja laita sopiva `legend`.
- Mille välille mallin mukaan varallisuustason 65 energiankulutus osuu 64% luottamusvälin perusteella?

Tehtävä B: Epälineaarisen kasvumallin sovittaminen

- Rukiinviljelija-systeemitietelijä Risto on havainnut, etta ruisjyvien maksimipaino ja hetki, jona tuo paino saavutetaan vaihtelevat vuosittain.
- Risto tuumii, että voisiko hän käyttää kasvukauden dataa ennustamaan hetkeä, jona paino on maksimissaan ja korjuu kannattaa suorittaa.
- Korjuuhetkea odotellessaan voisi hän vaikkapa käydä etelänmatkalla Helsingissä.

Malli ruisjyvien painolle

- Jyvien painon w (selitettävä) kehitystä ajan t (selittäjä) funktiona voi mallintaa yhtälöllä

$$w = f(t; \underbrace{w_{max}, t_{max}, t_v}_{\beta}) = w_{max}(1 + \frac{t_{max} - t}{t_{max} - t_v})(\frac{t}{t_{max}})^{\frac{t_{max}}{t_{max} - t_v}}, \quad (1)$$

  kun $t_{max} \geq t \geq 0$. Kun $t > t_{max}$, niin $f(t; w_{max}, t_{max}, t_v) = w_{max}$.

- Mallin parametrien tulkinnat ovat seuraavat:
  - w_{max} on maksimipaino, jonka jyvä saavuttaa,
  - t_{max} on hetki jona maksimipaino saavutetaan ja
  - t_v on hetki, jona jyvän painon kasvunopeus on suurimmillaan.

Malli ruisjyvien painolle

- Tehtävänä on siis selvittää parametrit w_{max}, t_{max} ja t_v, joilla malli sopii dataan mahdollisimman hyvin.
- Käytä PNS-menetelmää hyödyntävää `lsqcurvefit`-komentoa

Mittaus j	Jyvän paino w_j (0.1 g)	Päiviä 1. kukinnosta t_j
1	2	2
2	8	10
3	19	18
4	32	26
5	43	31
6	45	35

Ohjeita

1. Muodosta ensin funktio: `function w = kasvumalli(beta,t)`
   - Funktion argumentti `beta` sisältää kasvumallin parametrit $\beta = [w_{max}, t_{max}, t_v]$. Argumentti `t` on vektori ajanhetkistä.
   - Funktio tuottaa vektorin `w`, jossa on kasvumallilla lasketut jyvän painot, jotka vastaavat funktiolle annettuja ajanhetkiä.
   - *Vinkki:* Toteuta funktio ”kaksivaiheisesti”:

     Käytä ensin kaavaa (1) (ole huolellinen sulkujen kanssa). Huom: Käytä alkioittaisia laskutoimituksia! Voit kertoa vektoreita alkioittain käyttäen pistettä, esim: `[2 2].*[1 2]=[2 4]` ja `[2 2].^2=[4 4]`.

     Sitten huomioi, että kun w saavuttaa maksimiarvonsa, niin sen tulee pysyä siinä. Voit esim. asettaa funktiosi loppuun `w(i:end)=maxw`, jossa `i` on indeksi, josta löytyy `w`:n maksimi ja `maxw` on tuo maksimi. Tämän indeksin ja maksimin löydät esim. `max` -funktiolla.

Ohjeita

2. Käytä Matlabin `lsqcurvefit`-työkalua löytääksesi β, jolla malli sopii dataan parhaiten. Kts. `help lsqcurvefit`-helpin 3. kappaleen teksti ja 1. esimerkki. Käytä alkuarvausta $\beta = [40, 40, 20]$.

Piirrä kuva, jossa näkyy

- tehtävänannon data ja mallin ennusteet aikavälillä $[0, 50]$, eli arvot $w(t; w_{max}, t_{max}, t_v)$, kun $t \in [0, 1, 2, ..., 50]$
- lisää kuvaan ruudukko (`grid`)
- ota kuvaajasta ylimääräinen y-suuntainen tyhjä tila pois (`ylim`)
- nimeä käppyrät (`legend`), nimeä akselit sopivasti ja otsikoi kuva.

Ohjeita

- ✍ Kommentoi kuvaajaa: Miltä se näyttää?
- ✍ Tänään on 35. päivä sitten ensimmäisen kukinnon. Kuinka monta päivää malli ennustaa kuluvan siihen, että jyvät saavuttavat maksimipainonsa?

Kotitehtävä: Fysikaalisen systeemin parametrien estimointi

- Eräs lineaaristen mallien sovellus on fysikaalisten systeemien parametrien estimointi
- Tavallisesti lineaarinen regressiomalli olettaa, että jokaiseen havaintopisteeseen liittyvä epävarmuus on yhtä suurta
- Tässä tehtävässä sovitat lineaarisen mallin fysikaaliseen mittausdataan, jossa yksittäisten pisteiden tarkkuus vaihtelee

Kotitehtävä: Taustaa - Planckin vakion mittaaminen

- Haluat mitata Planckin vakion. Mittausjärjestelyssäsi irroitat katodilta valon avulla elektroneja, jotka kulkeutuvat anodille kunnes anodin ja katodin välinen pysätysjännite kasvaa niin suureksi, että se estää elektronien kulkeutumisen anodille. Katodin potentiaali pysyy maadoituksen vuoksi vakiona.
- Katodilta irtoavien elektroneiden energia riippuu valon taajuudesta. Fysiikan teorian pohjalta olet johtanut seuraavan yhtälön:

$$V_0 = -\frac{h}{e} f + \mathrm{C}, \tag{2}$$

jossa V_0 on pysäytysjännite, f on valon taajuus, h on Planckin vakio, e on elektronin varaus ja C on vakiotermi.

Kotitehtävä: Mittausdata

- Vaihtelemalla valon taajuutta saat kerättyä seuraavan mittausdatan pysäytysjännitteistä

Mittaus i	Valon taajuus f (10^{12} Hz)	Pysäytys jännite V_0 (V)
1	519	1.0 ± 0.15
2	549	1.2 ± 0.40
3	688	1.9 ± 0.20
4	740	2.4 ± 0.30
5	821	2.3 ± 0.10

- Arvioit, että valontaajuusdata on virheetöntä. Pysäytysjännitteille olet arvioinut virherajat.

Kotitehtävä: Lineaarisen mallin sovittaminen painotetun neliösumman avulla

- Haluat sovittaa mittausdataan suoran, jonka kulmakertoimesta voit päätellä Planckin vakion.
- Koska eri havaintopisteiden tarkkuus vaihtelee, käytä menetelmää, jossa parametrit valitaan siten, että painotettu virheneliösumma minimoituu.
- Etsit siis vakiotermiä b_0 ja kulmakerrointa b_1 s.e. seuraava painotettu virheneliösumma minimoituu

$$\sum_{i=1}^{n} \left(\frac{y_i - (b_1 x_i + b_0)}{\Delta y_i} \right)^2,$$

jossa Δy_i on havainnon i virhetermi ja n on havaintopisteiden lukumäärä. Esim. $x_1 = 519, y_1 = 1.0$ ja $\Delta y_1 = 0.15$.

KT: Analyyttinen ratkaisu estimaateille

- Painotetun neliösumman minimoiville parametrien arvoille saadaan seuraavat analyyttiset lausekkeet:

$$b_1 = \frac{1}{D}[(\sum_{i=1}^{n} \frac{1}{(\Delta y_i)^2})(\sum_{i=1}^{n} \frac{x_i y_i}{(\Delta y_i)^2}) - (\sum_{i=1}^{n} \frac{x_i}{(\Delta y_i)^2})(\sum_{i=1}^{n} \frac{y_i}{(\Delta y_i)^2})] \quad (3)$$

ja

$$b_0 = \frac{1}{D}[(\sum_{i=1}^{n} \frac{x_i^2}{(\Delta y_i)^2})(\sum_{i=1}^{n} \frac{y_i}{(\Delta y_i)^2}) - (\sum_{i=1}^{n} \frac{x_i}{(\Delta y_i)^2})(\sum_{i=1}^{n} \frac{x_i y_i}{(\Delta y_i)^2})], \quad (4)$$

missä

$$D = (\sum_{i=1}^{n} \frac{1}{(\Delta y_i)^2})(\sum_{i=1}^{n} \frac{x_i^2}{(\Delta y_i)^2}) - (\sum_{i=1}^{n} \frac{x_i}{(\Delta y_i)^2})^2$$

KT: Analyyttinen ratkaisu estimaateille

- Lisäksi parametrien estimaateille voidaan laskea (yhden standardipoikkeaman levyiset) virheet

$$\Delta b_1 = \sqrt{\frac{1}{D}\sum_{i=1}^{n}\frac{1}{(\Delta y_i)^2}} \tag{5}$$

ja

$$\Delta b_0 = \sqrt{\frac{1}{D}\sum_{i=1}^{n}\frac{x_i^2}{(\Delta y_i)^2}}. \tag{6}$$

- Näiden avulla parametrien estimaateille saa alarajat $b_i - \Delta b_i$ ja ylärajat $b_i + \Delta b_i$. Todellinen parametrin arvo on tuon välin sisällä 64% todennäköisyydellä.

Kotitehtävä: Ohjeistus

Sovita mittausdataan suora käyttäen edellä kuvattua menetelmää

1. Luo `function [b, bci] = sovittaja(x,y,deltay)`, jonka argumentteja ovat

   - taajuudet $x = [x_1, ..., x_5]$, mitatut pysäytysjännitteet $y = [y_1, ..., y_5]$ ja pysäytysjännitteiden virhetermit $\Delta y = [\Delta y_1, ..., \Delta y_5]$.

   Funktio palauttaa vektorin $b = [b_0, b_1]$, joka sisältää vakiotermin ja kulmakertoimen estimaatit (kaavat 3 ja 4),

   ja matriisin $b_{ci} = \begin{bmatrix} b_0 - \Delta b_0 & b_1 - \Delta b_1 \\ b_0 + \Delta b_0 & b_1 + \Delta b_1 \end{bmatrix}$, joka sisältää estimaattien ala- ja ylärajan (kts. edellinen sivu).

Kotitehtävä: Ohjeistus

Vinkki: Käytä lausekkeiden 3 - 6 summien laskemiseen matlabin alkioittaisia vektorioperaatioita. Välituloksien laskeminen auttaa vähentämään virheitä.

Huom: Summalausekkeet toistuvat. Niinpä riittää, että ne laskee erikseen kertaalleen ja kertoo sitten näitä tuloksia keskenään sopivilla tavoilla.

Esim. $\sum a_i b_i / c_i$ saadaan syntaksilla `sum((a.*b)./c)`, jossa `a,b,c` ovat vektoreita.

2. Estimoi b_0 ja b_1 käyttäen omaa funktiotasi.

3. Piirrä kuva datasta ja siihen estimoidusta regressiosuorasta $y = b_0 + b_1 x$. Piirrä dataan myös yksittäisten pisteiden virherajat. Tähän voit käyttää komentoa `errorbar`.

4. Kokeile vielä `fitlm`-funktiota sovittaaksesi dataan suoran tavallisella PNS-menetelmällä.

Kotitehtävä: Ohjeistus

- Liitä kuva, jossa näkyy `sovittaja` ja `fitlm` funktioilla sovitetut suorat. Nimeä akselit, anna otsikko ja lisää legend. Kommentoi suorien eroja.
- Mikä on `sovittaja`-funktion antama 64% luottamusväli Planckin vakiolle? (Vinkki: kirjallisuudesta löytyvän Planckin vakion tulisi kuulua tälle välille.) Huom! Sovitteen kulmakertoimen b_1 fysikaalinen tulkinta on $b_1 = -h/e$, jossa e on elektronin varaus $-1.6 \cdot 10^{-19}$ C. Huomaa myös, että tehtävänannon datassa taajuuden kertaluokka on 10^{12}.
- Liitä Matlab-koodi funktiostasi `sovittaja`.