1 Johdanto

1.1 Regressiomallit

Lineaarinen regression on yksi vanhimmista ja käytetyimmistä tilastollisista tekniikoista. Jos havainnot (t_i, y_i), $i = 1, \ldots, n$ on annettu, niin perusajatus on, että aineistoon sovitetaan malli, joka on muotoa

$$y = \beta_0 + \beta_1 t + \epsilon.$$

Mallinnuksella voidaan yleensä katsoa olevan kaksi päätarkoitusta. Ensinnäkin mallia voidaan käyttää, kun halutaan tutkia ja kuvata vastemuuttujan ja selitettävien muuttujien yhteyksiä. Toinen tärkeä käyttöalue on ennustaminen.

Yleisemmin voisimme ajatella, että vastemuuttujan ja selittäjän yhteyden ei tarvitse olla lineaarinen, jolloin

$$y = g(t) + \epsilon,$$

missä g voi olla jokin sopiva, tyypillisesti riittävän tasainen, funktio. Yksi mahdollisuus on käyttää ns. polynomiregressiota, jolloin

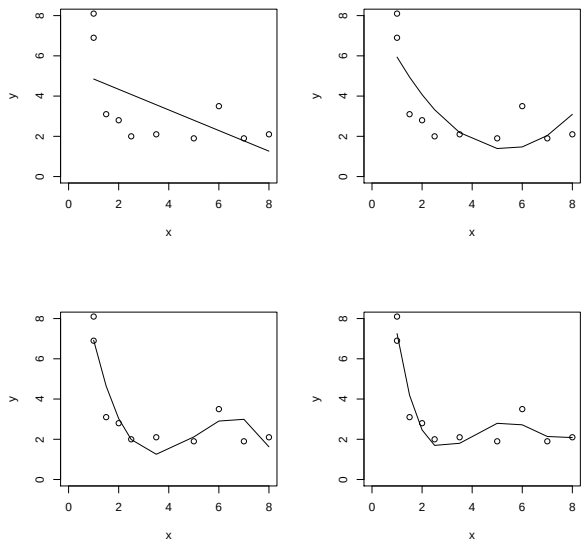
$$g(t) = \beta_0 + \beta_1 t + ... + \beta_p t^p.$$

Polynomien ongelmana voi kuitenkin olla esimerkiksi huono lokaali yhteensopivuus tai liiallinen herkkyys poikkeaville havaintoarvoille. Ennustettaessa ongelmaksi voi muodostua polynomin epämääräinen käyttäytyminen varsinaisen havaintovälin ulkopuolella.

Esimerkki 1. Työtilaan on tehty yksinkertainen havaintoaineisto, joka sisältää muuttujat y ja x. Aineistoon on sitten sovitettu polynomi asteilla 1, 2, 3 ja 4.

```
     y   x
1  8.1 1.0
2  6.9 1.0
3  3.1 1.5
4  2.8 2.0
5  2.0 2.5
6  2.1 3.5
7  1.9 5.0
8  3.5 6.0
9  1.9 7.0
10 2.1 8.0
> par(mfrow=c(2,2))
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> l<-lm(y~x); lines(x, predict(l))
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> l<-lm(y~x+I(x^2)); lines(x, predict(l))
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> l<-lm(y~x+I(x^2)+I(x^3)); lines(x, predict(l))
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> l<-lm(y~x+I(x^2)+I(x^3)+I(x^4)); lines(x, predict(l))
> par(mfrow=c(1,1))
```

Polynomimallit ovat yksi esimerkki parametriensa suhteen lineaarisista malleista. Tällainen malli on esitettävissä muodossa $g = \sum_{j=0}^{p} \beta_j f_j(t)$, missä funktiot f_j muodostavat lineaarisen kannan funktiolle g. Polynomin tapauksessa kantafunktiot ovat $\{1, t, t^2, \ldots, t^p\}$. Esimerkiksi, jos polynomin aste on kaksi, niin



Kuva 1: Esimerkkiaineistoon sovitetut polynonomit asteilla 1-4.

saadaan $\{1, t, t^2\}$. Tätä vastaava matriisiesitys on

$$\mathbf{X} = \begin{pmatrix} 1 & t_1 & t_1^2 \\ \vdots & \vdots & \vdots \\ 1 & t_n & t_n^2 \end{pmatrix}$$

ja malliksi saadaan

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{LM}$$

missä $\mathbf{y} = (y_1, \ldots, y_n)'$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)'$ ja $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)$. Parametrien estimointiin voidaan käyttää ns. pienimmän neliösumman menetelmää (PNS). PNS-estimaatit saadaan minimoimalla funktio

$$SS = \sum_i \{y_i - \sum_j \beta_j f_f(t)\}^2$$

parametrien $\boldsymbol{\beta}$ suhteen. Helposti nähdään, että ratkaisu toteuttaa normaaliyhtälöt

$$(\mathbf{X}'\mathbf{X})\boldsymbol{\beta} = \mathbf{X}'\mathbf{y}. \tag{NY}$$

Jos $\mathbf{X}$ on täysiasteinen, niin

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}. \tag{PNS}$$

Jos oletetaan, että satunnaisvirheet ovat riippumattomia ja että niillä on sama varianssi, niin $\hat{\boldsymbol{\beta}}$ on parametrien $\boldsymbol{\beta}$ paras lineaarinen ja harhaton estimaattori BLUE. Helposti nähdään, että

$$Var(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}.$$

Lisäksi saadaan

$$\hat{g}(t) = \mathbf{x}'(t)\hat{\boldsymbol{\beta}}$$

ja koko käyrän sovite on

$$\hat{\mathbf{g}} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{H}\mathbf{y},$$

missä $\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' = \mathbf{H}$ on ns. "hattumatriisi". Nyt jos lasketaan

$$tr(\mathbf{H}) = tr(\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}') = tr((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X}) = tr(\mathbf{I}_p) = p,$$

niin saadaan kyseisen mallin estimoitujen parametrien lukumäärä. Jatkossa muotoa $tr(\mathbf{H})$ olevaa lauseketta tullaan käyttämään eräänlaisena mallin rakenteen monimutkaisuuden mittana.

Lineaariset mallit muodostavat sekä teoreettisesti selkeän että helposti hallittavan kokonaisuuden. Joissakin tilanteissa perusmalli on kuitenkin selvästi epälineaarinen eikä se välttämättä myöskään ole muunnoksella linearisoitavissa. Seuraavassa eräitä esimerkkejä parametriensa suhteen epälineaarisista malleista

$$g(t) = \beta_0 t^{\beta_1},$$

$$g(t) = \beta_0 + \beta_1 exp(-\beta_2 t^{\beta_3})$$

ja

$$g(t) = \beta_0 cos(\beta_1 t) + \beta_2 sin(\beta_3 t).$$

Myös parametriensa suhteen epälineaariset mallit voidaan estimoida pienimmän neliösumman menetelmällä minimoimalla

$$SS = \sum_i \{y_i - g(\mathbf{z}_i; \boldsymbol{\beta})\}^2,$$

missä $\mathbf{z}_i$ selittäjien vektori ja $g(\mathbf{z}_i; \boldsymbol{\beta})$ on funktio, jota estimoidaan. Yleisessä tapauksessa minimointiongelmalla ei ole suljetun muodon ratkaisua. Jos g on derivoituva parametriensa β_i suhteen, niin ratkaisu toteuttaa yhtälön

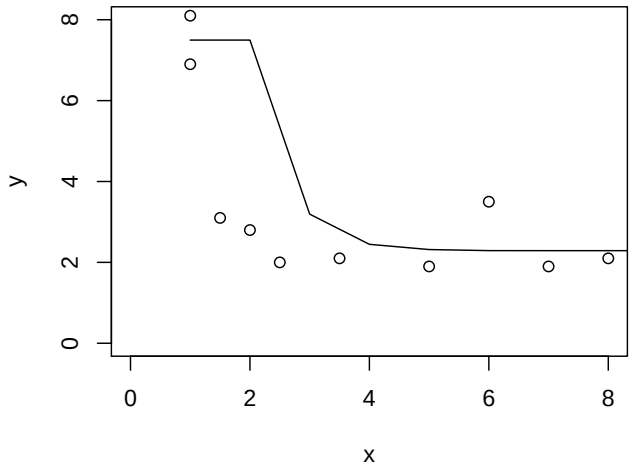
$$\mathbf{X}'\mathbf{g} = \mathbf{X}'\mathbf{y},$$

missä matriisin $\mathbf{X}$ elementit x_{ij} muodostuvat yhtälöstä

$$x_{ij} = \frac{\partial}{\partial \beta_j} g(\mathbf{z}_i; \hat{\beta}).$$

Estimoinnissa käytetään iteratiivisia algortimeja kuten Gauss-Newtonin tai Levenberg-Marquandin algoritmia. Saatu ratkaisu ei kuitenkaan välttämättä ole yksikäsitteinen. Estimoitujen parametrien kovarianssimatriisille ei nyt saada tarkkaa lauseketta, mutta kun n on tarpeeksi iso, niin likimain

$$Var(\hat{\beta}) \approx \sigma^2(\mathbf{X}'\mathbf{X})^{-1}.$$



Kuva 2: Aineistoon sovitetu malli $g(t) = \beta_0 + \beta_1 e^{(-t/\beta_2)}$.

Esimerkki 2. Esimerkkin 1 aineistoon sovitetaan malli $g(t) = \beta_0 + \beta_1 e^{(-t/\beta_2)}$.

```
> library(stats)
> b<-c(b0=2.3,b1=100,b2=0.3)
> l<-nls(y~b0+b1*exp(-x/b2), data=datam, start=b); summary(l)
...
   Estimate Std. Error t value Pr(>|t|)
b0   2.2899     0.2535   9.032 4.17e-05 ***
b1 173.0095   252.5578   0.685   0.5154
b2   0.2854     0.1196   2.387   0.0484 *
...
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> lines(predict(l))
```

1.2 Ei-parametriset mallit

Eräs vaihtoehto parametrisille malleille on erilaiset ei-parametriset tekniikat. Tyyppillisesti näissä tekniikoissa sovitteen $\hat{g}$ muoto on hyvin joustava eikä siis varsinaista oletusta käyrän parametrisesta muodosta tehdä. Tyypillinen oletus, joka tehdään voi olla esimerkiksi, että funktio kuuluu johonkin funktioavaruuteen. Esimerkiksi voidaan olettaa, että käyrä on derivoituva tai että käyrä on derivoituva ja sen toisen derivaatan neliö on integroituva jne.

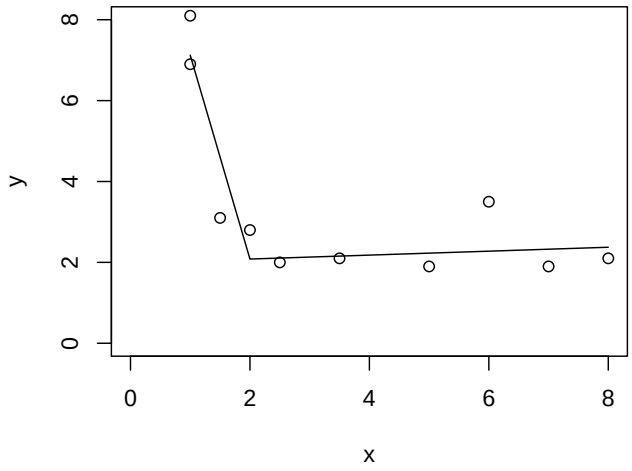
Tärkeä ero parametrisen ja ei-parametrisen menetelmän välillä on siinä kuinka paljon luotetaan siihen informaatioon jota on saatavilla käyrästä g. Tyypillisesti, kun käytetään ei-parametrista regressiota, valitaan sopiva funktio-avaruus, johon g:n ajatellaan kuuluvan. Havaitun aineiston perusteella valitaan sitten kyseisestä avaruudesta se käyrä joka parhaiten kuvaa estimoitavaa regressiokäyrää g. Parametrisessa analyysissä puolestaan valitaan jokin hypoteettinen käyräperhe, mihin tilastollinen inferenssi perustetaan. Analyysin tulos riippuu nyt vahvasti oletetusta parametrisesta mallista. Ei-parametrisessa lähestymistavassa analyysin tulos on selkeämmin seurausta havaitun aineiston rakenteesta, koska mitään parametrista oletusta käyrän muodosta ei tehdä.

Yleensä kuitenkin, jos mallin parametrinen muoto tunnetaan, on parametrisilla malleilla suoritettu analyysi tilastollisilta ominaisuuksiltaan ei-parametrista analyysiä tehokkaampi. Käytännössä mallin parametrista muotoa ei yleensä tunneta, vaan käytetään erilaisia ad hoc-oletuksia. Jos parametrinen oletus on väärä, muuttuukin tilanne usein ei-parametrisen tekniikan eduksi.

Seuraavassa tarkastellaan esimerkkejä ns. lineaarisista tasoitusmenetelmistä, jotka ovat osa ei-parametrisista lähestymistavoista. Näissä menetelmissä käyrän sovite saadaan havaintojen lineaarisena funktiona

$$\hat{\mathbf{g}} = \mathbf{L}\mathbf{y},$$

missä $\mathbf{L}$ on ns. tasoitusmatriisi, joka on mittapisteiden $t_1, \ldots, t_n$ sekä mahdollisen ns. tasoituskertoimen λ funktio.



Kuva 3: Paloittain lineaarinen käyrä solmukohtana $\kappa_1 = 2$.

1.2.1 Splini-käyrät

Yksinkertaisin ns. Splini-käyrä on paloittain lineaarinen malli. Tarkastellaan seuraavaksi paloittain lineaarisen käyrän sovittamista esimerkin avulla.

Esimerkki 3. Paloittain lineaarinen malli. Mikäli valitaan vain yksi ns. solmukohta κ_1 (ks. luku 3.2), niin paloittain lineaarinen käyrä voidaan lausua muodossa

$$g(t) = \beta_0 + \beta_1 t + u_1(t - \kappa_1)_+,$$

missä $(a)_+ = a$, jos $a > 0$ muutoin $a = 0$. Estimointi voidaan nyt suorittaa tavallisella pienimmän neliösumman menetelmällä. Esimerkkiaineistossa $\hat{g}$ selvästi kääntyy pisteessä $\kappa_1 = 2$, missä funktion kulmakerroin muuttuu u_1 yksikköä. Tässä siis kun $t \leq 2$, niin $\hat{\beta}_1 = -5.0420$ ja kun $t > 2$, niin $\hat{\beta}_1 = -5.0420 + 5.0907 = 0.0487$.

```
> knots<-c(0,2)
> rhs<-function(x,c) ifelse (x>c,x-c,0) # tehdään +-funktio
> # muodostetaan tarvittava selittäjien matriisi ulkotulolla
> dm<-outer(x, knots, rhs)
> dm
      [,1] [,2]
 [1,]  1.0  0.0
 [2,]  1.0  0.0
 [3,]  1.5  0.0
 [4,]  2.0  0.0
 [5,]  2.5  0.5
 [6,]  3.5  1.5
 [7,]  5.0  3.0
 [8,]  6.0  4.0
 [9,]  7.0  5.0
[10,]  8.0  6.0
>
> summary(g<-lm(y~dm))
...
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  12.1662     1.3717   8.869 4.69e-05 ***
dm1          -5.0420     0.8530  -5.911 0.000593 ***
dm2           5.0907     0.9498   5.360 0.001053 **

...
> lines(x, predict(g))
```

Esimerkki 4. Tasoitettu kuutiosplini.

Jos käytetään tavallista pienimmän neliösumman menetelmää käyrän sovitukseen eikä aseteta käyrän muodolle erikseen mitään rajoitetta, niin mikä tahansa käyrä, mikä kulkee havaintopisteiden kautta minimoi kriteerin SS. Hyvän lokaalin yhteensopivuuden sijaan usein ollaan kuitenkin kiinnostuneita siitä onko havaitussa aineistossa esimerkiksi jotakin yleisempää trendiä. Yleisellä käyrän sovituksella on siis kaksi hieman ristiriitaista tavoitetta. Yhtäältä hyvä lokaali yhteensopivuus ja toisaalta saada sellainen sovite, joka ei reagoisi liikaa pelkkään satunnaisvaihteluun.

On tietenkin monia mahdollisuuksia määritellä käyrän g tasaisuus havaintovälillä $[a, b]$. Eräs tapa on laskea kaksi kertaa derivoituvan käyrän integroitu toisen derivaatan neliö

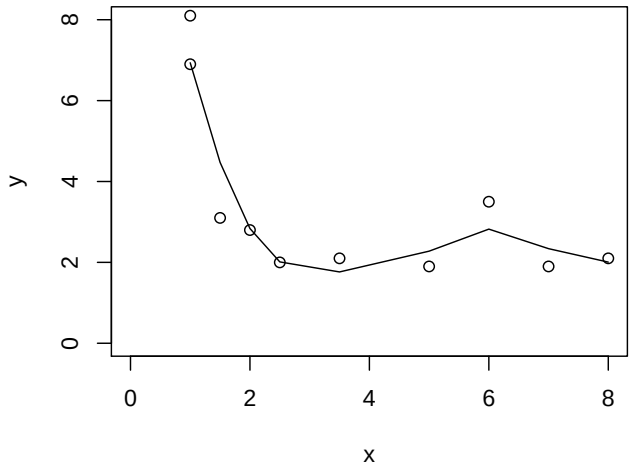
$$RP = \int_a^b g''(t)^2 dt.$$

Eräiden hyvien laskennallisten ominaisuuksien lisäksi menetelmää voidaan regressiomallien tapauksessa perustella esimerkiksi seuraavasti: Käyrän tasaisuus ei saa riippua vakiotermistä eikä kulmakertoimesta. Nyt siis ne käyrät jotka eroavat vain vakiotermin ja kulmakertoimen suhteen ovat termin RP mielessä identtisiä. Käyrän tasaisuus riippuu tässä funktion toisesta derivaatasta.

Tarkastellaan minimointiongelmaa

$$PLS = \sum_i [y_i - g(t)]^2 + \lambda \int [f''(t)]^2 dt = SS + \lambda \times RP,$$

missä λ on annettu positiivinen vakio. Tässä siis kriteeriin SS on lisätty käyrän tasaisuutta mittava termi RP. Tasoituskertoimella λ säädellään sitä kuinka paljon termejä SS ja RP painotetaan. Lopullinen estimaatti on siten eräänlainen yhteensopivuuden ja tasaisuuden kompromissi. Helposti huomataan, että jos λ on kovin iso, niin termi RP saa suuren painon ja sovite on hyvin tasainen. Ääritapauksessa, kun $\lambda \to \infty$ saadaan regressiosuora. Jos taas λ on kovin pieni, niin pääpaino on termissä SS, jolloin soviteena saadaan sellainen käyrä, joka kulkee mahdollisimman tarkoin havaintopisteiden kautta.



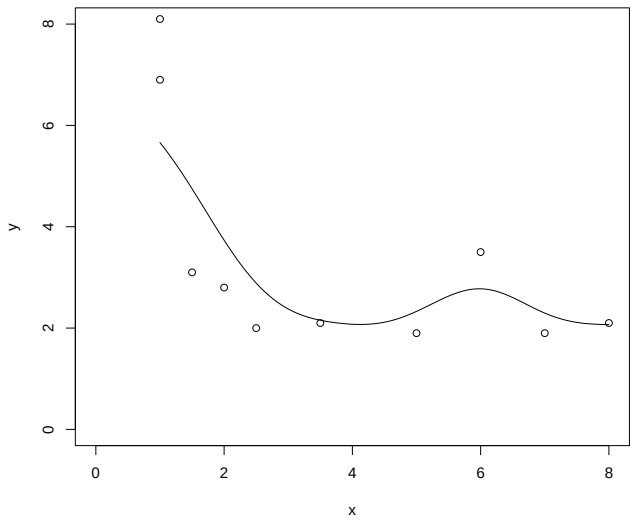
Kuva 4: Esimerkkiaineistoon sovitettu kuutiosplini.

Voidaan osoittaa, että minimointiongelman PLS ratkaisu on ns. tasoitettu kuutiosplini. Ratkaisu on esitettävissä muodossa

$$\hat{\mathbf{g}} = \mathbf{S}_\lambda \mathbf{y},$$

missä $\mathbf{S}$ on lausuttavissa havaintopisteiden $t_1, \ldots, t_n$ sekä tasoitusparametrin λ avulla. Esimerkkiaineiston tapauksessa saadaan, kun valitaan tasoituskerroin $\lambda = 0.000396$ (ks. luku 5).

```
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> lines(smooth.spline(x,y))
> smooth.spline(x,y)
...
Smoothing Parameter  spar= 0.4013015  lambda= 0.0003960459 (10 iterations)
Equivalent Degrees of Freedom (Df): 5.385384
...
```



Kuva 5: Kernel-menetelmällä tasoitetut havaintoarvot.

Esimerkki 4. Lokaalit polynomit.

Lokaaleista polynomimalleista johdettu ns. Nadaray-Watson-estimaattori voidaan esittää seuraavasti

$$\hat{g} = \sum_j y_j K(\frac{t_j - t_i}{b}) / K(\frac{t_j - t_i}{b}),$$

missä b on ns. ikkunan leveys ja K on Kernel-funktio, jolla havaintoja painotetaan (ks. luku 4.2). Seuraavassa esimerkissä käytetään normaalijakaumaan perustuvaa Kernel-funktiota.

```
> plot(y~x,ylim=c(0,8), xlim=c(0,8))
> lines(ksmooth(x,y, "normal", bandwidth=2))
```

2 Lineaariset sekamallit

Mallin yleinen muoto perustuu lineaariseen malliin

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\epsilon},$$

missä $\mathbf{X}$ on $n \times p$ kiinteän osan suunnittelumatriisi, $\mathbf{Z}$ on $n \times c$ satunnaisvaikutusten suunnitelumatriisi ja $\mathbf{u}$ on $c \times 1$ satunnaisvaikutusten vektori. Sekamallissa määritellään

$$E(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta} \qquad \text{ja} \qquad E(\mathbf{y} \mid \mathbf{u}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u}$$

sekä

$$\boldsymbol{\epsilon} = \mathbf{y} - E(\mathbf{y} \mid \mathbf{u}),$$

missä

$$E(\mathbf{u}) = \mathbf{0} \qquad \text{ja} \qquad E(\boldsymbol{\epsilon}) = \mathbf{0}.$$

Sekamallissa oletetaan lisäksi, että *satunnaisvirheet ja satunnaisvaikutukset ovat riippumattomia,* jolloin

$$Cov(\mathbf{u}, \boldsymbol{\epsilon}) = \mathbf{O}.$$

Sekamallin yleisessä muodossa oletetaan, että $Var(\mathbf{u}) = \mathbf{D}$ ja $Var(\boldsymbol{\epsilon}) = \mathbf{R}$, missä $\mathbf{D}$ ja $\mathbf{R}$ ovat kovarianssimatriiseja. Havaintojen $\mathbf{y}$ kovarianssimatriisi on nyt muotoa

$$Var(\mathbf{y}) = \mathbf{Z}\mathbf{D}\mathbf{Z}' + \mathbf{R}.$$

Mallin ns. marginaalinen esitysmuoto on nyt

$$E(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta} \qquad \text{ja} \qquad Var(\mathbf{y}) = \mathbf{Z}\mathbf{D}\mathbf{Z}' + \mathbf{R}.$$

2.1 Parametrien estimointi

2.1.1 Kiinteä osa

Pienimmän neliösumman (PNS) estimaattori kiinteälle osalle on

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}.$$

PNS-estimaattori ei kuitenkaan ota huomioon havaintojen korreloituneisuutta, joten se ei ole kovin mielenkiintoinen käytännön analyyseissa. Paras lineaarinen ja harhaton estimaattori kiinteälle osalle on

$$\tilde{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{y}$$

ja

$$Var(\tilde{\boldsymbol{\beta}}) = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1},$$

jos $\mathbf{V}$ on ei-singulaarinen. Jos oletetaan normaalijakauma, niin $\tilde{\boldsymbol{\beta}}$ on myös suurimman uskottavuuden estimaattori. Edellä oletettiin matriisi $\mathbf{V}$ tunnetuksi. Käytännössä matriisia $\mathbf{V}$ ei yleensä kuitenkaan tunneta, vaan se korvataan estimaatilla $\hat{\mathbf{V}}$. Nyt kuitenkaan $\tilde{\boldsymbol{\beta}}$ ei ole enää paras lineaarinen ja harhaton estimaattori, mutta $\tilde{\boldsymbol{\beta}}$ on suurimman uskottavuuden estimaattori, jos $\hat{\mathbf{V}}$ on $\mathbf{V}$:n SU-estimaatti. Samoin

$$(\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1}$$

on $Var(\tilde{\boldsymbol{\beta}})$:n alaspäin harhainen estimaattori, koska estimaattorina $(\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1}$ ei ota huomioon sitä, että $\mathbf{V}$:n parametrit on estimoitu.

2.1.2 Satunnaisvaikutusten ennustaminen

Tarkastellaan mallia

$$y_{ij} = \mu + \alpha_i + \epsilon_{ij},$$

missä satunnaismuuttujat α_i ja ϵ_{ij} ovat keskenään riippumattomia ($j = 1, \ldots, n$; $i = 1, \ldots, a$) sekä riippumattomasti normaalisti jakautuneita $N(0, \sigma_\alpha^2)$ ja $N(0, \sigma_\epsilon^2)$. Parametrien α_i prediktorina voidaan käyttää ehdollista odotusarvoa

$$\hat{\alpha}_i = E(\alpha_i \mid \bar{y}_{i.}).$$

Jos oletetaan normaalijakauma

$$\begin{pmatrix} \alpha_i \\ \bar{y}_{i.} \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ \mu \end{pmatrix}, \begin{pmatrix} \sigma_\alpha^2 & \sigma_\alpha^2 \\ \sigma_\alpha^2 & \sigma_\alpha^2 + \frac{\sigma_\epsilon^2}{n} \end{pmatrix} \right),$$

niin

$$\begin{aligned} \hat{\alpha}_i &= E(\alpha_i \mid \bar{y}_{i.}) \\ &= \frac{n\sigma_\alpha^2}{n\sigma_\alpha^2 + \sigma_\epsilon^2}(\bar{y}_{i.} - \mu). \end{aligned}$$

Prediktori $\hat{\alpha}_i$ ei kuitenkaan vielä ole käyttökelpoinen käytännön tarkasteluissa, koska se sisältää tuntemattomat parametrit σ_α^2, σ_ϵ^2 ja μ. Ennusteelle saadaan numeerinen arvo, kun tuntemattomat parametrit korvataan estimaateillaan. Saadaan siis

$$\hat{\alpha}_i = \frac{n\hat{\sigma}_\alpha^2}{n\hat{\sigma}_\alpha^2 + \hat{\sigma}_\epsilon^2}(\bar{y}_{i.} - \bar{y}_{..}).$$

Jos esimerkiksi $\hat{\sigma}_\alpha^2 = 120$, $\hat{\sigma}_\epsilon^2 = 4.06$, $\bar{y}_{i.} = 41$ ja $\bar{y}_{..} = 44.94$, niin saadaan

$$\hat{\alpha}_i = \frac{2 \times 120}{2 \times 120 + 4.06}(41 - 44.94) = -3.87.$$

Yleisesti $\boldsymbol{\beta}$ ja $\mathbf{u}$ ratkaistaan ns. sekamalliyhtälöistä, jotka johdetaan vektoreiden $\mathbf{y}$ ja $\mathbf{u}$ yhteisjakaumasta

$$f(\mathbf{y}, \mathbf{u}) = g(\mathbf{y} \mid \mathbf{u})h(\mathbf{u})$$

$$= C \exp[-\frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})'\mathbf{R}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\mathbf{u})] \exp[-\frac{1}{2}\mathbf{u}'\mathbf{D}^{-1}\mathbf{u}].$$

Sekamalliyhtälöt voidaan lausua seuraavasti:

$$\begin{pmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + \mathbf{D}^{-1} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \mathbf{u} \end{pmatrix} = \begin{pmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{y} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{y} \end{pmatrix}.$$

Ratkaisuiksi saadaan

$$\tilde{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-}\mathbf{X}'\mathbf{V}^{-1}\mathbf{y}$$

ja

$$\tilde{\mathbf{u}} = \mathbf{D}\mathbf{Z}'\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}\tilde{\boldsymbol{\beta}}).$$

Nyt kun $\mathbf{V}$ tunnetaan, $\tilde{\boldsymbol{\beta}}$ on parametrin $\boldsymbol{\beta}$ BLUE (Best Linear Unbiased Estimator) ja $\tilde{\mathbf{u}}$ on parametrin $\mathbf{u}$ BLUP (Best Linear Unbiased Predictor). Huomaa, että normaalisuutta ei nyt välttämättä tarvitse olettaa, vaikka sekamalliyhtälöt sinänsä johdettiinkin normaalisuusoletuksesta.

2.1.3 Kovarianssimatriisin estimointi

Mikäli kovarianssimatriisin $\mathbf{V}$ parametrit on estimoitu, saadaan helposti laskettua estimaatit mallin kiinteälle osalle sekä tarvittaessa ennusteet satunnaisvaikutuksille. Jos oletetaan normaalijakauma

$$\begin{pmatrix} \mathbf{u} \\ \boldsymbol{\epsilon} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \right),$$

saadaan estimaatit matriisien $\mathbf{D}$ ja $\mathbf{R}$ parametreille $\boldsymbol{\theta}$ maksimoimalla logaritmoitu uskottavuusfunktio

$$2l(\mathbf{D}, \mathbf{R}) = -\log \mid \mathbf{V} \mid -\mathbf{r}'\mathbf{V}^{-1}\mathbf{r},$$

missä $\mathbf{r} = \mathbf{y} - \mathbf{X}(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-}\mathbf{X}'\mathbf{V}^{-1}\mathbf{y}$. Maksimointi voidaan suorittaa esimerkiksi Newton-Raphson menetelmällä tai EM-algoritmilla. Eräs tärkeä SU-estimaattorin ominaisuus on, että asymptoottisesti

$$Var(\hat{\boldsymbol{\theta}}) \approx [\mathbf{I}(\theta)]^{-1},$$

missä

$$\mathbf{I}(\theta) = E[\frac{\partial l}{\partial \boldsymbol{\theta}} \frac{\partial l}{\partial \boldsymbol{\theta}'}]$$

on ns. Fisherin informaatiomatriisi. Lisäksi voidaan näyttää, että hyvin yleisten ehtojen vallitessa SUE on konsistentti ja asymptoottisesti normaalisti jakautunut odotusarvona $\boldsymbol{\theta}$ ja kovarianssimatriisina Fisherin informatiomatriisin käänteismatriisi

$$\hat{\boldsymbol{\theta}} \sim AN(\boldsymbol{\theta}, [\mathbf{I}(\theta)]^{-1}).$$

Suurimman uskottavuuden menetelmässä maksimodaan uskottavuusfunktio, kun havainnot $\mathbf{y}$ on annettu. Niin sanotussa REML- menetelmässä maksimoidaan uskottavuusfunktio, joka saadaan muunnoksesta $\mathbf{K}'\mathbf{y}$, missä $\mathbf{K}$ on annettu täysiasteinen (sarakeaste) matriisi siten, että $\mathbf{K}'\mathbf{X} = \mathbf{O}$. REML-menetelmässä maksimoidaan funktio

$$2l(\mathbf{D}, \mathbf{R})_R = 2l(\mathbf{D}, \mathbf{R}) - \log \mid \mathbf{X}'\mathbf{V}^{-1}\mathbf{X} \mid .$$

Tämä funktio voidaan johtaa $\mathbf{K}'\mathbf{y}$:n tiheysfunktiosta $N(\mathbf{0}, \mathbf{K}'\mathbf{V}\mathbf{K})$. Huomaa, että uskottavuusfunktio ei nyt sisällä mallin kiinteätä osaa.

Huomautus 1. Suurimman uskottavuuden menetelmällä saadut varianssiestimaatit ovat (alaspäin) harhaisia.

Huomautus 2. REML-tekniikalla saadaan yleensä vähemmän harhaisia estimaatteja.

*Huomautus 3. REML-menetelmää käytettäessä on huomioitava, että uskottavuusfunktio ei pysy samana, kun muutetaan mallin kiinteää osaa (**X**-matriisia). Tästä seuraa esimerkiksi se, että kun käytetään uskottavuusfunktiopohjaisia menetelmiä mallin valintaan, on mallin kiinteän osan pysyttävä samana.*

2.1.4 Toistomittaukset sekamallin avulla

Olkoon nyt N aineiston riippumattomien yksilöiden lukumäärä ja olkoon $\mathbf{y}_i = (y_{i1}, \ldots, y_{in_i})'$ vastearvojen vektori. Oletetaan jokaiselle yksilölle sekamalli

$$\mathbf{y}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i + \boldsymbol{\epsilon}_i,\ i = 1, \ldots, N,$$

missä $\mathbf{y}_i$ on $n_i \times 1$ havaintovektori ja $\mathbf{X}_i$ ja $\mathbf{Z}_i$ ovat annettuja matriiseja. Lisäksi oletetaan, että

$$\begin{cases} \mathbf{u}_i \sim N(\mathbf{0}, \mathbf{D}) \\ \boldsymbol{\epsilon}_i \sim N(\mathbf{0}, \mathbf{R}_i) \\ \mathbf{u}_1, \ldots, \mathbf{u}_N, \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_N \ \text{ ovat riippumattomia,} \end{cases}$$

kun $i = 1, \ldots, N$. Oletuksista seuraa, että

$$\mathbf{y}_i \sim N(\mathbf{X}_i\boldsymbol{\beta}, \mathbf{V}_i)$$

ja

$$Cov(\mathbf{y}_i, \mathbf{y}_j) = \mathbf{O}, i \neq j,$$

missä

$$\mathbf{V}_i = \mathbf{Z}_i\mathbf{D}\mathbf{Z}_i' + \mathbf{R}_i,$$

kun $i, j = 1, \ldots, N$. Usein oletetaan, että $\mathbf{R}_i = \sigma^2\mathbf{I}_{n_i}$, jolloin

$$\mathbf{V}_i = \mathbf{Z}_i\mathbf{D}\mathbf{Z}_i' + \sigma^2\mathbf{I}_{n_i}.$$

Matriisiesityksenä

$$\begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_N \end{pmatrix} = \begin{pmatrix} \mathbf{X}_1 & \mathbf{Z}_1 & \mathbf{O} & \ldots & \mathbf{O} \\ \mathbf{X}_2 & \mathbf{O} & \mathbf{Z}_2 & \ldots & \mathbf{O} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ \mathbf{X}_N & \mathbf{O} & \mathbf{O} & \ldots & \mathbf{Z}_N \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_N \end{pmatrix} + \boldsymbol{\epsilon}$$

$$\mathbf{y} = (\mathbf{X}, \mathbf{Z}) \begin{pmatrix} \boldsymbol{\beta} \\ \mathbf{u} \end{pmatrix} + \boldsymbol{\epsilon} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\epsilon}.$$

Jos tunnetaan

$$\mathbf{V} = \begin{pmatrix} \mathbf{V}_1 & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{V}_2 & & \vdots \\ \vdots & & \cdots & \\ \mathbf{O} & & \cdots & \mathbf{V}_N \end{pmatrix},$$

niin parametrien $\boldsymbol{\beta}$ BLUE on

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{y} \\ &= (\sum_{i=1}^{N} \mathbf{X}_i\mathbf{V}_i^{-1}\mathbf{X}_i)^{-1} \sum_{i=1}^{N} \mathbf{X}_i'\mathbf{V}_i^{-1}\mathbf{y}_i \end{aligned}$$

ja havainnolle i $\mathbf{u}_i$:n BLUP on

$$\hat{\mathbf{u}}_i = \mathbf{D}\mathbf{Z}_i'\mathbf{V}_i^{-1}(\mathbf{y}_i - \mathbf{X}_i\hat{\boldsymbol{\beta}}).$$

Esimerkki 6. Dental-aineiston mallintaminen.

Esimerkkiaineistossa koululaisilta on mitattu tietty hampaistoon liittyvä etäisyys eri ikävuosina.

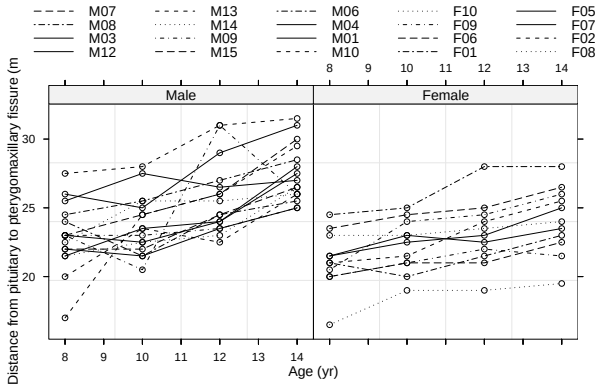
- 4 mittauskertaa: mittaukset 8, 10, 12 ja 14-vuoden iässä
- 11 tyttöä ja 16 poikaa

Jos etäisyys (y) kasvaa lineaarisesti ajan (t) funktiona, niin

$$y = \beta_0 + \beta_1 t + \epsilon.$$

Voidaan ajatella, että tasoparametriin β_0 liittyy satunnaisvaikutus u (ks. kuva 6). Tästä saadaan sekamalli

$$y = (\beta_0 + u) + \beta_1 t + \epsilon,$$



Kuva 6: Kasvukäyrät Dental-aineistossa

missä u ja ϵ ovat riippumattomia sekä $u \sim N(0, d^2)$ ja $\epsilon \sim N(0, \sigma^2)$. Matriisiesityksenä

- tytöt: $\mathbf{y}_i = \mathbf{X}\boldsymbol{\beta}_1 + \mathbf{Z}u_i + \boldsymbol{\epsilon}_i,\ i = 1, \ldots, 11$
- pojat: $\mathbf{y}_i = \mathbf{X}\boldsymbol{\beta}_2 + \mathbf{Z}u_i + \boldsymbol{\epsilon}_i,\ i = 12, \ldots, 27$

missä siis

$$\mathbf{X} = \begin{pmatrix} 1 & 8 \\ 1 & 10 \\ 1 & 12 \\ 1 & 14 \end{pmatrix} \quad \text{ja} \quad \mathbf{Z} = \mathbf{1}.$$

Jos nyt oletetaan, että $\mathbf{R} = \sigma^2\mathbf{I}$, niin

$$Var(\mathbf{y}_i) = d^2\mathbf{1}\mathbf{1}' + \sigma^2\mathbf{I}.$$

Edellä olevalla kovarianssimatriisilla on ns. *tasakorrelaatiorakenne (uniform structure, compound symmetry)*. Yhtenä sekamallina lausuttuna saadaan

$$\mathbf{y} = \begin{pmatrix} \mathbf{X}_1 & \mathbf{O} \\ \vdots & \vdots \\ \mathbf{X}_{11} & \mathbf{O} \\ \mathbf{O} & \mathbf{X}_{12} \\ \vdots & \vdots \\ \mathbf{O} & \mathbf{X}_{27} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{Z}_1 & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{Z}_2 & & \vdots \\ \vdots & & \ddots & \\ \mathbf{O} & \cdots & & \mathbf{Z}_{27} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{27} \end{pmatrix} + \boldsymbol{\epsilon}$$

eli

$$\mathbf{y} = \mathbf{X}^*\boldsymbol{\beta} + \mathbf{Z}^*\mathbf{u} + \boldsymbol{\epsilon}.$$

Huom. Koska nyt $\mathbf{X}_1 = \cdots = \mathbf{X}_{27}$ ja $\mathbf{Z}_1 = \cdots \mathbf{Z}_{27}$, voidaan aineisto esittää myös ns. kasvukäyrämallina (GMANOVA). Nyt saadaan

$$\mathbf{X}^* = \left(\begin{pmatrix} \mathbf{1}_{11} & \mathbf{0}_{11} \\ \mathbf{0}_{16} & \mathbf{1}_{16} \end{pmatrix} \otimes \mathbf{X} \right) \text{ ja } \mathbf{Z}^* = (\mathbf{I}_{27} \otimes \mathbf{Z}),$$

jolloin malli on

$$\begin{aligned} \mathbf{y} &= \left(\begin{pmatrix} \mathbf{1}_{11} & \mathbf{0}_{11} \\ \mathbf{0}_{16} & \mathbf{1}_{16} \end{pmatrix} \otimes \mathbf{X} \right) \boldsymbol{\beta} + (\mathbf{I}_{27} \otimes \mathbf{Z})\mathbf{u} + \boldsymbol{\epsilon} \\ &= (\mathbf{A} \otimes \mathbf{X})\boldsymbol{\beta} + (\mathbf{I}_{27} \otimes \mathbf{Z})\mathbf{u} + \boldsymbol{\epsilon}, \end{aligned}$$

missä

$$(\mathbf{A} \otimes \mathbf{X}) = \left(\begin{pmatrix} \mathbf{1}_{11} & \mathbf{0}_{11} \\ \mathbf{0}_{16} & \mathbf{1}_{16} \end{pmatrix} \otimes \mathbf{X} \right).$$

Tällöin

$$E(\mathbf{y}) = (\mathbf{A} \otimes \mathbf{X})\boldsymbol{\beta}$$

ja

$$\begin{aligned} Var(\mathbf{y}) &= (\mathbf{I}_{27} \otimes \mathbf{Z})(\mathbf{I} \otimes \mathbf{D})(\mathbf{I}_{27} \otimes \mathbf{Z}') + \sigma^2\mathbf{I}_{27} \\ &= (\mathbf{I}_{27} \otimes \mathbf{ZDZ}') + \sigma^2\mathbf{I}_{27}. \end{aligned}$$

Koska $\mathbf{Z} = \mathbf{1}_4$, saadaan

$$Var(\mathbf{y}) = (\mathbf{I}_{27} \otimes d^2 \mathbf{1}_4 \mathbf{1}_4') + \sigma^2 \mathbf{I}_{27}.$$

Jos sovelletaan tulosta

$$vec(\mathbf{ABC}) = (\mathbf{C}' \otimes \mathbf{A}) vec\mathbf{B},$$

niin saadaan

$$(\mathbf{A} \otimes \mathbf{X})\boldsymbol{\beta} = vec(\mathbf{XBA}'),$$

missä $\boldsymbol{\beta} = vec(\mathbf{B})$. Malli $\mathbf{XBA}'$ on ns. kasvukäyrämalli (GMANOVA, Generalized Multivariate Analysis of Variance). Tässä mallissa $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$, jolloin

$$E(\mathbf{Y}) = \mathbf{XBA}'.$$

Perinteinen kasvukäyrämallien teoria perustuu kuitenkin siihen, että satunnaisvirheille ei oleteta mitään erityistä rakennetta, jolloin analyysit voidaan perustaa perinteisten monimuuttujamallien teoriaan. Huomaa, että kasvukäyrämallissa oletetaan, että yksilöt $\mathbf{y}_i$ on mitattu samoissa aikapisteissä eikä puuttuvaa tietoa sallita. Yleisessä sekamallissa ei näitä rajoituksia ole.

Seuraavassa ajossa R-ohjelmistolla on Dental-aineistoon sovitettu esimerkin 6 sekamalli, jossa virhevarianssit estimoidaan erikseen tytöille ja pojille.

```
>  library(nlme)
Loading required package: nls
>  data(Orthodont)
>  l1<-lme(distance~Sex+Sex:age-1, data=Orthodont, random=~1, weights=
+ varIdent(c(Female=0.5), ~1|Sex))
>  summary(l1)
    ...
Random effects:
  Formula: ~1 | Subject
```

```
         (Intercept) Residual
StdDev:    1.847574 1.669823

Variance function:
  Structure: Different standard deviations per stratum
  Formula: ~1 | Sex
  Parameter estimates:
     Male    Female
1.0000000 0.4678937
Fixed effects: distance ~ Sex + Sex:age - 1
                   Value Std.Error DF   t-value p-value
SexMale       16.340625 1.1450949 25 14.270105  <.0001
SexFemale     17.372727 0.8123610 25 21.385478  <.0001
SexMale:age    0.784375 0.0933460 80  8.402882  <.0001
SexFemale:age  0.479545 0.0526752 80  9.103815  <.0001
    ...
```

Estimoidut mallit tytöille ja pojille ovat:

$$y = (17.372727 + u_0) + 0.479545 \times t + \epsilon,$$

missä $\hat{Var}(u_0) = 1.847574^2$ ja $\hat{Var}(\epsilon) = (0.4678937 * 1.669823)^2$ sekä

$$y = (16.340625 + u_0) + 0.784375 \times t + \epsilon,$$

missä $\hat{Var}(u_0) = 1.847574^2$ ja $\hat{Var}(\epsilon) = 1.669823^2$.

Seuraavassa on vastaavan tyyppinen ajo uudemmalla R-ohjelmalla *lmer*, joka on suunniteltu yleistettyihin lineaarisiin sekamalleihin. Huomaa, että virhetermin varianssia ei nyt mallinneta erikseen tytöille ja pojille.

```
> library(nlme)
> library(lme4)
```

```
Loading required package: Matrix
Loading required package: lattice
> data(Orthodont)
> mmod<-lmer(distance~Sex+Sex:age-1+(1|Subject), data=Orthodont)
> summary(mmod)
Linear mixed-effects model fit by REML
Formula: distance ~ Sex + Sex:age - 1 + (1 | Subject)
   Data: Orthodont
   AIC   BIC logLik MLdeviance REMLdeviance
 443.8 457.2 -216.9      428.7        433.8
Random effects:
 Groups   Name        Variance Std.Dev.
 Subject  (Intercept) 3.2986   1.8162
 Residual             1.9221   1.3864
number of obs: 108, groups: Subject, 27

Fixed effects:
              Estimate Std. Error t value
SexMale       16.34063    0.98131   16.65
SexFemale     17.37273    1.18351   14.68
SexMale:age    0.78438    0.07750   10.12
SexFemale:age  0.47955    0.09347    5.13

Correlation of Fixed Effects:
            SexMal SexFml SxMl:g
SexFemale    0.000
SexMale:age -0.869  0.000
SexFemale:g  0.000 -0.869  0.000
```

3 Regressiosplinit

3.1 Kantafunktioesitys

Seuraavassa tarkasteltavat menetelmät voidaan esittää ns. kantafunktioiden avulla. Nämä funktiot on tyypillisesti saatu jollakin sopivalla muunnoksella alkuperäisten selittäjien arvoista. Varsinainen malli on siten muotoa

$$g = \sum_{j=0}^{p} \beta_j f_j(t),$$

missä funktiot f_j ovat nyt kantafunktioita. Seuraavassa eräitä esimerkkejä mahdollisista kantafunktioista.

- Lineaarinen regressio: $f_1(t) = t$,
- Polynomiregressio: $f_1(t) = t,\ f_2(t) = t^2, \ldots, f_p(t) = t^p$,
- Selittäjien muunnokset: $f_j(t) = log(t),\ \sqrt{}(t)$ jne
- Indikaattorifunktiot: $f_j(t) = I(L_j \leq t < U_j)$.

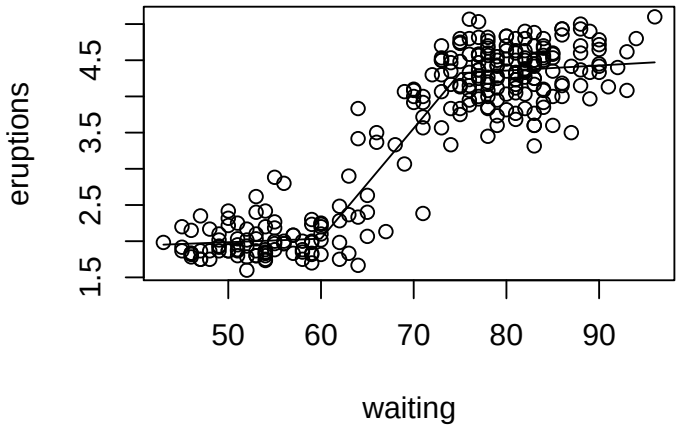
3.2 Paloittain lineaarinen malli

Jos tarkoituksena on sovittaa paloittain lineaarinen malli siten, että havaintoväli $[a, b]$ jaetaan kahteen osaan solmukohdan κ_1 perusteella, niin tämä saataisiin esimerkiksi valitsemalla

$$f_0 = I(t < \kappa_1);\ f_1 = I(t < \kappa_1)t;\ f_2 = I(t \geq \kappa_1) \text{ ja } f_3 = I(t \geq \kappa_1)t$$

Näin saatu käyrä ei kuitenkaan ole välttämättä jatkuva solmukohdassa κ_1. Usein halutaan käyrä, joka toteuttaa tietyt jatkuvuusehdot. Esimerkiksi tässä voitaisiin vaatia

$$\beta_0 + \beta_1\kappa_1 = \beta_2 + \beta_3\kappa_1.$$



Kuva 7: Old Faithfull aineiston sovitettu *broken stick*-malli.

Tämä johtaa estimoitavien parametrien suhteen yhteen rajoitteeseen, joten estimoitavia parametrejä jää 3 kappaletta.

Yleisemmin solmukohtia $\kappa_1, \ldots, \kappa_K$ voi tietenkin olla useampia välillä $[a, b]$. *Solmukohdat* $a < \kappa_1 < \cdots < \kappa_K < b$ *jakavat havaintovälin* $[a, b]$ *pienempiin osaväleihin.* Lineaarinen Splini-estimaattori on muotoa

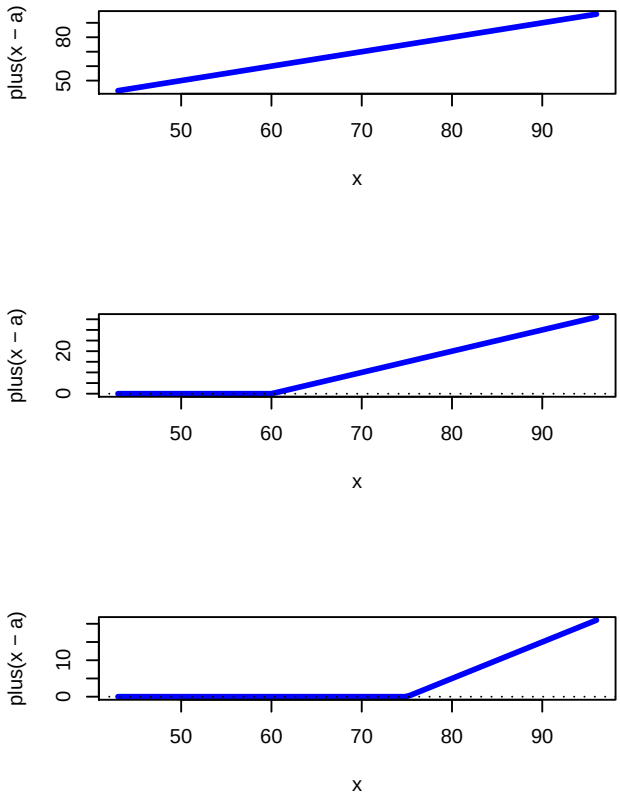
$$g(t) = \beta_0 + \beta_1 t + \sum_{k=1}^{K} u_k (t - \kappa_k)_+,$$

missä $(a)_+ = a$, jos $a > 0$ muuten $a = 0$. Kyseisen funktion kanta on nyt

$$1, t, (t - \kappa_1)_+, \ldots, (t - \kappa_K)_+.$$

Seuraavassa aineistoon Old Faithfull on sovitettu paloittain lineaarinen malli.

```
> library(MASS)
> data(faithful)
> names(faithful)
[1] "eruptions" "waiting"
> faithful<-faithful[order(faithful$waiting),]
> attach(faithful)
> knots<-c(0,60,75)
> rhs<-function(x,c) ifelse (x>c, x-c,0)
> dm<-outer(waiting, knots,rhs)
> l<-lm(eruptions~dm)
> summary(l)
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.743701   0.433474   4.023 7.49e-05 ***
dm1          0.004848   0.008048   0.602    0.547
dm2          0.147633   0.012554  11.760  < 2e-16 ***
dm3         -0.145383   0.010164 -14.304  < 2e-16 ***


...
> plot(eruptions~waiting); lines(waiting, predict(l))
```



Kuva 8: Kuvaajat kantafunktioista.

Kuvaan 8 on piirretty tarvittavat kantafunktiot ja alla on käytetty R-koodi.

```
> plot.basis<-function(x,a){ma<-max(x); mi<-min(x)
curve( plus(x-a), from=mi, to=ma,  lwd = 3, col = "blue" )
abline(h=0, v=0, lty=3)}
> plus<-function (x) ifelse( x >= 0, x, 0 )
> op <- par(mfrow=c(3,1))
> plot.basis(waiting,0); plot.basis(waiting,60); plot.basis(waiting,75)
> par(op)
```

3.3 MARS-menetelmä

MARS on lyhenne sanoista Multivariate Adaptive Regression Splines. Se soveltuu erittäin hyvin tilanteeseen, missä selittäjien määrä on suuri. Menetelmän perustana on paloittain lineaarisen mallin kantafunktiot $(x-\kappa)_+$ ja $(\kappa-x)_+$. Lähtökohtana on kantafunktioiden kokoelma

$$C = \{(x_j - \kappa)_+, (\kappa - x_j)_+\},\ \kappa \in \{x_{1j}, x_{2j}, \ldots, x_{Nj}\}, j = 1, 2, \ldots, p.$$

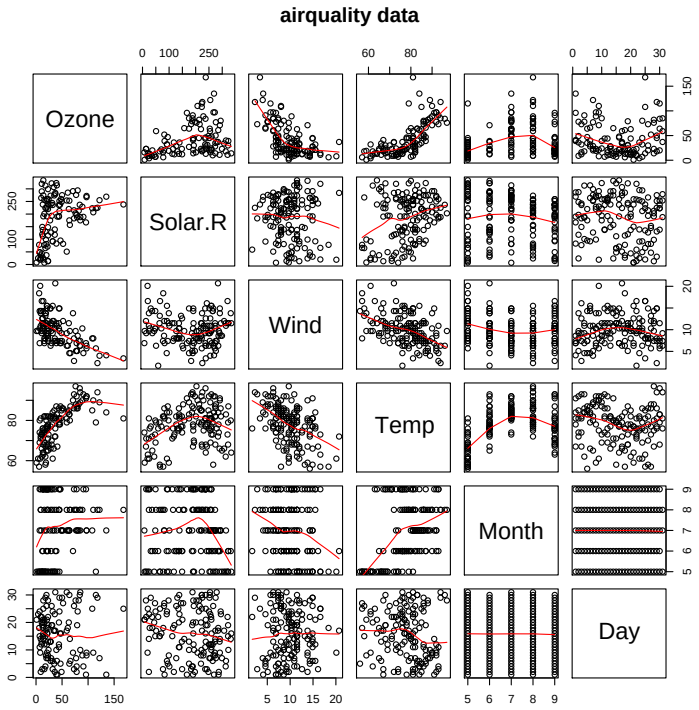
Mallinvalinta suoritetaan periaatteessa samaan tapaan kuin eteenpäin askeltavassa regressioanalyysissä, kun selittäjät ja niiden tulot valitaan joukosta C. Saatu malli on siten muotoa

$$g = \beta_0 + \sum_{m=1}^{M} \beta_m f_m,$$

missä jokainen f_m on joukon C funktio tai kahden tai useamman funktion tulo.

Malli muodostetaan iteratiivisesti siten, että alkutilanteessa on mukana vain vakiotermi. Seuraavaksi käydään läpi kaikki mahdolliset selittäjäkandidaatit kaikkien mahdollisten solmukohtin kanssa (solmukohdat ovat nyt muuttujien mitattuja arvoja) ja parhaan yhteensopivuuden tuottava kantafunktio valitaan selittäjäkandidaatiksi. Sama toistetaan seuraavassa vaiheessa, kun mallissa on mukana jo valittu(t) selittäjä(t). Erilaisia sääntöjä voidaan lisäksi soveltaa mahdollisten interaktiotermien valintaan.

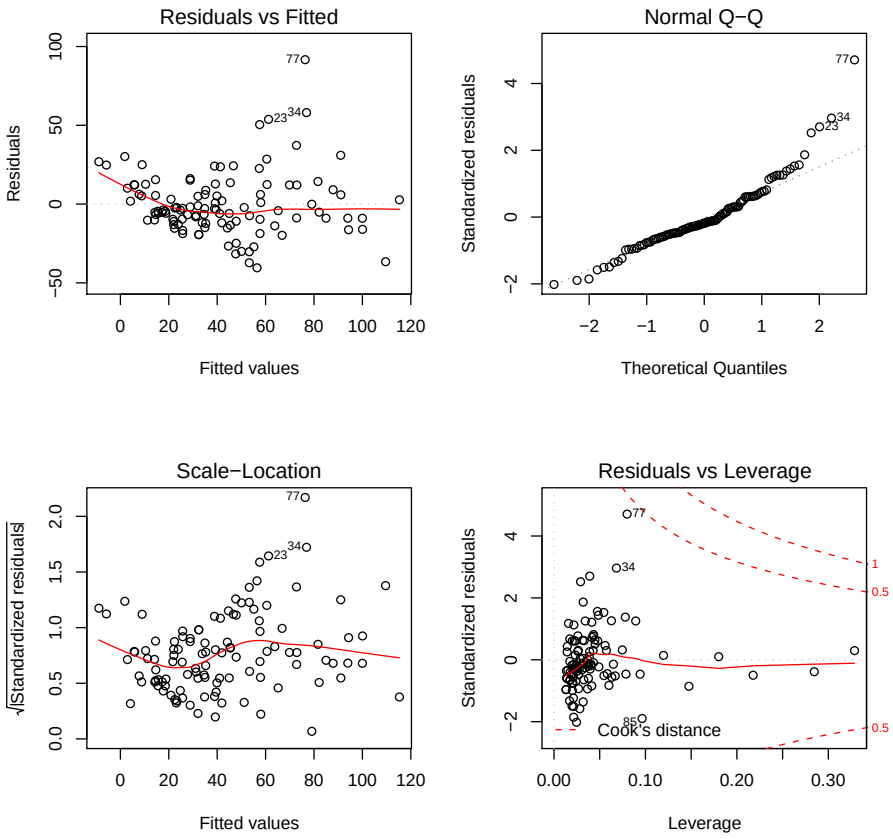
Tarkastellaan mallinnusta aineistossa *airquality*, jossa selitettävä muuttuja on otsonipitoisuus (Ozone). Koko aineiston sirontakuvio on kuviossa 9.



Kuva 9: Aineiston Airquality sirontakuvio.

```
> library(mda)
> data(airquality)
> require(graphics)
> pairs(airquality, panel = panel.smooth, main = "airquality data")
> dim(airquality)
[1] 153   6
> sum(is.na(airquality)) # aineistossa puuttuvia
[1] 44
> m<-matrix(nr=0,nc=6, data=0) # tehdään täydellinen data
> for (i in 1:153) {if (6==sum(!is.na(airquality[i,])))
 {m<-rbind(m, airquality[i,])}}
```

```
> names(m)
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
> library(mda)
> # ohjelmalle annetaan y ja x, nk=max dim, order=int. max dim
> l<-mars(m[,-1], m[,1], nk=5, order=2)
> l$s # valitut termit
[1] 1 2 3 4 5
> l$factor[l$s,] # malliin tulee (temp-k1), (k1-temp), (wind-k2) ja (k2-wind)
     Solar.R Wind Temp Month Day
[1,]       0    0    0     0   0
[2,]       0    0    1     0   0
[3,]       0    0   -1     0   0
[4,]       0    1    0     0   0
[5,]       0   -1    0     0   0
> l$cuts[l$s,] # Solmukohdat valituille
     [,1] [,2] [,3] [,4] [,5]
[1,]    0  0.0    0    0    0
[2,]    0  0.0   85    0    0
[3,]    0  0.0   85    0    0
[4,]    0 13.2    0    0    0
[5,]    0 13.2    0    0    0
> l$coef # kerrointen estimaatit
          [,1]
[1,] 29.825050
[2,]  3.039206
[3,] -1.433630
[4,]  2.782741
[5,]  5.332001
> par(mfrow=c(2,2)); ll<-lm(m[,1] ~l$x-1); plot(ll); par(mfrow=c(1,1))
```



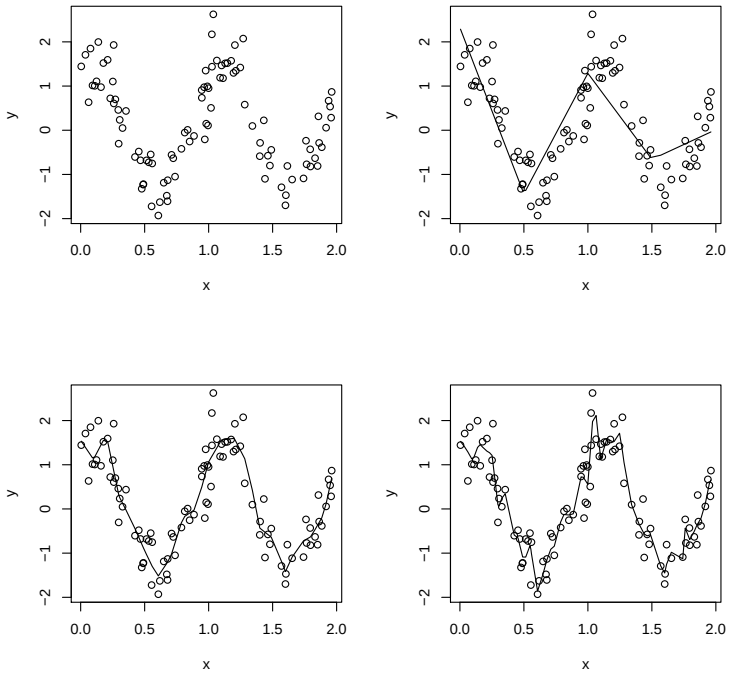
Kuva 10: MARS-malliin liittyvää diagnostiikkaa.

Valittu malli on nyt

$$\hat{g} = 29.8251 + 3.0392(Temp - 85)_+ - 1.4336(85 - Temp)_+$$

$$+2.7827(Wind - 13.2)_+ + 5.3320(13.2 - Wind)_+$$

ja sen selitysaste on noin 86 %. Kyseessä on paloittain lineaarinen malli, jossa muuttujan $Temp$ solmukohta on 85 ja muuttujan $Wind$ solmukohta 13.2. Jos sovitetaan tavallinen regressiomalli, jossa on mukana myös muuttujien $Temp$ ja $Wind$ toisen asteen termit, niin selitysasteeksi saadaan n. 68 %.



Kuva 11: Alkuperäinen aineisto ja sovitettu paloittain lineaarinen mallli, kun solmukohtia on 3, 11 ja 41 kappaletta.

3.4 Solmukohtien määrä ja sijainti

Käytettäessä esimerkiksi katkaistua kantaa

$$1, t, (t - \kappa_1)_+, \ldots, (t - \kappa_K)_+$$

on solmukohtien $\kappa_1, \ldots, \kappa_K$ valinta tärkeä osa käyrän sovitusta. Jos solmukohtia on liian vähän, niin saatetaan menettää joitakin tärkeitä ykstyiskohtia sovitteesta. Jos taas solmukohtia on liikaa, satunnaisvaihtelu saattaa vaikuttaa liikaa sovitteeseen. Seuraavassa tarkastellaan solmukohtien määrän vaikutusta simuloidussa aineistossa.

```
> op <- par(mfrow=c(2,2))
> N <- 100; x <- runif(N, 0, 2); x <- x[order(x)]
> y <- sin(2*pi*x) + cos(2*pi*x)+0.5*rnorm(N)
> plot(x,y)
> knots<- c(0,0.5, 1, 1.5) # Solmut 0.5,1 ja 1.5
> dm<-outer(x, knots,rhs)
> y.p <- predict(lm(y ~ dm)); plot(x,y); lines(x,y.p)
> knots<- c(0,seq(0,2,by=.1))
> rhs<-function(x,c) ifelse (x>c, x-c,0)
> dm<-outer(x, knots,rhs)
> y.p <- predict(lm(y ~ dm)); plot(x,y); lines(x,y.p)
> knots<- c(0,seq(0,2,by=.05))
> dm<-outer(x, knots,rhs)
> y.p <- predict(lm(y ~ dm)); plot(x,y); lines(x,y.p)
> par(op)
```

Yksi mahdollisuus on käyttää solmukohtien (kantafunktioiden) valintaan samoja mallinvalintamenetelmiä kuin regressionanalyysissä (ks. esim. MARS-menetelmä). Jos solmukohtien määrä on iso, saattaa tämä kuitenkin johtaa laskennallisesti melko raskaisiin operaatioihin. Eräs mahdollisuus on sijoittaa solmukohdat tasaisesti havaintovälille, jolloin

$$\kappa_k = a + (b-a)r/(K+1),\ r = 1, \ldots, K.$$

Jos mittapisteet eivät ole tasaisesti jakautuneita, niin saattaa puolestaan olla järkevää sijoittaa solmukohdat valittuihin otoskvantiileihin. Esimerkiksi seuraavanlaista sääntöä on käytetty:

$$\kappa_k = 100 \times \{(k+1)/(K+2)\}\%\mathrm{n}$$

otoskvantiili, kun $k = 1 \ldots, K$, missä $K = \max\{5, \min(0.25 * n, 35)\}$.

3.5 Kerrointen rajoittaminen

3.5.1 Harja(Ridge)-estimaattori

Eräs mahdollisuus on käyttää kaikki solmukohdat, mutta rajoittaa niiden vaikutusta. Jos valitaan aluksi esimerkiksi

$$\mathbf{D} = \begin{pmatrix} \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times K} \\ \mathbf{0}_{K\times 2} & \mathbf{I}_{K\times K} \end{pmatrix},$$

niin eräs rajoite voisi olla

$$\boldsymbol{\beta}'\mathbf{D}\boldsymbol{\beta} \leq C.$$

(Huomaa, että kahta ensimmäistä kerrointa ei nyt rajoiteta.) Lagrangen kertojan menetelmällä voidaan osoittaa, että tämä johtaa minimoitavaan funktioon

$$PLS = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}'\mathbf{D}\boldsymbol{\beta},$$

jonka ratkaisu on

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{D})^{-1}\mathbf{X}'\mathbf{y},$$

missä $\lambda > 0$. Estimaattori $\hat{\boldsymbol{\beta}}$ on nyt eräänlainen yleistetty harja-estimaattori. Tavalliseen harja-estimaattoriin päädytään, kun valitaan $\mathbf{D} = \mathbf{I}$, jolloin

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y}.$$

Harja-estimaattoria on käytetty parametrisessa regressiossa vähentämään estimoitujen kertoimien vaihtelua. Niin sanottuun *Lasso*-menetelmään päädytään, jos asetetaan rajoite

$$\sum \mid \beta_j \mid \leq C.$$

3.5.2 Sekamallin käyttö

Kertoimia voidaan rajoittaa myös sekamallin avulla. Tällöin voidaan valita

$$\mathbf{X} = [\mathbf{1}, \mathbf{x}],$$

missä $\mathbf{x} = (t_1, \ldots, t_n)'$ ja

$$\mathbf{Z} = [(\mathbf{x} - \kappa_1 \mathbf{1})_+, \ldots, (\mathbf{x} - \kappa_K \mathbf{1})_+],$$

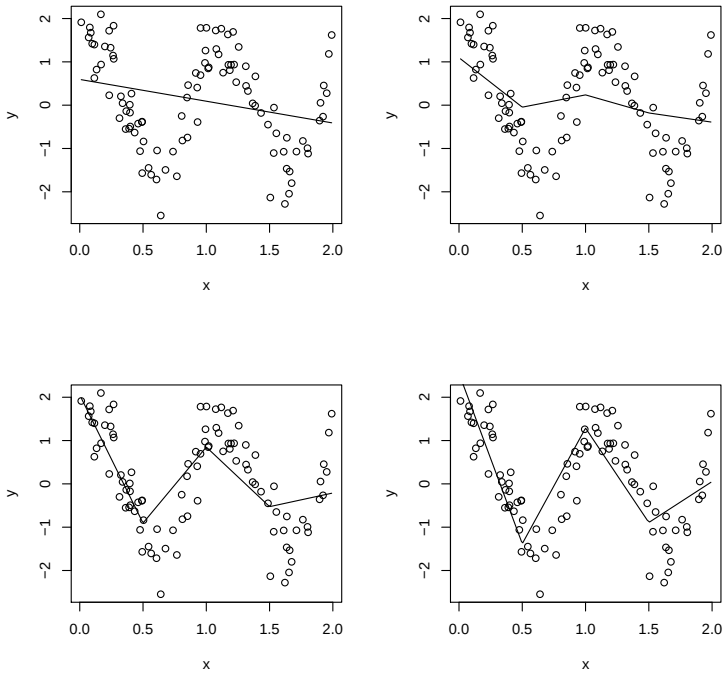
jolloin

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\epsilon},$$

missä

$$\text{Var} \begin{pmatrix} \mathbf{u} \\ \boldsymbol{\epsilon} \end{pmatrix} = \begin{pmatrix} \sigma_u^2 \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \sigma_\epsilon^2 \mathbf{I} \end{pmatrix},$$

Nyt esimerkiksi jos σ_u^2 on pieni, niin kerrointen $\mathbf{u}$ vaihtelu on vähäistä, jolloin estimaatteja tasoitetaan voimakkaasti. Jos taas σ_u^2 on iso, niin kertoimet $\mathbf{u}$ vaihtelevat voimakkasti, eli estimaatteja tasoitetaan vähemmän. Havainnollistetaan asiaa edellisen esimerkin aineiston avulla. Tarkastellaan tilannetta, jossa on valittu ainoastaan kolme solmukohtaa 0.5, 1 ja 1.5. Kuten kuviosta 9 havaitaan, on saatu sovite melko karkea. Tasoitetaan nyt saatua sovitetta sekamallin avulla. Valitaan $\sigma_\epsilon^2 = 0.25$ sekä $\sigma_u^2 = 0$, $\sigma_u^2 = 0.25$, $\sigma_u^2 = 2.5$ ja $\sigma_u^2 = 250$. Seuraavassa on esitetty saatu kuvio sekä tarvittava R-koodi.



Kuva 12: Simuloituun aineistoon sovitettu sekamallin avulla tasoitettu käyrä, kun $\sigma_\epsilon^2 = 0.25$ sekä $\sigma_u^2 = 0$, $\sigma_u^2 = 0.25$, $\sigma_u^2 = 2.5$ ja $\sigma_u^2 = 250$.

Nähdään, että kun $\sigma_u^2 = 0$ saadaan regressiosuora. Jos sitten kasvatetaan varianssin σ_u^2 arvoa, niin käyrä saa lisää piirteitä, jotka poikkeavat regressiosuorasta. Voitaisiin ajatella, että varianssien suhde $\sigma_\epsilon^2/\sigma_u^2 = \lambda$ on eräänlainen tasoituskerroin.

```
> N <- 100
> x <- runif(N, 0, 2); x <- x[order(x)]
> y <- sin(2*pi*x) + cos(2*pi*x)+0.5*rnorm(N)
> knots<- c(0,0.5, 1, 1.5) # Solmut 0.5,1 ja 1.5
> Z<-outer(x, knots,rhs)
> X<-cbind(1,x)
> sfitti<-function(y,X,Z, su,se){
n<-dim(Z)[1]; p<-dim(Z)[2]
D<-su*diag(p); R<-se*diag(n)
V<-Z%*%D%*%t(Z)+R
bh<-solve(t(X)%*%solve(V)%*%X)%*%t(X)%*%solve(V)%*%y
uh<-D%*%t(Z)%*%solve(V)%*%(y-X%*%bh)
yh<-X%*%bh+Z%*%uh
yh}
> yh<-sfitti(y,X,Z,0,0.25)
> op <- par(mfrow=c(2,2))
> plot(x,y); lines(x,yh)
> yh<-sfitti(y,X,Z,0.25,0.25)
> plot(x,y); lines(x,yh)
> yh<-sfitti(y,X,Z,2.5,0.25)
> plot(x,y); lines(x,yh)
> yh<-sfitti(y,X,Z,250,0.25)
> plot(x,y); lines(x,yh)
> par(op)
```

3.6 Käyrän tasoittaminen

Sen sijaan, että rajoitetaan vain yksittäisten kertoimien β_j vaikutusta, voidaan myös tasoittaa koko sovitetta $\hat{\mathbf{g}}$. Tarkastellaan lauseketta

$$PLS = \sum_{i=1}^{n}[y_i - g_i(t)]^2 + \lambda \int [f''(t)]^2 dt$$

missä λ on annettu positiivinen vakio. Jos nyt termi $\int (f''(t))^2 dt$ korvataan diskreetillä (tasavälinen aineisto) approksimaatiolla $\sum_{i=2}^{n-1}(g_{i+1} - 2g_i + g_{i-1})^2$, niin minimoitava funktio on

$$PLS = \sum_{i=1}^{n}(y_i - g_i(t))^2 + \lambda \sum_{i=2}^{n-1}(g_{i+1} - 2g_i + g_{i-1})^2.$$

Olkoon nyt

$$\mathbf{Q} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ -2 & 1 & \ddots & 0 & 0 \\ 1 & -2 & 1 & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & & & 1 \\ & & & & -2 \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix}.$$

Helposti nähdään, että termi RP on nyt muotoa

$$RP = \mathbf{g}'\mathbf{Q}\mathbf{Q}'\mathbf{g},$$

jolloin

$$PLS = (\mathbf{y} - \mathbf{g})'(\mathbf{y} - \mathbf{g}) + \lambda \mathbf{g}'\mathbf{Q}\mathbf{Q}'\mathbf{g}.$$

Jos nyt merkitään

$$\mathbf{g} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u},$$

missä $\mathbf{X} = (\mathbf{1}, \mathbf{x})$ ja $\mathbf{x} = (1, \ldots, n)'$, niin saadaan

$$PLS = [\mathbf{y} - (\mathbf{X}\boldsymbol{\beta} + \mathbf{u})]'[\mathbf{y} - (\mathbf{X}\boldsymbol{\beta} + \mathbf{u})] + \lambda \mathbf{u}'\mathbf{Q}\mathbf{Q}'\mathbf{u},$$

koska $\mathbf{Q}'\mathbf{X} = \mathbf{0}$. Derivoimalla saadaan helposti

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

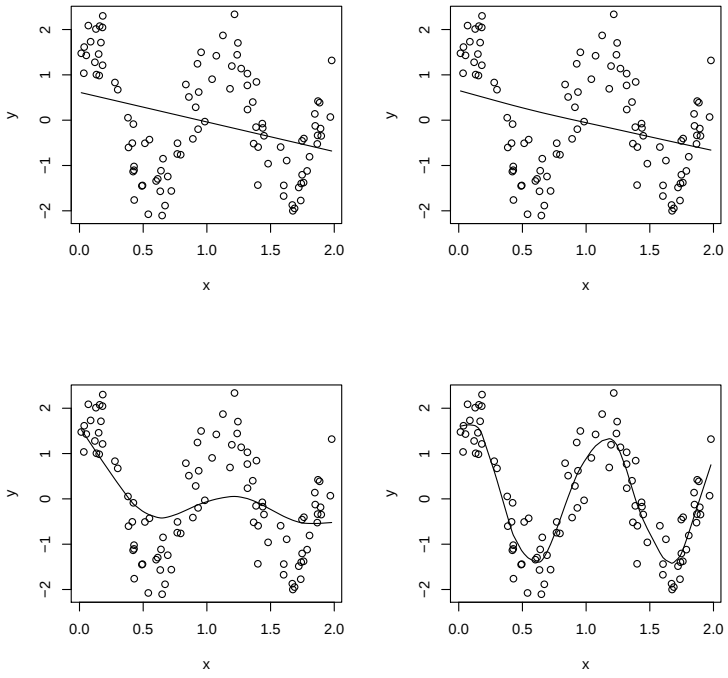
ja

$$\hat{\mathbf{u}} = (\mathbf{I} + \lambda\mathbf{Q}\mathbf{Q}')^{-1}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}).$$

Voidaan myös näyttää, että yhtäpitävästi

$$\hat{\mathbf{g}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \hat{\mathbf{u}} = (\mathbf{I} + \lambda\mathbf{Q}\mathbf{Q}')^{-1}\mathbf{y} = \mathbf{S}_\lambda\mathbf{y}.$$

```
> op <- par(mfrow=c(2,2))
> N <- 100
> x <- runif(N, 0, 2); x <- x[order(x)]
> y <- sin(2*pi*x) + cos(2*pi*x)+0.5*rnorm(N)
> nabla<-function(x){
+ h<-diff(x); n<-length(x)
+ m<-matrix(nr=n, nc=(n-2), data=c(0))
+ for (i in 1:(n-2)) {m[i,i]<-1/h[i]}
+ for (i in 1:(n-2)) {m[i+1,i]<-((1/h[i])+(1/h[i+1]))*-1}
+ for (i in 1:(n-2)) {m[i+2,i]<-1/h[i+1]}; m}
> q<-nabla(x)
> l<-lm(y ~ x)
> plot(x,y); lines(x,predict(l))
> lambda<-1000
> u<-solve(diag(100)+lambda*q%*%t(q))%*%(y-predict(l))
> plot(x,y); lines(x,predict(l)+u)
> lambda<-10
> u<-solve(diag(100)+lambda*q%*%t(q))%*%(y-predict(l))
> plot(x,y); lines(x,predict(l)+u)
> lambda<-0.1
> u<-solve(diag(100)+lambda*q%*%t(q))%*%(y-predict(l))
> plot(x,y); lines(x,predict(l)+u)
```



Kuva 13: Simuloituun aineistoon sovitettu tasoitettu käyrä, kun tasoitusparametri $\lambda \to \infty$, $\lambda = 1000$, $\lambda = 10$ ja $\lambda = 0.1$.

Helposti havaitaan, että kun $\lambda \to \infty$, saadaan regressiosuora. Pienentämällä tasoituskertoimen arvoa saadaan käyrä kulkemaan yhä tarkemmin havaintopisteiden kautta.

3.6.1 Yhteys sekamalleihin

Voidaan osoittaa, että

$$(\mathbf{I} + \lambda \mathbf{Q}\mathbf{Q}')^{-1} = \mathbf{I} - \mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}\mathbf{Q}' + \mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}\{(\mathbf{Q}'\mathbf{Q})^{-1} + \lambda \mathbf{I}\}^{-1}\mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}.$$

Kun nyt $\mathbf{Q}'\mathbf{X} = \mathbf{0}$, niin

$$\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' + \mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}\{(\mathbf{Q}'\mathbf{Q})^{-1} + \lambda \mathbf{I}\}^{-1}\mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}$$

ja kun merkitään $\mathbf{Z} = \mathbf{Q}(\mathbf{Q}'\mathbf{Q})^{-1}$, niin saadaan

$$(\mathbf{I} + \lambda \mathbf{Q}\mathbf{Q}')^{-1} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' + \mathbf{Z}\{\mathbf{Z}'\mathbf{Z} + \lambda \mathbf{I}\}^{-1}\mathbf{Z}'.$$

Edelleen voidaan osoittaa, että

$$\{\mathbf{Z}'\mathbf{Z} + \lambda \mathbf{I}\}^{-1}\mathbf{Z}' = \mathbf{Z}'(\lambda \mathbf{I} + \mathbf{Z}\mathbf{Z}')^{-1}.$$

Nyt siis

$$\hat{\mathbf{g}} = (\mathbf{I} + \lambda \mathbf{Q}\mathbf{Q}')^{-1}\mathbf{y}$$

voidaan kirjoittaa

$$\hat{\mathbf{g}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{u}},$$

missä

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

ja

$$\hat{\mathbf{u}} = \mathbf{Z}'(\mathbf{Z}\mathbf{Z}' + \lambda \mathbf{I})^{-1}\mathbf{y}.$$

Estimaattorit $\hat{\boldsymbol{\beta}}$ ja $\hat{\mathbf{u}}$ seuraavat myös sekamalliyhtälöiden ratkaisuista, kun valitaan $\mathbf{D} = \lambda^{-1}\mathbf{I}$ ja $\mathbf{R} = \mathbf{I}$. Tämä on tulkittavissa niin, että kun $Var(\mathbf{u})$ on suuri, on tasoituskerroin pieni, jolloin tasoitetaan vähän. Toisessa ääripäässä $Var(\mathbf{u}) = \mathbf{0}$, jolloin $\lambda \to \infty$ ja sovitteena saadaan regressiosuora.

3.7 Muita mahdollisia kantoja

3.7.1 Katkaistu polynomikanta

Lineaarista Splini-estimaattoria vastaava kanta (*truncated line basis*) on

$$1, t, (t-\kappa_1)_+, \ldots, (t-\kappa_K)_+,$$

jolloin sovitettu käyrä on paloittain lineaarinen malli. Jonkinverran tasaisemman näköinen sovite saadaan, jos vaaditaan, että sovitettavan funktion derivaatta on jatkuva. Tarvittava kanta on tällöin

$$1, t, t^2, (t-\kappa_1)_+^2, \ldots, (t-\kappa_K)_+^2.$$

Yleisemmin astetta p oleva katkaistu polynomikanta (*truncated power basis*) voidaan esittää muodossa

$$1, t, \ldots, t^p, (t-\kappa_1)_+^p, \ldots, (t-\kappa_K)_+^p$$

ja tätä vastaava sovite on

$$g(t) = \sum_{s=0}^{p} \beta_s t^s + \sum_{r=1}^{K} \beta_{p+r}(t-\kappa_r)_+^p.$$

Nähdään, että osavälillä $[\kappa_r, \kappa_{r+1})$ saadaan

$$g(t) = \sum_{s=0}^{p} \beta_s t^s + \sum_{l=1}^{r} \beta_{p+l}(t-\kappa_l)_+^p,$$

mikä on astetta p oleva polynomi. Kuitenkin, kun $r = 1, 2, \ldots, K$, on p-kertainen derivaatta

$$g^{(p)}(\kappa_{r-}) = p!(\beta_p + \sum_{l=1}^{r-1} \beta_{p+l})$$

ja

$$g^{(p)}(\kappa_{r+}) = p!(\beta_p + \sum_{l=1}^{r} \beta_{p+l}).$$

Nyt siis

$$g^{(p)}(\kappa_{r+}) - g^{(p)}(\kappa_{r-}) = p!\beta_{p+r},$$

jolloin astetta p olevat derivaatat ovat epäjatkuvia (funktio hyppää kohdassa κ_r termin $p!\beta_{p+r}$ verran, kun $r = 1, 2, \ldots, K$), mutta astetta $p-1$ olevat derivaatat ovat jatkuvia. Nyt siis esimerkiksi jos $p = 3$, niin kyseisellä sovitteella on jatkuvat toiset derivaatat. Yleensä p on korkeintaan 3. Jos estimoinnissa halutaan käyttää yleistettyä harja-estimaattoria, niin

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{D})^{-1}\mathbf{X}'\mathbf{y}, \text{ missä } \mathbf{D} = \begin{pmatrix} \mathbf{0}_{p+1\times p+1} & \mathbf{0}_{p+1\times K} \\ \mathbf{0}_{K\times p+1} & \mathbf{I}_{K\times K} \end{pmatrix},$$

jolloin siis kertoimia $\beta_0, \ldots, \beta_p$ ei tasoiteta, mutta kertoimia $\beta_{p+1}, \ldots, \beta_{p+k}$ tasoitetaan riippuen tasoituskertoimesta λ.

3.7.2 B-Splinit

Katkaistu polynomikanta on varsin käyttökelpoinen, jos solmukohdat on valittu huolella tai, jos sovitetta tasoitetaan tarvittaessa. Käytännössä ongelmaksi voi muodostua se, että kanta ei ole ortogonaalinen. Tämä saattaa johtaa numeerisesti ei-stabiiliin sovitteeseen, jos solmukohtia on paljon tai jos tasoituskertoimen arvo on lähellä nollaa. Usein pienimmän neliösumman menetelmän tapauksessa käytetään jotakin ekvivalenttia kantaa, jolla on paremman numeeriset ominaisuudet. Yleisin tällainen vaihtoehto lienee B-Splini-kanta.

Jos nyt katkaistua polynomikantaa vastaava suunnittelumatriisi on $\mathbf{X}$ ja $\mathbf{X}_B$ on vastaavasta B-splini kannasta saatu saman asteen ja samoin solmukohdin saatu suunnittelumatriisi, niin $\mathbf{X}_B = \mathbf{X}\mathbf{M}$, missä $\mathbf{M}$ on ei-singulaarinen. On helppo nähdä, että yleistetyllä harja-estimaattorilla saatu sovite on nyt

$$\hat{\mathbf{g}} = \mathbf{X}_B(\mathbf{X}_B'\mathbf{X} + \lambda\mathbf{M}'\mathbf{D}\mathbf{M})^{-1}\mathbf{X}_B\mathbf{y}.$$

Käytännössä $\mathbf{X}$-matriisin sarakkeiden muunnos tehdään sopivilla algoritmeilla tietokoneohjelmissa. Tällöin mallin muodostuksessa käytetyt kantafunktiot eivät välttämättä ole samat kuin ne joita käytetään estimoinnissa.

3.8 Lineaarisista tasoitusmenetelmistä

Vaikka regressiosplineissä mallinnus tapahtuu eri pohjalta kuin tavallisissa regressiomalleissa, on menetelmissä kuitenkin periaatteessa yhteinen perusta. Molemmissa lähestymistavoissa ajatellaan, että malli saadaan havaintojen $\mathbf{y}$ lineaarisena funktiona. Tavallisessa regressiomallissa

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

ja vastaava sovite saadaan lausekkeella

$$\hat{\mathbf{y}} = \mathbf{H}\mathbf{y},$$

missä $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}$ on ns. hattumatriisi. Vastaavasti regressiosplineissä saadaan

$$\hat{\mathbf{y}} = \mathbf{S}_\lambda \mathbf{y},$$

missä $\mathbf{S}_\lambda$ on ns. tasoittajamatriisi. Nyt jos λ ajatellaan annetuksi (ei riipu havainnoista $\mathbf{y}$), niin molemmissa tapauksissa sovite saadaan havaintojen lineaarisena funktiona, ts

$$\hat{\mathbf{y}} = \mathbf{L}\mathbf{y}.$$

3.8.1 Virhehajotelma

Usein sovitteen yhteensopivuuden mittana käytetään keskineliövirhettä

$$MSE[\hat{g}(t)] = E[(\hat{g}(t) - g(t))^2],$$

joka voidaan myös esittää hajotelmana

$$MSE[\hat{g}(t)] = [E(\hat{g}(t) - g(t))]^2 + Var(\hat{g}(t)).$$

Käytännössä yhteensopivuutta tarkastellaan koko sovitteen osalta, jolloin usein käytetään suuretta (Mean Summed Squared Error)

$$MSSE[\hat{g}(t)] = E\sum_{i=1}^{n}[\hat{g}(t_i) - g(t_i)]^2.$$

Matriisiesityksenä saadaan

$$MSSE(\hat{\mathbf{g}}) = E[(\hat{\mathbf{g}} - \mathbf{g})'(\hat{\mathbf{g}} - \mathbf{g})],$$

missä

$$\hat{\mathbf{g}} = \mathbf{L}\mathbf{y}.$$

Helposti nähdään, että

$$MSSE(\hat{\mathbf{g}}) = [(\mathbf{L} - \mathbf{I})\mathbf{g}]'[(\mathbf{L} - \mathbf{I})\mathbf{g}] + tr[Var(\mathbf{L}\mathbf{y})]$$

ja jos oletetaan $Var(\mathbf{y}) = \sigma^2\mathbf{I}$, niin

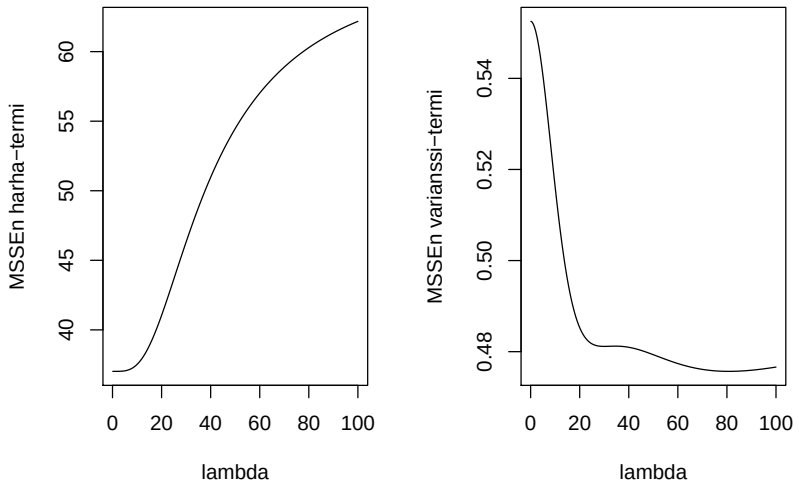
$$MSSE(\hat{\mathbf{g}}) = [(\mathbf{L} - \mathbf{I})\mathbf{g}]'[(\mathbf{L} - \mathbf{I})\mathbf{g}] + \sigma^2 tr(\mathbf{L}\mathbf{L}').$$

Selvästi suureen $MSSE$ ensimmäinen termi mittaa harhaa ja toinen varianssia. Usein harhaa voidaan pienentää varianssin kustannuksella ja päinvastoin. Tutkija joutuu siten tekemän kompromissin näiden kahden ristiriitaisen päämäärän välillä (ns. *bias-variance trade-off*). Jos $\mathbf{L} = \mathbf{S}_\lambda$, niin isolla λ arvolla harha kasvaa, mutta varianssi pienenee ja päinvastoin.

Seuraavassa kuviossa pyritään havainnollistamaan varianssin ja harhan käyttäytymistä aineistossa *Faithful*, johon on sovitettu paloittain lineaarinen malli (ks Kuva 7). Estimoinnissa käytetään harjaestimaattori siten, että kahta ensimmäistä kerrointa ei rajoiteta.

```
library(MASS)
faithful<-faithful[order(faithful$waiting),]
attach(faithful)
knots<-c(0,60,75)
rhs<-function(x,c) ifelse (x>c, x-c,0)
dm<-outer(waiting, knots,rhs)
X<-cbind(1,dm)
x<-1:1000
```

```
d<-diag(c(0,0,1,1))
bias2<-function(x,d,y){
D<-x*d
l<-X%*%solve(t(X)%*%X+x*D)%*%t(X)
e<-(l%*%y)-y
bias<-t(e)%*%e
bias}
par(mfrow=c(1,2))
v<-numeric(1000)
for (i in 1:1000){v[i]<-bias2(0.1*i,d, eruptions)}
plot((0.1*x),v, type="l", xlab="lambda", ylab="MSSEn harha-termi")
variance<-function(x,d,y){
D<-x*d
l<-X%*%solve(t(X)%*%X+x*D)%*%t(X)
rss<-(l%*%y)-y
rss<-t(rss)%*%rss
ll<-sum(diag(l%*%t(l)))
dfr<-length(y)+ll-2*sum(diag(l))
s2<-(1/dfr)*rss
s2<-s2*ll
s2}
v<-numeric(1000)
for (i in 1:1000){v[i]<-variance(0.1*i,d, eruptions)}
plot((0.1*x),v, type="l", xlab="lambda", ylab="MSSEn varianssi-termi")
par(mfrow=c(1,1))
```



Kuva 14: "bias-variance trade-off"aineistossa Faithful

3.8.2 Tasoittajan vapausasteet

Käytettyä tasoitusparametrin λ arvoa voi olla melko vaikea mitenkään tulkita. Näin erityisesti silloin, kun ollaan kaukana arvoalueen ääripäistä. Usein käytetty menettely on määritellä

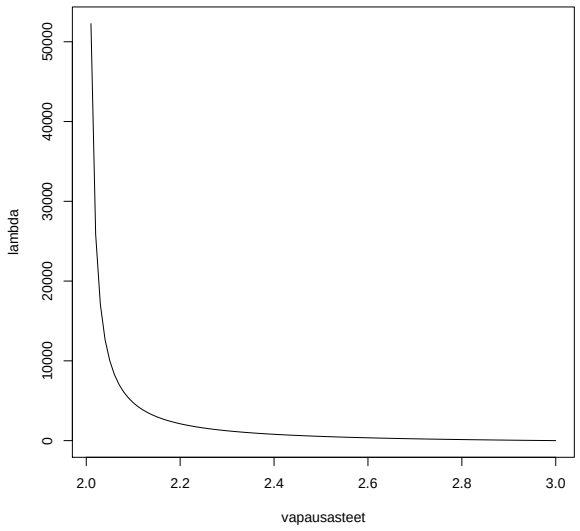
$$df = tr(\mathbf{S}_\lambda).$$

Voisimme siis ajatella, että vastaavaan sovitteeseen tarvittaisiin df kantafunktiota, jolloin

$$df = tr(\mathbf{H}),$$

missä $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Helposti nähdään, että esimerkiksi lineaaristen Splinien tapauksessa, jos käytetään harja-estimaattoria, niin saadaan

$$tr(\mathbf{S}_0) = 2 + K,$$



Kuva 15: Vapausasteiden ja tasoitusparametrin yhteys esimerkkitilanteessa.

missä K on solmukohtien (tai kantafunktioiden) lukumäärä, ja kun $\lambda \to \infty$, niin

$$tr(\mathbf{S}_\lambda) \to 2.$$

Voidaan näyttää, että $df = tr(\mathbf{S}_\lambda)$ ja parametrilla λ on monotoninen yhteys, jolloin relaatio on käännettävissä. Voimme siis suoritaa mallinvalintaa myös niin, kiinnitämme vapausasteet df ja sitten laskemme vastaavan λ arvon. Kasvattamalla vapausasteita yhdellä voitaisiin ajatella, että malliin tuodaan lisää yksi selittäjä jne. Diagnostisia tarkasteluja voidaan nyt tehdä regressiomalleihin kehitettyjen tekniikoiden avulla.

Kuvassa 14 tarkastellaan vapausasteiden ja tasoituskertoimen yhteyttä toisen asteen polynomimallissa, jossa toisen asteen kerrointa rajoitetaan.

```
> X<-outer(1:10, 0:2, "^")
> v<-numeric(100)
> for (i in 1:100){v[i]<-dftolambda(X,2,2+0.01*i)}
> plot((2+0.01*1:100),v, type="l", xlab="vapausasteet", ylab="lambda")
>
> dftolambda(X,2,2.1)
[1] 4752
> dftolambda(X,2,2.9)
[1] 58.66667

> dftolambda
function(X,p,df){
k<-dim(X)[2]; d<-diag(k); d[1:p, 1:p]<-0
v<-optimize(harjal, c(0,1000000), df,d)
v$min}
>
> harjal
function(x,df,d){
D<-x*d
l<-X%*%solve(t(X)%*%X+D)%*%t(X)
e<-df-sum(diag(l))
e^2}
>
```

Esimerkissä siis $df \in [2,3]$. Erityisesti, jos valitaan $df = 2.1$, niin $\lambda = 4752$ ja kun $df = 2.9$, niin $\lambda = 58.66667$.

3.8.3 Residuaalivarianssin estimointi

Tavallisen regressiomallin tapauksessa residuaalivarianssin harhaton estimaattori on

$$\hat{\sigma}^2 = \frac{1}{n-p}\mathbf{y}'(\mathbf{I}-\mathbf{H})\mathbf{y} = \frac{1}{n-p}\text{RSS},$$

missä $\text{RSS} = \mathbf{y}'(\mathbf{I}-\mathbf{H})\mathbf{y}$. Ei-parametrisessa regressiossa likimain harhaton estimaattori voidaan johtaa seuraavasti. Olkoon nyt

$$\hat{\mathbf{g}} = \mathbf{L}\mathbf{y}.$$

Tarkastellaan odotusarvoa

$$E[(\hat{\mathbf{g}}-\mathbf{y})'(\hat{\mathbf{g}}-\mathbf{y})] = E[\mathbf{y}'(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})\mathbf{y}] = E(\text{RSS}),$$

missä $\text{RSS} = \mathbf{y}'(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})\mathbf{y}$. Kun nyt $E(\mathbf{y}) = \mathbf{g}$ ja koska $E(\mathbf{y}'\mathbf{A}\mathbf{y}) = E(\mathbf{y})'\mathbf{A}E(\mathbf{y}) + tr[\mathbf{A}Cov(\mathbf{y})]$, niin saadaan

$$E(\text{RSS}) = \mathbf{g}'(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})\mathbf{g} + \sigma^2 tr[(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})].$$

Koska

$$\sigma^2 tr[(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})] = \sigma^2\{tr(\mathbf{L}\mathbf{L}') - 2tr(\mathbf{L}) + n\},$$

niin olettamalla $\mathbf{g}'(\mathbf{L}-\mathbf{I})'(\mathbf{L}-\mathbf{I})\mathbf{g} \approx 0$ (harhattomuus), saadaan

$$\hat{\sigma}^2 = \frac{1}{df_r}\text{RSS},$$

missä $df_r = tr(\mathbf{L}\mathbf{L}') - 2tr(\mathbf{L}) + n$. Nyt, jos $\mathbf{L} = \mathbf{H}$ niin

$$df_r = n - tr(\mathbf{H}) = n - p.$$

Tällöin vapausasteet olisivat samat kuin tavallisessa regressiomallissakin. Usein kuitenkin $\mathbf{L} = \mathbf{S}_\lambda$, jolloin $\mathbf{S}_\lambda\mathbf{S}_\lambda \leq \mathbf{S}_\lambda$ ja $2tr(\mathbf{L}) - tr(\mathbf{L}\mathbf{L}') \neq p$.

3.9 Päättelystä

Tilastollinen päättely ei-parametrisessa regressiossa ei ole aivan yhtä suoraviivaista kuin parametrisesa regressiossa. Seuraavaksi tarkastellaan joitakin päättelyn osa-alueita.

3.9.1 Luottamusvälit

Merkitään kohdassa t laskettua sovitetta

$$\hat{g}(t) = \mathbf{l}_t'\mathbf{y},$$

missä sovite saadaan harja-estimaattorilla

$$\mathbf{l}_t = \mathbf{X}(\mathbf{X}'\mathbf{X} + \lambda\mathbf{D})^{-1}\mathbf{x}_t,$$

missä $\mathbf{D} = diag(0, 0, 1, \ldots, 1)$ (jos kahta ensimmäistä kerrointa ei rajoiteta) ja $\mathbf{x}_t$ on kantafunktioista pisteessä t muodostettu ennustevektori. Approksimatiivinen luottamusväli sovitteelle pisteessä t saadaan esimerkiksi seuraavasti:

$$\hat{g}(t) \pm \begin{cases} t_{(1-\alpha/2);df_r)}\ \hat{\sigma}\ \sqrt{\mathbf{l}_t'\mathbf{l}_t}, & \text{jos } n \text{ on pieni} \\ z_{(1-\alpha/2)}\ \hat{\sigma}\ \sqrt{\mathbf{l}_t'\mathbf{l}_t}, & \text{jos } n \text{ on iso.} \end{cases}$$

Jos käytetään sekamalliformulointia (ks. 3.5.2), niin

$$g(t) = \mathbf{x}_t'\beta + \mathbf{z}_t'\mathbf{u}$$

ja vastaava sovite on

$$\hat{g}(t) = \mathbf{x}_t'\hat{\beta} + \mathbf{z}_t'\hat{\mathbf{u}}.$$

Varianssi saadaan nyt ehdollisesta jakaumasta $\mathbf{y} \mid \mathbf{u}$, jolloin

$$Var(\hat{g} \mid \mathbf{u}) = \mathbf{l}_t'\ Var(\begin{pmatrix} \hat{\beta} \\ \hat{u} \end{pmatrix} \mid \mathbf{u})\ \mathbf{l}_t,$$

missä $\mathbf{l}_t' = (\mathbf{x}_t', \mathbf{z}_t')$ ja

$$Var(\begin{pmatrix} \hat{\beta} \\ \hat{u} \end{pmatrix} \mid \mathbf{u}) = \sigma^2(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1}\mathbf{C}'\mathbf{C}(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1},$$

missä $\mathbf{C} = (\mathbf{X}, \mathbf{Z})$. Tästä saadaan

$$s(\hat{g} \mid \mathbf{u}) = \hat{\sigma} \sqrt{\mathbf{l}_t'(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1}\mathbf{C}'\mathbf{C}(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1}\mathbf{l}_t}.$$

Jos nyt oletetaan, että $\epsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I})$, niin

$$\hat{g} \mid \mathbf{u} \sim N(E\{\hat{g} \mid \mathbf{u}\}, Var\{\hat{g} \mid \mathbf{u}\}),$$

missä

$$E\{\hat{g} \mid \mathbf{u}\} = \mathbf{x}_t'\hat{\beta} + \mathbf{z}_t'\hat{\mathbf{u}}$$

vastaa sovitettua käyrää. Nyt saadaan

$$\frac{\hat{g} - E\{\hat{g} \mid \mathbf{u}\}}{s\{\hat{g} \mid \mathbf{u}\}} \mid \mathbf{u} \sim N(0, 1),$$

jolloin approksimatiivinen $100(1-\alpha)\%$ luottamusväli ehdolliselle odotusarvolle $E\{\hat{g} \mid \mathbf{u}\}$ on

$$\hat{g}(t) \pm z_{(1-\alpha/2)}\ s\{\hat{g} \mid \mathbf{u}\}.$$

Mikäli harha on pieni, niin $E\{\hat{g} \mid \mathbf{u}\} \approx g(t)$, jolloin tämä voidaan tulkita luottamusväliksi käyrälle $g(t)$. Harhaa voidaan ottaa huomioon sekamallin avulla. Harhakorjatuksi luottamusväliksi saadaan

$$\hat{g}(t) \pm \begin{cases} t_{(1-\alpha/2;df_r)}\ \ s\{\hat{g}(t) - g(t)\}, & \text{jos } n \text{ on pieni} \\ z_{(1-\alpha/2)}\ \ s\{\hat{g}(t) - g(t)\}, & \text{jos } n \text{ on iso.} \end{cases}$$

missä

$$s\{\hat{g}(t) - g(t)\} = \hat{\sigma}\sqrt{\mathbf{l}_t'(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1}\mathbf{l}_t} \quad \text{ja} \quad \lambda = \frac{\sigma^2}{\sigma_u^2}.$$

Edellä lasketut luottamusvälit on laskettu pisteessä t. Usein ollaan kuitenkin kiinnostuneita ns. samanaikaisista luottamusväleistä, jolloin useammassa pisteessä laskettu luottamusväli pätee samanaikaisesti kaikissa pisteissä. Olkoon nyt muuttujan t mahdollisten arvojen joukko T. Pisteessä t lasketut $100(1-\alpha)\%$ luottamusvälit $\{(L(t), U(t)) : t \in T)$ toteuttavat likimain epäyhtälön

$$P(L(t) \leq g(t) \leq U(t)) \geq 1 - \alpha, \text{ kun } t \in T.$$

Samanaikaisissa luottamusväleissä

$$P(L(t) \leq g(t) \leq U(t) \text{ kaikille } t \in T) \geq 1 - \alpha.$$

Olkoon nyt

$$\hat{\mathbf{g}} - \mathbf{g} = \mathbf{C_t} \begin{pmatrix} \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \\ \hat{\mathbf{u}} - \mathbf{u} \end{pmatrix},$$

missä lineaariselle Splini-estimaatorille

$$\mathbf{C_t} = [\mathbf{1}, \mathbf{t}, (\mathbf{t} - \kappa_1 \mathbf{1})_+, \ldots, (\mathbf{t} - \kappa_K \mathbf{1})_+]$$

ja

$$\begin{pmatrix} \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \\ \hat{\mathbf{u}} - \mathbf{u} \end{pmatrix} \approx N\{\mathbf{0}, \sigma^2(\mathbf{C}'\mathbf{C} + \lambda \mathbf{D})^{-1}\}.$$

Nyt saadaan $100(1-\alpha)\%$ samanaikaisiksi luottamusväleiksi

$$\hat{\mathbf{g}} \pm m_{1-\alpha} \times \begin{pmatrix} s(\hat{g}(t_1) - g(t_1)) \\ s(\hat{g}(t_2) - g(t_2)) \\ \cdots \\ s(\hat{g}(t_n) - g(t_n)) \end{pmatrix},$$

missä $m_{1-\alpha}$ on $(1-\alpha)$ prosentin kvantiili satunnaismuuttujalle

$$\sup_{t \in T} [\frac{\hat{g}(t) - g(t)}{s\{\hat{g}(t) - g(t)\}}] \approx \max_{1 \leq t \leq n} [\frac{\mathbf{C_t} \begin{pmatrix} \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \\ \hat{\mathbf{u}} - \mathbf{u} \end{pmatrix}_t}{s\{\hat{g}(t) - g(t)\}}.]$$

Käytännössä suuretta $m_{(1-\alpha)}$ approksimoidaan simuloimalla. Simuloidaan ensin realisaatio satunnaismuuttujan

$$\begin{pmatrix} \hat{\boldsymbol{\beta}} - \boldsymbol{\beta} \\ \hat{\mathbf{u}} - \mathbf{u} \end{pmatrix} \approx N\{\mathbf{0}, \sigma^2(\mathbf{C}'\mathbf{C} + \lambda\mathbf{D})^{-1}\}$$

jakaumasta ja lasketaan satunnaismuuttujan m realisoitu arvo. Toistetaan menettelyä N kertaa ja järjestetään saadut arvot. Valitaan sitten $m_{1-\alpha}$-arvoksi saadun jakauman järjestyksessä $[(1-\alpha)N]$s luku.

3.9.2 Mallin riittävyyden testaminen

Tarkastellaan lineaarista Splini-estimaattoria

$$g(t) = \beta_0 + \beta_1 t + \sum_{k=1}^{K} u_k (t - \kappa_k)_+,$$

missä riippumattomat satunnaismuuttujat u_k noudattavat normaalijakauma $u_k \sim N(0, \sigma_k^2)$. Yksi mahdollinen hypoteesi voisi olla

$$H_0 : \sigma_u^2 = 0 \quad \text{vs.} \quad H_1 : \sigma_u^2 > 0,$$

jota voidaan periaatteessa testata uskottavuussuhde-testillä. Jos oletetaan

$$\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \mathbf{V}), \quad \text{missä} \quad \mathbf{V} = \sigma_u^2 \mathbf{Z}\mathbf{Z}' + \sigma^2 \mathbf{I},$$

niin logaritmoitu uskottavuusfunktio $(-2\times)$ on

$$-2\ l(\hat{\sigma}_u^2, \hat{\sigma}^2; \mathbf{y})\} = n\ log(2\pi) + log \mid \mathbf{V} \mid + (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

ja jos käytetään ns. REML-menetelmää, niin

$$-2\ l_R(\hat{\sigma}_u^2, \hat{\sigma}^2; \mathbf{y})\} = -2\ l(\hat{\sigma}_u^2, \hat{\sigma}^2; \mathbf{y})\} + log \mid \mathbf{X}'\mathbf{V}^{-1}\mathbf{X} \mid .$$

Testaus perustuu testisuureeseen

$$-2log\ LR(\mathbf{y}) = -2\{l(0, \hat{\sigma}_0^2; \mathbf{y}) - l(\hat{\sigma}_u^2, \hat{\sigma}^2; \mathbf{y})\},$$

missä $\hat{\sigma}_0^2$ minimoi uskottavuusfunktion $-2\ l(0, \hat{\sigma}_0^2; \mathbf{y})$ ja $(\hat{\sigma}_u^2, \hat{\sigma}^2)$ minimoi uskottavuusfunktion $-2\ l(\hat{\sigma}_u^2, \hat{\sigma}^2; \mathbf{y})$. Vaihtoehtoisesti voidaan käyttää myös REML-funktiota. Nollahypoteesi lineaarisuudesta hylätään mikäli saatu testisuuren arvo ylitää nollajakauman kriittisen arvon. Perinteinen χ^2-approksimaatio toimii tässä tilanteessa huonosti, koska kiinnostuksen kohteena oleva parametri on parametrialueen rajalla. Tästä syystä merkitsevyystaso tulisi määritellä simuloimalla.

Esimerkissä tarkastellaan mallin riittävyyden testaamista aineistossa *faithful*. Huomaa, että käytetyissä malleissa on "teknisistä syistä"mukana ylimääräinen satunnainen vakiotermi.

```
> library(MASS)
> faithful<-faithful[order(faithful$waiting),]
> knots<-c(0,60,75)
> rhs<-function(x,c) ifelse (x>c, x-c,0)
> dm<-outer(faithful$waiting, knots,rhs)
> faithful<-data.frame(faithful, z1=dm[,2], z2=dm[,3], ind=1)
> library(nlme)
> faithful<-groupedData(eruptions~waiting|ind, data=faithful)
> la<-lme(eruptions~waiting, data=faithful, random=pdDiag(~z1+z2))
> l0<-lme(eruptions~waiting, data=faithful, random=pdDiag(~1))
> anova(l0,la)
  Model df      AIC      BIC    logLik   Test  L.Ratio p-value
l0     1  4 412.5754 426.9691 -202.2877
la     2  6 270.9932 292.5837 -129.4966 1 vs 2 145.5823  <.0001
> sim<-simulate.lme(l0, m2=la, nsim=1000)
> lrsim<-2*(sim$alt$REML-sim$null$REML)
> sum(lrsim>145.5823)/1000
[1] 0
```

Testaamiseen voidaan käyttää myös parametrisesta regressiosta tuttua F-statistiikkaa. Ei-parametrisen regression tapauksessa testisuureen oletetaan noudattavan approksimatiivisesti F-jakaumaa. Testisuureen lauseke voidaan esittää muodossa

$$F = \frac{R_l^2 - R_s^2}{(1 - R_l^2)(df_{r,s} - df_{r,l})/df_{r,l}},$$

missä R^2 on muuttujien $\mathbf{y}$ ja $\hat{\mathbf{y}}$ korrelaatiokertoimen neliö (mallissa mukana vakiotermi). Indeksillä l viitataan malliin, jonka vapausasteet df_r ovat suuremmat ja $s(< l)$ viittaa testattavaan H_0-malliin. Testisuure noudattaa approksimatiivisesti F-jakaumaa vapausastein

$$df_{r,s} - df_{r,l}, \quad df_{r,l}.$$

Käytännössä jakauma-approksimaatio voi olla huono, joten merkitsevyydestä tulisi varmistua simuloimalla.

Sekä uskottavuussuhde-testiä, että F-statistiikkaa voidaan käyttää myös testaamaan sitä onko kyseisellä muuttujalla vaikutusta vasteeseen. Jos käytetään lineaarista Splini-estimaattoria

$$g(t) = \beta_0 + \beta_1 t + \sum_{k=1}^{K} u_k (t - \kappa_k)_+,$$

niin

$$H_0; \beta_1 = \sigma_u^2 = 0 \quad \text{vs.} \quad H_1 : \beta_1 \neq 0 \text{ tai } \sigma_u^2 > 0.$$

4 Lokaalit polynomimallit

Perusidea lokaalisissa polynomimalleissa on lokaalisesti approksimoida funktiota g jollakin sopivan asteisella polynomilla. Lähtökohtana on Taylorin sarjakehitelmä. Olkoon t_0 kohta, missä funktiota g halutaan approksimoida. Oletetaan, että funktion $g(t)$ derivataat ovat jatkuvia asteeseen $p+1 \geq 0$ saakka pisteessä t_0. Taylorin sarjakehitelmän perusteella funktiota $g(t)$ voidaan nyt lokaalisti approksimoida p-asteisella polynomilla. Tällöin

$$g(t) \approx g(t_0) + (t-t_0)g^{(1)}(t_0) + \cdots + (t-t_0)^p g^{(p)}(t_0)/p!,$$

missä $g^{(r)}(t_0)$ on funktion r-kertainen derivaatta pisteessä t_0. Merkitään nyt $\beta_r = g^{(r)}(t_0)$, $r = 0, \ldots, p$. Olkoon $\hat{\beta}_r, r = 0, 1, 2, \ldots, p$ kertoimet, jotka minimoivat painotetun PNS-kriteerin

$$WLS = \sum_i \{y_i - [\beta_0 + (t_i - t_0)\beta_1 + \cdots + (t_i - t_0)^p \beta_p]\}^2 K_h(t_i - t_0)$$

missä $K_h(.) = K(./h)/h$ on saatu ns. Kernel-funktiosta $K(.)$ ja vakiota $h > 0$ kutsutaan ikkunan leveydeksi (tai tasoituskertoimeksi), koska

$$I_h(t_0) = [t_0 - h, t_0 + h]$$

määrittelee "ikkunan", missä funktion approksimointi suoritetaan. Kernel-funktiolla puolestaan määrätään se miten havainnot ikkunan $I_h(t_0)$ sisällä painotetaan. Merkitään derivaatan $g^{(r)}(t_0)$ estimaattia $\hat{g}_h^{(r)}(t_0)$, jolloin

$$\hat{g}_h^{(r)}(t_0) = r!\hat{\beta}_r, r = 0, 1, \ldots, p.$$

Nyt erityisesti funktion arvon $g(t_0)$ approksimaatioksi saadaan

$$\hat{g}_h(t_0) = \hat{\beta}_0.$$

Merkitään seuraavaksi

$$\mathbf{X} = \begin{pmatrix} 1 & (t_1 - t_0) & \ldots & (t_1 - t_0)^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & (t_n - t_0) & \ldots & (t_n - t_0)^p \end{pmatrix}$$

ja

$$\mathbf{W} = diag(K_h(t_1 - t_0), \ldots, K_h(t_n - t_0)).$$

Minimoitava kriteerifunktio voidaan nyt kirjoittaa muodossa

$$WLS = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'\mathbf{W}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}),$$

missä $\mathbf{y} = (y_1, \ldots, y_n)'$ ja $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)'$. Nyt saadaan

$$\hat{g}_h^{(r)}(t_0) = r!\mathbf{e}'_{r+1}(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y},$$

missä $\mathbf{e}_{r+1}$ on koordinaattivektori siten, että $e_{r+1} = 1$, muutoin nolla. Koko käyrän $g^{(r)}(t)$ estimaatti saadaan, kun menettely toistetaan yli t:n arvoalueen. Tässä keskitytään käyrän $g(t)$ estimointiin, jolloin

$$\hat{g}_h(t_0) = \mathbf{e}'_1(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}.$$

Nyt jos asetetaan $\hat{y}_i = \hat{g}_h(t_i)$, niin

$$\hat{y}_i = \mathbf{l}'(t_i)\mathbf{y},$$

missä $\mathbf{l}'(t_i) = \mathbf{e}'_1(\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}$ ja koko käyrälle saadaan

$$\hat{\mathbf{y}}_h = \mathbf{L}_h\mathbf{y},$$

missä $\hat{\mathbf{y}}_h = (\hat{y}_1, \ldots, \hat{y}_n)'$ ja $\mathbf{L}_h = (\mathbf{l}(t_1), \ldots, \mathbf{l}(t_n))'$.

4.1 Lokaali vakio ja lineaarinen tasoitin

Lokaalisti vakio tasoitin tunnetaan Nadaraya-Watson-estimaattorina. Tämä saadaan, kun asetetaan $p = 0$, jolloin

$$\hat{g}_h(t_0) = \frac{\sum_i K_h(t_i - t_0)y_i}{\sum_i K_h(t_i - t_0)}.$$

Nyt ikkunassa $I_h(t_0) = [t_0 - h, t_0 + h]$ aineistoon sovitetaan vakio. Tällöin siis käyrän sovite $\hat{\beta}_0$ saadaan minimoimalla

$$\sum_i (y_i - \beta_0)^2 K_h(t_i - t_0).$$

Nyt, jos käytetään ns. Uniform-Kernel-funktiota

$$K(t) = I_{[-1,1]}(t)/2 = \begin{cases} 1/2, & t \in [-1,1], \\ 0, & \text{muutoin}, \end{cases}$$

niin nähdään, että estimaattori on havaintojen lokaali keskiarvo

$$\hat{g}_h(t_0) = \frac{\sum_i I_{[t_0-h,t_0+h]}(t_i) y_i}{\sum_i I_{[t_0-h,t_0+h]}(t_i)}.$$

Lineaarinen tasoitin puolestaan saadaan minimoimalla parametrien β_0 ja β_1 suhteen neliösumma

$$\sum_i (y_i - \beta_0 - (t_i - t_0)\beta_1)^2 K_h(t_i - t_0)$$

ja käyrän estimatti pisteessä t_0 on $\hat{g}_h(t_0) = \hat{\beta}_0$. Voidaan osoittaa, että

$$\hat{g}_h(t_0) = \frac{\sum_i [s_2(t_0) - s_1(t_0)(t_i - t_0)] K_h(t_i - t_0) y_i}{\sum_i s_2(t_0) s_0(t_0) - s_1^2(t_0)},$$

missä

$$s_r(t_0) = \sum_i K_h(t_i - t_0)(t_i - t_0)^r, \ r = 0, 1, 2.$$

Usein asteen p valinta ei ole niin tärkeä kuin ikkunan leveyden h. Käytännössä usein lokaali vakio ($p = 0$) tai lineaarinen tasoitin ($p = 1$) ovat tarpeeksi hyviä, jos Kernel-funktio K ja ikkunan leveys h on oikein valittu.

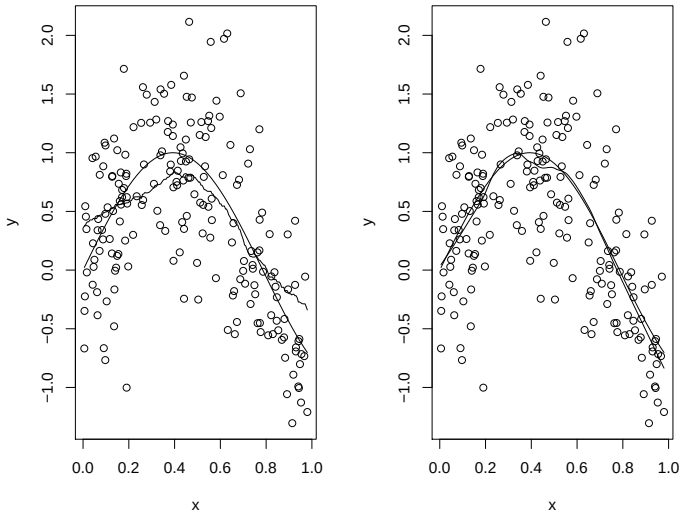
Seuraavassa tarkastellaan simuloitua aineistoa

$$y = \sin(4x) + \epsilon,$$

missä riippumattomat satunnaismuuttujat $x \sim U(0,1)$ ja $\epsilon \sim N(0, 0.25)$. Huomataan helposti, että lokaali vakio on jonkinverran harhainen alueen reunoilla ja keskellä. Funktiolla *loess* saadaan lokaali toisen asteen sovite, kun painofunktiona käytetään ns. *tricube*-funktiota

$$K(t) = (1 - |\, t \,|^3)^3, \ |\, t \,| < 1.$$

Saatu sovite on selvästi parempi kuin lokaalin vakion tapauksessa.



Kuva 16: Lokaali vakio ja toisen asteen polynomitasoitin.

```
x<-runif(200); x<-x[order(x)]
y<-sin(4*x)+rnorm(200,sd=0.5)
par(mfrow=c(1,2))
plot(x,y, type="p"); lines(ksmooth(x,y)); lines(x,sin(4*x))
plot(x,y, type="p"); f<-loess(y~x, span=0.5, degree=2); lines(f$x, f$fitted)
lines(x,sin(4*x))
par(mfrow=c(1,1))
```

Vasen kuvio on saatu sovittamalla lokaali vakio (R-funktio *ksmooth*) ja oikea kuvio saadaan, kun sovitetaan lokaali toisen asteen polynomi (R-funktion *loess*). Tässä on käytetty kahta funktion *loess* argumenttia. Argumentilla *span* määritellään lokaalissa sovitteessa käytettyjen havaintojen suhteellinen osuus ja argumenttilla *degree* määritellään lokaalin polynomin aste.

4.2 Kernel-funktio

Polynomitasoittajissa käytetty Kernel-funktion on yleensä jokin symmetrinen tiheysfunktio. Tasoittajassa h määrittelee ikkunan leveyden ja K sen miten havaintoja painotetaan. Uniform-Kernel lisäksi usein käytetään ns. Gaussian-Kerneliä, joka saadaan standartoidun normaalijakauman tiheysfunktiosta seuraavasti

$$K(t) = exp(-t^2/2)/\sqrt{2\pi}.$$

Uniform- ja Gaussian-Kernelit ovat erikoistapauksia symmetrisestä *Beta*-perheestä

$$K(t) = \frac{1}{Beta(1/2, 1+\gamma)}(1-t^2)_+^{\gamma}, \gamma = 0, 1, \ldots,$$

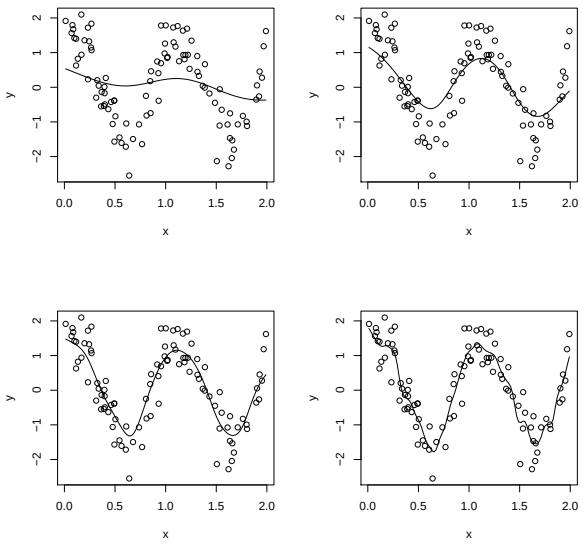
missä *Beta(a,b)* on *Beta*-funktio. Valinnat $\gamma = 0, 1, 2$ ja 3 vastaavat Uniform-, Epanechnikov-, Biweight- ja Triweight-Kerneleitä ja Gaussian-Kernel saadaan, kun $\gamma \to \infty$. Esimerkiksi Epanechnikov Kernel-funktioksi saadaan $K(t) = 3/4(1-t^2)$, kun $\mid t \mid < 1$ ja nolla muulloin. Eräiden yleisten ehtojen vallitessa voidaan näyttää että, kun $n \to \infty$, niin

$$Bias(\hat{g}_h(t_0)) = \begin{cases} O_p(h^{p+1}), \ p \text{ parillinen} \\ O_p(h^{p+2}), \ p \text{ pariton} \end{cases}$$

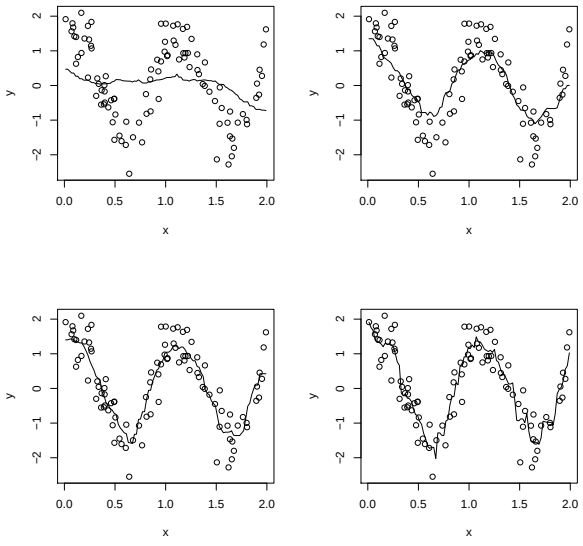
ja

$$Var(\hat{g}_h(t_0)) = O_p\{(nh)^{-1}\},$$

missä $X = O_p(Y)$ tarkoittaa, että X/Y on stokastisesti rajoitettu. Tästä nähdää, että ikkunanleveydellä h voidaan kontrolloida varianssia ja harhaa. Kun h on pieni, harha on pieni mutta varianssi iso ja kun h on iso, niin harha on suuri, mutta varianssi pieni. Seuraavassa ajossa tutkitaan valitun Kernel-funktion ja ikkunan leveyden vaikutusta tasoitukseen.



Kuva 17: Aineisto on tasoitettu Gaussian-Kernelillä, kun h=1, 0.5, 0.25 ja 0.1.

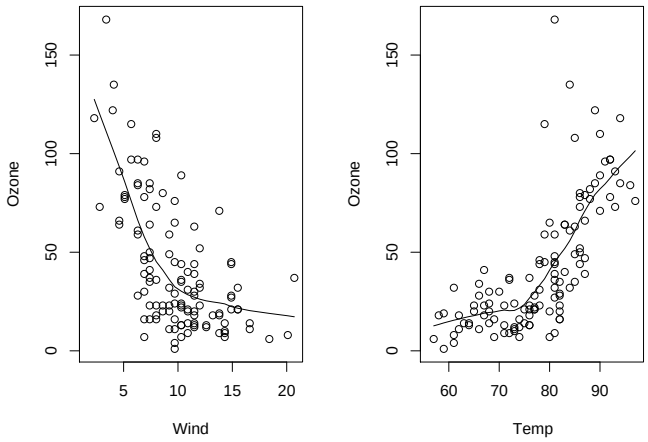


Kuva 18: Aineisto on tasoitettu Uniform-Kernelillä, kun h=1, 0.5, 0.25 ja 0.1.

```
> op <- par(mfrow=c(2,2))
>  x <- runif(N, 0, 2); x <- x[order(x)]
> y <- sin(2*pi*x) + cos(2*pi*x)+0.5*rnorm(N)
> plot(x,y); lines(ksmooth(x,y, "normal", bandwidth=1))
> plot(x,y); lines(ksmooth(x,y, "normal", bandwidth=0.5))
> plot(x,y); lines(ksmooth(x,y, "normal", bandwidth=0.25))
> plot(x,y); lines(ksmooth(x,y, "normal", bandwidth=0.1))
> plot(x,y); lines(ksmooth(x,y, "box", bandwidth=1))
> plot(x,y); lines(ksmooth(x,y, "box", bandwidth=0.5))
> plot(x,y); lines(ksmooth(x,y, "box", bandwidth=0.25))
> plot(x,y); lines(ksmooth(x,y, "box", bandwidth=0.1))
> par(op)
```

Seuraavassa esimerkissä tarkastellaan lokaalisten polynomien käyttöä aineistossa *airquality*. MARS-menetelmän perusteella muuttujalle *Ozone* saatiin paloittain lineaarinen malli, jossa oli mukana selittäjät *Temp* ja *Wind* solmukohtina 85 ja 13.2. Sovitetaan aluksi malli, jossa muuttujaa *Ozone* selitetään erikseen kummallakin selittäjällä.

```
> data(airquality)
> airquality<-airquality[order(airquality$Wind),]
> par(mfrow=c(1,2))
plot(airquality$Ozone~airquality$Wind, ylab="Ozone", xlab="Wind")
> mo1<-loess(airquality$Ozone~airquality$Wind, degree=1, span=0.5, na.action=na.omit)
> summary(mo1)
...
Number of Observations: 116
Equivalent Number of Parameters: 4.73
Residual Standard Error: 23.78
Trace of smoother matrix: 5.57
```



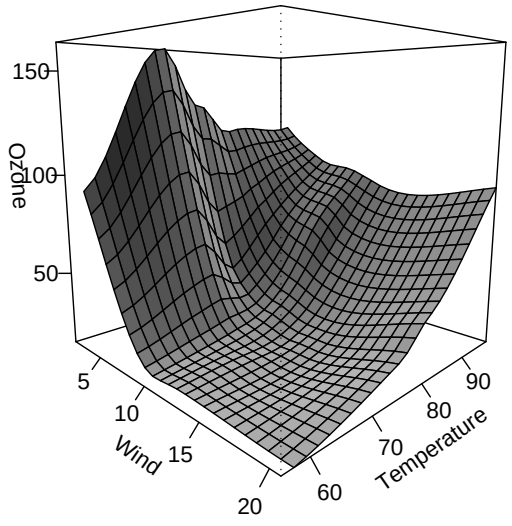
Kuva 19: Aineiston *airquality* muuttujaa *Ozone* selitetty erikseen muuttujilla *Wind* ja *Temp*.

```
...
> lines(mo1$x, mo1$fitted)
> airquality<-airquality[order(airquality$Temp),]
plot(airquality$Ozone~airquality$Temp, ylab="Ozone", xlab="Temp")
> mo2<-loess(airquality$Ozone~airquality$Temp, degree=1, span=0.5, na.action=na.omit)
> summary(mo2)
...
Number of Observations: 116
Equivalent Number of Parameters: 4.14
Residual Standard Error: 22.14
Trace of smoother matrix: 4.85
...
> lines(mo2$x, mo2$fitted)
> par(mfrow=c(1,1))
```

Ilmeisesti kummankaan selittäjän vaikutus vasteeseen ei ole lineaarinen. Seuraavassa tutkitaan mallia, missä on mukana molemmat selittäjät.

```
> mo3<-loess(airquality$Ozone~airquality$Temp+ airquality$Wind,
degree=1, span=0.5, na.action=na.omit)
> anova(mo1,mo3)
...
Analysis of Variance:   denominator df 105.37

       ENP   RSS F-value    Pr(>F)
[1,]     5 61977
[2,]     8 31919      17 6.386e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> anova(mo2,mo3)
...
Analysis of Variance:   denominator df 105.37

       ENP   RSS F-value    Pr(>F)
[1,]     4 54121
[2,]     8 31919      11 5.017e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Malli, jossa on mukana molemmat selittäjät on selvästi parempi kuin yhden selittäjän malli. Seuraavaksi piirretään mallin avulla muodostettu pinta. Havaitaan, että kummankin selittäjän kontribuutio vasteeseen on varsin epälineaarinen.



Kuva 20: Aineistoon *airquality* muuttujien *Wind* ja *Temp* avulla muodostettu pinta.

```
data(airquality)
attach(airquality)
w<-seq(min(Wind), max(Wind), len=25)
t<-seq(min(Temp), max(Temp), len=25)
newdata<-expand.grid(Wind=w, Temp=t)
mo3<-loess(Ozone~Temp+ Wind, degree=1, span=0.5, na.action=na.omit)
fit.ozone<-matrix(predict(mo3, newdata), 25, 25)
persp(w,t,fit.ozone, xlab="Wind", ylab="Temperature",
zlab="Ozone", shade=0.5, ticktype="detailed", theta=45, phi=10)
```

5 Kuutiosplinit

Funktiota g, joka on määritelty välillä $[a, b]$ $(a < t_1 < t_2 < \ldots t_n < b)$, kutsutaan kuutiospliniksi, jos kaksi (KSPL) ehtoa toteutuvat:

- Jokaisella osavälillä $(a, t_1), (t_1, t_2), \ldots, (t_n, b)$ funktio g on kolmannen asteen polynomi.
- Funktio g ja sen ensimmäiset ja toiset derivaatat ovat jatkuvia pisteissä t_i ja siten koko välillä $[a, b]$.

Funktiota g kutsutaan *luonnolliseksi kuutiospliniksi LKSPL*, jos (KSPL) ehtojen lisäksi funktion toiset ja kolmannet derivaatat ovat nollia pisteissä a ja b, jolloin funktio on lineaarinen väleillä $[a, t_1]$ ja $[t_n, b]$.

Seuraavaksi esitellään eräs esitys (*The value-second derivative representation*) luonnolliselle kuutiosplinille. Olkoon

$$g_i = g(t_i) \text{ ja } \gamma_i = g''(t_i),\ i = 1, \ldots, n.$$

Määritelmän mukaan funktio on lineaarinen väleillä $[a, t_1]$ ja $[t_n, b]$, jolloin siis pisteissä t_1 ja t_n toiset derivaatat ovat nollia eli $\gamma_1 = \gamma_n = 0$. Olkoon nyt

$$\mathbf{g} = (g_1, \ldots, g_n)' \text{ ja } \boldsymbol{\gamma} = (\gamma_2, \ldots, \gamma_{n-1})'.$$

Huom. $\boldsymbol{\gamma}$ on nyt $n-2$-vektori. Vektoreiden $\mathbf{g}$ ja $\boldsymbol{\gamma}$ avulla voidaan käyrän g arvot laskea missä tahansa pisteessä t, jolloin esimerkiksi piirrettäessä saadaan jatkuva käyrä. Välttämättömät ja riittävät ehdot voidaan esittää matriisien $\boldsymbol{\Delta}$ ja $\boldsymbol{\nabla}$ avulla. Määritellään ensin

$$h_i = t_{i+1} - t_i,\ i = 1, \ldots, n-1$$

ja sitten $n \times (n-2)$ matriisi $\boldsymbol{\nabla}$ siten, että sen nollasta poikkeavat alkiot ovat

$$\nabla_{j-1,j} = h_{j-1}^{-1},\ \nabla_{jj} = -h_{j-1}^{-1} - h_j^{-1} \text{ ja } \nabla_{j+1,j} = h_j^{-1},$$

kun $j = 2, \ldots, n-1$. Huomaa, että matriisin alkiot on nyt numeroitu hieman epästandartilla tavalla siten, että alkio $\nabla_{1,2}$ on matriisin $\boldsymbol{\nabla}$ vasen yläkulma. Nyt esimerkiksi jos, $\mathbf{t} = (1, 2, 3, 4, 5)'$, niin $h_i = 1$, kun $i = 1, \ldots, 4$ ja

$$\nabla_{1,2} = 1,\ \nabla_{2,2} = -2 \text{ ja } \nabla_{3,2} = 1,$$

$$\nabla_{2,3} = 1,\ \nabla_{3,3} = -2 \text{ ja } \nabla_{4,3} = 1,$$

$$\nabla_{3,4} = 1,\ \nabla_{4,4} = -2 \text{ ja } \nabla_{5,4} = 1.$$

Tästä saadaan matriisi

$$\boldsymbol{\nabla} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}.$$

Vastaavasti määritellään symmetrisen $(n-2) \times (n-2)$ matriisin $\boldsymbol{\Delta}$ nollasta poikkeavat alkiot

$$\delta_{ii} = (1/3)(h_{i-1} + h_i),\ i = 2, \ldots, n-1$$

ja

$$\delta_{i,i+1} = \delta_{i+1,i} = (1/6)h_i,\ i = 2, \ldots, n-2.$$

Jos $\mathbf{t} = (1, 2, 3, 4, 5)'$, niin alkioiksi saadaan

$$\delta_{2,2} = \delta_{3,3} = \delta_{4,4} = 2/3$$

ja

$$\delta_{2,3} = \delta_{3,2} = \delta_{3,4} = \delta_{4,3} = 1/6.$$

Matriisi $\boldsymbol{\Delta}$ on nyt

$$\boldsymbol{\Delta} = \begin{pmatrix} 2/3 & 1/6 & 0 \\ 1/6 & 2/3 & 1/6 \\ 0 & 1/6 & 2/3 \end{pmatrix}.$$

Voidaan näyttää, että $\mathbf{\Delta}$ on positiivisesti definiitti. Määritellään seuraavaksi rosoisuusmatriisi *(roughness matrix)*

$$\mathbf{K} = \mathbf{\nabla}\mathbf{\Delta}^{-1}\mathbf{\nabla},'$$

jolla on tärkeä rooli laskennallisissa tarkasteluissa.

Lause 1. Vektorit $\mathbf{g}$ ja $\boldsymbol{\gamma}$ määrittelevät luonnollisen kuutiosplinin *jos ja vain jos*

$$\mathbf{\nabla}'\mathbf{g} = \mathbf{\Delta}\boldsymbol{\gamma}. \qquad (SPL)$$

Jos ehto (SPL) toteutuu, niin

$$RP = \int_a^b g''(t)^2 dt = \boldsymbol{\gamma}'\mathbf{\Delta}\boldsymbol{\gamma} = \mathbf{g}'\mathbf{K}\mathbf{g}.$$

Luonnollisella kuutiosplinillä on on tiettyjä hyviä ominaisuuksia interpoloitaessa. Olkoon $S[a,b]$ kaikkien sellaisten funktioiden g avaruus, jotka ovat kaksi kertaa derivoituvia. Jos nyt halutaan tasaisin mahdollinen käyrä joka interpoloi aineiston, niin voisimme valita koko käyrästöstä sen, joka tuottaa pienimmän arvon termille $\int_a^b g''(t)^2 dt$. Osoittautuu, että kaikkien käyrien g avaruudessa $S[a,b]$ kriteerin $\int_a^b g''(t)^2 dt$ minimoiva käyrä on luonnollinen kuutiosplini solmukohtina t_i. Lisäksi, jos $n \geq 2$, on kyseinen luonnollinen kuutiosplini yksikäsitteinen.

Lause 2. Jos $n \geq 2$ ja $t_1 < \cdots < t_n$, niin mille tahansa arvoille $z_1, \ldots, z_n$ on olemassa luonnollinen kuutiosplini solmukohtina t_i siten, että

$$g(t_i) = z_i, i = 1, \ldots, n.$$

Todistus. Jos nyt $\mathbf{g}$ on sellainen käyrä, että $\mathbf{g} = \mathbf{z}$. Lauseen 1 mukaan sellainen luonnollinen kuutiosplini on olemassa, jos löytyy $\boldsymbol{\gamma}$ siten, että $\mathbf{\nabla}'\mathbf{g} = \mathbf{\Delta}\boldsymbol{\gamma}$. Koska $\mathbf{\Delta}$ on positiivisesti definiitti, on olemassa yksikäsitteinen $\boldsymbol{\gamma} = \mathbf{\Delta}^{-1}\mathbf{\nabla}'\mathbf{g}$, joka toteuttaa annetun ehdon.

Lauseen 2 tulos on yleistettävissä ns. Sobolevin avaruuteen $S_2[a,b]$, jossa g on jatkuva ja derivoituva sekä $\int_a^b g''(t)^2 dt < \infty$.

Lause 3. Olkoon $n \geq 2$ ja g luonnollinen kuutiosplini solmukohdissa $t_1, \ldots, t_n$. Olkoon nyt $\tilde{g}$ mikä tahansa funktio avaruudessa S_2, missä $\tilde{g}(t_i) = z_i$, niin $\int \tilde{g}''(t)^2 dt \geq \int g''(t)^2 dt$. Yhtäsuuruus toteutuu vain, kun $\tilde{g}$ ja g ovat identtisiä.

Olkoon g mikä tahansa funktio Sobolevin avaruudessa $S_2[a,b]$. Tarkastellaan seuraavaksi minimointiongelmaa

$$PLS(g) = \sum_i [y_i - g(t)]^2 + \lambda \int [g''(t)]^2 dt,$$

missä λ on annettu positiivinen vakio. Voidaan osoittaa, että $\hat{g}$ on luonnollinen kuutiosplini solmukohtina $a < t_1 < \ldots t_n < b$.

Olkoon g mikä tahansa käyrä, joka ei ole luonnollinen kuutiosplini solmukohtina t_i ja olkoon $\tilde{g}$ luonnollinen interpoloiva kuutiosplini ($g = \tilde{g}$). Ilmeisesti nyt $\sum\{y_i - \tilde{g}(t_i)\}^2 = \sum\{y_i - g(t_i)\}^2$ ja lauseen 3 perusteella $\int \tilde{g}''(t)^2 dt \leq \int g''(t)^2 dt$. Koska $\lambda > 0$, niin $PLS(\tilde{g}) < PLS(g)$. Jos siis käyrä ei itse ole luonnollinen kuutiosplini, niin on aina löydettävissä kuutiosplini joka tuottaa pienemmän PLS arvon. Tästä seuraa se, että kriteerin PLS minimoiva funktio on luonnollinen kuutiosplini. Matriisimerkinnöin

$$PLS(\mathbf{g}) = (\mathbf{y} - \mathbf{g})'(\mathbf{y} - \mathbf{g}) + \lambda \mathbf{g}' \mathbf{K} \mathbf{g}$$

ja yksikäsitteinen minimi saavutetaan, kun

$$\hat{\mathbf{g}} = (\mathbf{I} + \lambda \mathbf{K})^{-1} \mathbf{y},$$

mikä siis on luonnollinen kuutiosplini laskettuna solmukohdissa $t_1, \ldots, t_n$.

5.1 Luonnollisen kuutiosplinin kantafunktioesitys

Lähtökohtana voidaan pitää kolmannen asteen katkaistua polynomikantaa

$$1, t, t^3, (t-\kappa_1)^3_+, \ldots, (t-\kappa_K)^3_+$$

jota redusoidaan rajoitteilla, jotka seuraavat luonnollisen kuutiosplinin määritelmästä. Kantafunktioiksi saadaan

$$f_1(t) = 1,\ f_2(t) = t,\ f_{k+2}(t) = d_k(t) - d_{K-1}(t),$$

missä

$$d_k(t) = \frac{(t-\kappa_k)^3_+ - (t-\kappa_K)^3_+}{\kappa_K - \kappa_k}.$$

Olkoon nyt tarvittavat kantafunktiot matriisissa $\mathbf{X}$, jolloin

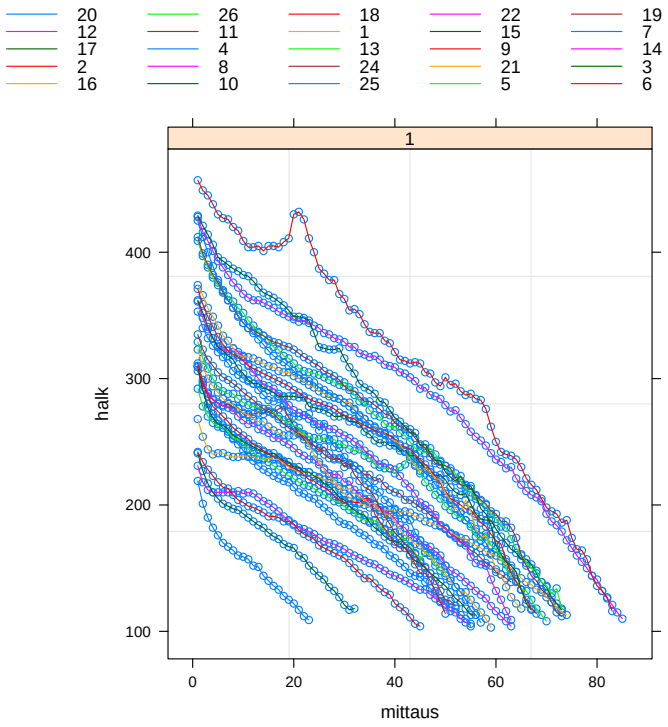
$$PLS(g) = (\mathbf{y} - \mathbf{X}\boldsymbol{\theta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\theta}) + \lambda\boldsymbol{\theta}'\boldsymbol{\Omega}\boldsymbol{\theta},$$

missä $\{\boldsymbol{\Omega}\}_{jk} = \int f_j''(t) f_k''(t) dt$, ja

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}'\mathbf{X} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{X}'\mathbf{y}$$

sekä

$$\hat{g}(t) = \sum_j f_j(t)\hat{\theta}_j.$$



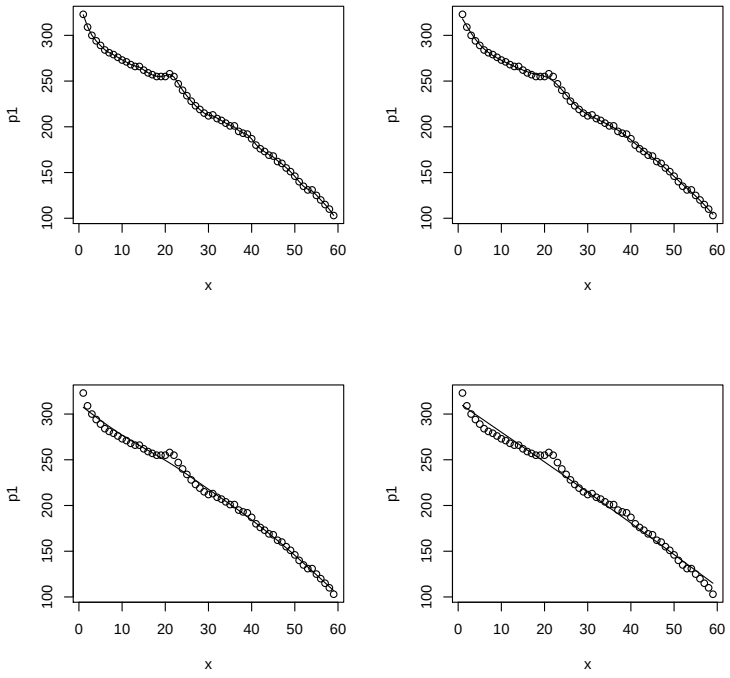
Kuva 21: Käsin mitattujen mäntyjen muodostama aineisto.

Tarkastellaan oheista männyn runkokäyräaineistoa. Aineistossa on mitattu 26 männyn rungon halkaisijat 30 senttimetrin välein. Mittaukset on suoritettu manuaalisesti.

```
> manty<-read.table("F:\\aineistot\\mvirhedata\\MANU.TXT")
> names(manty)<-c("puu", "mittaus", "halkaisija")
> library(nlme)
> puut<-groupedData(halk~mittaus|puu, data=manty)
> plot(manty, outer=~1)
```

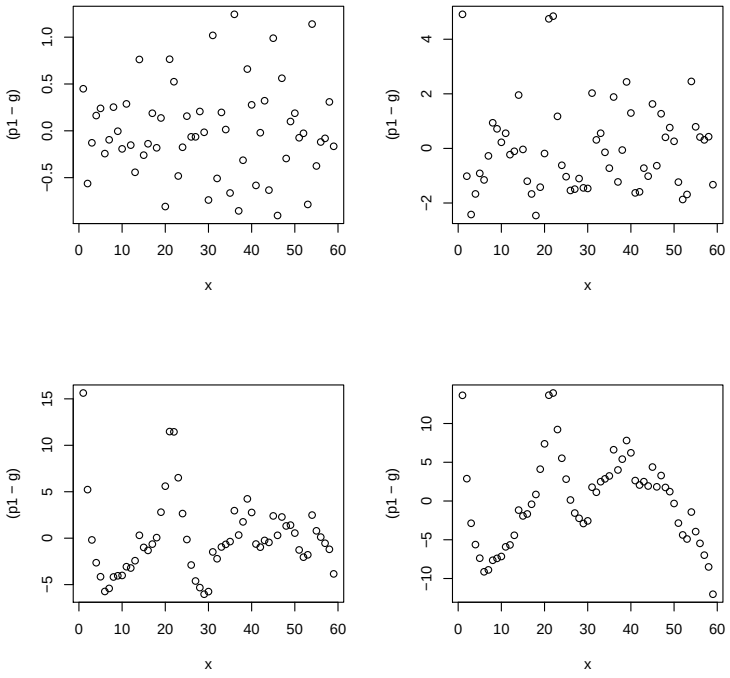
Seuraavaksi 1. puun runkokäyrään on sovitettu luonnollinen tasoitettu kuutiosplini eri tasoituskertoimen λ arvoilla.

```
> p1<-manty[manty$puu==1, 3]
> length(p1)
[1] 59
> x<-1:59
> plot(p1~x)
> n<-nabla(x); d<-delta(x); k<-n%*%solve(d)%*%t(n)
> nabla<-function(x){
h<-diff(x); n<-length(x); m<-matrix(nr=n, nc=(n-2), data=c(0))
for (i in 1:(n-2)) {m[i,i]<-1/h[i]}
for (i in 1:(n-2)) {m[i+1,i]<-((1/h[i])+(1/h[i+1]))*-1}
for (i in 1:(n-2)) {m[i+2,i]<-1/h[i+1]}
m}
> delta<-function(x){
h<-diff(x); n<-length(x); m<-matrix(nr=n-2,nc=n-2, data=c(0))
for (i in 1:(n-2)) {m[i,i]<-(h[i]+h[i+1])/3}
for (i in 1:(n-3)) {m[i,(i+1)]<-h[i+1]/6}
for (i in 1:(n-3)) {m[(i+1),i]<-h[i+1]/6}
m}
```



Kuva 22: Männyn runkokäyrään sovitettu tasoitettu kuutiosplini tasoituskertoimen arvoilla λ=0.1, 10, 1000 ja 100000.

```
> par(mfrow=c(2,2))
> l<-0.1; g<-solve(diag(59)+l*k)%*%p1
> plot(p1~x); lines(x,g)
> l<-10; g<-solve(diag(59)+l*k)%*%p1
> plot(p1~x); lines(x,g)
> l<-1000; g<-solve(diag(59)+l*k)%*%p1
> plot(p1~x); lines(x,g)
> l<-100000; g<-solve(diag(59)+l*k)%*%p1
> plot(p1~x); lines(x,g)
> par(mfrow=c(1,1))
```



Kuva 23: Männyn runkokäyrään sovitettun tasoitetun kuutiosplinin residuaalit tasoituskertoimen arvoilla λ=0.1, 10, 1000 ja 100000.

5.2 Yhteys sekamalleihin

Voidaan osoittaa (ks. 3.6.1), että

$$(\mathbf{I}+\mathbf{K})^{-1} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' + \mathbf{Z}(\boldsymbol{\Delta}^{-1} + \mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}',$$

missä $\mathbf{Z} = \boldsymbol{\nabla}(\boldsymbol{\nabla}'\boldsymbol{\nabla})^{-1}$ ja $\mathbf{X} = [\mathbf{1}, (1, \ldots, n)']$. Jolloin siis

$$\hat{\mathbf{g}} = (\mathbf{I}+\lambda\mathbf{K})^{-1}\mathbf{y} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{u}},$$

missä

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

ja

$$\hat{\mathbf{u}} = (\mathbf{Z}'\mathbf{Z} + \lambda\boldsymbol{\Delta}^{-1})^{-1}\mathbf{Z}'\mathbf{y},$$

missä

$$\lambda = \frac{\sigma^2}{\sigma_u^2}.$$

Ratkaisu on siis esitettävissä sekamallin

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\epsilon},$$

missä

$$Var\begin{pmatrix}\mathbf{u}\\ \boldsymbol{\epsilon}\end{pmatrix} = \begin{pmatrix}\sigma_u^2\boldsymbol{\Delta} & \mathbf{0}\\ \mathbf{0} & \sigma^2\mathbf{I}\end{pmatrix}$$

BLUP ratkaisuna. Sekamalli voidaan edelleen kirjoittaa muodossa

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}_*\mathbf{u}_* + \boldsymbol{\epsilon},$$

missä $\mathbf{Z}_* = \mathbf{Z}\boldsymbol{\Delta}^{1/2}$ ja $\boldsymbol{u}_* = \boldsymbol{\Delta}^{-1/2}\mathbf{u}$ missä

$$Var\begin{pmatrix}\mathbf{u}_*\\ \boldsymbol{\epsilon}\end{pmatrix} = \begin{pmatrix}\sigma_u^2\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \sigma^2\mathbf{I}\end{pmatrix},$$

jolloin estimointi voidaan suorittaa helposti esimerkiksi R-ohjelmistolla.

5.3 Tasoituskertoimen valinta

Tasoituskertoimen valinnassa voidaan nähdä kaksi päälinjaa: Joko valita tasoituskerroin täysin vapaasti tai sitten käyttää ns. automaattisia menetelmiä. Ensin mainitussa valinta voidaan perustaa esimerkiksi aikaisempaan kokemukseen tai sitten vaikkapa visuaalisiin tarkasteluihin siitä miltä sovite näyttää eri tasoituskertoimien arvoilla, kun sitä verrataan käytettyyn aineistoon. Moniin käytännön tilanteisiin tästä saadaankin varsin käyttökelpoinen tasoituskertoimen arvo. Päähuomio tässä esityksessä kohdistetaan kuitenkin automaattisiin menetelmiin.

5.3.1 Uskottavuusfunktioon perustuva menetelmä

Kirjoitetaan Splini-ratkaisu sekamallin

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}_*\mathbf{u}_* + \boldsymbol{\epsilon},$$

missä riippumattomat

$$\mathbf{u}_* \sim N(\mathbf{0}, \sigma_u^2\mathbf{I}) \quad \text{ja} \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$$

ratkaisuna. Nyt siis

$$Var(\mathbf{y}) = \sigma_u^2\mathbf{Z}_*\mathbf{Z}_*' + \sigma^2\mathbf{I}$$

jolloin tasoituskerroin olisi

$$\lambda = \frac{\sigma^2}{\sigma_u^2}.$$

Estimoidaan seuraavaksi suurimman uskottavuuden menetelmällä (tai REML-menetelmällä) parametrit σ^2 ja σ_u^2, jolloin saadaan

$$\hat{\lambda} = \frac{\hat{\sigma}^2}{\hat{\sigma}_u^2}.$$

5.3.2 Ristiinvalidointi

Tämä menetelmä perustuu suureen

$$CV(\lambda) = n^{-1}\sum(y_i - \hat{g}^{(-i)}(t_i\lambda))^2,$$

missä $\hat{g}^{(-i)}(t_i\lambda)$ on sovite pisteessä t_i ilman havaintoa (y_i, t_i), minimointiin. Voidaan osoittaa, että kriteerille on vaihtoehtoinen laskennallisesti kätevämpi esitysmuoto

$$CV(\lambda) = n^{-1}\sum_{i=1}^{n}\left(\frac{y_i - \hat{g}(t_i;\lambda)}{1 - S_{i,i}(\lambda)}\right)^2,$$

missä alkiot $S_{i,i}(\lambda)$ muodostuvat tasoittajamatriisin

$$\mathbf{S}(\lambda) = (\mathbf{I} + \lambda\mathbf{K})^{-1}$$

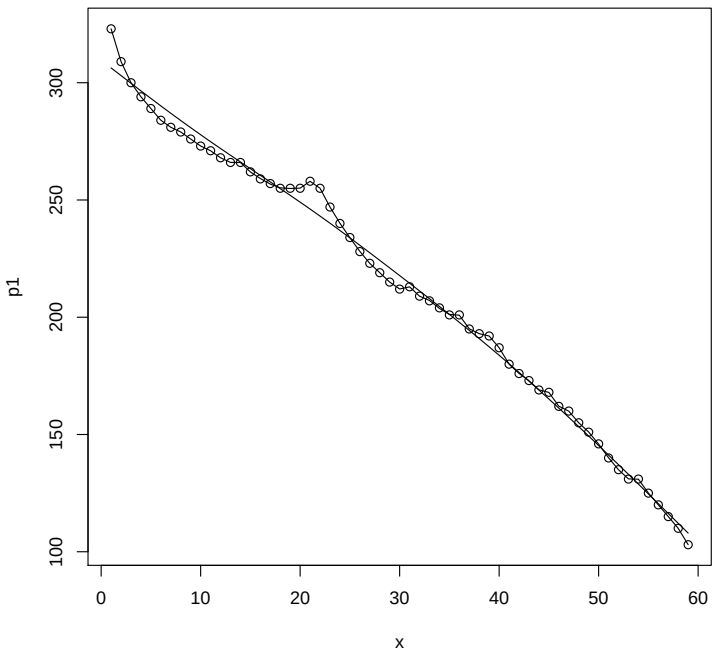
diagonaalialkioista. Yleistetty ristiinvalidointi-kriteeri saadaan, jos korvataan alkiot $S_{i,i}(\lambda)$ niiden keskiarvolla. Tästä saadaan

$$GCV(\lambda) = n^{-1} \sum_{i=1}^{n} \left(\frac{y_i - \hat{g}(t_i; \lambda)}{1 - n^{-1}\ tr\mathbf{S}(\lambda)} \right)^2 .$$

Luonnollisesti myös monia muita menetelmiä, kuten esimerkiksi informaatiokriteereihin perustuvia, voidaan käyttää.

Seuraavassa esimerkissä on Mänty-aineistoon sovitettu luonnollinen tasoitettu kuutiosplini R-ohjelmiston funktiolla *smooth.spline*.

```
> manty<-read.table("O:\\eparam\\Aineistoja\\MANU.TXT")
> names(manty)<-c("puu", "mittaus", "halkaisija")
> p1<-manty[manty$puu==1, 3]
> x<-1:59
> l<-smooth.spline(x, p1)
> plot(p1~x); lines(l)
> l
...
Smoothing Parameter  spar= 0.02673929  lambda= 8.052738e-09 (17 iterations)
Equivalent Degrees of Freedom (Df): 52.09729
Penalized Criterion: 0.3532997
GCV: 0.4374778
> l2<-smooth.spline(x, p1, df=4)
> lines(l2)
> l2
...
Smoothing Parameter  spar= 0.8810029  lambda= 0.01196076 (13 iterations)
Equivalent Degrees of Freedom (Df): 3.999378
Penalized Criterion: 1157.302
GCV: 22.57166
```



Kuva 24: Männyn runkokäyrään sovitettu tasoitettu kuutiosplini GCV-kriteerin perusteella sekä kiinnittämällä *df=4*.

5.4 Painotettu tasoitus

Joissakin tilanteissa voi olla mielekästä tarkastella painotettua kriteeriä

$$WPLS = \sum_{i=1}^{n} w_i\{y_i - g(t_i)\}^2 + \lambda \int g''^2,$$

missä painokertoimet $w_1, \ldots, w_n$ ovat kaikki positiivisia. Näin esimerkiksi silloin, kun havaintojen y_i varianssit eivät ole yhtäsuuria, jolloin luonnolinen valinta painokertoimeksi olisi varianssin käänteisluku. Olkoon nyt $\mathbf{W} = diag(w_1, w_2, \ldots, w_n)$. Voidaan osoittaa, että WPLS-funktion minimoiva luonnollinen kuutiosplini on nyt

$$\hat{\mathbf{g}} = (\mathbf{W} + \lambda\mathbf{K})^{-1}\mathbf{W}\mathbf{y}.$$

Voidaan lisäksi osoittaa, että

$$\mathbf{I} - \mathbf{S}_W(\lambda) = \lambda \mathbf{W}^{-1} \nabla (\mathbf{\Delta} + \lambda \nabla' \mathbf{W}^{-1} \nabla)^{-1} \nabla',$$

jolloin CV-kriteeriksi saadaan

$$CV(\lambda) = \sum_{i=1}^{n} w_i \left(\frac{y_i - \hat{g}(t_i)}{\{\mathbf{I} - \mathbf{S}_W(\lambda)\}_{ii}} \right)^2.$$

Eräs tärkeä sovellus painotetulle menetelmälle on tilanne, jossa pisteissä t_i saadaan useampi (m_i kpl) havaintoarvo. Olkoon nyt

$$\bar{y}_i = m_i^{-1} \sum_{j=1}^{m_i} y_{ij}.$$

Voidaan osoittaa, että painotettu kriteerifunktio on nyt muotoa

$$WPLS = \sum_{i=1}^{n} m_i \{\bar{y}_i - g(t_i)\}^2 + \lambda \int g''^2.$$

Olkoon $\mathbf{N}$ indikaattorimatriisi siten, että alkio $\{N\}_{ji}$ on 1, jos arvo on havaittu, muutoin nolla. Nyt saadaan

$$\hat{\mathbf{g}} = (\mathbf{N}'\mathbf{N} + \lambda \mathbf{K})^{-1} \mathbf{N}'\mathbf{y} \quad \text{ja} \quad \hat{\mathbf{y}} = \mathbf{N}\hat{\mathbf{g}}.$$

Jos nyt tasoituskertoimen valinta halutaan suorittaa tavallisella CV-menetelmällä, niin käytetään modifioitua kriteeriä

$$CV(\lambda) = N^{-1} \sum_{i=1}^{n} m_i \frac{[y_i - \hat{g}(t_i)]^2 + s_i^2}{[(\mathbf{I} - m_i^{-1} \mathbf{S}_W(\lambda))_{ii}]^2},$$

missä

$$s_i^2 = \sum_{j=1}^{m_i} (y_{ij} - \bar{y}_i)^2.$$

Seuraavassa esimerkissä tarkastellaan aineistoon *airquality* sovitettua yleistettyä ei-parametrista regressiomallia

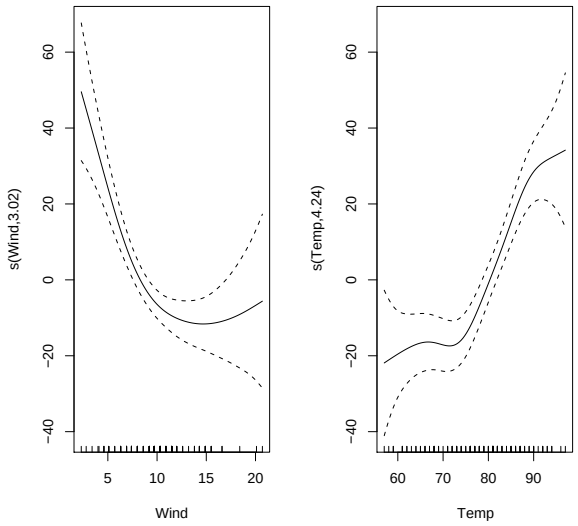
$$g = \alpha + g_1 + \cdots + g_k,$$

missä funktiot g_j voivat olla peräisin jostakin ei-parametrisesta sovitteesta kuten lokaalisista polynomeista tai tasoittavista splineistä.

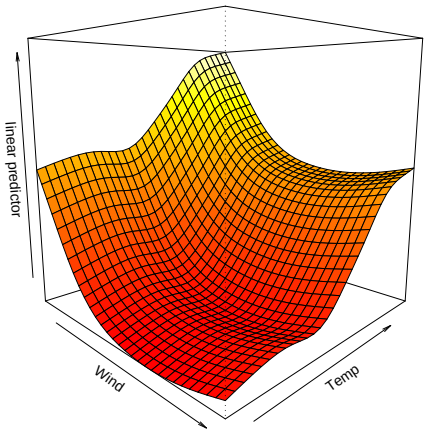
```
> library(mgcv)
> data(airquality)
> attach(airquality)
> gam1<-gam(Ozone~s(Wind)+s(Temp), na.action=na.omit)
Warning message:
In any(G$offset) : coercing argument of type 'double' to logical
> par(mfrow=c(1,2))
> plot(gam1)
Press return for next page....
> vis.gam(gam1, theta=45, phi=10)
> gam1

Family: gaussian
Link function: identity

Formula:
Ozone ~ s(Wind) + s(Temp)

Estimated degrees of freedom:
 3.016882 4.236479   total =  8.25336

GCV score:  375.7313
```

Seuraavaksi testataan malli, jossa muuttujan *Temp* vaikutus oletetaan lineaariseksi.

```
> gam2<-gam(Ozone~s(Wind)+Temp, na.action=na.omit)
Warning message:
In any(G$offset) : coercing argument of type 'double' to logical
> summary(gam2)

Family: gaussian
Link function: identity

Formula:
Ozone ~ s(Wind) + Temp

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -86.0352    17.9908  -4.782 5.38e-06 ***
Temp          1.6459     0.2298   7.161 9.31e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
          edf Est.rank     F  p-value
s(Wind) 3.211        7 7.678 1.52e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.642   Deviance explained = 65.5%
GCV score = 407.79   Scale est. = 389.47    n = 116
```



Kuva 25: Muuttujien *Wind* ja *Temp* vaikutus vasteeseen.



Kuva 26: Mallin antama sovite muuttujalle *Ozone*.

```
> anova(gam2, gam1, test="F")
Analysis of Deviance Table

Model 1: Ozone ~ s(Wind) + Temp
Model 2: Ozone ~ s(Wind) + s(Temp)
  Resid. Df Resid. Dev       Df Deviance      F   Pr(>F)
1  110.7890      43149
2  107.7466      37603   3.0424     5546 5.2233 0.001996 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Tässä siis malli, jossa oletetaan, että muuttujan *Temp* vaikutus vasteeseen olisi lineaarinen, tulee selvästi hylätyksi.