# Inversio-ongelmien laskennallinen peruskurssi
# Luento 6

Kevät 2012

## 1 Iteratiivisista menetelmistä

Tähän mennessä on tarkasteltu niin sanottuja suoria menetelmiä, joissa (likimääräinen) ratkaisu saadaan kerralla esimerkiksi pseudoinverssiä käyttämällä. Suorat menetelmät kuitenkin tavallisesti vaativat matriisien käänteismatriisien laskemista, mikä on suurien ongelmien tapauksissa numeerisesti raskasta.

Iteratiivisissa menetelmissä sen sijaan tavallisesti konstruoidaan jono likimääräisiä ratkaisuja, jotka (toivon mukaan) konvergoivat kohti jonkinlaista hyväksyttävää ratkaisua. Lisäksi iteratiiviset menetelmät eivät useimmiten vaadi matriisien kääntämistä, jolloin ne saattavat sopia paremmin isoille ongelmille.

## 2 Konjugaattigradienttimenetelmä (CG)

Tarkastellaan ongelmaa

$$m = Ax.$$

Oletetaan, että matriisi $A$ on ns. SPD, eli symmetrinen ja positiividefiniitti. Toisin sanoen, matriisille $A$ pätee:

1. $A^T = A$ (symmetrisyys),
2. kaikilla $x \in \mathbb{R}^n$, $x \neq 0$, pätee $x^T Ax > 0$ (positiividefiniittisyys)

Jos matriisi $A$ on symmetrinen, on sen singulaariarvohajotelma muotoa

$$A = U\Sigma U^T.$$

Tällöin yo. hajotelma on myös matriisin $A$ ominaisarvohajotelma: jos $U = [u_1, u_2, \ldots, u_n]$, niin

$$Au_j = \sigma_j u_j,$$

ja erityisesti, jos $A$ on positiividefiniitti, niin

$$\sigma_j = u_j^T \sigma_j u_j = u_j^T A u_j > 0,$$

eli matriisin $A$ kaikki ominaisarvot ovat positiivisia.

Matriisi $A$ määrittelee ns. $A$-normin kaavalla

$$||x||_A^2 = x^T A x.$$

Koska $\{u_1, u_2, \ldots, u_n\}$ muodostavat ortogonaalisen kannan avaruudessa $\mathbb{R}^n$, voidaan kirjoittaa

$$x = \sum_{j=1}^{n} \tilde{x}_j u_j, \quad \tilde{x}_j = u_j^T x = \langle u_j, x \rangle.$$

Tällöin

$$\begin{aligned} x^T A x &= \left( \sum_{j=1}^{n} \tilde{x}_j u_j \right)^T \left( \sum_{k=1}^{n} \tilde{x}_k \sigma_k u_k \right) \\ &= \sum_{j=1}^{n} \sigma_j \tilde{x}_j^2 = ||x||_A^2, \end{aligned}$$

joten $|| \cdot ||_A$ on painotettu euklidinen normi $\{u_1, u_2, \ldots, u_n\}$ koordinaattisysteemissä.

Olkoon $x_* \in \mathbb{R}^n$ ongelman $m = Ax$ "oikea"ratkaisu, eli $x_* = A^{-1}m$. Kaikille $x \in \mathbb{R}^n$ määritellään

$$e = x - x_* \quad \text{(virhe)},$$

$$r = Ae = Ax - Ax_* = Ax - m \quad \text{(residuaali)}.$$

Virheen $e$ "koon"mittaamiseksi määritellään

$$\phi(x) = ||e||_A^2 = e^T A e = e^T r = r^T A^{-1} r.$$

Koska "oikea"ratkaisu $x_*$ on tuntematon, ei funkionaalia $\phi(x)$ voida laskea. Huomaa, että

$$\phi(x) \geq 0, \quad \phi(x) = 0 \Leftrightarrow x = x_*.$$

Siis minimoimalla $\phi$ voitaisiin saada oikeaa ratkaisua $x_*$ lähellä oleva likimääräinen ratkaisu. Vaikka funktionaalin $\phi$ arvoa ei voida suoraan laskea, on se kuitenkin mahdollista minimoida iteratiivisesti.

Konjugaattigradienttimenetelmä on yksi vanhimmista ns. Krylovin aliavaruus -menetelmistä. Olkoon $x_1 \in \mathbb{R}^n$ (usein voidaan valita $x_1 = 0$) ensimmäinen arvaus ratkaisuksi. Tällöin ensimmäinen residuaali on

$$r_1 = m - Ax_1,$$

ja oletetaan, että $r_1 \neq 0$. Niin sanottu $k$. asteen Krylovin aliavaruus on

$$K_k(r_1, A) = \mathrm{sp}\{r_1, Ar_1, A^2 r_1, \ldots, A^{k-1} r_1\}.$$

Krylovin aliavaruus -menetelmissä ajatuksena on yrittää löytää likimääräsiratkaisu aliavaruudesta $K_k(r_1, A)$.

Olkoon alkuarvaus $x_1$ ja ensimmäinen residuaali $r_1$ kuten edellä. Määritellään ensimmäinen hakusuunta $s_1$,

$$s_1 = r_1,$$

ja yritetään minimoida funktio $\mathbb{R} \to \mathbb{R}$,

$$\alpha \mapsto \phi(x_1 + \alpha s_1).$$

Nyt

$$\begin{aligned}
\phi(x_1 + \alpha s_1) &= (x_1 + \alpha s_1 - x_*)^T A (x_1 + \alpha s_1 - x_*) \\
&= \alpha^2 s_1^T A s_1 + 2\alpha s_1^T A (x_1 - x_*) + (x_1 - x_*)^T A (x_1 - x_*) \\
&= \alpha^2 s_1^T A s_1 - 2\alpha s_1^T r_1 + \phi(x_1),
\end{aligned}$$

ja minimi saavutetaan derivaatan nollakohdassa

$$2\alpha s_1^T A s_1 - 2 s_1^T r_1 = 0$$

eli

$$\alpha = \frac{s_1^T r_1}{s_1^T A s_1}.$$

Määritellään nyt ratkaisun uusi iteraatio $x_2$,

$$x_2 = x_1 + \alpha s_1,$$

ja uusi residuaali $r_2$,

$$r_2 = m - Ax_2 = r_1 - \alpha A s_1.$$

Huomaa, että

$$s_1^T r_2 = s_1^T r_1 - \alpha s_1^T A s_1 = 0,$$

eli vektorit $s_1$ ja $r_2$ ovat kohtisuorassa toisiaan vastaan.

Tämän jälkeen jatketaan kuten edellä: valitaan uusi hakusuunta $s_2$ ja minimoidaan funktio

$$\alpha \mapsto \phi(x_2 + \alpha s_2).$$

Yleisesti iteraatiokierroksella $k = 1, 2, \ldots$,

$$\begin{aligned} \alpha_k &= \arg\min \phi(x_k + \alpha s_k) \\ &= \frac{s_k^T r_k}{s_k^T A s_k} \end{aligned}$$

Ongelmana on, miten tulisi valita hakusuunta $s_k$?

**Määritelmä 2.1.** Olkoon $\{s_1, s_2, \ldots, s_k\}$ joukko lineaarisesti riippumattomia vektoreita. Niitä sanotaan $A$-konjugaateiksi, jos

$$s_j^T A s_l = 0, \quad \text{kun } j \neq l.$$

Ideana on valita hakusuunnat siten, että

1. $\{s_1, s_2, \ldots, s_k\}$ ovat $A$-konjugaatteja, ja
2. $\text{sp}\{s_1, s_2, \ldots, s_k\} = K_k(r_1, A)$.

Nyt

$$s_1 = r_1,$$

eli

$$\text{sp}\{s_1\} = K_1(r_1, A).$$

Oletetaan, että olemme valinneet $A$-konjugaatit $\{s_1, s_2, \ldots, s_k\}$, ja että

$$\text{sp}\{s_1, s_2, \ldots, s_k\} = K_k(r_1, A).$$

Olkoon

$$r_{k+1} = m - Ax_{k+1},$$

ja oletetaan että $r_{k+1} \neq 0$. Etsitään uutta hakusuuntaa $s_{k+1}$ muodossa

$$s_{k+1} = r_{k+1} + \beta s_k.$$

Jotta $A$-konjugaattisuusehto toteutuisi, vaaditaan

$$s_k^T A s_{k+1} = s_k^T A r_{k+1} + \beta s_k^T A s_k = 0,$$

eli

$$\beta = -\frac{s_k^T A r_{k+1}}{s_k^T A s_k}.$$

Nyt, kun $j < k$,

$$s_j A s_{k+1} = s_j^T A r_{k+1} + \beta s_j^T A s_k = (A s_j)^T r_{k+1},$$

ja

$$A s_j \in A(\mathrm{sp}\{s_1, \ldots, s_{k-1}\}) = A K_{k-1}(r_1, A) \subset K_k(r_1, A) = \mathrm{sp}\{s_1, \ldots, s_k\} \perp r_{k+1},$$

joten

$$s_j^T A s_{k+1} = 0.$$

Täten siis uusin hakusuunta $s_{k+1}$ on $A$-konjugaatti vektoreille $s_1, s_2, \ldots, s_k$ ja

$$\mathrm{sp}\{s_1, \ldots, s_{k+1}\} = K_{k+1}(r_1, A).$$

Huomioi vielä, että

$$\begin{aligned} s_k^T r_k &= (r_k + \beta_{k-1} s_{k-1})^T r_k \\ &= r_k^T r_k + \beta_{k-1} s_{k-1}^T r_k \\ &= ||r_k||^2, \end{aligned}$$

sillä $s_k^T r_k = 0$, joten

$$\alpha_k = \frac{s_k^T r_k}{s_k^T A s_k} = \frac{||r_k||^2}{s_k^T A s_k}.$$

Lisäksi

$$\begin{aligned} ||r_{k+1}||^2 &= r_{k+1}^T r_{k+1} \\ &= r_{k+1}^T (m - A x_{k+1}) \\ &= r_{k+1}^T (m - A x_k - \alpha_k A s_k) \\ &= r_{k+1}^T r_k - \alpha_k r_{k+1}^T A s_k \\ &= -\alpha_k r_{k+1}^T A s_k, \end{aligned}$$

koska $r_k \in K_k(r_1, A) = \mathrm{sp}\{s_1, \ldots, s_k\} \perp r_{k+1}$. Täten

$$||r_{k+1}||^2 = -\frac{||r_k||^2}{s_k A s_k} r_{k+1}^T A s_k = ||r_k||^2 \beta_k,$$

joten

$$\beta = \frac{||r_{k+1}||^2}{||r_k||^2}.$$

Kootaan nyt kaikki edellä oleva iteratiiviseksi algoritmiksi:

### CG-algoritmi

1. $k = 1$: Valitse $x_1$ (esim $x_1 = 0$), $r_1 = m - Ax_1$, $s_1 = r_1$

2. Iteroi, kunnes lopetusehto täyttyy:

$$\alpha_k = \frac{||r_k||^2}{s_k^T A s_k};$$

$$x_{k+1} = x_k + \alpha_k s_k;$$

$$r_{k+1} = r_k - \alpha_k A s_k;$$

$$\beta_k = \frac{||r_{k+1}||^2}{||r_k||^2};$$

$$s_{k+1} = r_{k+1} + \beta_k s_k.$$

Lopetusehtona voi käyttää esimerkiksi Morozovin diskrepanssiperiaatetta, eli iterointi lopetetaan, kun

$$||r_k|| \leq ||\varepsilon||,$$

missä $||\varepsilon||$ on (arvioitu) mittausvirheen normi. Lisäksi kannattaa myös asettaa iteroinneille jokin yläraja, mikä estää iterointisilmukan joutumisen äärettömään looppiin, jos Morozovin diskrepanssiehto ei täyty.

# 3 Conjugate Gradient for Least Squares (CGLS)

CG-menetelmän rajoituksena on se, että matriisin $A$ tulee olla symmetrinen ja positiividefiniitti. Yleinen (ylideterminoitu) ongelma voidaan kuitenkin palauttaa muotoon, jossa CG-menetelmää voidaan soveltaa.

Tarkastellaan ongelmaa

$$m = Ax,$$

missä $A \in \mathbb{R}^{n \times n}$ ja oletetaan, että $A^T A$ on kääntyvä. Tällöin yo. ongelman pienimmän neliösumman ratkaisu saadaan kaavalla

$$\begin{aligned} \hat{x} &= A^+ m \\ &= (A^T A)^{-1} A^T m, \end{aligned}$$

eli

$$B\hat{x} = b,$$

missä $B = A^T A$ ja $b = A^T m$. Siis ongelman pienimmän neliösumman ratkaisun $\hat{x}$ löytämiseksi voidaan soveltaa CG-algoritmiä yhtälöön

$$B\hat{x} = b.$$

Huomaa, että nyt residuaali

$$\begin{aligned} r_k &= b - Bx_k \\ &= A^T m - A^T A x_k, \end{aligned}$$

joten määritellään erikseen diskrepanssi

$$d_k = m - Ax_k.$$

Tällöin

$$r_k = A^T d_k,$$

ja lisäksi nyt

$$\alpha_k = \frac{||r_k||^2}{s_k^T B s_k} = \frac{||r_k||^2}{||As_k||^2}.$$

Kootaan ylläoleva iteratiiviseksi algoritmiksi:

## CGLS-algoritmi

1. $k = 1$: Valitse $x_1$, $d_1 = m - Ax_1$, $r_1 = A^t d_1$, $y_1 = As_1$.

2. Iteroi, kunnes lopetusehto täyttyy:

$$\alpha_k = \frac{||r_k||^2}{||y_k||^2};$$

$$x_{k+1} = x_k + \alpha_k s_k;$$

$$d_{k+1} = d_k - \alpha_k y_k;$$

$$r_{k+1} = A^T d_{k+1};$$

$$\beta_{k+1} = \frac{||r_{k+1}||^2}{||r_k||^2};$$

$$s_{k+1} = r_{k+1} + \beta_{k+1} s_k;$$

$$y_{k+1} = As_{k+1}.$$