Datasta Tietoon, exercises material, autumn 2011

Datasta tietoon = From data to knowledge.

Contents

Notations:

- data samples or observations, sample $\mathbf{X}$, which contains n pieces of d-dimensional data points. When $d = 1$, then typicall one data point x, whereas for $d > 1$ $\mathbf{x}$. Data matrix $\mathbf{X}$ is written (in this course) so that features are rows and observations as columns

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(1) & \mathbf{x}(2) & \dots & \mathbf{x}(n) \end{bmatrix} = \begin{bmatrix} x_1(1) & x_1(2) & \dots & x_1(n) \\ x_2(1) & x_2(2) & \dots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_d(1) & x_d(2) & \dots & x_d(n) \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & \dots & x_{dn} \end{bmatrix}$$

Example. We measure height and weight of Bill and Bob. Bill is 180 cm tall and 73 kg. Bob 174 and 65, respectively. That is $\mathbf{x}(1) = [180\ 73]^T$ and $\mathbf{x}(2) = [174\ 65]^T$ and as a matrix

$$\mathbf{X} = \begin{bmatrix} 180 & 174 \\ 73 & 65 \end{bmatrix}$$

If you do not have Matlab at your own computer, you can download GNU Octave (Matlab clone, `octave.org`), which is available in Windows-, Mac- and Linux. It is probably easiest to download with extra packages (Octave-Forge) in `octave.sourceforge.net`. Most Matlab commands work in Octave.

Comments and corrections to `t612010@ics.tkk.fi`.

Datasta Tietoon, autumn 2011, esitietotehtäviä

E1. Kombinatoriikkaa ja potensseja. Muista joitakin laskusääntöjä

$$\begin{aligned} x^0 &= 1 \\ x^{B+C} &= x^B \cdot x^C \\ x^{-B} &= 1/x^B \\ (x^B)^C &= x^{B \cdot C} \end{aligned}$$

Huomaa vielä, että $2^{10} = 1024 \approx 10^3$ (k), $2^{20} = 2^{10} \cdot 2^{10} = 1048576 \approx 10^6$ (M), $2^{30} = 2^{10} \cdot 2^{10} \cdot 2^{10} \approx 10^9$ (G), jne.

a) Jokeriarvonnassa arvotaan seitsemän numeroa, joista jokainen voi saada arvon $0, \ldots, 9$. Kuinka monta vaihtoehtoa on olemassa?

b) Tarinan "Shakkilaudan joka ruudulla riisinjyvien lukumäärä kaksinkertaistuu" perusteella esitä luvulle $2^{64} = 2^4 \cdot 2^{60}$ suuruusluokka 10-kantaisessa järjestelmässä.

c) Tutkitaan "DataMatrix"/"UpCode"-tyyppistä 2-ulotteista bittikarttaa, jossa kuvan koko on 2×2 kuvapistettä ($2 \times 2 = 4$ pikseliä) ja kukin pikseli on joko musta (0) tai valkea (1). Kuinka monta erilaista esitystä saadaan? Voit myös piirtää.

d) Tutkitaan "thumbnail"-valokuvaa, jonka koko on $19 \times 19 = 361$ pikseliä, ja jokainen pikseli on esitettynä harmaasävyarvolla 8 bitillä. Vaihtoehtoja yhdelle pikselille on tällöin $2^8 = 256$, jolloin 0 vastaa mustaa ja 255 valkeaa. Kuinka monta erilaista kuvaa voidaan esittää?

Huomaa, että d-kohdassa $19 \cdot 19 \cdot 256 = 92416$ on väärä vastaus. Tulos on suurempi. Reilusti suurempi.

Matlabissa ja Octavessa

```
2^64
log(8)
log2(8)
log10(8)
exp(1)
```

E2. Lasketaan tässä tehtävässä logaritmeja. Muutamia logaritmien laskusääntöjä ja -esimerkkejä ovat

$$\begin{aligned}
\log_A B = C \quad &\Leftrightarrow \quad A^C = B \\
\log(A \cdot B) &= \log(A) + \log(B) \\
\log(A/B) &= \log(A) - \log(B) \\
A \cdot \log(B^C) &= A \cdot C \cdot \log(B) \\
\log_A B &= \log_C B / \log_C A \\
\log_A A^B &= B \log_A A = B
\end{aligned}$$

Logaritmeissa käytetään erilaisia kantalukuja, joista tyypillisimmät ovat 2, e (“luonnollinen logaritmi”) ja 10.

Esimerkki:

$$\log_2(\frac{1}{\sqrt{2}} \cdot 4^4) = \log_2(1) - \log_2((2)^{1/2}) + 4\log_2 4 = 0 - 0.5 + 8 = 7.5$$

Joskus on kivaa esittää jotkut luvut 2-kantaisina tai 10-kantaisina. Esimerkiksi 7^5:

$$\begin{aligned}
7^5 &= 10^x \qquad |\log_{10} \\
5 \cdot \log_{10} 7 &= x \\
x &\approx 4.23 \\
10^{0.23} \cdot 10^4 &\approx 1.7 \cdot 10^4 = 17000
\end{aligned}$$

Taskulaskimissa usein “ln” viittaa luonnolliseen logaritmiin, kirjallisuudessa ln tai $\log_e$ tai log, ja taskulaskimen “log” viittaa usein 10-kantaiseen logaritmiin, log tai $\log_{10}$. Useissa “kaavojen johtamisissa” ei ole väliä, minkä kantainen logaritmi on.

Laske taskulaskimella

a) $S = 0.0012 \cdot 0.00046 \cdot 0.00043$.

b) $T = 0.000073 \cdot 0.0000026 \cdot 0.0000343$.

c) Laske $\log(S)$ ja $\log(T)$. Varmista myös, että $\log(S) = \log(0.0012) + \log(0.00046) + \log(0.00043)$.

d) Muunna edellisen tehtävän 19×19 -kokoisten valokuvien lukumäärä 10-kantaiseksi.

Kuten c-kohdasta huomaa, joskus joutuu “arvaamaan” sopivan kannan. Huomaa myös, että koska $S > T$, niin myös $\log(S) > \log(T)$.

Matlabissa ja Octavessa `log2(4),log(exp(1)),log10(1000)`.

E3. Skalaareja, vektoreita ja matriiseja. Tällä kurssilla skaalari (yksi muuttuja) vaikkapa pituus $x = 186$. Vektoriin voidaan tallettaa useita muuttujia, esimerkiksi pituus ja paino: $\mathbf{x} = [186 \;\; 83]^T$, jossa vaakavektori transponoitu pystyvektoriksi. Viidestä havainnosta (viiden ihmisen pituus ja paino) saadaan sitten 2×5 -matriisi (2 riviä, 5 saraketta)

$$\mathbf{X} = \begin{bmatrix}\mathbf{x}(1) & \mathbf{x}(2) & \mathbf{x}(3) & \mathbf{x}(4) & \mathbf{x}(5)\end{bmatrix} = \begin{bmatrix}186 & 163 & 171 & 1.74 & 190 \\ 83 & 55 & 68 & 80 & 89\end{bmatrix}$$

Matriisikertolaskussa pitää muistaa, että "dimensiot täsmää". Esimerkkejä matriisien kertolaskusta, vakiolla kertominen:

$$4 \cdot \begin{bmatrix}186 & 163 & 171 & 1.74 & 190 \\ 83 & 55 & 68 & 80 & 89\end{bmatrix} = \begin{bmatrix}744 & 732 & 684 & 6.96 & 760 \\ 332 & 220 & 272 & 320 & 356\end{bmatrix}$$

matriisilasku $\mathbf{X}^T\mathbf{b}$, jossa $\mathbf{b} = [0.8 \; 0.2]^T$ ("painotettu keskiarvo")

$$\begin{bmatrix}186 & 83 \\ 163 & 55 \\ 171 & 68 \\ 1.74 & 80 \\ 190 & 89\end{bmatrix} \cdot \begin{bmatrix}0.8 \\ 0.2\end{bmatrix} = \begin{bmatrix}186 \cdot 0.8 + 83 \cdot 0.2 \\ 163 \cdot 0.8 + 55 \cdot 0.2 \\ 171 \cdot 0.8 + 68 \cdot 0.2 \\ 1.74 \cdot 0.8 + 80 \cdot 0.2 \\ 190 \cdot 0.8 + 89 \cdot 0.2\end{bmatrix} = \begin{bmatrix}165.4 \\ 157.4 \\ 150.4 \\ 17.392 \\ 169.8\end{bmatrix}$$

Yllä olevassa esimerkissä dimensiotarkastelu: $(5 \times \underline{2})(\underline{2} \times 1) = (5 \times 1)$. Vielä kolmantena kahden matriisin kertolasku $\mathbf{X}\mathbf{X}^T$, josta tulee 2×2 -matriisi dimensiotarkastelulla:

$$\begin{bmatrix}186 & 163 & 171 & 1.74 & 190 \\ 83 & 55 & 68 & 80 & 89\end{bmatrix} \cdot \begin{bmatrix}186 & 83 \\ 163 & 55 \\ 171 & 68 \\ 1.74 & 80 \\ 190 & 89\end{bmatrix}$$

$$= \begin{bmatrix}(186 \cdot 186 + 163 \cdot 163 + 171 \cdot 171 + 1.74 \cdot 1.74 + 190 \cdot 190) & (\ldots) \\ (83 \cdot 186 + 55 \cdot 163 + 68 \cdot 171 + 80 \cdot 1.74 + 89 \cdot 190) & (\ldots)\end{bmatrix} \approx \begin{bmatrix}133430 & 54180 \\ 54180 & 28859\end{bmatrix}$$

a) Jos matriisi $\mathbf{X}$ on kokoa 361×92 ja matriisi $\mathbf{P}$ on kokoa 92×27, niin mitkä seuraavista matriisituloista ovat sallittuja operaatioita: $\mathbf{XP}$, $\mathbf{X}^T\mathbf{P}$, $\mathbf{X}^T\mathbf{P}^T$, $\mathbf{XP}^T$, $\mathbf{PX}$, $\mathbf{P}^T\mathbf{X}$, $\mathbf{P}^T\mathbf{X}^T$, $\mathbf{PX}^T$. Anna sallittujen matriisitulojen koot.

b) Laske käsin (voit varmistaa koneella)

$$\begin{bmatrix}3 & -2 \\ 4 & 1\end{bmatrix} \cdot \begin{bmatrix}2 & 5 & 3 \\ -2 & 0 & 0\end{bmatrix}$$

Tarkistustulos alkuun: tulomatriisin vasemman ylänurkan arvo on 10.

Matlabissa ja Octavessa

```
X = [186 183 171 1.74 190; 83 55 68 80 89]
4*X
b = [0.8 0.2]'
X'*b
(X'*b)'
X*X'
```

E4. Tutkitaan vektorien välisiä etäisyyksiä. Olkoon tunnettuna neljän kaupungin sijannit xy-koordinaatistossa:

Helsingrad (HSG)	24.8	60.2
Öby (ÖBY)	22.3	60.4
Kurjuu (KRJ)	24.7	62.3
Ulapori (UPI)	25.5	65.0

Tässä etäisyysmatriisi on neliömatriisi, jossa diagonaalilla on nollia: kaupungin etäisyys itsestään on nolla. Vertaa maantiekartaston etäisyystaulukkoon.

Laske etäisyysmatriisi $\mathbf{D} = (d_{ab})$ käyttäen

a) euklidista etäisyyttä L_2 $d_{ab} = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$

b) L_∞-metriikaa $d_{ab} = \max_i\{|a_i - b_i|\}$

c) Manhattan-etäisyyttä (Cityblock) $d_{ab} = \sum_{i=i}^{2} |a_i - b_i|$

Entä jos kaupungeista olisi lisäksi ilmoitettu korkeus ja veroprosenttitiedot? Miten etäisyydet laskettaisiin nyt?

Matlabissa ja Octavessa

```
X  = [24.8 22.3 24.7 25.5; 60.2 60.4 62.3 65.0]
n  = size(X,2);                 % lkm
D  = zeros(n, n);
for a = [1 : n]
  for b = [1 : n]
     D(a,b) = sqrt((X(1,a)-X(1,b))^2 + (X(2,a)-X(2,b))^2);
  end;
end;
D
D1 = squareform(pdist(X', 'euclidean'))
D2 = squareform(max(pdist(X', 'chebychev')))
D3 = squareform(pdist(X', 'cityblock'))
```

Huomaa, että Matlabissa ja Octavessa matriisi $\mathbf{X}$ on toisin päin (transponoitu) kuin tällä kursilla. Toisin sanoen Matlabissa riveillä havainnot (lukumäärä n) ja sarakkeissa piirteet (dimensio d). Tästä syystä koodissa käytetään `X'` eli $\mathbf{X}^T$.

E5. Derivointisääntöjä ja -esimerkkejä löytyy matematiikan kirjoista

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}x} a x^n &= a \frac{\mathrm{d}}{\mathrm{d}x} x^n = a n x^{n-1} \\
\frac{\mathrm{d}}{\mathrm{d}x} a e^{kx} &= a k e^{kx} \\
\frac{\mathrm{d}}{\mathrm{d}x} \log_e(x) &= 1/x \\
\frac{\mathrm{d}}{\mathrm{d}x} \big(p(x) + q(x)\big) &= \frac{\mathrm{d}}{\mathrm{d}x} p(x) + \frac{\mathrm{d}}{\mathrm{d}x} q(x) \\
\frac{\mathrm{d}}{\mathrm{d}x} \big(p(x) \cdot q(x)\big) &= (p(x) \cdot \frac{\mathrm{d}}{\mathrm{d}x} q(x)) + (\frac{\mathrm{d}}{\mathrm{d}x} p(x) \cdot q(x))
\end{aligned}$$

Osittaisderivoinnissa derivoidaan kerrallaan yhden muuttujan suhteen ja pidetään muita vakioina. Tällöin esimerkiksi saman lausekkeen derivointi eri muuttujien suhteen antaa

$$\begin{aligned}
\frac{\partial}{\partial x}\big(x \cdot e^{-x\mu}\big) &= (x \cdot (-\mu) \cdot e^{-x\mu}) + (1 \cdot e^{-x\mu}) \\
\frac{\partial}{\partial \mu}\big(x \cdot e^{-x\mu}\big) &= x \cdot (-x) \cdot e^{-x\mu}
\end{aligned}$$

jossa siis jälkimmäisessä x on vakio derivoinnin suhteen.

a) Hahmottele funktion käyrä $p(x) = x^2 + 3x + 4$. Laske sen derivaatan nollakohta eli $\frac{\mathrm{d}}{\mathrm{d}x}\big(x^2 + 3x + 4\big) = 0$, josta tulee yksi ratkaisu. Ääriarvopiste kertoo, missä kohdassa $p(x)$ saa minimin/maksimin (kumman?) Laske tuossa pisteessä funktion arvo.

b) Pitäisi etsiä μ:lle ääriarvo derivoimalla lauseke, asettamalla se nollaksi ja ratkaisemalla μ:n arvo:

$$\frac{\mathrm{d}}{\mathrm{d}\mu}\Big(\big(K \cdot e^{-(190-\mu)^2/162}\big) \cdot \big(K \cdot e^{-(171-\mu)^2/162}\big) \cdot \big(K \cdot e^{-(174-\mu)^2/162}\big)\Big) = 0$$

Koska logaritmi on monotoninen funktio eli ei muuta ääriarvokohtien sijaintia, lasketaankin derivaatta alkuperäisen sijaan logaritmista

$$\begin{aligned}
& \frac{\mathrm{d}}{\mathrm{d}\mu} \log_e\Big(\big(K \cdot e^{-(190-\mu)^2/162}\big) \cdot \big(K \cdot e^{-(171-\mu)^2/162}\big) \cdot \big(K \cdot e^{-(174-\mu)^2/162}\big)\Big) \\
= & \frac{\mathrm{d}}{\mathrm{d}\mu}\Big(\log_e\big(K \cdot e^{-(190-\mu)^2/162}\big) + \log_e\big(K \cdot e^{-(171-\mu)^2/162}\big) + \log_e\big(K \cdot e^{-(174-\mu)^2/162}\big)\Big) \\
= & \frac{\mathrm{d}}{\mathrm{d}\mu}\Big(\log_e K + (-(190-\mu)^2/162) + \log_e K + (-(171-\mu)^2/162) \\
& \quad + \log_e K + (-(174-\mu)^2/162)\Big) \\
= & \ldots \\
= & 0
\end{aligned}$$

Johda lausekkeen pyörittely loppuun siistiin muotoon ja ratkaise μ:n ääriarvokohta. Huomaa, että $\frac{\mathrm{d}}{\mathrm{d}\mu}C = 0$, jos C on vakio μ:n suhteen. Vastaavasti $\frac{\mathrm{d}}{\mathrm{d}\mu}Kp(\mu) = K\frac{\mathrm{d}}{\mathrm{d}\mu}p(\mu)$, eli vakiot voi nostaa eteen.

Matlabissa ja Octavessa

```
x  = [-5 : 0.01 : 5];
p  = x.^2 + 3*x + 4;
plot(x, p);
[minValue, minIndex] = min(p)
x(minIndex)
p(minIndex)
```

E6. Todennäköisyyslaskentaa. Mitataan ihmisten (mies) pituuksia. Saadaan havainnot

$$\mathbf{X} = [174\ 181\ 179\ 165\ 190\ 171\ 188\ 185\ 192\ 173\ 196\ 184\ 188\ 180\ 178]$$

a) Hahmottele pisteet lukusuoralle x

b) Hahmottele histogrammiesitys, jossa kunkin lokeron leveys on 5 cm

c) Sovita ainestoon (käsivaralta hahmotellen) Gaussin normaalijakauma keskiarvolla μ ja keskihajonnalla σ

Voit ajaa Matlabin tai Octaven komentoriviltä komentoja:

```
% octave / matlab
X     = [174 181 179 165 190 171 188 185 192 173 196 184 188 180 178];
plot(X, ones(size(X)), '*');
%
figure,
hist(X, [162.5 : 5 : 200]);
% matlabissa:
[muHattu, sigmaHattu] = normfit(X);
xc = [150 : 200];
pc = normpdf(xc, muHattu, sigmaHattu);
figure,
plot(xc, pc);
```

E7. Yksiulotteisen normaalijakauman (Gauss) tiheysfunktio on

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

(**ONGELMA tulostetussa PDF:ssä:** eksponentin jakajassa pitäisi olla 2 kertaa sigma toiseen ($2\sigma^2$), mutta sigmaa ei tule printteristä ulos vaikka näkyy ruudulla Acrobat Readerissä?!)

Aivan sama kaava voidaan kirjoittaa hieman eri notaatioilla. Usein yritetään välttää noita hyvin pieniä kirjasinkokoja:

$$p(x) = (2\pi\sigma^2)^{-1/2} \cdot \exp(-(x-\mu)^2/(2\sigma^2))$$

Laske taskulaskimella arvo $p(x) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}$, kun

a) $\sigma = 9$, $\mu = 174$ ja $x = 174$.

b) $\sigma = 9$, $\mu = 174$ ja $x = 190$.

c) $\sigma = 9$, $\mu = 174$ ja $x = 171$.

Muistuta mieliin, miltä käyrä $y = p(x)$ näyttää (katso kaava yllä tai katso netistä "normaalijakauma"). Huomaa, että $p(x) > 0$ aina ja symmetrinen μ:n suhteen. Huomaa myös, että $p(x)$:n lopputulos on siis yksi lukuarvo ja se **ei** tässä esimerkissä ole todennäköisyysarvo; voidaan kysyä, mikä on todennäköisyys $P(X < 174)$, mutta ei ole järkevää kysyä mikä on todennäköisyys $P(X = 174)$.

Hahmottele piirtämällä $p(x)$ yllä olevilla arvoilla $\mu = 174$ ja $\sigma = 9$. Katso $p(x)$:n arvot yllä mainituissa kohdissa x_i. b-kohdan vastaus pitäisi olla välillä $(0.008, 0.212)$.

Voit ajaa Matlabin tai Octaven komentoriviltä komentoja:

```
% octave / matlab
sigma = 9;
mu    = 174;
x     = [130:210];              % x-akselin arvot
K     = 1/(sqrt(2*pi)*sigma);
M     = -(x-mu).^2./(2*sigma^2);
p     = K*exp(M);               % y-akselin arvoiksi p(x)
plot(x, p);                     % piirtokomento
```

E8. a) Esimerkki neliöksi täydentämisestä:

$$\begin{aligned}3x^2+4x+7 &= 3\cdot(x^2+(4/3)x+(7/3))\\ &= 3\cdot(x^2+2\cdot(2/3)x+(2/3)^2-(2/3)^2+(7/3))\\ &= 3\cdot((x+(2/3))^2+(17/9))\end{aligned}$$

b) Kaksi normaalitiheysjakaumaa $p_1(x|\mu_1,\sigma)$ ja $p_2(x|\mu_2,\sigma)$, joilla on sama varianssi σ^2 kerrotaan keskenään ja joilla molemmilla siten sama kerroin $K=1/(\sqrt{2\pi\sigma^2})$:

$$\begin{aligned}p_1(x|\mu_1,\sigma) &= K\cdot e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}\\ p_2(x|\mu_2,\sigma) &= K\cdot e^{-\frac{(x-\mu_2)^2}{2\sigma^2}}\\ p_1(x|\mu_1,\sigma)\cdot p_2(x|\mu_2,\sigma) &= K\cdot e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}\cdot K\cdot e^{-\frac{(x-\mu_2)^2}{2\sigma^2}}\\ &= K^2\cdot e^{-\frac{(x-\mu_1)^2+(x-\mu_2)^2}{2\sigma^2}}\\ &= \dots\\ &= K_n\cdot e^{-\frac{(x-\mu_n)^2}{2\sigma_n^2}}\end{aligned}$$

Miten tulkitset alinta riviä? Mitä on μ_n lausuttuna μ_1:n ja μ_2:n avulla? Pura auki ja täydennä neliöksi puuttuvalla rivillä. Huomaa, että jos a on vakio ja x muuttuja, niin $e^{a+x}=e^a\cdot e^x$, jossa e^a on myös vakio (skaalaustermi).

c) Tee vastaava kertolasku kun yllä b-kohdassa mutta tiheysfunktioille $p_1(x|\mu_1,\sigma_1)$ ja $p_2(x|\mu_2,\sigma_2)$, joiden varianssit ovat myös erilaisia. Mikä on nyt μ_n?

Voit ajaa Matlabin tai Octaven komentoriviltä komentoja:

```
% octave / matlab
sigma = 9;
mu1   = 174;
mu2   = 191;
x     = [140:220];               % x-akselin arvot
K     = 1/(sqrt(2*pi)*sigma);
M1    = -(x-mu1).^2./(2*sigma^2);
M2    = -(x-mu2).^2./(2*sigma^2);
p1    = K*exp(M1);               % y-akselin arvoiksi p1(x)
p2    = K*exp(M2);               % y-akselin arvoiksi p2(x)
hanska= 42;
pn    = p1.*p2*hanska;           % skaalataan hanskavakiolla
plot(x, p1, 'b', x, p2, 'g', x, pn, 'k');   % piirtokomento
```

Datasta Tietoon, autumn 2011, paper exercises 1-5

Round 1 **[Fri 4.11.2011, Mon 7.11.2011]**

H1 / 1. (Convolution filter)
Convolution filtering is computed by the formula

$$g_k = \sum_{m=-\infty}^{\infty} f_m s_{k-m}$$

where f_k is the (discrete) input signal, s_k is the filter sequence, and g_k is the output signal. Compute and plot the output signal when

a)

$$\begin{aligned} f_0 &= 1,\ f_m = 0 \text{ otherwise}; && (1)\\ s_0 &= 2,\ s_1 = 1,\ s_n = 0 \text{ otherwise} && (2)\end{aligned}$$

b)

$$\begin{aligned} f_0 &= 2,\ f_1 = -1,\ f_m = 0 \text{ otherwise}; && (3)\\ s_0 &= -1,\ s_1 = 2,\ s_2 = 1,\ s_n = 0 \text{ otherwise}. && (4)\end{aligned}$$

H1 / 2. (Filtering in frequency domain)
In the frequency domain, the convolution formula of Exercise 1 reads

$$G(\omega) = H(\omega)S(\omega)$$

where the functions are discrete-time Fourier transforms (DTFT) of the corresponding discrete sequences, e.g.,

$$F(\omega) = \sum_{m=-\infty}^{\infty} f_m e^{-i\omega m}$$

a) Show that for sequences f and s in Problem 1 b we get Fourier transforms $F(\omega) = 2 - e^{-i\omega}$ and $S(\omega) = -1 + 2e^{-i\omega} + e^{-2i\omega}$. Compute the product $G(\omega) = H(\omega)S(\omega)$ and compare the coefficients of the polynomial to sequence g in Problem 1 b.

H1 / 3. (Fourier transform)
Discrete-time Fourier transform is defined

$$F(\omega) = \sum_{m=-\infty}^{\infty} f_m e^{-i\omega m}$$

a) Show that the inverse transform is

$$f_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} F(\omega) e^{i\omega n} d\omega.$$

b) The Fourier transform of an ideal low-pass filter (on interval $-\pi \le \omega \le \pi$) is

$$H(\omega) = 1 \text{ if } |\omega| \le \omega_0,\ 0 \text{ otherwise.} \qquad (5)$$

By the inverse transform, compute the corresponding sequence h_n and plot it when $\omega_0 = \pi/2$.

H1 / 4. (Substring histograms)
The DNA molecule can be written as a string with four possible letters A, C, G, T, e.g. ...AAGTACCGTGACGGAT ... Assume that the length is a million letters. We want to form histograms for substrings of length n (if $n = 1$, for A, C, G, T; if $n = 2$, for AA, AC, ... TT and so on). How large should n be chosen if we need on the average at least 10 substrings in each histogram bin?

H1 / 5. (High-dimensional spaces)
The d dimensional data vectors are uniformly distributed over a hypercube with side length 1. Inner points are those whose distance from the surface of the hypercube is at least $\epsilon > 0$. Show that the relative volume of the inner points tends to 0 as $d \to \infty$, that is, for very large dimensions almost all the points are on the surface.

H1 / 6. (High-dimensional spaces)

In the lectures, it was stated without proof that the average distance of n points in a d dimensional hypercube is

$$D(d,n) = \frac{1}{2}(\frac{1}{n})^{\frac{1}{d}}.$$

This is an approximate formula. Let us look at a special case: the n points are in the centers of n similar non-overlapping smaller cubes whose union makes up the whole hypercube. Show that then the distances of the points are

$$D(d,n) = (\frac{1}{n})^{\frac{1}{d}}.$$

The distance between two points $\mathbf{x}_1, \mathbf{x}_2$ is defined as $\max_i |x_{i1} - x_{i2}|$. Consider the case $d = 2$, $n = 4$ and verify your result.

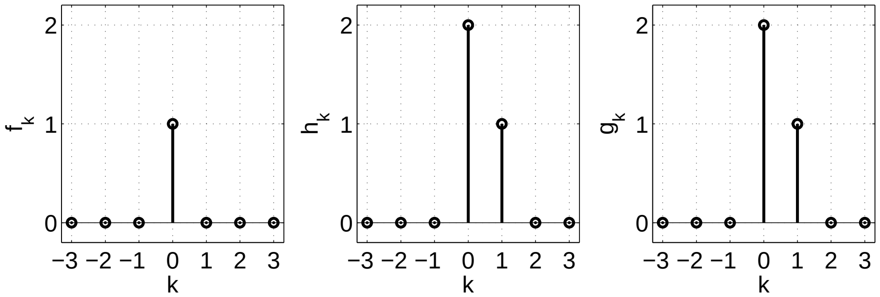
H1 / Problem 1.
Convolution sum is computed as

$$g_k = \sum_{m=-\infty}^{\infty} f_m s_{k-m} = \ldots + f_{-2}s_{k+2} + f_{-1}s_{k+1} + f_0 s_k + f_1 s_{k-1} + f_2 s_{+2} + \ldots$$

a) Now

$$\begin{aligned} f_0 &= 1,\ f_m = 0 \text{ otherwise;} && (6)\\ s_0 &= 2,\ s_1 = 1,\ s_n = 0 \text{ otherwise} && (7) \end{aligned}$$

Thus $g_k = f_0 s_{k-0} = s_k$, which is $g_0 = 2$, $g_1 = 1$, and $g_k = 0$ elsewhere.



The other sequence f_k was an identity sequence (only one at $k = 0$, zero elsewhere), so it just copies the other sequence s_k into the output.

b) Now

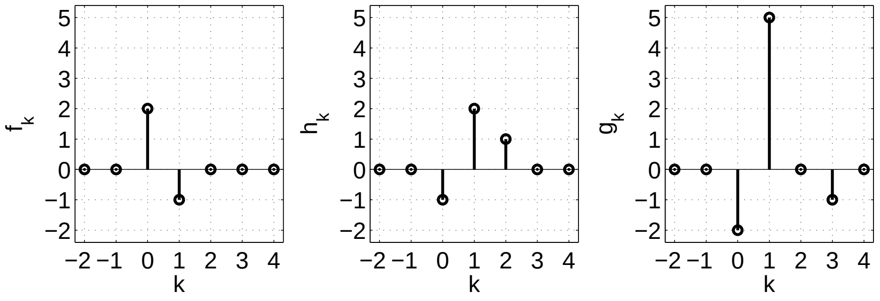
$$\begin{aligned} f_0 &= 2,\ f_1 = -1,\ f_m = 0 \text{ otherwise;} && (8)\\ s_0 &= -1,\ s_1 = 2,\ s_2 = 1,\ s_n = 0 \text{ otherwise.} && (9) \end{aligned}$$

Thus

$$g_k = f_0 s_{k-0} + f_1 s_{k-1} = 2s_k - s_{k-1}$$

and we get

$$\begin{aligned} g_0 &= 2s_0 - s_{-1} = -2 && (10)\\ g_1 &= 2s_1 - s_0 = 4 + 1 = 5 && (11)\\ g_2 &= 2s_2 - s_1 = 2 - 2 = 0 && (12)\\ g_3 &= 2s_3 - s_2 = -1 && (13)\\ g_k &= 0 \quad \text{otherwise} && (14) \end{aligned}$$



Sequence $f_k = \{\underline{2}, -1\}$ was now a sum sequence of an identity filter multiplied by two ($f_0 = 2$) and a shifted identity filter multiplied by -1 ($f_1 = -1$). Therefore the output consisted of a sum of s_k multiplied by two and a shifted s_k multiplied by -1.

$$\begin{aligned} 2s_k - s_{k-1} &= 2 \cdot \{\underline{-1}, 2, 1\} - 1 \cdot \{\underline{0} - 1, 2, 1\}\\ &= \{\underline{-2+0}, 4+1, 2-2, 0-1\} = \{\underline{-2}, 5, 0, -1\} \end{aligned}$$

See more examples in the computer session **T1**.

H1 / Problem 2.
a) From Problem 1 b

$$\begin{aligned} f_0 &= 2,\ f_1 = -1,\ f_m = 0 \text{ otherwise;} && (15)\\ s_0 &= -1,\ s_1 = 2,\ s_2 = 1,\ s_n = 0 \text{ otherwise} && (16)\end{aligned}$$

we get using the definition

$$F(\omega) = \sum_{m=-\infty}^{\infty} f_m e^{-i\omega m}$$

$$\begin{aligned} F(\omega) &= f_0 \cdot e^{-i\omega 0} + f_1 e^{-i\omega 1} = 2 - e^{-i\omega} && (17)\\ S(\omega) &= s_0 \cdot e^{-i\omega 0} + s_1 \cdot e^{-i\omega 1} + s_2 \cdot e^{-i\omega 2} = -1 + 2e^{-i\omega} + e^{-2i\omega} && (18)\end{aligned}$$

Convulution of two sequences in time-domain corresponds multiplication of two transforms in transform/frequency-domain. The real argument ω gets normally values $-\pi \ldots \pi$ or $0 \ldots \pi$

$$\begin{aligned} G(\omega) &= F(\omega)S(\omega) && (19)\\ &= (2 - e^{-i\omega}) \cdot (-1 + 2e^{-i\omega} + e^{-2i\omega}) && (20)\\ &= -2 + 5e^{-i\omega} - e^{-3i\omega} && (21)\end{aligned}$$

We find out that the coefficients $\{-2, 5, 0, -1\}$ of the polynomial $G(\omega)$ are equal to the sequence g_k.

Remark. There are several integral transforms that are used in specific cases:

- *Fourier series*, where signal $f(t)$ is analog and periodic (Ω_0), gives discrete and aperiodic Fourier series coefficients F_n with multiples of the fundamental angular frequency Ω_0
- (Continuous-time) Fourier transform, where signal $f(t)$ is analog and aperiodic, gives continuous and aperiodic transform $F(\Omega)$
- Discrete-time Fourier transform, where signal f_k is discrete and aperiodic, gives continuous and periodic transform $F(\omega)$ as above
- Discrete Fourier transform (DFT), where signal f_k is discrete and periodic (length N), gives discrete and periodic transform F_n (length N)

H1 / Problem 3.

a) Substitute $F(\omega)$ into the integral:

$$I = \frac{1}{2\pi}\int_{-\pi}^{\pi}[\sum_{m=-\infty}^{\infty} f_m e^{-i\omega m}]e^{i\omega n}d\omega = \frac{1}{2\pi}\sum_{m=-\infty}^{\infty} f_m \int_{-\pi}^{\pi} e^{i\omega(n-m)}d\omega$$

with $i=\sqrt{-1}$ the imaginary unit (sometimes also denoted j).

For the integral we get (note that $n, m \in \mathbb{Z}$)

$$\int_{-\pi}^{\pi} e^{i\omega(n-m)}d\omega = \begin{cases} 2\pi & \text{if } n = m, \\ /_{-\pi}^{\pi} \frac{1}{i(n-m)}e^{i\omega(n-m)} = \frac{1}{i(n-m)}\left(e^{i\pi(n-m)} - e^{-i\pi(n-m)}\right) & \text{if } n \neq m \end{cases}$$

We can easily see that $e^{i\pi(n-m)} = e^{-i\pi(n-m)}$ because $e^{i\pi} = e^{-i\pi} = -1$. Thus the integral is 2π if $n = m$ and zero otherwise. Substituting this into the full expression gives $I = f_n$ which was to be shown.

b)

$$\begin{aligned} h_n &= \frac{1}{2\pi}\int_{-\omega_0}^{\omega_0} e^{i\omega n}d\omega = \frac{1}{2\pi}/_{-\omega_0}^{\omega_0}\frac{1}{in}e^{i\omega n} && (22) \\ &= \frac{1}{2\pi i n}(e^{i\omega_0 n} - e^{-i\omega_0 n}) && (23) \\ &= \frac{1}{2\pi i n}[\cos(\omega_0 n) + i\sin(\omega_0 n) - \cos(\omega_0 n) + i\sin(\omega_0 n)] && (24) \\ &= \frac{1}{\pi n}\sin(\omega_0 n). && (25) \end{aligned}$$

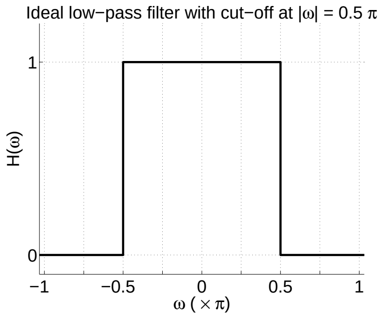
Using the cut-off frequency $\omega = \pi/2$ we get

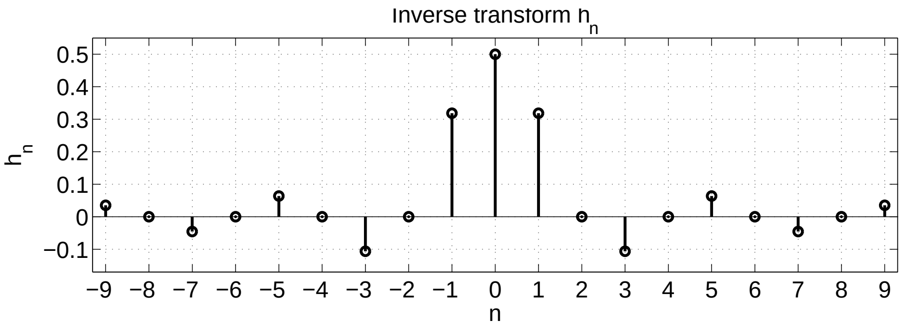
$$h_n = \frac{1}{\pi n}\sin(\frac{\pi n}{2})$$

which is sometimes written as $h_n = (1/2)\mathrm{sinc}(n/2)$, where sinc function is $\mathrm{sinc}(\omega n) = \sin(\pi\omega n)/(\pi\omega n)$. Some values: $h_0 = 0.5$, $h_1 = 1/\pi$, $h_2 = 0$.

Note that at $n = 0$ we end up to $0/0$. It can be solved, e.g., either Taylor series $(1/x)\sin(x/2) = (1/2)(2/x)\sin(x/2) = (1/2)-(x^2/48)+\ldots$, or l'Hospital's rule by derivating both sides. Thus at zero the value is 0.5. In addition, $\mathrm{sinc}(0) = 1$.

Note also that the sequence h_n is infinitely long.





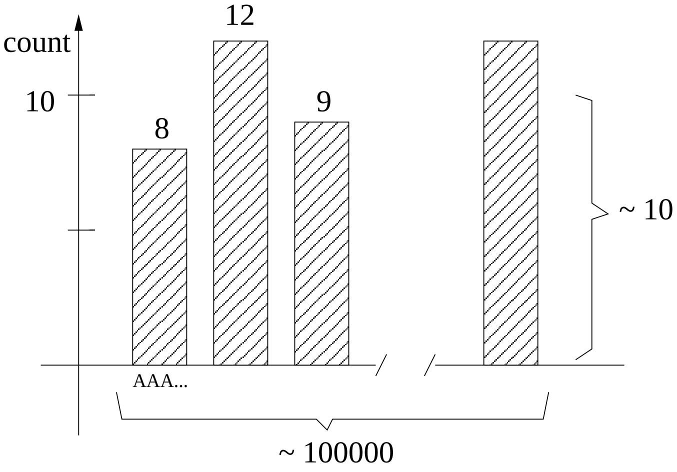
H1 / Problem 4.

Now the number of bins is at most 100000, because the average number of substrings in a bin must be at least 10. The number of different substrings of length n is 4^n. We get

$$4^n \leq 100000$$

giving $n \leq 8$.

An example of a histogram of a data sample given below. It is assumed that letters are drawn independently from uniform distribution, i.e., the total amount of each letter is the same.



Another example on building a histogram with the sequence 'AAGTACCGTGACGGAT'.

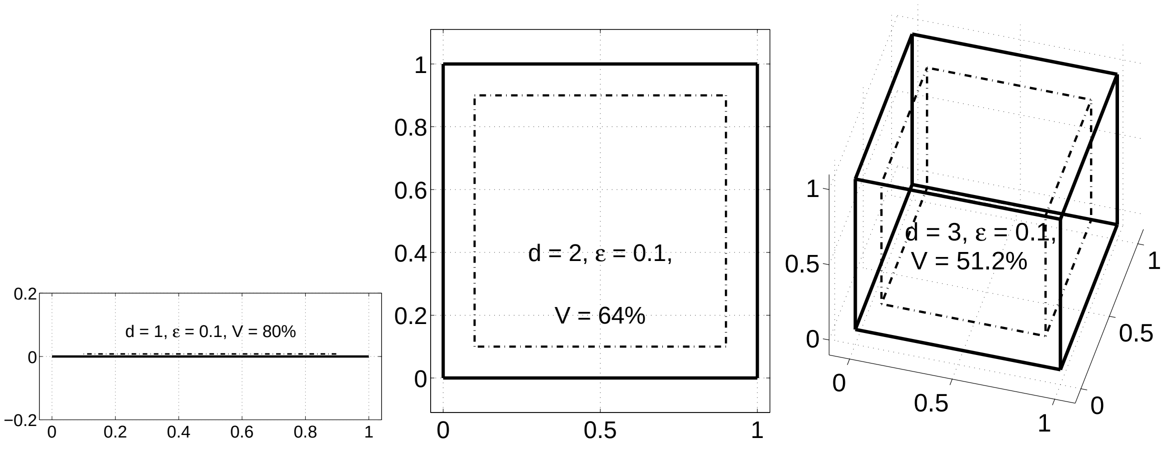
If $n = 1$, all possible substrings are 'A', 'C', 'G', and 'T', shortly A, C, G, T. The number of substrings is $4^1 = 4$. The count for each substring: $|A| = 5$, $|C| = 3$, $|G| = 5$, and $|T| = 3$.

If $n = 2$, all possible substrings are 'AA', 'AC', 'AG', 'AT', 'CA', 'CC', 'CG', 'CT', 'GA', 'GC', 'GG', 'GT', 'TA', 'TC', 'TG', 'TT', that is, $4^2 = 16$ substrings. The count for each substring: $|AA| = 1$, $|AC| = 2$, $|AG| = 1$, $|AT| = 0$, etc.

H1 / Problem 5.

The volume of the unit hypercube is 1 and the volume of the set of inner points is $V_d = (1-2\epsilon)^d$. For any ϵ, this tends to 0 as $n \to \infty$.

Below an illustration of hypercubes in dimensions $d = 1,\ 2,\ 3$ with $\epsilon = 0.1$. We can see that the volume of inner points decreases when the dimension increases.



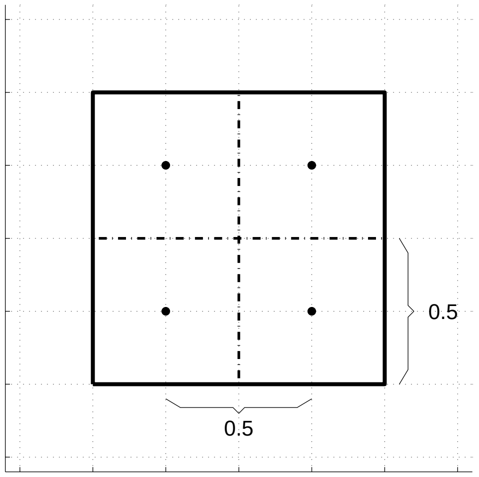
H1 / Problem 6.

Now the small hypercubes are similar, hence all have the same volume which must be $\frac{1}{n}$ times the volume of the large unit hypercube. (This is only possible for certain values of (n, d); for $d = 2$, n must be 4, 9, 16, ...; for $d = 3$, n must be 8, 27, 64 ... etc.)

Also, we assume here a special distance which is not Euclidean distance but $D(\mathbf{x}_1, \mathbf{x}_2) = \max_i |x_{i1} - x_{i2}|$, that is, the largest distance along the coordinate axes.

Then it is easy to see that the distance of the centres of the small hypercubes is equal to the length of their side s. Because the volume is $s^d = \frac{1}{n}$, we have $s = \frac{1}{n}^{\frac{1}{d}}$.

The case of $d = 2, n = 4$ is shown below.



Round 2 **[Fri 11.11.2011, Mon 14.11.2011]**

H2 / 1. (Principal component analysis)

We have the following data matrix $\mathbf{X}$:

$$\mathbf{X} = \begin{bmatrix} 2 & 5 & 6 & 7 \\ 1 & 3 & 5 & 7 \end{bmatrix}$$

a) Plot the columns of $\mathbf{X}$ in the (x_1, x_2) - coordinate system

b) Normalize $\mathbf{X}$ to zero mean by subtracting from the columns their mean vector

c) Compute the covariance matrix $\mathbf{C}$ and the eigenvector corresponding to its largest eigenvalue. Plot the eigenvector in the coordinate system of item a). How would you interpret the results according to PCA?

H2 / 2. (Principal component analysis)

Assume that $\mathbf{x}$ is a zero mean random vector and we have a sample $\mathbf{x}(1), ..., \mathbf{x}(n)$. Assume $\mathbf{w}$ is a unit vector (such that $\|\mathbf{w}\| = 1$) and define $y = \mathbf{w}^T\mathbf{x}$. We want to maximize the variance of y given as $E\{y^2\} = E\{(\mathbf{w}^T\mathbf{x})^2\}$. Prove that it will be maximized when $\mathbf{w}$ is the eigenevctor of the matrix $E\{\mathbf{x}\mathbf{x}^T\}$ corresponding to its largest eigenvalue.

H2 / 3. (ML-estimation)

Derive the maximum likelihood estimate for the parameter λ of the exponential probability density

$$p(x|\lambda) = \lambda e^{-\lambda x}$$

when there is available a sample $x(1), ..., x(n)$ of the variable x.

H2 / 4. (Bayesian estimation)

We are given a sample $x(1), ..., x(n)$ of a variable x known to be normally distributed

$$p(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

We have good reason to assume that the average value μ is close to zero. Let us code this assumption into a prior density

$$p(\mu) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\mu^2}.$$

Derive the Bayes MAP estimate for the value μ and interpret your result when the variance σ^2 changes from a small to a large value.

H2 / Problem 1.

a) See the figure below left.

b) Compute first mean and substract it from **X**.:

$$E\{\mathbf{x}\} = \frac{1}{4}\sum \mathbf{x}(i) = \begin{bmatrix}5\\4\end{bmatrix}$$

Thus the normalized data matrix is

$$\mathbf{X}_0 = \begin{bmatrix}-3 & 0 & 1 & 2\\ -3 & -1 & 1 & 3\end{bmatrix}$$

c) The covariance matrix is

$$\mathbf{C}_x = \frac{1}{4}\mathbf{X}_0\mathbf{X}_0^T = \frac{1}{4}\begin{bmatrix}14 & 16\\ 16 & 20\end{bmatrix}$$

The eigenvalues are computed from $\mathbf{C}_x\mathbf{u} = \lambda\mathbf{u}$, or by multiplying with 4, $\begin{bmatrix}14 & 16\\ 16 & 20\end{bmatrix}\mathbf{u} = \mu\mathbf{u}$ where μ is 4 times λ. (It may be easier to solve the equation if the coefficients are integer numbers).

We have determinant $\begin{vmatrix}14-\mu & 16\\ 16 & 20-\mu\end{vmatrix} = 0$ which gives the characteristic equation $(14-\mu)(20-\mu) - 256 = 0$ or $\mu^2 - 34\mu + 24 = 0$. The roots are 33.28 and 0.72, hence the eigenvalues λ of the covariance matrix are these divided by 4, $\lambda_1 = 8.32$ and $\lambda_2 = 0.18$.

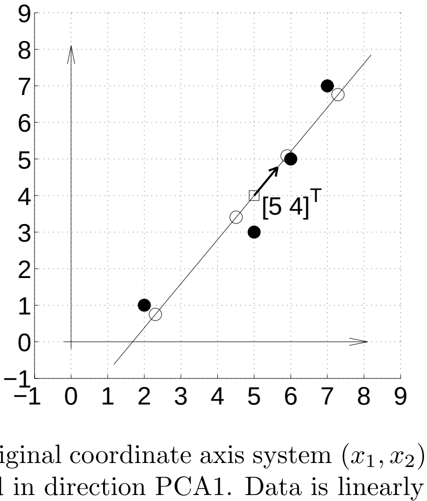
The eigenvector $\mathbf{u}$ corresponding to the larger eigenvalue λ_1 can be computed from $\mathbf{C}_x\mathbf{u} = \lambda_1\mathbf{u}$ by

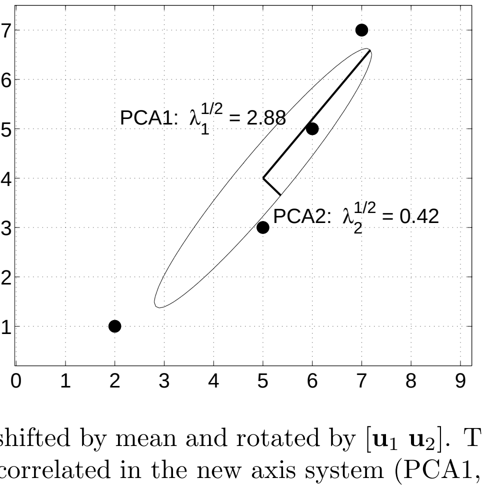
$$\begin{aligned}
\begin{bmatrix}14 & 16\\ 16 & 20\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix} &= 33.28\begin{bmatrix}u_1\\ u_2\end{bmatrix}\\
\begin{cases}14u_1 + 16u_2 = 33.28u_1\\ 16u_1 + 20u_2 = 33.28u_2\end{cases} &\Rightarrow \quad u_1 = 0.83u_2\\
\Rightarrow \mathbf{u}_1 &= a\begin{bmatrix}0.83\\ 1\end{bmatrix}, \quad a \in \mathbb{R}
\end{aligned}$$

After normalization to the unit length ($\mathbf{u}/||\mathbf{u}||$) the eigenvector corresponding largest eigenvalue λ_1 is $\mathbf{u} = [0.64\ 0.77]^T$.

The empty circles in the figure (below, left) are the projections onto 1D hyperplane (PCA1 line) by $Y = \mathbf{u}_1^T\mathbf{X}_0 = [-4.23\ -0.77\ 1.41\ 3.59]$. First PCA1 axis explains $8.32/(8.32+0.18) \approx 97.9$ % of the total variance.

In the other figure (right) it can be seen that the propotional length of the axis of an ellipse in direction PCA1 is $\sqrt{\lambda_1} \approx 2.88$, and in direction PCA2 $\sqrt{\lambda_2} \approx 0.42$. The ellipse is derived from the Gaussian covariance matrix $\mathbf{C}_x$.





In PCA the original coordinate axis system (x_1, x_2) is shifted by mean and rotated by $[\mathbf{u}_1\ \mathbf{u}_2]$. The variance of the data is maximized in direction PCA1. Data is linearly uncorrelated in the new axis system (PCA1, PCA2).

H2 / Problem 2.

We can use the *Lagrange optimization* principle for a constrained maximization problem. The principle is saying that if we need to maximize $E\{(\mathbf{w}^T\mathbf{x})^2\}$ under the constraint $\mathbf{w}^T\mathbf{w} = 1$, we should find the zeroes of the gradient of

$$E\{(\mathbf{w}^T\mathbf{x})^2\} - \lambda(\mathbf{w}^T\mathbf{w} - 1)$$

where λ is the Lagrange constant.

We can write $E\{(\mathbf{w}^T\mathbf{x})^2\} = E\{(\mathbf{w}^T\mathbf{x})(\mathbf{x}^T\mathbf{w})\} = \mathbf{w}^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w}$ because inner product is symmetrical and the E or expectation means computing the mean over the sample $\mathbf{x}(1), ..., \mathbf{x}(n)$, thus $\mathbf{w}$ can be taken out.

We need the following general result: if $\mathbf{A}$ is a symmetrical matrix, then the gradient of the quadratic form $\mathbf{w}^T\mathbf{A}\mathbf{w}$ equals $2\mathbf{A}\mathbf{w}$. It would be very easy to prove this by taking partial derivatives with respect to the elements of $\mathbf{w}$. This is a very useful formula to remember.

Now the gradient of the Lagrangian becomes:

$$2E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w} - \lambda(2\mathbf{w}) = 0$$

or

$$E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w} = \lambda\mathbf{w}$$

This is the eigenvalue - eigenvector equation for matrix $E\{\mathbf{x}\mathbf{x}^T\}$. But there are d eigenvalues and vectors: which one should be chosen?

Multiplying from the left by $\mathbf{w}^T$ and remembering that $\mathbf{w}^T\mathbf{w} = 1$ gives

$$\mathbf{w}^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w} = \lambda$$

showing that λ should be chosen as the largest eigenvalue in order to maximize $\mathbf{w}^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w} = E\{y^2\}$. This was to be shown.

H2 / Problem 3.

Problem: Given data sample $\mathbf{X}$, compute estimator $\hat{\lambda}_{\text{ML}}$ with which data has been most probably generated. The maximum likelihood (ML) method is summarized as (p. 233. Milton, Arnold: Introduction to probability and statistics. Third edition. McGraw-Hill, 1995)

1. Obtain a random sample $\mathbf{X} = \{x(1), x(2), \ldots, x(n)\}$ from the distribution of a random variable X with density p and associated parameter θ
2. Define a likelihood function $L(\theta)$ by
$$L(\theta) = \prod_{i=1}^{n} p(x(i))$$
3. Find the expression for θ that maximizes the likelihood function. This can be done directly or by maximizing $\ln L(\theta)$
4. Replace θ by $\hat{\theta}$ to obtain an expression for ML estimator for θ
5. Find the observed value of this estimator for a given sample

Let us assume that data samples $\mathbf{X} = \{x(1), x(2), \ldots, x(n)\}$ are i.i.d., that is, they are independent and identically-distributed. Independence means that joint density function $P(A, B, C)$ can be decomposed to product of marginal density functions: $P(A, B, C) = P(A) \cdot P(B) \cdot P(C)$. Each sample $x(i)$ is from the same (identical) distribution $p(x|\lambda)$ with the same λ. One-dimensional exponential propability density function is $p(x|\lambda) = \lambda e^{-\lambda x}$ where rate $\lambda = 1/\mu$.

In this case the likelihood function $L(\lambda)$ ("uskottavuusfunktio") is

$$L(\lambda) = p(\mathbf{X}|\lambda) = p(x(1), x(2), \ldots, x(n)|\lambda) \overset{\text{i.i.d.}}{=} \prod_{i=1}^{n} p(x(i)|\lambda) = \prod_{i=1}^{n} \lambda e^{-\lambda x(i)}$$

Because samples $x(i)$ are known, likelihood is a function of λ only.

In order to find estimator $\hat{\lambda}_{\text{ML}}$ we maximize likelihood by, e.g., setting derivative to zero. Because we are finding the extreme point, we can take logarithm and still find the same maximum point of likelihood function. The computation comes much easier because $\ln(A \cdot B \cdot C) = \ln A + \ln B + \ln C$. The log-likelihood:

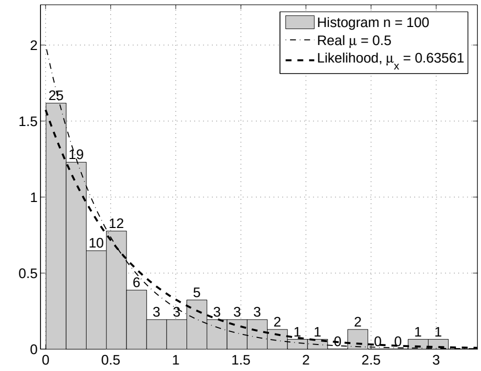
$$\begin{aligned}
\ln L(\lambda) = \ln p(\mathbf{X}|\lambda) &= \ln \prod_{i=1}^{n} [\lambda e^{-\lambda x(i)}] \\
&= \sum_{i=1}^{n} [\ln \lambda - \lambda x(i)] \\
&= n \ln \lambda - \lambda \sum_{i=1}^{n} x(i)
\end{aligned}$$

Putting the derivative with respect to λ to zero gives the solution $\hat{\lambda}_{\text{ML}}$

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\lambda} \ln L(\lambda) = \frac{\mathrm{d}}{\mathrm{d}\lambda} \Big\{ n \ln \lambda - \lambda \sum_{i=1}^{n} x(i) \Big\} &= \frac{n}{\lambda} - \sum_{i=1}^{n} x(i) = 0 \\
\frac{1}{\lambda} &= \frac{1}{n} \sum_{i=1}^{n} x(i)
\end{aligned}$$

Thus the ML (maximum likelihood) estimate for λ is the inverse of the mean value of the sample.

An example in the figure below. $n = 100$ samples $x(1), \ldots, x(100)$ are drawn from the exponential distribution with $\mu = 0.5 \Leftrightarrow \lambda = 1/\mu = 2$. Sample mean $1/\hat{\lambda}_{\text{ML}} = \bar{X} = 0.636$ at this time.

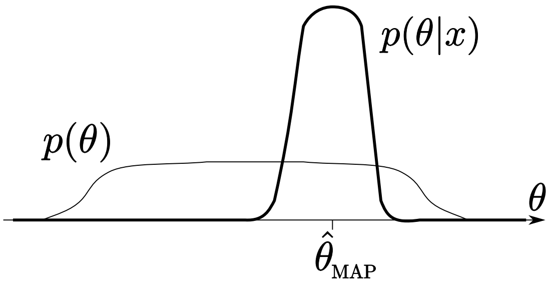


H2 / Problem 4.

Problem: Given data sample $\mathbf{x}$ and prior distribution for μ, compute estimator $\hat{\mu}_{\text{MAP}}$. This Bayesian maximum posterior (MAP) method follows that of maximum likelihood (Problem H2/3) but now the function to be maximized is not likelihood but posterior = likelihood × prior. Inference using Bayes' theorem can be written as

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$$

where $p(\theta|x)$ is posterior, $p(x|\theta)$ likelihood, $p(\theta)$ prior and $p(x)$ evidence, which is just a scaling factor. θ contains all parameters. This results to a posterior distribution (our knowledge after seeing data) with respect to θ which is more exact (with smaller variance) than prior (our knowledge or guess before seeing any data), see figure below. Note that here we have a distribution for θ whereas the maximum likelihood gives a point estimate. Finally, however, the MAP estimate is a single value that maximizes the posterior.



Let us again assume that data samples $\mathbf{X} = \{x(1), x(2), \ldots, x(n)\}$ are i.i.d., that is, they are independent and identically-distributed. The one-dimensional normal (Gaussian) density function is

$$p(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Each sample $x(i)$ is from the same (identical) distribution $p(x|\mu,\sigma)$ with the same μ and σ.

Likelihood function is

$$L(\mu,\sigma) = p(\mathbf{X}|\mu,\sigma) \overset{\text{i.i.d.}}{=} \prod_{i=1}^{n} p(x(i)|\mu,\sigma)$$

Our prior for μ (with hyperparameters $\mu_0 = 0$ and $\sigma_0 = 1$) is

$$p(\mu) = \frac{1}{\sqrt{2\pi}}e^{-\frac{\mu^2}{2}}$$

Posterior is

$$p(\mu,\sigma|\mathbf{X}) \propto L(\mu,\sigma)p(\mu)p(\sigma)$$

where the constant denominator $p(\mathbf{X})$ can be omitted when searching maximum. The symbol $\propto$ can be read "is propotional".

Taking logarithm of likelihood function and setting the derivative with respect to μ to zero follows computation as in Problem H2/3. The log-likelihood:

$$\begin{aligned}\ln L(\mu,\sigma) = \ln p(\mathbf{X}|\mu,\sigma) &= \ln\prod_{i=1}^{n}[\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x(i)-\mu)^2}{2\sigma^2}}] \\ &= \sum_{i=1}^{n}[\ln(\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x(i)-\mu)^2}{2\sigma^2}})] \\ &= \sum_{i=1}^{n}[-\ln\sigma - \ln(\sqrt{2\pi}) - \frac{(x(i)-\mu)^2}{2\sigma^2}]\end{aligned}$$

The log-prior probability for μ is

$$\ln p(\mu) = -\ln(\sqrt{2\pi}) - \frac{1}{2}\mu^2$$

The log-posterior can be written with Bayes' theorem as a sum of log-likelihood and log-prior

$$\ln p(\mu,\sigma|\mathbf{X}) \propto \ln L(\mu,\sigma) + \ln p(\mu) + \ln p(\sigma)$$

In other words, all parts depending on μ in the Bayesian log-posterior probability are:

$$-\frac{1}{2\sigma^2}\sum_{i=1}^{n}[(x(i)-\mu)^2]-\frac{1}{2}\mu^2$$

Setting derivative of log-posterior with respect to μ to zero gives

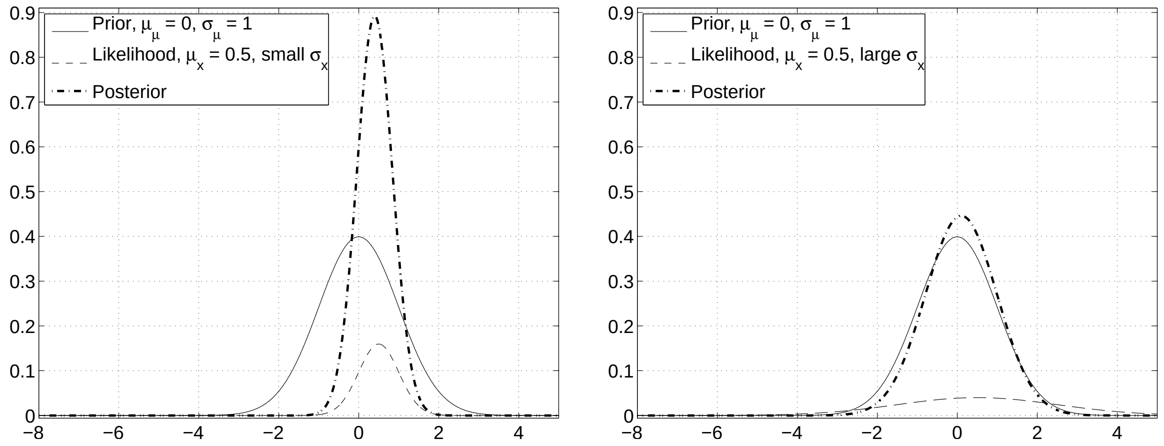
$$\begin{aligned}0 &= \frac{\mathrm{d}}{\mathrm{d}\mu}\Big\{\Big(-\frac{1}{2\sigma^2}\Big)\sum_{i=1}^{n}[(x(i)-\mu)^2]-\frac{1}{2}\mu^2\Big\}\\ &= -\frac{1}{2\sigma^2}\sum_{i=1}^{n}[2(x(i)-\mu)(-1)]-\mu\\ &= \sum_{i=1}^{n}[x(i)]-n\mu-\sigma^2\mu\end{aligned}$$

which finally gives $\hat{\mu}_{\mathrm{MAP}}$

$$\mu=\frac{1}{n+\sigma^2}\sum_{i=1}^{n}x(i)$$

The interpretation is as follows: if the variance σ^2 of the sample is very small, then the sample can be trusted. Therefore μ is very close to the sample mean $\frac{1}{n}\sum_{i=1}^{n}x(i)$ (likelihood estimate). See an example in the figure below left: $\hat{\mu}_{\mathrm{MAP}}\approx 0.48$ (posterior) is close to $\mu_{\mathrm{ML}}=0.5$ (likelihood).

On the other hand, if σ^2 is very large, then the sample cannot be trusted and the prior information dominates. Density function of μ becomes close to that of prior assumption. See an example in the figure below right: $\hat{\mu}_{\mathrm{MAP}}\approx 0.04$ (posterior) is close to $\mu_{\mathrm{PRIOR}}=0$.



In case of maximum likelihood, the estimator is $\hat{\mu}_{\mathrm{ML}}=\frac{1}{n}\sum_{i=1}^{n}x(i)=\bar{X}$. The only, but remarkable difference is the variance term in the denominator.

Round 3 [**Fri 18.11.2011, Mon 21.11.2011**]

H3 / 1. (MLE-regression)

We have n pairs of observations $(y(i), x(i))$, $i = 1, \ldots, n$ of some variables x, y which are believed to be linearly related by the model $y = \theta x$. However, the observations contain some errors: $y(i) = \theta x(i) + \epsilon(i)$ where $\epsilon(i)$ is the observation error ("noise") at the i:th point. Assume that the observation error $\epsilon(i)$ is gaussian distributed with mean value 0 and standard deviation σ.

Solve the parameter θ in the model using Maximum Likelihood estimation.

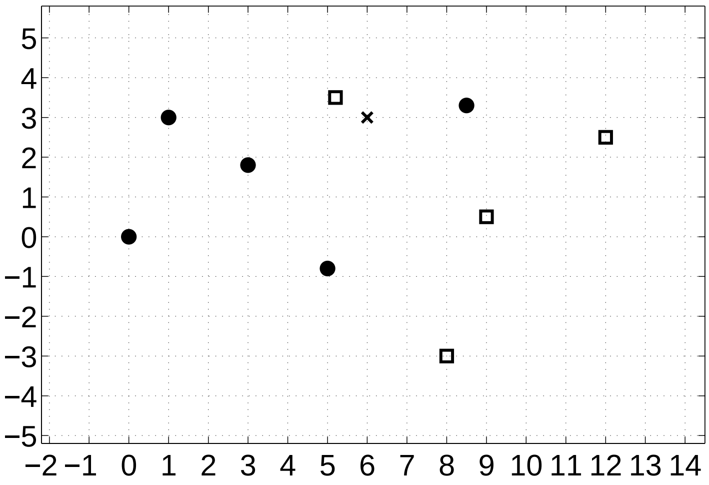
H3 / 2. (Bayes regression)

Let us add some prior knowledge to the previous problem. 1. Suppose that the parameter θ is roughly equal to 1. We model this uncertainty by assuming a gaussian prior density with mean value 1 and standard deviation 0.5.

2. Suppose that the regression line may not pass through the origin after all, but it has the form $y = \alpha + \theta x$. The relation between our observations is then $y(i) = \alpha + \theta x(i) + \epsilon(i)$. We model the uncertainty related to the new parameter α by assuming that is has a gaussian prior density with mean value 0 and standard deviation 0.1. Compute the Bayes estimates for the parameters α, θ.

H3 / 3. (Nearest neighbor classifier, k-NN)

In the following picture there are 2 classes (circles and squares) in 2 dimensions. Using k-NN classify a new data point $\mathbf{x} = (6, 3)$ with $k = 1$ (only closest) and $k = 3$. Plot the decision boundary (border between classes) of the 1-NN classifier.



H3 / 4. (Bayes classifier)

Assume two classes for a scalar variable x. The class densities $p(x|\omega_1), p(x|\omega_2)$ are gaussian such that both have mean value 0 but different standard deviations σ_1, σ_2. The prior probabilities are $P(\omega_1), P(\omega_2)$. Plot the densities. Where would you place the decision boundaries? Then, derive the decision boundaries of the Bayes classifier.

H3 / Problem 1.

About regression: See lectures slides, chapter 5. A typical example of regression is to fit a polynomial curve into data $(x(j), y(j))$ with some error $\epsilon(j)$:

$$y = b_0 + b_1 x + b_2 x^2 + \ldots + b_P x^P + \epsilon$$

We often assume that $\epsilon(j)$ is, e.g., Gaussian noise with zero-mean and variance σ_2. After estimating b_k, a regression output (missing $y(j)$) can be derived for any new sample x_{new} by

$$y_{new} = b_0 + b_1 x_{new} + b_2 x_{new}^2 + \ldots + b_P x_{new}^P$$

About ML: See lectures slides, chapter 5. See also H2/3 and H2/4. Given a data set $\mathbf{X} = (x(1), x(2), \ldots, x(n))$ and a model of a probability density function $p(x|\theta)$ with an unknown constant parameter vector θ, maximum likelihood method ("suurimman uskottavuuden menetelmä") estimates vector $\hat{\theta}$ which maximizes the likelihood function: $\hat{\theta}_{ML} = \max_\theta p(\mathbf{X}|\theta)$. In other words, find the values of θ which most probably have generated data $\mathbf{X}$.

Normally the data vectors $\mathbf{X}$ are considered independent so that likelihood function $L(\theta)$ is a product of individual terms $p(\mathbf{X}|\theta) = p(x(1), x(2), \ldots, x(n)|\theta) = p(x(1)|\theta) \cdot p(x(2)|\theta) \cdot \ldots \cdot p(x(n)|\theta)$. Given a numerical data set $\mathbf{X}$, likelihood is function of only θ. Because the maximum of the likelihood $p(\mathbf{X}|\theta)$ and log-likelihood $\ln p(\mathbf{X}|\theta)$ is reached at the same value θ, log-likelihood function $L(\theta)$ is prefered for computational reasons. While $\ln(A \cdot B) = \ln A + \ln B$, we get $\ln L(\theta) = \ln p(\mathbf{X}|\theta) = \ln \prod_j p(x(j)|\theta) = \sum_j \ln p(x(j)|\theta)$.

Remember also that $p(x, y|\theta)$ can be written with conditional probabilities $p(x, y|\theta) = p(x)p(y|x, \theta)$.

In this problem the model is $y(i) = \theta x(i) + \epsilon(i)$ which implies $\epsilon(i) = y(i) - \theta x(i)$. If there were no noise ϵ, θ could be computed from a single observation $\theta = y(1)/x(1)$. However, now the error ϵ is supposed to be zero-mean Gaussian noise with standard deviation σ: $\epsilon \sim N(0, \sigma)$, that is $E(\epsilon) = 0$, $Var(\epsilon) = \sigma^2$. This results to

$$\begin{aligned}
E(y(i)|x(i), \theta) &= E(\theta x(i) + \epsilon(i)) = E(\theta x(i)) + E(\epsilon(i)) = \theta x(i) \\
Var(y(i)|x(i), \theta) &= Var(\theta x(i) + \epsilon(i)) \\
&= E((\theta x(i) + \epsilon(i))^2) - (E(\theta x(i) + \epsilon(i)))^2 \\
&= E((\theta x(i))^2 + 2\theta x(i)\epsilon(i) + \epsilon(i)^2) - (\overbrace{E(\theta x(i) + \epsilon(i))}^{\text{see above}})^2 \\
&= E((\theta x(i))^2) + \overbrace{E(2\theta x(i)\epsilon(i))}^{=0 \text{ no correlation}} + E(\epsilon(i)^2) - (\theta x(i))^2 \\
&= E(\epsilon(i)^2) = Var(\epsilon(i)) = \sigma^2
\end{aligned}$$

Hence $(y(i)|x(i), \theta) \sim N(\theta x(i), \sigma)$ the density function is

$$p(y(i)|x(i), \theta) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(y(i)-\theta x(i))^2}{2\sigma^2}} \tag{26}$$

The task is to maximize $p(x, y|\theta) = p(x)p(y|x, \theta)$ with respect to (w.r.t.) θ. Assuming data vectors independent we get likelihood as

$$L(\theta) = \prod_i p(x(i))p(y(i)|x(i), \theta)$$

After taking logarithm the log-likelihood function is

$$\begin{aligned}
\ln L(\theta) &= \text{const} + \sum_{i=1}^{n} (\ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{(y(i) - \theta x(i))^2}{2\sigma^2}) && (27) \\
&= \text{const}_2 - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y(i) - \theta x(i))^2 && (28)
\end{aligned}$$

Maximizing $L(\theta)$ (or $\ln L(\theta)$) is equal to minimizing its opposite number:

$$\min_\theta \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y(i) - \theta x(i))^2 = \min_\theta \frac{1}{2\sigma^2} \sum_{i=1}^{n} (\epsilon(i))^2$$

This equals to least squares estimation ("pienimmän neliösumman menetelmä") because of the certain properties of ϵ in this problem.

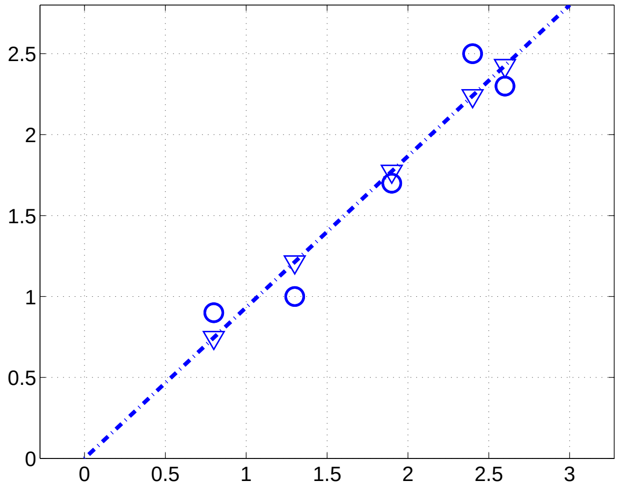
Minimum is fetched by setting the derivative w.r.t. θ to zero (the extreme point):

$$\begin{aligned} 0 &= \frac{\partial}{\partial\theta}\sum_{i=1}^{n}(y(i)-\theta x(i))^2 && (29)\\ &= \sum_{i=1}^{n}\big(2(y(i)-\theta x(i))(-x(i))\big) && (30)\\ &= -2\sum_{i=1}^{n}y(i)x(i)+2\theta\sum_{i=1}^{n}(x(i))^2 && (31)\end{aligned}$$

which gives finally the estimator $\hat{\theta}_{ML}$

$$\hat{\theta}_{ML} = \frac{\sum_{i=1}^{n}x(i)y(i)}{\sum_{i=1}^{n}x(i)^2} \quad (32)$$

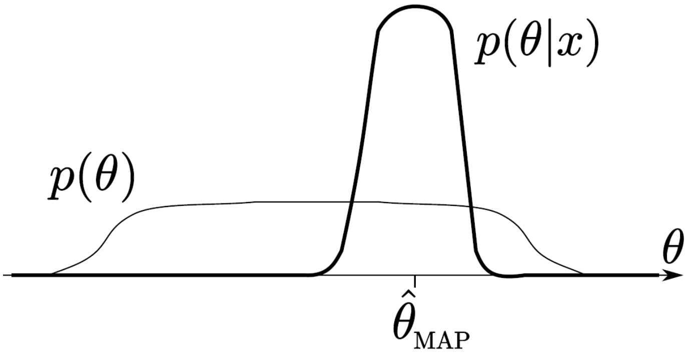
Example. Consider dataset $\mathbf{X} = \{(0.8,0.9)^T,(1.3,1.0)^T,(1.9,1.7)^T,(2.4,2.5)^T,(2.6,2.3)^T\}$. Now $\hat{\theta}_{ML} = 0.9334$, $f(x(i),\hat{\theta}_{ML}) = \{0.7467,1.2134,1.7734,2.2401,2.4268\}$, and $\sum_i(y(i)-f(x(i),\hat{\theta}_{ML}))^2 = 0.1580$.



H3 / Problem 2.
See lectures slides, chapter 5, and Problems H3/1, H2/3, and H2/4. Bayes rule is

$$\begin{aligned} p(\theta|x) &= \frac{p(x|\theta)p(\theta)}{p(x)} && (33)\\ p(\text{model}|\text{data}) &= \frac{p(\text{data}|\text{model})p(\text{model})}{p(\text{data})} && (34)\\ \text{posterior} &\propto \text{likelihood} \times \text{ prior} && (35)\end{aligned}$$

The parameters are now variables with densities. Prior gives us belief what the parameters probably are before seeing any data. After seeing data (likelihood) we have more exact information about parameters.



Often only the maximum posterior estimate of θ (MAP) is computed. Taking logarithm gives $\ln p(\theta|x) = \ln p(x|\theta) + \ln p(\theta) - \ln p(x)$, and the derivative w.r.t. θ is set to zero: $\frac{\partial}{\partial\theta}\ln p(x|\theta) + \frac{\partial}{\partial\theta}\ln p(\theta) = 0$. Compared to ML-estimation (Problem 1), there is an extra term $\frac{\partial}{\partial\theta}\ln p(\theta)$.

In this problem we have also a data set $\mathbf{X}$ and now two variables θ and α to be estimated. The model is $y(i) = \alpha + \theta x(i) + \epsilon(i)$, where $\epsilon \sim N(0,\sigma)$ as in Problem 1. Now $E(y(i)|x(i),\alpha,\theta) = \alpha + \theta x(i)$, and $Var(y(i)|x(i),\alpha,\theta) = Var(\epsilon) = \sigma^2$. Thus $y(i) \sim N(\alpha + \theta x(i), \sigma)$ and the likelihood function is

$$\begin{aligned} L(\alpha,\theta) = \prod_i p(y(i)|x(i),\alpha,\theta) &= \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(y(i)-\alpha-\theta x(i))^2}{2\sigma^2}} && (36)\\ \ln L(\alpha,\theta) = \ln \prod_i p(y(i)|x(i),\alpha,\theta) &= \text{const} - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y(i)-\alpha-\theta x(i))^2 && (37)\end{aligned}$$

Parameters have also normal density functions (“prior densities”)

$$\begin{aligned} \alpha \sim N(0,0.1) \quad &\rightarrow \quad p(\alpha) = \frac{1}{\sqrt{2\pi}\cdot 0.1}e^{-\frac{(\alpha-0)^2}{2\cdot 0.1^2}} = \text{const}\cdot e^{-50\alpha^2} && (38)\\ \theta \sim N(1,0.5) \quad &\rightarrow \quad p(\theta) = \frac{1}{\sqrt{2\pi}\cdot 0.5}e^{-\frac{(\theta-1)^2}{2\cdot 0.5^2}} = \text{const}\cdot e^{-2(\theta-1)^2} && (39)\end{aligned}$$

In Bayes MAP-estimation the log posterior probability to be maximized is $\ln p(x,y|\alpha,\theta) + \ln p(\alpha) + \ln p(\theta)$, where the first term is the likelihood and the two latter terms prior densities:

$$\begin{aligned} \ln p(\alpha) &= \text{const} - 50\alpha^2 && (40)\\ \ln p(\theta) &= \text{const} - 2(\theta-1)^2 && (41)\end{aligned}$$

Hence, the task is

$$(\hat{\alpha},\hat{\theta}) = \arg\max_{\alpha,\theta}\Big\{(-\frac{1}{2\sigma^2})\sum_{i=1}^{n}\big[(y(i)-\alpha-\theta x(i))^2\big] - 50\alpha^2 - 2(\theta-1)^2\Big\} \qquad (42)$$

First, maximize w.r.t. α,

$$\begin{aligned} 0 &= \frac{\partial}{\partial\alpha}(-\frac{1}{2\sigma^2})\sum_{i=1}^{n}\big[(y(i)-\alpha-\theta x(i))^2\big] - 50\alpha^2 - 2(\theta-1)^2 && (43)\\ &= (-\frac{1}{2\sigma^2})\sum_i\big[2\cdot(y(i)-\alpha-\theta x(i))\cdot(-1)\big] - 100\alpha && (44)\\ &= \sum_i y(i) - n\alpha - \theta\sum_i x(i) - 100\sigma^2\alpha && (45)\\ \hat{\alpha}_{MAP} &= \frac{\sum_i y(i) - \theta\sum_i x(i)}{n+100\sigma^2} && (46)\end{aligned}$$

and similarly θ, using previous result of α,

$$\begin{aligned}
0 &= \frac{\partial}{\partial\theta}(-\frac{1}{2\sigma^2})\sum_{i=1}^{n}\big[(y(i)-\alpha-\theta x(i))^2\big]-50\alpha^2-2(\theta-1)^2 && (47)\\
&= (-\frac{1}{2\sigma^2})\sum_i\big[2\cdot(y(i)-\alpha-\theta x(i))\cdot(-x(i))\big]-4(\theta-1) && (48)\\
&= \sum_i\big[y(i)x(i)-\alpha x(i)-\theta x(i)^2\big]-4\sigma^2(\theta-1) \qquad \mid \quad \alpha \leftarrow \hat{\alpha}_{MAP} && (49)\\
&= \sum_i y(i)x(i)-\Big(\frac{\sum_i y(i)-\theta\sum_i x(i)}{n+100\sigma^2}\Big)\sum_i x(i)-\theta\sum_i x(i)^2-4\sigma^2\theta+4\sigma^2 && (50)\\
\hat{\theta}_{MAP} &= \frac{\sum_i y(i)x(i)-\frac{(\sum_i y(i))(\sum_i x(i))}{n+100\sigma^2}+4\sigma^2}{\sum_i x(i)^2-\frac{(\sum x(i))^2}{n+100\sigma^2}+4\sigma^2} && (51)
\end{aligned}$$

Some interpretations of the results. If $\sigma^2=0$:

$$\begin{aligned}
\theta &= \frac{\sum_i y(i)x(i)-\frac{(\sum_i y(i))(\sum_i x(i))}{n}}{\sum_i x(i)^2-\frac{(\sum x(i))^2}{n}} && (52)\\
&= (1/n)\cdot\frac{\sum_i y(i)x(i)-((1/n)\cdot(\sum_i y(i)))((1/n)\cdot(\sum_i x(i)))}{(1/n)\cdot\sum_i x(i)^2-((1/n)\sum x(i))^2} && (53)\\
&= \frac{E(YX)-E(Y)E(X)}{E(X^2)-(E(X))^2} && (54)\\
&= \frac{Cov(X,Y)}{Var(X)} && (55)\\
\alpha &= (1/n)\sum_i y(i)-\theta(1/n)\sum_i x(i) && (56)\\
&= E(Y)-\theta E(X) && (57)
\end{aligned}$$

which are also the estimates of PNS method as well as by least squares.

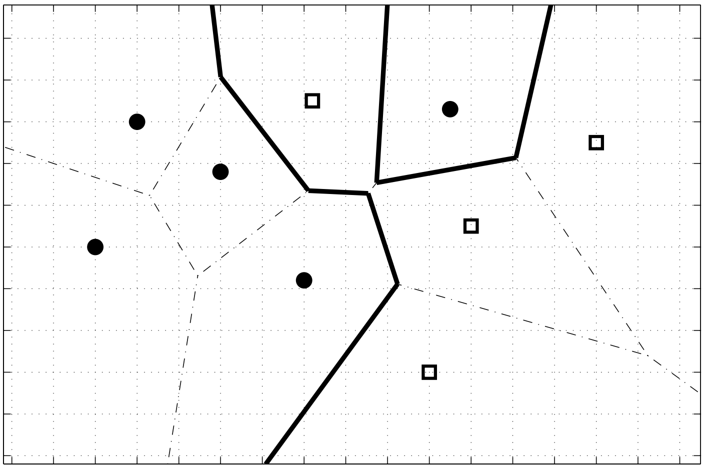
If $\sigma^2\to\infty$:

$$\begin{aligned}
\theta &\to 4/4=1 && (58)\\
\alpha &= \frac{\sum_i y(i)-\theta\sum_i x(i)}{n+100\sigma^2} && (59)\\
&\to 0 && (60)
\end{aligned}$$

then it is better to believe in the prior information.

H3 / Problem 3.

Using Euclidean distance $d(\mathbf{a}, \mathbf{b}) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$ (taking square root not necessary) we get (a) 1-NN: closest neighbour is square, $\mathbf{x}$ is classified as a square, (b) 3-NN: three closest: square, circle, circle, $\mathbf{x}$ is classified as a circle. See also T3 computer session. 1-NN border plotted with a thick line:



H3 / Problem 4.

Bayes rule

$$p(\omega|x) = \frac{p(x|\omega)p(\omega)}{p(x)}$$

Classification rule: when having observation x, choose class ω_1 if

$$p(\omega_1|x) > p(\omega_2|x) \quad \Leftrightarrow \quad \frac{p(x|\omega_1)p(\omega_1)}{p(x)} > \frac{p(x|\omega_2)p(\omega_2)}{p(x)} \quad \Leftrightarrow \quad p(x|\omega_1)p(\omega_1) > p(x|\omega_2)p(\omega_2)$$

Now the both data follow the normal distribution $x|\omega_1 \sim N(0, \sigma_1)$ and $x|\omega_2 \sim N(0, \sigma_2)$. Assume that $\sigma_1^2 > \sigma_2^2$. The density function of a normal distribution with mean μ and variance σ^2 is

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Now the rule is

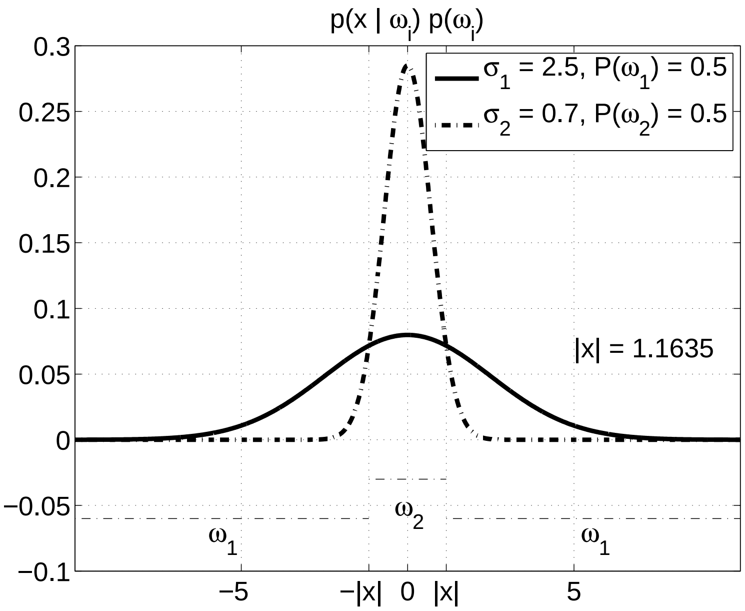
$$\frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{x^2}{2\sigma_1^2}} p(\omega_1) \quad > \quad \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{x^2}{2\sigma_2^2}} p(\omega_2) \tag{61}$$

$$\frac{e^{-\frac{x^2}{2\sigma_1^2}}}{e^{-\frac{x^2}{2\sigma_2^2}}} \quad > \quad \frac{\sigma_1}{\sigma_2} \frac{p(\omega_2)}{p(\omega_1)} \qquad | \text{ ln on both sides} \tag{62}$$

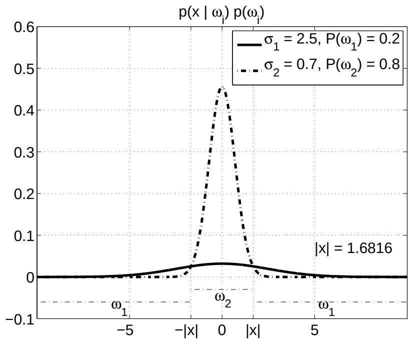
$$\Big(\frac{1}{2\sigma_2^2} - \frac{1}{2\sigma_1^2}\Big) x^2 \quad > \quad \ln\Big(\frac{\sigma_1}{\sigma_2} \frac{p(\omega_2)}{p(\omega_1)}\Big) \tag{63}$$

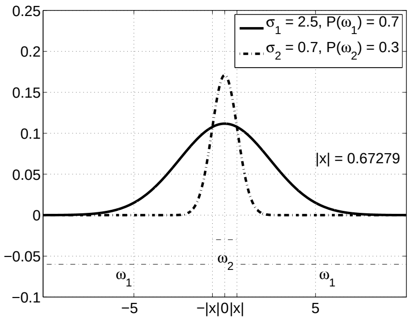
$$x^2 \quad > \quad \frac{2 \ln(\frac{\sigma_1}{\sigma_2} \frac{p(\omega_2)}{p(\omega_1)})}{(\frac{1}{\sigma_2^2} - \frac{1}{\sigma_1^2})} \tag{64}$$

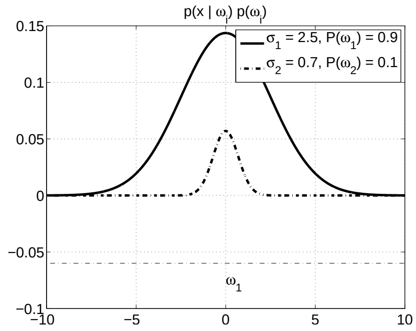
In the figure below the density functions and class borders when using sample values $\sigma_1 = 2.5$, $\sigma_2 = 0.7$, $P(\omega_1) = 0.5$, and $P(\omega_2) = 0.5$, yielding $x^2 > 1.3536$ and decision borders $|x| = 1.1635$. E.g., if we are given a data point $x = 2$, we choose the class ω_1.



However, if the class probabilities $P(\omega_i)$ differ, then the optimal border changes. Below there are three other examples. Assume that only 20% / 70% / 90% of samples are from class ω_1, i.e., $P(\omega_1) = \{0.2, 0.7, 0.9\}$ and $P(\omega_2) = \{0.8, 0.3, 0.1\}$. In the last case data samples from class 2 are so rare that the classifier chooses always class 1.







Round 4 [**Fri 25.11.2011, Mon 28.11.2011**]

H4 / 1. (Cluster analysis)

We are given n vectors. In how many ways can we divide them into two clusters (groups)? Solve at least the cases $n = 2, 3, 4, 5$.

H4 / 2. (Cluster analysis)

We are given the following data matrix:

$$\mathbf{X} = \begin{bmatrix} 0 & 1 & 2.5 & 3 & 3 & 5 \\ 0 & 1 & 2.5 & 2 & 4 & 3 \end{bmatrix}$$

a) Plot the data vectors in the coordinate plane.

b) Perform a hierarchical clustering based on your plot. As the distance between two clusters, use the smallest distance between two vectors belonging to the two clusters. Plot the clustering tree. What is the best clustering into three clusters ?

H4 / 3. (Cluster analysis)

We are given three vectors $\mathbf{x}, \mathbf{z}_1, \mathbf{z}_2$. In the beginning $C_1 = \{\mathbf{x}\}$, $C_2 = \{\mathbf{z}_1, \mathbf{z}_2\}$.

a) Derive the means $\mathbf{m}_1$, $\mathbf{m}_2$ of the two clusters.

b) It turns out that $\|\mathbf{z}_1 - \mathbf{m}_1\| < \|\mathbf{z}_1 - \mathbf{m}_2\|$ and thus in the c-means-algorithm the vector $\mathbf{z}_1$ is moved from cluster C_2 into the cluster C_1. Denote the new clusters by $C_1' = \{\mathbf{x}, \mathbf{z}_1\}$, $C_2' = \{\mathbf{z}_2\}$. Derive the new means $\mathbf{m}_1'$, $\mathbf{m}_2'$.

c) Prove that

$$\sum_{\mathbf{x} \in C_1} \|\mathbf{x} - \mathbf{m}_1\|^2 + \sum_{\mathbf{x} \in C_2} \|\mathbf{x} - \mathbf{m}_2\|^2 > \sum_{\mathbf{x} \in C_1'} \|\mathbf{x} - \mathbf{m}_1'\|^2 + \sum_{\mathbf{x} \in C_2'} \|\mathbf{x} - \mathbf{m}_2'\|^2$$

meaning that the criterion used in the c-means clustering is decreasing.

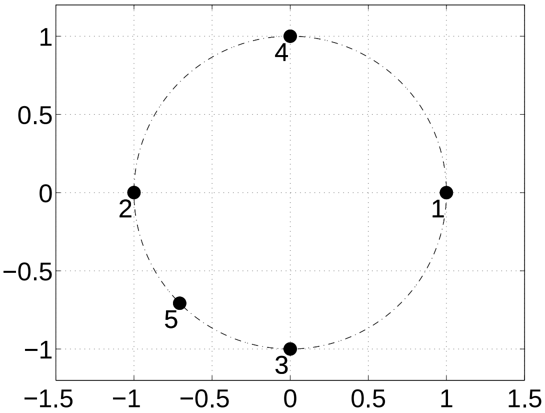
H4 / 4. (SOM)

Let us consider the computational complexity of the SOM algorithm. Assume that the size of the map is $N \times N$ units (neurons), and the dimension of the input and weight vectors is d. How many additions and multiplications are needed when the winner neuron is found for an input vector $\mathbf{x}$, when the Euclidean distance is used ?

H4 / 5. (SOM)

Let us assume that the weight vectors $\mathbf{m}_i$ and input vectors $\mathbf{x}$ of the SOM are on the unit circle (they are 2-dimensional unit vectors). The map is a 1-dimensional grid of 5 units whose weight vectors are initially as shown in Fig. 1. The neighborhood is defined cyclically so that the neighbors of units $b = 2, 3, 4$ are $b-1, b+1$, those of unit 5 are 4 and 1, and those of unit 1 are 5 and 2.

In the training, the coefficient $\alpha = 0.5$, meaning that at each step the weight vectors of the winner unit and its neighbors move along the unit circle halfway towards the input $\mathbf{x}$. You are free to choose your inputs freely from the unit circle. Choose a sequence of training vectors $\mathbf{x}$ so that the weight vectors become ordered.



H4 / Problem 1.

Case $n = 2$. There are two vectors $\{1, 2\}$. Only one possibility,

$$C_1 = \{1\}, C_2 = \{2\}$$

Case $n = 3$. There are three vectors $\{1, 2, 3\}$. There are three possible groupings,

$$\begin{aligned} C_1 = \{1\}, C_2 = \{2, 3\}, &\text{ or} \\ C_1 = \{2\}, C_2 = \{1, 3\}, &\text{ or} \\ C_1 = \{3\}, C_2 = \{1, 2\}. & \end{aligned}$$

Case $n = 4$. There are four vectors $\{1, 2, 3, 4\}$. There are seven possible groupings,

$$\begin{aligned} C_1 = \{1\}, C_2 = \{2, 3, 4\}, &\text{ or} \\ C_1 = \{2\}, C_2 = \{1, 3, 4\}, &\text{ or} \\ C_1 = \{3\}, C_2 = \{1, 2, 4\}, &\text{ or} \\ C_1 = \{4\}, C_2 = \{1, 2, 3\}, &\text{ or} \\ C_1 = \{1, 2\}, C_2 = \{3, 4\}, &\text{ or} \\ C_1 = \{1, 3\}, C_2 = \{2, 4\}, &\text{ or} \\ C_1 = \{1, 4\}, C_2 = \{2, 3\}. & \end{aligned}$$

For $n = 5$ there are $5 + 4 + 3 + 2 + 1 = 15$ possible groupings. It seems that the number of groupings for n points is $2^{n-1} - 1$.

Let us prove that the number is $2^{n-1} - 1$. Take a binary vector of length n such that its i-th element

$$b_i \quad = \quad \begin{cases} 0, & \text{if } i\text{-th point is first cluster} \\ 1, & \text{if } i\text{-th point is second cluster} \end{cases}$$

All possible combinations are allowed except $b_i = 0$ for all i, $b_i = 1$ for all i, because then there is only one cluster. Thus the number is $2^n - 2$ (there are 2^n different binary vectors of length n).

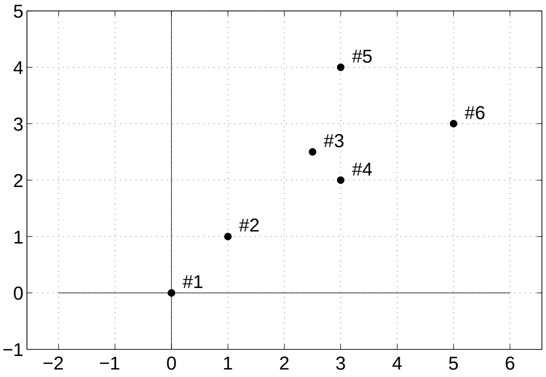
But one half are equivalent to the other half because “first” and “second” cluster can be changed (consider case $n = 2$).

The final number is $\frac{1}{2}(2^n - 2) = 2^{n-1} - 1$.

H4 / Problem 2.

See also c-means clustering and hierarchical clustering examples in computer session T4.

Here we use hierarchical clustering and a dendrogram ("ryhmittelypuu"). Clusters are combined using the nearest distance (often "single linkage"). In the beginning each data point is a cluster. Then clusters are combined one by one, and a dendrogram is drawn. When all clusters are combined to one single cluster and the dendrogram is ready, one can choose where to cut the dendrogram.



In the beginning there are six clusters

$$\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}$$

Items 3 and 4 are nearest and combined

$$\{1\}, \{2\}, \{3, 4\}, \{5\}, \{6\}$$

Then the nearest clusters are 1 and 2

$$\{1, 2\}, \{3, 4\}, \{5\}, \{6\}$$

Next, 5 is connected to the cluster $\{3, 4\}$, because the distance from 5 to 3 (nearest) is smallest

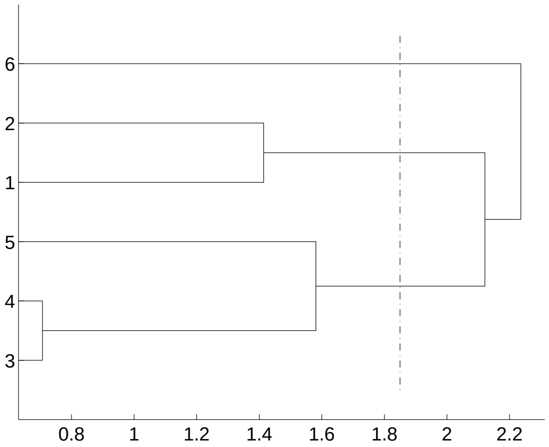
$$\{1, 2\}, \{3, 4, 5\}, \{6\}$$

Note that distance between 2 and 3 is smaller that of 6 to 4 or 5, and therefore

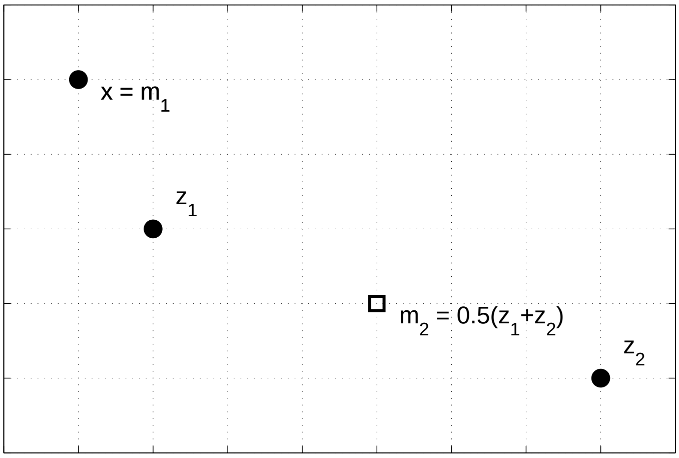
$$\{1, 2, 3, 4, 5\}, \{6\}$$

The algorithm ends when all points/clusters are combined to one big cluster.

The result can be visualized using the dendrogram, see the figure below. The x-axis gives the distance of the combined clusters. The best choice for three clusters is $\{1, 2\}$, $\{3, 4, 5\}$, $\{6\}$.



H4 / Problem 3.



Now $||\mathbf{z}_1 - \mathbf{m}_1|| < ||\mathbf{z}_1 - \mathbf{m}_2||$ and so $\mathbf{z}$ moves together with $\mathbf{x}$. New centers are:

$$\mathbf{m}_1' = 0.5(\mathbf{x} + \mathbf{z}_1), \qquad \mathbf{m}_2' = \mathbf{z}_2$$

$$\begin{aligned}
J^{OLD} &= ||\mathbf{z}_1 - \mathbf{m}_2||^2 + ||\mathbf{z}_2 - \mathbf{m}_2||^2 + \underbrace{||\mathbf{x} - \mathbf{m}_1||^2}_{0} \\
&= ||\mathbf{z}_1 - 0.5(\mathbf{z}_1 + \mathbf{z}_2)||^2 + ||\mathbf{z}_2 - 0.5(\mathbf{z}_1 + \mathbf{z}_2)||^2 \\
&= 0.25||\mathbf{z}_1 - \mathbf{z}_2||^2 + 0.25||\mathbf{z}_1 - \mathbf{z}_2||^2 \\
&= 0.5||\mathbf{z}_1 - \mathbf{z}_2||^2 \\
J^{NEW} &= ||\mathbf{z}_1 - \mathbf{m}_1'||^2 + \underbrace{||\mathbf{z}_2 - \mathbf{m}_2'||^2}_{0} + ||\mathbf{x} - \mathbf{m}_1'||^2 \\
&= 0.5||\mathbf{x} - \mathbf{z}_1||^2
\end{aligned}$$

Now we remember that $||\mathbf{z}_1 - \mathbf{m}_1||^2 < ||\mathbf{z}_1 - \mathbf{m}_2||^2$ (that is why $\mathbf{z}_1$ moved to the other cluster).

$$\Rightarrow ||\mathbf{z}_1 - \underbrace{\mathbf{x}}_{\mathbf{m}_1}||^2 < ||\mathbf{z}_1 - \underbrace{0.5(\mathbf{z}_1 + \mathbf{z}_2)}_{\mathbf{m}_2}||^2 = 0.25||\mathbf{z}_1 - \mathbf{z}_2||^2$$

So,

$$J^{NEW} = 0.5||\mathbf{x} - \mathbf{z}_1||^2 < 0.5 \cdot 0.25||\mathbf{z}_1 - \mathbf{z}_2||^2 < 0.5||\mathbf{z}_1 - \mathbf{z}_2||^2 = J^{OLD}$$

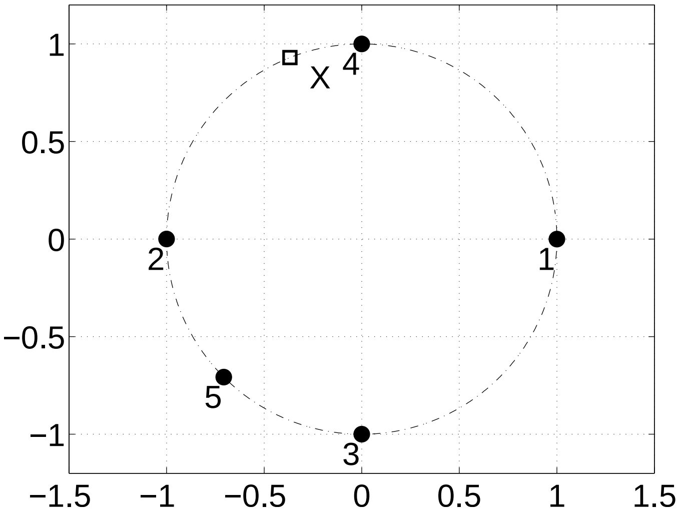
H4 / Problem 4.

Number or neurons is N^2. For each neuron j, whe have to compute

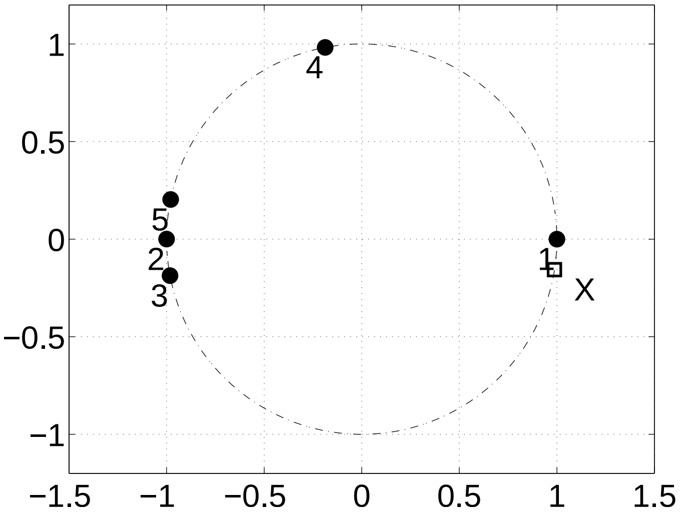
$$\sum_{i=1}^{d} (x_i - m_{ij})^2$$

which takes d subtractions, d multiplications, $d - 1$ additions. This means totally $N^2(2d - 1)$ additions (subtraction and addition are usually equivalent) and $N^2 d$ multiplications.

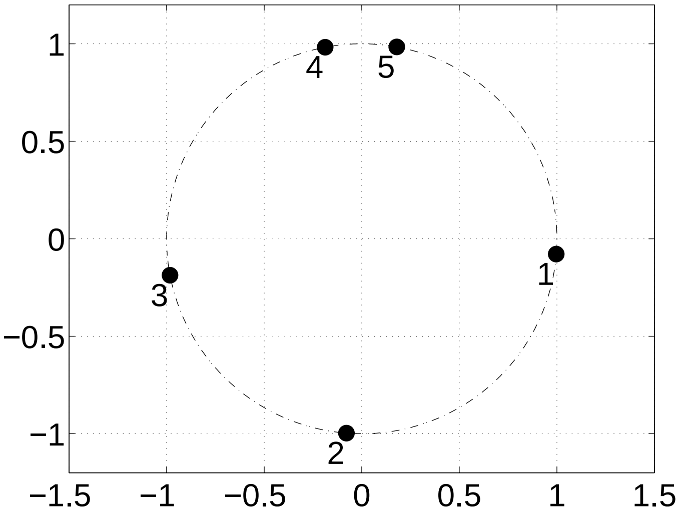
H4 / Problem 5.
Choose **x** so that its angle is a little less than 135°.



Now best matching unit (BMU): 4, neighbours: 5 and 3. They move on the circle half-way towards **x**.



Now choose x so that its angle is very small negative. BMU: 1, neighbours: 5 and 2. They are moving closer to **x** along unit circle. 5 jumps over 4, and 2 jumps over 3. Now 1D SOM is in order: 1, 2, 3, 4, 5.



Round 5 [**Fri 2.12.2011, Mon 5.12.2011**]

H5 / 1. (Frequent itemsets)

Consider 0-1 observation set

a	0	1	1	1	1	0	0	1	1	1
b	1	1	0	1	0	1	1	0	0	1
c	0	0	0	1	1	1	1	1	1	1
d	1	1	0	1	1	1	1	0	0	1

There are four variables a, b, c, d and ten observations. Find the frequent itemsets when threshold value is $N = 4$.

H5 / 2. (Levelwise algorithm)

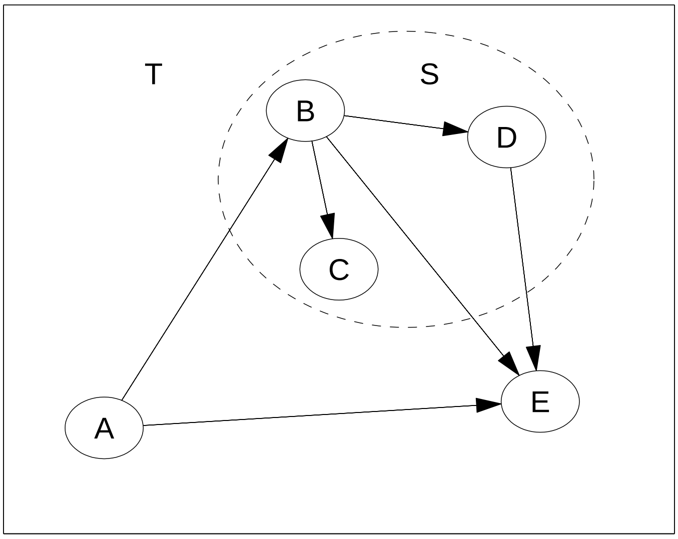
What is the time complexity of levelwise algorithm as a function of the size of data and the number of examined candidate sets?

H5 / 3. (Tšernov border)

Examine the Tšernov (Chernoff) border mentioned in the lectures. How does the border behave as a function of different parameters?

H5 / 4. (Hubs and authorities)

Consider a directed graph below depicting links in web pages $s = 1, 2, 3, 4, 5$ (named 'A', 'B', 'C', 'D', 'E'). Apply "hubs and authorities" algorithm to find good hubs (collection pages) and authorities (refered pages). Initialize hub weights k_s of each page s as $k_s = 1/\sqrt{5} \approx 0.447$, and authority weights similarly $a_s = 1/\sqrt{5} \approx 0.447$. Iterate weights until the change is small. Explain the result.



H5 / Problem 1.

Simulate the levelwise algorithm. In the first phase the candidates are all sets of one variable $\{a\}$, $\{b\}$, $\{c\}$ ja $\{d\}$. To be more convenient, we will omit all { and } from now on, and write all sets simply a, b, c, and d. The frequencies of these

a	b	c	d
7	6	7	7.

Frequencies of all sets are equal or more than the threshold, so all sets are frequent. Now the following level candidates are all sets of two variables (again $ab = \{a, b\}$ and so on):

ab	ac	ad	bc	bd	cd
3	5	4	4	6	5.

All sets except ab are frequent. The candidates of 3-size sets are

acd	bcd
3	4.

Here only bcd is frequent. Therefore any larger set (in this case $abcd$) cannot be frequent and algorithm stops.

The frequent itemsets are a, b, c, d, ac, ad, bc, bd, cd, and bcd. (Often the empty set is also considered to be a frequent set.)

You can consider, e.g., observations as bags (customers), and variables as products in a supermarket, for example, a is for apples, b is for bread, c is for cheese, and d is for soda. In the 0-1-matrix each 1 means that the particular item is found in the shopping bag. The first customer has bought bread and soda, the last tenth customer all four products.

H5 / Problem 2.

When computing time complexities of algorithms it is interesting to see the asymptotic behavior of algorithms, that is, when the size of input grows to infinity. In this case time complexity is examined as a function of both input size and number of candidates. The latter connection is more difficult to explain. If the number of candidates were not taken into account, the worst case would be trivially that where the data contains only 1s. In that case all possible variables sets would become candidates, that is exponential case.

The levelwise algorithm shown in the lectures is written with pseudocode below. Let us call the size of data (number of observations) with m, and the number of all processed candidate sets with n. Candidate sets with k size candidate is marked C_k. Let t be the biggest value of k, i.e., the maximum size of candidates. Clearly, $n = \sum_{k=1}^{t} |C_k|$ and $k \leq t = O(\ln n)$

While-loop in row 3 is executed t times. At one execution for-loop in row 5 is executed m times, and at one execution step of that for-loop in row 6 is executed $|C_k|$ times. Totally, this for-loop is executed mn times. At one execution the for-loop in row 8 is computed k times, and those operations can be considered as taking a constant time. As well the if-statement in row 11 takes a constant time. Hence, the time complexity of the for-loop in row 5 is $O(mn \ln n)$.

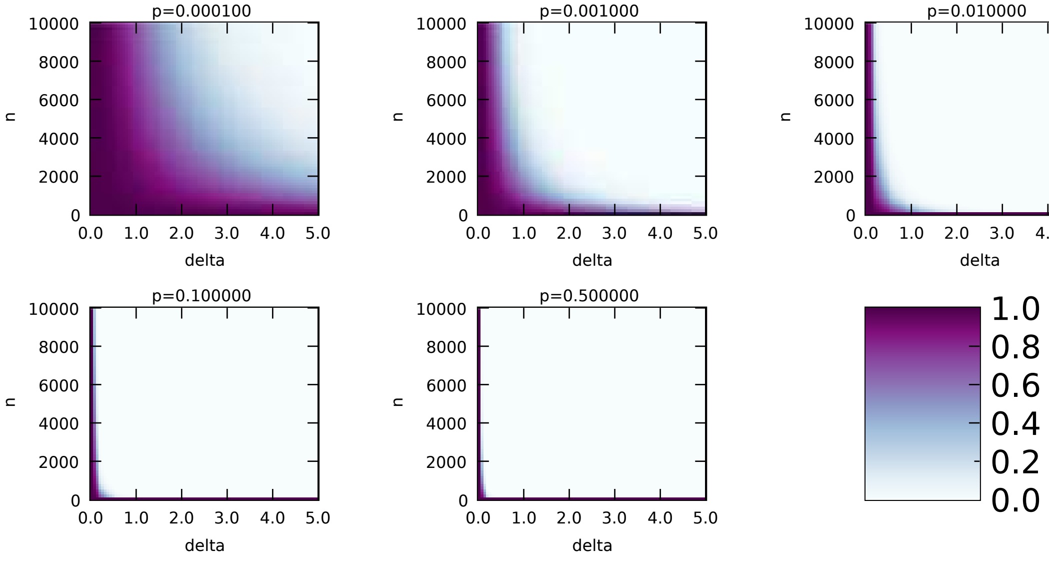
```
 1: $k \leftarrow 1$
 2: $C_k \leftarrow \{ \{ a \} \mid a \in \text{variables} \}$
 3: while $C_k \neq \emptyset$ do
 4:     counter[$X$] $\leftarrow 0$ for all $X$
 5:     for observation in data do                         ▷ Count frequencies of candidates
 6:         for $X$ in $C_k$ do                            ▷ Check if all variables in $X$ are present
 7:             good $\leftarrow$ True
 8:             for var in $X$ do
 9:                 if observation[var] $= 0$ then
10:                     good $\leftarrow$ False
11:             if good then
12:                 counter[$X$] $\leftarrow$ counter[$X$] $+ 1$
13:     $F_k \leftarrow \emptyset$
14:     for $X$ in $C_k$ do                                ▷ Select frequent candidates
15:         if counter[$X$] $\geq N$ then
16:             $F_k \leftarrow F_k \cup \{ X \}$
17:     $C_{k+1} \leftarrow \emptyset$
18:     for $A$ in $F_k$ do                                ▷ Generate next candidates
19:         for $B$ in $F_k$ do
20:             $X \leftarrow A \cup B$
21:             if $|X| = k + 1$ then
22:                 good $\leftarrow$ True
23:                 for var in $X$ do
24:                     if $X \setminus \{ \text{var} \}$ not in $F_k$ then
25:                         good $\leftarrow$ False
26:                 if good then
27:                     $C_{k+1} \leftarrow C_{k+1} \cup \{ X \}$
28:     $k \leftarrow k + 1$
```

The for-loop in row 14 is executed n times and the lines inside it have constant times. The time complexity for rows 13–17 is $O(n)$, and becausedd $n = O(mn \ln n)$, it has not asymptotical meaning.

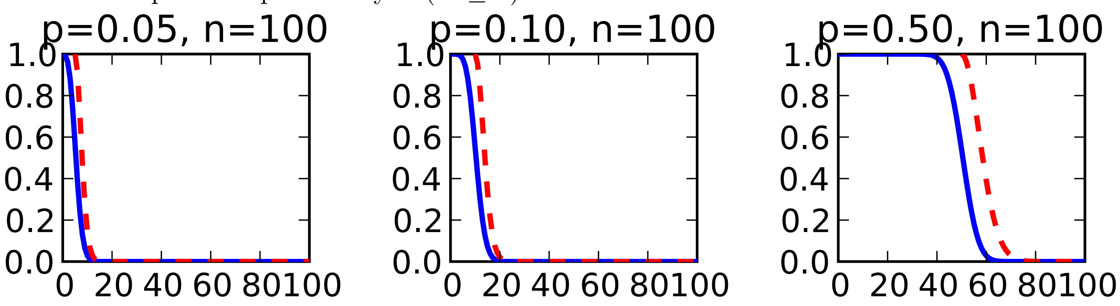
For-loops in rows 18 and 19 are executed totally $t|F_k|^2 \leq t|C_k|^2 = O(n^2 \ln n)$ times. The statement in row 20 takes at most $O(2k) = O(\ln n)$. The for-loop in row 23 is executed $k + 1 = O(\ln n)$ times, and the lines inside it as constants (F_k can be implemented with hash tables where testing is practically constant-time). The for-loop in row 18 is therefore $O(n^2 (\ln n)^2)$. Because $mn \ln n$ and $n^2 (\ln n)^2$ are not asymptotically comparable, the whole time complexity of the algorithm is $O(mn \ln n + n^2 (\ln n)^2)$.

H5 / Problem 3.

The following figures show the behavior of the border.



The distance of the border and true probability can be examined with small enough values of n. In the following the solid curve depicts the probability $\Pr(X \geq x)$ and dashed line Tšernov border:



Tšernov border can be shown quite easily. Consider *Markov's inequality*, which deals non-negative random variables X and for which

$$\Pr(X \geq a) \leq \frac{E[X]}{a}.$$

In this case it is needed for discrete random variables. Let possible values of X be $x_1 < x_2 < \cdots$. Let x_j be the biggest of those values, which is smaller than a. Then

$$E[x] = \sum_{i=1}^{\infty} x_i \Pr(X = x_i) \geq \sum_{i=j+1}^{\infty} x_i \Pr(X = x_i)$$

$$\geq a \sum_{i=j+1}^{\infty} \Pr(X = x_i) = a \Pr(X \geq x_{j+1}) = a \Pr(X \geq a).$$

The other needed result is the inequality $1 + x \leq e^x$, which hold for all real values x and can be seen true by examining the function $f(x) = e^x - x - 1$: because $f''(x) = e^x > 0$ and $f'(0) = 0$, $f(x) \geq f(0) = 0$ for all x.

Let $Y_1, Y_2, \ldots, Y_n$ be independent random variables, where each gets the value 1 with probability p and the value 0 with probability $1-p$, and let $X = \sum_{i=1}^{n} Y_i$. Let $t \geq 0$. Now using Markov's inequality

$$\Pr(X \geq (1+\delta)np) \leq \Pr(e^{tX} \geq e^{t(1+\delta)np}) \leq e^{-t(1+\delta)np} E[e^{tX}].$$

For the expectation of variable e^{tX} holds (independence)

$$E[e^{tX}] = E[e^{t(Y_1+\cdots+Y_n)}] = E[e^{tY_1} \cdots e^{tY_n}] = E[e^{tY_1}]^n.$$

Using the definition of expectation and the inequality proved above, we get

$$E[e^{tY_1}] = pe^t + (1-p) = 1 + p(e^t - 1) \leq e^{p(e^t-1)}.$$

By choosing $t = \ln(1+\delta)$ we get

$$\Pr(X \geq (1+\delta)np) \leq e^{-\ln(1+\delta)(1+\delta)np} e^{p(1+\delta-1)n} = (1+\delta)^{-(1+\delta)np} e^{\delta np} = \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{np}.$$

H5 / Problem 4.

Hubs and authorities algorithm for finding relevant web pages. The main idea is that a good hub points to good authorities and there are links coming to good authorities from good hubs. The "circular statement" is solved with iteration. The algorithm:

1. Find pages with requested key words
2. Choose best N pages (heuristically); name this set as S
3. Create a set T so that it includes all pages of S, all pages that point to any page of S, and all pages that are pointed from any page of S
4. Create a set of arcs E: $(u, v) \in E$ when there is a link from page u to page v.
5. Assign a hub weight k_s and an authority weight a_s for each page $s \in T$
6. Initialize all weights $a_s = k_s = 1/\sqrt{n}$ where $n = |T|$
7. Iteratively update the weights until the change is small
   (a) sum
   $$k_s \leftarrow \sum_{t \in T, (s,t) \in E} a_t, \qquad a_s \leftarrow \sum_{t \in T, (t,s) \in E} k_t$$
   (b) scale to unity so that $\sum_s a_s^2 = 1$ and $\sum_s k_s^2 = 1$ □

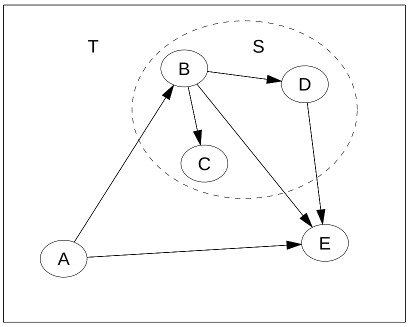
The weights can be thought as vectors $\mathbf{a} = \begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{k} = \begin{bmatrix} k_1 & k_2 & \dots & k_n \end{bmatrix}^{\mathrm{T}}$ and the graph and its arcs E as a square matrix M of size $(n \times n)$ where $M(s,t) = 1$ if $(s,t) \in E$. Now the final step iterations can be written as

$$\mathbf{a} \leftarrow M^T \mathbf{k}, \qquad \mathbf{k} \leftarrow M\mathbf{a}$$

and the scaling as

$$\mathbf{a} \leftarrow \mathbf{a}/\|\mathbf{a}\|, \qquad \mathbf{k} \leftarrow \mathbf{k}/\|\mathbf{k}\|$$

In this problem we have five webpages 'A', 'B', 'C', 'D', 'E', with indices $s = 1, 2, 3, 4, 5$, which have links (e.g., `<a href=''http://www.aalto.fi/''>`).



Read carefully what is written in Step 7: $k_s \leftarrow \sum_{t \in T, (s,t) \in E} a_t$. For instance, when computing k_s for node s ='B' we pick from all values a_t ($T \in$ {'A','B','C','D','E'}) only those there is an arc (s,t) ($E \in$ {{ 'B'→'C' }, { 'B'→'D' }, { 'B'→'E' }}). Therefore $k_{\text{'B'}} = a_{\text{'C'}} + a_{\text{'D'}} + a_{\text{'E'}}$.

We can write the data matrix M of size $(n \times n)$ (having 1s in the diagonal):

$$M = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Assign weights a_s (authority) and k_s (hub) for each page s, and write them as vectors $\mathbf{a}$ and $\mathbf{k}$. Initialize them so that $\sum_s a_s^2 = 1$ and $\sum_s k_s^2 = 1$.

$$\mathbf{a} \;=\; \frac{1}{\sqrt{5}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \qquad \mathbf{k} = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Iteration (as long as needed) can be done by

$$\mathbf{a}_{new} = M^T\mathbf{k} \quad \Leftrightarrow \quad \mathbf{a}_{new}^T = \mathbf{k}^T M \quad , \qquad \mathbf{a} \leftarrow \mathbf{a}_{new}/\sqrt{\sum_s a_{new,s}^2}$$

$$\mathbf{k}_{new} = M\mathbf{a} \quad , \qquad \mathbf{k} \leftarrow \mathbf{k}_{new}/\sqrt{\sum_s k_{new,s}^2}$$

1st round:

$$\mathbf{a}_{new}^T \approx \begin{bmatrix}0.447 & 0.447 & 0.447 & 0.447 & 0.447\end{bmatrix} \cdot \begin{bmatrix}0&1&0&0&1\\0&0&1&1&1\\0&0&0&0&0\\0&0&0&0&1\\0&0&0&0&0\end{bmatrix} = \begin{bmatrix}0 & 0.447 & 0.447 & 0.447 & 1.342\end{bmatrix}$$

$$\mathbf{k}_{new} \approx \begin{bmatrix}0&1&0&0&1\\0&0&1&1&1\\0&0&0&0&0\\0&0&0&0&1\\0&0&0&0&0\end{bmatrix} \cdot \begin{bmatrix}0.447\\0.447\\0.447\\0.447\\0.447\end{bmatrix} = \begin{bmatrix}0.894\\1.342\\0\\0.447\\0\end{bmatrix}$$

$$\mathbf{a} \approx \mathbf{a}_{new}/\sqrt{0+0.447^2+0.447^2+0.447^2+1.342^2} \approx \begin{bmatrix}0\\0.289\\0.289\\0.289\\0.867\end{bmatrix}$$

$$\mathbf{k} \approx \mathbf{k}_{new}/\sqrt{0.894^2+1.342^2+0+0.447^2+0} \approx \begin{bmatrix}0.535\\0.802\\0\\0.267\\0\end{bmatrix}$$

2nd round:

$$\mathbf{a}_{new}^T \approx \begin{bmatrix}0.535 & 0.802 & 0 & 0.267 & 0\end{bmatrix} \cdot \begin{bmatrix}0&1&0&0&1\\0&0&1&1&1\\0&0&0&0&0\\0&0&0&0&1\\0&0&0&0&0\end{bmatrix} = \begin{bmatrix}0 & 0.535 & 0.802 & 0.802 & 1.604\end{bmatrix}$$

$$\mathbf{k}_{new} \approx \begin{bmatrix}0&1&0&0&1\\0&0&1&1&1\\0&0&0&0&0\\0&0&0&0&1\\0&0&0&0&0\end{bmatrix} \cdot \begin{bmatrix}0\\0.289\\0.289\\0.289\\0.867\end{bmatrix} = \begin{bmatrix}1.155\\1.443\\0\\0.866\\0\end{bmatrix}$$

$$\mathbf{a} \approx \mathbf{a}_{new}/\sqrt{0+0.535^2+0.802^2+0.802^2+1.604^2} \approx \begin{bmatrix}0\\0.263\\0.394\\0.394\\0.788\end{bmatrix}$$

$$\mathbf{k} \approx \mathbf{k}_{new}/\sqrt{1.155^2+1.443^2+0+0.866^2+0} \approx \begin{bmatrix}0.566\\0.707\\0\\0.424\\0\end{bmatrix}$$

After fifteen rounds:

$$\mathbf{a} \approx \begin{bmatrix}0\\0.254\\0.368\\0.368\\0.815\end{bmatrix} \qquad \mathbf{k} \approx \begin{bmatrix}0.521\\0.756\\0\\0.397\\0\end{bmatrix}$$

Compared to initial weight values 0.447 it can be seen, for instance, that 'E' has a high value ($a_5 = 0.815$) for being a trusted authority which good hubs link to. In addition, 'B' seems to be a good collective page ($k_2 = 0.756$), which links to good authorities.

Remark: $\mathbf{a}$ and $\mathbf{k}$ can also be computed as eigenvectors of the biggest corresponding eigenvalue of $(M^T M)\mathbf{x} = \lambda\mathbf{x}$ and $(MM^T)\mathbf{x} = \lambda\mathbf{x}$, respectively. Using Matlab, `[V_a,D_a] = eig(M'*M)` and `[V_k,D_k] = eig(M*M')` give

$$\mathbf{a} \approx \begin{bmatrix} 0 \\ 0.254 \\ 0.368 \\ 0.368 \\ 0.815 \end{bmatrix} \qquad \mathbf{k} \approx \begin{bmatrix} 0.521 \\ 0.756 \\ 0 \\ 0.397 \\ 0 \end{bmatrix}$$

Datasta Tietoon, autumn 2011, bonus point exercises

Some problems (if ready):

P1) bonus exercise P1

P2) bonus exercise P2

P3) bonus exercise P3

P4) bonus exercise P4

P5) bonus exercise P5

Some solutions:

P1) solution to bonus exercise P1

P2) solution to bonus exercise P2

P3) solution to bonus exercise P3

P4) solution to bonus exercise P4

P5) solution to bonus exercise P5

Datasta Tietoon, autumn 2011, computer exercises 1-5

Download the associated files DT_TX.zip, where X = {1, 2, 3, 4, 5} in course Noppa pages.

Computer #1: convolution sum, filtering.
Computer #2: PCA.
Computer #3: Normal distribution. kNN.
Computer #4: Clustering.
Computer #5: Frequent sets. Hubs and authorities.

Datasta Tietoon, autumn 2011, Matlab commands

Remember also GNU Octave, `http://www.octave.org`.

- `help <function>`, `doc <function>`,
- `min`, `max`, `mean`, `median`, `std`, `var`, tunnuslukuja: minimi, maksimi, keskiarvo, mediaani, keskihajonta, varianssi
- `plot`, tyypillisiin piirtokomento lukujonolle (käyrä): yhdistetään pisteet suorilla viivoilla toisiinsa
- `title`, `xlabel`, `ylabel`, `legend`, piirtämiseen liittyviä komentoja: kuvan otsikko, akselien nimiä, käyrille nimet
- `conv`, konvoluutiosumma
- `fft`, diskreetti Fourier-muunnos (DFT)
- `freqz`, suotimen taajuusvaste
- `soundsc`, lukujonon soittaminen äänenä
- `eig`, ominaisarvot ja -vektorit
- `normpdf`, `exppdf`, gaussisen / eksponenttisen tiheysfunktion arvojen laskeminen
- `polyfit`, datan sovittaminen polynomikäyrällä
- `kmeans`, c-means-ryhmittely
- `pdist`, `linkage`, `dendrogram`, hierarkinen ryhmittely
- SOM Toolbox `http://www.cis.hut.fi/projects/somtoolbox/`

Datasta Tietoon, autumn 2011, kaavakokoelma

Comments and corrections to `t612010@ics.tkk.fi`.

- summamerkintä $\sum$ ("iso sigma")

  Esimerkki geometrinen sarjan summa ja osasumma.

  $\sum_{i=0}^{\infty} a^N = a^0 + a^1 + a^2 + \ldots = \frac{1}{1-a}, \quad |a| < 1$

  $\sum_{i=0}^{N} a^N = a^0 + a^1 + a^2 + \ldots a^N = \frac{1-a^{N+1}}{1-a}, \quad |a| < 1$

- tulomerkintä $\prod$ ("iso pii"). Katso H2/3.

  Esimerkki: lasketaan ns. uskottavuusfunktio n:lle datapisteelle käyttäen (1-ulotteisesta) gaussisesta tiheysfunktiota $p(x|\mu,\sigma)$

$$\begin{aligned}\prod_{i=1}^{n} p(x(i)|\mu,\sigma) &= p(x(1)|\mu,\sigma)\cdot p(x(2)|\mu,\sigma)\cdot\ldots\cdot p(x(n)|\mu,\sigma)\\ &= \frac{1}{\sqrt{2\pi}\,\sigma}\cdot e^{-\frac{(x(1)-\mu)^2}{2\sigma^2}}\cdot\frac{1}{\sqrt{2\pi}\,\sigma}\cdot e^{-\frac{(x(2)-\mu)^2}{2\sigma^2}}\cdot\ldots\cdot\frac{1}{\sqrt{2\pi}\,\sigma}\cdot e^{-\frac{(x(n)-\mu)^2}{2\sigma^2}}\\ &= \left(\frac{1}{\sqrt{2\pi}\,\sigma}\right)^n\cdot e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x(n)-\mu)^2}\end{aligned}$$

- konvoluutiosumma (H1/1 ja tietokoneharjoitukset T1)

  Laskuoperaatio kahdelle lukujonolle f_n ja h_n tuottaen jonon g_n

$$\begin{aligned} g_n &= \sum_{k=-\infty}^{\infty} h_k f_{n-k}\\ &= \ldots + h_{-1}f_{n+1} + h_0 f_n + h_1 f_{n-1} + \ldots\end{aligned}$$

- Diskreettiaikainen Fourier-muunnos (H1/2)

  $F(\omega) = \sum_{m=-\infty}^{\infty} f_m\, e^{-i\omega m}$

  Tässä f_m alkuperäinen lukujono ja $F(\omega)$ (diskreettiaikainen) Fourier-muunnos, jossa normaalisti $-\pi/2 \leq \omega \leq \pi/2$. Olemassa vastaava käänteismuunnos jossa $F(\omega)$:sta f_m.

- Diskreetti Fourier-muunnos (DFT)

  $F_n = \sum_{m=0}^{N-1} f_m\, e^{-i2\pi mn/N}$

  Tässä f_m alkuperäinen lukujono ja F_n diskreetti Fourier-muunnos, jossa $n, m = 0, \ldots, N-1$. Yhteys edelliseen $F_n = F(2\pi n\omega/N)$.

- Taylorin polynomi (H1/3)

  $f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$

  Funktiota voidaan approksimoida siitä kehitetyllä Taylorin polynomilla.

- logaritmit (H1/4, H2/4, H2/5)

  Tyypillisesti luonnollinen (e), 2- tai 10-kantainen logaritmi. Yleisiä laskusääntöjä:

  $\log_m(A\cdot B) = \log_m A + \log_m B$

  $\log_m(A/B) = \log_m A - \log_m B$

  $\log_m A^C = C\log_m A$

  $\log_m D = \log_n D/\log_n m$

  Esimerkki.

  $\log_{10} 2 \approx 0.301$, $2^{10} \approx 10^3$ (k), $2^{20} \approx 10^6$ (M), $2^{30} \approx 10^9$ (G).

  Esimerkki.

$$\begin{aligned} 4^n &= 100000\\ \log_4 4^n &= \log_4 100000\\ n\overbrace{\log_4 4}^{1} &= \log_{10} 100000/\log_{10}\overbrace{4}^{2^2}\\ n &= 5/(2\log_{10} 2) \approx 5/0.602 \approx 8.3\end{aligned}$$

Esimerkki.

$$
\begin{aligned}
10^x &= 2^{2006} \\
\log_{10} 10^x &= \log_{10} 2^{2006} \\
x \log_{10} 10 &= 2006 \log_{10} 2 \\
x &\approx 2006 \cdot 0.301 \approx 630.8 \\
10^{630.8} &= 10^{0.8} \cdot 10^{630} \approx 6.3 \cdot 10^{630}
\end{aligned}
$$

- matriisien kertolasku (H2/1)

  Esimerkki.

  Olkoon A matriisi kooltaan (3×2) (kolme saraketta, kaksi riviä) ja B (2×1). Muistetaan, että kertolaskussa “dimensioiden pitää täsmätä”. Tulossa AB dimensiotarkastelu: $(3 \times \underline{2})(\underline{2} \times 1) \rightarrow (3 \times 1)$.

  Olkoon $A = \begin{bmatrix} 3 & 4 \\ 2 & 5 \\ 6 & 1 \end{bmatrix}$ ja $B = \begin{bmatrix} 8 \\ 9 \end{bmatrix}$

  Niiden matriisitulo $C = AB$, on

  $C = \begin{bmatrix} 3 & 4 \\ 2 & 5 \\ 6 & 1 \end{bmatrix} \cdot \begin{bmatrix} 8 \\ 9 \end{bmatrix} = \begin{bmatrix} 3 \cdot 8 + 4 \cdot 9 \\ 2 \cdot 8 + 5 \cdot 9 \\ 6 \cdot 8 + 1 \cdot 9 \end{bmatrix} = \begin{bmatrix} 60 \\ 61 \\ 57 \end{bmatrix}$.

- determinantti (H2/2)

  Esimerkki.

  Olkoon $C = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, jolloin $\det(C)$:

$$
\det(C) = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc
$$

- matriisin käänteismuunnos, yleisesti

$$
\mathbf{C}^{-1} = \frac{1}{\det(\mathbf{C})} \operatorname{adj}(\mathbf{C})
$$

  jossa adjungoidussa matriisissa osadeterminanttien A_{ij} kerroin $(-1)^{i+j}$

$$
\operatorname{adj}(\mathbf{C}) = \begin{bmatrix} A_{11} & -A_{12} & A_{13} & \dots \\ -A_{21} & A_{22} & \dots & \dots \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}^T
$$

  Esimerkki 2-ulotteiselle $\mathbf{C}$

$$
\begin{aligned}
\mathbf{C} &= \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 4 & 6 \end{bmatrix} \\
\mathbf{C}^{-1} &= \frac{1}{\det(\mathbf{C})} \operatorname{adj}(\mathbf{C}) = \frac{1}{c_{11}c_{22} - c_{12}c_{21}} \begin{bmatrix} c_{22} & -c_{21} \\ -c_{12} & c_{11} \end{bmatrix}^T \\
&= \frac{1}{4} \begin{bmatrix} -6 & 4 \\ 4 & -2 \end{bmatrix}
\end{aligned}
$$

- toisen asteen polynomin ratkaisukaava (H2/1)

  $ax^2 + bx + c = 0 \rightarrow x_i = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}, \quad i = \{1, 2\}$

- ominaisarvot ja -vektorit (H2/1)

  Neliömatriisin C ominaisarvot lasketaan yhtälöstä $Cu = \lambda u \Rightarrow (C - \lambda I)u = 0$. Yhtälöryhmän ratkaisu on yhdenpitävä determinantin kanssa, mistä saadan polynomiaalinen karakteristinen yhtälö $\det(C - \lambda I) = 0 \Rightarrow p_n \lambda^n + p_{n-1}\lambda^{n-1} + \ldots + p_0 = 0 \Rightarrow \lambda_1, \ldots, \lambda_n$.

  Esimerkki.

  Olkoon $C = \begin{bmatrix} 14 & 16 \\ 16 & 20 \end{bmatrix}$, jolloin $\det(C - \lambda I)$:

$$
\begin{aligned}
\det(C) = \begin{vmatrix} 14 - \lambda & 16 \\ 16 & 20 - \lambda \end{vmatrix} &= 0 \\
(14 - \lambda)(20 - \lambda) - 256 = \lambda^2 - 34\lambda + 24 &= 0
\end{aligned}
$$

josta ominaisarvot $\lambda_1 \approx 33.28$ ja $\lambda_2 \approx 0.72$.

Ominaisvektorit saadaan yhtälöstä $\begin{vmatrix} 14 & 16 \\ 16 & 20 \end{vmatrix} u = 33.28u$ ja yhtälöryhmästä $\begin{cases} 14u_1 + 16u_2 = 33.28u_1 \\ 16u_2 + 20u_1 = 33.28u_2 \end{cases}$ josta ominaisvektorit $e_1 = \begin{vmatrix} 0.64 & 0.77 \end{vmatrix}^T$ ja $e_2 = \begin{vmatrix} 0.77 & -0.64 \end{vmatrix}^T$. Huomaa, että nämä ovat pituudeltaan 1.

- pisteen $\mathbf{x}$ projektio suoralle $\mathbf{w}$

  Esimerkki. Olkoon $\mathbf{x} = [3\ 2]^T$ ja se projisoidaan kohtisuoraan suoraa $\mathbf{w} = [0.6\ 0.8]^T$ vasten. Projektiopiste

$$y = \mathbf{w}^T\mathbf{x} = [0.6\ 0.8] \cdot [3\ 2]^T = 0.6 \cdot 3 + 0.8 \cdot 2 = 3.4$$

  Pisteen y sijainti alkuperäisessä koordinaatistossa $\hat{\mathbf{x}} = \mathbf{w}y = [2.04\ 2.72]^T$.

- pisteiden $\mathbf{X} = [\mathbf{x}(1)\ \mathbf{x}(2)\ \ldots\ \mathbf{x}(n)]$ projektio ortogonaaliselle hypertasolle $\mathbf{W}$ (kts. PCA-esimerkit H2 ja tietokonelaskarit T2)

$$\begin{aligned} \mathbf{Y} &= \mathbf{W}^T\mathbf{X} \\ \hat{\mathbf{X}} &= \mathbf{W}\mathbf{Y} \end{aligned}$$

- osittaisderivointi, gradientti (H2/2, H2/3, H2/4, H2/5, H3)

$$\begin{aligned} \frac{\partial}{\partial x}\left(4x^2 + 3xy + 2y^2\right) &= 8x + 3y \\ \frac{\partial}{\partial y}\left(4x^2 + 3xy + 2y^2\right) &= 3x + 4y \end{aligned}$$

- ääriarvokohdat: etsitään (osittais)derivaatan nollakohta

- euklidinen etäisyys (Euclidian distance)

  $d(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_d - y_d)^2}$

- etäisyysmatriisi D näytteiden $\mathbf{x}(1), \ldots, \mathbf{x}(n)$ välillä

$$D = \begin{bmatrix} d(\mathbf{x}(1), \mathbf{x}(1)) & d(\mathbf{x}(1), \mathbf{x}(2)) & \ldots & d(\mathbf{x}(1), \mathbf{x}(n)) \\ d(\mathbf{x}(2), \mathbf{x}(1)) & d(\mathbf{x}(2), \mathbf{x}(2)) & \ldots & d(\mathbf{x}(2), \mathbf{x}(n)) \\ \vdots & \vdots & \ddots & \vdots \\ d(\mathbf{x}(N), \mathbf{x}(1)) & d(\mathbf{x}(N), \mathbf{x}(2)) & \ldots & d(\mathbf{x}(N), \mathbf{x}(n)) \end{bmatrix}$$

  josta nähdään, että matriisi on symmetrinen ja lävistäjällä nollia (jos etäisyysmittana $d(.)$ esimerkiksi euklidinen etäisyys). Vertaa esim. kaupunkien väliset etäisyydet karttakirjassa.

- kuinka monella tavalla voidaan permutoida n muuttujaa? $n!$ (kertoma).

  Esimerkiksi $\{'A','B','C'\} \rightarrow \{'ABC','ACB','BAC','BCA','CAB','CBA'\}$ eli $3! = 1 \cdot 2 \cdot 3 = 6$

- kuinka monta erilaista k:n kokoista joukkoa saadaan n:stä muuttujasta? $\binom{n}{k} = \frac{n!}{(n-k)!k!}$ (binomikerroin).

  Esimerkiksi $\{'A','B','C'\} \rightarrow \{'','A','B','C','AB','AC','BC','ABC'\}$ eli esimerkiksi $\binom{3}{1} = 3$.

  Toinen esimerkki, kuinka monella tapaa voidaan valita 7 numeroa 39:stä

  $\binom{39}{7} = \frac{39!}{32! \cdot 7!} = \frac{39 \cdot 38 \cdot 37 \cdot 36 \cdot 35 \cdot 34 \cdot 33 \cdot 32!}{32! \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2} \approx 15000000$

- normaalijakauman tiheysfunktio

$$p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

  jossa keskihajonta σ ja keskiarvo μ. Varianssi σ^2.

- normaalijakauman d-ulotteinen tiheysfunktio

$$p(\mathbf{x}) = K \cdot \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x} - \mathbf{m})\right)$$

  jossa skaalaustermi

$$K = \frac{1}{(2\pi)^{d/2}\det(\mathbf{C})^{1/2}}$$

ja $\mathbf{C}$ (toisinaan myös merkinnällä Σ) on kovarianssimatriisi sekä $\mathbf{m} = (\mu_1 \; \mu_2 \; \ldots \mu_d)^T$ keskiarvovektori (huippukohta). Kovarianssimatriisi $\mathbf{C}$ on symmetrinen neliömatriisi kokoa $(d \times d)$.

$$\begin{aligned}
\mathbf{C} &= \begin{bmatrix} \mathbf{E}[(x_1-\mu_1)^2] & \mathbf{E}[(x_1-\mu_1)(x_2-\mu_2)] & \ldots & \mathbf{E}[(x_1-\mu_1)(x_d-\mu_d)] \\ \mathbf{E}[(x_2-\mu_2)(x_1-\mu_1)] & \mathbf{E}[(x_2-\mu_2)^2] & \ldots & \mathbf{E}[(x_2-\mu_2)(x_d-\mu_d)] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{E}[(x_d-\mu_d)(x_1-\mu_1)] & \mathbf{E}[(x_d-\mu_d)(x_2-\mu_2)] & \ldots & \mathbf{E}[(x_d-\mu_d)^2] \end{bmatrix} \\
&= \begin{bmatrix} \sigma_{11} & \sigma_{12} & \ldots & \sigma_{1d} \\ \sigma_{21} & \sigma_{22} & \ldots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \ldots & \sigma_{dd} \end{bmatrix}
\end{aligned}$$

jossa siis $\sigma_{ij} = \sigma_{ji}$ ja σ_{ii} ovat yksittäisten muuttujien varianssejа σ_i^2.

Jos kaikki ristitermit $\sigma_{ij} = 0$, niin muuttujat ovat lineaarisesti riippumattomia toisistaan. Jos kaikki ristitermit $\sigma_{ij} = 0$ ja vielä kaikki σ_{ii} vakioita, niin tiheysfunktio on symmetrinen akseleiden suhteen ("hyperpallomainen"). Katso kuvat luentokalvojen luvusta 5.

Käytännössä datasetin $\mathbf{X}$ (dimensio d, n kpl näytteitä, koko $(d \times n)$) kovarianssimatriisi $\mathbf{C}$ (koko $d \times d$) lasketaan

$$\mathbf{C} = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T$$

- eksponentiaalinen jakauma

  $p(x) = \lambda e^{-\lambda x}$

- tasainen jakauma (välillä $[a, b]$)

  $p(x) = 1/(b-a)$, kun $x \in [a, b]$, 0 muuten

- odotusarvo $E\{.\}$; (painotettu) keskiarvo

  $E\{X\} = \int x p(x)\,\mathrm{d}x = \mu$

  $E\{X\} = \sum_i x_i p_i = \mu$

- varianssi Var$\{.\}$, kun $\mu = E\{X\}$ on keskiarvo; otosvarianssi

  $\mathrm{Var}\{X\} = E\{(X-\mu)^2\} = E\{X^2\} - (E\{X\})^2 = \sigma^2$

  $\sigma^2 = \frac{1}{n}\sum_{i=1}^n (x_i - \mu)^2$

  $s^2 = \frac{1}{n-1}\sum_{i=1}^n (x_i - \mu)^2$