Kuidas mõõta mudeli üldistusvõimet?

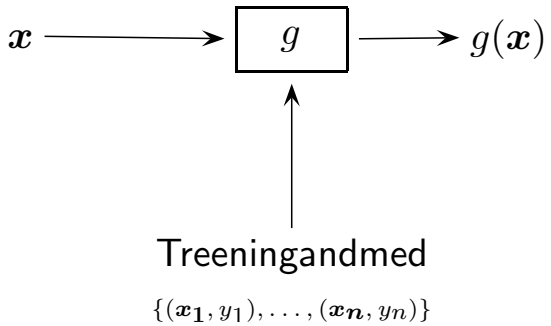
Sven Laur
swen@math.ut.ee

On kolme sorti valesid:
väikesed, suured ja statistika.

Käsitletavad teemad

- Ülesande püstitus:
  - Keskmise oodatava kahju ehk riski hindamine
  - Sobiva mudeli (masinõppe meetodi) valik
- Monte-Carlo meetod integraalide ja summade leidmiseks
- Klassikalised *cross-validation*'i meetodid:
  - *Hold-out* ja *Early-stopping*
  - *Multifold cross validation* ja *Leave-one-out*
- Taasvalikumeetodid süstemaatilise vea hindamiseks:
  - *Bootstrap*, *Bootstrap-632* ja *Bootstrap-632+*
- Teoreetilised meetodid (BIC, AIC ja SLT) — kahjuks ei jõua käsitleda.

Keskmine oodatav kulu ehk risk



$L(g(\boldsymbol{x}), y)$

Pärast treenimist teeb mudel siiski vigu ning keskmine oodatav kulu

$$R(\mathcal{D}_n) = \int\limits_{(\boldsymbol{x},y)\in\mathcal{D}} L(g(\boldsymbol{x}), y)dF(\boldsymbol{x}, y), \qquad \mathcal{D}_n = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_1}, y_1)\}\,.$$

See pole sama, mis SLT uuritav keskmine oodatav risk $\mathbf{E}(R(\mathcal{D}_n))$. Viimane on võetud üle kõigi valimite, meil on vaatluse all konkreetne valim.

Miks on keskmise oodatava kulu hindamine raske?

- Meil puudub täielik informatsioon jaotuse $\mathcal{D}$ kohta. Vastasel korral poleks masinõpet vaja — jaotus võimaldaks anda optimaalse vastuse.
- Reeglina pole jaotuse kohta teada midagi muud kui treeningandmed.
- Treeningandmed võivad sisaldada erandeid (*outliers*) ning seega saame moonutatud infot jaotuse $\mathcal{D}$ kohta.
- Valim võib olla liiga väike ja statistiliselt ebastabiilne.
- Treeningvalim erineb oluliselt pärastisest jaotusest $\mathcal{D}$ (nö. andmete triiv).
- Teooreetilisi tulemusi nagu SLT ei saa reeglina praktikasse otse üle kanda. *Praktikas võib teooria ja praktika oluliselt lahkneda.*

Millist masinõppe meetodit kasutada?

Erivevad mudelid käituvad treeningandmetel erinevalt:

- Mida keerukam on mudel, seda väiksem on treeningviga.
- Reeglina on keerukamad meetodid statistiliselt ebastabiilsemad.
  Väike muutus valimis võib tuua kaasa olulisi erinevusi ennustustes.
- Lihtsad mudelid ei kirjeldada adekvaatselt keerulisi protsesse.
- Tuleb leida mõistlik kompromiss, mille risk oleks väike.

Rusikareegel: Kui võrreldavates mudelites on erinev arv parameetreid, siis vähim treeningviga ei vasta alati parimale mudelile.

Vahel võib suurema statistilise stabiilsuse saavutamiseks keskmistada ennustust üle mitme mudeli. Igale mudelile tuleb siis anda kaal.

Lihtne lahendus

Praktiline lahendus robustne-stabiilne (*bias-variance*) dilemmale:

1. Treenime kõiki mudeleid samadel treeningandmetel.
2. Hindame mingil moel iga mudeli riski.
3. Valime välja mudeli mille risk on vähim.
4. Jätkame parima mudeli treenimist, hindame riski uuesti.

Meetodi peamiseks puuduseks on suur võimalik arvutusmaht ja *ad hoc* meetodid riski hindamiseks.

Tihtipeale kasutatakse seda kombineeritult koos teiste keerukamate meetoditega.

Keerukam lahendus

Teoreetiline lahendus robustne-stabiilne (*bias-variance*) dilemmale:

1. Treenime kõiki mudeleid samadel treeningandmetel.
2. Leiame vastavad treeningvead $\mathrm{E}_{\mathsf{sam}}$ ning lisame neile parandusliikme $\mathrm{C}_{\mathsf{mod}}$, mis annab eelise lihtsamatele mudelitele.
3. Valime välja mudeli, mille korral on $\mathrm{E}_{\mathsf{sam}} + \mathrm{C}_{\mathsf{mod}}$ vähim.

Erinevaid meetodeid parandusliikmete leidmiseks on palju neist tuntumad on BIC, AIC ja MDL.

Varjatud mudeliklassid: On ainult üks mudel, kuid "kaval" parandusliige surub ebavajalikud parameetrid mudelist välja.

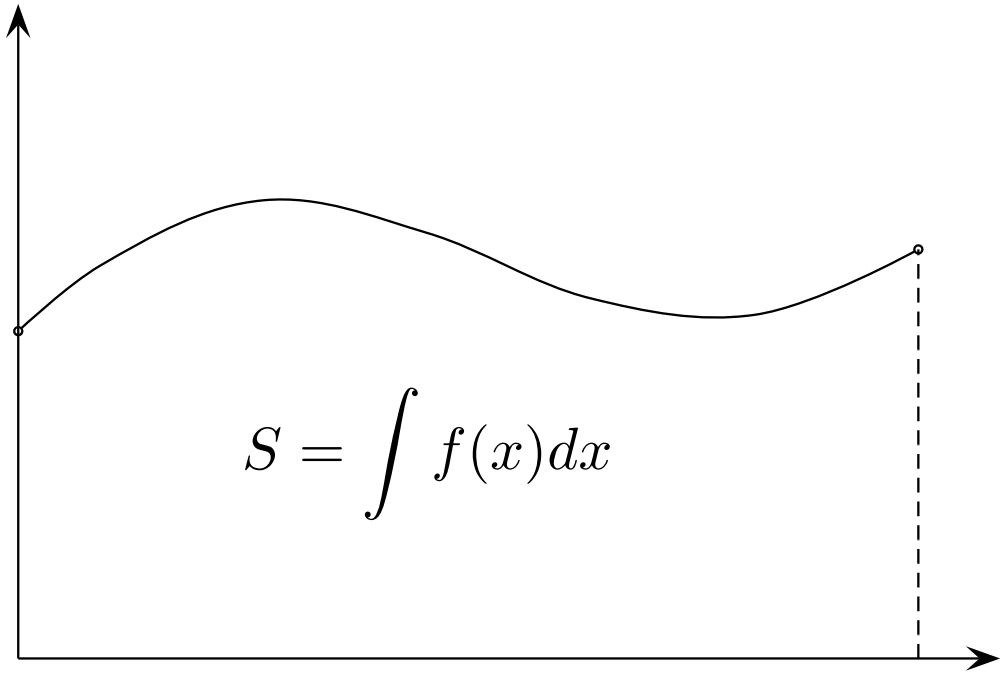
Probleem: Enamasti sisaldavad parandusliikmed vabu parameetreid, mis tuleb kuidagi fikseerida.

Näiteid erinevatest mudelitest

Valik erinevate mudeliklasside vahel võib olla varjatud:

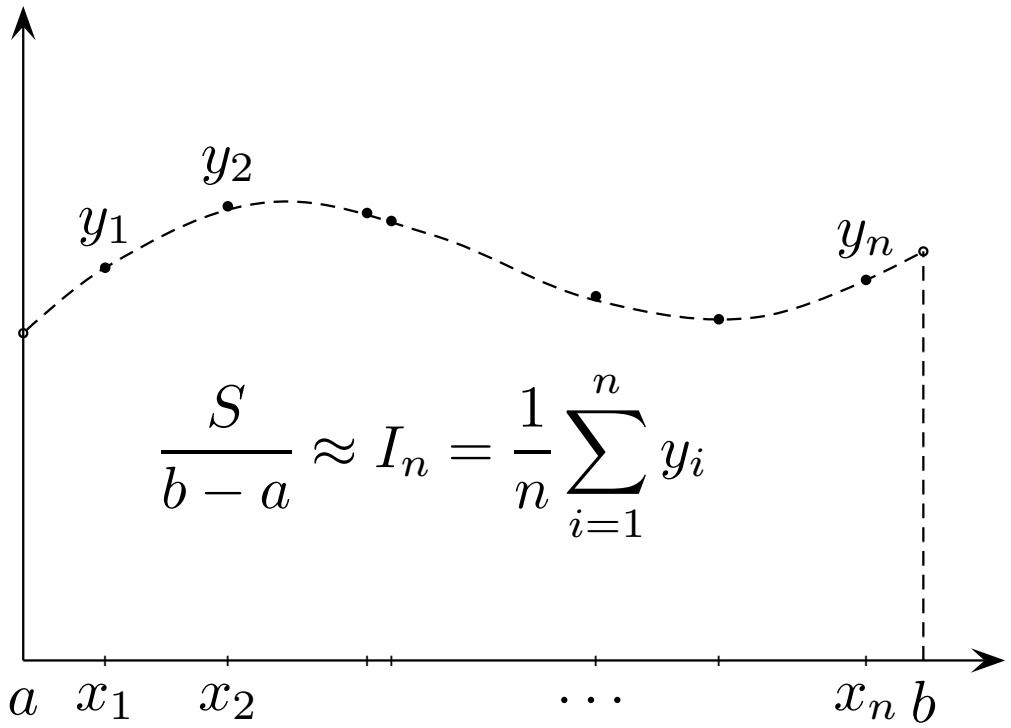
- Mitut sisendit (tunnust) kasutada ennustamiseks?
- Mis on optimaalne neuronite arv?
- Milline on λ optimaalne väärtus *Ridge regressioon*i korral?
- Milliseid sõlmfunktsioone kasutada?
- Millised on optimaalsed parameetrite väärtused algoritmis X?

Monte-Carlo integreerimismeetod



Me tahame arvutada integraali S, aga see on analüütiliselt võimatu.

Monte-Carlo integreerimismeetod



Selle asemel valime n juhuslikku punkti ja leiame keskväärtuse lähendi.

Monte Carlo meetodite ajalugu

Monte Carlo meedodi lätted ulatuvad 18. sajandi teise poolde, kui Comte de Buffon sõnastas esimese stohhastilise meetodi π arvutamiseks.

Kuigi 19. sajandi lõpul ja 20. sajandi esimestel kümnenditel tegeldi stohhastiliste protsesside uurimisega, puudus praktiline vajadus Monte Carlo meetodite järele.

Kiiresti arenev tuumafüüsika tõi endaga kaasa vajaduse lähendada keerukaid mitmemõõtmelisi integraale ning "differentsiaalvõrrandeid".

Stanislav Marcin Ulam, Enirico Fermi, John von Neumann ja Nicolas Metropolis olid esimesed teadlased kes kasutasid süstemaatiliselt stohhastilisi meetodeid arvutuste lihtsustamiseks.

Monte Carlo meetodid olid olulisel kohal Manhattani projektis, mille käigus konstrueeriti esimesed tuumapommid.

Meetodi teoreetilised alused

$$\mathbf{E}(f(X)) \approx I_n = \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x_i}), \qquad \boldsymbol{x_i} \xleftarrow[i.i.d.]{} \mathcal{D}$$

- Meetod toetub kahele tõenäosusteooria põhitulemusele.
- Neid käsitletakse kursuses “Tõenäosusteooria II”.

Tugev suurte arvude seadus

Teoreem (Kolmogorov 1929). *Olgu* $X_1, X_2 \ldots$ *lõpliku dispersiooniga sõltumatud juhuslikud suurused ja* $S_n = X_1 + \cdots + X_n$ *vastav osasumma. Kui*

$$\sum_{i=1}^{\infty} \frac{\mathbf{D}(X_i)}{n^2} < \infty$$

siis jada

$$\frac{S_n - \mathbf{E}(S_n)}{n} \to 0$$

peaaegu kindlasti.

TSAS ja Monte Carlo meetod

Teoreem. *Olgu* $x_1, x_2, \ldots$ *saadud sõltumatult jaotusest* $\mathcal{D}$, *mille dispersioon on lõplik, siis empiiriline keskväärtuse hinnang*

$$\frac{1}{n}\sum_{i=1}^{n} x_i \to \mathbf{E}(x)$$

peaaegu kindlalt.

Kui integreeritav funktsioon on tõkestatud, siis võttes piisava arvu punkte, saame Monte Carlo meetodiga "alati" kuitahes täpse tulemuse.

Koonduvuskiirus ja Tsentraalne piirteoreem

Teoreem (Tsentraalne piirteoreem). *Olgu* $x_1, x_2, \ldots$ *saadud sõltumatult jaotusest* $\mathcal{D}$, *mille dispersioon* σ *on lõplik, siis juhusliku suuruse*

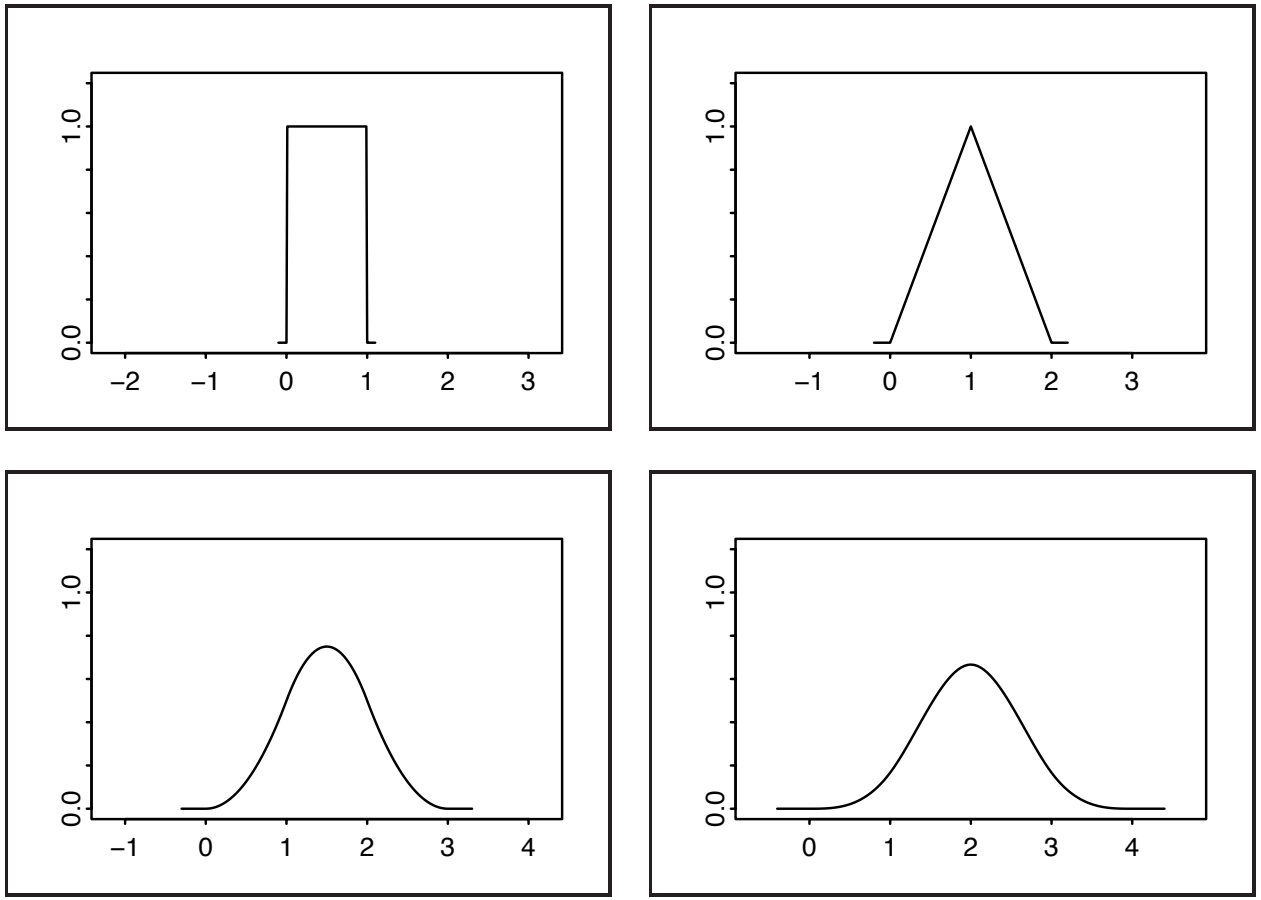
$$\frac{\sum_{i=1}^{n} x_i - n\mathbf{E}(X)}{\sigma\sqrt{n}}$$

jaotus koondub normaaljaotuseks $\mathcal{N}(0,1)$.

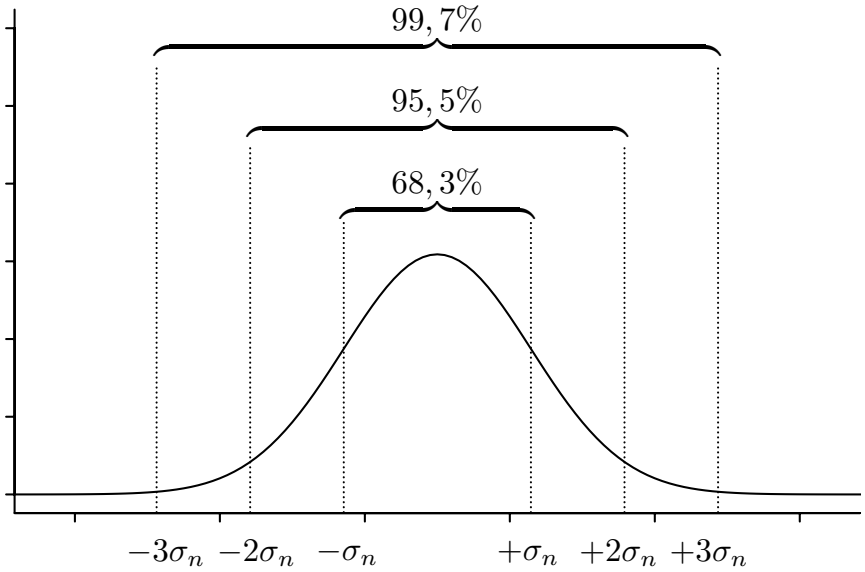
Inimkeeli: Keskväärtuse stohhastiline approximatsioon I_n on ligikaudu normaal jaotusega $\mathcal{N}(\mu, \sigma^2/n)$, kus

$$\mu = \mathbf{E}(f(X)) \qquad \text{ning} \qquad \sigma^2 = \mathbf{D}(f(X)).$$

Näide: juhuslike suuruste $x_i \leftarrow [0, 1]$ summa



Lähendamisviga. Usaldusintervallid



Juhusliku suuruse $f(X)$ dispersioonist σ^2, saame leida

$$\sigma_n = \frac{\sigma}{\sqrt{n}}$$

ning seejärel hinnata lähendi I_n täpsust.

Keskmise oodatava kulu hindamine

Risk on keskväärtus üle tundmatu jaotuse $\mathcal{D}$

$$R(g) = \int\limits_{(\boldsymbol{x},y)\in\mathcal{D}} L(g(\boldsymbol{x}), y) dF(\boldsymbol{x}, y).$$

Üldine algoritm

1. Valime juhusliku testvalimi $(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n) \xleftarrow[i.i.d.]{} \mathcal{D}$.
2. Arvutame lähendi $\mathrm{E}_{\mathsf{gen}} = \frac{1}{n}\sum_{i=1}^{n} L(g(\boldsymbol{x_i}), y_i)$.
3. Arvutame empiirilise dispersiooni $\widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\big(L(g(\boldsymbol{x_i}), y_i) - \mathrm{E}_{\mathsf{gen}}\big)^2$
4. Määrame usaldusintervallid, võttes $\sigma = \widehat{\sigma}$.

Klassikaline *Hold-out* ehk *Split-sample*

Monte Carlo meetod toimib vaid siis, kui funktsioon ei sõltu valimist.

Mudeli risk ja treeningviga ei ole *apriori* korreleeritud.

Lihtsaim viis probleemi lahendada on jagada andmehulk juhuslikult kolmeks.

- Treeningvalimit kasutada mudeli treenimiseks.
- Testvalimit kasutada parima mudeliklassi valikuks.
- Kontrollvalimit kasutada riski hindamiseks.

Märkus: *Holdout*-meetod nõuab suhteliselt suurt valimit, eriti kui kulufunktsiooni $L(\cdot, \cdot)$ variatsioon $\widehat{\sigma}^2$ on suur.

Miks *Hold-out* pole alati sobilik?

Riski lähendusviga on suurusjärgus $\mathcal{O}(1/\sqrt{n})$.

- Meetod nõuab suhteliselt suurt valimit;
- Vaid ühte kolmandikku andmetest kasutatakse mudeli valikuks;
- Üks kolmandik andmetest jääb alati kasutamata.

Meetod ei paljasta lähendi $\mathrm{E}_{\mathsf{gen}}$ suuri fluktuatsioone.

- Halva õnne korral saame täiesti vale tulemuse.
- Hälbed kogu valimis võivad tugevalt varieeruda.
- Treening või testhulka võib sattuda palju erandeid.

Mis on *Early-stoping*?

Reeglina on neurovõrkude treeningmeetodid iteratiivsed.

Early-stopping (varajase lõpetamise) algoritm:

1. Muuta neuronite kaale vastavalt treeningandmetele.
2. Hinnata mudeli riski testandmetel.
3. Kui riski hinnang $\mathrm{E}_{\mathrm{gen}}$ väheneb, korrata sammu 1.

Reeglina peatub algoritm varem, kui tavaliselt. Ning vähemalt näiliselt minimiseeritakse õiget suurust.

Sobilik, kui treeningvalim on suur ja häiritus (müratase) on suur.

Kuna meetod sõltub siiski kaudselt testandmetest, tuleb lõppmudeli riski hinnata siiski kontrollvalimil.

Monte-Carlo Cross-validation

Kui treeningandmeid on vähe või nende hankimine on kulukas on *hold-out* meetod ebasobiv, sest liiga palju andmeid jääb kasutamata.

Mudeli valik kasutades Monte-Carlo *cross-validation*'i:

1. Jaga andmehulk juhuslikult treening- ja testvalimiks.
2. Hinda mudeli riski $\mathrm{E}^i_{\mathsf{gen}}$ testvalimil.
3. Korda samme 1. ja 2. kuni hinnang $\widehat{\mathrm{E}_{\mathsf{gen}}} = \frac{1}{k}\sum_{i=1}^{k} \mathrm{E}^i_{\mathsf{gen}}$ on küllalt täpne.

Idee: Võime võtta oluliselt väiksema testvalimi, kuna keskmistamine anulleerib statistilised fluktuatsioonid.

Probleem I: Saadud jada $\mathrm{E}^1_{\mathsf{gen}}, \ldots, \mathrm{E}^k_{\mathsf{gen}}$ elemendid pole sõltumatud.

Probleem II: Mudelite konstrueerimine on töömahukas.

K-fold cross-validation

Üks levinumaid meetodeid, praktiliselt standard.

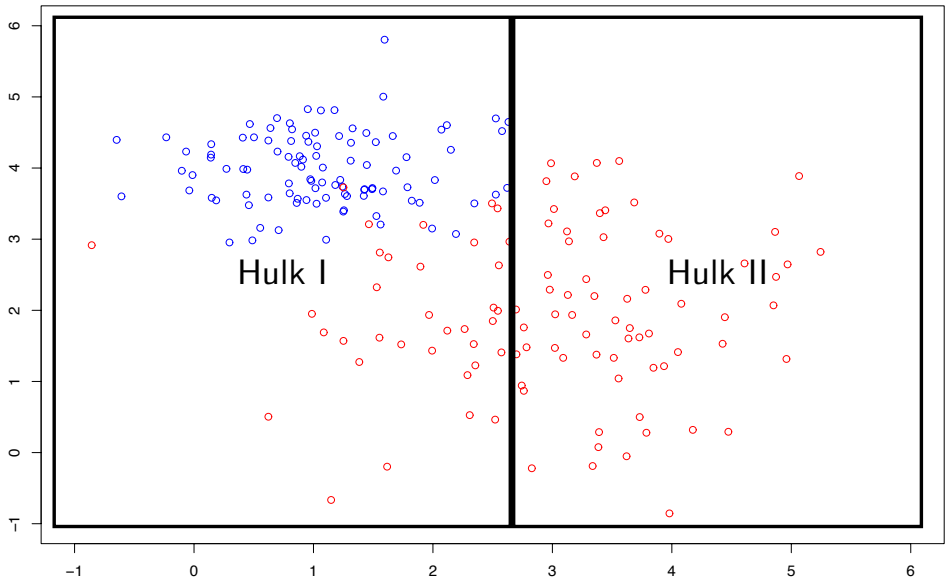
Mudeli valik kasutades *k-fold cross-validation*'i:

1. Jaga andmehulk juhuslikult k ligikaudu võrdseks hulgaks.
2. Treeni $k-1$ osal ja hinda riski $\mathrm{E}^{i}_{\mathsf{gen}}$ välja jäetud hulgal.
3. Vaata läbi kõik k võimalust. Leia keskmine risk ja selle standardhälve.

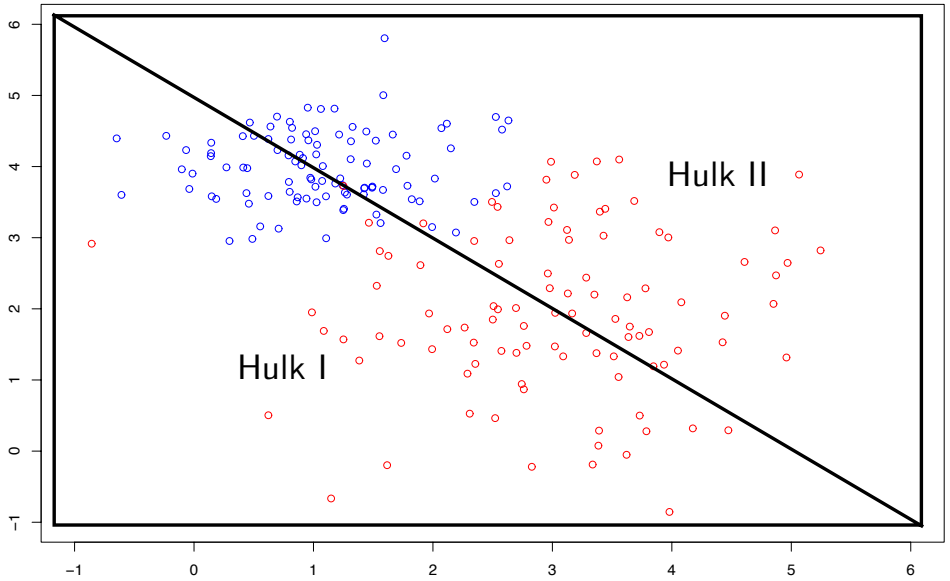
Standard: Enamasti jagatakse andmed 3, 5 või 10 hulgaks. Protseduuri võib korrata täpsuse parandamiseks.

Tasakaalustamine: Reeglina püütakse iga osavalim valida võimalikult lähedane kogu valimile. Eriti oluline on see klassifitseerimisel.

Tasakaalustamata tükeldus



Tasakaalustatud (*stratified*) tükeldus



Leave-one-out

Leave-one-out on *n-fold cross-validation*, kus n on valimi suurus.

Eelised:

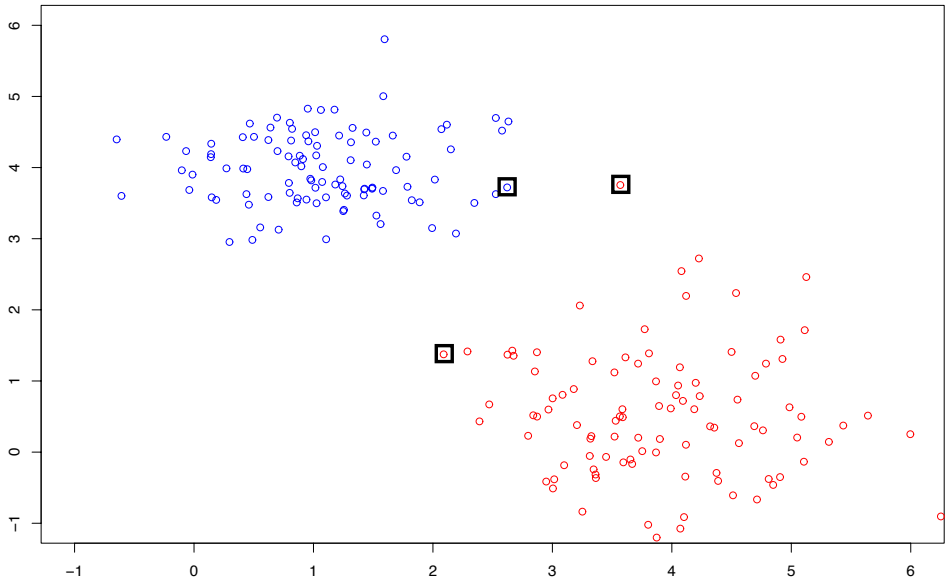
- Meetod on statistilistes ringkondades levinud ja tunnustatud.
- Meetodi eeliseks on lihtne ja determineeritud arvutusreegel.

Puudused:

- Suur arvutusmaht, mida saab teatud erijuhtudel vähendada.
- Meetod ei tööta statistilistelt stabiilsete masinõppe meetodite korral:
  — tulemus ei muutu ühe andmepunkti lisamisel,
  — seega E_{gen} lähend langeb kokku treeningveaga.
- Meetod hinnangu E_{gen} dispersioon on suur.

SVM kui ülistabiilne meetod

Vaid kastides olevate punktide eemaldamisel muutub õpitav mudel.



Seega on *leave-one-out* meetodi kasutamine SVM-ide korral küsitav.

Cross-validation' meetodite peamised omadused

Enamasti annab *cross-validation* piisavalt täpse tulemuse:

- Kesmine risk on on nihketa punkthinnang

$$\widehat{\mathrm{E}_{\mathsf{gen}}} = \frac{1}{k}\sum_{i=1}^{k}\mathrm{E}^{i}_{\mathsf{gen}} \qquad \Rightarrow \qquad \mathbf{E}(\widehat{\mathrm{E}_{\mathsf{gen}}}) = \mathbf{E}_{\mathcal{D},\mathcal{D}'_n}(R).$$

- Samas on $\widehat{\mathrm{E}_{\mathsf{gen}}}$ pessimistlik hinnang, sest osa andmeid jääb treenimisel kasutamata.
- Koondumiskiirust on raske teoreetiliselt hinnata, kuna lähendid $\mathrm{E}^{i}_{\mathsf{gen}}$ pole sõltumatud juhuslikud suurused.
- Lähendite rea $\mathrm{E}^{i}_{\mathsf{gen}}$ dispersioon, kui naiivne standardhälbe hinnang, sisaldab süstemaatilist viga.

Statistiline stabiilsus *vs* mudeli täpsus

Kui palju sõltub risk konkreetsest treeningvalimist?

- Olgu meil juurdepääs jaotusele $\mathcal{D}$.
- Olgu treeningvalim $\mathcal{D}_n \xleftarrow[i.i.d.]{} \mathcal{D}$ ning $R(\mathcal{D}_n)$ vastav risk.
- Kui suur on riski varieeruvus $R(\mathcal{D}_n)$ erinevate $\mathcal{D}_n$ korral?

Rusikareegel: Suure varieeruvusega meetod on ebasobilik — treenimisel saame mudeli, mis suure tõenäosusega on ebaadekvaatne.

Probleem: Meetod võib olla stabiilne (näiteks konstantne regressor), kuid samas täiesti ebasobilik, sest risk on suur.

Lahendus: Tuleb kuidagi hinnata optimismi $\mathrm{Opt} = \mathrm{E}_{\mathsf{gen}} - \mathrm{E}_{\mathsf{sam}}$, mis tekib ainult treeninvea arvestamisel.

Bootstrap? ...See on imelihtne!

Bootstrap on mitteparameetriline statistiline taasvalikumeetod punktihinnangute ning neile vastavate usaldusintervallide leidmiseks.

1. Olgu meil antud valim $\mathcal{D}_n = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}$ ning on tarvis leida $f(\boldsymbol{x_1}, \ldots, \boldsymbol{x_n})$ keskväärtus ja standardhälve üle jaotuse $\mathcal{D}$.
2. Moodustame uue valimi $\mathcal{D}'_n$ võttes n korda juhuslikult ühe elemendi hulgast $\mathcal{D}_n$. Hulga $\mathcal{D}_n$ elemendid võivad korduda!
3. Leiame suuruse $f(\boldsymbol{x'_1}, \ldots, \boldsymbol{x'_n})$ hulgal $\mathcal{D}'_n = \{\boldsymbol{x'_1}, \ldots, \boldsymbol{x'_n}\}$.
4. Kordame protseduuri, kuni saab hinnata adekvaatselt tekkiva jada keskväärtust ja dispersiooni.

Põhjendus: Meetod toimib sest iga element hulgast $\mathcal{D}'_n$ on jaotusega $\mathcal{D}$. Ja elemendid on peaaegu teineteisest sõltumatud.

Optimismi hindamine lihtsa *Bootstrap* meetodiga

1. Korda k korda:
   (a) Moodusta uus treeninghulk $\mathcal{D}'_n = \{(\boldsymbol{x'_1}, y'_1), \ldots, (\boldsymbol{x'_n}, y'_n)\}$, valides juhulikult n-elementi hulgast $\mathcal{D}_n = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$.
   (b) Treeni meetodit g hulgal $\mathcal{D}'_n$ ning leia

$$\mathrm{E}_{\mathsf{sam}} = \frac{1}{n}\sum_{i=1}^{n} L(g(\boldsymbol{x'_i}), y'_i) \qquad \mathrm{E}_{\mathsf{gen}} = \frac{1}{n}\sum_{i=1}^{n} L(g(\boldsymbol{x_i}), y_i).$$

   (c) Leia optimism $\mathrm{Opt}_i = \mathrm{E}_{\mathsf{gen}} - \mathrm{E}_{\mathsf{sam}}$.
2. Leia keskmine optimism $\widehat{\mathrm{Opt}}$ ning usaldusintervallid.
3. Treeni meetodit esialgsel valimil $\mathcal{D}_n$ ning hinda $\mathrm{E}_{\mathsf{gen}} \approx \widehat{\mathrm{Opt}} + \mathrm{E}_{\mathsf{sam}}$.

Lihtsa *Bootstrap*'i omadused

Keskmiselt satub hulka $\mathcal{D}'_n$ umbes 63.2% hulga $\mathcal{D}_n$ elementidest:

- Meetod alahindab süstemaatiliselt riski $\mathrm{E}_{\mathsf{gen}}$, kuna testhulk sisaldab treeninghulka.
- Kui masinõppemeetod ignoreerib elementide kordusi, on lihtne bootstrap samaväärne *Monte Carlo cross-validation*'iga.
  – Sealjuures optimismi alahinnatakse tugevalt!
- Eriti märgatav on süstemaatiline viga väikese ja olematu treeningveaga meetodite korral korral nagu SVM ning NN klassifitseerimine.

Järeldus: Lihtne bootstrap on *sub-optimaalne* meetod, mis on sobilikum regressioonülesannete lahendamiseks.

Crossvalidation Bootstrap

Treeningandmeid ei tohi kasutada testimiseks!

Parandatud esimene samm:

(a) Moodusta treeninghulk $\mathcal{D}'_n$, valides juhulikult n-elementi hulgast $\mathcal{D}_n$.

(b) Treeni meetodit g hulgal $\mathcal{D}'_n$ ning leia $\mathcal{D}_{\text{ots}} = \mathcal{D}_n \setminus \mathcal{D}'_n$ ning

$$\mathrm{E}_{\text{sam}} = \frac{1}{n}\sum_{i=1}^{n} L(g(\boldsymbol{x}'_{\boldsymbol{i}}), y'_i) \qquad \mathrm{E}_{\text{ots}} = \frac{1}{|\mathcal{D}_{\text{ots}}|}\sum_{(\boldsymbol{x}_{\boldsymbol{i}}, y_i)\in\mathcal{D}_{\text{ots}}} L(g(\boldsymbol{x}'_{\boldsymbol{i}}), y'_i).$$

(c) Leia optimism $\mathrm{Opt}_i = \mathrm{E}_{\text{ots}} - \mathrm{E}_{\text{sam}}$.

Märkus: Pideva kaofuntsiooni L korral (nagu keskmine ruutviga) langevad *crossvalidation bootstrap*'i ja *Monte Carlo crossvalidation*'i hinnangud enamasti kokku. Muidu vähendab bootstrappimine variatsiooni.

Bootstrap 632

Crossvalidation bootstrap ülehindab riski, kuna vaid 63.2% andmetest kasutatakse mudeli treenimiseks.

Bootstrap 632 reegel püüab seda tasakaalustada

$$\mathrm{E}_{\mathsf{gen}} \approx 0,368 \cdot \mathrm{E}_{\mathsf{sam}} + 0.632 \cdot \mathrm{E}_{\mathsf{ots}}$$

Optimismi jaoks ümber sõnastatult saame

$$\mathrm{E}_{\mathsf{gen}} \approx \mathrm{E}_{\mathsf{sam}} + 0.632 \cdot \mathrm{Opt}_{\mathsf{ots}}, \qquad \mathrm{Opt}_{\mathsf{ots}} = \mathrm{E}_{\mathsf{ots}} - \mathrm{E}_{\mathsf{sam}}$$

Märkus: On hea hinnang vaid nende mudelite korral, kus treeningviga pole olematu või väga väike. Ei sobi näiteks SVM ja NN klassifitsejate riski hindamiseks.

Suhteline ülesobituskordaja

Hindame meetodi riski korreleerimata $\boldsymbol{x}$ ja y jaoks, mis on jaotunud siiski vastavalt $\mathcal{D}_n$ marginaaljaotusele (*no-information error rate*)

$$\gamma = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} L(g(\boldsymbol{x_i}), y_j)$$

Siis suhteline ülesobitus kordaja avaldub

$$R = \frac{\mathrm{E}_{\mathsf{ots}} - \mathrm{E}_{\mathsf{sam}}}{\gamma - \mathrm{E}_{\mathsf{sam}}} \in [0, 1]$$

Nüüd saame käsitleda ka väikese või olematu treeningveaga mudeleid.

Bootstrap 632+

Bootstrap 632 on liiga optimistlik, kui treeningviga on väike. Relatiivne ülesobituskordaja lubab hinnangut parandada

$$\mathrm{E}_{\mathsf{gen}} \approx (1-\omega)\cdot \mathrm{E}_{\mathsf{sam}} + \omega \cdot \mathrm{E}_{\mathsf{ots}}$$

$$\omega = \frac{0,638}{1-0,368\cdot R}$$

Märkus: Viimane modifikatsioon võimaldab meetodit kasutada peaaegu kõikide meetodite korral.

Eeliseks teiste meetodite ees on väiksem variatsioon mittepidevate kaofunktsioonide (klassifitseerimine) korral.

Süstemaatiline viga veahinnangutes

Kõik veahinnangud kannavad endas *apriori* süstemaatilist viga:

- Veahinnang on aktuaalne, vaid siis kui ta on küllalt väike.
- Aksepteerides "valimatult" väikeseid hinnanguid kallutame tulemusi ning teoreetilised põhjendused ei kehti 100%.
- Meetodid toimivad vaid hästi lahenduvate ülesannete korral.

Näide: Olgu meil klassifikatsiooniülesannete klass $R > 40\%$.

- Siis 95% juhtudest saame riski hinnanguks $\mathrm{E}_{\mathsf{gen}} > 40\%$.
- Vaid 5% juhtudest võime saada veaks $\mathrm{E}_{\mathsf{gen}} < 10\%$, mille korral raporteerime ülesande edukast lahendamisest.
- Niisiis 100% olulistest juhtudest on veahinnang vale!

Viited

Internetis leiduvad materjalid:

- Mark E.P. Plutowski, Survey: Cross-validation in theory and in practice. (1996)
- Matemaatilise Statistika Instituudi materjalid: *Tõenäosusteooria II* ning *Monte Carlo meetodid*
- B. Efron, R. Tibshirani, Cross-Validation and the Bootstrap: Estimating the Error Rate of a Prediction Rule (1995)
- D. H. Wolpert, Off-training Set Error and a Prior Distinctions Between Learning Algorithms (1992)

Käsiraamat: B. Efron, R. Tibshirani, An introduction to the bootstrap (1993).

Vead slaidide esialgses terminoloogias

- Esialgsetes slaidides oli ekslikult termini riski asemel kasutatud empiirilist riski. Reeglina kasutatakse neurovõrkude ja masinõppe kirjanduses rohkem terminit *generalisation error*, mida võiks tõlkida mudeli üldistusvõimeks. SLT kontekstis on rohkem levinud termin risk.
- Erinevus riski tähistustes. Jüri Lember kasutas mudeli g riski tähistamiseks $R(g)$, mina kasutan $R(\mathcal{D}_n) = R(g(\mathcal{D}_n))$, rõhutamaks seda, et mudel g on määratud ära treeningvalimiga $\mathcal{D}_n$.
- Enamik SLT-st tegeleb siiski keskmise riski $\mathbf{E}(R(\mathcal{D}_n))$ hindamisega. Ka teised peamised SLT hinnangud $R(g)$ usaldusintervallidele tulenevad suurte arvude seadustest ning nende saamiseks kasutatakse samuti Monte Carlo veahinnanguid. Peamine erinevus on tehnikas ja eesmärgis: SLT uurib riski ja treeningvea teooreetilist vahekorda.

Küsimus 1: Usaldusintervalli interpretatsioon

Olgu meil vaja hinnata integraali $\mu = \int g(x)dF(x)$ väärtust. Olgu teada $g(x)$ standardhälve σ. Enne punktide x_i võtmist jaotusest $\mathcal{D}$ teame lähendi I_n kohta

$$\Pr\left[|I_n - \mu| \geq \frac{\sigma}{\sqrt{n}}\right] \leq 31.7\%$$

Mis on vea tõenäosus peale I_n arvutamist?

– Õige vastus kas 0 või 1, sest väärtus $|I_n - \mu|$ on selleks hetkeks fikseeritud.

Mis on vea tõenäosus kui on teada $\mu < 0.5$ ja $I_n < 0.5$?

Mis on vea tõenäosus kui on teada $\mu < 0.5$ ja $I_n > 0.5$?

Kas informatsiooni $\mu < 0.5$ saaks kuidagi kasutada μ lähendamisel?

Küsimus 2: Standardhälbe valem

Standardhälbe hindamiseks kasutatakse kahte valemit

$$\widehat{\sigma_1}^2 = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - \widehat{\mu})^2, \qquad \widehat{\mu} = \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

$$\widehat{\sigma_2}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(f(x_i) - \widehat{\mu})^2, \qquad \widehat{\mu} = \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

Neist esimene on suurima tõepäraga punkthinnang, kuid teine nihketa punkthinnang (on täpsem väiksemate valimite korral). Eriti huvitav on võrrelda omavahel valemite standardset tuletust ja Bayesi statistikal põhinevat tuletust.

Crossvalidationi korral peaks kasutama $\widehat{\sigma_2}$, sest k on väike.

Küsimus 3: Kas *leave-one-out* võib teha suuri vigu?

Vaatame kahte klassifitserijat

$$g_{minvote}(x) = \begin{cases} 1, \text{kui klass 1 on vähemuses,} \\ 0, \text{kui klass 0 on vähemuses,} \end{cases}$$

$$g_{maxvote}(x) = \begin{cases} 1, \text{kui klass 1 on enamuses,} \\ 0, \text{kui klass 0 on enamuses,} \end{cases}$$

jaotusel $\mathcal{D}$, kus klassid 0 ja 1 on võrdtõenäosed. Olgu valimis täpselt pooled elemendid klassist 0 ja pooled klassist 1. Veenduda

$$\mathrm{E}^{\mathsf{loo}}_{\mathsf{gen}}(g_{minvote}) = 0 \qquad \text{ja} \qquad \mathrm{E}^{\mathsf{loo}}_{\mathsf{gen}}(g_{maxvote}) = 1$$

Kuidas saadud tulemust interpreteerida?

– Näide illustreerib, et hinnangud $\mathrm{E}^{i}_{\mathsf{gen}}$ pole sõltumatud.

Küsimus 4: Miks *leave-one-out* on halb?

Kuna erinevad vead $\mathrm{E}^i_{\mathsf{gen}}$ pole sõltumatud, siis võib patoloogilistel juhtudel *leave-one-out* anda totaalselt valesid tulemusi. Sealjuures ei anna jada $\mathrm{E}^i_{\mathsf{gen}}$ ühtki ohusignaali.

Kas leidub näiteid, kus *leave-one-out* meetod teeb suuri vigu märgataval osal võimalikest treeninghulkadest?

Miks on *crossvalidation* parem? Too kaks põhjust.

Vaadata kirjandusest teoreetilisi põhjendusi (Efroni artiklid).

Küsimus 5: *Leave-one-out* ja ülistabiisus

Olgu meil statistiline meetod, mille korral mudel ei muutu kui asendada üks andmepunkt. Tegelikult on see nõue liiga range, kuid lihtsustab järgnevat analüüsi.

Veenduda, et *leave-one-out* annab tulemuseks treeningvea.

Loomulikult on ülistabiilse meetodi korral treeningviga ja risk enamasti lähedased. Veahinnangu mõte on avastada olukordi, kus treeningviga on liiga optimistlik hinnang riskile.

Kuna $\mathrm{E}_{\mathsf{gen}} = \mathrm{E}_{\mathsf{sam}}$, siis *leave-one-out* ei toimi indikaatorina ning meil ei õnnestu avastada veidraid juhtumeid, kus $\mathrm{E}_{\mathsf{sam}} \ll R(\mathcal{D}_n)$.

Probleem pole mitte meedodi ülistabiilsuses, vaid *leave-one-out* sobimatuses indikaatormeetodina.

Küsimus 6: Miks *crossvalidation* on pessimistlikum kui *leave-one-out*

Põhjus on ilmne: *leave-one-out* kasutab treenimiseks kogu valimit, samas kui *crossvalidation* vaid $2/3$ või $9/10$ valimist. Kuna suurema valimi korral varieeruvad mudeli parameetrid vähem, siis enamasti on suuremal andmemahul treenitud meetod täpsem. Seega on riski hinnang CV korral enamasti liiga pessimistlik.

Küsimus 7: Miks võivad veahinnangud olla kehtetud?

Olgu meil firma Datamining Inc., kus RD osakonnas otsitakse sobivaid mudeleid ning osakonnas QA hinnatakse mudelite täpsust.

- QA osakonnas töötav teadlane A kasutab SLT-d riski hindamiseks.
- QA osakonnas töötav teadlane B kasutab *crossvalidationi* ja *bootstrap*'i.

Miks on mõlema teadlase hinnangud ilmselt valed, kui vaid üks tuhandik mudelitest läheb QA osakonda, kuna ülejäänute treeningviga on liiga suur?

Miks on mõlema teadlase hinnangud 90% juhtudest õiged, kui vaid 5% mudelitest ei jõua QA osakonda ning usaldusintervallid on võetud kindluse 95% jaoks?

Kas saab teha vahet kumma juhtumiga on tegu, kui on lahendamisel vaid üks ülesanne?