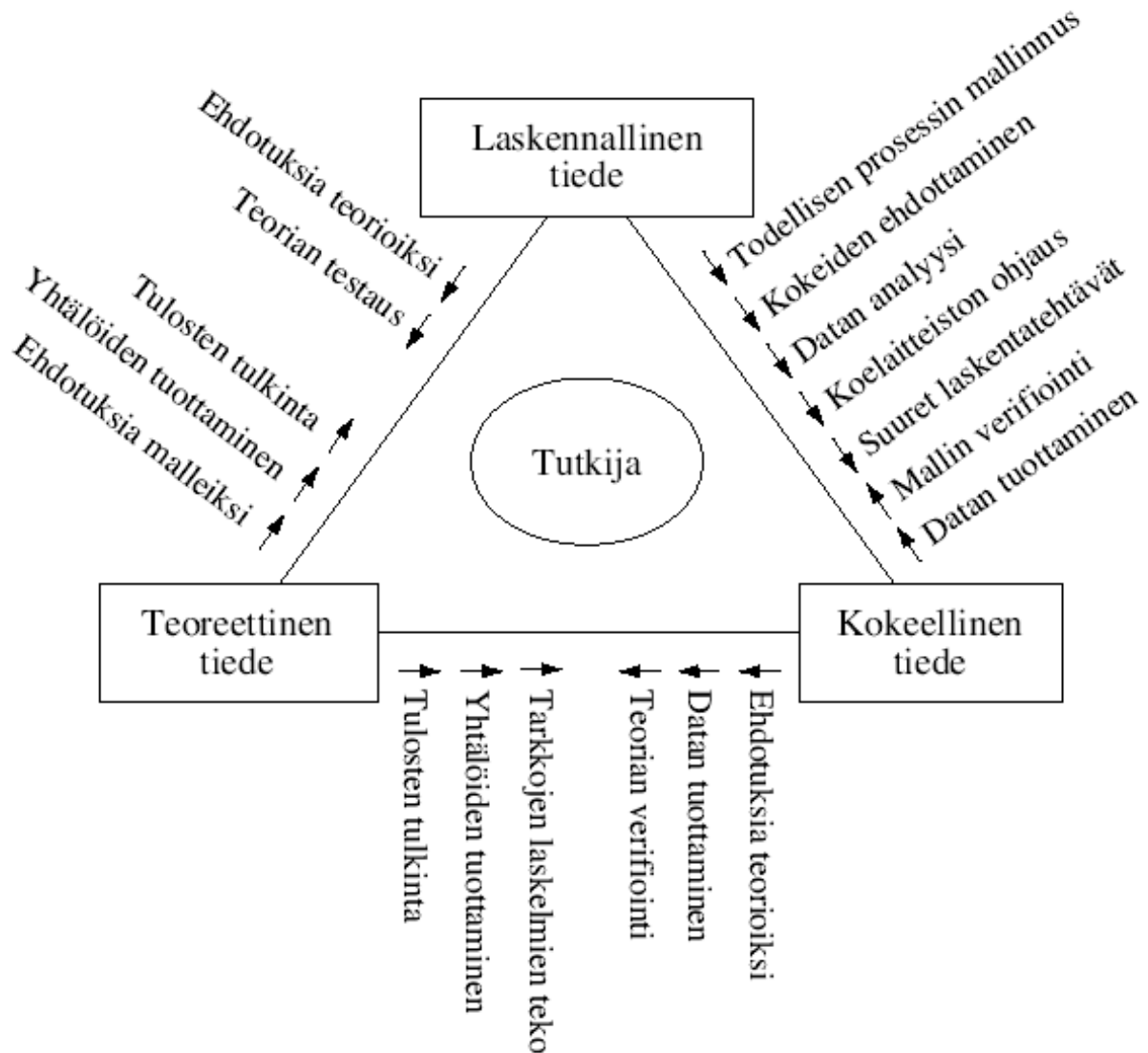


LUENTO 1 19.9.2005

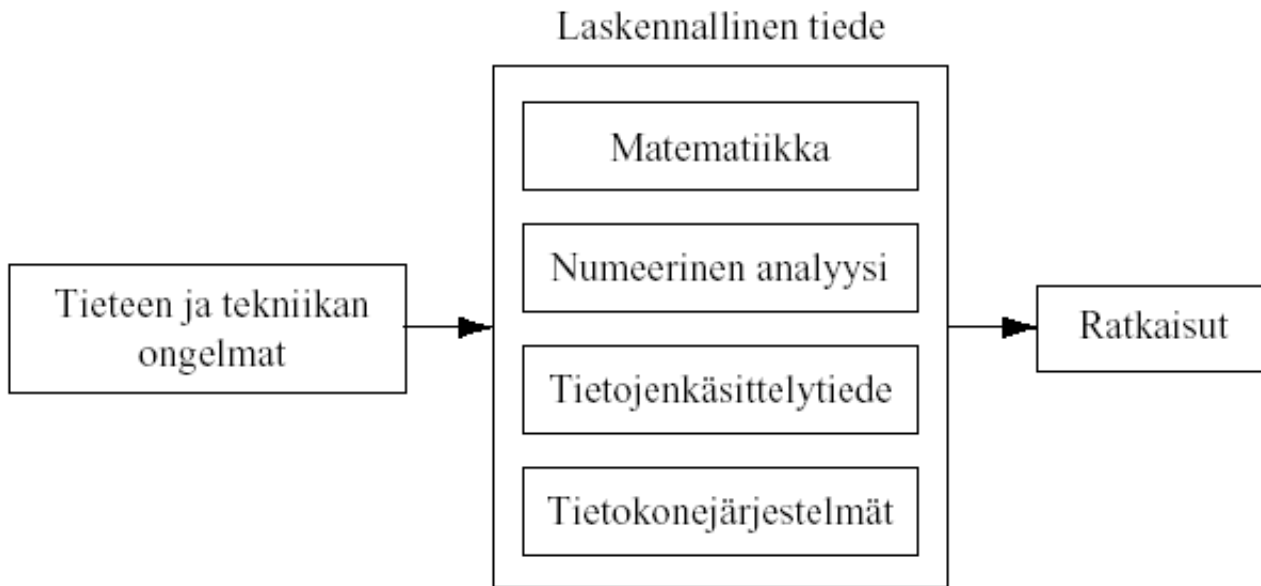
(alla näkyvät kaaviot ja esimerkit oppasta [Haa02])

Mihin laskennallista tiedettä tarvitaan?



Laskennallisen tieteen osa-alueita

(huom. ”matematiikka” sisältää esim. fysikaalisen / kemiallisen / biologisen / teknisen ongelman matemaattisen mallin)



Esimerkki: Peto-saalis –malli $\rightarrow$ yksinkertainen tavallisen differentiaaliyhtälön alkuarvotehtävä

(tavallinen diff.yhtälö $\leftrightarrow$ vain yksi riippumaton muuttuja, esim. aika)

■ **Esimerkki 7.1.1** Olkoon $j(t)$ jänisten ja $k(t)$ kettujen lukumäärä hetkellä t . Voimme kuvata jänis- ja kettupopulaatioita yksinkertaisella peto-saalis-mallilla

$$j'(t) = (\beta_j - \mu_j)j(t) - \alpha_j j(t)k(t),$$

$$k'(t) = (\beta_k - \mu_k)k(t) + \alpha_k j(t)k(t),$$

$$j(t_0) = j_0,$$

$$k(t_0) = k_0,$$

missä j_0 ja k_0 ovat jänisten ja kettujen määrät alkuhetkellä $t = t_0$. Parametrit β_j ja β_k kuvaavat syntyvyyksiä ja parametrit μ_j ja μ_k kuolleisuuksia. Parametrit α_j ja α_k mallintavat saalistuksen aiheuttamaa jänisten ja kettujen vuorovaikutusta.

Virhelähteistä

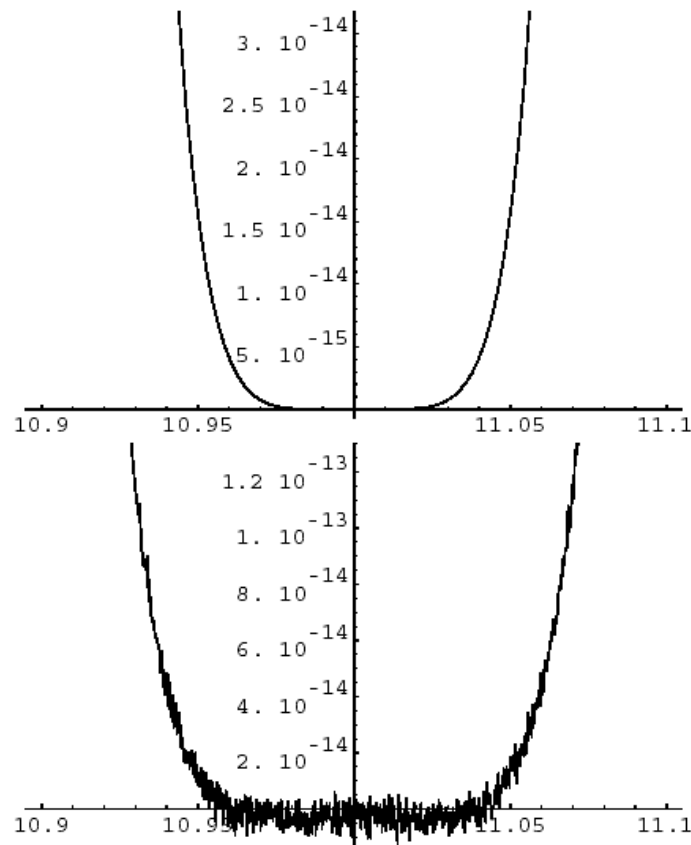
- reaalilukujen esittäminen tietokoneessa $\leftarrow \rightarrow$ katkaisu- tai pyöristysvirheet, ks [Haa02] ss. 393 – 399
- katkaisuvirhe
- pyöristysvirhe
- IEEE – aritmetiikka
- suurin liukuluku = ylivuototaso, overflow level, OFL, $\approx 10^{38}$ (32 bits)
- pienin positiivinen liukuluku = alivuototaso, underflow level, UFL, $\approx 10^{-38}$ (32 bits)
- konevakio ϵ_{kone} : pienin liukuluku, joka toteuttaa epäyhtälön
$$1.0 + \epsilon_{\text{kone}} > 1.0 \quad (\text{suhteellinen tarkkuus})$$
- IEEE – 32 bits: $\epsilon_{\text{kone}} = 2^{-23} = 10^{-7}$
- IEEE – 64 bits: $\epsilon_{\text{kone}} = 2^{-52} = 2 \times 10^{-16}$
- Symbolit Inf = ääretön, esim. 1.0/0.0 ja NaN = Not a Number, esim. 0.0/0.0

Kaksi esimerkkiä liukulukuaritmetiikan ongelmista:

■ **Esimerkki 2.3.4** Korkea-asteiset polynomit ovat virhealttiita. Piirrämme polynomin

$$f(x) = (x/10 - 1.1)^6$$

kuvaajan (kuva 2.6) käyttäen yllä olevaa esitysmuotoa ja tämän jälkeen auki kirjoitettua muotoa. Johtuen liukulukuaritmetiikan epätarkkuudesta tuottaa auki kirjoitettu esitys kohinaisen kuvaajan, josta ei voi määrittää funktion nollakoh-
taa tai minimiä.

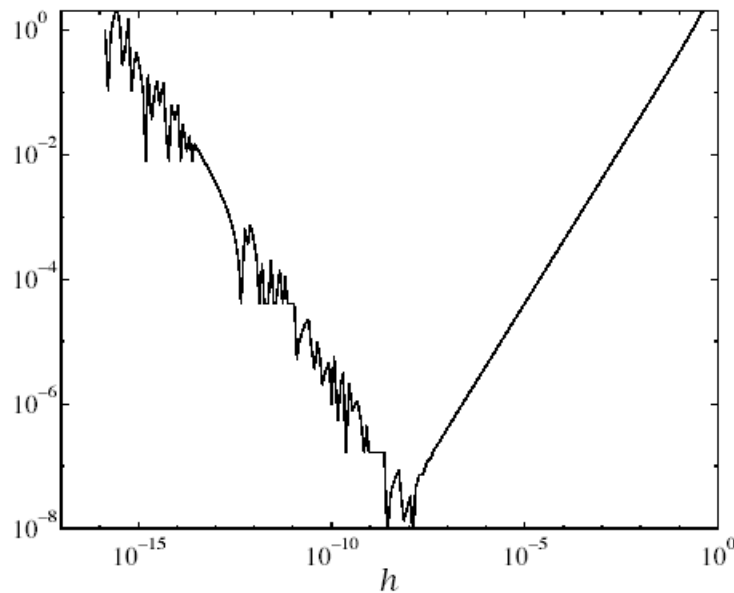


Kuva 2.6: Funktion $f(x) = (x/10 - 1.1)^6$ kuvaajan piirtäminen kahdella eri esitystavalla. Erot johtuvat korkea-asteisen polynomin häiriöalttiudesta.

■ **Esimerkki 2.3.2** Voimme arvioida funktion $f(x)$ derivaattaa differenssikaavalla

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}.$$

Jos asetamme askeleen h pieneksi, voimme pienentää katkaisuvirhettä. Toisaalta tällöin pyöristysvirheet voivat kasvaa ja pudottaa arvion tarkkuutta: kun askel h on pieni, arvot $f(x+h)$ ja $f(x)$ ovat samaa suuruusluokkaa ja siten erotuksessa on vähän merkitseviä numeroita. Kuva 2.5 esittää differenssiarvion virhettä eri askelpituuksilla funktiolle $f(x) = xe^x$ pisteessä $x = 1$. Laskennassa on käytetty 64-bittistä IEEE-aritmetiikkaa, jolloin paras approksimaatio saadaan askelpituudella $h \approx \sqrt{\epsilon_{\text{kone}}} \approx 10^{-8}$.



Kuva 2.5: Funktion derivaatan arviointi differenssillä. Kuva esittää differenssiarvion virheen askelpituuden funktiona.