Harjoitus 5: Symbolinen laskenta I (Mathematica)

MS-C2107 Sovelletun matematiikan tietokonetyöt 2020

Aalto University
School of Science

Systems
Analysis Laboratory

Harjoituksen aiheita

- Tutustuminen Mathematica-ohjelmistoon
- Mathematican sisäänrakennettujen funktioiden käyttö
- Yhtälöiden ratkaiseminen Mathematicalla
- Derivointi Mathematicalla
- Mathematican graafiikkatulostukset

Osaamistavoitteet

- Osaat Mathematican käytön perusteet
- Osaat derivoida lausekkeita ja ratkaista yhtälöitä Mathematicalla

Mathematica

- Laskentaohjelmat on perinteisesti jaettu numeerisiin (esim. Matlab) ja symbolisiin (esim. Mathematica) ohjelmiin.
  - Raja on kuitenkin hämärtynyt: monilla ohjelmilla voi nykyään laskea sekä symbolisesti että numeerisesti.
- Mathematica on ohjelmisto, joka pystyy sekä symboliseen että numeeriseen laskentaan.

Mathematican rakenne

- Mathematica koostuu kahdesta osasta:
  - Kernel: Laskennan suorittava osa
  - Front end: Ohjelman käyttöliittymä (vaihtelee käyttöympäristön mukaan)
- Yleisin käyttöliittymä on muistikirjapohjainen:
  - Istunnosta muodostuu tallennettava dokumentti (muistikirja).
  - Muistikirjoja voi olla avoinna useampia samanaikaisesti.
  - Muistikirjassa voidaan muokata ja ajaa syötteitä ja siihen tulostuu myös ohjelman tulokset.
  - Myös tekstin lisääminen on mahdollista: ohjelmalla voidaan tuottaa myös dokumentteja (esim. html-muotoon).
- Myös tekstipohjainen käyttöliittymä on olemassa.

Komentojen syöttäminen 1/2

- Mathematicassa syötteet kirjoitetaan joko
  - tekstipohjaisesti suoraan muistikirjaan
  - palettien avulla (graafiset ikonit): File → Palettes
- Muistikirja koostuu soluista. (Merkitty muistikirjan oikeassa laidassa olevilla hakasilla).
- Solun sisällä Enter vaihtaa rivin, jolloin voidaan syöttää useampi komento peräkkäin. Komentojen kirjoittaminen samaan soluun on hyvää tyyliä ja vähentää virheitä Mathematicassa!
- Shift+Enter ajaa kaikki solussa olevat komennot. Tämä vastaisi Matlabissa m-tiedoston ajoa.
- Kaikki ohjelman tulokset (myös grafiikka) tulostetaan syötesolun jälkeiseen soluun.
- Tulostuksen voi estää lisäämällä puolipiste (;) syötteen perään.

Komentojen syöttäminen 2/2

- Standardifunktioiden ja -vakioiden nimet alkavat isolla kirjaimella. Esim. `Solve`, `Sin`, `Pi`.
  - Sekaannuksien välttämiseksi tulee omat muuttujat nimetä eri nimisiksi kuin sisäänrakennetut funktiot ja aloittaa ne pienellä kirjaimella.
- Funktioiden argumentit tulevat hakasuluissa `[ ]`. Esim. `Sqrt[x]`
  - Jos funktiolla on vain yksi argumentti, voidaan funktio kirjoittaa myös `//` -merkkien jälkeen. Esim. `x // Sqrt`.
- Kaarisuluilla `( )` osoitetaan ainoastaan laskujärjestys.
  - Esim. `sin(x)` on muuttuja `sin` kertaa muuttuja `x`.

Opastustoiminnot

- Tietoa mistä tahansa Mathematican komennosta saat kysymysmerkillä. Esim. `?Solve`

```
In[1]:= ?Solve
Solve[expr,vars] attempts to solve the system expr of equations
or inequalities for the variables vars...
```

- Kahdella kysymysmerkillä saat tietoa myös komennon optioista
  - Esim. `??Plot`, kokeile myös `Options[Plot]`
- Kysymysmerkkiä voidaan käyttää myös muiden kuin funktioiden yhteydessä.
  - Esim. `?=` antaa tietoa sijoitusoperaattorin `=` käytöstä.
- Mathematicassa on myös kattava Help Browser (Help-valikko → Help), josta löytyy myös paljon esimerkkejä komentojen käytöstä.

Sijoitusoperaattori =

- = asettaa muuttujalle arvon.

```
In[1]:= la = x+y
Out[1]= x + y
In[2]:= la
Out[2]= x + y
In[3]:= x = 7
Out[3]= 7
In[4]:= la
Out[4]= 7 + y
In[5]:= y = k
Out[5]= k
```

```
In[6]:= la
Out[6]= 7 + k
In[7]:= x
Out[7]= 7
In[8]:= y
Out[8]= k
In[9]:= y = 3
Out[9]= 3
In[10]:= la
Out[10]= 10
```

Muuttujien poistaminen

- Muuttujan voi poistaa muistista komennolla `Clear[muuttuja]`.
  - Useamman muuttujan poistaminen: `Clear[muuttuja1,muuttuja2]`.
  - Kaikkien muuttujien poistaminen: `ClearAll["Global‘*"]`
- Monet virhetilanteet johtuvat siitä, että symbolille on jäänyt voimaan jokin määrittely, esim. ”`x=7`”, minkä seurauksena esimerkiksi seuraava yhtälön ratkaisuyritys x:n suhteen ei toimi:

```
In[37]:= Solve[x+1==0,x]
General::ivar: 7 is not a valid variable.
Out[37]= Solve[False, 7]
```

Listat

- Mathematicassa vektorit ja matriisit esitetään listoina.
- Listat muodostetaan aaltosulkuja `{ }` käyttämällä.
  - Lista on muotoa `{alkio1, alkio2, ...}`
- Listan alkio voi olla melkein mitä tahansa, vaikkapa yksittäinen luku tai lista.
  - Esim. listassa `{{7,5,9},4}` on kaksi alkiota, `{7,5,9}` ja `4`. Ensimmäinen alkio on lista, jossa on kolme alkiota: `7`, `5` ja `9`. Toinen alkio on tavallinen luku.
- Listan alkioon i viitataan notaatiolla `{...}[[i]]`.

Listaan viittaukset 1/3

```
In[92]:= a={{7,5,9},4}
Out[92]= {{7, 5, 9}, 4}
In[93]:= b=a[[1]]
Out[93]= {7, 5, 9}
In[94]:= b[[2]]
Out[94]= 5
```

- Listan alkiossa i olevan listan alkioon j voidaan viitata notaatiolla `{...}[[i,j]]`.

```
In[95]:= a={{7,5,9},4}
Out[95]= {{7, 5, 9}, 4}
In[96]:= a[[1,2]]
Out[96]= 5
```

Listaan viittaukset 2/3

- Samalla notaatiolla voidaan viitata mm. symbolien `+`, `-`, jne., sekä `->` erottamiin alkioihin:

```
Out[118]= yhtalo=-6 + x + y
In[119]:= yhtalo[[2]]
Out[119]= x
In[120]:= ratkaisu=Solve[x+y==3*z,x]
Out[120]= {{x -> -y + 3 z}}
In[121]:= ratkaisu[[1]]
Out[121]= {x -> -y + 3 z}
In[122]:= ratkaisu[[1,1]]
Out[122]= x -> -y + 3 z
In[123]:= ratkaisu[[1,1,2]]
Out[123]= -y + 3 z
```

Listaan viittaukset 3/3

- Edellinen esimerkki yksinkertaisemmin:

```
In[124]:= Solve[x+y==3*z,x][[1,1,2]]
Out[124]= -y + 3 z
```

- Huom! Symbolit `->` ja `+` eivät ole samanarvoisia erottimia, Out[124] olisi muuten ”`-y`”.
- ”`-y`”:n saa vasta komennolla `Solve[x+y==3*z,x][[1,1,2,1]]`

Korvausoperaattori /.

- Muuttujan arvo korvataan tilapäisesti muutossäännöllä:
  - `muuttuja-> arvo`
- Listanotaatio korvattaessa useampia muuttujia:
  - `{muuttuja1->arvo1,muuttuja2->arvo2}`

```
In[11]:= a+b
Out[11]= a + b
In[12]:= a+b /. {a->5,b->z}
Out[12]= 5 + z
In[13]:= a
Out[13]= a
In[14]:= b
Out[14]= b
In[15]:= lb = a+b
Out[15]= a + b
```

```
In[16]:= lb /. {a->6,b->k}
Out[16]= 6 + k
In[17]:= lb
Out[17]= a + b
In[18]:= lb /. {a->2,b->3}
Out[18]= 5
In[19]:= lb
Out[19]= a + b
```

Funktioiden määritteleminen

- Funktioita määritellään ja käytetään seuraavan esimerkin mukaisesti

```
In[142]:= f[x_,y_]=x^2*y+a
Out[142]= x^2*y+a
In[143]:= f[2,a]
Out[143]= 5a
```

- Funktioita käyttämällä voidaan välttää mutkikkaiden viittausten ja korvausoperaatioiden tarvetta

Derivointi ja yhtälöryhmien ratkominen

- Komennolla `D` saat (osittais)derivaatan halutun muuttujan suhteen. Voit luoda derivoitavan funktion erikseen tai kirjoittaa derivoitavan lausekkeen derivaattakomennon sisään.
- Huom! Tässä esimerkissä syötämme kaksi komentoa soluun. Ekan perässä on puolipiste, joten se ei tulostu.

```
In[1]:= f[x_]=3 x^2 + 2;
        D[f[x],x]
Out[2]= 6x
```

- Samalla komennolla saat joko derivaatan tai gradientin:

```
In[3]:=g[x_, y_]= x^2 + 2 y^2;
       D[g[x, y], y]
       D[g[x, y], {{x, y}}]
Out[4]=4y
Out[5]={2x, 4y}
```

Vertaailuoperaattori (==) 1/2

```
In[6]:= a=3
Out[6]= 3
In[7]:= a==3
        a==4
Out[7]= True
Out[8]= False
```

- Yhtälöitä ja yhtälöryhmiä voidaan ratkaista `Solve`-komennolla. Numeerinen ratkaisu onnistuu komennolla `NSolve`.
- Esim.

```
In[58]:= Solve[{x+1==c,c==2x-2},{x,c}]
Out[58]= {{x -> 3, c -> 4}}
```

Vertailuoperaattori (==) 2/2

- Huomaa mikä sotku voi seurata yhdestä pienestä kirjoitusvirheestä ("c=2", kun pitäisi olla "c==2"):

```
In[48]:= Solve[{x+1==c,c=2},{x,c}]
General::ivar: 2 is not a valid variable.
Out[48]= Solve[{1 + x == c, 2}, {x, 2}]
In[49]:= Solve[{x+1==c,c==2},{x,c}]
General::ivar: 2 is not a valid variable.
Out[49]= Solve[{1 + x == 2, True}, {x, 2}]
In[50]:= c
Out[50]= 2
In[51]:= Clear[c]
In[52]:= Solve[{x+1==c,c==2},{x,c}]
Out[52]= {{x -> 1, c -> 2}}
```

Edellisiin tulostuksiin viittaaminen (%)

- Edelliseen tulostukseen viittaus: `%`, sitä edelliseen: `%%`, jne., tulostukseen n (Out[n]): `%n`.

```
In[85] = 13 + x - x (x - y) + 7 y
       = % /. {x->3,y->1}
Out[85]= 13 + x - x (x - y) + 7 y
Out[86]= 17
In[87]:= % + 3
Out[87]= 20
In[88]:= %85 /. {x->1,y->7}
Out[88]= 69
```

- Viittaukset ovat käteviä lähinnä solujen sisällä. Monesti on kuitenkin selvempää tallentaa tulokset muuttujiin.

Lausekkeiden muokkaaminen

- Mathematicassa on paljon eri komentoja lausekkeiden muokkaamiseen.
  - Käyttötarkoituksesta riippuu, mikä muoto on kulloinkin paras.
- Joitakin komentoja:
  - `Simplify` ja `FullSimplify`: Lausekkeen sievennys
  - `Expand` ja `ExpandAll`: Kerrotaan auki tulot ja positiiviset kokonaislukupotenssit.
  - `Factor`: Tekijöihin jako
  - `Apart`: Osamurtokehitelmä
  - `Normal`: Muuttaa lausekkeen ”tavalliseen”muotoon useista eri muodoista. Esim. virhearvion poistaminen.

Tarkat arvot ja likiarvot

- Tarkkoja esityksiä: Kokonaisluvut ja murtoluvut, sekä symbolisessa muodossa esitetyt vakiot, esim. π, $\sqrt{2}$.
- Likiarvoja: desimaaliluvut
- Mathematica esittää tulokset symbolisessa muodossa (tarkkoina arvoina), jos lausekkeen kaikki arvot ovat tarkkoja.
  - Esim. komennot `Pi/4` ja `Pi*0.25` antavat eri tuloksen!
  - Tästä syystä kannattaa aina käyttää tarkkoja arvoja ja laskea likiarvot vasta lopuksi, jos se on tarpeellista.
- Likiarvoja saadaan komennolla `N`. Esim. `N[Pi]`.

Hyödyllisiä esimerkkejä 1/2

- %-viittaus voi säästää kirjoitusvaivaa. Halutuaan ratkaista yhtälöpari ja sijoittaa ratkaisu yhtälöön `2x+c`

```
In[131]:= yhtalo = 2*x + c;
          Solve[{x+c==3,x-2c==-3},{x,c}]
          yhtalo /. %[[1]]
Out[132]= {{x -> 1, c -> 2}}
Out[133]= 4
```

- Liian pitkän rivin voi katkaista painamalla <Enter>. Toiselle riville jatkuvan tekstin tulee olla sulkujen sisällä

```
In[138]:= Plot[{Sin[x],Cos[x]},{x,-Pi,Pi},GridLines->
          Automatic, PlotLabel->"Sini ja cosini"]
Out[138]= -Graphics-
```

Hyödyllisiä esimerkkejä 2/2

- Kertolasku (huomaa: ”`xy`” on yksi symboli, ”`x y`” on sama kuin ”`x*y`”):

```
In[142]:= 4*x + 5y - x y - 3x + xy
          4*x + 5y - x y - 3x + x y
Out[142]= x + xy + 5 y - x y
Out[143]= x + 5 y
```

- Matriisien kertolasku (pistetulo) (`.`)
  - Huom! `*` operaattori kertoo matriisit alkioittain.

```
In[144]:= A.B
```

Tehtävä A: Tutustuminen Mathematicaan

Aloita komennolla `ClearAll["Global‘*"]}`, jotta alustuksen aikana tallennetut muuttujat pyyhkiytyvät. Tämän komennon ‘-hipsukan saat painamalla shift + backspacen vasemmalla puolella oleva näppäin.

Tee alusta alkaen käyttäen soluja mahdollisimman paljon.

1. Luo muutama matriisi, esim. `A={{1,2,3},{4,5,6},{7,8,9}}` ja `B={{a,b,c},{a,b,c},{a,b,c}}`. Kokeile matriisien yhteen- ja kertolaskua sekä alkioittain kertolaskua. Määrittele matriisit ja anna komennot samassa solussa.

2. Tutustu seuraaviin Mathematica-komentoihin: Clear, MatrixForm; Exp, Series, Normal, Range, N, Abs, Sqrt, Sin, Log, Simplify; D, Plot, Plot3D

   - Tutustumiseen pääset alkuun kokeilemalla näitä komentoja `?Clear`, `MatrixForm[A]`,

`?Series`, `Series[Exp[x], {x,0,5}]`,

`Normal[Series[Exp[x], {x,0,5}]]`,

`?N`, `N[Pi]`, `N[Pi,11]`

✍ Mitä tekee komento `Series[Sin[x],{x,0,3}]`? Miten tulostus muuttuu, jos `Series` laitetaan komentoon `Normal`.

✍ Kuinka saat esille piin likiarvon kahdenkymmenen merkitsevän numeron tarkkuudella?

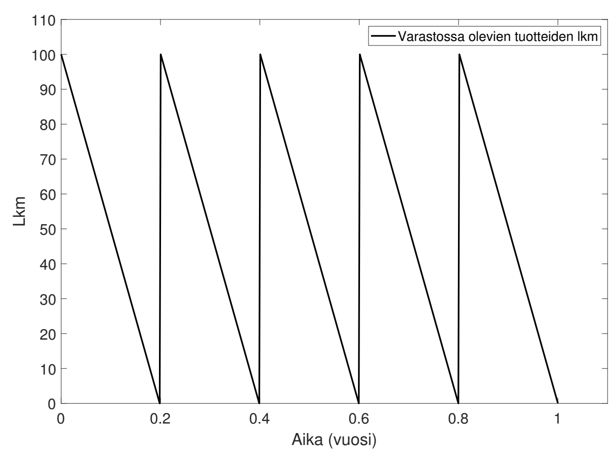
✍ Miten ratkaiset funktion $f(x) = x^2 + 3x - 5$ nollakohdan Mathematicassa?

🖩 Liitä vastauksiisi kuva, johon on piirretty sinifunktion ja sen viidennen asteen Taylor-approksimaation kuvaaja välillä $-2\pi \leq x \leq 2\pi$ sekä ruudukko. Anna kuvan otsikoksi oma nimesi.

Tehtävä B: EOQ-malli

Erään varastoitavan tuotteen kysyntä on D yksikköä vuodessa. Varastoa täydennettäessä tilauskustannus on C_1 per tilauskerta. Varastointikustannus on C_2 per varastossa oleva yksikkö per vuosi. Olkoon Q kerrallaan tilattavien yksiköiden lukumäärä.

Kuvassa esimerkki varaston toiminnasta. Siinä näkyy varastoitavan tuotteen lukumäärä varastossa ajan funktiona. Esimerkin varastolla kysyntä D on 500 ja kerralla tilattavien tuotteiden määrä Q on 100.



Tehtävä B: EOQ-malli

- ✍ Mikä funktio kuvaa kokonaiskustannuksia EOQ-mallissa?

  Vinkki: Tot. cost $= C_1 \times$ tilausten lkm. $+ C_2 \times$ kesk. varasto. Mieti mikä on tilausten lkm ja keskimääräinen varasto!

- ✍ Mikä on optimaalinen tilauskoko?

- ✍ Mikä on optimaalinen tilauskertojen lukumäärä?

- ✍ Kuinka suuret ovat tällöin kokonaiskustannukset?

Vinkki: Mistä löydät funktion ääriarvot?

Tehtävä C: Rosenbrockin banaanilaakso

Tarkastellaan funktiota

$$f(x_1, x_2) = 100 \cdot (x_2 - x_1^2)^2 + (a - x_1)^2 \tag{1}$$

missä a on vakioparametri.

✍ Millä x_1:n ja x_2:n arvoilla funktio minimoituu ja mikä on funktion arvo minimipisteessä, kun a on tuntematon vakio?

Vinkit: Luo funktio. Minimipiste löytyy gradientin nollakohdasta.

🖩 Muodosta 3D-pinta funktiosta minimipisteen läheisyydessä, kun $a = 1$. Lisää kuva palautukseesi.

Vinkki: Kuvan saat komennolla `Plot3D`.

Kotitehtävä 1: Mathematican sovelluksia

- Valitse ja katso yksi video verkkosivulta
  http://www.wolfram.com/mathematica/customer-stories/

✍ Mikä oli videolla esitetelty Mathematican sovellus? Miksi Mathematica nähtiin hyödylliseksi?

✍ Miten voit hyötyä Mathematicasta tämän kurssin ulkopuolella?

Kotitehtävä 2: Kokonaisdifferentiaalin laskeminen

Kun fysikaalinen suure F ei ole suoraan mitattavissa, mitataan suureiden $x_1, ..., x_n$ arvot ja lasketaan suureelle F arvo kaavalla $F(x_1, ..., x_n)$. Usein suureen F arvoon liittyvät virherajat lasketaan kaavalla

$$\Delta F = |\frac{\partial F}{\partial x_1}|\Delta x_1 + |\frac{\partial F}{\partial x_2}|\Delta x_2 + ... \qquad (2)$$

missä Δx_i on mittauksen x_i keskivirhe ja $\frac{\partial F}{\partial x_i}$ on F:n osittaisderivaatta x_i:n suhteen.

Kotitehtävä 2: Kokonaisdifferentiaalin laskeminen

1. Laske virhearvio linssin polttovälille, kun polttoväli saadaan kaavalla $f = \frac{ab}{a+b}$, jossa a on esineen etäisyys ja b kuvan etäisyys. Ohje: Laske ensin f:n gradientti käyttämällä komentoa D. Ota vastauksesta itseisarvo. Virhearvion saat ottamalla pistetulon saadusta listasta ja parametrien a ja b keskivirheistä.

✍ Mikä on virhearvion analyyttisen lauseke?

Vinkit: Tee virhemuuttujista lista ja ota tästä listasta ja polttovälin gradientin itseisarvosta pistetulo. Käytä `Simpilfy` -komentoa tuloksen sieventämiseksi.

✍ Sijoita lausekkeen mittaustulokset $a = 85 \pm 1$ mm ja $b = 196 \pm 2$ mm. Mitä saat polttoväliksi ja sen virherajoiksi?

🖩 Liitä laatimasi Mathematica-koodi vastauksiin.

Kotitehtävä 2: Kokonaisdifferentiaalin laskeminen

2. Haluat seuraavaksi mitata ympyräsektorin muotoisen tontin pinta-alan $A(r, \phi) = \frac{\phi}{2\pi}\pi r^2$, siten että virheen suuruus on korkeintaan 0,5m^2. Pystyt mittaamaan kulman teodoliitillä tarkkuudella ± 0.01 astetta. Kuinka tarkasti tontin säde on pystyttävä mittaamaan, jotta virhe pysyy annetuissa rajoissa? Ohje: Laske kokonaisdifferentiaali r:n ja ϕ:n suhteen ja ratkaise virhearvion lauseke. Ratkaise Solve-komentoa käyttämällä lausekkesta Δr:n.

✍ Mikä on Δr:n analyyttinen lauseke?

Huom! älä sijoita mitään arvoja analyyttistä lauseketta ratkaistessasi.

✍ Tiedetään, että $r \approx 50$m, $\phi \approx 2\pi/3$ ja pinta-alan virhe ΔA saa olla enintään 0.5m^2. Kuinka suuri Δr saa suurimmillaan olla?

Huomioi, että tehtävänannossa sekä radiaaneja, että asteita. Käytä samaa yksikköä arvoja sijoittaessasi.

Liitä vastauksiin käyttämäsi Mathematica-koodi.