

Inlämningsuppgift i kursen inledande matematik

Anders Logg, 760107–xxxx

12 oktober 2013

1 Introduktion

I den här uppgiften¹ studerar vi funktionen

$$f(x) = \begin{cases} \frac{\sin x}{x}, & x \neq 0, \\ 1, & x = 0. \end{cases} \quad (1)$$

Notera att från l'Hôpitals regel (se [1], avsnitt 4.3) vet vi att

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = \lim_{x \rightarrow 0} \frac{\cos x}{1} = 1, \quad (2)$$

vilket innebär att funktionen är kontinuerlig.

2 Lösningar till uppgifter

Uppgift 1

Plotta funktionen (1) i MATLAB och i Python. Inkludera koden samt en av plottarna i rapporten.

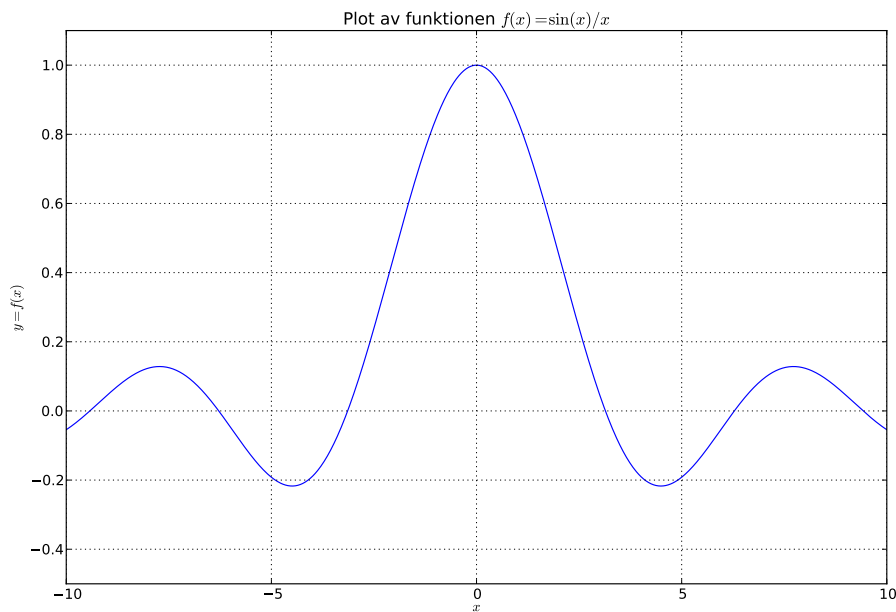
Vi börjar med att plotta funktionen i MATLAB. Såhär ser koden ut:

MATLAB code

```
1 x = linspace(-10, 10, 1000); % Note, does not contain x = 0
2 y = sin(x) ./ x;
3
4 plot(x, y)
5 xlabel('x')
6 ylabel('y=f(x)')
7 title('f(x)=sin(x)/x')
8 axis([-10, 10, -0.5, 1.1])
9 grid on
```

Sedan plottar vi funktionen i Python. Notera att koden ser nästan likadan ut (med några små skillnader). Detta är för att paketet `pylab` som används är gjort för att likna MATLAB.

¹Detta är en påhittad uppgift som illustrerar hur man kan använda L^AT_EX för att skriva snygga rapporter.



Figur 1: En snygg plot av funktionen definierad i (1).

Python code

```

1 from pylab import *
2
3 x = linspace(-10, 10, 1000) % Note, does not contain x = 0
4 y = sin(x) / x
5
6 plot(x, y)
7 xlabel('x')
8 ylabel('$y = f(x)$')
9 title('f(x) = $\sin(x) / x$')
10 axis([-10, 10, -0.5, 1.1])
11 grid(True)
12
13 show()
```

Plotten av funktionen (1) kan ses i figur 1. Notera att figuren placeras automatiskt av L^AT_EX på lämplig plats för att det skall bli så snyggt som möjligt (undviker onödiga tomma utrymmen i dokumentet som resultat av sidbrytningar).

För att illustrera hur man kan inkludera koden i dokumentet utan att behöva klistra in den² i dokumentet kommer här koden på nytt.

MATLAB code

```

1 x = linspace(-10, 10, 1000); % Note, does not contain x = 0
2 y = sin(x) ./ x;
```

²Detta är ju väldigt praktiskt eftersom man då kan uppdatera koden utan att behöva klistra in den på nytt!

```
3
4 plot(x, y)
5 xlabel('x')
6 ylabel('y=f(x)')
7 title('f(x)=sin(x)/x')
8 axis([-10, 10, -0.5, 1.1])
9 grid on
```

Python code

```
1 from pylab import *
2
3 x = linspace(-10, 10, 1000) % Note, does not contain x = 0
4 y = sin(x) / x
5
6 plot(x, y)
7 xlabel('x')
8 ylabel('$y = f(x)$')
9 title('f(x) = $\sin(x) / x$')
10 axis([-10, 10, -0.5, 1.1])
11 grid(True)
12
13 show()
```

Uppgift 2

Studera konvergensen för $\sin(x)/x \rightarrow 1$ då $x \rightarrow 0$ numeriskt genom att beräkna kvoten för $x = h$ och $h = 0.1, 0.01, \dots, 10^{-16}$. Gör detta genom att skriva ett program för beräkningen i såväl MATLAB som Python. Presentera resultatet i två tabeller.

Här visar vi först koden för vårt MATLAB-program.

MATLAB code

```
1 h = 1;
2
3 hlist = [];
4 flist = [];
5
6 for n = 1:16
7     h = h / 10;
8     f = sin(h) / h;
9
10    hlist = [hlist h];
11    flist = [flist f];
12 end
13
14 % Print values as a table (matrix)
15 format long
16 [hlist' flist']
```

h	$f(h)$
1.0000000000000000e-01	9.98334166468282e-01
1.0000000000000000e-02	9.99983333416666e-01
1.0000000000000000e-03	9.99998333333342e-01
1.0000000000000000e-04	9.9999998333333e-01
1.0000000000000000e-05	9.9999999983333e-01
1.0000000000000000e-06	9.9999999999833e-01
1.0000000000000000e-07	9.9999999999998e-01
1.0000000000000000e-08	1.00000000000000e+00
1.0000000000000000e-09	1.00000000000000e+00
1.0000000000000000e-10	1.00000000000000e+00
1.0000000000000000e-11	1.00000000000000e+00
1.0000000000000000e-12	1.00000000000000e+00
1.0000000000000000e-13	1.00000000000000e+00
1.0000000000000000e-14	1.00000000000000e+00
1.0000000000000000e-15	1.00000000000000e+00
1.0000000000000000e-16	1.00000000000000e+00

Tabell 1: Konvergensresultat i MATLAB.

Motsvarande kod i Python följer härunder. Notera att eftersom vi är lata, så ser vi till att skriva ut värdena på ett sådant sätt att de enkelt kan klistras in i en $\text{\LaTeX}$ -tabell.

Python code

```

1 from math import sin
2
3 h = 1.0
4
5 hlist = []
6 flist = []
7
8 for n in range(16):
9     h = h / 10
10    f = sin(h) / h
11
12    hlist.append(h)
13    flist.append(f)
14
15 # Print values for easy cut-and-paste into a LaTeX table
16 print "\\begin{tabular}{ll}"
17 for n in range(16):
18     print " %.16g & %.16g \\\\" % (hlist[n], flist[n])
19 print "\\end{tabular}"

```

Resultaten visas i tabell 1 och tabell 2. Notera att $\text{\LaTeX}$ själv placerar in dessa tabeller på lämplig plats, precis som med figurer (de är så kallade “floats”).

h	$f(h)$
0.1	0.9983341664682815
0.01	0.9999833334166665
0.001	0.9999998333333416
0.0001	0.9999999983333334
1e-05	0.9999999999833332
1e-06	0.9999999999983334
1e-07	0.999999999999983
1e-08	1
1e-09	1
1e-10	1
1e-11	1
1e-12	1
1e-13	1
1e-14	1
1e-15	1
1e-16	1

Tabell 2: Konvergensresultat i Python.

Referenser

- [1] R. A. Adams and C. Essex. *Calculus: A Complete Course*. Pearson, 2013.