

UPPSALA UNIVERSITET

FÖRELÄSNINGSANTECKNINGAR VT22

# Ordinära Differentialekvationer

*Rami Abou Zahra*

## CONTENTS

1. Interpolation	2
1.1. Interpolation med polynom	2
1.2. Numerisk integration	3
1.3. Numerisk kvadratur	3
1.4. Newton-Coates kvadratur	3
1.5. Sammansatta trapetsregeln	4

## 1. INTERPOLATION

Givet  $n$  punkter  $(x_i, y_i) \in \mathbb{N}$  där alla  $x_i$  är olika. Då vill vi anpassa en funktion  $p(x)$  till detta data så att den går igenom alla dessa punkter, det vill säga  $p(x_i) = y_i \forall i \in \mathbb{N}$ . Detta kallas för interpolation.

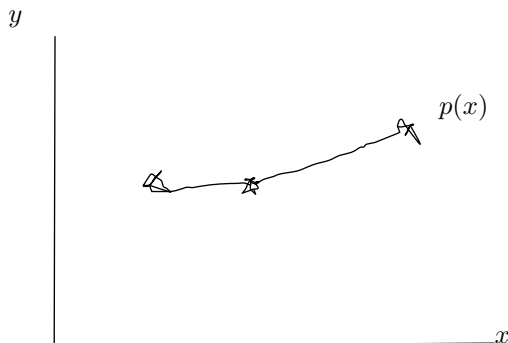


FIGURE 1. Interpolation idé

Vanliga användningsområden kan vara:

- Att läsa mellan raderna i en tabell, även om vi bara har data för  $x = 3$  och  $x = 2$  så kan vi finna värden för  $x = 2.5$
- Anpassa matematisk modell till data.
- Approximera en "svår" funktion, ex. Weierstrass, med en enklare funktion.

## 1.1. Interpolation med polynom.

- Interpolera alla datapunkter med ett enda polynom:
  - $p_{n-1}(x) = \sum_{i=0}^{n-1} a_i x_i$  där  $n - 1$  är polynomgraden

I allmänhet krävs det polynom av grad  $n - 1$  för att interpolera  $n$  punkter.

Vi sätter in de punkter vi vill lösa i polynomet och löser för koefficienterna. Vi kommer då få ett ekvationssystem.

Ansats med gradtal mindre än  $n - 1$  ger generellt ingen interpolationenslösning.

Vid ansats med gradtal högre än  $n - 1$  så finns ingen unik lösning.

Läs gärna vidare om Newton interpolation.

- Styckvis polynominterpolation:
  - Anpassar med polynom av låg grad (exvis grad 1 så att den blir linjär) på varje delintervall (mellan varje punkt).
  - $p_k^j(x)$  för  $x \in [x_j, x_{j+1}]$  så att  $j = 1, \dots, n - 1$  och att punkterna är ordnade så att  $x_j < x_{j+1}$ . Vanligtvis är  $k = 1$  (graden är 1, styckvis linjär) eller  $k = 3$  (styckvis kubisk).
  - Vid kubisk interpolation får vi 2 styckna "frihetsgrader", det vill säga  $x, x^2$ . Dessa brukar kombineras med en metod som kallas för *spline* för att ge en kontinuerlig första och andra derivata i hela interpolationen så att kurvan blir mer *smooth*.
  - En annan variant kallas för *pchip* som enbart ger en kontinuerlig första-deriv. Den är monoton mellan datapunkter.

## 1.2. Numerisk integration.

Problemet som vi vil lösa är att vi förstås vill beräkna integralen av en given funktion:

$$I = \int_a^b f(x)dx$$

Speciellt när det inte går att analytiskt integrera funktionen så blir det som intressantast, eller att det är väldigt svårt, eller då  $f$ -värden endast finns som mätvärden i vissa punkter.

## 1.3. Numerisk kvadratur.

$$I \approx \sum_{i=1}^n w_i f(x_i)$$

Där  $w_i$  är "vikter". Då är frågan hur man skall bestämma dessa vikter och vilka punkter  $x_i$  som är relevanta att ta med.

Vi gör detta baserat på de interpolerade polynomen och polynom kan vi integrera analytiskt.

## 1.4. Newton-Coates kvadratur.

Vi väljer  $x_1, x_2, \dots, x_n$  värden som ekvi-distanta punkter. Sedan approximeras integralen  $\int_a^b f(x)dx$  av  $\int_a^b p(x)dx$  där  $p(x)$  är polynomet av grad  $n-1$  som interpolerar alla punkter  $(x_i, f(x_i)), i = 1, \dots, n$

Exempel: När jag bara har två punkter ( $n=2$ ) kallas det för *trapetsregeln*:

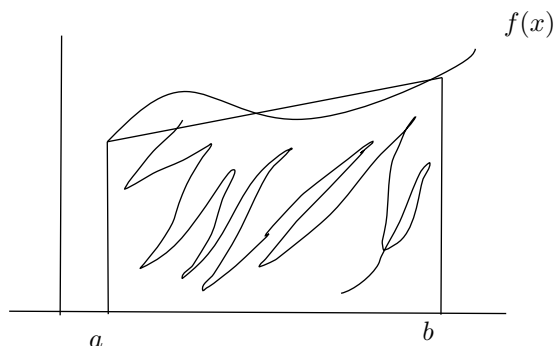


FIGURE 2. Trapetsregeln

En bättre approximering än att använda en första-gradare är simpsons formel ( $n=3$ ):

Det som kanske verkar rimligt kanske är att ta högre grad, men det som händer är att man får starka oscillationer eftersom ju högre grad desto fler "kupor". Istället tar vi mindre intervall och kör bitvis linjär/simpsons istället.

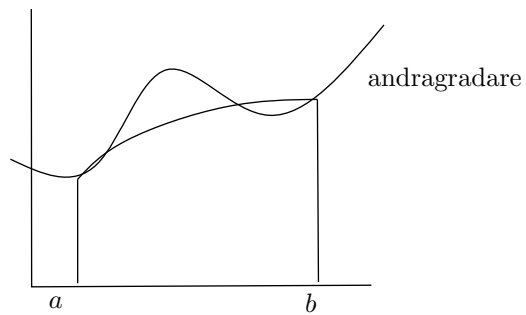


FIGURE 3. Simpsons



FIGURE 4. Exempel: 4 intervall

### 1.5. Sammansatta trapetsregeln.

Generella trapetsformeln:

$$x_i = a + ih, h = \frac{b-a}{n}$$

$$\int_a^b f(x)dx \approx T(h) = h \left( \sum_{i=0}^n f(x_i) - \frac{f(a) + f(b)}{2} \right)$$

Simpsons formel:

$$x_i = a + ih, h = \frac{b-a}{n}$$

$$\int_a^b f(x)dx \approx S(h) = \frac{h}{3} (f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n))$$

$n$  måste vara jämn.