100002

# Numeriese Metodes / *Numerical Methods* 324

Rekenaaropdrag 03 / *Computer Assignment* 03 - 2018

**Sperdatum / *Deadline* - 16/03/2018**

**Let asseblief op:**

- Opdragte word met die aanvang van die tutoriaalsessie op die sperdatum ingehandig. Laat inhandiging sal gepenaliseer word. Handig gedrukte weergawes in asb, geen elektroniese weergawes word aanvaar nie. **Jou opdrag moet beide jou kode en jou resultate insluit.**
- Alle werk ingehandig moet jou eie werk wees. Jy kan idees uitruil met ander studente, maar alle berekeninge en grafieke moet jou eie wees. **Stappe sal geneem word teen plagiaat.**
- Rekenaaropdragte moet in MATLAB of Python voltooi word. As jy ‘n alternatiewe program wil gebruik, maak eers seker dat dit reg is met die dosent.
- Wanneer opdragte ingehandig word, is die voorkeurmetode MATLAB se `publish` of `livescript` funksie, maar dis nie verpligtend nie. Maar dit is heeltemal aanvaarbaar om jou kode in ’n MS Word dokument te plak of dit in LaTeX op te tik.

**Please Note:**

Computer assignments are handed in at the start of the tutorial session on the due date. There is a penalty if you hand in late. Please hand in hard copy, no electronic versions are accepted. 2**You should include both your code and your results.**

All work handed in must be your own work. You may exchange ideas with other students, but all calculations and graphs must be your own. **Action will be taken against plagiarism.**

Computer assignments must be completed using MATLAB or Python. If you would like to use an alternative, please clear this in advance with the course lecturer.

The preferred method for submitting assignments is MATLAB’s `publish` or `livescript` function, but this is not compulsory. However, it is perfectly acceptable to paste your code to an MS Word document or to type it up in LaTeX.

**Probleme:**

1. Onthou die Bessel funksies $J_\nu(x)$ van Rekenaar opdrag 01. Alhoewel ons net die geval waar $\nu$ ’n heelgetal was oorweeg het, is dit nie noodwendig die geval nie. In hierdie probleem interpoleer ons $J_\nu(x)$ by heelgetal waardes van $\nu$ en gebruik hierdie om the waardes tussen die te benader.

   **(a)** Gebruik jou sagteware om weer die volgende waardes van $J_\nu(1)$ te bereken (bv. gebruik `besselj`) Gee die resultate tot tenminste 10 beduidende syfers.

**Problems:**

1. Recall the Bessel functions $J_\nu(x)$ from Computer Assignment 01. Although there we considered only the case where $\nu$ was an integer, this need not be the case. In this problem we interpolate $J_\nu(x)$ at integer values of $\nu$ and use this to approximate between these values.

   **(a)** Use your software to compute again the following values of $J_\nu(1)$ (e.g., using `besselj`). Give the results to at least ten significant digits.

| $\nu$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $J_\nu(1)$ | ? | ? | ? | ? |

**(b)** Implementeer die barycentriese interpolasie formule van Bladsy 4 van Lesing 3 vir die data hierbo en plot dus die polinoominterpolante $p(\nu)$ en $J_\nu(1)$ op dieselfde stel asse, vir $\nu$ in $[0, 3]$. Plot ook die fout, $p(\nu)-J_\nu(1)$ as ’n funksie van $\nu$ op ’n ander figuur.

**(c)** Gebruik die begrensing op Bladsy 4 van Lesing 12 om ’n teoretiese skatting vir die maksimum fout $\max|J_\nu(x) - p(\nu)|$ te bereken. (Wenk: neem $M_4 = 3$.) Hoe vergelyk hierdie met die fout bereken in deel (b)?

**(b)** Implement the barycentric interpolation formula from Slide 4 of Lecture 3 for the data above and hence plot the polynomial interpolant $p(\nu)$ and $J_\nu(1)$ on the same set of axis, for $\nu$ in $[0, 3]$. Also plot the error, $p(\nu) - J_\nu(1)$, as a function of $\nu$ on a different figure.

**(c)** Use the bound on slide 4 of Lecture 12 to derive a theoretical estimate for the maximum error $\max|J_\nu(x)-p(\nu)|$. (Hint: take $M_4 = 3$.) How does this compare to the error computed in part (b)?

(d) Differensieer die nodal polinoom $\pi(x) = x(x-1)(x-2)(x-3)$ met die hand en gebruik `roots` om sy maxima/minima te bepaal. Sluit dit af met $\max_{0\leq x\leq 3}|\pi(x)| = 1$ en dus verbeter die begrensing van deel (c).

2. Herhaal probleem 1(b), maar nou deur gebruik te maak van kubiese spline interpolante. Gebruik beide (i) not-a-knot voorwaardes en (ii) 'n vasgeklem spline met $s'(0) = s'(3) = 0$. Lewer komentaar op die ooreenkomste en verskille in vergelyking met die polinoominterpolant wat vroeër gebruik is.

3. **(a)** Maak 'n <u>kolom</u>vektor $\boldsymbol{x}$ wat die 4-punt Chebyshev rooster op die interval $[-1, 1]$ bevat. Gebruik die formule op bladsy 5 van Lesing 14 om die matrix $V$ te maak waarvan die $ij$de inskrywing die graad $j$ Chebyshev polinoom voorstel wat geëvalueer word by die $i$de Chebyshev punt, m.a.w., $T_j(x_i)$. (Hierdie word die Chebyshev–Vandermonde matriks genoem, en is ekwivalent aan die "Diskrete cosinus transform" (DCT). As jy die Fourier Analise of Beelverwerkingsprosesse kursusse vat, sal jy hierdie gereeld teëkom.)

   **(b)** Bereken die oplossing vir $V\boldsymbol{c} = \exp(\boldsymbol{x})$ (gebruik \ in MATLAB of iets soortgelyk.) Wat stel die vektor $\boldsymbol{c}$ voor?

   **(c)** Gebruik die ongelykheid op Bladsy 10 van Lesing 13 om te skat hoeveel Chebyshev punte, $x_1, \ldots, x_n$, sal nodig wees om die funksie $\exp(x)$ te benader tot 'n akkuraatheid van $10^{-10}$ in die interval $[0, 3]$.

   **(d)** Gebruik die tegniek op Bladsy 9 van Lesing 13 om die $n$ Chebyshev punte $x_1, \ldots, x_n$, op die interval $[0, 3]$ te bereken, waar $n$ die getal is wat jy in (c) bereken het.

   **(e)** Konstrueer 'n polinoom interpolant vir die data $(x_k, e^{x_k})$, $k = 1, \ldots, n$ deur enige manier te gebruik (soos `polyfit` of die barycentric formule op bladsy 12 van Lesing 13) en plot die fout in jou benadering tot $e^x$ op 'n fyn rooster in $[0, 3]$ om jou resultate van (c) te bevestig.

   Let op: As jy kies om `polyfit` te gebruik vir deel 3(d), moet nie bekommerd wees oor enige boodskappe wat jou waarsku teen 'ill-conditioning' nie. Die rede hiervoor is dat alhoewel die Chebyshev punte 'n goeie stel is, die `polyfit`/`polyval` proses is '*ill-conditioned*' (in besonders, berekening van die polinoom koëffisiënte van die interpolant in die basis $\{x^0, x^1, \ldots, ..., x^{n-1}\}$ is 'ill-conditioned') Die barycentric interpolasie formule ly nie aan hierdie probleem nie, en ook nie die benadering in 3(b) nie.

**(d)** Differentiate the nodal polynomial $\pi(x) = x(x-1)(x-2)(x-3)$ by hand and use `roots` to determine its maxima/minima. Conclude that $\max_{0\leq x\leq 3}|\pi(x)| = 1$ and hence improve the bound from part (c).

2. Repeat Problem 1(b), but now using cubic spline interpolants. Use both (i) not-a-knot conditions and (ii) a clamped spline with $s'(0) = s'(3) = 0$. Comment on the similarities and differences compared to the polynomial interpolant used earlier.

3. **(a)** Make a <u>column</u> vector $\boldsymbol{x}$ containing the 4-point Chebyshev grid on the interval $[-1, 1]$. Using the formula on page 5 of Lecture 14 to make the matrix $V$ whose $ij$ entry is the degree $j$ Chebyshev polynomal evaluated at the $i$th Chebyshev point, i.e., $T_j(x_i)$. (This is often called the Chebyshev–Vandermonde matrix, and it turns out that it's equivalent to something called the "Discrete cosine transform" (DCT). If you take the Fourier Analysis or Image Processing courses, you will encounter these often.)

   **(b)** Compute the solution to $V\boldsymbol{c} = \exp(\boldsymbol{x})$ (using \ in MATLAB or some equivalent). What does the vector $\boldsymbol{c}$ represent?

   **(c)** Use the inequality on slide 10 of Lecture 13 to estimate how many Chebyshev points, $x_1, \ldots, x_n$, will be necessary to approximate the function $\exp(x)$ to an accuracy of $10^{-10}$ on the interval $[0, 3]$.

   **(d)** Use the technique from slide 9 of Lecture 13 to compute the $n$ Chebyshev points $x_1, \ldots, x_n$, on the interval $[0, 3]$, where $n$ is the number you determined in part (c).

   **(e)** Construct a polynomial interpolant to the data $(x_k, e^{x_k})$, $k = 1, \ldots, n$ using any means (such as `polyfit` or the barycentric formula on slide 12 of Lecture 13) and plot the error in your approximation to $e^x$ on a fine grid in $[0, 3]$ to valid your result from (c).

   Note: If you choose to use `polyfit` for part 3(d), do not worry about any messages warning you of ill-conditioning. The reason for this is that the even though the Chebyshev points are good point set, the `polyfit`/`polyval` process is *ill-conditioned* (in particular, computing the polynomial coefficients of the interpolant in the basis $\{x^0, x^1, \ldots, ..., x^{n-1}\}$ is ill-conditioned). The barycentric interpolation formula does not suffer from this problem, nor does the approach in 3(b).