Spring 2017: Numerical Analysis
Assignment 7 (due May 4, 2017)

1 extra credit point will be given for cleanly plotted and labeled figures (see also rules on the first assignment). Use a legend and different line styles to label multiple graphs in one plot (no colors needed). Do not export figures using raster graphics (.jpg, .png) but use vector graphics (.eps, .pdf, .dxf) that do not mess up lines. Label axes and use titles. Use `help plot`, `help legend`, `help xlabel` to better understand MATLAB's plotting capabilities.

1. **[Interpolation and optimal norm approximation, 2+1+1+1pt]** For an interval (a, b), $n \in \mathbb{N}$ and disjoint points $x_0, \ldots, x_n$ in $[a, b]$, we define for polynomials p, q

$$\langle p, q\rangle := \sum_{i=0}^{n} p(x_i)q(x_i).$$

   (a) Show that $\langle \cdot\,, \cdot\rangle$ is an inner product for each $\mathcal{P}_k$ with $k \leq n$, where $\mathcal{P}_k$ denotes the space of polynomials of degree k or less.

   (b) Why is $\langle \cdot\,, \cdot\rangle$ not an inner product for $k > n$?

   (c) Show that the Lagrange polynomials L_i corresponding to the nodes $x_0, \ldots, x_n$ are orthonormal with respect to the inner product $\langle \cdot\,, \cdot\rangle$.

   (d) For a continuous function $f : [a, b] \to \mathbb{R}$, compute its optimal approximation in $\mathcal{P}_n$ with respect to the inner product $\langle \cdot\,, \cdot\rangle$ and compare with the interpolation of f.

2. **[Euler and trapezoidal methods, 2+2pt]**

   (a) Consider the initial value problem (IVP) $y' = t(1 - e^y)$ with $y(t_0) = 1$ and $t_0 = 0$. Compute y_2 using Euler's method with mesh size h (give your answer in terms of h).

   (b) Consider the initial value problem $y' = -y^2$ with $y(t_0) = 1$ and $t_0 = 0$. Compute y_1 using the trapezoidal method with mesh size h (give your answer in terms of h).

3. **[Higher-order one step method, 1+1+1+2pt]** Consider the IVP $y' = f(t, y)$ with initial value $y(0) = y_0$. We define a higher-order one-step method $y_{n+1} = y_n + h\Phi(t_n, y_n; h)$, where

$$\Phi(t, y; h) = f(t, y) + \frac{h}{2}(\partial_t f(t, y) + f(t, y)\partial_y f(t, y)),$$

where ∂_t and ∂_y denotes partial derivatives with respect to t and y.

   (a) Justify this choice of Φ using the Taylor expansion:

$$y(t + h) = y(t) + hy'(t) + \frac{1}{2}h^2 y''(t) + O(h^3).$$

   (b) Find an expression for the truncation error T_n for this method, in terms of an $\eta_n \in [t_n, t_{n+1}]$, and y'''.

(c) What is the order of accuracy of this method? Justify your answer.

(d) Using MATLAB, solve the IVP $y' = (1 - \frac{4t}{3})y$, $y(0) = 1$ with Euler's method and with the one-step method above, with $N = 100$, $t \in [0, 1]$. By finding the explicit solution to the IVP (by hand), compute the error $e_N = y_N - y(1)$ for the two methods. Please hand in your code.

4. **[Implicit and Explicit Euler, 2+1+2+(2+1)pt]** Consider the linear IVP $y' = -100y$ with $y(0) = y_0$.

   (a) Write down Euler's method with step size h and find an explicit formula for y_n in terms of y_0. Repeat for the backward Euler method.

   (b) For the two approximations, if we take $h = 0.1$, what happens to y_n as $n \to \infty$? Which method is more consistent with the limit of the actual solution $y(t)$ as $t \to \infty$?

   (c) Implement the forward Euler method for the IVP $y' = -100y + y^2$ in MATLAB. Use the time interval $[0, 1]$, the initial condition $y(0) = 1$. Compute and plot the solutions for step sizes $h = 0.1, 0.02, 0.001$.

   (d) **[Extra credit]** Implement the backward Euler method. At each step, you will need to solve $y_{n+1} = y_n + hf(x_{n+1}, y_{n+1})$, which constitutes a nonlinear scalar equation for y_{n+1}. This can be done using Newton's method (Chapter 1!), i.e., we define $g(z) = z - y_n - hf(x_{n+1}, z)$ and then, to compute the y_{n+1}, we iterate (here, the superindex k is the Newton iteration index:

$$y_{n+1}^0 = y_n + hf(x_n, y_n)$$
$$y_{n+1}^{k+1} = y_{n+1}^k - \frac{g(y_{n+1}^k)}{g'(y_{n+1}^k)}.$$

   A few (e.g., 5) iterations are usually sufficient to obtain a good result for y_{n+1}.

   (e) **[Extra credit]** Repeat (c), but using the implementation from (d). Discuss the differences in behavior compared to (c).

   Please hand in your code.

5. **[Error behavior, (2)+2+2+1pt]** Consider the IVP $y' = f(x, y)$, for $f(x, y) = x \sin(y)$ and $y(0) = \pi/2$ for $x \in [0, 3] =: I$. We showed in class that for the forward Euler method, the following error estimate holds:

$$|e_n| \leq \frac{M_2}{2L} \left(e^{L(x_n - x_0)} - 1\right) h,$$

   where $e_n = y(x_n) - y_n$, L, the Lipschitz constant of f, $M_2 = \max_{x \in I} |y''(x)|$ and h the step size.

   (a) **[Extra credit]** Verify that $y(x) = \pi - \arctan\left(\frac{2\exp(\frac{x^2}{2})}{\exp(x^2) - 1}\right)$ solves the IVP.

   (b) Show that the constants in the estimate can be choses as $L = 3$ and $M_2 = 10$ on I.[1]

[1] Compare with the derivations in Example 12.2 in the book, and with the class notes.

(c) Using the estimate, what is the step size h that guarantees that the error at $x = 3$ is less than $\epsilon := 10^{-2}$? Using your implemenation of the forward Euler Method and the analytical solution given in (a), what step size do you actually need to obtain the desired tolerance?

(d) Using the step sizes $h = \frac{1}{2^j}$ for $j = 1, 2, \ldots, 5$, report the errors at $x = 3$. Use the theoretical estimate to explain the error behavior, i.e., how does the error change as h is decreased?