AMath 585
Homework #4
Due Thursday, February 20, 2020

Homework is due to Canvas by 11:00pm PDT on the due date.

To submit, see `https://canvas.uw.edu/courses/1352870/assignments/5253097`

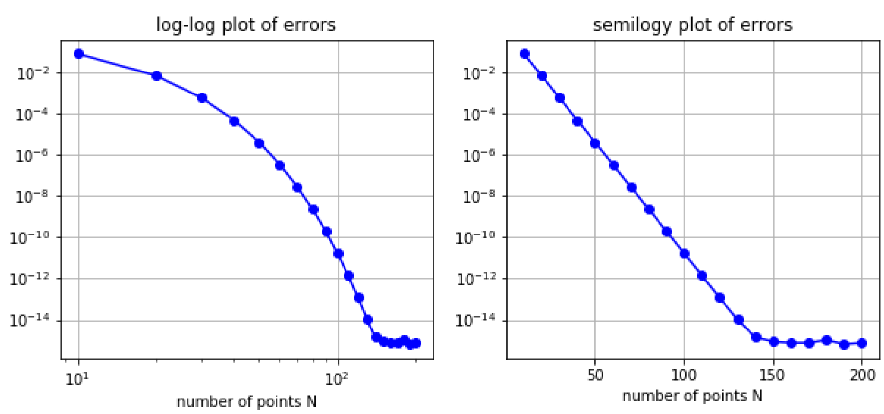
Preamble. Recall that "spectral accuracy" often means the error decays faster than any algebraic power of N^{-1} as we increase the number of points N used in our approximation.

Recall that if the error (in e.g. the max-norm) behaves like $E(N) \approx CN^{-p}$ as $N \to \infty$ then $\log E(N) \approx \log(C) - p\log(N)$ so we expect the slope to be $-p$ if we do a log-log plot of $E(N)$ vs. N. If we are seeing spectral accuracy then we expect the slope to keep decreasing as N grows, until rounding errors take over.

Sometimes we have even better accuracy than this suggests, and in fact $E(N) \approx C\exp(-\gamma N)$ for some constants $C, \gamma > 0$. This means the error is decaying exponentially fast with N (which in particular is faster than any algebraic power, but there are things in between).

If we have exponential convergence then we expect $\log(E(N)) \approx \log(C) - \gamma N$. Note that we now have N not $\log(N)$ on the right and so this looks linear in a semilogy plot in which we plot $\log(E(N))$ vs. N itself (e.g. using the `semilogy` command in Matlab or Python. The slope in such a plot shows the decay rate γ.

Here's a comparison of these two plots for the case of the Runge function $u(x) = 1/(1 + 16x^2)$ with interpolation at N Chebyshev points.



We see exponential convergence!

Problem 1. (a) Adapt the code from the `ChebyshevSpectral.ipynb` notebook to reproduce these figures. Use the max-norm of the error, approximated by evaluating the interpolating polynomial at 1000 points and comparing to the original function at these points.

(b) Change to this variant of the Runge function: $u(x) = 1/(1 + 25x^2)$, which is also commonly used. This function has poles at $\pm 0.2i$ in the complex plane, closer to the real axis than the previous function. As a result the convernence is not quite as fast, though still exponential. Make these plots.

It can be shown that the convergence is exponential provided the extension of the function $u(x)$ to the complex plane is analytic near the real axis, and the rate depends on how far out it is analytic.

(c) If the function is analytic in the entire complex plane then convergence is even faster than exponential, meaning the slope continues to decrease even in a semilogy plot.

Illustrate this for $u(x) = (x - 0.5)\sin(10x)$, using smaller values of N since convergence is so fast!

See e.g. Trefethen's books if you want to read more about this.

Problem 2. The notebook `ChebyshevSpectral.ipynb` illustrates how to approximate $u'(x)$ by doing Chebyshev polynomial interpolation to get $p(x)$ and then computing $p'(x)$. Extend this to give an approximation to $u''(x)$ by computing $p''(x)$. You will have to write a function to compute $T_n''(x)$ for each basis function $T_n(x)$, similar to what was done for $T_n'(x)$.

Test this out by computing the second derivative of these functions, and also produce semilogy plots of the errors as N is increased:

- $u(x) = \sin(2x)$,
- $u(x) = (x - 0.5)\sin(10x)$,
- $u(x) = 1/(1 + 16x^2)$.

Comment on any interesting behavior you observe.

Problem 3.

Now write a routine to solve the Boundary value problem

$$u''(x) = f(x), \quad -1 \leq x \leq 1,$$

with Dirichlet boundary conditions $u(-1) = \alpha,\ u(1) = \beta$, using spectral collocation at the Chebyshev points.

Test it out at least on the case where the true solution is $u(x) = \sin(2x)$ (which defines $f(x)$ and the boundary values) and produce a semilogy plot of the errors.

Hints: You need to set up and solve a linear system of equations for the coefficients c_n in the polynomial

$$p(x) = \sum_{n=0}^{N} c_n T_n(x).$$

The equations in the system include the boundary conditions,

$$\sum_{n=0}^{N} c_n T_n(-1) = \alpha, \quad \sum_{n=0}^{N} c_n T_n(1) = \beta,$$

and then the collocation equation at each interior node ($j = 1, 2, \ldots, N - 1$):

$$\sum_{n=0}^{N} c_n T_n''(x_j) = f(x_j).$$

Recall also that if you define $x_j = \cos(j\pi/N)$ then these are ordered from right to left with $x_0 = 1$ and $x_N = -1$.