I've copied the assignment below verbatim, it helps me keep track of things...

Problem Statement

Consider the singular Volterra integral equation

$$x(s) = (1+s)^{-1/2} + \frac{\pi}{8} - \frac{1}{4}\arcsin\left(\frac{1-s}{1+s}\right) - \frac{1}{4}\int_0^s \frac{x(t)}{(s-t)^{1/2}}dt. \tag{1}$$

The exact answer is $x_e(s) \equiv (1+s)^{-1/2}$. Using a product integration analogue of the Trapezoid rule with constant step size h, solve the integral equation over the range $0 \leq s \leq 1$.

Quadrature Rule Derivation

Instructions

Determine a quadrature rule of the form

$$\int_0^{s_i} (s_i - t)^{-1/2} x(t)dt = \sum_{j=0}^{i} w_{ij} x(t_j) \tag{2}$$

where $t_i = s_i$, $i = 0, ..., N$. The weights w_{ij} are constructed by insisting that the rule be exact when $x(t)$ is a first degree polynomial. Identify the weights w_{ij}.

Solution

With the given discretization, we begin by partitioning the integral into the sum of $N+1$ integrals over sub-domains, i.e.

$$\int_0^{s_i} (s_i - t)^{-1/2} x(t)dt = \int_0^h (s_i - t)^{-1/2} x(t)\,dt + \int_h^{2h} (s_i - t)^{-1/2} x(t)\,dt + ... + \int_{(i-1)h}^{ih} (s_i - t)^{-1/2} x(t)\,dt.$$

Next we assume that $x(t)$ is given by a piece-wise linear, such that in each of the N intervals evenly spaced by the set of points $x(ih)$, $x(t)$ is given by linear interpolation. This implies

$$\begin{aligned}
&= \int_0^h (s_i - t)^{-1/2}\Big(x(0) + \frac{(t-0)(x(h)-x(0))}{h}\Big)\,dt \\
&\quad + \int_h^{2h} (s_i - t)^{-1/2}\Big(x(h) + \frac{(t-h)(x(2h)-x(h))}{h}\Big)\,dt \\
&\quad\ \vdots \qquad\qquad\qquad\qquad \vdots \\
&\quad + \int_{(i-1)h}^{ih} (s_i - t)^{-1/2}\Big(x((i-1)h) + \frac{(t-(i-1)h)(x(ih)-x((i-1)h))}{h}\Big)\,dt
\end{aligned}$$

Collapsing into summation notation, we have that

$$= h^{-1} \sum_{j=0}^{i-1} \int_{jh}^{(j+1)h} (s_i - t)^{-1/2} \Big(x(jh)h + (t - jh)(x((j+1)h) - x(jh)) \Big)$$
$$= h^{-1} \sum_{j=0}^{i-1} \int_{jh}^{(j+1)h} (s_i - t)^{-1/2} \Bigg[x(jh) \Big(((j+1)h - t) \Big) + x((j+1)h) \Big(t - jh \Big) \Bigg]$$

We can shift the indices, yielding the weights

$$w_{ij} = h^{-1} \begin{cases} \int_0^h (s_i - t)^{-1/2} (h - t) dt, & j = 0 \\ \int_{(j-1)h}^{jh} (s_i - t)^{-1/2} (t - (j-1)h) dt + \int_{jh}^{(j+1)h} (s_i - t)^{-1/2} ((j+1)h - t) dt, & j = 1, 2, ..., i-1 \\ \int_{(i-1)h}^{ih} (s_i - t)^{-1/2} (t - (i-1)h) dt, & j = i \end{cases}$$

which satisfy the given criteria. The relevant indefinite integral is given by

$$\int \frac{a + bt}{\sqrt{c + t}} dt = \frac{2}{3} \sqrt{c + t} \big(3a + b(t - 2c) \big) + C$$

therefore

$$w_{ij} = \begin{cases} \frac{2\sqrt{h}}{3} \big((3 - 2i)\sqrt{i} + 2(i-1)^{3/2} \big), & j = 0 \\ \frac{2}{3}\sqrt{h} (2(i - j + 1)^{3/2} - (2i - 2j + 3)\sqrt{i - j}) + \frac{2}{3}\sqrt{h} (2(i - j - 1)^{3/2} - (2i - 2j - 3)\sqrt{i - j}) \\ \frac{2}{3}\sqrt{h} (h(i-1) - h(i-3)), & j = i \end{cases}$$

simplifying gives

$$w_{ij} = \sqrt{h} \begin{cases} \frac{2}{3} \big((3 - 2i)\sqrt{i} + 2(i-1)^{3/2} \big), & j = 0 \\ \frac{4}{3} \Big((i - j + 1)^{3/2} - 2(i - j)^{3/2} + (i - j - 1)^{3/2} \Big), & j = 1, 2, ..., i-1 \\ \frac{4}{3}, & j = i \end{cases}$$

With this in mind we continue with a more general discussion of how to solve the now-linear system and an analysis of the results.

Implementation

Instructions

Solve the singular integral equation numerically and determine the N to give 6 decimal place accuracy. What is the observed order of convergence?

Solution

We implement our discretization of (1) by explicitly defining a vector $\boldsymbol{b}$ whose elements are given by

$$b_i = (1+s_i)^{-1/2} + \frac{\pi}{8} - \frac{1}{4}\arcsin\left(\frac{1-s_i}{1+s_i}\right)$$

Accordingly, we seek to obtain an approximate solution vector $\boldsymbol{sax}$ whose elements are satisfy $x_i \approx x(s_i)$.
Lastly, we construct the matrix of weights W whose elements are $1/4$ (the integral coefficient in (1)) the value of those derived in the last section. Notice that we only defined weights for $j \leq i$ Naturally we must ask: what are the remaining elements?
Consider the first few elements of $\boldsymbol{x}$. For small i, the integral is only evaluated for/approximated by a few quadrature points. The remaining weights should be zero.
Therefore the discretized system is given by

$$\boldsymbol{x} = \boldsymbol{b} - \frac{1}{4}\sum_{j=0}^{i} w_{ij} x(t_j) \tag{3}$$

$$= \boldsymbol{b} - W\boldsymbol{x} \tag{4}$$

$$= (I+W)^{-1}\boldsymbol{b} \tag{5}$$

where I is the identity matrix.

$$\tag{6}$$

Solving this system directly for large N yields an approximation qualitatively indistinguishable from the analytical solution, is shown in Figure 1. Focusing on the error (since fortunately we are given the solution!), we obtain 6th decimal place accuracy at $N = 94$ as shown in Figure 2. Plotting the error against resolution on a log-log scale, we obtain a slope of -2, indicating a power-law relationship

$$||\boldsymbol{x} - \boldsymbol{x_e}||_\infty \propto N^{-2}$$

with order of convergence 2. This is precisely what we expect for a trapezoidal method in a large N and negligible floating-point error context. Surprisingly, we obtain the theoretical result for very modest resolutions, indicating rapid convergence.

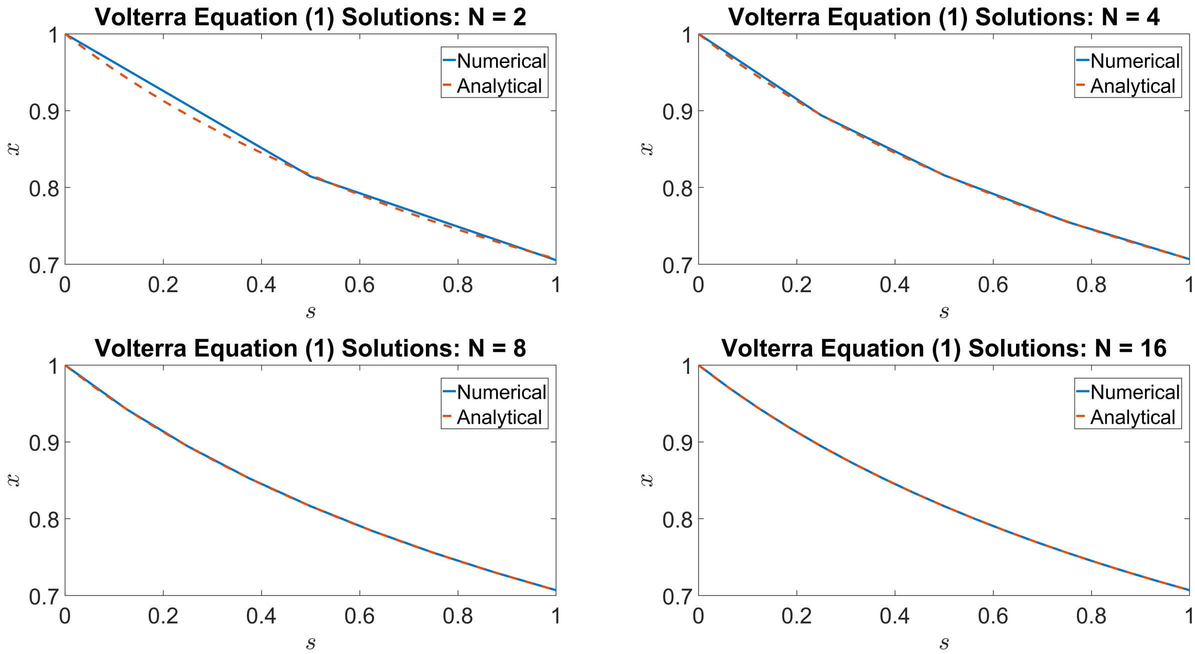

Figure 1: Solutions to the Singular Volterra (1): the numerical solution appears to converge to the analytical solution with increasing resolution N.

L-infinity Error Norms										
N	10	20	30	40	50	60	70	80	90	100
Error	8.28E-05	2.11E-05	9.47E-06	5.36E-06	3.44E-06	2.40E-06	1.77E-06	1.35E-06	1.07E-06	8.69E-07

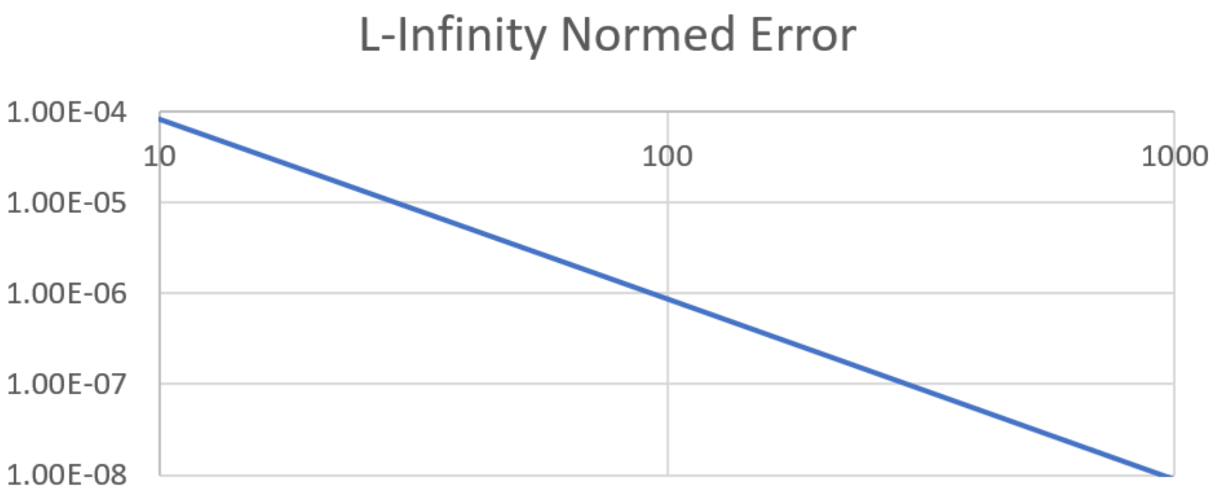

Figure 2: $L-\infty$ errors are given for modest resolutions.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Solves the Singular Volterra Integral Equation:
% $x(s) = (1 + s)^{-1/2} + \frac{\pi}{8} - \frac{1}{4}\arcsin \Big( \frac{1 - s}{1 + ...
    s} \Big)
%  - \frac{1}{4} \int_{0}^{s} \frac{x(t)}{(s - t)^{1/2}} dt$ on $0 < s < 1$
%   via Trapezoidal product integration
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Parameters
N_vec = [90:1:100];

for N = N_vec
    % Allocation
    s = linspace(0, 1, N + 1)';
    b = 1 ./ sqrt(1 + s) + pi / 8 - asin((1 - s) ./ (1 + s)) / 4;
    W = zeros(N + 1, N + 1);
    xe = 1 ./ sqrt(s + 1);

    % We let the first row have zero weights because this is associated with
    % s = 0. The contribution of the integral term is zero here so the weight
    % shall be as well.
    for row = 2:N + 1 %i
        for col = 1:row %j
            if col == 1
                W(row, col) = (row - 2)^(3/2) - sqrt(row - 1) * (2 * (row - 1) - 3)/2;
            elseif col == row
                W(row, col) = 1;
            else
                W(row, col) = (row - col + 1)^(3/2) - 2 * (row - col)^(3/2) + (row - ...
                    col - 1)^(3/2);
            end
        end
    end

    x = linsolve(eye(N + 1) + W / sqrt(N)/3,  b);
%     disp(strcat("N = ", num2str(N), ": error = ", num2str(norm(x - xe, inf))))
    disp(num2str(norm(x - xe, inf)))

end
```