[1]

CS 714 HOMEWORK 1

Zihao Zheng
Department of Statistics
University of Wisconsin-Madison
Madison, WI, 53705
zihao.zheng@wisc.edu

October 5, 2020

1 Problem A

1.1 (a)

This is equivalent to show,

$$\mu^2 u(x) = (1 + \frac{1}{4}\delta^2)u(x) \tag{1}$$

The left hand side of equation 1 is,

$$\mu^2 u(x) = \frac{1}{2}(\frac{u(x+h) + u(x)}{2} + \frac{u(x) + u(x-h)}{2}) = \frac{u(x+h) + 2u(x) + u(x-h)}{4} \tag{2}$$

and the right hand side of equation 1 is,

$$(1 + \frac{1}{4}\delta^2)u(x) = u(x) + \frac{1}{4}(u(x+h) - u(x) - u(x) + u(x-h)) = \frac{u(x+h) + 2u(x) + u(x-h)}{4} \tag{3}$$

1.2 (b)

This does not provide proper Finite Difference (FD) schemes since it is not on the grid. In other words, for any positive ODD number k, the operator $\delta^k u(x)$ requires $\frac{h}{2}$ grid rather than h.

1.3 (c)

Starting with the following equation,

$$1 = \mu(1 + \frac{1}{4}\delta^2)^{-1/2} \tag{4}$$

$$hD = \delta - \frac{\delta^3}{24} + \frac{3\delta^5}{640} + ... \tag{5}$$

Multiply these two equations together and get the following formulation,

[1] Codes can be found at https://github.com/ZihaoZheng-Stat/CS714

$$hD = \mu(1 + \frac{1}{4}\delta^2)^{-1/2}(\delta - \frac{\delta^3}{24} + \frac{3\delta^5}{640} + ...) \tag{6}$$

$$= \mu(1 - \frac{1}{8}\delta^2 + \frac{3}{8}(\frac{\delta^2}{4})^2 + ...)(\delta - \frac{\delta^3}{24} + \frac{3\delta^5}{640} + ...) \tag{7}$$

$$= \mu(\delta - \frac{1}{6}\delta^3 + ...) \tag{8}$$

$$= \mu\delta(1 - \frac{1}{6}\delta^2 + ...) \tag{9}$$

By simply checking $\mu\delta$ and δ^2 in equation 9 are both on grid, we can get a proper FD scheme for hD. Explicitly,

$$\mu\delta u(x) = \frac{u(x+h) + u(x)}{2} - \frac{u(x) + u(x-h)}{2} = \frac{u(x+h) - u(x-h)}{2} \tag{10}$$

$$\delta^2 u(x) = h^2 D_{c,2} u(x) \tag{11}$$

2 Problem B

Denote for a fixed integer m, $\tau^h(-\epsilon + mh)$ be the error with the following definition,

$$\tau^h(-\epsilon + mh) = D_c u(-\epsilon + mh) - \frac{d}{dx} x_+^n \tag{12}$$

This definition Equation 12 had different expression using different m.

- When $m \leq -1$, $D_c u(-\epsilon + mh) = 0$ and $\frac{d}{dx} x_+^n = 0$ therefore $\tau^h(-\epsilon + mh) = 0$.
- When $m = 0$,
$$D_c u(-\epsilon) = \frac{u(-\epsilon + h) - u(-\epsilon - h)}{2h} = \frac{(-\epsilon + h)^n}{2h} \tag{13}$$
Meanwhile, $\frac{d}{dx} x_+^n = 0$ therefore $\tau^h(-\epsilon + mh) = \frac{(-\epsilon+h)^n}{2h}$.
- When $m = 1$,
$$D_c u(-\epsilon + h) = \frac{u(-\epsilon + 2h) - u(-\epsilon)}{2h} = \frac{(-\epsilon + 2h)^n}{2h} \tag{14}$$
Meanwhile,
$$\frac{d}{dx} x_+^n = \frac{d}{dx} x^n | \{x = -\epsilon + h\} \tag{15}$$
- When $m \geq 2$,
$$D_c u(-\epsilon) = \frac{(-\epsilon + (m+1)h)^n - (-\epsilon + (m-1)h)^n}{2h} \tag{16}$$
Meanwhile,
$$\frac{d}{dx} x_+^n = \frac{d}{dx} x^n | \{x = -\epsilon + mh\} \tag{17}$$

Also, we need to discuss for various option of n. This would provide different performance especially when $m \geq 0$. Before that, note that Taylor expansion could be applied when $m \geq 2$ as the following because when $m \geq 2$, $x_0(-\epsilon + mh)$, $x_0 - h(-\epsilon + (m-1)h)$ and $x_0 + h(-\epsilon + (m+1)h)$ all larger than 0. Denote $x_0 = -\epsilon + mh$, $f(x) = x^n$

$$f(x_0 + h) = f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + O(h^3) \tag{18}$$

$$f(x_0 - h) = f(x_0) - hf'(x_0) + \frac{h^2}{2} f''(x_0) + O(h^3) \tag{19}$$

1. When $n = 0$:

$$\tau^h(-\epsilon + mh) = \begin{cases} 0, & m \leq -1 \\ \frac{1}{2h}, & m = 0 \\ \frac{1}{2h}, & m = 1 \\ 0, & m \geq 2 \end{cases} \tag{20}$$

2. when $n = 1$:

$$\tau^h(-\epsilon + mh) = \begin{cases} 0, & m \leq -1 \\ O(1), & m = 0 \\ O(1), & m = 1 \\ 0, & m \geq 2 \end{cases} \tag{21}$$

3. when $n = 2$:

$$\tau^h(-\epsilon + mh) = \begin{cases} 0, & m \leq -1 \\ O(h), & m = 0 \\ O(h), & m = 1 \\ 0, & m \geq 2 \end{cases} \tag{22}$$

4. when $n \geq 3$:

$$\tau^h(-\epsilon + mh) = \begin{cases} 0, & m \leq -1 \\ O(h^{n-1}), & m = 0 \\ O(h^{n-1}), & m = 1 \\ O(h^2), & m \geq 2 \end{cases} \tag{23}$$

2.1 (a)

By previous argument, it is consistent when $n \geq 2$ and furthermore, the order is $O(h)$ when $n = 2$ and $O(h^2)$ when $n \geq 3$.

2.2 (b)

This could be simply addressed following the previous argument with the definition of the l_1 norm. When $n = 0$, it is still not consistent (of order $O(1)$ but not goes to 0 when $h \to 0$). When $n = 1$, it is consistent with order $O(h)$. When $n = 2$, it is consistent with order $O(h^2)$. When $n \geq 3$, it is still consistent with order $O(h^2)$ since the number of j in the summation $h \sum_j ||E_j||$ is of order $O(1/h)$.

2.3 (c)

When $n = 3$, consider also $x_0 = -\epsilon + mh$. By Taylor expansion,

$$f(x_0 + h) = f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + O(h^3) \tag{24}$$

$$f(x_0 - h) = f(x_0) - hf'(x_0) + \frac{h^2}{2} f''(x_0) + O(h^3) \tag{25}$$

Therefore the Taylor expansion would yield $O(h^2)$ consistency.

At the same time, by definition of cubic function,

- When $m \leq -1$, the derivative and the finite approximation are both 0.
- When $m = 0$, the true derivative is 0 and the finite approximation, as well as the error is $\frac{(-\epsilon+h)^3}{2h} = O(h^2)$.
- When $m = 1$, the true derivative is $3(-\epsilon + h)^2$ and the finite approximation is $\frac{(-\epsilon+2h)^3}{2h}$. Therefore the error is $|h^2 - \frac{\epsilon^3}{2h}|$ and it is also of order $O(h^2)$.
- When $m \geq 2$, denote $x_0 = -\epsilon + mh$ and therefore the true derivative is $3x_0^2$ and the finite approximation is $3x_0^2 + h^2$ therefore the error is h^2.

Therefore the maximum norm of error is $O(h^2)$, which is same as the Taylor expansion.

3 Problem C

3.1 (a)

Consider the finite difference approximation on the grid $[0,1] * [0,1]$. Denote $M+1$ and $N+1$ be the number of grid on x and y. Then, denote $u_{i,j}$ be $u(x_i, y_j)$ where

$$
\begin{aligned}
h_x &= \frac{1}{M+1} \\
h_y &= \frac{1}{N+1} \\
x_i &= ih_x, i = 0, 1, ..., M+1 \\
y_j &= jh_y, j = 0, 1, ..., N+1
\end{aligned}
$$

Therefore each cell is a triangle of size $h_x * h_y$.

Consider the finite difference on x level and the Dirichlet boundary condition. For any j,

- $i = 1$, the second derivative over x
$$u_{x,x}(x_1, y_j) = \frac{1}{h_x^2}(u_{0,j} - 2u_{1,j} + u_{2,j}) = \frac{1}{h_x^2} f_j + \frac{1}{h_x^2}(-2u_{1,j} + u_{2,j}) \tag{26}$$
where $f_j = f(y_j) = \cos 2\pi y_j$
- $i = 2, ..., M-1$, the the second derivative over x
$$u_{x,x}(x_i, y_j) = \frac{1}{h_x^2}(u_{i-1,j} - 2u_{i,j} + u_{i+1,j}) \tag{27}$$
- $i = M$, the second derivative over x
$$u_{x,x}(x_M, y_j) = \frac{1}{h_x^2}(u_{M-1,j} - 2u_{M,j} + u_{M+1,j}) = \frac{1}{h_x^2}(u_{M-1,j} - 2u_{M,j}) \tag{28}$$

Similarly, consider the finite difference on y level and the Neumann boundary condition. For any i,

- $j = 1$, the second derivative over y
$$u_{y,y}(x_i, y_1) = \frac{1}{h_y^2}(u_{i,0} - 2u_{i,1} + u_{i,2}) = \frac{1}{h_y^2}(-u_{i,1} + u_{i,2}) \tag{29}$$
- $j = 2, ..., N-1$, the second derivative over y
$$u_{y,y}(x_i, y_j) = \frac{1}{h_y^2}(u_{i,j-1} - 2u_{i,j} + u_{i,j+1}) \tag{30}$$
- $j = N$, the second derivative over y
$$u_{y,y}(x_i, y_N) = \frac{1}{h_y^2}(u_{i,N-1} - 2u_{i,N} + u_{i,N+1}) = \frac{1}{h_y^2}(u_{i,N-1} - u_{i,N}) \tag{31}$$

Therefore, we can conclude the following,

$$-\Delta = \frac{1}{h_x^2} A \otimes I_y + \frac{1}{h_y^2} I_x \otimes B \tag{32}$$

where A, I_x are $M * M$ matrices and B, I_y are $N * N$ matrices.

$$
A = \begin{pmatrix}
-2 & 1 & 0 & 0 & \cdots & 0 \\
1 & -2 & 1 & 0 & \cdots & 0 \\
0 & 1 & -2 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & 1 & -2
\end{pmatrix}
B = \begin{pmatrix}
-1 & 1 & 0 & 0 & \cdots & 0 \\
1 & -2 & 1 & 0 & \cdots & 0 \\
0 & 1 & -2 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & 0 & 1 & -1
\end{pmatrix}
$$

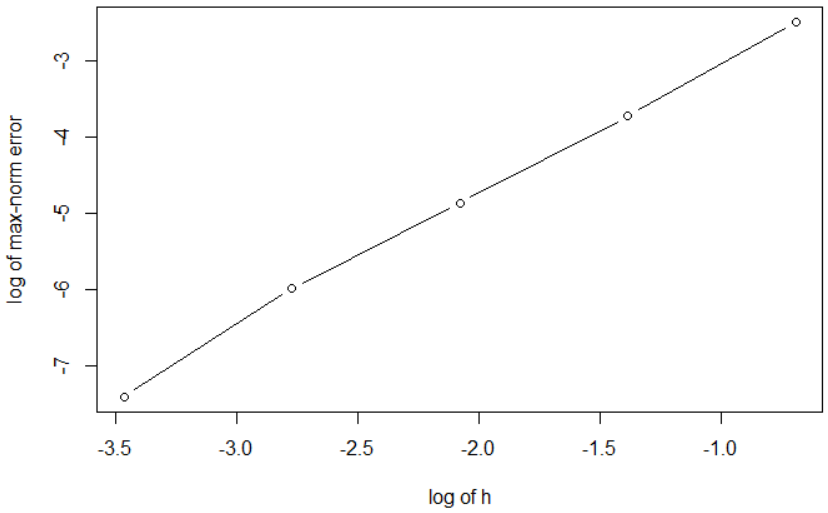

Figure 1: Figure of (b) in Problem C, between $\log$ error and $\log h$.

Furthermore, if denote U be a column vector of length MN, where for $i = 1, 2, ..., M$, its $(i-1)N+1$-th to iN-th entries are $[u_{i,1}, u_{i,2}, ..., u_{i,N}]'$. To be more clear, its first N entries are $[u_{1,1}, u_{1,2}, ..., u_{1,N}]'$. Also, if denote F be a column vector of length MN where its first N entries are $-\frac{1}{h_x^2}[f_1, ..., f_N]'$ and the rest all zero's.

By the previous all together, we have the linear system as the following,

$$QU = F \tag{33}$$

$$Q = \frac{1}{h_x^2} A \otimes I_y + \frac{1}{h_y^2} I_x \otimes B \tag{34}$$

3.2 (b)

We can think of the problem as a linear system in general as $QU = c$, where Q, c can be written explicitly with the information in (a), and U is a vector of length $(M+1)(N+1)$ as defined also in (a).

For simplicity, suppose we consider the case where $h_x = h_y = h$ and then $\frac{1}{M+1} = \frac{1}{N+1}$ and each cell is a square rather than a rectangle. In order to evaluate the error with maximum norm, we consider the situation where $\frac{1}{M+1} = \frac{1}{N+1} = \frac{1}{2}, \frac{1}{4}, \cdots$ and use the finest one as the "correct answer".

To solve the linear system, we use Jacobi method, where we decompose Q into $Q = Q_1 + Q_2$, where Q_1 is a diagonal matrix and the iteration step is:

$$x^{k+1} = Q_1^{-1}(c - Q_2 x^k) \tag{35}$$

Note that this is matrix form of iteration. When implementation, we do not need to store all the matrix and calculate every entries since the majority of them are zeros.

The following figure shows the figure between $\log$ error and $\log h$ and it is clear that it is a straight line with slope approximately 2.

3.3 (c)

From that in (b), the finite grid approximation of two-dimensional Poisson equation is consistent of order $O(h^2)$.

3.4 (d)

Denote the eigendecomposition of A and B as $A = X\Lambda_x X^{-1}$ and $B = Y\Lambda_y Y^{-1}$. Therefore,

$$\frac{1}{h_x^2} A \otimes I_y = X \frac{1}{h_x^2} \Lambda_x X^{-1} \otimes I_y \tag{36}$$

$$= (X \otimes Y)(\frac{1}{h_x^2} \Lambda_x X^{-1} \otimes I_y y^{-1}) \tag{37}$$

$$= (X \otimes Y)(\frac{1}{h_x^2} \Lambda_x \otimes I_y)(X \otimes Y)^{-1} \tag{38}$$

Similarly,

$$\frac{1}{h_y^2} I_x \otimes B = (X \otimes Y)(\frac{1}{h_y^2} I_x \otimes \Lambda_y)(X \otimes Y)^{-1} \tag{39}$$

Therefore, the eigendecomposition of $Q = \frac{1}{h_x^2} A \otimes I_y + \frac{1}{h_y^2} I_x \otimes B$ is

$$Q = \frac{1}{h_x^2} A \otimes I_y + \frac{1}{h_y^2} I_x \otimes B = (X \otimes Y)(\frac{1}{h_x^2} \Lambda_x \otimes I_y + \frac{1}{h_y^2} I_x \otimes \Lambda_y)(X \otimes Y)^{-1} \tag{40}$$

3.5 (e)

We known from class, the eigenvalues of A (might be slightly different from that in lecture, but only a denominator as $\frac{1}{h_x^2}$) are $2(cos(i\pi h) - 1)$, which is bounded by its smallest eigenvalue, $2h^2\pi^2 + O(h^4)$. Also from the information of this question, we known that the eigenvalues of matrix B are all real and positive.

Then from equation 40, all eigenvalues of $-\Delta$ has the form for any (i, j) pair where $i \in \{1, 2, \cdots, M+1\}$ and $j \in \{1, 2, \cdots, N+1\}$,

$$\frac{\lambda_{x,i}}{h_x^2} + \frac{\lambda_{y,j}}{h_y^2} \tag{41}$$

where $\lambda_{x,i}$ is the i-th eigenvalue of matrix A and $\lambda_{y,j}$ is the j-th eigenvalue of matrix B. Combining all information together, the minimum of absolute value among all of those $(M+1)(N+1)$ eigenvalues is bounded away from 0.

3.6 (f)

In order to check the convergence rate of the iteration method, we pick two examples which in one case, $h_x = h_y = \frac{1}{64}$ and in other case $h_x = h_y = \frac{1}{16}$. For each case, I firstly solve the linear system using non-iterative step and regard this as the correct answer $U_{\{\text{correct}\}}$. Suppose our initial guess is $U_0 = (0, 0, \cdots, 0)$ and the initial guess and denote the initial error as:

$$e_0 = ||U_0 - U_{\{\text{correct}\}}||^2 \tag{42}$$

At each iteration step i, I calculate the error e_i and plot with the iteration step. Also, I denote the ratio between the error at each step and initial error $\epsilon_i = \frac{e_i}{e_0}$ as a metric demonstrating how fast the convergence behaves. Left panel shows the relationship between e_i and i where the right panel shows the relationship between the ϵ_i and i. Note that, when plotting the relationship between ϵ_i and i, the log-transformation on the y-axis is pre-processed. The red line takes an example when we would like to threshold ϵ by 10^{-3} and indicates how many iteration step is required to achieve this bound. Combining all together, the number of iteration k to achieve the bound is as order $\frac{\log \epsilon}{\log \rho}$ and this is consistent with the theory.(Note that $\log \rho = O(h^2)$ of matrix A).

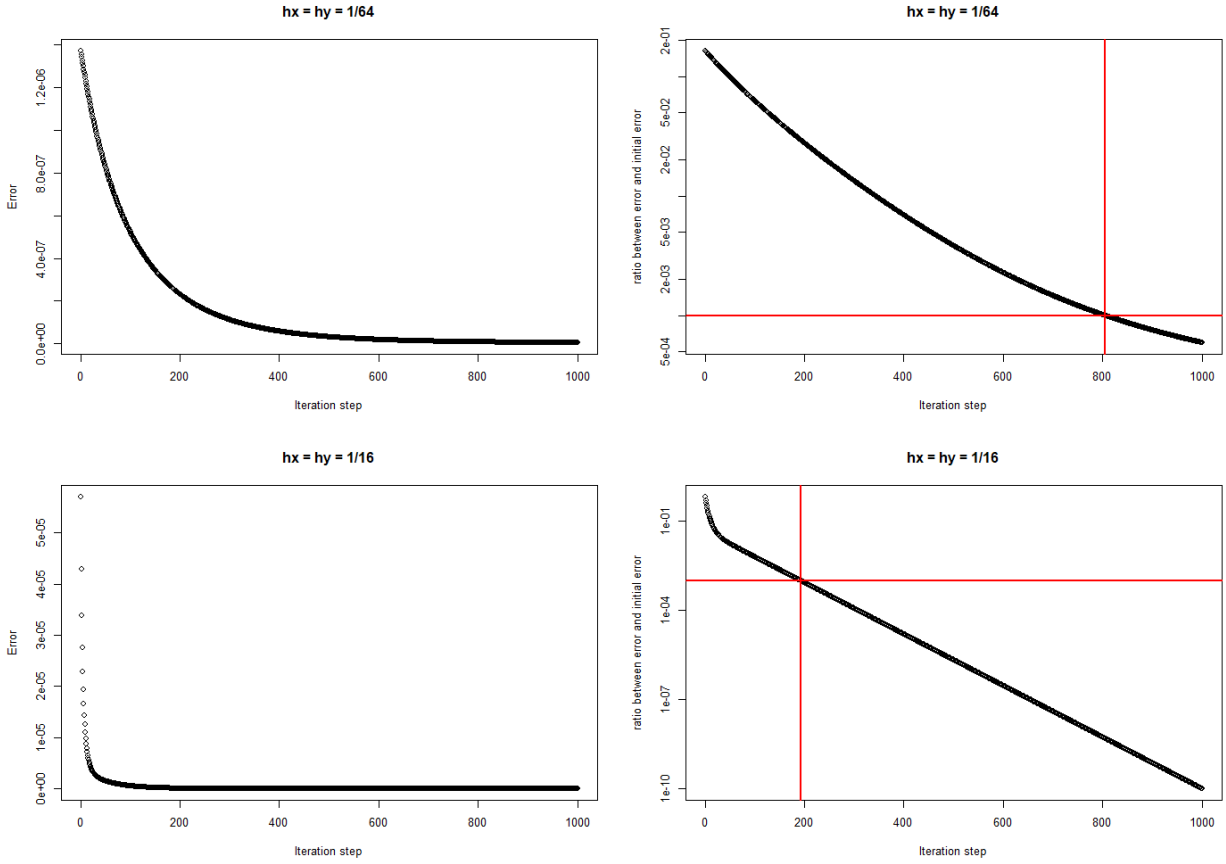

Figure 2: Figure of (f) in Problem C.

3.7 (g)

Using the similar technique as in problem (b), we can plot the log-log figure as the following. Note that we also do not explicitly form the exact answer but work on nested grid and treat the answer with the finest grid as the "exact answer". Numerically, the log-log figure still looks like a linear form and the slope is slightly smaller than 2, which means the order of convergence might be smaller than $O(h^2)$. The interpretation of this might come from the continuity of function $\hat{f}$. When evaluated between $[0, 1]$, the function f is smooth however the function $\hat{f}$ has two breakpoints.

4 Problem D

Since this function is a two-dimensional function, it is hard to visualize it on a $2-d$ plot. However, we can report the error at a certain level of iteration times, comparing the traditional iteration method and multi grid method.

Here we report some basic statistics comparing the traditional iteration methods and multi grid methods.

We start with the grid level at $h = \frac{1}{64}$. Using the same definition, the initial error $e_0 = 8.39 * 10^{-6}$. Take 3 iteration steps, the error drops to $e_3 = 6.85 * 10^{-6}$, with $\epsilon_3 = 0.81$. After a thousand times of iterations, the error drops to $e_{1000} = 6.33 * 10^{-9}$ with $\epsilon_{1000} = 7.54 * 10^{-4}$.

Using multi grid method, the convergence rate is much faster. Following the steps in the book, with the finer grid $h = \frac{1}{64}$ and coarser grid $h = \frac{1}{32}$, we can drop the error (at only a single round of six steps mentioned in the book) to $5.67 * 10^{-9}$ and $\epsilon = 6.76 * 10^{-4}$. This is even slightly smaller than 1000 iterations using the traditional approach. Of course we can iteratively reduce the grid into $h = \frac{1}{16}, \frac{1}{8}, \cdots$ and it would converge at a much more desired rate. The above illustration is evident enough that the multi grid approach is better with respect to convergence property.

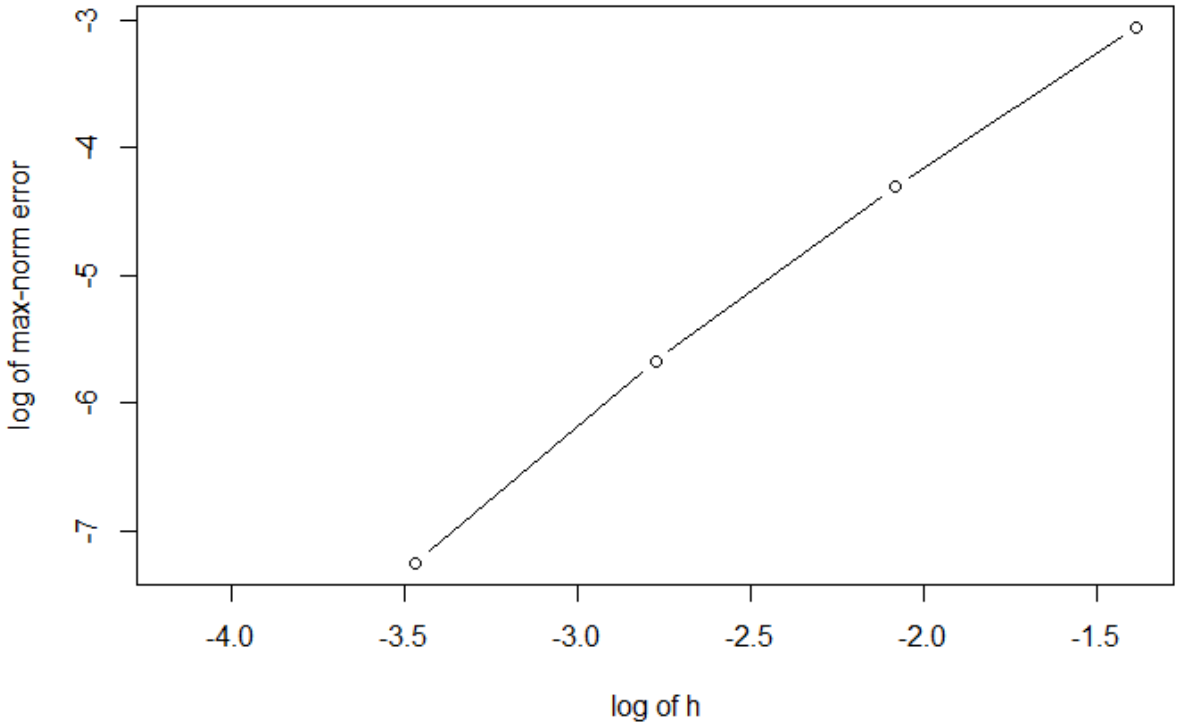

Figure 3: Figure of (g) in Problem C, between $\log$ error and $\log h$.

5 Problem E

Here we reproduce the analysis in problem C(f) but replacing the function f into $\hat{f}$. The following summarizes the similar figure.

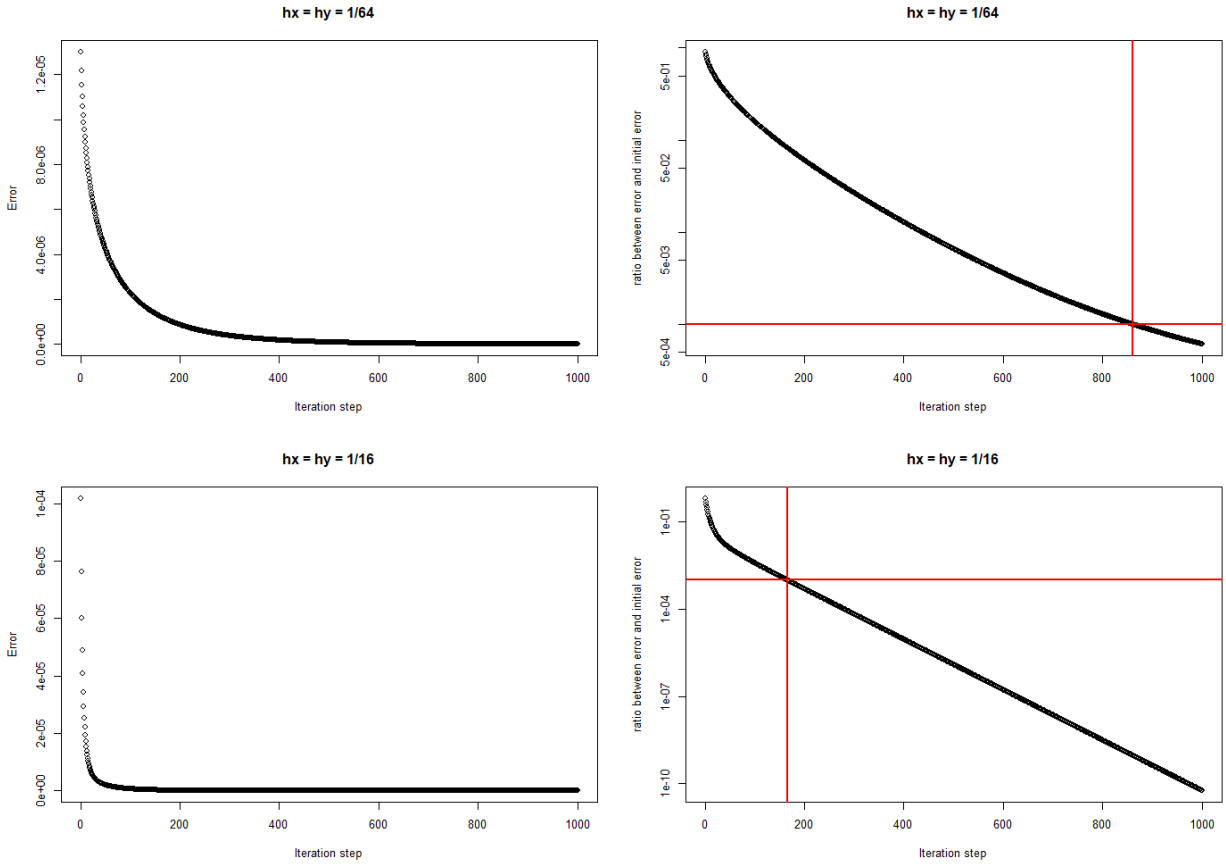

Figure 4: Figure of problem E.