Introduction to Numerical Optimization
Assignment 1

March 29, 2022

1 Analytical and Numerical Differentiation

1.1 Analytical Differentiation

1.1.1 Gradient and Hessian of f_1

$$f_1(x) = \phi(Ax)$$

Where:

$$f_1 : \mathbb{R}^n \to \mathbb{R}$$

$$x \in \mathbb{R}^n,$$

$$A \in \mathbb{R}^{m \times n},$$

$$\phi : \mathbb{R}^m \to \mathbb{R}$$

Assuming $\nabla\phi$ and $\nabla^2\phi$ are known (relative to its $\mathbb{R}^m$ variable).

$$\nabla f_1 = ?$$

$$\nabla^2 f_1 = ?$$

We denote $u = Ax \Rightarrow du = Adx$, by the external definition of the gradient:

$$df_1 = d\phi(u) = (\nabla\phi(u))^T du = (\nabla\phi(Ax))^T Adx =$$

$$\langle A^T \nabla\phi(Ax), dx\rangle \Rightarrow$$

$$\boxed{\nabla f_1 = A^T \nabla\phi(Ax)}$$

Next, We denote $\nabla f_1 = g(x)$:

$$dg \equiv Hdx, d(\nabla\phi(u)) = \nabla^2\phi(u)du \Rightarrow$$

$$dg = d(\nabla f_1) = d(A^T\nabla\phi(Ax)) = A^T d(\nabla\phi(u)) = A^T\nabla^2\phi(u)du = A^T\nabla^2\phi(Ax)Adx \Rightarrow$$

$$\boxed{\nabla^2 f_1 = A^T\nabla^2\phi(Ax)A}$$

1.1.2 Gradient and Hessian of f_2

$$f_2(x) = h(\phi(x))$$

Where:

$$f_2 : \mathbb{R}^n \to \mathbb{R}$$

$$x \in \mathbb{R}^n,$$

$$\phi : \mathbb{R}^n \to \mathbb{R}$$

$$h : \mathbb{R} \to \mathbb{R},$$

Assuming $\nabla\phi, \nabla^2\phi, h'(t)$ and $h''(t)$ are known.

$$\nabla f_2 =?$$

$$\nabla^2 f_2 =?$$

We denote $t = \phi(x) \Rightarrow dt = d\phi(x) = (\nabla\phi(x))^T dx$

$$df_2 = dh = h'(t)dt = h'(\phi(x))(\nabla\phi(x))^T dx =$$

Since $h'(t)$ is scalar, it doesn't change on transpose operation:

$$\langle h'(\phi(x))\nabla\phi(x), dx\rangle \Rightarrow$$

$$\boxed{\nabla f_2 = h'(\phi(x))\nabla\phi(x)}$$

Next, We denote $\nabla f_2 \equiv g(x)$:

$$dg = d(h'(\phi(x))\nabla\phi(x)) = d(h'(\phi(x)))\nabla\phi(x) + h'(\phi(x))d(\nabla\phi(x)) =$$

$$h''(\phi(x))((\nabla\phi(x))^T dx)\nabla\phi(x) + h'(\phi(x))\nabla^2\phi(x)dx =$$

$$h''(\phi(x))\nabla\phi(x)((\nabla\phi(x))^T dx) + h'(\phi(x))\nabla^2\phi(x)dx =$$

$$h''(\phi(x))\nabla\phi(x)(\nabla\phi(x))^T dx + h'(\phi(x))\nabla^2\phi(x)dx =$$

$$[h''(\phi(x))\nabla\phi(x)(\nabla\phi(x))^T + h'(\phi(x))\nabla^2\phi(x)]dx =$$

by definition $dg = Hdx$, therefore:

$$\boxed{\nabla^2 f_2 = h''(\phi(x))\nabla\phi(x)(\nabla\phi(x))^T + h'(\phi(x))\nabla^2\phi(x)}$$

1.1.3 Gradient and Hessian of ϕ

Given the following $\phi : \mathbb{R}^3 \to \mathbb{R}$ function:

$$\phi \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = cos(x_1 x_2^2 x_3)$$

We need to calculate its gradient vector and hessian matrix. We can use the previous section results when there is a chain of functions to compute the gradient and hessian as we can define:

$$f = cos : \mathbb{R} \to \mathbb{R}$$

and

$$u = x_1 x_2^2 x_3 : \mathbb{R}^3 \to \mathbb{R}$$

Thus, the gradient is as follows:

$$\nabla \phi = f'(u) \nabla u(x) = -sin(x_1 x_2^2 x_3) \begin{pmatrix} x_2^2 x_3 \\ 2x_1 x_2 x_3 \\ x_1 x_2^2 \end{pmatrix}$$

$$\boxed{\nabla \phi = -sin(x_1 x_2^2 x_3) \begin{pmatrix} x_2^2 x_3 \\ 2x_1 x_2 x_3 \\ x_1 x_2^2 \end{pmatrix}}$$

We can also use the hessian formula we got on the previous section and get:

$$\nabla^2 \phi = f''(u(x)) \nabla u(x) (\nabla u(x))^T + f'(u(x)) \nabla^2 u(x) =$$

$$-cos(x_1 x_2^2 x_3) \begin{pmatrix} x_2^2 x_3 \\ 2x_1 x_2 x_3 \\ x_1 x_2^2 \end{pmatrix} \begin{pmatrix} x_2^2 x_3 & 2x_1 x_2 x_3 & x_1 x_2^2 \end{pmatrix} - sin(x_1 x_2^2 x_3) \begin{pmatrix} 0 & 2x_2 x_3 & x_2^2 \\ 2x_2 x_3 & 2x_1 x_3 & 2x_1 x_2 \\ x_2^2 & 2x_1 x_2 & 0 \end{pmatrix} \Rightarrow$$

$$\boxed{\nabla^2 \phi = -cos(x_1 x_2^2 x_3) \begin{pmatrix} x_2^4 x_3^2 & 2x_1 x_2^3 x_3^2 & x_1 x_2^4 x_3 \\ 2x_1 x_2^3 x_3^2 & 4x_1^2 x_2^2 x_3^2 & 2x_1^2 x_2^3 x_3 \\ x_1 x_2^4 x_3 & 2x_1^2 x_2^3 x_3 & x_1^2 x_2^4 \end{pmatrix} - sin(x_1 x_2^2 x_3) \begin{pmatrix} 0 & 2x_2 x_3 & x_2^2 \\ 2x_2 x_3 & 2x_1 x_3 & 2x_1 x_2 \\ x_2^2 & 2x_1 x_2 & 0 \end{pmatrix}}$$

1.1.4 First and Second Derivatives of h

$$h = \sqrt{1 + sin^2(x)}$$

We can solve directly using derivative formulas and we can use the external gradient definition.
We can denote $v(x) = sin(x), u(v) = 1 + v^2$ and $h = f(u) = \sqrt{u} \Rightarrow$

$$h = f(u(v)) \Rightarrow$$

$$dv = d(sin(x)) = cos(x)dx$$

$$du = 2vdv$$

$$dh = df(u) = \frac{1}{2}\frac{1}{\sqrt{u}}du$$

$$\Rightarrow$$

$$dh = \frac{1}{2}\frac{1}{\sqrt{u}}du = \frac{1}{2}\frac{1}{\sqrt{1+v^2}}2vdv = \frac{sin(x)cos(x)}{\sqrt{1+sin^2(x)}}dx \Rightarrow$$

$$\boxed{h'(x) = \frac{sin(x)cos(x)}{\sqrt{1+sin^2(x)}}}$$

For the second derivative we can define the following functions and compute again:

$$v(x) = sin(x)$$

$$t(x) = cos(x)$$

$$u(v) = 1 + v^2$$

$$z(u) = \sqrt{u}$$

$$q(z) = \frac{1}{z}$$

$$h'(q, v, t) = vtq \Rightarrow$$

$$dv = cos(x)dx$$

$$dt = -sin(x)dx$$

$$du = 2vdv$$

$$dz = \frac{1}{2\sqrt{u}}du$$

$$dq = -\frac{1}{z^2}dz$$

Using the total differential property that we can take the differential with respect to each inner variable separately and sum $(df(u,v) = \nabla_u f(u,v)^T du + \nabla_v f(u,v)^T dv)$
And in our case:

$$dh'(v,t,q)) = d(vtq) = qtdv + qvdt + vtdq$$

We put the previous differentials and get:

$$dh' = qtdv + qvdt + vtdq = \frac{tdv + vdt}{z} - \frac{vt}{z^2}dz = \frac{tdv + vdt}{\sqrt{u}} - \frac{vt}{u}\frac{1}{2\sqrt{u}}du =$$

$$\frac{tdv + vdt}{\sqrt{1+v^2}} - \frac{vt}{2(1+v^2)\sqrt{1+v^2}}2vdv = \frac{cos^2(x) - sin^2(x)}{\sqrt{1+sin^2(x)}}dx - \frac{sin^2(x)cos^2(x)}{(1+sin^2(x))^{\frac{3}{2}}}dx$$

$$\Rightarrow$$

$$\boxed{h''(x) = \frac{cos^2(x)-sin^2(x)}{\sqrt{1+sin^2(x)}} - \frac{sin^2(x)cos^2(x)}{(1+sin^2(x))^{\frac{3}{2}}}}$$

1.1.5 Analytical Evaluation

in code

1.2 Numerical Differentiation

1.2.1 Numerical Gradient

Background information given in the assignment's pdf

1.2.2 Numerical Hessian

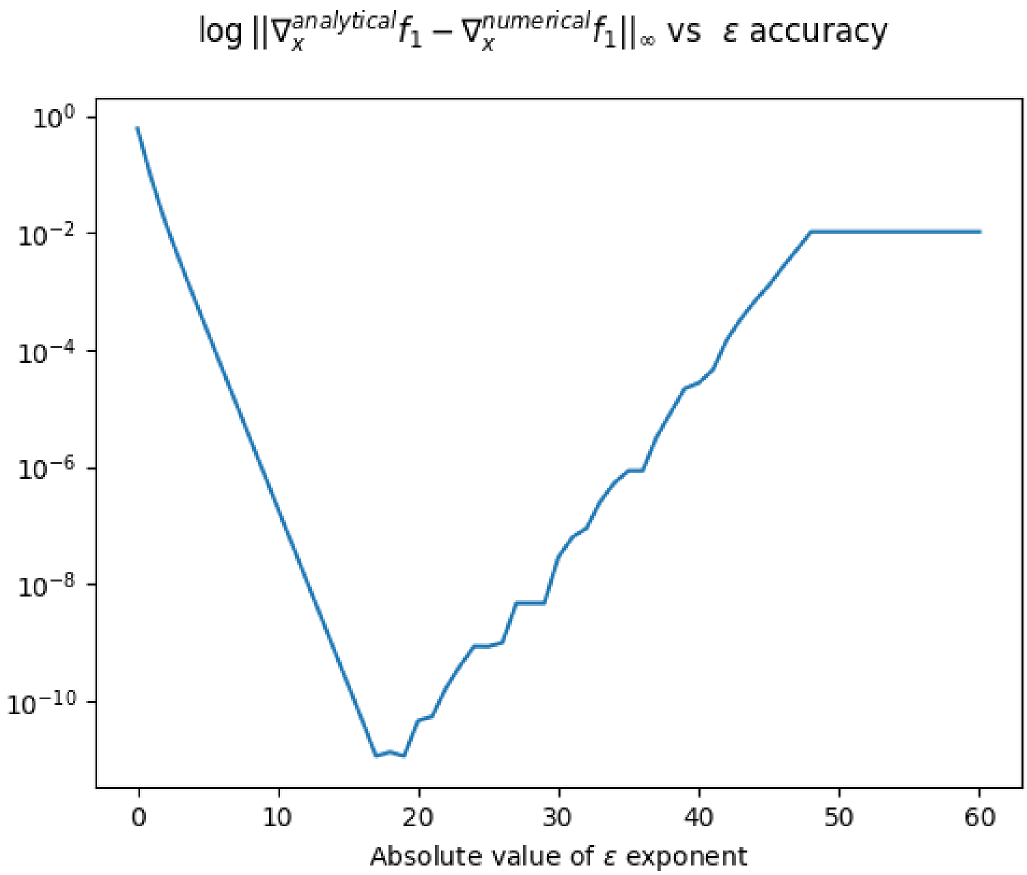
Background information given in the assignment's pdf

1.2.3 Numerical Evaluation of Gradient and Hessian

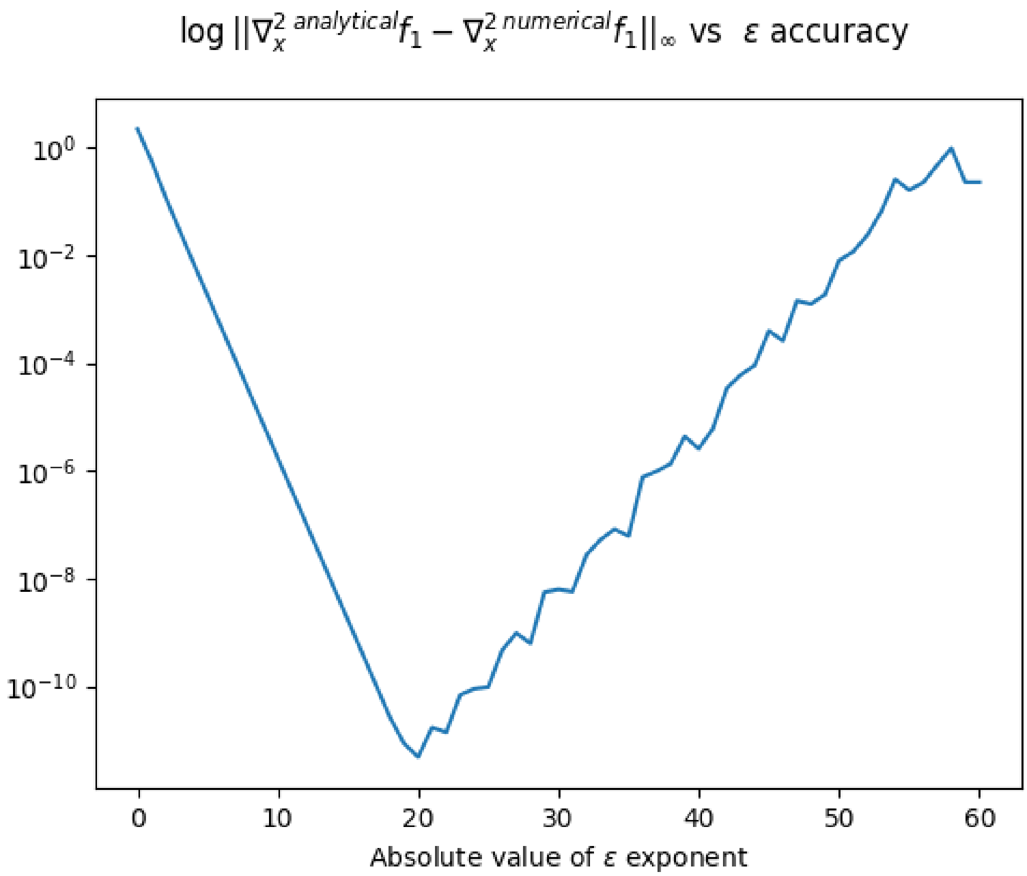
in code

1.3 Comparison

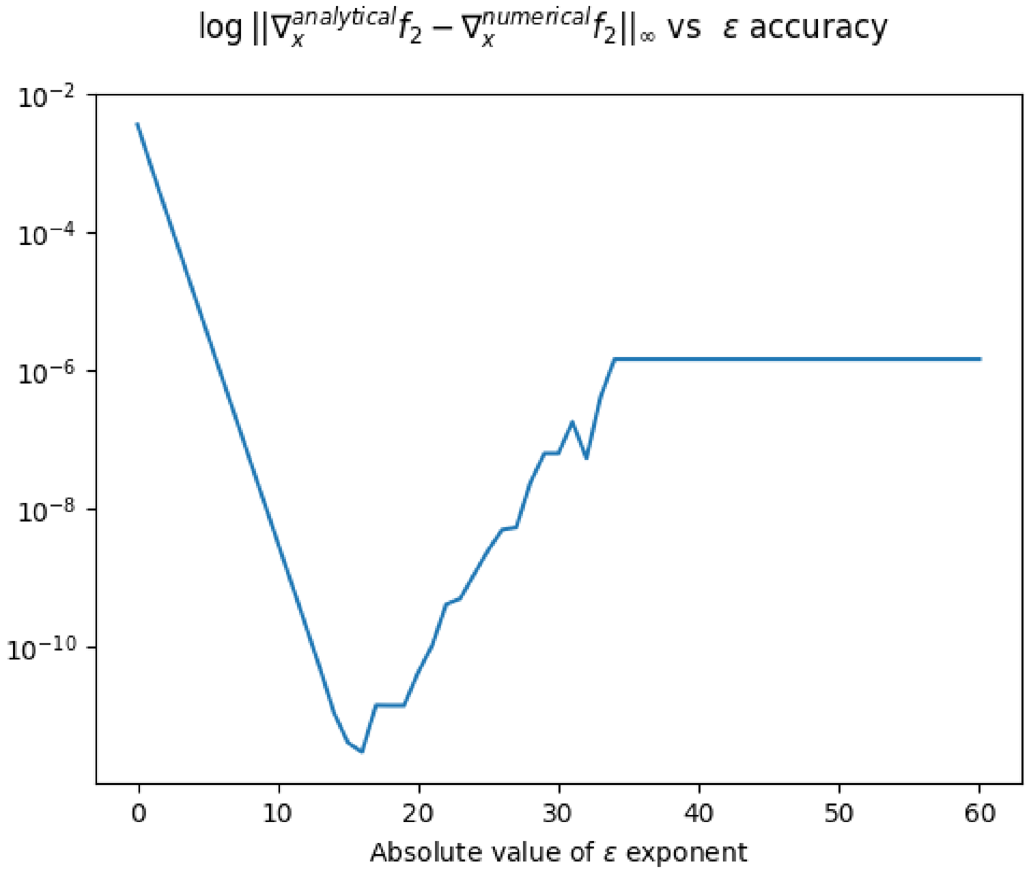
See hw1.py for the code that does the comparison. With random seed set to 10 we got the following results:



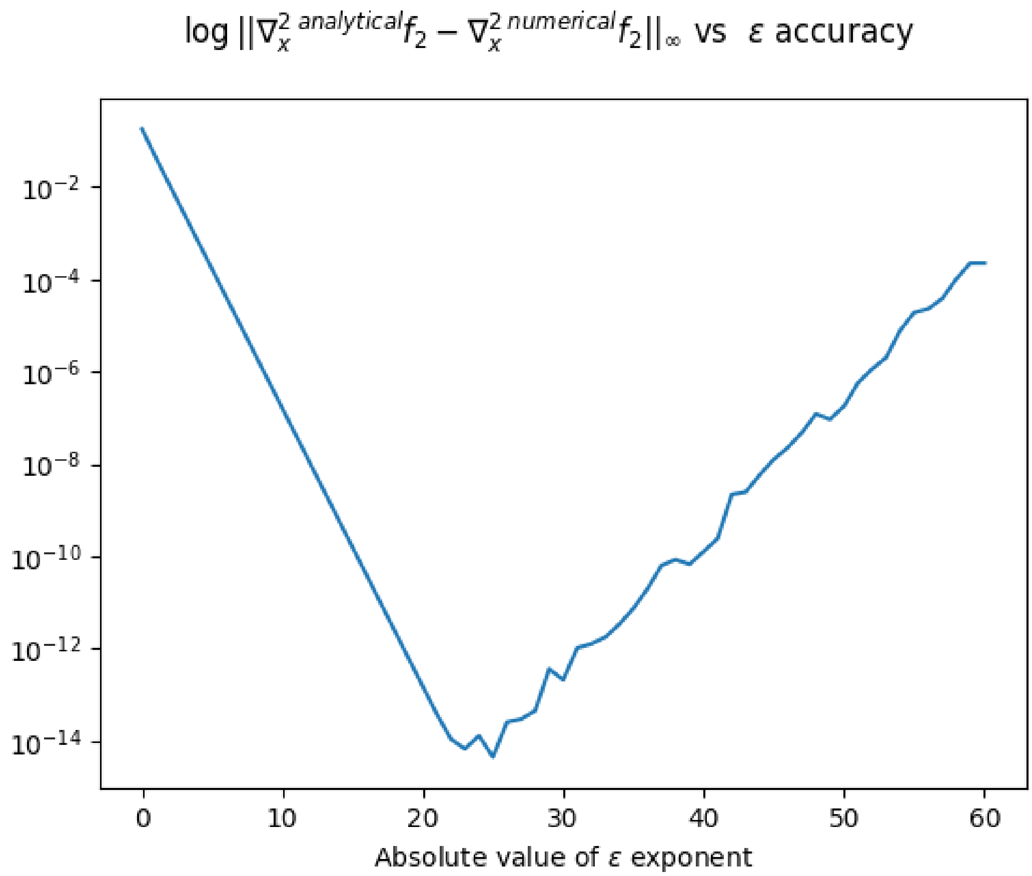
f1 gradient min infinity norm error : 1.1572293470707429e-11
epsilon exponent: -17
epsilon value: 7.62939453125e-06
We get the optimal ϵ for f1 gradient min infinity norm error at 2^{-17}
We can explain the graph as getting better when $\epsilon \to 0$
up to a certain value from which, numerical errors starts to happen due to our PC limited accuracy and numerical problems when dividing by a very small values. This causes inaccuracies compared to the analytical computation. At some point the diff in f seems like there is no diff and we get a flat line of a constant value (around 2^{-50})



f1 hessian min infinity norm error : 4.915796936177941e-12
epsilon exponent: -20
epsilon value: 9.5367431640625e-07
The same explanation as for the gradient norm graph. except that we don't see the flat line at the end. probably since the gradients values are small and thus adding ϵ still makes a difference (compared to function values that are higher and thus adding small ϵ is not observable due to max float range of our PC)



f2 gradient min infinity norm error : 3.077997186522305e-12
epsilon exponent: -16
epsilon value: 1.52587890625e-05
The same explanation as to f1 gradient norm.



f2 hessian min infinity norm error : 4.519438310732465e-15
epsilon exponent: -25
epsilon value: 2.9802322387695312e-08
The same explanation as to f1 hessian norm.