

Tarea 3: Optimización y análisis básico de funciones

1. Encontrar la dirección de máximo descenso de gradiente para la función $f(\mathbf{x}) = \sum_{i=1}^2 x_i^2$. Utilizando la derivada parcial direccional para los valores de \mathbf{x} . Mostrar procedimiento completo. (20/100)
 1. $\mathbf{x} = [1, 1]^T$ (10/100)
 2. $\mathbf{x} = [0, 0]^T$ (10/100)
2. Derivar el método de optimización de Newton a partir de las series de Taylor (A partir de la Ecuación 4.11 llegar a la expresión 4.12). (10/100)
3. Comparar la función $f(x)$ original con la aproximación (que no alguna vista en clase) utilizando serie de Taylor alrededor de de un punto x_0 . (Utilizar `matplotlib.pyplot` de `Python`). (10/100)
4. Dada las funciones: (60/100)
 - a) Ackley
 - b) Adjimon
 - c) Beale
 - d) Branin
 - 4.1 Calcular sus derivadas parciales. (10/100)
 - 4.2 Calcular sus Hessianos. (10/100)
 - 4.3 Minimizarlas con algoritmo de Descenso de Gradiente calculando ϵ utilizando la serie de Taylor. Mostrar gráfica de convergencia en un plot y sobre las curvas de nivel. (20/100)
 - 4.4 Minimizarlas con algoritmo de Newton. Mostrar gráfica de convergencia en un plot y sobre las curvas de nivel.(20/100)

Ejercicio Extra (no cuenta).

A partir de la serie de Taylor, describir los pasos necesarios para llegar a la solución del tasa de aprendizaje $\epsilon^* = \frac{\mathbf{g}^T \mathbf{g}}{\mathbf{g}^T \mathbf{H} \mathbf{g}}$ (Ecuación 4.10 del Libro de Deep Learning de Goodfellow).

Notas:

El ejercicio 1,2 puede Entregarse en escaneo de papel, markdown con ecuaciones en Latex, libreta de Python.

Ejercicio 3 Hacer y entregar en libreta de Python.

Ejercicio 4 Hacer y entregar en libreta de Python, los ejercicios de derivadas entregar en escaneo de notas en papel.