

# Tarea 1: Optimización ciega

Fernando Carrillo A01194204

24/08/20

## Descripción del problema

En esta tarea se busca encontrar el valor de 'x' para el cual su evaluación en la función  $f(x) = x^3 - 2x^2 + 1$  sea la menor posible. En otras palabras, queremos encontrar el mínimo de la función dentro del intervalo  $0 \leq x \leq 5$ .

Función objetivo:  $f(x) = x^3 - 2x^2 + 1$

Espacio de búsqueda: Valores de 'x' dentro del rango  $0 \leq x \leq 5$

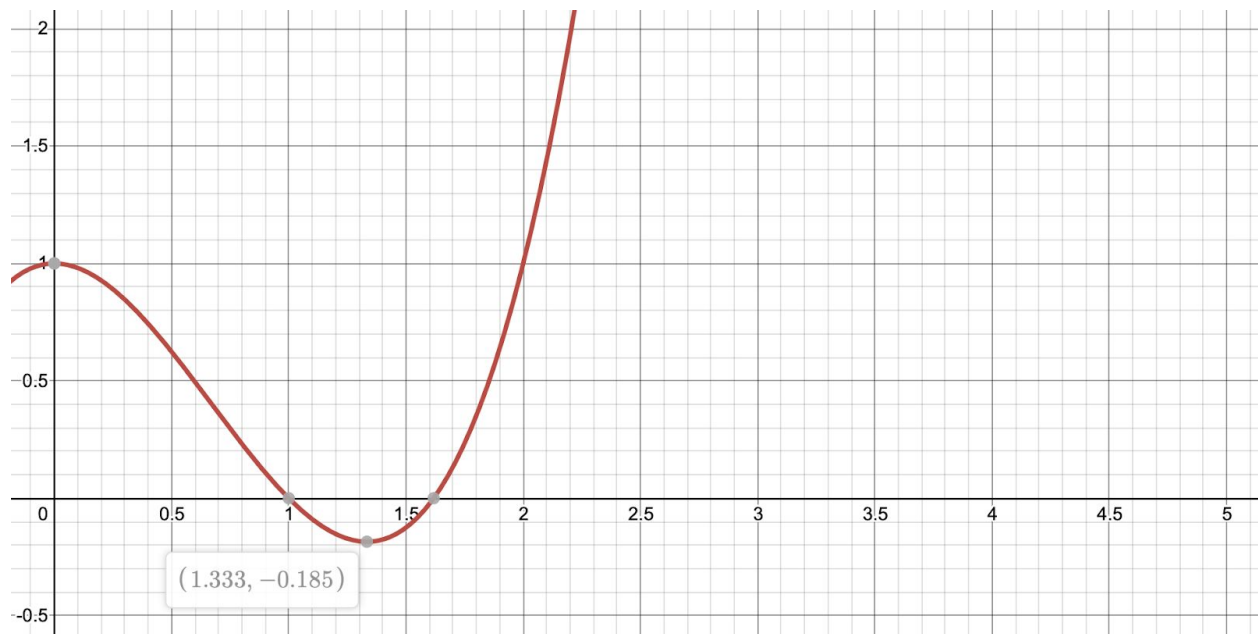
Intento: Evaluación de 'x', ej.  $x = 2.4$

## Descripción de la representación

Los valores están en términos de 'x' y 'y'. Se utilizan valores de 'x' dentro del rango  $0 \leq x \leq 5$  para buscar el menor valor de 'y' posible.

## Resultados obtenidos

Gráfica de la función  $f(x) = x^3 - 2x^2 + 1$



Viendo la gráfica podemos esperar que las respuestas estén cerca de los siguientes valores-

Valor de 'x' que da el menor valor al ser evaluado:

$$x = 1.3333$$

Valor mínimo de 'y' dentro del rango:

$$f(1.3333) = -0.185$$

### Resultados con

Iteraciones: 10

Num. Hijos: 10

Num. Padres 5

Con pocas iteraciones y pocos padres/hijos, los resultados eran algo acertados. Sin embargo hubieron veces en que los algoritmos de Varios padres, varios hijos no eran consistentes. Aquí podemos ver la diferencia entre dos ejecuciones del programa, y cómo el algoritmo de Varios padres, varios hijos con traslape generacional produce un resultado lejos del correcto. El algoritmo que sí fue consistente con valores de iteraciones pequeñas fue el de Un Padre, varios hijos.

(ejecución 1)

```
Un padre, varios hijos -
  Valor x: 1.3698      Evaluacion de x: -0.1825
Varios padres, varios hijos sin traslape -
  Valor x: 1.2482      Evaluacion de x: -0.1713
Varios padres, varios hijos con traslape -
  Valor x: 1.3202      Evaluacion de x: -0.1848
```

(ejecución 2)

```
Un padre, varios hijos -  
    Valor x:  1.341          Evaluacion de x:  -0.1851  
Varios padres, varios hijos sin traslape -  
    Valor x:  1.3326        Evaluacion de x:  -0.1852  
Varios padres, varios hijos con traslape -  
    Valor x:  0.7147        Evaluacion de x:  0.3435
```

### Resultados con

Iteraciones: 50  
Num. Hijos: 50  
Num. Padres 30

Cuando se elevaron los valores para iteraciones y padres/hijos, los resultados se acercaron más al valor correcto, y las respuestas no variaban mucho con cada ejecución del programa. En ambos casos, el más acertado fue el de Varios padres, varios hijos con traslape generacional.

(ejecución 1)

```
Un padre, varios hijos -  
    Valor x:  1.3264          Evaluacion de x:  -0.1851  
Varios padres, varios hijos sin traslape -  
    Valor x:  1.3253          Evaluacion de x:  -0.1851  
Varios padres, varios hijos con traslape -  
    Valor x:  1.3332          Evaluacion de x:  -0.1852
```

(ejecución 2)

```
Un padre, varios hijos -  
    Valor x:  1.3313          Evaluacion de x:  -0.1852  
Varios padres, varios hijos sin traslape -  
    Valor x:  1.3302          Evaluacion de x:  -0.1852  
Varios padres, varios hijos con traslape -  
    Valor x:  1.3336          Evaluacion de x:  -0.1852
```

## Conclusiones y retos encontrados

Esta actividad me recordó un poco a mi clase de métodos numéricos donde tuvimos que programar algunos algoritmos que nos ayudaban a encontrar aproximados de las raíces de alguna función. También me recordó un poco a la técnica de búsqueda binaria para encontrar el mínimo/máximo de una función, pero esta tiene la desventaja de que no tiene oportunidades de salir de mínimos/máximos locales.

Yo creo que el reto más grande para esta tarea fue volver a empezar a usar Python y recordar las funciones y trucos del lenguaje. Los algoritmos fueron algo sencillos

de implementar dado que se tenía al pseudocódigo en las presentaciones del curso, y el maestro lo explicó en clase.