

IPF Y MÉTODO DE NEWTON-RAPHSON

1. ¿Cuántas iteraciones del método de Newton-Raphson se requieren para encontrar la raíz de una función afín $f(x) = mx + n$?
¿Por qué tiene sentido el resultado anterior?

2. Considera la función $y = f(x) = xe^x$. Su inversa es conocida como [la función W de Lambert](#) o [la función log-producto](#): $x = f^{-1}(y) = W(y)$. Esta función **no se puede expresar en términos de funciones elementales**. Esto provoca que no sea trivial hallar, por ejemplo, el valor a tal que $f(a) = ae^a = 2$ o, en otras palabras, $a = W(2)$.

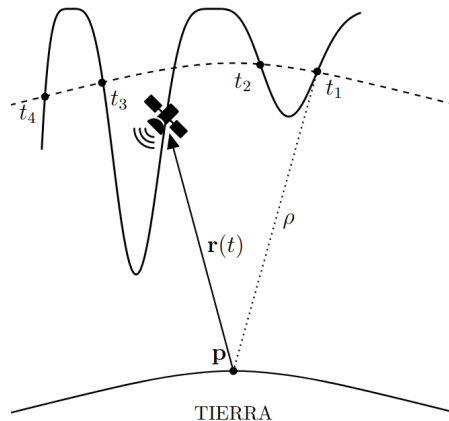
Construye un algoritmo que, dado un entero n , use n iteraciones del método de Newton-Raphson para aproximar aquel valor a tal que $ae^a = 2$.

3. Un satélite orbita alrededor de la Tierra, siguiendo una órbita descrita por un vector posición en función del tiempo: $\mathbf{r}(t) = (x(t), y(t))$.

Cuando este satélite está cerca de un punto de transmisión en la superficie terrestre, cuya posición es $\mathbf{p} = (p_x, p_y)$, pueden comunicarse y transmitir datos.

Si la distancia entre el satélite y el punto de transmisión es mayor a una distancia crítica ρ , entonces no hay transmisión de datos. Si la distancia entre el satélite en $\mathbf{r}(t)$ y el punto de transmisión en \mathbf{p} es $\|\mathbf{r}(t) - \mathbf{p}\|_2 = \sqrt{(x(t) - p_x)^2 + (y(t) - p_y)^2}$, entonces la condición de transmisión es que $\|\mathbf{r}(t) - \mathbf{p}\| < \rho$.

A medida que el satélite orbita la Tierra, va entrando y saliendo de la zona crítica. Los tiempos t en los que entra o sale de la zona crítica se denotan $t_1, t_2, t_3, t_4, \dots, t_k, \dots$



Dado el último tiempo t_k donde el satélite entró en (o salió de) la zona crítica, el desafío es encontrar el próximo tiempo t_{k+1} donde volverá a salir (o entrar).

Los científicos que estudian este sistema conocen una estimación de la duración de tiempo Δ_k que transcurrirá entre t_k y t_{k+1} . Con esta estimación, pueden aproximar $t_{k+1} \approx t_k + \Delta_k$.

La idea es usar un algoritmo para refinar esta aproximación. En particular, usarán una función $f(t)$ cuyas raíces son los tiempos t_k y usarán métodos de búsqueda de raíces para encontrar la raíz t_{k+1} tal que $f(t_{k+1}) = 0$.

La función $f(t)$ que están utilizando es

$$f(t) = (\|\mathbf{r}(t) - \mathbf{p}\|_2^2 - \rho^2)^2,$$

sobre la cual están aplicando el **método de Newton-Raphson**, debido a que saben que converge cuadráticamente en la mayoría de los casos.

Sin embargo, **su método no está convergiendo cuadráticamente, sino linealmente: los científicos estiman una tasa $S = \frac{1}{2}$** . ¿Por qué? ¿Cómo corregirías el algoritmo para lograr convergencia cuadrática? Implementa, en Python y con ayuda de NumPy, este algoritmo corregido para aproximar el próximo t_{k+1} con convergencia cuadrática, dados los parámetros ρ , Δ_k , p_x , p_y y t_k , las funciones $x(t)$ e $y(t)$ para representar las coordenadas del satélite, sus derivadas $x'(t)$ e $y'(t)$, y la función `Newton1D(f, fp, x0, m=1)` que aplica el método de Newton sobre la función `f`, cuya derivada es `fp`, con *initial guess* `x0` y con la capacidad opcional de especificar la multiplicidad `m` de la raíz que se busca aproximar.