Gradient Descent Example

April 5, 2022

GradientDescent
M : int function : Callable[ndarray(M,), float] gradient : Callable[ndarray(M,), ndarray(M,)] epsilon : float gamma : float
run(x0 : ndarray(M,)) : Dict[str : float]

Figure 1: Gradient descent class that optimizes a scalar function using its gradient with tolerance epsilon.

Data: Starting Point of Gradient Descent x_0 : ndarray(M,), Maximum iteration K
Result: Dictionary with keys:values {"success" : bool, "value" : float, "pt" : ndarray(M,)}
Raise TypeError if x_0 is not of type np.ndarray;
Raise ValueError if x_0 is not of length M;
Raise ValueError if *gamma* is not inbetween zero and one;
$x_{n-1} := x_0$;
$x_n := x_0 + \text{epsilon} \cdot M$;
$i := 0$;
while $||x_n - x_{n-1}||_2 <$ *epsilon* and $i < K$ **do**
    temp := x_n;
    $x_n = x_{n-1} - \gamma \cdot \text{gradient}(x_{n-1})$;
    x_{n-1} :=temp;
    $i := i + 1$;
end
If $i == K$ then success := False, Else success := True;
value := function(x_n);
pt := x_n

Algorithm 1: GradientDescent.run (Gradient Descent Algorithm)