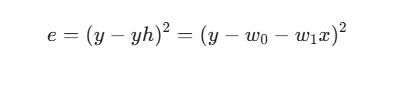
Mean squared error (MSE) as your cost function,

Let's break down the calculations you've provided:

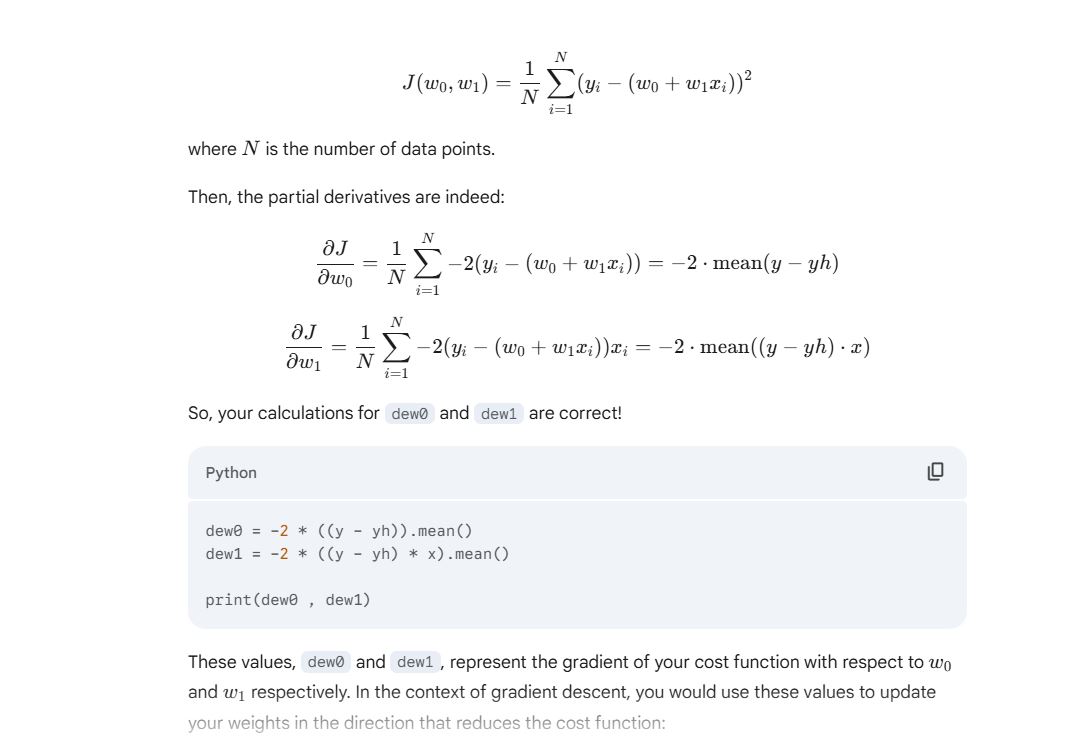
defined your error e for a single data point as the squared difference between the actual value y and the predicted value yh. Your prediction yh is based on a simple linear model:



where:

* y is the actual target value.
* yh is the predicted target value.
* w0​ is the bias (or intercept).
* w1​ is the weight (or slope) associated with the feature x.
* x is the feature value.

Now, you're calculating the partial derivatives of the cost function with respect to the parameters w0​ and w1​. Assuming you have multiple data points and you're using the *mean* squared error as your cost function J:



Here's a breakdown of why that's so important in the context of your linear regression example:

1. **Guiding the Optimization Process:** In machine learning, particularly when using gradient-based optimization algorithms like gradient descent, our goal is to find the set of parameters (in your case, w0​ and w1​) that minimize the cost function (like the mean squared error).
   * The partial derivative ∂w0​∂J​ tells us how much the cost function J will change if we make a small change in the value of w0​, *keeping w1​ constant*.
   * Similarly, ∂w1​∂J​ tells us how much the cost function J will change if we make a small change in the value of w1​, *keeping w0​ constant*.
2. **Determining the Direction of Steepest Descent (or Ascent):** The gradient, which is a vector composed of all the partial derivatives (∇J=[∂w0​∂J​,∂w1​∂J​] in your case), points in the direction of the **steepest increase** of the function at a particular point in the parameter space (w0​,w1​).
   * To *minimize* the cost function, we want to move in the *opposite* direction of the gradient. This is why in gradient descent, we update the parameters by subtracting a fraction (determined by the learning rate) of the gradient.
3. **Understanding the Sensitivity of the Cost Function:** The magnitude of the partial derivative tells us how sensitive the cost function is to changes in that particular parameter.
   * A large absolute value of ∂w1​∂J​ indicates that even a small change in w1​ will lead to a significant change in the cost function.
   * A small absolute value of ∂w0​∂J​ suggests that the cost function is less sensitive to changes in w0​ at the current parameter values.
4. **Finding Critical Points:** In calculus, we know that at local minima, local maxima, or saddle points of a function, the derivative (or all partial derivatives for a multivariable function) are equal to zero. By finding the points where the gradient is zero, we can identify potential optimal parameter values.

**In essence, partial derivatives provide the local "slope" of the cost function with respect to each parameter. This local information is the key that allows optimization algorithms to iteratively adjust the parameters in a way that gradually reduces the error and leads to a better model.**