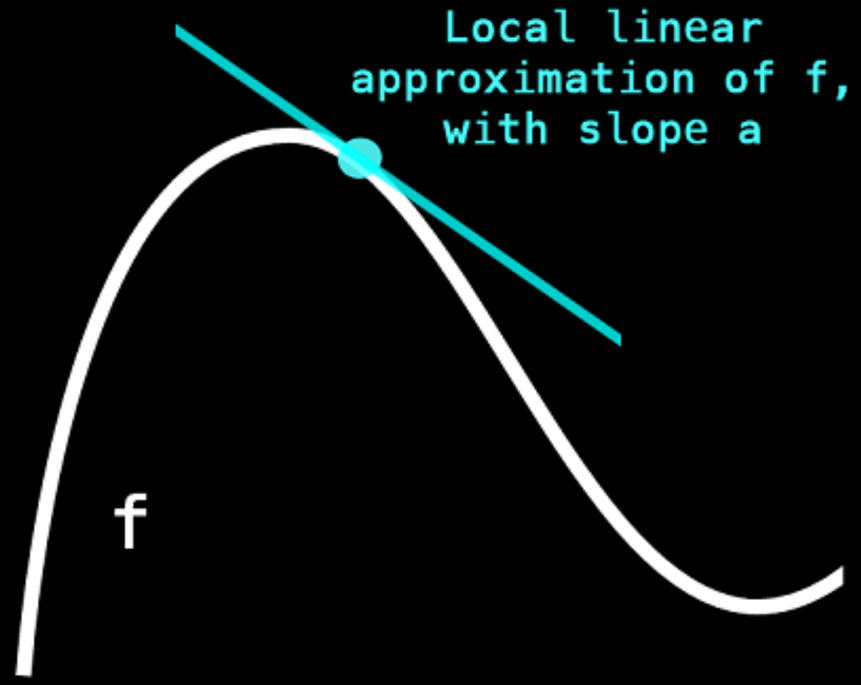
Training loop

1. Draw a batch of training samples x and corresponding targets y
2. Run the network on x (this is called "forward pass”) obtain predictions y_pred
3. Compute the "loss" of the network on the batch, a measure of the mismatch between y_pred and y
4. **Update all weights of the network in a way that slightly reduces the loss on this batch:**

    4.1. Compute the gradient of the loss with regard to the parameters of the network (this is called "backward pass")

    4.2. Move the parameters a little in the direction opposite to the gradient, thus lowering the loss on the batch by a bit.

Derivative in 1D



(c) Francois Chollet. “Deep Learning with Python”