Building Blocks of ML algorithms

- From previous slide

$$\underset{\mathbf{w}}{\operatorname{argmin}} C(\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{m} L(f(\mathbf{x}_i, \mathbf{w}), y_i) + \lambda \mathbf{w}^T \mathbf{w}$$

How to find?
* Gradient Descent gives improved estimates of w
a.k.a. C(w) should decrease

- To make it even more general, we can write

good "w" ~ small C(w)
the best "w" ~ w that minimises C(w)

$$\underset{\mathbf{w}}{\operatorname{argmin}} C(\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{i=1}^{m} L(f(\mathbf{x}_i, \mathbf{w}), y_i) + \lambda R(\mathbf{w})$$

no. of samples; Error; pred; true; output; inputs; parameters; eg. in ridge (lin) reg w/w.

- Learning model f reflects our belief about the relationship between the features $\mathbf{x}_i$ & target y_i
- Loss function L is the penalty for predicting $f(\mathbf{x}_i, \mathbf{w})$ when the true value is y_i
- Regularization R encourages less complex models
- Cost function C is the final optimization criterion we want to minimize
- Optimization routine to find solution to cost function

Derivative and Gradient

The gradient of a function is a vector of **partial derivatives**

Differentiation of a scalar function w.r.t. a vector

If $f(\mathbf{x})$ is a scalar function of d variables, $\mathbf{x}$ is a d x1 vector. Then differentiation of $f(\mathbf{x})$ w.r.t. $\mathbf{x}$ results in a d x1 vector

$$\frac{df(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_d} \end{bmatrix}$$

This is referred to as the **gradient** of $f(\mathbf{x})$ and often written as $\nabla_{\mathbf{x}} f$.

E.g. $f(\mathbf{x}) = ax_1 + bx_2 \quad \nabla_{\mathbf{x}} f = \begin{bmatrix} a \\ b \end{bmatrix}$

Ref: Duda, Hart, and Stork, "Pattern Classification", 2001 (Appendix)

Derivative and Gradient

Partial Derivatives

Differentiation of a vector function w.r.t. a vector

If $\mathbf{f}(\mathbf{x})$ is a vector function of *size h* x1 and $\mathbf{x}$ is a *d* x1 vector. Then differentiation of $\mathbf{f}(\mathbf{x})$ results in a *h* x *d* matrix

$$\frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_h}{\partial x_1} & \cdots & \frac{\partial f_h}{\partial x_d} \end{bmatrix}$$

The matrix is referred to as the **Jacobian** of $\mathbf{f}(\mathbf{x})$

(total) derivative.

Ref: Duda, Hart, and Stork, "Pattern Classification", 2001 (Appendix)

Linear Alg. recap

① $L = h\begin{pmatrix} L_1 & L_2 & \cdots & L_d \end{pmatrix}$: $h \times d$ matrix $\sim$ Linear map

$L(-): \mathbb{R}^d \to \mathbb{R}^h$

$v \mapsto L(v)$

sending

$\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \mapsto L_1$

$\begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \mapsto L_2$

$\vdots$

$\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \mapsto L_d$

② The function $x^T(\bullet): \mathbb{R}^d \to \mathbb{R}$

$w \mapsto x^T w$

Fix $x = \begin{pmatrix} x^1 \\ \vdots \\ x^d \end{pmatrix}$ is just a linear map

$\begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix} \to \begin{pmatrix} x^1 & \cdots & x^d \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix}$

$= x^1 w_1 + \cdots + x^d w_d$

$= \begin{pmatrix} x^1 \\ \vdots \\ x^d \end{pmatrix} \bullet \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix}$

↑ dot product

Terminology

Fix a function $f : \mathbb{R}^d \to \mathbb{R}^h$ (position)

dummy var. $\to x_0 \mapsto f(x_0) = \begin{pmatrix} f_1(x_0) \\ f_2(x_0) \\ \vdots \\ f_h(x_0) \end{pmatrix}$

The **Jacobian** or (total) **derivative** is a function

$$\frac{df}{dx} : \mathbb{R}^d \to \mathbb{R}^{h \times d}$$

$$x_0 \mapsto h\left(\frac{df}{dx}(x_0)\right)_d = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_0) & \cdots & \frac{\partial f_1}{\partial x_d}(x_0) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_h}{\partial x_1}(x_0) & \cdots & \frac{\partial f_h}{\partial x_d}(x_0) \end{bmatrix}$$

x_0: dummy variable for position

this is called the **derivative of x_0** (position). If is a **linear map**

key idea:

$$\underbrace{f(x_0 + \varepsilon)}_{\text{output in } \mathbb{R}^h} - f(x_0) \approx \underbrace{\left(\frac{df}{dx}(x_0)\right)}_{h \times d \text{ matrix, aka linear map}} (\varepsilon)$$

x_0: position in $\mathbb{R}^d$

ε: must be small direction change in $\mathbb{R}^d$

$f(x_0)$: output in $\mathbb{R}^h$

ε: direction vector; $\mathbb{R}^d$

Special case: $h = 1$

$$f : \mathbb{R}^d \to \mathbb{R}, \quad x_0 \mapsto f(x_0)$$

The gradient is a function $\nabla_x f : \mathbb{R}^d \to \mathbb{R}^d$, $x_0 \mapsto \nabla_x f(x_0)$

this is called the gradient of f at position x_0

$$= \begin{pmatrix} \frac{\partial f}{\partial x_1}(x_0) \\ \vdots \\ \frac{\partial f}{\partial x_d}(x_0) \end{pmatrix}$$

The Jacobian or (total) derivative is a function $\frac{df}{dx} : \mathbb{R}^d \to \mathbb{R}^{1 \times d}$

x_0 ← dummy variable for position

$$\mapsto \left(\frac{df}{dx}(x_0)\right) = \left(\frac{\partial f}{\partial x_1}(x_0), \frac{\partial f}{\partial x_2}(x_0), \dots, \frac{\partial f}{\partial x_d}(x_0)\right) = \nabla_x f(x_0)^T$$

this is called the derivative of f at position x_0. If it is a linear map

Key idea:

$$f(x_0 + \varepsilon) - f(x_0) \approx \left[\frac{df}{dx}(x_0)\right](\varepsilon) = \nabla_x f(x_0)^T(\varepsilon) \overset{\text{same as}}{=} \nabla_x f(x_0) \cdot \vec{\varepsilon}$$

$x_0 + \varepsilon$: $\varepsilon \in \mathbb{R}^1$; x_0 position in $\mathbb{R}^d$; ε small step size in $\mathbb{R}^d$

$f(x_0)$: $\in \mathbb{R}^1$

$\frac{df}{dx}(x_0)$: $\in \mathbb{R}^{1 \times d}$, linear map

ε: small step in $\mathbb{R}^d$

$\nabla_x f(x_0)$: $\in \mathbb{R}^d$

$\cdot$: dot product

Relationship between gradient & derivative ($h=1$)

$$\frac{df}{dx}(x_0)(-) = \left(\frac{\partial f}{\partial x_1}(x_0), \frac{\partial f}{\partial x_2}(x_0), \dots, \frac{\partial f}{\partial x_d}\right)(-)$$

linear map $\mathbb{R}^d \to \mathbb{R}$.

$$= \begin{pmatrix} \frac{\partial f}{\partial x_1}(x_0) \\ \vdots \\ \frac{\partial f}{\partial x_d}(x_0) \end{pmatrix}^T (-)$$

linear map

$$= \nabla_x f(x_0)^T (-) : \mathbb{R}^d \to \mathbb{R}$$

i.e. : the gradient is the transpose of the $(1 \times d)$ matrix $\left(\frac{df}{dx}\right)^T$

Example 1: Fix $x = \begin{pmatrix} x^1 \\ x^2 \\ \vdots \\ x^d \end{pmatrix}$. Consider the function

$$x^T(\cdot) : \mathbb{R}^d \to \mathbb{R}$$

$$w = \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix} \mapsto x^T w = (x^1 \; x^2 \cdots x^d) \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix}$$

$$= x^1 w_1 + x^2 w_2 + \cdots + x^d w_d$$

$$= x \cdot w$$

(dot product)

The derivative is

$$\frac{dx^T(\cdot)}{dw} : \mathbb{R}^d \to \mathbb{R}^{1 \times d}$$

$$w_0 \mapsto \left(\frac{\partial x^T(\cdot)}{\partial w_1}\Big|_{w=w_0} \quad \frac{\partial x^T(\cdot)}{\partial w_2}\Big|_{w=w_0} \cdots \quad \frac{\partial x^T(\cdot)}{\partial w_d}\Big|_{w=w_0} \right)$$

$$= (x^1 \; x^2 \cdots x^d) = x^T$$

The gradient is $\nabla_w x^T : \mathbb{R}^d \to \mathbb{R}^d$

$$w_0 \longmapsto x$$

Example 2:

$\sigma : \mathbb{R} \to \mathbb{R}$

$a_0 \longmapsto \dfrac{1}{1+\exp(-\beta a_0)}$

where $\beta \in \mathbb{R}^+$ is some positive real number

$$\begin{aligned}
\frac{\partial \sigma(a)}{\partial a} &= \frac{\partial}{\partial a}\left(\frac{1}{1+\exp(-\beta a)}\right) \\
&= -\frac{1}{(1+e^{-\beta a})^2}\frac{\partial(1+e^{-\beta a})}{\partial a} \\
&= \frac{\beta}{(1+e^{-\beta a})^2}e^{-\beta a} \\
&= \frac{\beta}{(1+e^{-\beta a})^2}(1+e^{-\beta a}-1) \\
&= \beta\left(\frac{1}{1+e^{-\beta a}} - \frac{1}{(1+e^{-\beta a})^2}\right) \\
&= \beta\left(\sigma(a) - \sigma^2(a)\right) \\
&= \beta\sigma(a)(1-\sigma(a))
\end{aligned}$$

$\nabla_a \sigma : \mathbb{R} \to \mathbb{R}^1$

$a_0 \longmapsto \left[\dfrac{\partial}{\partial a}\dfrac{1}{1+\exp(-\beta a)}\right]\Big|_{a=a_0}$

$= \beta\,\sigma(a_0)(1-\sigma(a_0))$

$= \beta \dfrac{1}{1+\exp(-\beta a_0)} \cdot \left(1 - \dfrac{1}{1+\exp(-\beta a_0)}\right)$

$\dfrac{d\sigma}{da} : \mathbb{R} \to \mathbb{R}^{|K|}$

$a_0 \longmapsto \left[\nabla_a \sigma(a_0)\right]^T \cdot (\ -\ )$

1×1 matrix.

Example 3:

$$f : \mathbb{R}^2 \to \mathbb{R}$$

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \mapsto w_1^2 + w_2^2$$

$$\nabla_w f : \mathbb{R}^2 \to \mathbb{R}^2$$

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \mapsto \begin{pmatrix} 2w_1 \\ 2w_2 \end{pmatrix}$$

$$\frac{df}{dw} : \mathbb{R}^2 \to \mathbb{R}^{1\times 2}$$

$$\begin{pmatrix} w_1^\circ \\ w_2^\circ \end{pmatrix} \mapsto \left(\frac{\partial f}{\partial w_1}\begin{pmatrix} w_1^\circ \\ w_2^\circ \end{pmatrix} \quad \frac{\partial f}{\partial w_2}\begin{pmatrix} w_1^\circ \\ w_2^\circ \end{pmatrix} \right)$$

$$= \left(2w_1\begin{pmatrix} w_1^\circ \\ w_2^\circ \end{pmatrix} \quad 2w_2\begin{pmatrix} w_1^\circ \\ w_2^\circ \end{pmatrix} \right)$$

$$= \begin{pmatrix} 2w_1^\circ & 2w_2^\circ \end{pmatrix}$$

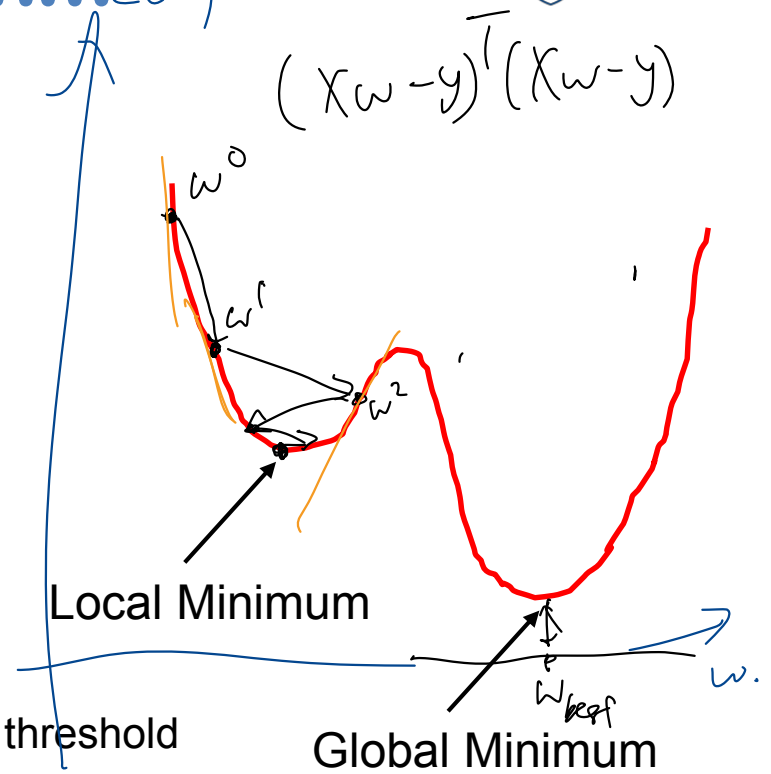
$$\begin{pmatrix} 2 \\ 3 \end{pmatrix} \mapsto \begin{pmatrix} 4 & 6 \end{pmatrix}$$

Gradient Descent Algorithm

- Gradient Descent:

Initialize $\mathbf{w}_0$ and learning rate η;
while *true* **do**
  Compute $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k - \eta \nabla_{\mathbf{w}} C(\mathbf{w}_k)$
  if *converge* **then**
    return $\mathbf{w}_{k+1}$
  end
end

opposite direction
gradient
adjust for suitable step size



- Possible convergence criteria
  - Set maximum iteration k
  - Check percentage or absolute change in C below a threshold
  - Check percentage or absolute change in $\mathbf{w}$ below a threshold
- Gradient descent can only find local minimum
  - Because gradient = 0 at local minimum, so $\mathbf{w}$ won't change after that
- Many variations of gradient descent, e.g., change how gradient is computed or learning rate η decreases with increasing k

Gradient Descent Algorithm

- Suppose we want to minimize $C(\mathbf{w})$ with respect to $\mathbf{w} = [w_1, \cdots, w_d]^T$
- Gradient $\nabla_{\mathbf{w}} C(\mathbf{w}) = \begin{pmatrix} \frac{\partial C}{\partial w_1} \\ \frac{\partial C}{\partial w_2} \\ \vdots \\ \frac{\partial C}{\partial w_d} \end{pmatrix}$
  - $\nabla_{\mathbf{w}} C(\mathbf{w})$ is vector & function of $\mathbf{w}$
  - $\nabla_{\mathbf{w}} C(\mathbf{w})$ is direction at $\mathbf{w}$ where C is increasing most rapidly, so $-\nabla_{\mathbf{w}} C(\mathbf{w})$ is direction at $\mathbf{w}$ where C is decreasing most rapidly
- Gradient Descent:

Initialize $\mathbf{w}_0$ and learning rate η;
while *true* **do**
  Compute $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k - \eta \nabla_{\mathbf{w}} C(\mathbf{w}_k)$
  if *converge* **then**
    return $\mathbf{w}_{k+1}$
  end
end

According to multi-variable calculus, if eta is not too big, then $C(\mathbf{w}_{k+1}) < C(\mathbf{w}_k)$ => we get better $\mathbf{w}$ after each iteration

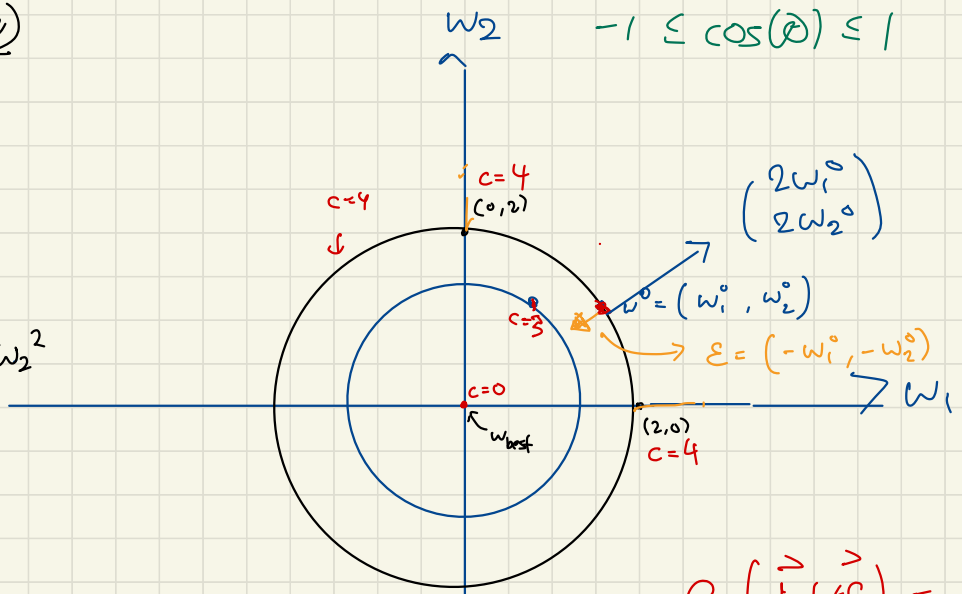
gradient descent (example)

$C(\vec{w}) = w_1^2 + w_2^2$

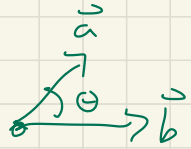
$C : \mathbb{R}^2 \to \mathbb{R}$

positiv

$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \mapsto w_1^2 + w_2^2$



$-1 \leq \cos(\theta) \leq 1$



$\vec{a} \cdot \vec{b} = \|\vec{a}\| \, \|\vec{b}\| \cdot \cos\theta$

$\cos(0) = 1$

$\cos(180) = -1$

$\cos(90) = 0$

$\nabla_w C : \mathbb{R}^2 \to \mathbb{R}^2$

$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \to \begin{pmatrix} \frac{\partial C}{\partial w_1}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \\ \frac{\partial C}{\partial w_2}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 2w_1 \\ 2w_2 \end{pmatrix}$

positiv.

$C(\vec{w}_0 + \vec{\varepsilon}) - C(\vec{w}_0)$

$\approx \nabla_w C(w_0) \cdot \vec{\varepsilon}$

$= \|\nabla_w C(w_0)\| \, \|\vec{\varepsilon}\| \cos\theta$

$\|\vec{\varepsilon}\| \sim 1$

ε ... unit. direction vector

EE2211 Tutorial 8

Question 1

Suppose we are minimizing $f(x) = x^4$ with respect to x. We initialize x to be 2. We perform gradient descent with learning rate 0.1. What is the value of x after the first iteration?

Question 2

Please consider the csv file (government-expenditure-on-education.csv), which depicts the government's educational expenditure over the years. We would like to predict expenditure as a function of year. To do this, fit an exponential model $f(\mathbf{x}, \mathbf{w}) = \exp(-\mathbf{x}^T\mathbf{w})$ with squared error loss to estimate $\mathbf{w}$ based on the csv file and gradient descent. In other words, $C(\mathbf{w}) = \sum_{i=1}^{m}(f(\mathbf{x}_i, \mathbf{w}) - y_i)^2$.

Note that even though year is one dimensional, we should add the bias term, so $\mathbf{x} = [1 \text{ year}]^T$. Furthermore, optimizing the exponential function is tricky (because a small change in $\mathbf{w}$ can lead to large change in f). Therefore for the purpose of optimization, divide the "year" variable by the largest year (2018) and divide the "expenditure" by the largest expenditure, so that the resulting normalized year and normalized expenditure variables have maximum values of 1. Use a learning rate of 0.03 and run gradient descent for 2000000 iterations.

(a) Plot the cost function $C(\mathbf{w})$ as a function of the number of iterations.

(b) Use the fitted parameters to plot the predicted educational expenditure from year 1981 to year 2023.

(c) Repeat (a) using a learning rate of 0.1 and learning rate of 0.001. What do you observe relative to (a)?

The goal of this question is for you to code up gradient descent, so I will provide you with the gradient derivation. First, please note that in general, $\nabla_{\mathbf{w}}(\mathbf{x}^T\mathbf{w}) = \mathbf{x}$. To see this:

$$\nabla_{\mathbf{w}}(\mathbf{x}^T\mathbf{w}) = \begin{bmatrix} \frac{\partial(\mathbf{x}^T\mathbf{w})}{\partial w_1} \\ \frac{\partial(\mathbf{x}^T\mathbf{w})}{\partial w_2} \\ \vdots \\ \frac{\partial(\mathbf{x}^T\mathbf{w})}{\partial w_d} \end{bmatrix} = \begin{bmatrix} \frac{\partial(w_1x_1+w_2x_2+\cdots+w_dx_d)}{\partial w_1} \\ \frac{\partial(w_1x_1+w_2x_2+\cdots+w_dx_d)}{\partial w_2} \\ \vdots \\ \frac{\partial(w_1x_1+w_2x_2+\cdots+w_dx_d)}{\partial w_d} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix} = \mathbf{x} \tag{1}$$

The above equality will be very useful for the other questions as well. Now, going back to our question,

$$\nabla_{\mathbf{w}} C(\mathbf{w}) = \nabla_{\mathbf{w}} \sum_{i=1}^{m} (f(\mathbf{x}_i, \mathbf{w}) - y_i)^2 \tag{2}$$

$$= \sum_{i=1}^{m} \nabla_{\mathbf{w}} (f(\mathbf{x}_i, \mathbf{w}) - y_i)^2 \tag{3}$$

$$= \sum_{i=1}^{m} 2(f(\mathbf{x}_i, \mathbf{w}) - y_i) \nabla_{\mathbf{w}} f(\mathbf{x}_i, \mathbf{w}) \quad \text{chain rule} \tag{4}$$

$$= \sum_{i=1}^{m} 2(f(\mathbf{x}_i, \mathbf{w}) - y_i) \nabla_{\mathbf{w}} \exp(-\mathbf{x}_i^T \mathbf{w}) \tag{5}$$

$$= -\sum_{i=1}^{m} 2(f(\mathbf{x}_i, \mathbf{w}) - y_i) \exp(-\mathbf{x}_i^T \mathbf{w}) \nabla_{\mathbf{w}} (\mathbf{x}_i^T \mathbf{w}) \quad \text{chain rule} \tag{6}$$

$$= -\sum_{i=1}^{m} 2(f(\mathbf{x}_i, \mathbf{w}) - y_i) \exp(-\mathbf{x}_i^T \mathbf{w}) \mathbf{x}_i \tag{7}$$

$$= -\sum_{i=1}^{m} 2(f(\mathbf{x}_i, \mathbf{w}) - y_i) f(\mathbf{x}_i, \mathbf{w}) \mathbf{x}_i \tag{8}$$

+

Question 3

Given the linear learning model $f(\mathbf{x}, \mathbf{w}) = \mathbf{x}^T \mathbf{w}$, where $\mathbf{x} \in \mathbb{R}^d$. Consider the loss function $L(f(\mathbf{x}_i, \mathbf{w}), y_i) = (f(\mathbf{x}_i, \mathbf{w}) - y_i)^4$, where i indexes the i-th training sample. The final cost function is $C(\mathbf{w}) = \sum_{i=1}^{m} L(f(\mathbf{x}_i, \mathbf{w}), y_i)$, where m is the total number of training samples. Derive the gradient of the cost function with respect to $\mathbf{w}$.

Question 4

Repeat Question 3 using $f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{x}^T \mathbf{w})$, where $\sigma(a) = \frac{1}{1+\exp(-\beta a)}$

Question 5

Repeat Question 3 using $f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{x}^T \mathbf{w})$, where $\sigma(a) = \max(0, a)$

$\vec{w} \longrightarrow \boxed{f(-, \vec{x}_i)} \longrightarrow f(\vec{x}_i, w) \longrightarrow \boxed{L(-, y_i)} \longrightarrow L(f(\vec{x}_i, w), y_i)$

$f(-, \vec{x}_i) : \mathbb{R}^d \to \mathbb{R}$

$L(-, y_i) : \mathbb{R} \to \mathbb{R}$.

$C : \mathbb{R}^d \to \mathbb{R}$.

↓ add everything up

* differentiation is linear.

$\nabla \sum$ stuff. $= \sum \nabla$ stuff.

Question 3

Given the linear learning model $f(\mathbf{x}, \mathbf{w}) = \mathbf{x}^T\mathbf{w}$, where $\mathbf{x} \in \mathbb{R}^d$. Consider the loss function $L(f(\mathbf{x}_i, \mathbf{w}), y_i) = (f(\mathbf{x}_i, \mathbf{w}) - y_i)^4$, where i indexes the i-th training sample. The final cost function is $C(\mathbf{w}) = \sum_{i=1}^{m} L(f(\mathbf{x}_i, \mathbf{w}), y_i)$, where m is the total number of training samples. Derive the gradient of the cost function with respect to $\mathbf{w}$.

For each $x_i = \begin{pmatrix} x_i^1 \\ x_i^2 \\ \vdots \\ x_i^d \end{pmatrix} \in \mathbb{R}^d$, $y_i \in \mathbb{R}$

HAVE

$f(x_i, -) : \mathbb{R}^d \to \mathbb{R}$

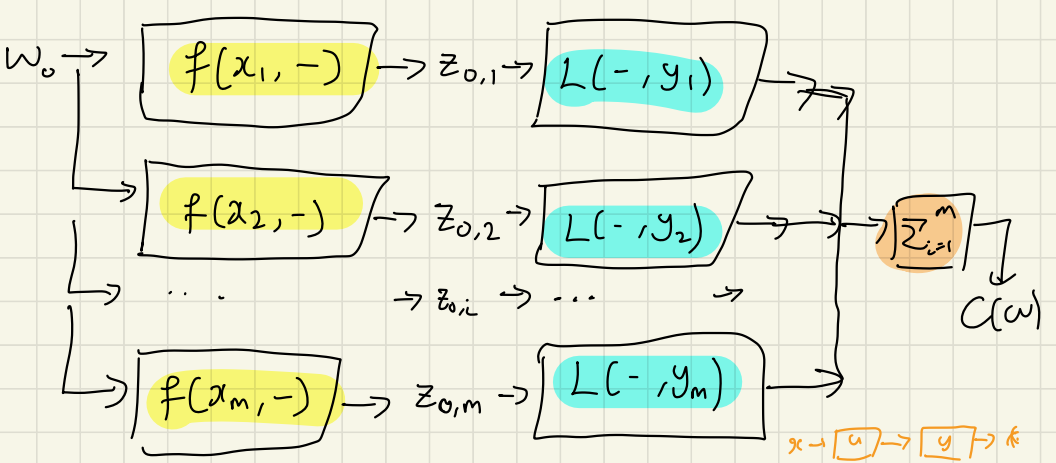
$w \mapsto x_i^T w$

$L(-, y_i) : \mathbb{R} \to \mathbb{R}$

$z \mapsto (z - y_i)^4$

$\sum_{i=1}^{m} : \mathbb{R}^m \to \mathbb{R}$

$\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{pmatrix} \mapsto c_1 + c_2 + \cdots + c_m$



How to differentiate?

Slogan : $\frac{dy}{dx} = \left[\frac{dy}{du}\right]\left[\frac{du}{dx}\right]$

matrix = matrix · matrix

linear map = linear map (compose) linear map

$x_0 \mapsto u(x_0)$

$u : \mathbb{R}^d \to \mathbb{R}^h$

$y : \mathbb{R}^h \to \mathbb{R}^k$, $u_0 \mapsto y(u_0)$

$\frac{du}{dx} : \mathbb{R}^d \to \mathbb{R}^{h \times d}$, $x_0 \mapsto \left[\frac{du}{dx}(x_0)\right]$

$\frac{dy}{du} : \mathbb{R}^h \mapsto \mathbb{R}^{k \times h}$, $u_0 \mapsto \left[\frac{dy}{du}(-)\right]$

Intuition : $y \circ u(x_0 + \varepsilon) - y \circ u(x_0) \approx_k \left[\frac{dy}{du}(u(x_0))\right]_h \left[\frac{du}{dx}(x_0)\right]_d (\varepsilon)$, $\varepsilon \in \mathbb{R}^d$

Let's calculate the derivative of w_0 $\quad \frac{dc}{dw}(w_0) \in \mathbb{R}^{1\times d}$

$$\left[\frac{df(x_i,-)}{dw}(w_0)\right] = \begin{pmatrix} \frac{\partial x_i^T(-)}{\partial w_1} & \frac{\partial x_i^T(-)}{\partial w_2} & \cdots & \frac{\partial x_i^T(-)}{\partial w_d} \end{pmatrix}$$

$$= 1\begin{pmatrix} x_i^1 & x_i^2 & \cdots & x_i^d \end{pmatrix} = x_i^T \in \mathbb{R}^{1\times d}$$

(d)

$$\frac{df(x_i,-)}{dw} : \mathbb{R}^d \to \mathbb{R}^{1\times d}$$

$$\frac{dL(-,y_i)}{dz} : \mathbb{R}^1 \to \mathbb{R}^{1\times 1}$$

$$\left[\frac{dL(-,y_i)}{dz}(z_0)\right] = 1\left[\frac{d(z-y)^4}{dz}\Big|_{z=z_0}\right] = \left[4(z_0-y)^3\right] \in \mathbb{R}^{1\times 1}$$

what is z_0 in chain rule?

$$= \left[4\left(x_i^T w_0 - y\right)^3\right] \in \mathbb{R}^{1\times 1}$$

must be $z_0 = f(x_i, w_0) = x_i^T w_0$

$$\left[\frac{d\sum_{i=1}^{m}}{dc}(c_0)\right] = 1\left[\underbrace{1 \quad 1 \quad \cdots \quad 1}_{m}\right] =$$

"sum over all entries".

OBV! Because differentiation is linear!

$$\frac{d}{dw}\sum_{i=1}^{m} F_i(w_0) = \sum_{i=1}^{m} \frac{dF_i}{dw}(w_0)$$

Everything TGT:

$L(-, y_i) \circ f(x_i, -) : \mathbb{R}^1 \to \mathbb{R}^1$

$$\left[\frac{dC}{dw}(w_0)\right] = \sum_{i=1}^{m} \left[\frac{d\left(L(f(x_i,-), y_i)\right)}{dw}(w_0)\right]$$

differentiation is linear

$$= \sum_{i=1}^{m} \underset{1}{\left[\frac{dL(-, y_i)}{dz}\left(z_0 = f(x_i, w_0)\right)\right]} \underset{1}{\left[\frac{df(x_i,-)}{dw}(w_0)\right]}$$

($\frac{dL}{dz}$: 1; $\frac{df}{dw}$: d)

$$= \sum_{i=1}^{m} \underset{1}{\left[4(z_0 - y)^3 \Big|_{z_0 = x_i^T w_0}\right]} \underset{d}{\left[x_i^T\right]}$$

$$= 1\left[\sum_{i=1}^{m} 4(x_i^T w_0 - y)^3 x_i^T\right]$$

AT w_0, we know

① $z_{0,i} = f(x_i, w_0) = x_i^T w_0$.

② $c_0 = \text{sth}$

$\therefore$ the gradient of w_0 is ($\text{gradient}^T(-) = \text{derivative}(-)$)

$$\vec{\nabla}_w C(w_0) = \left[\frac{dC}{dw}(w_0)\right]^T = \sum_{i=1}^{m} 4(x_i^T w_0 - y)\, x_i$$

this is a vector in $\mathbb{R}^d$

Alternative Perspective:

no. of training samples

Define $F(X,-): \mathbb{R}^d \to \mathbb{R}^m$

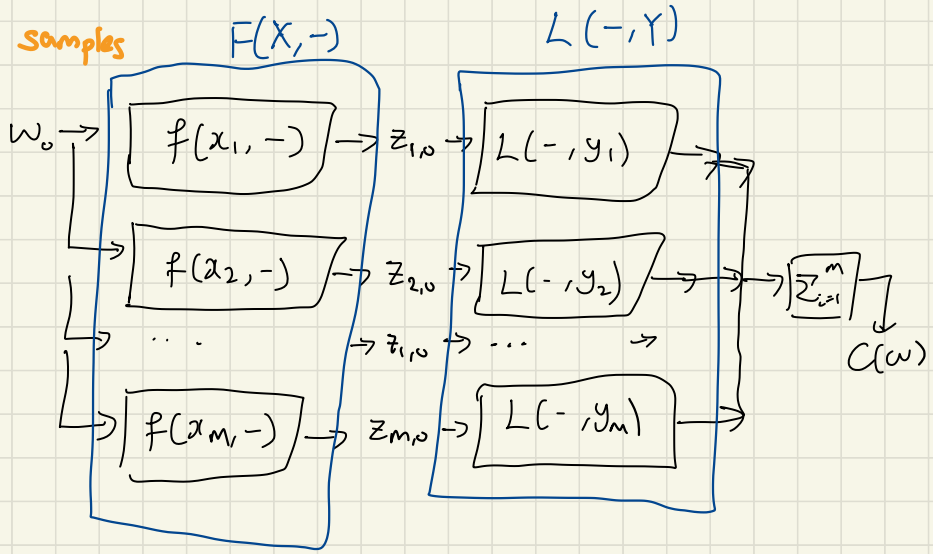
by $w_0 \mapsto \begin{pmatrix} x_1^T w_0 \\ x_2^T w_0 \\ \vdots \\ x_m^T w_0 \end{pmatrix}$

$L(-,Y): \mathbb{R}^m \to \mathbb{R}^m$

by $z_0 = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{pmatrix} \mapsto \begin{pmatrix} (z_1 - y_1)^4 \\ (z_2 - y_2)^4 \\ \vdots \\ (z_m - y_m)^4 \end{pmatrix}$

$\sum_{i=1}^m : \mathbb{R}^m \to \mathbb{R}$

by $\begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix} \mapsto c_1 + \dots + c_m$



Then, $\frac{dF(X,-)}{dw} : \mathbb{R}^d \to \mathbb{R}^{m \times d}$ is

$w_0 \mapsto \left[\frac{dF(X,-)}{dw}(w_0)\right] = \begin{bmatrix} \frac{\partial (x_1^T w)}{\partial w_1}\big|_{w=w_0} & \cdots & \frac{\partial (x_1^T w)}{\partial w_d}\big|_{w=w_0} \\ \frac{\partial (x_2^T w)}{\partial w_1}\big|_{w=w_0} & \cdots & \frac{\partial (x_2^T w)}{\partial w_d}\big|_{w=w_0} \\ \vdots & & \vdots \\ \frac{\partial (x_m^T w)}{\partial w_1}\big|_{w=w_0} & \cdots & \frac{\partial (x_m^T w)}{\partial w_d}\big|_{w=w_0} \end{bmatrix}$

$= \begin{pmatrix} w_1 \\ \vdots \\ w_d \end{pmatrix}$

observe by $x_i^T w$, I mean the fn $w \mapsto x_i^T w$. This is the component fn of F.

$$\begin{bmatrix} \frac{\partial (x_1^T w)}{\partial w_1}\big|_{w=w_0} & \cdots & \frac{\partial (x_1^T w)}{\partial w_m}\big|_{w=w_0} \\ \frac{\partial (x_2^T w)}{\partial w_2}\big|_{w=w_0} & & \vdots \\ \vdots & \ddots & \\ \frac{\partial (x_m^T w)}{\partial w_2}\big|_{w=w_0} & \cdots & \frac{\partial (x_m^T w)}{\partial w_m}\big|_{w=w_0} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_1^{(2)} & \cdots & x_1^{(d)} \\ x_2^{(1)} & x_2^{(2)} & \cdots & x_2^{(d)} \\ \vdots & \vdots & \ddots & \vdots \\ x_m^{(1)} & x_m^{(2)} & \cdots & x_m^{(d)} \end{bmatrix} = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix} \in \mathbb{R}^{m \times d}$$

(m rows, d columns)

Similarly we get

$$\frac{dL(-,Y)}{dz} : \mathbb{R}^m \to \mathbb{R}^{m\times m}$$

$$\begin{pmatrix} z_{1,0} \\ \vdots \\ z_{m,0} \end{pmatrix} = z_0 \mapsto \left[\frac{dL(-,Y)}{dz}(z_0) \right] = \begin{bmatrix} \frac{\partial (z_1 - y_1)^4}{\partial z_1}\Big|_{z=z_0} & \cdots & \frac{\partial (z_1 - y_1)^4}{\partial z_m}\Big|_{z=z_0} \\ \vdots & \ddots & \vdots \\ \frac{\partial (z_m - y_m)^4}{\partial z_1}\Big|_{z=z_0} & \cdots & \frac{\partial (z_m - y_m)^4}{\partial z_m}\Big|_{z=z_0} \end{bmatrix} = \begin{bmatrix} 4(z_{1,0}-y_1)^3 & & 0 \\ & 4(z_{2,0}-y_2)^3 & \\ & & \ddots \\ 0 & & 4(z_{m,0}-y_m)^3 \end{bmatrix}$$

diagonal matrix

(RECALL) $L(-,Y) : \mathbb{R}^m \to \mathbb{R}^m$ by $z_0 = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{pmatrix} \mapsto \begin{pmatrix} (z_1 - y_1)^4 \\ (z_2 - y_2)^4 \\ \vdots \\ (z_m - y_m)^4 \end{pmatrix}$

RECALL

$L(-,y_i) : \mathbb{R} \to \mathbb{R}$, $z \mapsto (z - y_i)^4$

$\Downarrow$

$\frac{dL(-,y_i)}{dz} : \mathbb{R}^1 \to \mathbb{R}^{1\times 1}$, $z_{i,0} \mapsto 4(z_{i,0} - y_i)^3$

OBSERVE :

$$= \begin{bmatrix} \frac{dL(-,y_1)}{dz}(z_{1,0}) & & 0 \\ & \ddots & \\ 0 & & \frac{dL(-,y_m)}{dz}(z_{m,0}) \end{bmatrix}$$

Similarly, $\frac{d\sum_{i=1}^m}{dc} : \mathbb{R}^m \to \mathbb{R}^{1\times m}$

$$c_0 \mapsto \left[\frac{d\sum_{i=1}^m}{dc}(c_0) \right] = 1\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}$$

$$\sum_{i=1}^m : \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{pmatrix} \mapsto c_1 + c_2 + \cdots + c_m$$

TGT: $C : \mathbb{R}^d \to \mathbb{R}$

$$w \longmapsto C(w) = \sum_{i=1}^{m} \circ L(-,Y) \circ F(X,-)(w)$$

$$\to \boxed{F(X,-) : \mathbb{R}^d \to \mathbb{R}^m} \to \boxed{L(-,Y) : \mathbb{R}^m \to \mathbb{R}^m} \to \boxed{\textstyle\sum_{i=1}^{m} : \mathbb{R}^m \to \mathbb{R}} \to$$

at position w_0, get

$$w_0 \longmapsto \begin{pmatrix} x_1^T w_0 \\ x_2^T w_0 \\ \vdots \\ x_m^T w_0 \end{pmatrix} \longmapsto z_0 = \begin{pmatrix} z_{1,0} \\ z_{2,0} \\ \vdots \\ z_{m,0} \end{pmatrix} = \begin{pmatrix} (x_1^T w_0 - y_1)^4 \\ (x_2^T w_0 - y_2)^4 \\ \vdots \\ (x_m^T w_0 - y_m)^4 \end{pmatrix} \longmapsto \underbrace{\sum_{i=1}^{m} (x_i^T w_0 - y_i)^4}_{c_0}$$

$$\frac{dC}{dw} : \mathbb{R}^d \to \mathbb{R}^{1\times d}$$

$$w_0 \longmapsto \left[\frac{dC}{dw}(w_0)\right] = \underset{1}{1}\left[\frac{d\sum_{i=1}^{m}}{dc}(c_0)\right] \underset{m}{m}\left[\frac{dL(-,Y)}{dz}(z_0)\right] \underset{d}{m}\left[\frac{dF(X,-)}{dw}(w_0)\right]$$

$$= \underset{m}{1}\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} \underset{m}{m}\begin{bmatrix} 4(x_1^T w_0 - y_1)^3 & & 0 \\ & \ddots & \\ 0 & & 4(x_m^T w_0 - y_m)^3 \end{bmatrix} \underset{d}{m}\begin{bmatrix} \vec{x}_1^T \\ \vec{x}_2^T \\ \vdots \\ \vec{x}_m^T \end{bmatrix}$$

($\vec{x}_i^T$: $1\times d$ matrix)

$$= \underset{d}{1}\left[\sum_{i=1}^{m} \underbrace{4(x_i^T w_0 - y_i)^3}_{1\times 1} \underbrace{x_i^T}_{1\times d \text{ matrix}}\right]$$

OBSERVE: each entry is raised by a scalar since this is diagonal

$$\therefore \quad \nabla_w C(w_0) = \left[\frac{dC}{dw}(w_0)\right]^T = \sum_{i=1}^{m} 4(x_i^T w_0 - y_i)\, x_i \quad (\in \mathbb{R}^d)$$

Question 4

Repeat Question 3 using $f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{x}^T\mathbf{w})$, where $\sigma(a) = \frac{1}{1+\exp(-\beta a)}$

$\sigma: \mathbb{R}^1 \to \mathbb{R}^1$

abuse notation

$C: \mathbb{R}^d \to \mathbb{R}$

$F(X,-): \mathbb{R}^d \to \mathbb{R}^m$

$$w \longmapsto \begin{pmatrix} x_1^T w \\ x_2^T w \\ \vdots \\ x_m^T w \end{pmatrix} \quad \sigma: \mathbb{R}^m \to \mathbb{R}^m \quad \begin{pmatrix} \sigma(x_1^T w) \\ \sigma(x_2^T w) \\ \vdots \\ \sigma(x_m^T w) \end{pmatrix} \quad L(-,Y): \mathbb{R}^m \to \mathbb{R}^d \quad \begin{pmatrix} (\sigma(x_1^T w) - y_1)^4 \\ \vdots \\ (\sigma(x_m^T w) - y_m)^4 \end{pmatrix} \quad \sum_{i=1}^m : \mathbb{R}^d \to \mathbb{R}$$

$$\begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix} \mapsto \begin{pmatrix} \sigma(a_1) \\ \vdots \\ \sigma(a_m) \end{pmatrix} \qquad \begin{pmatrix} z_1 \\ \vdots \\ z_m \end{pmatrix} \mapsto \begin{pmatrix} (z_1 - y_1)^4 \\ \vdots \\ (z_m - y_m)^4 \end{pmatrix} \qquad \mapsto \sum_{i=1}^m (\sigma(x_i^T w) - y_i)^4$$

$$\frac{dC}{dw}: \mathbb{R}^d \to \mathbb{R}^{1 \times d}$$

$$w_0 \mapsto \underset{1}{\left[\frac{d\sum_{i=1}^m}{dc}(c_0)\right]} \underset{m}{\left[\frac{dL(-,Y)}{dz}(z_0)\right]} \underset{m}{\begin{bmatrix} \frac{d\sigma}{da}(a_{1,0}) & & 0 \\ & \ddots & \\ 0 & & \frac{d\sigma}{da}(a_m) \end{bmatrix}} \underset{m}{\left[\frac{dF(X,-)}{dw}(w_0)\right]}$$

diagonal

diagonal

$$= \left[\text{add everything up. Differentiation is linear}\right] \left[\text{scalar mult. the } i^{th} \text{ entry by } \frac{dL(-,y_i)}{dz}(\sigma(x_i^T w_0))\right] \left[\text{scalar mult. the } i^{th} \text{ entry by } \frac{d\sigma}{da}(x_i^T w_0)\right] \underset{m}{\begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix}}$$

vector in $\mathbb{R}^{1\times d}$; m copies

$$= \sum_{i=1}^m 4(\sigma(x_i^T w_0) - y_i)^3 \cdot \beta\,\sigma(x_i^T w_0)(1 - \sigma(x_i^T w_0))\, x_i^T$$

$$\therefore \quad \nabla_w C(w_0) = \left[\frac{dC}{dw}(w_0)\right]^T = \sum_{i=1}^m 4(\sigma(x_i^T w_0) - y_i)^3 \cdot \beta\,\sigma(x_i^T w_0)(1 - \sigma(x_i^T w_0))\, x_i$$

Question 5

Repeat Question 3 using $f(\mathbf{x}, \mathbf{w}) = \sigma(\mathbf{x}^T\mathbf{w})$, where $\sigma(a) = \max(0, a)$

ReLU.

$$\sigma(a) = \begin{cases} a, & a > 0 \\ 0, & a \le 0 \end{cases}$$

same story. what is derivative of σ?

$$\frac{d\sigma}{da}(a_0) = \begin{cases} 1, & a_0 > 0 \\ 0, & a_0 < 0 \\ \text{undefined}, & a_0 = 0 \end{cases}$$

$$\frac{dC}{dw} : \mathbb{R}^d \to \mathbb{R}^{1\times d}$$

$$w_0 \mapsto \underline{1}\left[\frac{d\sum_{i=1}^m}{dc}(c_0)\right]\; m\left[\frac{dL(-,Y)}{dz}(z_0)\right]\; m\left[\begin{matrix}\frac{d\sigma}{da}(a_{1,0}) & & 0 \\ & \ddots & \\ 0 & & \frac{d\sigma}{da}(a_m)\end{matrix}\right]\; m\left[\frac{dF(X,-)}{dw}(w_0)\right]$$

diagonal (m) diagonal (m) d

$$= \left[\text{add everything up. Differentiation is linear}\right]\left[\text{scalar mult. the } i^{th} \text{ entry by } \frac{dL(-,y_i)}{dz}(\sigma(x_i^T w_0))\right]\left[\text{scalar mult. the } i^{th} \text{ entry by } \frac{d\sigma}{da}(x_i^T w_0)\right]\left[\begin{matrix}x_1^T \\ x_2^T \\ \vdots \\ x_m^T\end{matrix}\right]$$

$\sigma(x_i^T w_0)$: what this? $= x_i^T w_0$, if $x_i^T w_0 \ge 0$; 0 if $x_i^T w_0 \le 0$.

$\frac{d\sigma}{da}(x_i^T w_0)$: what is this? This is 1, if $x_i^T w_0 > 0$; 0, if $x_i^T w_0 < 0$

x_i^T: vector in $\mathbb{R}^{1\times d}$, m copies

$$= \sum_{i=1}^m 4\left(\mathbb{1}_{\{x_i^T w_0 \ge 0\}}(x_i^T w_0) - y_i\right)^3 \cdot \begin{pmatrix} 1 \text{ if } x_i^T w_0 > 0 \\ 0 \text{ if } x_i^T w_0 < 0 \end{pmatrix} \cdot x_i^T$$

If sure ready.

$$= \sum_{i=1}^m 4\left(\mathbb{1}_{\{x_i^T w_0 \ge 0\}}(x_i^T w_0) - y_i\right)^3 \cdot \mathbb{1}_{\{x_i^T w_0 > 0\}} \cdot x_i^T = \sum_{\substack{i=1, \\ x_i^T w_0 > 0}}^m 4\left(x_i^T w_0 - y_i\right)^3 \cdot x_i^T$$

delta fn. indicator fn. Just a name.

$$\therefore \nabla_w C(w_0) = \left[\frac{dC}{dw}(w_0)\right]^T = \sum_{\substack{i=1, \\ x_i^T w_0 > 0}}^m 4\left(x_i^T w_0 - y_i\right)^3 \cdot x_i$$

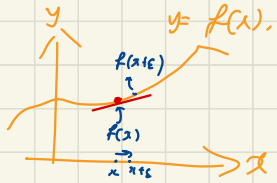
Recall

If $f: \mathbb{R} \to \mathbb{R}$ is a differentiable fn,

then $x \mapsto \frac{df}{dx}(x) = f'(x)$ is a function

$f': \mathbb{R} \to \mathbb{R}$ s.t.

$$f(x+\varepsilon) - f(x) \approx f'(x) \cdot \varepsilon .$$



If $f: \mathbb{R}^d \to \mathbb{R}$ is a cts differentiable fn,

then $\vec{x} \mapsto \vec{\nabla} f(\vec{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \vdots \\ \frac{\partial f}{\partial x_d} \end{pmatrix}^* \in \mathbb{R}^d$

$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix}$

is a function $\vec{\nabla} f: \mathbb{R}^d \to \mathbb{R}^d$

dot prod. by $\vec{\nabla} f(\vec{x})$ study $\mathbb{R}^d \to \mathbb{R}$. and is a linear map.

$$f(\vec{x}+\vec{\varepsilon}) - f(\vec{x}) \approx \vec{\nabla} f(\vec{x}) \cdot \vec{\varepsilon}$$

($\vec{\nabla} f(\vec{x})$: vector in $\mathbb{R}^d$; $\vec{\varepsilon}$: vector in $\mathbb{R}^d$; $\cdot$: dot product.)

Recall dot product of vectors.

$\vec{v} \cdot \vec{w} = \|\vec{v}\| \|\vec{w}\| \cos\theta$

$f(x,y) = x^2 + y^2$

$$\nabla f \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial x}(x_0, y_0) \\ \frac{\partial f}{\partial y}(x_0, y_0) \end{pmatrix} = \begin{pmatrix} 2x_0 \\ 2y_0 \end{pmatrix}.$$