

**Homework 2: Stochastic learning: Stochastic Gradient Descent**

Lecturer: Georgios P. Karagiannis

georgios.karagiannis@durham.ac.uk

As formative assessment, submit the solutions to Exercise 1.2, 1.3, and 1.4.

**Exercise 1.** (★★) <sup>1</sup>Consider the binary classification problem with inputs  $x \in \mathcal{X}$  where  $\mathcal{X} := \{x \in \mathbb{R}^d : \|x\|_2 \leq L\}$  for some given value  $L > 0$ , target  $y \in \mathcal{Y}$  where  $\mathcal{Y} := \{-1, +1\}$ , and prediction rule  $h_w : \mathbb{R}^d \rightarrow \{-1, +1\}$  with

$$(0.1) \quad h_w(x) = \text{sign}(w^\top x)$$

$$(0.2) \quad = \text{sign}\left(\sum_{j=1}^d w_j x_j\right)$$

Let the hypothesis class is

$$(0.3) \quad \mathcal{H} = \{x \rightarrow w^\top x : \forall w \in \mathbb{R}^d\}$$

In other words, the hypothesis  $h_w \in \mathcal{H}$  is parametrized by  $w \in \mathbb{R}^d$ , it receives an input vector  $x \in \mathcal{X} := \mathbb{R}^d$  and it returns the label  $y = \text{sign}(w^\top x) \in \mathcal{Y} := \{\pm 1\}$  where

$$\text{sign}(\xi) = \begin{cases} -1, & \text{if } \xi < 0 \\ +1, & \text{if } \xi > 0 \end{cases}$$

Consider a loss function  $\ell : \mathbb{R}^d \rightarrow \mathbb{R}_+$  with

$$(0.4) \quad \ell(w, z = (x, y)) = \max(0, 1 - yw^\top x) + \lambda \|w\|_2^2$$

for some given value  $\lambda > 0$ .

Assume there is available a dataset of examples  $S_n = \{z_i = (x_i, y_i) : i = 1, \dots, n\}$  of size  $n$ .

Do the following:

- (1) Show that the function  $f : \mathbb{R} \rightarrow \mathbb{R}_+$  with  $f(x) = \max(0, 1 - x)$  is convex in  $\mathbb{R}$ ; and show that the loss (0.4) is convex.

**Hint::** You may use Proposition 12 from Handout 2: Elements of convex learning problems.

<sup>1</sup>We use standard notation

$$\text{sign}(\xi) = \begin{cases} -1, & \text{if } \xi < 0 \\ +1, & \text{if } \xi > 0 \end{cases}$$

$\pm 1$  means either  $-1$  or  $+1$ ,  $\mathbb{R}_+ := (0, +\infty)$ , and  $\|x\|_2 := \sqrt{\sum_{j=1}^d (x_j)^2}$  for the Euclidean distance.

- (2) Show that the loss  $\ell(w, z)$  for  $\lambda = 0$  (0.4) is  $L$ -Lipschitz (with respect to  $w$ ) when  $x \in \mathcal{X}$  where  $\mathcal{X} := \{x \in \mathbb{R}^d : \|x\|_2 \leq L\}$ .

**Hint::** You may use the definition of Lipschitz function. Without loss of generality, you can consider any  $w_1 \in \mathbb{R}^d$  and  $w_2 \in \mathbb{R}^d$  such that  $1 - yw_2^\top x \leq 1 - yw_1^\top x$ , and then take cases  $1 - yw_2^\top x > \text{or} < 0$  and  $1 - yw_1^\top x > \text{or} < 0$  to deal with the max.

- (3) Construct the set of sub-gradients  $\partial f(x)$  for  $x \in \mathbb{R}$  of the function  $f : \mathbb{R} \rightarrow \mathbb{R}_+$  with  $f(x) = \max(0, 1 - x)$ . Show that the vector  $v$  with

$$v = \begin{cases} 2\lambda w, & yw^\top x > 1 \\ 2\lambda w, & yw^\top x = 1 \\ -yx + 2\lambda w, & yw^\top x < 1 \end{cases}$$

is  $v \in \partial_w \ell(w, z = (x, y))$ , aka a sub-gradient of  $\ell(w, z = (x, y))$  at  $w$ , for any  $w \in \mathbb{R}^d$ .

- (4) Write down the algorithm of online AdaGrad (Adaptive Stochastic Gradient Descent) with learning rate  $\eta_t > 0$ , batch size  $m$ , and termination criterion  $t > T_{\max}$  for some  $T_{\max} > 0$  in order to discover  $w^*$  such as

$$(0.5) \quad w^* = \arg \min_{w: h_w \in \mathcal{H}} (\mathbb{E}_{z \sim g} (\ell(w, z = (x, y))))$$

The formulas in your algorithm should be implemented for the above learning problem and tailored to 0.1, 0.3, and 0.4.

- (5) Use the R code given below in order to generate the dataset of observed examples  $S_n = \{z_i = (x_i, y_i)\}_{i=1}^n$  that contains  $n = 10^6$  examples with inputs  $x$  of dimension  $d = 2$ . Consider  $\lambda = 0$ . Use a seed  $w^{(0)} = (0, 0)^\top$ .
- By using appropriate values for  $m$ ,  $\eta_t$  and  $T_{\max}$ , code in R the algorithm you designed in part 4, and run it.
  - Plot the trace plots for each of the dimensions of the generated chain  $\{w^{(t)}\}$  against the iteration  $t$ .
  - Report the value of the output  $w_{\text{adaGrad}}^*$  (any type) of the algorithm as the solution to (0.5).
  - To which cluster  $y$  (i.e.,  $-1$  or  $1$ )  $x_{\text{new}} = (1, 0)^\top$  belongs?

```

# R code. Run it before you run anything else
#
data_generating_model <- function(n,w) {
  z <- rep( NaN, times=n*3 )
  z <- matrix(z, nrow = n, ncol = 3)
  z[,1] <- rep(1,times=n)
  z[,2] <- runif(n, min = -10, max = 10)
  p <- w[1]*z[,1] + w[2]*z[,2] p <- exp(p) / (1+exp(p))
  z[,3] <- rbinom(n, size = 1, prob = p)
  ind <- (z[,3]==0)
  z[ind,3] <- -1
  x <- z[,1:2]
  y <- z[,3]
  return(list(z=z, x=x, y=y))
}
n_obs <- 1000000
w_true <- c(-3,4)
set.seed(2023)
out <- data_generating_model(n = n_obs, w = w_true)
set.seed(0)
z_obs <- out$z #z=(x,y)
x <- out$x
y <- out$y
#z_obs2=z_obs
#z_obs2[z_obs[,3]==-1,3]=0
#w_true <- as.numeric(glm(z_obs2[,3]~ 1+ z_obs2[,2],family = "binomial"
)$coefficients)

```

**Solution.**

- (1)  $f_1(x) = 0$  is convex,  $f_2(x) = 1 - x$  is convex, hence from the example in Handout 1,  $f(x) = \max(f_1(x), f_2(x))$  is convex as well. Regarding the loss function, we just have  $f_2(w) = 1 - yx^\top w$  which is convex as a composition due to linearity.
- (2) Given a fixed example  $(x, y) \in \{x \in \mathbb{R}^d : \|x'\|_2 \leq R\} \times \{-1, 1\}$ .

Assume  $w_1, w_2 \in \mathbb{R}^d$ . Let  $\ell_i = \max\{0, 1 - yx^\top w_i\}$ , for  $i = 1, 2$ . It suffices to show that  $|\ell_1 - \ell_2|_2 \leq R|w_1 - w_2|_2$ . I take cases

**Case-1:** Assume  $yx^\top w_1 \geq 1$  and  $yx^\top w_2 \geq 1$  then  $|\ell_1 - \ell_2|_2 = 0 \leq R|w_1 - w_2|_2$

**Case-2:** Assume that at least one of  $yx^\top w_1 < 1$  or  $yx^\top w_2 < 1$  but not both is true.

Assume without loss of generality that  $1 - yx^\top w_1 < 1 - yx^\top w_2$ . Then

$$\begin{aligned}
|\ell_1 - \ell_2|_2 &= \ell_1 - \ell_2 \\
&= 1 - yx^\top w_1 - \max(0, 1 - yx^\top w_2) \\
&\leq 1 - yx^\top w_1 - (1 - yx^\top w_2) \\
&= yx^\top (w_2 - w_1) \\
&\leq y \left\| x^\top \right\|_2 \|w_1 - w_2\|_2 \quad \text{because} \quad a^\top b \leq \|a\| \|b\|
\end{aligned}$$

(3) It is

$$f(x) = \max(0, 1 - x) = \begin{cases} 0 & x > 1 \\ 0 & x = 1 \\ 1 - x & x < 1 \end{cases}$$

- For  $x > 1$ ,  $f$  is differentiable so  $\partial f(x) = \{f'(x)\} = \{0\}$ .
- For  $x < 1$ ,  $f$  is differentiable so  $\partial f(x) = \{f'(x)\} = \{-1\}$ .
- For  $x = 1$ ,  $f$  is not differentiable. By definition I have that  $v$  is subgradient of  $f(x)$  at  $x = 0 \in S$  if

$$\forall u \in \mathbb{R}, \quad f(u) \geq f(x) + \langle u - x, v \rangle$$

So, for  $u \geq 1$ , it is  $0 \geq (u - 1)v \implies v \leq 0$ , and for  $u < 1$  it is  $(1 - u) \geq (u - 1)v \implies v \geq -1$ . Hence the common space is  $v \in [0, 1]$  So  $\partial f(x) = [0, 1]$ . Hence,

$$\partial f(x) = \begin{cases} 0, & x > 1 \\ [-1, 0], & x = 1 \\ -1, & x < 1 \end{cases}$$

Now regarding the loss  $\partial_w \ell(w, z = (x, y))$

- for  $yw^\top x > 1$  it is differentiable so  $\nabla_w \ell(w, z = (x, y)) = \nabla_w (0 + \lambda \sum_{j=1}^d w_j^2) = 2\lambda w$ ;  
as

$$\frac{d}{dw_j} \sum_{j'=1}^d w_{j'}^2 = 2\lambda w_j$$

- for  $yw^\top x < 1$  it is differentiable so  $\nabla_w \ell(w, z = (x, y)) = \nabla_w (1 - yw^\top x + \lambda \sum_{j=1}^d w_j^2) = yx + 2\lambda w$  as

$$\frac{d}{dw_j} (1 - yw^\top x) = \frac{d}{dw_j} \left( 1 - y \sum_{j'=1}^d w_{j'} x_{j'} \right) = -yx_j$$

- for  $yw^\top x = 1$ ,  $v = 0$  satisfies the definition of the sub-gradient

$$\begin{aligned} \forall u, \quad f(u) &\geq \cancel{f(w)}^0 + \langle u - w, v \rangle \\ \max(0, 1 - yu^\top x) &\geq 0 + (u - w)^\top 0 \end{aligned}$$

So

$$\begin{aligned} \partial \ell(w, z = (x, y)) &= \partial \left( \max(0, 1 - yw^\top x) + \lambda \|w\|_2^2 \right) \\ &= \partial \left( \max(0, 1 - yw^\top x) \right) + \partial \left( \lambda \|w\|_2^2 \right) \\ &= \partial \left( \max(0, 1 - yw^\top x) \right) + \nabla \left( \lambda \|w\|_2^2 \right) \\ &= 0 + 2\lambda w \end{aligned}$$

but  $\partial \left( \lambda \|w\|_2^2 \right) = \left\{ \nabla \left( \lambda \|w\|_2^2 \right) \right\}$  because  $\lambda \|w\|_2^2$  is differentiable. Hence

$$\partial \ell(w, z = (x, y)) = 0 + 2\lambda w$$

Hence

$$v = \begin{cases} 2\lambda w, & yw^\top x > 1 \\ 2\lambda w, & yw^\top x = 1 \\ -yx + 2\lambda w, & yw^\top x < 1 \end{cases}$$

(4)

**Algorithm.** For  $t = 1, 2, 3, \dots$  iterate:

- Get a random sub-sample  $\left\{ \tilde{z}_i^{(t)} = (\tilde{x}_i^{(t)}, \tilde{y}_i^{(t)}) ; i = 1, \dots, m \right\}$  of size  $m$  with or without replacement from the complete data-set  $\mathcal{S}_n$ .
- For  $j = 1, \dots, d$  (index  $j$  indicates the dimension of  $w$ ) compute

$$w_j^{(t+1)} = w_j^{(t)} - \eta_t \frac{1}{\sqrt{[G_t]_{j,j} + \epsilon}} \bar{v}_{t,j}$$

$[G_t]_{j,j} = [G_{t-1}]_{j,j} + (\bar{v}_{t,j})^2$  where  $\bar{v}_t = \frac{1}{m} \sum_{i=1}^m \tilde{v}_{t,i}$  and

$$\tilde{v}_{t,i} = \begin{cases} 2\lambda w^{(t)}, & \tilde{y}_i^{(t)} (w^{(t)})^\top \tilde{x}_i^{(t)} > 1 \\ 2\lambda w^{(t)}, & \tilde{y}_i^{(t)} (w^{(t)})^\top \tilde{x}_i^{(t)} = 1 \\ -\frac{1}{m} \tilde{y}_i^{(t)} \tilde{x}_i^{(t)} + 2\lambda w^{(t)}, & \tilde{y}_i^{(t)} (w^{(t)})^\top \tilde{x}_i^{(t)} < 1 \end{cases}$$

where index  $i$  indicates the sub-sample, and  $\epsilon > 0$  small.

- Terminate if a termination criterion is satisfied

(5)

- The R code can be found in the link [https://raw.githubusercontent.com/georgios-stats/Machine\\_Learning\\_and\\_Neural\\_Networks\\_III\\_Epiphany\\_2024/main/Exercises/supplementary/q6\\_adagrad.R](https://raw.githubusercontent.com/georgios-stats/Machine_Learning_and_Neural_Networks_III_Epiphany_2024/main/Exercises/supplementary/q6_adagrad.R)
- The figures are presented below

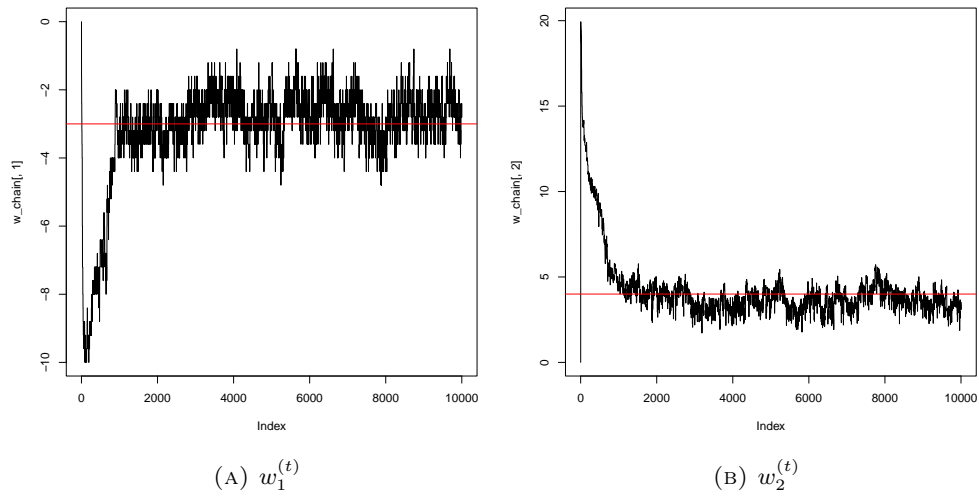


FIGURE 0.1. trace plots

- (c) I found  $w = (-2.674615, 3.205785)$
- (d) It belongs to  $-1$