Linear classification and optimisation

Machine Learning II (2022-2023)
UMONS

1 LAB 1

1.1 Exercise 1

Given a matrix $A \in \mathbb{R}^{m \times n}$ and a differentiable function $f : \mathbb{R}^m \to \mathbb{R}$. Let $h : \mathbb{R}^n \to \mathbb{R}$ defined by $h(x) = f(Ax)$. Find $\nabla h(x)$ and $\nabla^2 h(x)$.

1.2 Exercise 2

Consider pointwise error measures $e_{class}(s, y) = [\![y \neq sign(s)]\!]$, $e_{sq}(s, y) = (y - s)^2$, and $e_{log}(s, y) = \ln(1 + \exp(-ys))$, where the signal $s = \mathbf{w}^T\mathbf{x}$.

(a) For $y = +1$, plot e_{class}, e_{sq} and $\frac{1}{\ln 2}e_{log}$ versus s, on the same plot.

(b) Show that $e_{class}(s, y) \leq e_{sq}(s, y)$, and hence that the classification error is upper bounded by the squared error.

(c) Show that $e_{class} \leq \frac{1}{\ln 2}e_{log}(s, y)$, and, as in part (b), get an upper bound (up to a constant factor) using the logistic regression error.

These bounds indicate that minimizing the squared or logistic regression error should also decrease the classification error, which justifies using the weights returned by linear or logistic regression as approximations for classification.

1.3 Exercise 3

The output of the final hypothesis $g(\mathbf{x})$ learned by a probabilistic classifier can be thresholded to get a 'hard' (± 1) classification. The problem shows how to use a risk matrix to obtain such a threshold.

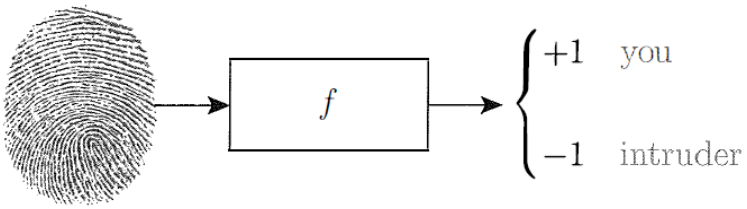

Figure 1: Fingerprint verification (Abu-Mostafa et al., 2012)

Consider fingerprint verification problem as shown in Figure 1. After learning by a probabilistic classifier from the data, you produce the final hypothesis

$$g(\mathbf{x}) = \mathbb{P}[y = +1|\mathbf{x}],$$

which is your estimate of the probability that $y = +1$. Suppose that the cost matrix is given by

		True classification	
		$+1$ (correct person)	-1 (intruder)
you say	$+1$	0	c_a
	-1	c_r	0

For a new person with fingerprint $\mathbf{x}$, you can compute $g(\mathbf{x})$ and you now need to decide whether to accept or reject the person (i.e., you need a hard classification). So, you will accept if $g(\mathbf{x}) \geq \kappa$ where κ is the threshold.

(a) Define the cost(accept) as your expected cost if you accept the person. Similarly define cost(reject). Show that

$$\begin{aligned} \text{cost(accept)} &= (1 - g(\mathbf{x}))c_a, \\ \text{cost(reject)} &= g(\mathbf{x})c_r. \end{aligned}$$

(b) Use part (a) to derive a condition on $g(\mathbf{x})$ for accepting the person and hence show that

$$\kappa = \frac{c_a}{c_a + c_r}$$

(c) Now, consider two potential clients of this fingerprint system. One is a supermarket who will use it at the checkout counter to verify that you are a member of a discount program. The other is the CIA who will use it at the entrance to a secure facility to verify that you are authorized to enter that facility.

For the supermarket, a false reject is costly because if a customer gets wrongly rejected, she may be discouraged from patronizing the supermarket in the future. On the other hand, the cost of a false accept is minor. You just gave away a discount to someone who didn't deserve it.

For the CIA, a false accept is a disaster. An unauthorized person will gain access to a highly sensitive facility. This should be reflected in a much higher cost for the false accept. False rejects, on the other hand, can be tolerated since authorized persons are employees.

The costs of the different types of errors can be tabulated in a matrix. For our examples, the matrices might look like:

Table 1: Supermarket

		f	
		$+1$	-1
g	$+1$	0	1
	-1	10	0

Table 2: CIA

		f	
		$+1$	-1
g	$+1$	0	1000
	-1	1	0

Now, compute threshold κ for each of these two cases. Give some intuition for the thresholds you get.

1.4 Exercise 4

For logistic regression,

$$E_{in}(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i}).$$

show that

$$\begin{aligned}\nabla E_{\text{in}}(\mathbf{w}) &= -\frac{1}{n}\sum_{i=1}^{N} \frac{y_i \mathbf{x}_i}{1 + e^{y_i \mathbf{w}^T \mathbf{x}_i}} \\ &= \frac{1}{n}\sum_{i=1}^{n} -y_i \mathbf{x}_i \theta(-y_i \mathbf{w}^T \mathbf{x}_i)\end{aligned}$$

and find the Hessian of $E_{in}(\mathbf{w})$.

1.5 Exercise 5

Find all $a \in \mathbb{R}$ such that $f : \mathbb{R}^2 \to \mathbb{R}$ defined by $f(x, y) = |xy| + a(x^2 + y^2)$ is convex.

1.6 Exercise 6

Let $f(x) = \max\{f_1(x), f_2(x)\}$, where $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$ are differentiable convex functions. Prove that

- If $f_1(x) > f_2(x)$, f has unique subgradient $v = \nabla f_1(x)$
- If $f_2(x) > f_1(x)$, f has unique subgradient $v = \nabla f_2(x)$.
- If $f_1(x) = f_2(x)$, then any point on the line segment between $\nabla f_1(x)$ and $\nabla f_2(x)$ is a subgradient of f at x.

2 LAB 2

2.1 Exercise 1

For logistic regression,

$$E_{in}(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} \ln(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i}).$$

Prove that E_{in} is an L-smooth function and determine L.

2.2 Exercise 2

For logistic regression, show that the Newton's weight update can be rewritten as the solution of a weighted least square problem.

2.3 Exercise 3

Suppose f is convex and L-smooth. Consider the gradient descent update

$$x^{k+1} = x^k + \alpha \nabla f(x^k), \text{for } k = 0, 1, \ldots,$$

where $0 < \alpha < \frac{2}{L}$. The following inequality holds for all $k \geq 0$

$$f(x^k) - f(x^*) \leq \frac{2(f(x^0) - f(x^*)\|x^0 - x^*\|^2}{2\|x^0 - x^*\|^2 + k\alpha(2 - L\alpha)(f(x^0) - f(x^*)}.$$

Consequently, if $\alpha = \frac{1}{L}$ then $f(x^k) - f(x^*) \leq \frac{2L\|x^0 - x^*\|^2}{k+4}$.

This is Theorem 2.1.14 in Y. Nesterov, Lectures on Convex Optimization, Springer Optimization and Its Applications book series, 2018. Please read and write again the proof.

Hint

Step 1: prove

$$\frac{1}{L}\|f(x) - \nabla f(y)\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle.$$

Step 2: Let $r_k = \|x_k - x^*\|$. Prove

$$r_{k+1}^2 \leq r_k^2 - h(\frac{2}{L} - h)\|\nabla f(x^k)\|^2.$$

Step 3: Let $\Delta_k = f(x^k) - f^*$. Prove

$$\Delta_{k+1} \leq \Delta_k - \frac{\omega}{r_0^2}\Delta_k^2,$$

where $\omega = h(1 - \frac{L}{2}h)$.

2.4 Exercise 4

Find prox_f with $f : \mathbb{R}^n \to \mathbb{R}, f(x) = t\|x\|_2^2$ and $t > 0$.

2.5 Exercise 5

Consider a function

$$E(u, v) = e^u + e^{2v} + e^{uv} + u^2 - 3uv + 4v^2 - 3u - 5v,$$

(a) Approximate $E(u + \Delta u, v + \Delta v)$ by $\hat{E}_1(\Delta u, \Delta v)$, where $\hat{E}_1$ is the first-order Taylor's expansion of E around $(u, v) = (0, 0)$. Suppose $\hat{E}_1(\Delta u, \Delta v) = a_u \Delta u + a_v \Delta v + a$. What are the values of a_u, a_v, and a?

(b) Minimize $\hat{E}_1$ over all possible $(\Delta u, \Delta v)$ such that $||\Delta u, \Delta v|| = 0.5$. In this chapter, we proved that the optimal column vector $\begin{pmatrix} \Delta u \\ \Delta v \end{pmatrix}$ is parallel to the column vector $-\nabla E(u, v)$, which is called the *negative gradient direction*. Compute the optimal $(\Delta u, \Delta v)$ and the resulting $E(u + \Delta u, v + \Delta v)$.

(c) Approximate $E(u + \Delta u, v + \Delta v)$ by $\hat{E}_2(\Delta u, \Delta v)$, where $\hat{E}_2$ is the second order Taylor's expansion of E around $(u, v) = (0, 0)$. Suppose

$$\hat{E}_2(\Delta u, \Delta v) = b_{uu}(\Delta u)^2 + b_{vv}(\Delta v)^2 + b_{uv}(\Delta u)(\Delta v) + b_u \Delta u + b_v \Delta v + b.$$

What are the values of b_{uu}, b_{vv}, b_{uv}, b_u, b_v, and b?

(d) Minimize $\hat{E}_2$ over all possible $(\Delta u, \Delta v)$ (regardless of length). Use the fact that $\nabla^2 E(u, v)|_{(0,0)}$ (the Hessian matrix at $(0, 0)$) is positive definite to prove that the optimal column vector

$$\begin{pmatrix} \Delta u^* \\ \Delta v^* \end{pmatrix} = -(\nabla^2 E(u, v))^{-1} \nabla E(u, v)$$

which is called the *Newton direction*.

(e) Numerically compute the following values:

(i) the vector $(\Delta u, \Delta v)$ of length 0.5 along the Newton direction, and the resulting $E(u + \Delta u, v + \Delta v)$.

(ii) the vector $(\Delta u, \Delta v)$ of length 0.5 that minimizes $E(u + \Delta u, v + \Delta v)$, and the resulting $E(u + \Delta u, v + \Delta v)$. [**Hint:** Let $\Delta u = 0.5 \sin \theta$]

Compare the values of $E(u + \Delta u, v + \Delta v)$ in (b), (e i), and (e ii)

2.6 Exercise 6

Consider the regularized linear regression problem with Lasso (least absolute shrinkage and selection operator)

$$\min_{x\in\mathbb{R}^n} \frac{1}{2n}\|Ax-b\|_2^2 + \lambda\|x\|_1.$$

Task:

- Implement proximal gradient descent method

  $$x^{k+1} = \text{prox}_{\alpha_k g}(x^k - \alpha_k \nabla f(x^k)),$$

  where $g(x) = \lambda\|x\|_1$ and $f(x) = \frac{1}{2n}\|Ax-b\|_2^2$. Note that the soft-thresholding operator is calculated by

  $$\text{prox}_{\gamma||\cdot||_1}(v) = \text{sign}(v)[|v|-\gamma]_+ = \text{sign}(v)\max\{|v|-\gamma, 0\}.$$

- By using different step-sizes, test the algorithm on the blog feedback data set https://archive.ics.uci.edu/ml/machine-learning-databases/00304/

Follow the instruction in this Python notebook template https://colab.research.google.com/drive/1LewtLVL_2Lex1JHCDY56j1l0l6di-A5U?usp=sharing

3 LAB 3

3.1 Exercise 1

(a) Define an error for a single data point $(\mathbf{x}_i, y_i)$ to be

$$e_i(\mathbf{w}) = \max(0, -y_i \mathbf{w}^T \mathbf{x}_i).$$

Argue that PLA can be viewed as a stochastic subgradient method on $e(w) = \sum_{i=1}^{n} e_i(w)$ with learning rate $\eta = 1$.

(b) For logistic regression with a very large $\mathbf{w}$, argue that minimizing E_{in} using SGD is similar to PLA. This is another indication that the logistic regression weights can be used as a good approximation for classification.

3.2 Exercise 2 - Effect of the batch size

Consider the finite sum problem

$$\min_x F(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x).$$

Standard stochastic gradient for solving the finite sum problem

$$g_s(x^k, \xi^k) = \nabla f_{i_k}(x^k), \quad x^{k+1} = x^k - \alpha_k g_s(x^k, \xi^k).$$

Mini-batch stochastic gradient for solving the finite sum problem

$$g_m(x^k, \xi^k) = \frac{1}{|S^k|}\sum_{j \in S^k} \nabla f_j(x^k), \quad x^{k+1} = x^k - \alpha_k g_m(x^k, \xi^k).$$

Compare the variance $var(g_s(x^k, \xi^k))$ of the standard stochastic gradient with the variance of the mini-batch stochastic gradient. Comment on the effect of the batch size $|S^k|$.

3.3 Exercise 3

Adaline (Adaptive Linear Neuron) algorithm for classification works like this: In each iteration, pick a random $(\mathbf{x}(t), y(t))$ and compute the 'signal' $s(t) = \mathbf{w}^T(t)\mathbf{x}(t)$. If $y(t) \cdot s(t) \leq 1$, update $\mathbf{w}$ by

$$\mathbf{w}(t+1) \leftarrow \mathbf{w}(t) + \eta \cdot (y(t) - s(t)) \cdot \mathbf{x}(t),$$

where η is a constant. That is, if $s(t)$ agrees with $y(t)$ well (their product is > 1), the algorithm does nothing. On the other hand, if $s(t)$ is further from $y(t)$, the algorithm changes $\mathbf{w}(t)$ more. Here, we derive Adaline from an optimisation perspective.

(a) Consider $e_i(\mathbf{w}) = (\max(0, 1 - y_i \mathbf{w}^T \mathbf{x}))^2$. Write down the gradient $\nabla e_i(\mathbf{w})$.

(b) Show that $e_i(\mathbf{w})$ is an upper bound for $[\![sign(\mathbf{w}^T \mathbf{x}_i) \neq y_i]\!]$. Hence, $\frac{1}{n} \sum_{i=1}^{n} e_n(\mathbf{w})$ is an upper bound for the in sample classification error $E_{in}(\mathbf{w})$.

(c) Argue that the Adaline algorithm performs stochastic gradient descent on $\frac{1}{n} \sum_{i=1}^{n} e_i(\mathbf{w})$.

3.4 Exercise 4

Consider the regularized linear regression problem with Lasso.

Task:

- Implement FISTA
- Observe the acceleration effect of FISTA when compared with PG.

Follow the instruction in this Python notebook template

https://colab.research.google.com/drive/1FRFtoF5GWrr46mgz8UdZj0Pf7DV5jR2K?usp=sharing

3.5 Exercise 5 - Stochastic gradient descent

Consider the logistic regression

$$\frac{1}{n}\sum_{i=1}^{n}\ln(1+e^{-y_i w^\top x_i}).$$

Task:

- Implement minibatch stochastic gradient method.
- Test the algorithm on the Gisette dataset https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary/gisette_scale.bz2

  Observe the performance of standard SG with constant step-sizes and diminishing step-sizes
- Observe the effect of the batch size.

Follow the instruction in this Python notebook template https://colab.research.google.com/drive/10vDCJdYI6UFz_dChaQmf9tM9DrixVmhi?usp=sharing

Some of the questions in this lab are based on Abu-Mostafa et al., 2012.

References

Abu-Mostafa, Y. S., Magdon-Ismail, M., & Lin, H.-T. (2012). *Learning from data*. AMLBook.