# MATH 462 LECTURE NOTES

ADAM M. OBERMAN

## 1. Review of vector calculus

We reviewed [DFO20, Chapter 5]

- §5.1 Difference Quotient, definition of the derivative as a limit, Taylor polynomial, differentiation rules
- Example: use order 1 Taylor approximation near $x = 9$ to estimate $\sqrt{9.1}$.
- §5.2 Partial differentiation.
- §5.3 Jacobian, §5.4 gradient of Matrix.
- To know: when $f = Mx$, $\nabla_x f = Jf = M$.
- Product rule $f = g^\top h$, Then $\nabla_x f = Jgh + Jhg$ (can apply this to regression loss below).

Critical points for a function of one variable $L(w)$, $w \in \mathbb{R}$,

- a critical point is when $L'(w) = 0$
- Every local minimum is a critical point. A critical point can be a local minimum, local maximum, or saddle point.
- If the second order condition holds $L''(w) > 0$, then the critical point is also a local minimum
- If the function is convex (for example when $L''(w) \geq 0$ at all $w$), then every critical point is a *global* minimum.

### 1.1. **Vector Calc for ML: losses.**

*Example* 1.1 (one dimensional MSE loss). Consider a typical Mean Squared Error (MSE) loss. Let $w, x_i \in \mathbb{R}$, define

$$\widehat{L}(w) = f(w, x_1, \ldots, x_m) = \frac{1}{m}\sum_{i=1}^{m}(w - x_i)^2$$

We showed that

$$\frac{\partial}{\partial w} f(w, x_1, \ldots, x_m) = 2w - 2\bar{x}, \quad \bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_i$$

and

$$w^* = \arg\min_w \widehat{L}(w) = \bar{x}, \qquad \widehat{L}(w^*) = \sum_{i=1}^{m}(\bar{x} - x_i)^2$$

We interpreted the second equation as the variance of the dataset. We also showed

$$\frac{\partial}{\partial x_1} f(w, x_1, \ldots, x_m) = \frac{2}{m}(x_1 - w)$$

*Date*: September 26, 2023.

1.2. **Vector calculus facts.** Now consider a function of $d$ variables, $L : \mathbb{R}^d \to \mathbb{R}$. The gradient of the function is a vector defined at each $w$,

$$g(w) = \nabla L(w) = [g_1(w), \dots g_d(w)]^T$$

where each component is partial derivative

$$g_j(w) = \frac{\partial}{\partial w_j} L(w)$$

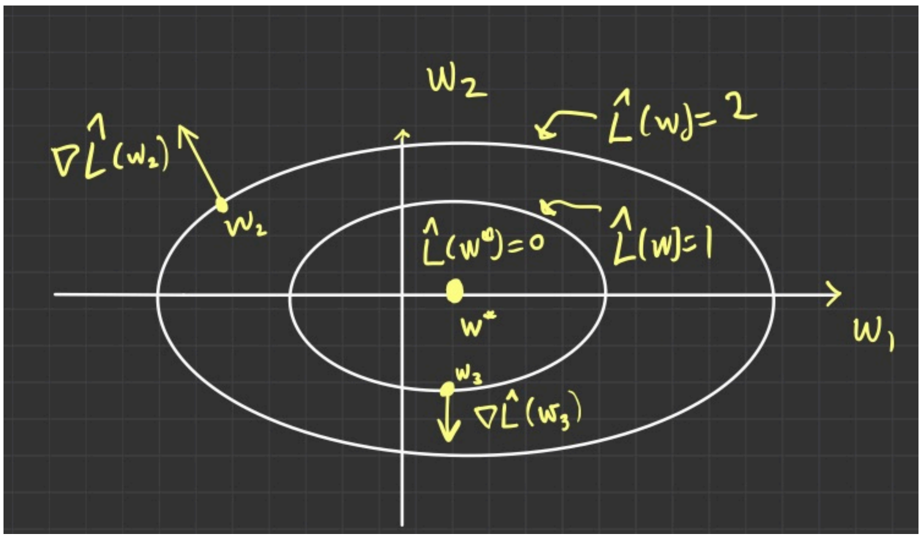


FIGURE 1. Illustration of the gradient of the loss

- The gradient vector $g(w) = \nabla L(w)$ points in the direction of greatest increase of the function $L$ at $w$.
- A critical point $w$ is a point where $g(w) = 0$.
- As in the one variable case, every local minimum is a critical point. A critical point can be a local minimum, local maximum, or saddle point.
- As in the one variable case, there is a condition for a critical point to be a local minimum: the Hessian matrix $H(w)$ is positive-definite. Here $H(w)_{ij} = \frac{\partial^2}{\partial_i \partial_j} L$. (This condition can be difficult to check).
- As in the one variable case, if the function is convex, then every critical point *global* minimum.

## 2. Linear Regression

We are given the dataset

$$(S_m) \qquad S_m = \{(x_1, y_1), \dots, (x_m, y_m)\}$$

Consisting of pairs of vectors $x_i \in \mathbb{R}^d$ and values $y_i \in \mathbb{R}$, for $i = 1, \dots, m$.

Our goal is to *fit* the dataset using linear models $h : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{H} = \{h_w(x) : \mathbb{R}^d \to \mathbb{R} \mid w \in \mathbb{R}^d\}$$

$$h_w(x) = w^T x = \sum_{i=1}^{d} w_i x_i$$

The error of the model, $h_w$ on data $(x, y)$, is defined to be

$$e = h_w(x) - y$$

We measure the error with the squared loss,

$$\ell_2(e) = e^2$$

**Definition 2.1** (General Empirical Loss)**.** Given

(1) the dataset $S_m$, as in ($S_m$),
(2) a model $h : \mathbb{R}^d \to \mathbb{R}$,
(3) the non-negative loss, $\ell : \mathbb{R} \to \mathbb{R}$

The empirical loss of the model $h$, on the dataset $S_m$, is given by

$$L(h) = L(h, S^m) = \frac{1}{m} \sum_{i=1}^{m} \ell(h(x_i) - y_i)$$

Given the hypothesis class $\mathcal{H}$, the empirical loss minimizer is given by

$$h^* = \arg\min_{h \in \mathcal{H}} L(h) \tag{ELM-h}$$

Note (ELM-h) is a minimization over *functions*. However, when the functions are parameterized by $w$, we can reduce this to a minimization over the *parameters* as given by (ELM-w)

$$w^* = \arg\min_{w} L(h_w) \tag{ELM-w}$$

We can apply the chain rule to (ELM-w) to find a critical point

$$\nabla_w L(h_w) = \frac{1}{m} \sum_{i=1}^{m} \ell'(h_w(x_i) - y_i) \nabla_w h_w(x_i) \tag{grad L}$$

So we can interpret each conmponent of the loss gradient as *the function gradient multiplied by the loss derivative*

## 2.1. **Gradient of a Least Squares Loss with Linear Model.**

In this case of the least squares loss,

$$L(w) = \frac{1}{m} \sum_{i=1}^{m} (h_w(x_i) - y_i)^2$$

Since

$$\ell_2(e) = e^2, \quad \ell_2'(e) = 2e$$

and with a linear model

$$h_w(x) = w^T x, \quad \nabla_w h_w(x_i) = x_i^T$$

(note the transpose). So (grad L) becomes

$$\nabla_w L(w) = \frac{1}{m}\sum_{i=1}^{m}(h_w(x_i) - y_i)\nabla_w h_w(x_i).$$

Which we can rewrite as

$$\nabla_w L(w) = \frac{1}{m}\sum_{i=1}^{m}(w^T x_i - y_i)x_i^T = \frac{1}{m}w^T\sum_{i=1}^{m}x_i x_i^T - \frac{1}{m}\sum_{i=1}^{m}y_i x_i^T$$

## 3. Vector calculus

Recall from vector calculus, `https://en.wikipedia.org/wiki/Gradient`.

(1) $x$ is a $d$-dimensional column vector,
(2) $f : \mathbb{R}^d \to R$, Then $\nabla f : \mathbb{R}^d \to \mathbb{R}^d$, $\nabla f(x)$ is also a column vector. The reason for this is we want to generalize the derivative: $f(x+h) \approx f(x) + hf'(x)$ becomes:

$$f(x+hv) \approx f(x) + h\nabla f(x)\cdot v$$

For the equation above to make sense, we need $\nabla f$ to be a column vector. (The total derivative $df = \nabla f^\top$ is a row vector, see, `https://en.wikipedia.org/wiki/Gradient` total derivative.)
(3) If $g : \mathbb{R}^d \to \mathbb{R}^n$ (the function is a column vector), then the jacobian, $Jg : \mathbb{R}^d \to \mathbb{R}^n$, is the matrix of partial derivatives,

$$(Jg)_{ij} = \frac{\partial g_i}{\partial x_j}$$

Each row of the jacobian, $Jg$, is the gradient transpose $(\nabla g_i)^\top$ of $g_i$.
(4) In particular, $g(x) = Mx$, then $Jg = M$. (Exercies: Verify the last statement)
(5) The dot product rule: for vector-valued functions $g(x), h(x) : \mathbb{R}^d \to \mathbb{R}^n$,

$$\nabla(g(x)^\top h(x)) = (Jg)^\top h + (Jh)^\top g$$

(6) Using these rules allows us to differentiate $f(x) = \|Mx - b\|^2 = (Mx-b)\cdot(Mx-b)$.

$$\nabla f = 2M^\top(Mx - b)$$

### 3.1. **Matrix vector notation.**

We can simplify this expression using matrix vector notation. This notation is also more compatible with vector programming languages. See also [DFO20, Example 5.11]

Given the dataset $S^m$, with each component $(x_i, y_i)$ consisting of a row vector and a real, we want to extract matrices and vectors from it as follows.

$$X = X(S^m) = [x_1, \ldots, x_m]^\top = \begin{bmatrix} x_1^\top \\ x_2^\top \\ \vdots \\ x_m^\top \end{bmatrix}, \quad X \in \mathbb{R}^{m\times d}$$

and

$$y = [y_1, \ldots, y_m]^T = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad y \in \mathbb{R}^{m\times 1}$$

Here $X$ has $m$ rows, and each row is a vector in $\mathbb{R}^d$ and $y$ is a column vector.

3.2. **Matrix vector notation for quadratic regression.** For the linear function $h_w(x) = w \cdot x$, writing $h_w(x) = x^\top w$, then the function values can be written as the matrix vector product,

$$h = h(X) = Xw.$$

With quadratic loss, we write

$$L(w) = \frac{1}{m}\|Xw - y\|^2$$

Then we have

$$\nabla_w L(w) = \frac{2}{m}(X^\top Xw - X^\top y)$$

so the minimizer satisfies the linear equation

$$X^\top Xw = X^\top y$$

or $w = (X^\top X)^{-1}X^\top y$. Then the function values are

$$h = Xw = X(X^\top X)^{-1}X^\top y$$

*Remark* 3.1. The formulas above looks complicated at first glance. However, there is a geometrical interpretation in terms of projection. `https://en.wikipedia.org/wiki/Projection_matrix`

- When there is a solution $Xw = y$, this corresponds to writing $y$ as a linear combination of the $x_i$ vectors, so $h = y$.
- When there is no solution $Xw = y$, then $h = Xw$ corresponds to the projection of the values $y$ onto the span of the $x_i$.

3.3. **Gradient of general loss with Linear Model.** The gradient of a general loss (grad L) can also be written in matrix-vector notation.

Define the column vector

$$(\ell_h)_i = \frac{\partial}{\partial h}\ell(h_i, y_i), \quad i = 1, \dots, m.$$

and recall that $\nabla_w h = X$, which corresponds to $(\nabla_w h)_i = x_i^\top$.

Then (grad L) becomes

$$\nabla_w L(w) = \frac{1}{m}X\ell_h \tag{1}$$

So we can interpret each conmponent of the loss gradient as *the function gradient multiplied by the loss derivative*

*Remark* 3.2. Note that for the least squares loss, (1) corresponds to $\nabla_w L(w) = \frac{2}{m}Xe$, where $e_i = h_i - y_i$.

## References

[DFO20] Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. *Mathematics for machine learning*. Cambridge University Press, 2020.