CS 412 — Introduction to Machine Learning (UIC) February 13, 2025

Lecture [8]

Instructor: Aadirupa Saha *Scribe(s): Farhad Ferdowsi*

[This draft is not fully proofread. Please email any typos/errors to the instructor or directly edit the latex file.]

Overview

In the last lecture, we covered the following main topics:

1. Quick review of Bernoulli, Binomial, Categorical, and Multinomial Distribution
2. MLE interpretation of logistic regression
3. MAP interpretation on regularized logistic regression
4. Convex functions

This lecture focuses on:

1. Convexity
2. Key properties of a loss function that improve gradient descent convergence.
3. Introduction to Gradient Descent

Gradient descent is one of the most effective methods to train a machine learning model by minimizing the model's loss function. However, this method only works well if certain conditions are met and convexity of the loss function is one of these conditions. Therefore, before explaining the gradient descent method, we need to elaborate on convexity, the conditions for a function to be convex, and the properties of convex functions.

1 Convex functions

A function is called convex on an interval if, for any two points X_1 and X_2 in that interval, the line segment connecting the points $(X_1, f(X_1))$ and $(X_2, f(X_2))$ lies above or on the graph of the function between those points. This intuitive behavior can be expressed in more precise ways as well. In what follows, we first introduce the conditions for convexity and discuss how to determine whether a given function is convex. A function is convex if it satisfies one of the following conditions. It is important to note that satisfying any one of these conditions is sufficient to ensure the convexity of a function. In other words, if one of these conditions is satisfied for a specific function, the other conditions are also automatically satisfied.

1.1 Convexity conditions

Condition 1.1: Jensen's inequality

A function $f : D \to \mathbb{R}$ is convex in a specific domain $D \subseteq \mathbb{R}^d$, if for every $X_1, X_2 \in D$ and $0 \leq \lambda \leq 1$ the following inequality holds:

$$f(\lambda X_1 + (1-\lambda)X_2) \leq \lambda f(X_1) + (1-\lambda)f(X_2) \tag{1}$$

and it is strictly convex, if for every $X_1, X_2 \in D$ and $0 \leq \lambda \leq 1$ the following inequality holds:

$$f(\lambda X_1 + (1-\lambda)X_2) < \lambda f(X_1) + (1-\lambda)f(X_2) \tag{2}$$

This condition is also known as the main definition of a convex function, as it is the mathematical expression of what we mentioned at the beginning about convex functions.

Condition 1.2: First derivative condition

A function $f : D \to \mathbb{R}$ is convex in a specific domain $D \subseteq \mathbb{R}^d$ if and only if f is differentiable in this domain and for all $X_1, X_2 \in D$ following inequality holds:

$$f(X_2) \geq f(X_1) + (\nabla f(X_1))^T.(X_2 - X_1) \tag{3}$$

$$\nabla f(X) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_d} \end{bmatrix} \tag{4}$$

This condition indicates that the graph of a convex function always lies above its supporting hyperplane (or tangent line in the case of a one-dimensional function) at any arbitrary point in its domain of convexity. For clarification, consider a convex function in one dimension ($d = 1$) fig. 1.

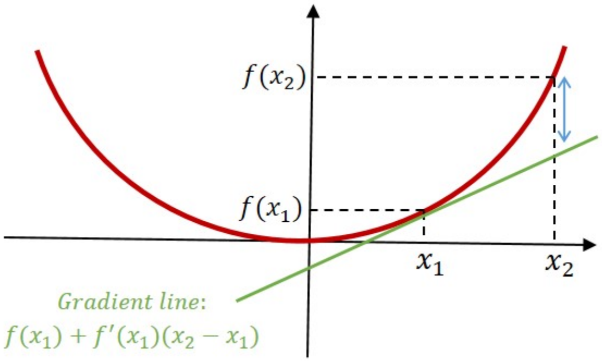

Figure 1: Convexity First derivative Condition for a one dimensional function

In this case gradient will be converted to simple derivative and convexity condition will be as follows:

$$f(X_2) \geq f(X_1) + f'(X_1).(X_2 - X_1). \tag{5}$$

Here, $f(X_1) + f'(X_1).(X_2 - X_1)$ represents the equation of the tangent line at X_1. An example for an one-dimensional convex function which satisfies this condition is illustrated in Figure 1, where the graph of the function lies above its tangent line at an arbitrary point X_1 in the domain.

Condition 1.3: Second derivative condition

A function $f : D \rightarrow \mathbb{R}$ is convex in a specific domain $D \subseteq \mathbb{R}^d$ if and only if f is twice-differentiable in this domain and for all $X \in D$, its Hessian matrix $H(f(X))$ (matrix of second derivatives) is positive semidefinite (PSD). The Hessian matrix is a matrix of all the second-order partial derivatives of a function $f(x_1, x_2, \ldots, x_d)$ with respect to the same variables and different ones.. It is given by:

$$H(f(X)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_d} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_d} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_d \partial x_1} & \frac{\partial^2 f}{\partial x_d \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_d^2} \end{bmatrix} \tag{6}$$

Note that a positive semidefinite (PSD) matrix is a matrix whose eigenvalues are all non-negative. In the case of one dimensional function ($d = 1$), Second derivative condition implies that function's second derivative is non zero (positive for strictly convex function) for all X in its domain of convexity

$$f''(X) \geq 0. \tag{7}$$

For example, the function $f(X) = X^2$ is convex because $f(X)'' = 2 \geq 0$ for all X, while $f(X) = X^3$ is not convex everywhere since $f(X)'' = 6X$ is not always non-negative.

Checking Convexity of some special functions using conditions 1, 2, and 3:

Example 1: Let $f(X) = W^T X + b$, where $X \in \mathbb{R}^d$

In order to verify condition 1 (Jensen's Inequality) for this function we begin with the left-hand side of the given equality:

$$f(\lambda X_1 + (1-\lambda)X_2) = W^T(\lambda X_1 + (1-\lambda)X_2) + b = \lambda W^T X_1 + (1-\lambda)W^T X_2 + (1-\lambda)b + \lambda b$$

$$= \lambda(W^T X_1 + b) + (1-\lambda)(X_2 + b) = \lambda f(X_1) + (1-\lambda)f(X_2)$$

Here X_1 and X_2 are any arbitrary points in $\mathbb{R}^n$. Therefor, it can be seen that Jensen's Inequality is satisfied. Convexity of this function can also be checked through other two conditions. In order to verify condition 2, starting from our given function:

$$f(X_2) = W^T X_2 + b = W^T X_2 + W^T X_1 - W^T X_1 + b = (W^T X_1 + b) + W^T(X_2 - X_1)$$

Since $\nabla f(X_1) = W$ and $f(X_1) = (W^T X_1 + b)$ we have:

$$f(X_2) = f(X_1) + (\nabla f(X_1))^T (X_2 - X_1)$$

Therefor, First derivative condition for convexity is satisfied. For verifying the third condition note that $\nabla f(X)$ is constant vector W. Therefore, the second derivative (the Hessian matrix) is a zero matrix:

$$\nabla^2 f(X) = 0$$

Since the Hessian is positive semi-definite (in this case, all zero entries), the function f is indeed convex.

Example 2: Let $f(X) = log(1 + e^{-X})$, where $X \in \mathbb{R}$

In this case for simplicity X is not a vector but a real number. To verify the third condition for convexity, we need to calculate the first and second derivatives of the function:

$$f'(X) = -\frac{e^{-X}}{1 + e^{-X}} = -\frac{1}{1 + e^{X}}$$

$$f''(X) = \frac{e^{X}}{(1 + e^{X})^2} > 0$$

Therefor, considering that $f''(X)$ is positive for all $X \in \mathbb{R}$, f is a convex function.

Exercise 1.1: Convexity

As homework, you can verify the convexity conditions for the following functions:

1. For graduate students: $f(X) = X^T H X$, where $X \in \mathbb{R}^d$ and H is a d-dimensional positive semidefinite matrix $H \in \mathbb{R}^{d \times d}$.

   For undergraduate Students: $f(X) = CX^2$, where $C > 0$ and X is a real number $X \in \mathbb{R}$.

2. $f(X) = log(X)$, where $X \in \mathbb{R}^d$

3. Check other two convexity conditions for $f(X) = log(1 + e^{-X})$, where $X \in \mathbb{R}$

1.2 Properties of convex functions

Let $f, g : \mathbb{R}^d \to \mathbb{R}$ be convex functions. The following properties hold:

Property 1.1: Property A

The sum of two convex functions is convex. Specifically, if

$$h(X) = f(X) + g(X), \tag{8}$$

then $h(X)$ is also a convex function.

Proof: Generally, we can use any of the three conditions of convexity. In this case, we will use the first condition since it is the simplest one to apply. Thus, we need to show that:

$$h(\lambda X_1 + (1-\lambda)X_2) \leq \lambda h(X_1) + (1-\lambda)h(X_2)$$

Starting from left-hand side:

$$h(\lambda X_1 + (1-\lambda)X_2) = f(\lambda X_1 + (1-\lambda)X_2) + g(\lambda X_1 + (1-\lambda)X_2)$$

Since f is convex, we have:

$$f(\lambda X_1 + (1-\lambda)X_2) \leq \lambda f(X_1) + (1-\lambda)f(X_2)$$

Similarly,since g is convex:

$$g(\lambda X_1 + (1-\lambda)X_2) \leq \lambda g(X_1) + (1-\lambda)g(X_2)$$

Combining these inequalities gives:

$$h(\lambda X_1 + (1-\lambda)X_2) \leq \lambda f(X_1) + (1-\lambda)f(X_2) + \lambda g(X_1) + (1-\lambda)g(X_2)$$

Rearranging terms, we get:

$$\lambda f(X_1) + (1-\lambda)f(X_2) + \lambda g(X_1) + (1-\lambda)g(X_2) = \lambda[f(X_1) + g(X_1)] + (1-\lambda)[f(X_1) + g(X_1)]$$

Thus, we have shown that:

$$\Rightarrow h(\lambda X_1 + (1-\lambda)X_2) \leq \lambda h(X_1) + (1-\lambda)h(X_2)$$

proving that $h(X)$ is indeed a convex function.

Property 1.2: Property B

The pointwise maximum of convex functions is convex. If

$$h(X) = \max\{f_1(X), f_2(X), \ldots, f_m(X)\}, \tag{9}$$

where each f_i is convex, then $h(X)$ is also convex.

Property 1.3: Property C

The composition of convex functions is convex when the outer function is convex and non-decreasing. If

$$h(X) = f(g(X)), \tag{10}$$

where $f : \mathbb{R} \to \mathbb{R}$ is convex and non-decreasing and g is convex, then $h(X)$ is convex. Without the non-decreasing condition on f, the convexity of $h(X)$ is not guaranteed.

Proof: In this case, we will use the first condition of convexity again. Thus, we need to show that:

$$h(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda h(X_1) + (1 - \lambda)h(X_2)$$

Starting from the left-hand side:

$$h(\lambda X_1 + (1 - \lambda)X_2) = f(g(\lambda X_1 + (1 - \lambda)X_2))$$

Since g is convex, we have:

$$g(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda g(X_1) + (1 - \lambda)g(X_2)$$

Because f is a non-decreasing function, it preserves:

$$f(a1) \leq f(a2) \qquad \forall a1, a2 \in \mathbb{R} \quad with \;\; a1 \leq a2$$

Applying this property to the previous result gives

$$f(g(\lambda X_1 + (1 - \lambda)X_2)) \leq f(\lambda g(X_1) + (1 - \lambda)g(X_2))$$

Now, considering convexity of f, we have::

$$f(\lambda g(X_1) + (1 - \lambda)g(X_2)) \leq \lambda f(g(X_1)) + (1 - \lambda)f(g(X_2)) = \lambda h(X_1) + (1 - \lambda)h(X_2)$$

Thus, we have shown that:

$$\Rightarrow h(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda h(X_1) + (1 - \lambda)h(X_2)$$

proving that $h(X)$ is indeed a convex function.

Property 1.4: Property D

If $f(X,Y)$ is convex in both X and Y meaning that for all $X_1, X_2 \in \mathbb{R}^d$ and $Y_1, Y_2 \in \mathbb{R}^d$ and $\lambda \in [0,1]$:

$$f(\lambda X_1 + (1-\lambda)X_2, \lambda Y_1 + (1-\lambda)Y_2) \leq \lambda f(X_1, Y_1) + (1-\lambda) f(X_2, Y_2). \tag{11}$$

then the function:

$$h(X) = \underset{Y}{min} f(X,Y) \tag{12}$$

is also convex. For more illustration the expression $h(X) = \underset{Y}{min} f(X,Y)$ means that for each fixed X, you are choosing the value of Y that minimizes $f(X,Y)$.

Property 1.5: Property E

The linear transformation of a convex function is convex. Specifically, if

$$h(X) = f(AX) \tag{13}$$

Where $X \in \mathbb{R}^d$,$A \in \mathbb{R}^{d' \times d}$, and $f : \mathbb{R}^{d'} \to \mathbb{R}$. then $h(X)$ is convex if f is convex. In other words, applying any linear transformation A to the input of a convex function f preserves convexity.
As an exercise, you can attempt to prove the other listed properties using the conditions of convexity or reprove the provided ones using different convexity conditions.

2 Some properties of a loss function that enhance the efficiency of gradient descent convergence

Convexity is an important property that makes it easier for the gradient descent algorithm to converge, but you should note that it does not guarantee convergence on its own. Other properties can further enhance the ease of convergence for gradient descent. Before explaining the gradient descent algorithm, it is essential to review these properties as well.

Property 2.1: Lipschitz continuity

We call a function $f : \mathbb{R}^d \to \mathbb{R}$, **Lipschitz continuous with constant L** (sometimes referred to as being L-Lipschitz continuous for simplicity) if there exists a constant $L \geq 0$ (called the Lipschitz constant of the function f) such that for all X_1 and X_2 in its domain:

$$|f(X_2) - f(X_1)| \leq L \|X_2 - X_1\|_p \tag{14}$$

Here p denotes the type of norm used.

Although any norm can be used in this property (as well as the following two properties), we choose $L2$ norm ($L2$ means p is equal to 2) for simplicity. This is a common choice in gradient-based optimization for several reasons:

1. Many gradient-based optimization methods (such as standard gradient descent) frequently use the $L2$ norm due to its mathematical convenience and well-established theoretical results. The $L2$ norm

simplifies gradient computations and is naturally induced by the Euclidean geometry of the parameter space.

2. The smoothness condition of a function, which will be discussed next, is typically defined using the $L2$ norm.
3. Many machine learning and optimization algorithms are designed with the $L2$ norm in mind due to its widespread adoption, ease of implementation, and well-studied convergence properties. However, it is worth noting that gradient-based methods can also be formulated with other norms, depending on the problem and desired regularization properties.

A Lipschitz function does not allow values to jump too rapidly and its rate of change is bounded. you can Think of it like a maximum slope that the function is allowed to have. For instance, The function $f(x) = |x|$ is Lipschitz continuous with $L = 1$ and $f(x) = 3x$ is Lipschitz continuous with $L = 3$. However, $f(x) = x^2$ is not Lipschitz continuous on the entire real line because its slope keeps increasing. But if you restrict x to a closed interval to zero, it can be Lipschitz continuous.

Property 2.2: Smoothness

A function $f : \mathbb{R}^d \to \mathbb{R}$ is called L-smooth if its gradient $\nabla f(X)$ is Lipschitz continuous with constant L, meaning there exists $L > 0$ such that for all X_1 and X_2 in its domain:

$$\|\nabla f(X_1) - \nabla f(X_2)\|_2 \leq L\|X_1 - X_2\|_2 \tag{15}$$

Equivalently, this can be rewritten in terms of function values:

$$f(X_2) \leq f(X_1) + (\nabla f(X_1))^T(X_2 - X_1) + \frac{L}{2}(\|X_2 - X_1\|_2)^2 \tag{16}$$

This condition ensures that the function does not change too abruptly. Intuitively, it means the gradient is not allowed to change too quickly and it has a bounded rate of variation.

For functions that are twice differentiable, smoothness can also be expressed using the Hessian matrix $\nabla^2 f(X)$. Specifically, for all X in its domain:

$$\lambda_{\max}(\nabla^2 f(X)) \leq L \tag{17}$$

where $\lambda_{\max}$ is the largest eigenvalue of the Hessian matrix $\nabla^2 f(X)$ (See Equation (6) for the definition of the Hessian matrix). This means that the function's curvature is bounded above, ensuring that it does not grow too steeply in any direction.

Property 2.3: Strong Convexity

A function $f : \mathbb{R}^d \to \mathbb{R}$ is strongly convex with parameter α if there exists $\alpha > 0$ such that for all X_1 and X_2 in its domain:

$$\|\nabla f(X_1) - \nabla f(X_2)\|_2 \geq \alpha\|X_1 - X_2\|_2 \tag{18}$$

Or equivalently, in terms of function values:

$$\underline{f(X_2) \geq f(X_1) + (\nabla f(X_1))^T(X_2 - X_1)} + \alpha(\|X_2 - X_1\|_2)^2. \tag{19}$$

Here, the underlined part is exactly same as First derivative condition for convexity. It means that for strong convexity not only $f(y)$ is greater than first two term it is even greater. It is greater than first two term plus some positive value $\alpha(\|X_2 - X_1\|)^2$. therefor function has a lot of cervature.
Similar to Smoothness case, for functions that are twice differentiable, strong Convexity can also be expressed using the Hessian matrix $\nabla^2 f(X)$. Specifically, for all X in its domain:

$$\lambda_{\min}(\nabla^2 f(X)) \geq \alpha \tag{20}$$

where $\lambda_{\min}$ is the smallest eigenvalue of the Hessian matrix $\nabla^2 f(X)$. As will be discussed in more detail in the next session, strong convexity helps gradient descent converge quickly because strong convexity, along with smoothness, ensures that the method moves toward the minimum at an exponential rate. Moreover, the function cannot be too flat, as it must grow at a certain minimum rate, preventing slow or unpredictable progress in optimization.

You can practice by working through the following exercises:

Exercise 2.1: Excersice 1

Let $f(X) = X^T H X$, where $X \in \mathbb{R}^d$ and H is a positive definite matrix satisfying:

$$\lambda_{\min}(H) \geq \alpha$$

show that f is strongly convex.

Exercise 2.2: Excersice 2

Let $f(X) = \frac{X^2}{2}$, where $X \in \mathbb{R}$. show that f is strongly convex.

Now that we are equipped with some of the prerequisites needed for a better understanding of the gradient descent algorithm, it is time to explore this important algorithm.

3 Gradient descent algorithm (GD)

In general, the primary objective of the gradient descent algorithm is to find the minimum of a function and its corresponding argument ($argmin_{X \in D} f(X)$). We typically denote this argmin (minimizer of f) as $X^* \in D$, where D represents the domain in which the function is valid.
For example, consider the function $f(X) = (X - 5)^2$ with $X \in \mathbb{R}$ (one-dimensional case). The minimum value of f is 0, and the minimizer of f is 5, i.e., $X^* = 5$.
Suppose we are given a function $f : D \to \mathbb{R}$ where $D \subseteq \mathbb{R}^d$. The gradient descent algorithm can be represented using the following pseudocode:

Algorithm 3.1: Gradient Descent

```
Input: An arbitrary starting point $X_0 \in D$
Output: Final value $X_T$.
Initialize $X_0 \leftarrow$ arbitrary starting point
for each element $t$ in $\{1, 2, ..., T\}$ do
    $X_t \leftarrow X_{t-1} - \eta \nabla f(X_{t-1})$
end for
return $X_T$
```

Here, t is the number of iterations, and η is the learning rate, which controls the step size of the updates. The negative sign in the update equation ensures that the algorithm moves in the opposite direction of the gradient. This makes sense because the gradient points in the direction of increasing function values, so moving in the negative direction decreases the function value. To further clarify, the iterative steps in more detail are as follows:

$$\begin{aligned}
t = 1: &\qquad X_1 \leftarrow X_0 - \eta \nabla f(X_0) \\
t = 2: &\qquad X_2 \leftarrow X_1 - \eta \nabla f(X_1) \\
t = 3: &\qquad X_3 \leftarrow X_2 - \eta \nabla f(X_2) \\
&\qquad \vdots \\
t = T: &\qquad X_T \leftarrow X_{T-1} - \eta \nabla f(X_{T-1})
\end{aligned} \tag{21}$$

Selecting the right value for η is crucial for efficient convergence. If η is too large, the algorithm may exhibit a zigzag pattern, overshooting the minimum and oscillating back and forth instead of converging smoothly. If η is too small, the convergence will be extremely slow, requiring many iterations to reach the minimum. In the following sections, we will explore strategies for selecting optimal values of T and η to ensure efficient and stable convergence.

To begin our discussion on choosing appropriate values for η and T, we consider a specific case where the function $f : \mathbb{R}^d \to \mathbb{R}$ is **convex** and **L-Lipschitz continuous**. We intentionally define the function over $\mathbb{R}^d$ to avoid concerns about stepping outside the function's domain during the iterative updates of the algorithm. This is an important consideration because, in practical cases where the function's domain is constrained, updates must remain within valid bounds. Such constraints can be handled using projection techniques, which we will discuss in later sections. In this case, an appropriate choice for η is:

$$\eta = \frac{\|X_0 - X^*\|}{L\sqrt{T}} \tag{22}$$

where L is the Lipschitz constant. But how can we be sure that this choice of η is appropriate? Let's attempt to justify it.

Our goal is to find a learning rate η that ensures a good convergence rate—meaning that the function value $f(X_t)$ approaches the optimal value $f(X^*)$ efficiently. During the optimization process, we define the regret R_t as the difference between the function value at each step and the optimal function value:

$$R_t = f(X_t) - f(X^*) \tag{23}$$

Regret is commonly used in optimization and machine learning to quantify how much worse a given decision is compared to the best possible decision in hindsight. Lower regret indicates better performance, as it

suggests the algorithm is making near-optimal choices.
Considering the entire optimization process, we aim to bound the cumulative regret:

$$R_T = \sum_{t=1}^{T} R_t = \sum_{t=1}^{T} f(X_t) - f(X^*) \tag{24}$$

We aim to show that our choice of step size η leads to a small regret, meaning that the iterative optimization algorithm effectively converges toward the optimal solution. Consider the iterative update rule:

$$X_{t+1} = X_t - \eta \nabla f(X_t) \tag{25}$$

Since f is convex, by utilizing the first convexity condition we have:

$$f(X_t) - f(X^*) \leq \nabla f(X_t)^\top (X_t - X^*). \tag{26}$$

By summing over all iterations from $t = 1$ to T, the cumulative regret is:

$$R_T = \sum_{t=1}^{T} (f(X_t) - f(X^*)) \leq \sum_{t=1}^{T} \nabla f(X_t)^\top (X_t - X^*). \tag{27}$$

Next, we take the squared norm of the update rule:

$$\|X_{t+1} - X^*\|^2 = \|X_t - \eta \nabla f(X_t) - X^*\|^2. \tag{28}$$

Expanding the squared term:

$$\|X_{t+1} - X^*\|^2 = \|X_t - X^*\|^2 - 2\eta \nabla f(X_t)^\top (X_t - X^*) + \eta^2 \|\nabla f(X_t)\|^2. \tag{29}$$

Rearranging:

$$\nabla f(X_t)^\top (X_t - X^*) = \frac{1}{2\eta} \left(\|X_t - X^*\|^2 - \|X_{t+1} - X^*\|^2 \right) + \frac{\eta}{2} \|\nabla f(X_t)\|^2. \tag{30}$$

Summing from $t = 1$ to T:

$$\sum_{t=1}^{T} \nabla f(X_t)^\top (X_t - X^*) \leq \frac{1}{2\eta} \left(\|X_0 - X^*\|^2 - \|X_T - X^*\|^2 \right) + \frac{\eta}{2} \sum_{t=1}^{T} \|\nabla f(X_t)\|^2. \tag{31}$$

Since f is L-Lipschitz, we have

$$\|\nabla f(X_t)\| \leq L, \tag{32}$$

Thus, summing over all iterations:

$$\sum_{t=1}^{T} \|\nabla f(X_t)\|^2 \leq T L^2. \tag{33}$$

If X_T is sufficiently close to X^*, we assume:

$$\|X_T - X^*\| \approx 0 \tag{34}$$

By using this into equation 32, we obtain:

$$\sum_{t=1}^{T} \nabla f(X_t)^\top (X_t - X^*) \leq \frac{\|X_0 - X^*\|^2}{2\eta} + \frac{\eta T L^2}{2}. \tag{35}$$

Considering this along with inequality in equation 28:

$$R_T \leq \frac{\|X_0 - X^*\|^2}{2\eta} + \frac{\eta T L^2}{2}. \tag{36}$$

To minimize the regret, we have to choose η to balance the two terms. Therefor:

$$\eta = \frac{\|X_0 - X^*\|}{L\sqrt{T}}$$

This is the same formula for η that we wanted to prove. But how to determine T. To determine T, we rely on minimizing the error in our optimization process. The error is defined as:

$$Error = f(X_T) - f(X^*) \tag{37}$$

For this purpose, we begin with the fact that the convexity of f implies:

$$f\left(\frac{1}{T}\sum_{t=1}^{T} X_t\right) \leq \frac{1}{T}\sum_{t=1}^{T} f(X_t) \tag{38}$$

Since our goal is to bound the suboptimality $f(X_T) - f(X^*)$, the best approach is to evaluate the function at an ”average iterate.” Convexity ensures that the function value at this average iterate is at most the average of the function values over all previous iterates. Therefore, subtracting $f(X^*)$ from both sides, we obtain:

$$f(X_T) - f(X^*) \leq \sum_{t=1}^{T} (f(X_t) - f(X^*)). \tag{39}$$

Thus:

$$f(X_T) - f(X^*) \leq \frac{R_T}{T}. \tag{40}$$

From inequality (37), we have:

$$f(X_T) - f(X^*) \leq \frac{\|X_0 - X^*\|^2}{2\eta T} + \frac{\eta L^2}{2}. \tag{41}$$

We already determined that an optimal choice of step size is:

$$\eta = \frac{\|X_0 - X^*\|}{LT}.$$

Substituting this into the bound and simplifying, we get:

$$f(X_T) - f(X^*) \leq \frac{L}{T}\|X_0 - X^*\|. \tag{42}$$

Now, To ensure the error bound is at most ε, we set::

$$\frac{L}{T}\|X_0 - X^*\| = \varepsilon. \tag{43}$$

Solving for T, we obtain::

$$T = \frac{L^2\|X_0 - X^*\|^2}{\varepsilon^2}. \tag{44}$$

This tells us how many iterations are required to achieve a desired accuracy ε.

As the last point of this session, one might question how we can use X^* in the formula for η while the optimization algorithm is designed to find X^*. To address this ambiguity, we typically replace $\|X_0 - X^*\|$ with an upper bound, which is often the known diameter of the feasible domain D, denoted as $\mathrm{diam}(D)$. This ensures that the step size η is chosen in a way that remains valid throughout the optimization process, even without direct knowledge of X^*. In unconstrained settings, where D is not explicitly bounded, alternative techniques such as adaptive step sizes or estimates based on problem-specific assumptions may be used.

Next Lecture

The next lecture will cover the following topics:
(i) Dradient descent convergence analysis,
(ii) Stochastic gradient descent + Convergence guarantees,
(iii) Batched Stochastic gradient descent.

References:

1. Book ”First-order and Stochastic Optimization Methods for Machine Learning” by Guanghui Lan, Chapter 2: Convex Optimization Theory.

2. Book ”Convex Optimization” by Stephen Boyd & Lieven Vandenberghe, Chapter 3: Convex Function.

3. Lecture note by Dimitri Bertsekas on Convex Analysis and Optimization

4. Lecture note by Vincent Duval, Optimization for machine learning

5. Lecture note by Andrew Ng on Gradient Descent, Linear Regression and Gradient Descent

6. Lecture note by Sivaraman Balakrishnan on Gradient Descent