Lecture 2: Optimization, Implicit Regularization, and Stochastic Gradient Descent

Jincheng Ou

Fall 25

Contents

1 Introduction and Review

In the previous lecture, we established that Gradient Descent (GD) converges for linear regression problems. A critical constraint for convergence is the learning rate η, which depends on the geometric properties of the data matrix X. Specifically, convergence requires $\eta < \frac{1}{\lambda_{max}(X^TX)}$. The geometric properties of the loss function dictate the choice of the learning rate.

This lecture explores how the singular value decomposition (SVD) of data helps explain the behavior of Ridge Regression and reveals the **implicit regularization** inherent in Gradient Descent. We then transition to Stochastic Gradient Descent (SGD), Momentum, and a convergence analysis for over-parameterized systems.

2 Regularization and Ridge Regression

To understand implicit regularization, we first analyze explicit regularization via Ridge Regression in the basis of the Singular Value Decomposition (SVD).

2.1 Ridge Solution in SVD Basis

The explicit solution for Ridge Regression is given by:

$$\vec{w}_{ridge} = (X^TX + \lambda I)^{-1}X^T\vec{y} = X^T(XX^T + \lambda I)^{-1}\vec{y} \tag{1}$$

Let the SVD of the data matrix be $X = U\Sigma V^T$. Substituting this into the Ridge solution allows us to analyze the weights component-wise:

$$\begin{aligned} \vec{w}_{ridge} &= V(\Sigma^T\Sigma + \lambda I)^{-1}\Sigma^T U^T\vec{y} \qquad (2) \\ &= V\begin{bmatrix} \frac{\sigma_1}{\sigma_1^2+\lambda} & 0 & \cdots \\ 0 & \frac{\sigma_2}{\sigma_2^2+\lambda} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} U^T\vec{y} \qquad (3) \end{aligned}$$

Let $\tilde{y} = U^T\vec{y}$ be the target rotated into the U-basis. The i-th component of the weight vector in the V-basis ($\tilde{w}$) is scaled by:

$$\frac{\sigma_i}{\sigma_i^2 + \lambda} \tag{4}$$

2.2 Spectral Filtering Interpretation

The interaction between the singular values σ_i and the regularization parameter λ acts as a filter:

- **Case 1: $\lambda \gg \sigma_i$ (Small singular values).** The term behaves like $\frac{\sigma_i}{\lambda} \approx 0$. The ridge solution suppresses components corresponding to small singular values (directions of low variance in the data).
- **Case 2: $\sigma_i \gg \lambda$ (Large singular values).** The term behaves like $\frac{\sigma_i}{\sigma_i^2} = \frac{1}{\sigma_i}$. This matches the standard Ordinary Least Squares (OLS) inversion.

Key Takeaway

Ridge regression prevents the model from moving in directions of small singular values (low data variance), which often correspond to noise. This improves generalization by avoiding overfitting to these specific directions.

3 Implicit Regularization of Gradient Descent

We now compare the explicit regularization of Ridge to the behavior of unregularized Gradient Descent.

3.1 Gradient Descent in SVD Basis

The standard GD update rule for the residual error vector is:

$$\vec{w}_{t+1} = \vec{w}_t - 2\eta X^T (X\vec{w}_t - \vec{y}) \tag{5}$$

Analyzing this in the SVD basis (where $\tilde{w}_t = V^T \vec{w}_t$ and $\tilde{y} = U^T \vec{y}$), the update for the i-th component becomes:

$$\tilde{w}_{t+1,i} = \tilde{w}_{t,i} - 2\eta\sigma_i(\sigma_i \tilde{w}_{t,i} - \tilde{y}_i) \tag{6}$$

This reveals that the convergence speed depends on σ_i.

- Directions with **large** σ_i (high curvature) converge quickly.
- Directions with **small** σ_i (low curvature) converge very slowly.

3.2 Early Stopping as Regularization

Because GD prioritizes learning along directions with large singular values first, stopping the algorithm early (Early Stopping) results in a solution where the components corresponding to small σ_i have not yet developed significant magnitude.

- **Ridge:** Explicitly penalizes directions with small singular values.
- **Early Stopping:** Implicitly avoids moving far in directions with small singular values because the convergence in those directions is slow.

Thus, GD with early stopping exhibits **implicit regularization** similar to explicit Ridge regression.

4 Stochastic Gradient Descent (SGD)

While GD is effective for convex functions, neural network landscapes are non-convex and datasets are often too large for full-batch GD.

4.1 Definition

Stochastic Gradient Descent approximates the full gradient using a "mini-batch" of size B. The update rule is:

$$\theta_{t+1} = \theta_t - \eta \cdot \frac{1}{B}\sum_{i=1}^{B} \nabla f_i(\theta_t) \tag{7}$$

where the indices i are drawn uniformly i.i.d from the dataset.

- B is the batch size.
- In expectation, the stochastic gradient equals the full gradient: $E[\nabla f_i(\theta)] = \nabla f(\theta)$.

4.2 Comparison with GD

- **Computational Cost:** SGD is significantly cheaper per iteration than full-batch GD.
- **Noise and Escaping Minima:** The noise introduced by the random sampling helps the optimization process escape local minima and saddle points, which is crucial for non-convex landscapes.

As shown in Figure 6.5 (from the source):

1. GD (Fig a) moves steadily toward the nearest minimum. If initialized in the "wrong valley" (e.g., point 2), it gets stuck in a local minimum.
2. SGD (Fig b) follows a noisy path. This noise allows it to potentially jump out of the wrong valley and find the global minimum.

5 Momentum

A scalar analysis of GD shows that the maximum learning rate is gated by the maximum singular value ($\eta < 1/\sigma_{max}^2$). This creates a dilemma:

- If η is set for σ_{max}, convergence along directions with small σ (shallow curvature) is extremely slow.
- If η is increased to speed up small σ directions, the system oscillates or diverges along the large σ directions.

5.1 Smoothing Oscillations

To solve this, we apply the concept of a low-pass filter (averaging) to the gradients, similar to an RC circuit in physics. This technique is called **Momentum**.

We introduce a state variable $\vec{z}$ (velocity) that accumulates history:

$$\vec{z}_{k+1} = \beta\vec{z}_k + (1-\beta)\nabla f(\vec{w}_k) \quad \text{with } \beta \in [0,1] \tag{8}$$

$$\vec{w}_{k+1} = \vec{w}_k - \eta\vec{z}_{k+1} \tag{9}$$

5.2 Interpretation

- $\vec{z}_{k+1}$ is an exponential moving average of past gradients.
- If $\beta = 0$, this reduces to standard Gradient Descent.
- Momentum "smooths out" the trajectory. In directions where the gradient oscillates (zig-zags), the terms cancel out. In directions where the gradient is consistent, the terms add up, increasing speed.

Figure 6.7 illustrates that Momentum takes a more direct, smoother path toward the minimum compared to the erratic path of standard SGD.

6 SGD Convergence Analysis

Can SGD converge with a constant step size? We analyze this for an under-determined linear system $X\vec{w} = \vec{y}$ where $X \in \mathbb{R}^{n\times d}$ with $d > n$ (full row rank).

6.1 Setup

We assume batch size 1 for simplicity. We analyze the error vector $\vec{q} = \vec{w} - \vec{w}^*$, where $\vec{w}^*$ is the minimum norm solution. The system equation becomes $X\vec{q} = 0$.

In the SVD basis, we focus on the variable $\vec{z} = V^T\vec{q}$. The update rule for SGD with batch size 1 at time t (sampling index I_t) is:

$$\vec{z}_{t+1} = \vec{z}_t - 2\eta\tilde{x}_{I_t}\tilde{x}_{I_t}^T\vec{z}_t \tag{10}$$

6.2 Lyapunov Function Approach

We define a Lyapunov function (an energy function) to prove convergence:

$$L(\vec{z}) = ||\tilde{X}\vec{z}||_2^2 = \vec{z}^T\tilde{X}^T\tilde{X}\vec{z} \tag{11}$$

We aim to show that $E[L(\vec{z}_{t+1})] \le (1-\epsilon)L(\vec{z}_t)$, proving contraction.

Expanding $L(\vec{z}_{t+1})$:

$$L(\vec{z}_{t+1}) = L(\vec{z}_t) + \underbrace{2\vec{z}_t^T\tilde{X}^T\tilde{X}(\vec{z}_{t+1} - \vec{z}_t)}_{A(\text{Linear})} + \underbrace{(\vec{z}_{t+1} - \vec{z}_t)^T\tilde{X}^T\tilde{X}(\vec{z}_{t+1} - \vec{z}_t)}_{B(\text{Quadratic})} \tag{12}$$

6.2.1 Analysis of Term A (Descent)

The expected value of the linear term A, given the randomness of SGD comes from $E[\tilde{x}_{I_t}\tilde{x}_{I_t}^T] = \frac{1}{n}\tilde{X}^T\tilde{X}$:

$$E[A|\vec{z}_t] = -\frac{4\eta}{n}||\tilde{X}^T\tilde{X}\vec{z}_t||_2^2 \leq -\frac{4\eta}{n}\sigma_{min}^2 L(\vec{z}_t) \tag{13}$$

This term provides the downward pressure (convergence).

6.2.2 Analysis of Term B (Variance/Noise)

The quadratic term B represents the variance added by the stochastic updates. It is bounded by the maximum norm of the data points (S^2) and the maximum singular value:

$$E[B|\vec{z}_t] \leq C \cdot \eta^2 L(\vec{z}_t) \tag{14}$$

6.3 Convergence Result

Combining A and B:

$$E[L(\vec{z}_{t+1})|\vec{z}_t] \leq (1 - a\eta + c_2\eta^2)L(\vec{z}_t) \tag{15}$$

Key Takeaway

For convergence, we need $(1 - a\eta + c_2\eta^2) < 1$. This is satisfied if η is small enough (specifically, the linear descent term $a\eta$ must dominate the quadratic noise term $c_2\eta^2$).

This proves that SGD can converge with a constant step size on over-parameterized linear systems if the loss is capable of reaching zero.