

EECS 182 Deep Neural Networks
Fall 2026 Gireeja Ranade

Homework 2

Due date: Sep 16, 2026, at 10:59 PM.

1. Stochastic Gradient Descent (when it is possible to interpolate)

This is a problem about the convergence of SGD for least-squares problems when there is actually a solution that achieves zero loss.

For simplicity, suppose that the problem we are given is

$$X\mathbf{w} = \mathbf{y} \tag{1}$$

where $X = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ is a wide matrix with $\mathbf{x}_i$ being d -dimensional vectors and $\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$ is an n -dimensional

vector. Here, we assume that X has full row-rank (i.e. the $\mathbf{x}_i$ are linearly independent) and so (1) indeed has solutions. (Note that as $d > n$, there are infinitely many solutions.)

While we already know lots of ways of solving this problem, it is an illustrative toy example to make sure we understand why SGD works in such settings. (This material was covered in lecture but you really do need to understand it yourself so you can deal with variations you might encounter as well as internalize the kinds of manipulations required.) This problem has a Demo Notebook ([demo link](#)) associated with it to help you play around with things to get an even deeper set of intuitions.

In this problem, we will just initialize $\mathbf{w}_0 = \mathbf{0}$ for simplicity.

- (a) Let's do some preliminaries. First, we want to change coordinates to notationally simplify our analysis of SGD.

Let $\mathbf{w}^*$ be the min-norm solution to (1).

Write out what $\mathbf{w}^*$ is explicitly with respect to X and $\mathbf{y}$ and then, change coordinates to $\mathbf{w}' = \mathbf{w} - \mathbf{w}^*$ to write the new equations as:

$$X\mathbf{w}' = \mathbf{0} \tag{2}$$

What is the new initial condition for $\mathbf{w}'_0$?

Solution: The min-norm solution $\mathbf{w}^*$ to (1) is

$$\mathbf{w}^* = X^T(XX^T)^{-1}\mathbf{y} \tag{3}$$

and so the new equation is

$$X\mathbf{w}' = X(\mathbf{w} - \mathbf{w}^*) = X\mathbf{w} - X\mathbf{w}^* = \mathbf{y} - \mathbf{y} = \mathbf{0} \tag{4}$$

and the new initial condition is $\mathbf{w}'_0 = \mathbf{w}_0 - \mathbf{w}^* = -\mathbf{w}^*$.

- (b) Next, let's leverage SVD coordinates to further simplify the problem. **Show that there exists an orthogonal matrix V giving the change of variables $\mathbf{w}'' = V^T \mathbf{w}'$ so that (2) looks like**

$$[\tilde{X} \quad \mathbf{0}_{n \times (d-n)}] \mathbf{w}'' = \mathbf{0} \quad (5)$$

and furthermore, **show that the initial condition for $\mathbf{w}'_0$ you computed in the previous part, when viewed as $\mathbf{w}''_0$ has all zeros in the final $(d - n)$ positions.**

Hint: use the full SVD $X = U\Sigma V^T$ and change only the parameter coordinates; do not multiply the equations by U^T .

Solution: SVD of X is $X = U\Sigma V^T$ where $U \in \mathbb{R}^{n \times n}$, $\Sigma \in \mathbb{R}^{n \times d}$, $V \in \mathbb{R}^{d \times d}$ and U and V are orthonormal. Non-zero singular values of X are $\sigma_1, \sigma_2, \dots, \sigma_n$; the remaining $d - n$ columns of Σ are zero. Σ can be written as

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \dots & 0 & 0 \\ & \sigma_2 & & \dots & 0 & 0 \\ & & \ddots & \dots & 0 & 0 \\ & & & \sigma_n & \dots & 0 & 0 \end{bmatrix} = \begin{bmatrix} \Sigma_1 & \mathbf{0}_{n \times (d-n)} \end{bmatrix} \quad (6)$$

, where $\Sigma_1 = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \in \mathbb{R}^{n \times n}$. Therefore, we can write $X\mathbf{w}'$ as

$$X\mathbf{w}' = U\Sigma V^T \mathbf{w}' \quad (7)$$

$$= U \begin{bmatrix} \Sigma_1 & \mathbf{0}_{n \times (d-n)} \end{bmatrix} V^T \mathbf{w}' \quad (8)$$

$$= \begin{bmatrix} U\Sigma_1 & \mathbf{0}_{n \times (d-n)} \end{bmatrix} V^T \mathbf{w}' \quad (9)$$

$$= \begin{bmatrix} \tilde{X} & \mathbf{0}_{n \times (d-n)} \end{bmatrix} V^T \mathbf{w}' \quad (10)$$

$$= \begin{bmatrix} \tilde{X} & \mathbf{0}_{n \times (d-n)} \end{bmatrix} \mathbf{w}'' \quad (11)$$

where $\tilde{X} = U\Sigma_1$ and $\mathbf{w}'' = V^T \mathbf{w}'$. Therefore, (2) becomes (5).

The initial condition is $\mathbf{w}_0'' = V^T \mathbf{w}_0' = -V^T \mathbf{w}^*$. Using the formula from part (a),

$$\mathbf{w}_0'' = -V^T X^T (X X^T)^{-1} \mathbf{y} \quad (12)$$

$$= -V^T (U \Sigma V^T)^T ((U \Sigma V^T)(U \Sigma V^T)^T)^{-1} \mathbf{y} \quad (13)$$

$$= -V^T V \Sigma^T U^T (U \Sigma V^T V \Sigma^T U^T)^{-1} \mathbf{y} \quad (14)$$

$$= -V^T V \begin{bmatrix} \Sigma_1 \\ \mathbf{0}_{(d-n) \times n} \end{bmatrix} U^T (U \begin{bmatrix} \Sigma_1 & \mathbf{0}_{n \times (d-n)} \end{bmatrix} \begin{bmatrix} \Sigma_1^T \\ \mathbf{0}_{(d-n) \times n} \end{bmatrix} U^T)^{-1} \mathbf{y} \quad (15)$$

$$= - \begin{bmatrix} \Sigma_1 \\ \mathbf{0}_{(d-n) \times n} \end{bmatrix} U^T (U \Sigma_1^2 U^T)^{-1} \mathbf{y} \quad (16)$$

$$= - \begin{bmatrix} \Sigma_1 \\ \mathbf{0}_{(d-n) \times n} \end{bmatrix} U^T U \Sigma_1^{-2} U^T \mathbf{y} \quad (17)$$

$$= - \begin{bmatrix} \Sigma_1^{-1} \\ \mathbf{0}_{(d-n) \times n} \end{bmatrix} U^T \mathbf{y} \quad (18)$$

$$= - \begin{bmatrix} \Sigma_1^{-1} U^T \mathbf{y} \\ \mathbf{0}_{d-n} \end{bmatrix} \quad (19)$$

Therefore, the initial condition for $\mathbf{w}_0''$ has all zeros in the final $(d - n)$ positions.

- (c) Let $\tilde{\mathbf{w}} \in \mathbb{R}^n$ be the first n coordinates of $\mathbf{w}''$, and let $\tilde{x}_i^\top$ be row i of $\tilde{X} \in \mathbb{R}^{n \times n}$. **Argue why what you have seen in the previous parts allows us to now focus on a square system of equations:**

$$\tilde{X} \tilde{\mathbf{w}} = \mathbf{0} \quad (20)$$

and furthermore show that each of the n constituent equations (corresponding to rows) of (20) can be obtained by means of coordinate changes from the same indexed equation in (1).

Solution: Let's first show that we can focus on a square system of equations. The last $d - n$ coordinates start at zero. Every individual squared loss is independent of those coordinates, because the corresponding columns of $XV = [\tilde{X} \mathbf{0}]$ are zero. Their gradients are therefore zero, so SGD leaves those coordinates at zero.

For each original row, $\mathbf{x}_i^\top V = [\tilde{x}_i^\top \mathbf{0}]$. Consequently,

$$\mathbf{x}_i^\top \mathbf{w} - y_i = \mathbf{x}_i^\top (\mathbf{w} - \mathbf{w}^*) = \mathbf{x}_i^\top V \mathbf{w}'' = \tilde{x}_i^\top \tilde{\mathbf{w}}. \quad (21)$$

Thus each row keeps its original index and residual; the coordinate change does not mix different training examples.

- (d) Let's now engage with SGD itself. Here, we will just use a minibatch length of 1 and batch sampling with replacement. This means that at every iteration of SGD, we roll a fair n -sided die and choose the single equation in (1) that corresponds to the row that came up on the die. Let I_t be the iid uniform random variable on $\{1, \dots, n\}$ sampled independently of all previous choices before updating $\mathbf{w}_t$ to $\mathbf{w}_{t+1}$.

At the $t + 1$ -th iteration, we compute

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla \mathcal{L}_{I_t}(\mathbf{w}_t) \quad (22)$$

where $\mathcal{L}_i(\mathbf{w}) = (y[i] - \mathbf{x}_i^\top \mathbf{w})^2$ is the squared loss on the i -th equation and η is the step-size (learning rate).

Show that an SGD step taken in (22) for the original optimization problem matches exactly to an SGD step taken for $\tilde{w}$ for solving (20), and that in particular these steps look like:

$$\tilde{w}_{t+1} = \tilde{w}_t - 2\eta \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{w}_t \quad (23)$$

Solution: Let's first derive the SGD update for the original optimization problem with respect to $\mathbf{w}'$. As $\mathbf{w}'_t = \mathbf{w}_t - \mathbf{w}^*$, we can write the SGD update for the original optimization problem as

$$\mathbf{w}'_{t+1} = \mathbf{w}'_t - \eta \nabla \mathcal{L}_{I_t}(\mathbf{w}'_t + \mathbf{w}^*) \quad (24)$$

$$= \mathbf{w}'_t - 2\eta (\mathbf{x}_{I_t}^T \mathbf{w}'_t + \mathbf{x}_{I_t}^T \mathbf{w}^* - y_{I_t}) \mathbf{x}_{I_t} \quad (25)$$

$$= \mathbf{w}'_t - 2\eta \mathbf{x}_{I_t} \mathbf{x}_{I_t}^T \mathbf{w}'_t \quad (26)$$

Define $\mathbf{z}_i = V^\top \mathbf{x}_i = [\tilde{x}_i^\top \mathbf{0}_{d-n}^\top]^\top \in \mathbb{R}^d$. Multiplying the update by $V^\top$ gives

$$\mathbf{w}''_{t+1} = \mathbf{w}''_t - 2\eta \mathbf{z}_{I_t} \mathbf{z}_{I_t}^\top \mathbf{w}''_t, \quad (27)$$

$$\tilde{w}_{t+1} = \tilde{w}_t - 2\eta \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{w}_t. \quad (28)$$

The second line takes the first n coordinates; the remaining coordinates stay zero. This is exactly (23).

- (e) At this point, we can focus entirely on the simplified square system (20) and the stochastic evolution of the iterations described by (23).

To show convergence to zero, we need to pick a suitable stochastic Lyapunov function $\mathcal{L}(\tilde{w})$ that is bounded below by zero and will decrease in expectation at every time step. Here, a Lyapunov function is a nonnegative measure of how far the iterate is from the solution. In particular, we want to establish

$$E[\mathcal{L}(\tilde{w}_{t+1}) | \tilde{w}_t] \leq (1 - \rho) \mathcal{L}(\tilde{w}_t) \quad (29)$$

with a $1 > \rho > 0$ so that this Lyapunov function tends to decrease exponentially to zero. We will have to have a suitably small step-size/learning-rate η for this to happen, of course.

Show that (29) indeed implies that for every $\epsilon > 0$ and $\delta > 0$, there exists a $T > 0$ for which

$$P(\mathcal{L}(\tilde{w}_T) < \epsilon) \geq 1 - \delta. \quad (30)$$

Hint: first take expectations over $\tilde{w}_t$ and iterate the bound. Then use Markov's inequality $P(Z \geq \epsilon) \leq E[Z]/\epsilon$ for $Z \geq 0$.

Solution: From (29), we can derive the relationship between $\mathcal{L}(\tilde{w}_{t+1})$ and $\mathcal{L}(\tilde{w}_t)$.

$$E[\mathcal{L}(\tilde{w}_{t+1})] = E[E[\mathcal{L}(\tilde{w}_{t+1}) | \tilde{w}_t]] \quad (31)$$

$$\leq (1 - \rho) E[\mathcal{L}(\tilde{w}_t)]. \quad (32)$$

Therefore, we have

$$E[\mathcal{L}(\tilde{w}_T)] \leq (1 - \rho)^T \mathcal{L}(\tilde{w}_0) \quad (33)$$

As Lyapunov function is bounded below by zero, we can apply Markov's inequality to get

$$P(\mathcal{L}(\tilde{w}_T) < \epsilon) \geq 1 - \frac{E[\mathcal{L}(\tilde{w}_T)]}{\epsilon} \quad (34)$$

$$\geq 1 - \frac{(1 - \rho)^T \mathcal{L}(\tilde{w}_0)}{\epsilon}. \quad (35)$$

The initial iterate is fixed, so $\mathcal{L}(\tilde{w}_0)$ is a finite constant. Choose an integer T large enough that $(1 - \rho)^T \mathcal{L}(\tilde{w}_0) \leq \epsilon \delta$. This gives the requested probability bound. Thus the expected Lyapunov function decays geometrically, and the Lyapunov function converges to zero in probability.

(f) One natural guess for a stochastic Lyapunov function is

$$\mathcal{L}(\tilde{w}) = \tilde{w}^\top \tilde{X}^\top \tilde{X} \tilde{w}. \quad (36)$$

Argue why the candidate Lyapunov function $\mathcal{L}(\tilde{w})$ in (36) is non-negative and is only equal to zero at $\tilde{w} = \mathbf{0}$.

Solution: Note that $\tilde{X} \in \mathbb{R}^{n \times n}$ is full rank. Therefore, the matrix $\tilde{X}^\top \tilde{X}$ is strictly positive definite. Therefore, $\tilde{w}^\top \tilde{X}^\top \tilde{X} \tilde{w} \geq 0$ and $\tilde{w}^\top \tilde{X}^\top \tilde{X} \tilde{w} = 0$ if and only if $\tilde{w} = \mathbf{0}$.

(g) To prove (29), split the change in the Lyapunov function as

$$\mathcal{L}(\tilde{w}_{t+1}) = \mathcal{L}(\tilde{w}_t) + A + B \quad (37)$$

where the term A is linear in the actual stochastic update $(\tilde{w}_{t+1} - \tilde{w}_t)$ and the term B is quadratic in that update.

Expand out $\mathcal{L}(\tilde{w}_t + (\tilde{w}_{t+1} - \tilde{w}_t))$ to give explicit forms for A and B .

Solution: Write $\Delta_t = \tilde{w}_{t+1} - \tilde{w}_t$ and $H = \tilde{X}^\top \tilde{X}$. Since $H = H^\top$,

$$\mathcal{L}(\tilde{w}_t + \Delta_t) = (\tilde{w}_t + \Delta_t)^\top H (\tilde{w}_t + \Delta_t) \quad (38)$$

$$= \mathcal{L}(\tilde{w}_t) + 2\tilde{w}_t^\top H \Delta_t + \Delta_t^\top H \Delta_t. \quad (39)$$

Therefore, we have

$$A = 2\tilde{w}_t^\top \tilde{X}^\top \tilde{X} (\tilde{w}_{t+1} - \tilde{w}_t) \quad (40)$$

$$B = (\tilde{w}_{t+1} - \tilde{w}_t)^\top \tilde{X}^\top \tilde{X} (\tilde{w}_{t+1} - \tilde{w}_t) \quad (41)$$

(h) We are counting on the term A in (37) to give us actual contraction in expectation since this looks like a gradient-descent step. **Show:**

$$E[A|\tilde{w}_t] \leq -c_1 \eta \mathcal{L}(\tilde{w}_t) \quad (42)$$

where the $c_1 > 0$ is a positive constant that depends on the problem.

Hint: use the update (23) and the smallest singular value of $\tilde{X}$. Why is that singular value positive?

Solution: Plugging in the SGD update for (23) into $\tilde{w}_{t+1}$, we have

$$E[A|\tilde{w}_t] = E[2\tilde{w}_t^\top \tilde{X}^\top \tilde{X}(\tilde{w}_{t+1} - \tilde{w}_t)|\tilde{w}_t] \quad (43)$$

$$= 2\tilde{w}_t^\top \tilde{X}^\top \tilde{X} E[(\tilde{w}_{t+1} - \tilde{w}_t)|\tilde{w}_t] \quad (44)$$

$$= 2\tilde{w}_t^\top \tilde{X}^\top \tilde{X} E[-2\eta \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{w}_t|\tilde{w}_t] \quad (45)$$

$$= -4\eta \tilde{w}_t^\top \tilde{X}^\top \tilde{X} E[\tilde{x}_{I_t} \tilde{x}_{I_t}^\top] \tilde{w}_t \quad (46)$$

$$= -\frac{4}{n} \eta \tilde{w}_t^\top \tilde{X}^\top \tilde{X} \tilde{X}^\top \tilde{X} \tilde{w}_t \quad (47)$$

$$\leq -\frac{4}{n} \eta \sigma_{\min}^2 \tilde{w}_t^\top \tilde{X}^\top \tilde{X} \tilde{w}_t \quad (48)$$

$$= -\frac{4}{n} \eta \sigma_{\min}^2 \mathcal{L}(\tilde{w}_t) \quad (49)$$

Therefore, we have $c_1 = \frac{4}{n} \sigma_{\min}^2$. Here $\sigma_{\min}$ is the smallest singular value of $\tilde{X}$, which is positive because $\tilde{X}$ is full rank.

- (i) We need to make sure that the “quadratic” term B in (37) cannot undo the progress made by A in expectation. **Show:**

$$E[B|\tilde{w}_t] \leq c_2 \eta^2 \mathcal{L}(\tilde{w}_t) \quad (50)$$

where $c_2 > 0$ is another positive constant that depends on the problem.

Hint: substitute (23). Bound the quadratic form using the largest singular value of $\tilde{X}$, then bound each row norm by $\beta = \max_i \|\tilde{x}_i\|_2$.

Solution: Plugging in the SGD update for (23) into $\tilde{w}_{t+1}$, we have

$$E[B|\tilde{w}_t] = E[(\tilde{w}_{t+1} - \tilde{w}_t)^\top \tilde{X}^\top \tilde{X}(\tilde{w}_{t+1} - \tilde{w}_t)|\tilde{w}_t] \quad (51)$$

$$= E[-2\eta \tilde{w}_t^\top \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{X}^\top \tilde{X}(-2\eta \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{w}_t)|\tilde{w}_t] \quad (52)$$

$$= 4\eta^2 E[\tilde{w}_t^\top \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{X}^\top \tilde{X} \tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{w}_t|\tilde{w}_t] \quad (53)$$

$$\leq 4\eta^2 \sigma_{\max}^2 \tilde{w}_t^\top E[\tilde{x}_{I_t} \tilde{x}_{I_t}^\top \tilde{x}_{I_t} \tilde{x}_{I_t}^\top] \tilde{w}_t \quad (54)$$

$$\leq 4\eta^2 \sigma_{\max}^2 \beta^2 \tilde{w}_t^\top E[\tilde{x}_{I_t} \tilde{x}_{I_t}^\top] \tilde{w}_t \quad (55)$$

$$= \frac{4}{n} \eta^2 \sigma_{\max}^2 \beta^2 \mathcal{L}(\tilde{w}_t) \quad (56)$$

where $\sigma_{\max}$ is the largest singular value of $\tilde{X}$ and $\beta = \max_i \|\tilde{x}_i\|_2$ is the largest row norm. Therefore, $c_2 = \frac{4}{n} \sigma_{\max}^2 \beta^2$. Both c_1 and c_2 depend only on the data, not on η .

- (j) Finally, we can put the pieces together to see that

$$E[\mathcal{L}(\tilde{w}_{t+1})|\tilde{w}_t] \leq (1 - c_1 \eta + c_2 \eta^2) \mathcal{L}(\tilde{w}_t) \quad (57)$$

where $c_1 > 0$ and $c_2 > 0$ as well. **Show that there exists a sufficiently small positive η so that $0 < 1 - c_1 \eta + c_2 \eta^2 < 1$, giving the contraction required in (29).**

Solution:

$$E[\mathcal{L}(\tilde{w}_{t+1})|\tilde{w}_t] = E[\mathcal{L}(\tilde{w}_t) + A + B|\tilde{w}_t] \quad (58)$$

$$= \mathcal{L}(\tilde{w}_t) + E[A|\tilde{w}_t] + E[B|\tilde{w}_t] \quad (59)$$

$$\leq (1 - c_1\eta + c_2\eta^2)\mathcal{L}(\tilde{w}_t). \quad (60)$$

Choose $0 < \eta < \min\{c_1/c_2, 1/c_1\}$. Then $\rho = c_1\eta - c_2\eta^2 = \eta(c_1 - c_2\eta) > 0$, while $\rho < c_1\eta < 1$. Hence the contraction factor is $1 - \rho \in (0, 1)$. (The quadratic upper bound $1 - c_1\eta + c_2\eta^2$ is minimized at $\eta = c_1/(2c_2)$; this need not be the best step size in practice.)

- (k) We can apply this interpolation argument to ridge regression through *feature augmentation*. Here is the needed reminder: for any data matrix X and $\lambda > 0$, set

$$\tilde{X} = [X \ \sqrt{\lambda}I_n], \quad \theta = \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix}.$$

The matrix $\tilde{X}$ is wide and has full row rank. The first d coordinates of the minimum-norm solution to $\tilde{X}\theta = \mathbf{y}$ give the ridge-regression solution: eliminating $\mathbf{u} = (\mathbf{y} - X\mathbf{w})/\sqrt{\lambda}$ from $\|\theta\|_2^2$ gives $\|\mathbf{w}\|_2^2 + \|\mathbf{y} - X\mathbf{w}\|_2^2/\lambda$.

Open the **demo notebook** and run the imports and helper-function cells at the top, followed by Part (4). Parts (1)–(3) are optional. Part (4) compares direct SGD on ridge regression (“Original Ridge”) with SGD on the augmented system (“Feature Augmented”), both with a constant step size. **In a few sentences, compare the convergence of the squared parameter error $\|\mathbf{w}_t - \mathbf{w}_{\text{ridge}}^*\|_2^2$ in the two plots. Which keeps decreasing, and which settles around a nonzero error level?** *Hint: approximately straight downward trends on these plots, which have a logarithmic vertical axis, indicate geometric decay. Focus on the trend, not on fluctuations at individual iterations. You may reduce Part (4)’s `N_iteration` to 100,000 to shorten the run. The notebook’s averaged-loss convention uses $\lambda = n\alpha$.*

Solution: For the notebook’s chosen step size, the feature-augmented method’s squared parameter error decreases approximately geometrically until numerical precision becomes relevant. Direct ridge SGD initially improves but then fluctuates around a nonzero error level. Individual examples generally have nonzero gradients at the ridge optimum, so constant-step SGD continues to receive noisy updates there.

For the augmented system, all per-example residuals and gradients vanish at its minimum-norm solution. Starting from zero, the preceding analysis therefore gives geometric convergence for a sufficiently small constant step size. Its first d coordinates converge to the ridge solution. The quantity tending to zero in the plots is parameter error; the original ridge objective need not have minimum value zero.

2. Accelerating Gradient Descent with Momentum

Let $X \in \mathbb{R}^{n \times d}$ have full column rank ($n \geq d$), and let $y \in \mathbb{R}^n$. Thus $X^T X$ is invertible and all d singular values of X are positive. Consider finding the minimizer of the following objective:

$$\mathcal{L}(w) = \|y - Xw\|_2^2 \quad (61)$$

Part (a) provides the gradient descent (GD) convergence results we will use. In this problem, we add momentum and see how it affects convergence. The update rule is

$$\begin{aligned} w_{t+1} &= w_t - \eta z_{t+1} \\ z_{t+1} &= (1 - \beta)z_t + \beta g_t, \end{aligned} \quad (62)$$

where $g_t = \nabla \mathcal{L}(w_t)$, $\eta > 0$ is the learning rate, and β is the weight on the current gradient in the moving average. For the momentum analysis, assume $0 < \beta < 1$. Setting $\beta = 1$ instead gives ordinary gradient descent. A system is *stable* here if its error converges to zero from every initial state; this holds when every eigenvalue of its update matrix has magnitude less than 1.

We will see how an appropriate choice of momentum and learning rate can accelerate convergence.

(a) Recall that the gradient descent update of (61) is

$$w_{t+1} = \left(I - 2\eta(X^T X) \right) w_t + 2\eta X^T y \quad (63)$$

and the minimizer is

$$w^* = (X^T X)^{-1} X^T y \quad (64)$$

The geometric convergence factor for the parameter error is

$$\text{rate} = \max_i |1 - 2\eta\sigma_i^2| \quad (65)$$

A factor below 1 gives convergence; a smaller factor means faster decay. Choosing the learning rate that minimizes (65) gives the optimal learning rate

$$\eta^* = \frac{1}{\sigma_{\min}^2 + \sigma_{\max}^2}, \quad (66)$$

where $\sigma_{\max}$ and $\sigma_{\min}$ are the maximum and minimum singular value of the matrix X . The corresponding optimal convergence rate is

$$\text{optimal rate} = \frac{(\sigma_{\max}/\sigma_{\min})^2 - 1}{(\sigma_{\max}/\sigma_{\min})^2 + 1} \quad (67)$$

Therefore, how fast ordinary gradient descent converges is determined by the ratio between the maximum singular value and the minimum singular value as above.

Now, let's consider using momentum to smooth the gradients before taking a step in (62).

$$\begin{aligned} w_{t+1} &= w_t - \eta z_{t+1} \\ z_{t+1} &= (1 - \beta)z_t + \beta(2X^T X w_t - 2X^T y) \end{aligned} \quad (68)$$

Use the SVD $X = U\Sigma V^T$, where V is a $d \times d$ orthogonal matrix and Σ has the same shape as X . In particular, $V^T X^T X V = \Sigma^T \Sigma = \text{diag}(\sigma_1^2, \dots, \sigma_d^2)$. This lets us express the parameter error and averaged gradient in the singular directions:

$$\begin{aligned} x_t &= V^T(w_t - w^*) \\ a_t &= V^T z_t. \end{aligned} \quad (69)$$

Please rewrite (68) with the reparameterized variables, $x_t[i]$ and $a_t[i]$. ($x_t[i]$ and $a_t[i]$ are i -th components of x_t and a_t respectively.)

Solution: Since $X^T X w^* = X^T y$ and $w_t - w^* = V x_t$, the gradient is $g_t = 2X^T X(w_t - w^*)$. Multiplying the updates by V^T gives

$$\begin{aligned} a_{t+1} &= (1 - \beta)a_t + 2\beta V^T X^T X V x_t \\ &= (1 - \beta)a_t + 2\beta \Sigma^T \Sigma x_t, \\ x_{t+1} &= x_t - \eta a_{t+1}. \end{aligned}$$

If we express the above result element-wise,

$$\begin{aligned} a_{t+1}[i] &= (1 - \beta)a_t[i] + 2\beta\sigma_i^2 x_t[i] \\ x_{t+1}[i] &= x_t[i] - \eta a_{t+1}[i] \end{aligned}$$

- (b) Notice that the above 2×2 vector/matrix recurrence has no external input. We can derive the 2×2 system matrix R_i from above such that

$$\begin{bmatrix} a_{t+1}[i] \\ x_{t+1}[i] \end{bmatrix} = R_i \begin{bmatrix} a_t[i] \\ x_t[i] \end{bmatrix} \quad (70)$$

Derive R_i .

Solution:

Since x_{t+1} is expressed with x_t and a_{t+1} , let's reformulate with x_t and a_t

$$\begin{aligned} x_{t+1}[i] &= x_t[i] - \eta a_{t+1}[i] \\ &= x_t[i] - \eta(1 - \beta)a_t[i] - 2\eta\beta\sigma_i^2 x_t[i] \\ &= (1 - 2\eta\beta\sigma_i^2)x_t[i] - \eta(1 - \beta)a_t[i] \end{aligned}$$

Therefore, the matrix R_i can be represented as below:

$$R_i = \begin{bmatrix} (1 - \beta) & 2\beta\sigma_i^2 \\ -\eta(1 - \beta) & 1 - 2\eta\beta\sigma_i^2 \end{bmatrix}$$

(c) A symbolic calculation gives the characteristic polynomial

$$\begin{aligned}
 f(\lambda) &= \det(R_i - \lambda I) \\
 &= (1 - \beta - \lambda)(1 - 2\eta\beta\sigma_i^2 - \lambda) + 2\eta\beta(1 - \beta)\sigma_i^2 \\
 &= \lambda^2 - (2 - \beta - 2\eta\beta\sigma_i^2)\lambda + 1 - \beta.
 \end{aligned} \tag{71}$$

Its discriminant is

$$\begin{aligned}
 D &= (2 - \beta - 2\eta\beta\sigma_i^2)^2 - 4(1 - \beta) \\
 &= \beta(4\beta\eta^2\sigma_i^4 + 4\beta\eta\sigma_i^2 + \beta - 8\eta\sigma_i^2),
 \end{aligned} \tag{72}$$

Using the characteristic polynomial and discriminant above, find the two eigenvalues $\lambda_{1,i}$ and $\lambda_{2,i}$ of R_i in terms of η , β , σ_i , and D . When are they distinct and real, repeated and real, or non-real complex conjugates? State the condition on D in each case.

Solution: Applying the quadratic formula gives

$$\lambda_{1,i} = \frac{2 - \beta - 2\eta\beta\sigma_i^2 - \sqrt{D}}{2}, \quad \lambda_{2,i} = \frac{2 - \beta - 2\eta\beta\sigma_i^2 + \sqrt{D}}{2}.$$

The eigenvalues are distinct and real when $D > 0$, repeated and real when $D = 0$, and non-real complex conjugates when $D < 0$.

(d) **For the case when the eigenvalues are repeated, which learning rates η give this case, and are the eigenvalues stable (strictly inside the unit circle)? What is the highest such learning rate as a function of β and σ_i ?**

Solution: Setting $D = 0$ gives two learning rates:

$$\frac{2 - \beta - 2\sqrt{1 - \beta}}{2\beta\sigma_i^2} \text{ or } \frac{2 - \beta + 2\sqrt{1 - \beta}}{2\beta\sigma_i^2}$$

Therefore, the highest learning rate is $\frac{2 - \beta + 2\sqrt{1 - \beta}}{2\beta\sigma_i^2}$.

At the lower and upper learning rates, respectively, the repeated root is

$$\lambda_{1,i} = \lambda_{2,i} = 1 - \beta\eta\sigma_i^2 - \frac{\beta}{2} = \pm\sqrt{1 - \beta} \tag{73}$$

Both are strictly inside the unit circle because $0 < \beta < 1$. Their magnitude depends only on β , but the learning rates that produce these repeated roots do depend on σ_i .

(e) **For the case when the eigenvalues are distinct and real, what range of η keeps both eigenvalues strictly inside the unit circle? Express your answer in terms of β and σ_i .**

Hint: Use $f(1)$, $f(-1)$, and $\lambda_{1,i}\lambda_{2,i} = 1 - \beta \in (0, 1)$ to locate the two roots inside $(-1, 1)$, then combine this with $D > 0$.

Solution: At first, to have the different real roots, the discriminant Eq.(72), D should be strictly positive:

$$D = \beta(4\beta\eta^2\sigma_i^4 + 4\beta\eta\sigma_i^2 + \beta - 8\eta\sigma_i^2) > 0 \tag{74}$$

If we solve Eq.(74) with respect to η ,

$$\eta < \frac{2 - \beta - 2\sqrt{1 - \beta}}{2\beta\sigma_i^2} \text{ or } \eta > \frac{2 - \beta + 2\sqrt{1 - \beta}}{2\beta\sigma_i^2} \quad (75)$$

The roots have product $1 - \beta \in (0, 1)$, so they have the same sign and cannot both have magnitude at least 1. Therefore, both roots lie in $(-1, 1)$ exactly when $f(1) > 0$ and $f(-1) > 0$. First compute $f(1)$:

$$\begin{aligned} f(1) &= 1 - \left(2 - \beta - 2\eta\beta\sigma_i^2\right) + 1 - \beta \\ &= 2\eta\beta\sigma_i^2 > 0 \end{aligned}$$

This is positive under our assumptions $\eta > 0$, $\beta > 0$, and $\sigma_i > 0$.

Now, let's look at $f(-1)$.

$$\begin{aligned} f(-1) &= 1 + \left(2 - \beta - 2\eta\beta\sigma_i^2\right) + 1 - \beta \\ &= 4 - 2\beta - 2\eta\beta\sigma_i^2 > 0 \end{aligned}$$

Thus,

$$0 < \eta < \frac{4 - 2\beta}{2\beta\sigma_i^2} \quad (76)$$

Let's compare Eq.(75) and Eq.(76). Since

$$4 - 2\beta \geq 2 - \beta + 2\sqrt{1 - \beta}, \text{ for all } \beta \in [0, 1]$$

the complete range with two distinct, stable real roots is

$$0 < \eta < \frac{2 - \beta - 2\sqrt{1 - \beta}}{2\beta\sigma_i^2} \text{ or } \frac{2 - \beta + 2\sqrt{1 - \beta}}{2\beta\sigma_i^2} < \eta < \frac{4 - 2\beta}{2\beta\sigma_i^2}$$

(f) **Given result: complex eigenvalues.** For $0 < \beta < 1$ and $\sigma_i > 0$, define

$$\eta_- = \frac{2 - \beta - 2\sqrt{1 - \beta}}{2\beta\sigma_i^2}, \quad \eta_+ = \frac{2 - \beta + 2\sqrt{1 - \beta}}{2\beta\sigma_i^2}.$$

Solving $D < 0$ gives the complete range with non-real complex eigenvalues:

$$\eta_- < \eta < \eta_+.$$

In this range, the eigenvalues are complex conjugates and their product is $1 - \beta$. Therefore,

$$|\lambda_{1,i}|^2 = \lambda_{1,i}\lambda_{2,i} = 1 - \beta, \quad |\lambda_{1,i}| = |\lambda_{2,i}| = \sqrt{1 - \beta} < 1.$$

Thus every learning rate in this open interval gives a stable system. There is no largest learning rate with non-real eigenvalues: η must stay strictly below η_+ . At either endpoint the eigenvalues are repeated and real. At η_- they both equal $+\sqrt{1 - \beta}$, and at η_+ they both equal $-\sqrt{1 - \beta}$; these endpoints are also stable.

The geometric convergence factor of a mode is $r_i = \max(|\lambda_{1,i}|, |\lambda_{2,i}|)$. Since the product of its eigenvalues is $1 - \beta$, we always have $r_i \geq \sqrt{1 - \beta}$. Repeated or complex-conjugate eigenvalues attain this smallest possible factor. The overall factor is $r = \max_i r_i$, since the slowest singular direction determines the long-run rate. This is a geometric factor, not an exact finite-step error: repeated roots can also introduce a factor of t multiplying r_i^t . These results are provided for use in part (g).

- (g) Consider a problem with two singular values $\sigma_{\min} = 1$ and $\sigma_{\max} = 2$, and fix $\beta = \frac{3}{4}$. Thus $\sigma_{\min}^2 = 1$, $\sigma_{\max}^2 = 4$, and $\sqrt{1 - \beta} = \frac{1}{2}$.

Find the full interval of learning rates η that gives the fastest geometric convergence for gradient descent with momentum, and give one convenient choice from that interval. What are the best geometric convergence factors for momentum and for ordinary gradient descent?

For a simple iteration comparison, use the geometric factor r^T , where r is the method's best convergence factor. **For each method, find the smallest integer T such that $r^T \leq \frac{1}{4}$.** Small integer powers suffice. This comparison uses the geometric factors; exact finite-step parameter errors can also depend on transient effects.

Solution: For each singular direction, the smallest possible convergence factor is $\sqrt{1 - \beta} = \frac{1}{2}$. It is attained when the roots are repeated or complex, so the allowed learning rates for that direction satisfy

$$\frac{(1 - \frac{1}{2})^2}{2(\frac{3}{4})\sigma_i^2} \leq \eta \leq \frac{(1 + \frac{1}{2})^2}{2(\frac{3}{4})\sigma_i^2}, \quad \text{or} \quad \frac{1}{6\sigma_i^2} \leq \eta \leq \frac{3}{2\sigma_i^2}.$$

For the two directions, these intervals are

$$\begin{aligned} \sigma_i^2 = 1 : \quad & \frac{1}{6} \leq \eta \leq \frac{3}{2}, \\ \sigma_i^2 = 4 : \quad & \frac{1}{24} \leq \eta \leq \frac{3}{8}. \end{aligned}$$

Their intersection is

$$\boxed{\frac{1}{6} \leq \eta \leq \frac{3}{8}}.$$

Every learning rate in this interval attains the optimal factor $r_{\text{momentum}} = \frac{1}{2}$. For example, $\eta = \frac{1}{3}$ lies strictly inside the interval and gives complex eigenvalues in both directions. Optimality follows from $|\lambda_{1,i}\lambda_{2,i}| = \frac{1}{4}$: both magnitudes cannot be smaller than $\frac{1}{2}$.

For ordinary gradient descent, the optimal learning rate and factor are

$$\eta_{\text{GD}}^* = \frac{1}{1 + 4} = \frac{1}{5}, \quad r_{\text{GD}} = \frac{4 - 1}{4 + 1} = \frac{3}{5}.$$

To compare the geometric factors, momentum needs $T = 2$, since $(\frac{1}{2})^1 > \frac{1}{4}$ and $(\frac{1}{2})^2 = \frac{1}{4}$. Ordinary GD needs $T = 3$, since

$$\left(\frac{3}{5}\right)^2 = \frac{9}{25} > \frac{1}{4}, \quad \left(\frac{3}{5}\right)^3 = \frac{27}{125} < \frac{1}{4}.$$

These counts compare geometric factors. A spectral radius alone does not give a bound with unit prefactor on the momentum parameter error; repeated roots at the interval endpoints can also introduce a factor of T in the error's transient behavior.

- (h) The two questions below use the Fall 2026 **momentum notebook**. It uses the same singular values and momentum parameter as part (g): $\sigma_{\min} = 1$, $\sigma_{\max} = 2$, and $\beta = \frac{3}{4}$, with $\eta_{\text{GD}} = \frac{1}{5}$ and $\eta_{\text{momentum}} = \frac{1}{3}$. Run the notebook and use the plots to answer the following questions. You do not need to submit the `ipynb` file. These are also the two questions in *Coding: Watching Momentum Work*; submit your answers only once.

How does the singular value σ_i influence the gradients and the parameter updates along direction i ?

Solution: The gradient along direction i is $2\sigma_i^2(w_t[i] - w^*[i])$. For the same coordinate error, direction 0 ($\sigma = 2$) has four times the gradient of direction 1 ($\sigma = 1$). With ordinary GD at $\eta = \frac{1}{5}$, the two error multipliers are $-\frac{3}{5}$ and $+\frac{3}{5}$: direction 0 oscillates across the optimum, while direction 1 approaches from one side, and both magnitudes shrink by $\frac{3}{5}$ per step. With momentum at the given settings, both directions have complex-conjugate eigenvalues, so either error can change sign.

- (i) Question: Comparing gradient descent and gradient descent with momentum, **which one converges faster for this task? Why?**

Solution: For these settings, momentum has the faster asymptotic geometric factor: $\frac{1}{2}$ versus $\frac{3}{5}$ for ordinary GD. The averaged gradient changes how the two directions evolve and gives a faster shrinking error envelope. This does not require momentum to have a smaller error on every early step or to decrease the loss monotonically. Both methods start from $w_0 = a_0 = 0$ in the notebook; both first reach an actual relative parameter error of at most $\frac{1}{4}$ at iteration 3, even though the geometric-factor comparison in part (g) gives 2 versus 3 steps.

Solution: A geometric illustration of the eigenvalue analysis. This illustration uses the Fall 2026 **companion notebook** or **companion video**.

The characteristic polynomial gives

$$\lambda_{1,i}\lambda_{2,i} = 1 - \beta, \quad \frac{\lambda_{1,i} + \lambda_{2,i}}{2} = 1 - \frac{\beta}{2} - \beta\eta\sigma_i^2.$$

For complex roots, the pair lies on the circle of radius $\sqrt{1 - \beta}$. As η increases, their midpoint moves left along the real axis. The roots coincide when the midpoint reaches either end of the circle's horizontal diameter; beyond those points they are distinct and real, with one root outside the circle. Figure 1 illustrates this motion.

In part (g), $\beta = \frac{3}{4}$, so the circle has radius $\frac{1}{2}$. Keeping both singular directions on their circles gives $\frac{1}{6} \leq \eta \leq \frac{3}{8}$. The convenient interior choice $\eta = \frac{1}{3}$ gives midpoints $+\frac{3}{8}$ and $-\frac{3}{8}$ for $\sigma_i^2 = 1$ and 4, respectively; both lie strictly between $-\frac{1}{2}$ and $+\frac{1}{2}$. Thus both modes have convergence factor $\frac{1}{2}$, compared with $\frac{3}{5}$ for optimally tuned ordinary gradient descent.

Figure 1: Eigenvalues for $\beta = \frac{3}{4}$ and singular values 2 and 1. The first two learning rates show distinct real roots and the lower boundary of the common optimal interval.

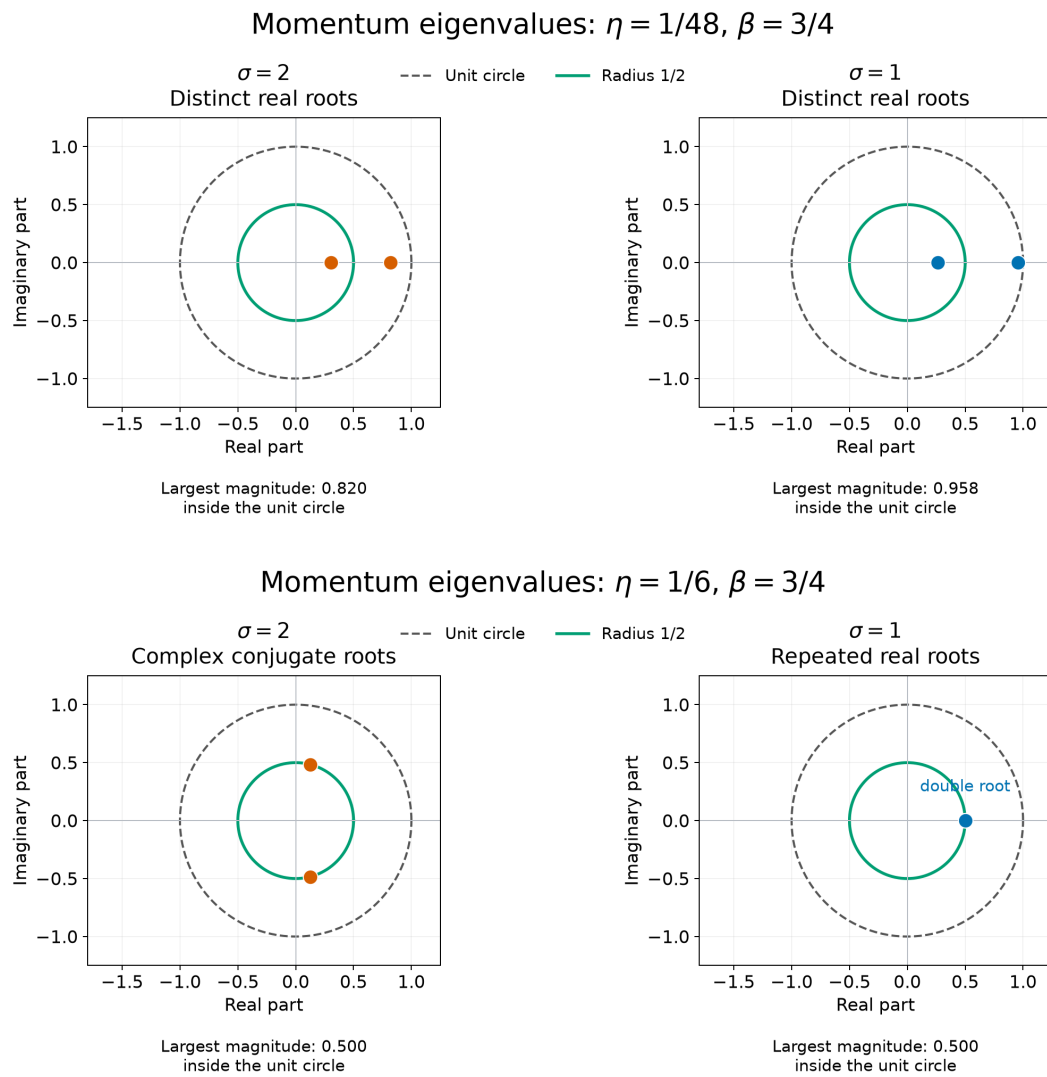
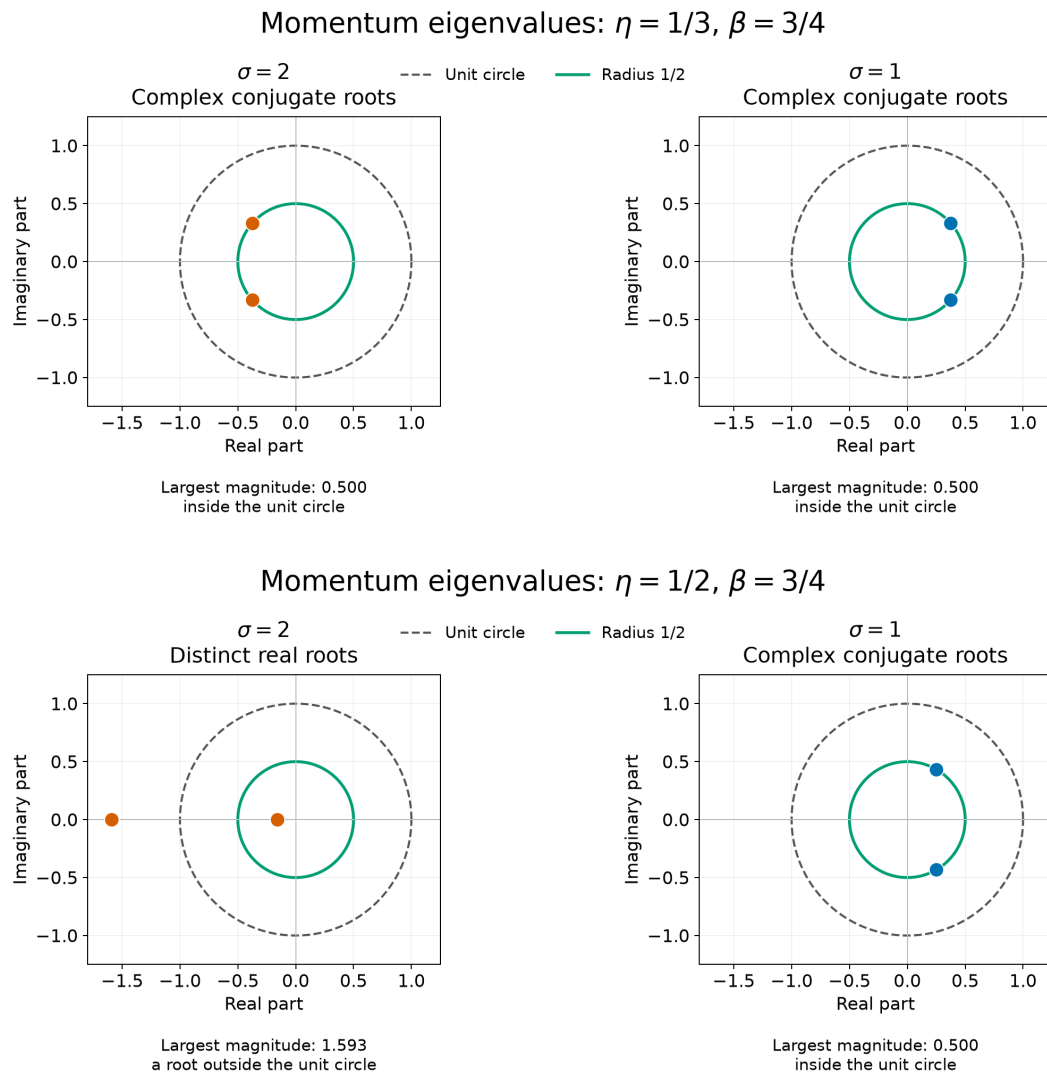


Figure 2: The same two directions at $\eta = \frac{1}{3}$ (both complex) and $\eta = \frac{1}{2}$ (the large-singular-value direction is unstable).



3. Optimizers

Here f_t is the loss on the minibatch used at step t , and g_t is its gradient. The momentum parameter β_1 below weights the previous average; it corresponds to $1 - \beta$ in the preceding momentum problem. In Adam, squares, square roots, and division act elementwise. Use a small constant $\epsilon > 0$ in the denominator to avoid division by zero; bias correction is omitted in this exercise.

Algorithm 1 SGD with Momentum

```

1: Given  $\eta = 0.001, \beta_1 = 0.9$ 
2: Initialize:
3:   time step  $t \leftarrow 0$ 
4:   parameter  $\theta_{t=0} \in \mathbb{R}^n, m_0 \leftarrow 0$ 
5: Repeat
6:    $t \leftarrow t + 1$ 
7:    $g_t \leftarrow \nabla f_t(\theta_{t-1})$ 
8:    $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$ 
9:    $\theta_t \leftarrow \theta_{t-1} - \eta m_t$ 
10: Until the stopping condition is met
```

Algorithm 2 Adam Optimizer (without bias correction)

```

1: Given  $\eta = 0.001, \beta_1 = 0.9, \beta_2 = 0.999$ 
2: Initialize time step  $t \leftarrow 0$ , parameter  $\theta_{t=0} \in \mathbb{R}^n$ ,
    $m_{t=0} \leftarrow 0, v_{t=0} \leftarrow 0$ 
3: Repeat
4:    $t \leftarrow t + 1$ 
5:    $g_t \leftarrow \nabla f_t(\theta_{t-1})$ 
6:    $m_t \leftarrow \text{---(A)---}$ 
7:    $v_t \leftarrow \text{---(B)---}$ 
8:    $\theta_t \leftarrow \theta_{t-1} - \eta \cdot \frac{m_t}{\sqrt{v_t} + \epsilon}$ 
9: Until the stopping condition is met
```

(a) Complete part (A) and (B) in the pseudocode of Adam.

Solution:

$$m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

(b) For this part, use **plain SGD without momentum or adaptive rescaling**, with learning rate $\eta > 0$ and regularization strength $\lambda \geq 0$. Establish the relationship between

- **L2 regularization** adds a squared-weight penalty to the minibatch loss:

$$f_t^{reg}(\theta) = f_t(\theta) + \frac{\lambda}{2} \|\theta\|_2^2$$

- **Weight decay** multiplies the weights by $1 - \gamma$ at each update:

$$\theta_{t+1} = (1 - \gamma)\theta_t - \eta \nabla f_t(\theta_t)$$

Setting $\gamma = 0$ recovers plain SGD.

Show that SGD with weight decay using the original loss $f_t(\theta)$ is equivalent to regular SGD on the L2-regularized loss $f_t^{reg}(\theta)$ when γ is chosen correctly, and find such a γ in terms of λ and η .

Solution: Differentiating the regularized loss gives $\nabla f_t^{reg}(\theta) = \nabla f_t(\theta) + \lambda\theta$. Therefore,

$$\begin{aligned}
 \theta_{t+1} &= \theta_t - \eta \nabla f_t^{reg}(\theta_t) \\
 &= \theta_t - \eta \nabla f_t(\theta_t) - \eta \lambda \theta_t \\
 &= (1 - \eta \lambda) \theta_t - \eta \nabla f_t(\theta_t).
 \end{aligned}$$

Thus the updates agree when $\gamma = \lambda\eta$, using the same initialization, learning rate, and minibatches. This equivalence is for plain SGD; the penalty gradient also enters the running averages in momentum or Adam, so the same argument does not apply to them.

4. Regularization and Instance Noise

Say we have m labeled data points $(\mathbf{x}_i, y_i)_{i=1}^m$, where each $\mathbf{x}_i \in \mathbb{R}^n$ and each $y_i \in \mathbb{R}$. We perform data augmentation by adding some noise to each vector every time we use it in SGD. This means for all points i , we have a true input $\mathbf{x}_i$ and add noise $\mathbf{N}_i$ to get the effective random input seen by SGD:

$$\tilde{\mathbf{X}}_i = \mathbf{x}_i + \mathbf{N}_i$$

The i.i.d. random noise vectors $\mathbf{N}_i$ are distributed as $\mathbf{N}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_n)$, with $\sigma^2 \geq 0$. The clean data and labels are fixed, and fresh noise is drawn independently each time a datapoint is used. Let $X \in \mathbb{R}^{m \times n}$ be the clean data matrix with rows $\mathbf{x}_i^\top$.

Arrange the augmented inputs as the rows of $\tilde{X} \in \mathbb{R}^{m \times n}$, so row i is $(\mathbf{x}_i + \mathbf{N}_i)^\top$, and let $\mathbf{y} = (y_1, \dots, y_m)^\top$.

One way of thinking about what SGD might do is to consider learning weights that minimize the *expected* least squares objective for the **noisy** data matrix:

$$\underset{\mathbf{w}}{\operatorname{argmin}} \mathbb{E}[\|\tilde{X}\mathbf{w} - \mathbf{y}\|^2] \quad (77)$$

(a) **Show that this problem (77) is equivalent to a regularized least squares problem:**

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{m} \|X\mathbf{w} - \mathbf{y}\|^2 + \lambda \|\mathbf{w}\|^2 \quad (78)$$

You will need to determine the value of λ .

Hint: write the squared norm of a vector as an inner product, expand, and apply linearity of expectation.

Solution: For a fixed $\mathbf{w}$, let $r_i = \mathbf{x}_i^\top \mathbf{w} - y_i$. Using $\mathbb{E}[\mathbf{N}_i] = \mathbf{0}$ and $\mathbb{E}[\mathbf{N}_i \mathbf{N}_i^\top] = \sigma^2 I_n$, we obtain

$$\begin{aligned} \mathbb{E}[\|\tilde{X}\mathbf{w} - \mathbf{y}\|^2] &= \sum_{i=1}^m \mathbb{E}[(r_i + \mathbf{N}_i^\top \mathbf{w})^2] \\ &= \sum_{i=1}^m \left[r_i^2 + 2r_i \mathbb{E}[\mathbf{N}_i^\top \mathbf{w}] + \mathbf{w}^\top \mathbb{E}[\mathbf{N}_i \mathbf{N}_i^\top] \mathbf{w} \right] \\ &= \sum_{i=1}^m \left[r_i^2 + \sigma^2 \|\mathbf{w}\|^2 \right] \\ &= \|X\mathbf{w} - \mathbf{y}\|^2 + m\sigma^2 \|\mathbf{w}\|^2 \end{aligned}$$

Dividing by the positive constant m preserves the minimizers and matches (78), giving $\lambda = \sigma^2$.

Now consider a simplified example where we only have a single scalar datapoint $x \in \mathbb{R}$ with $x \neq 0$ and its corresponding label $y \in \mathbb{R}$. We are going to analyze this in the context of gradient descent. At update $t = 0, 1, \dots$, from w_t to w_{t+1} , we use a noisy datapoint $\tilde{X}_t = x + N_t$ which is generated by adding different random noise values $N_t \sim \mathcal{N}(0, \sigma^2)$ to our underlying data point x . The noise values for each iteration of gradient descent are i.i.d., so the fresh N_t is independent of w_t . The loss used for this update is $\mathcal{L}_t(w) = \frac{1}{2}(\tilde{X}_t w - y)^2$. We initialize our weight to be $w_0 = 0$ and use a constant learning rate $\eta > 0$. Expectations below are over the noise draws. We analyze the mean $\mathbb{E}[w_t]$; its convergence alone does not establish convergence of individual noisy runs.

- (b) Let w_t be the weight learned after the t -th iteration of gradient descent with data augmentation. **Write the gradient descent recurrence relation between $\mathbb{E}[w_{t+1}]$ and $\mathbb{E}[w_t]$ in terms of x , σ^2 , y , and learning rate η .**

Solution: The gradient for the fresh noisy input is

$$\frac{\partial \mathcal{L}_t}{\partial w} = (w(x + N_t) - y)(x + N_t).$$

Substituting into the update gives

$$\begin{aligned} w_{t+1} &= w_t - \eta(w_t(x + N_t) - y)(x + N_t) \\ &= w_t(1 - \eta(x + N_t)^2) + \eta y(x + N_t). \end{aligned}$$

Taking expectations, and using the fact that N_t is independent of w_t since the noise values are i.i.d., we get:

$$\mathbb{E}[w_{t+1}] = \mathbb{E}[w_t](1 - \eta(x^2 + \sigma^2)) + \eta yx$$

- (c) **For what positive learning rates η does the mean recurrence converge to its fixed point from any initialization?**

Solution: Subtracting the fixed point from the mean recurrence multiplies the error by $1 - \eta(x^2 + \sigma^2)$ each step. For this error to decay from any initialization, that coefficient must lie strictly between -1 and 1 .

$$-1 < 1 - \eta(x^2 + \sigma^2) \text{ and } 1 - \eta(x^2 + \sigma^2) < 1$$

This implies

$$0 < \eta < \frac{2}{x^2 + \sigma^2}$$

- (d) Assuming η is in the range from part (c), **what would we expect $\mathbb{E}[w_t]$ to converge to as $t \rightarrow \infty$? How does this differ from the optimal value of w if there were no noise being used to augment the data?**

(HINT: You can also use this to help check your work for part (a).)

Solution: One way of doing this is to take an expectation of the gradient and set it to 0. Using the fact that $\mathbb{E}[N_t] = 0$ and $\mathbb{E}[N_t^2] = \sigma^2$,

$$w^*(x^2 + \sigma^2) - yx = 0$$

$$w^* = \frac{yx}{x^2 + \sigma^2}.$$

Equivalently,

$$w^* = \left(\frac{y}{x}\right) \frac{1}{1 + \frac{\sigma^2}{x^2}}.$$

The optimal value of w without data-augmenting noise is $w = \frac{y}{x}$. This means the optimal expected value with noise augmentation is scaled down by a factor of $\frac{1}{1 + \frac{\sigma^2}{x^2}}$. This is the scalar version of ridge regression's shrinkage of fitted coefficients. When σ is small relative to $|x|$, the learned weight barely changes. When σ is large relative to $|x|$, the learned weight shrinks significantly towards zero.

5. Coding: Watching Momentum Work

This notebook shows how different singular directions converge using the exact least-squares example from the momentum analysis problem: singular values 2 and 1, momentum parameter $\beta = \frac{3}{4}$, and learning rates $\eta_{\text{GD}} = \frac{1}{5}$ and $\eta_{\text{momentum}} = \frac{1}{3}$. Signed errors and gradients show how each singular direction behaves, while the loss and error plots compare the two methods over time.

Open the Fall 2026 `q_sgd_momentum_analysis.ipynb` and run all cells. The notebook includes the data and optimizer implementations. You do not need to submit the `.ipynb` file; answer the questions below in your write-up. These repeat parts (h) and (i) of *Accelerating Gradient Descent with Momentum*. Submit each answer only once; a reference to your earlier answers is sufficient here.

If you would like to see the eigenvalues move as the learning rate changes, there is a companion [notebook](#) and a [short video](#) using the same numerical values.

- (a) **How does σ_i influence the gradients and the parameter updates along direction i ?**

Solution: The gradient along direction i is $2\sigma_i^2(w_t[i] - w^*[i])$. For equal coordinate errors, dimension 0 ($\sigma = 2$) has four times the gradient of dimension 1 ($\sigma = 1$). Under GD at $\eta = \frac{1}{5}$, the errors are multiplied by $-\frac{3}{5}$ and $+\frac{3}{5}$, respectively. The first direction alternates across the optimum and the second stays on one side; both magnitudes shrink at the same rate. Under the chosen momentum settings, either direction can change sign because both mode matrices have complex-conjugate roots.

- (b) **Comparing gradient descent with and without momentum, which converges faster on this task, and why?**

Solution: With these learning rates, momentum has the faster asymptotic geometric factor, $\frac{1}{2}$ versus $\frac{3}{5}$ for ordinary GD. Its averaged gradient changes the recurrence in each direction, giving a faster shrinking error envelope. Momentum need not have a smaller error at every early step, and its loss can briefly increase. The notebook distinguishes these measured trajectories from the geometric factors in the written analysis.

Contributors:

- Matteo Guarrera.
- Mert Cemri.
- Sizhe Chen.
- Anant Sahai.
- Gireeja Ranade.
- Yaodong Yu.
- Suhong Moon.
- Gabriel Goh.
- Peter Wang.
- Yuxi Liu.
- Kevin Li.
- Saagar Sanghavi.