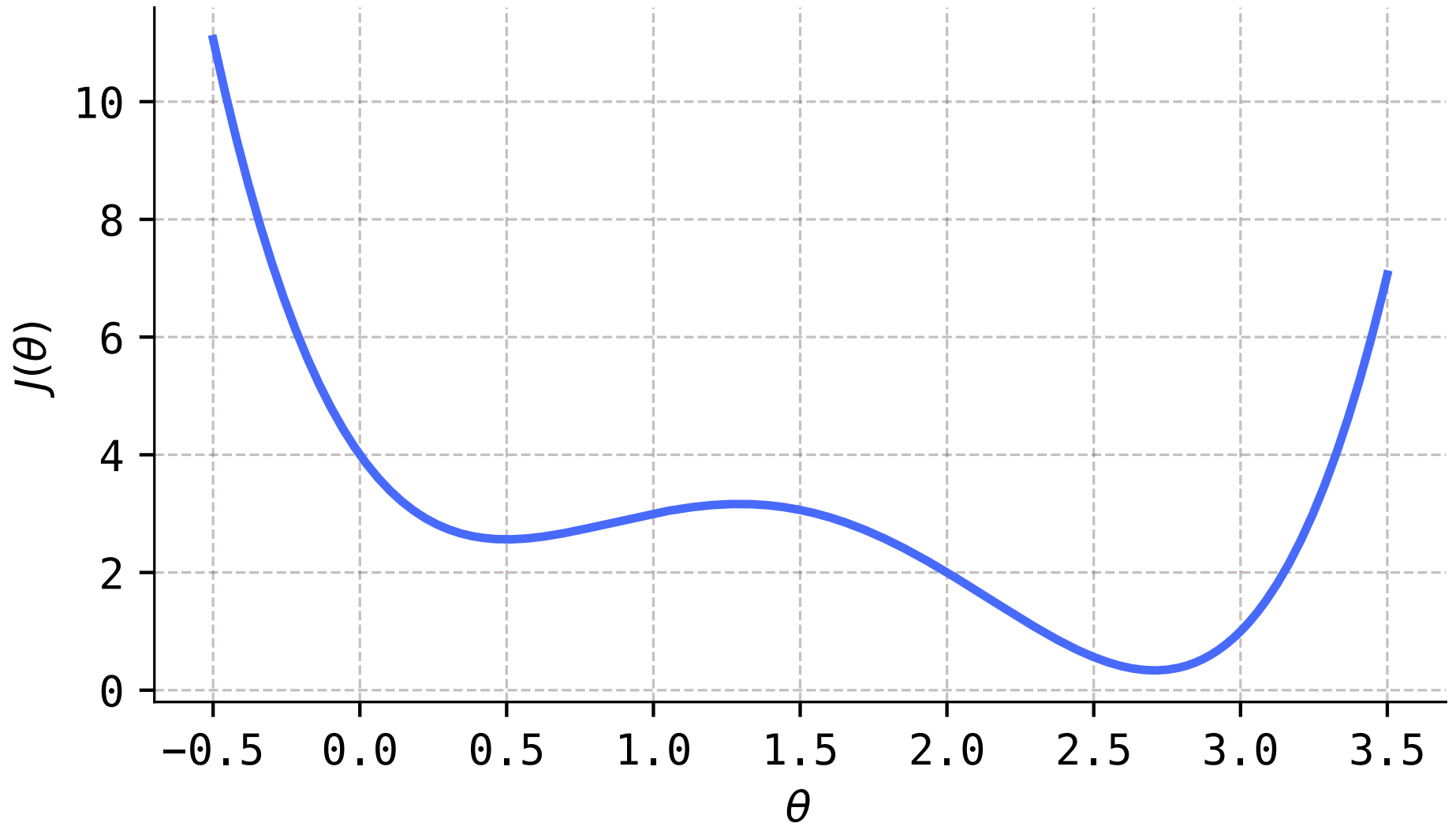
11. Optimization

11.1. Gradient descent in one variable


10
8
6
4
2
0
$J(\theta)$
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
θ

Definition 11.1

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function. A vector $\boldsymbol{\theta}^\star$ is a *local minimizer* of $J(\boldsymbol{\theta})$ provided that

$$J(\boldsymbol{\theta}^\star) \leq J(\boldsymbol{\theta})$$

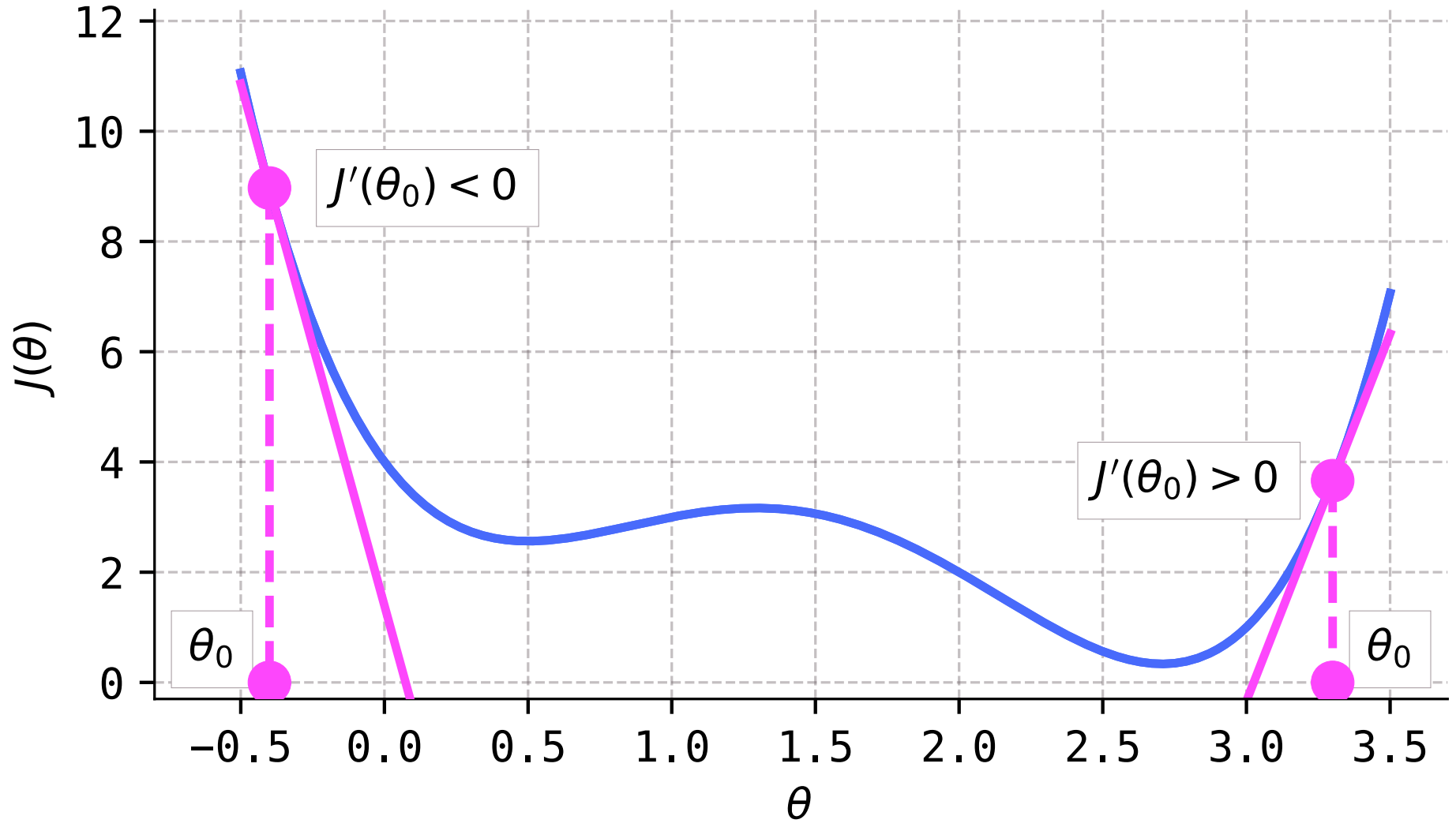
for all $\boldsymbol{\theta}$ in a neighborhood of $\boldsymbol{\theta}^\star$; if this inequality holds for *all* $\boldsymbol{\theta}$, then $\boldsymbol{\theta}^\star$ is called a *global minimizer* of $J(\boldsymbol{\theta})$. If we flip the inequality the other direction, then we obtain the definitions of *local* and *global maximizers*. Collectively, local and global minimizers and maximizers of $J(\boldsymbol{\theta})$ are called *extremizers*, and the values $J(\boldsymbol{\theta}^\star)$ of the function where $\boldsymbol{\theta}^\star$ is an extremizer are called *extrema* or *extreme values*.

Algorithm 11.1 (Single-variable gradient descent)

Input: A differentiable objective function $J : \mathbb{R} \to \mathbb{R}$, an initial guess $\theta_0 \in \mathbb{R}$ for a local minimizer $\theta^\star$, a learning rate $\alpha > 0$, and the number N of gradient steps.

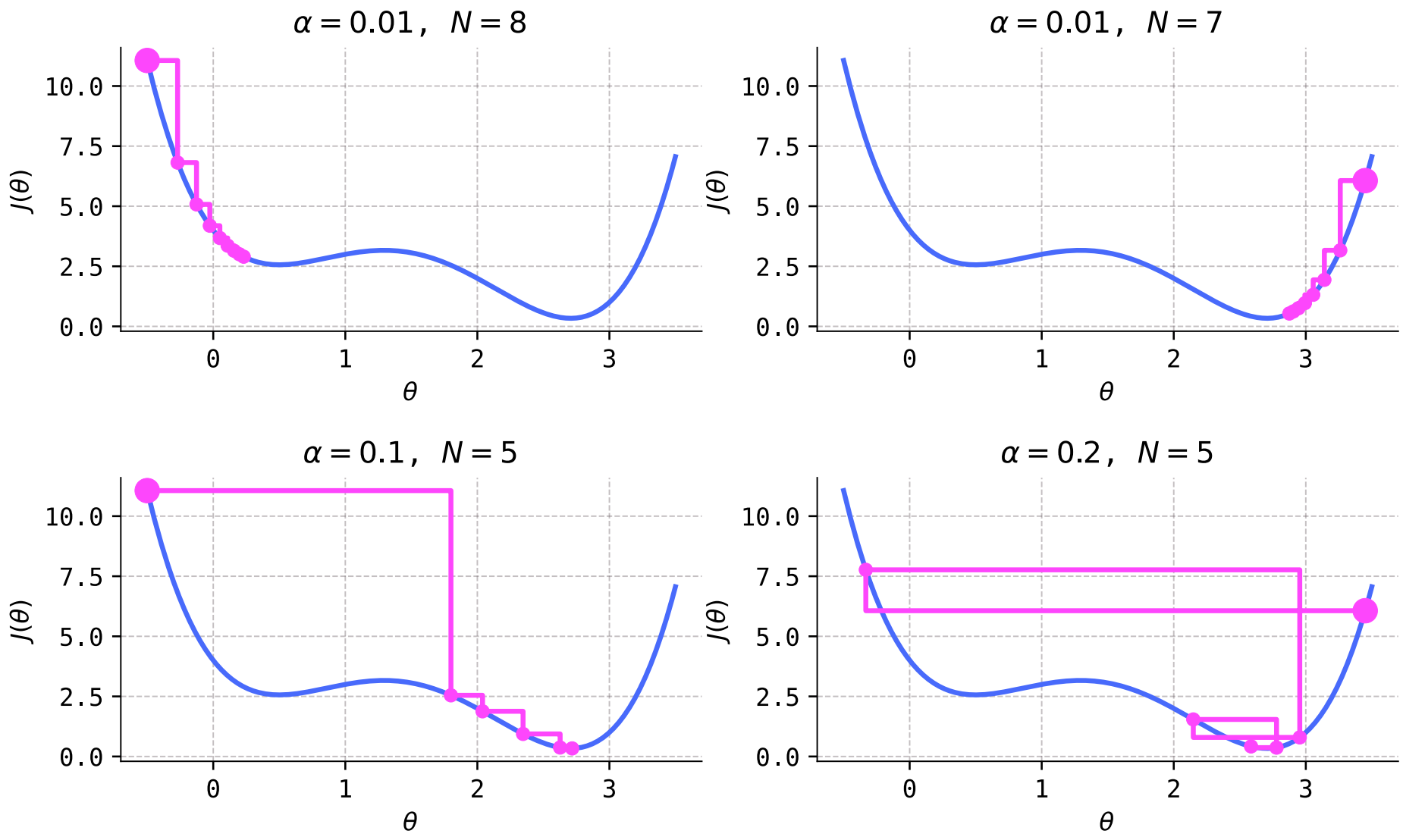
Output: An approximation to a local minimizer $\theta^\star$.

$\theta := \theta_0$
For t from 0 to $N - 1$, do:
    $\theta := \theta - \alpha J'(\theta)$
Return θ


12
10
8
6
4
2
0
$J(\theta)$
$J'(\theta_0) < 0$
$J'(\theta_0) > 0$
θ_0
θ_0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
θ

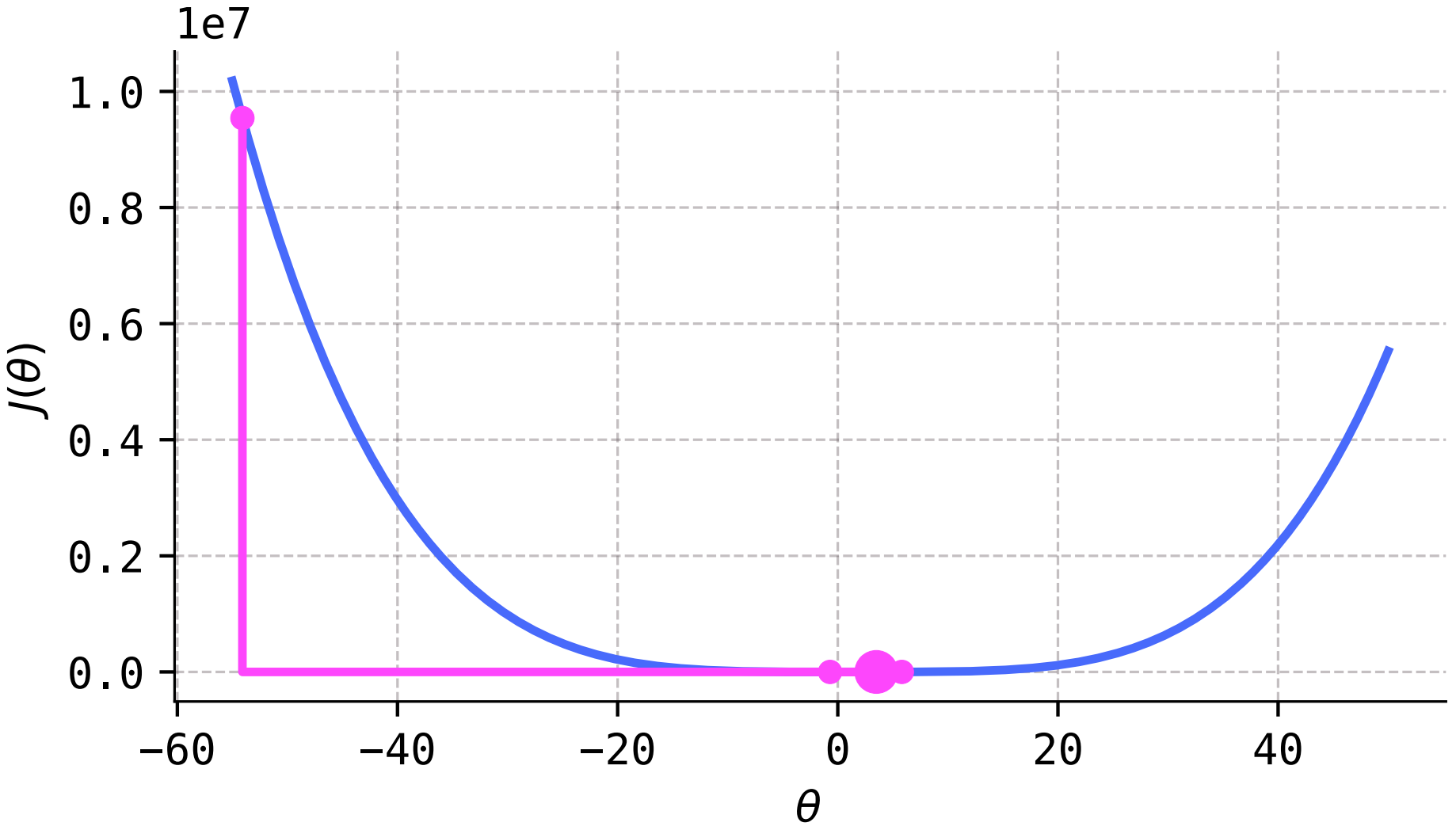
Observation 11.1

- The negative derivative $-J'(\theta)$ always "points downhill."
- When the gradient descent algorithm works, it locates a minimizer by following the negative derivative "downhill."


$\alpha = 0.01,\ N = 8$
$\alpha = 0.01,\ N = 7$
$\alpha = 0.1,\ N = 5$
$\alpha = 0.2,\ N = 5$
$J(\theta)$
θ
0.0
2.5
5.0
7.5
10.0
0
1
2
3

Problem Prompt

Do problem 1 on the worksheet.


1e7
1.0
0.8
0.6
0.4
0.2
0.0
J(θ)
−60
−40
−20
0
20
40
θ

Problem Prompt

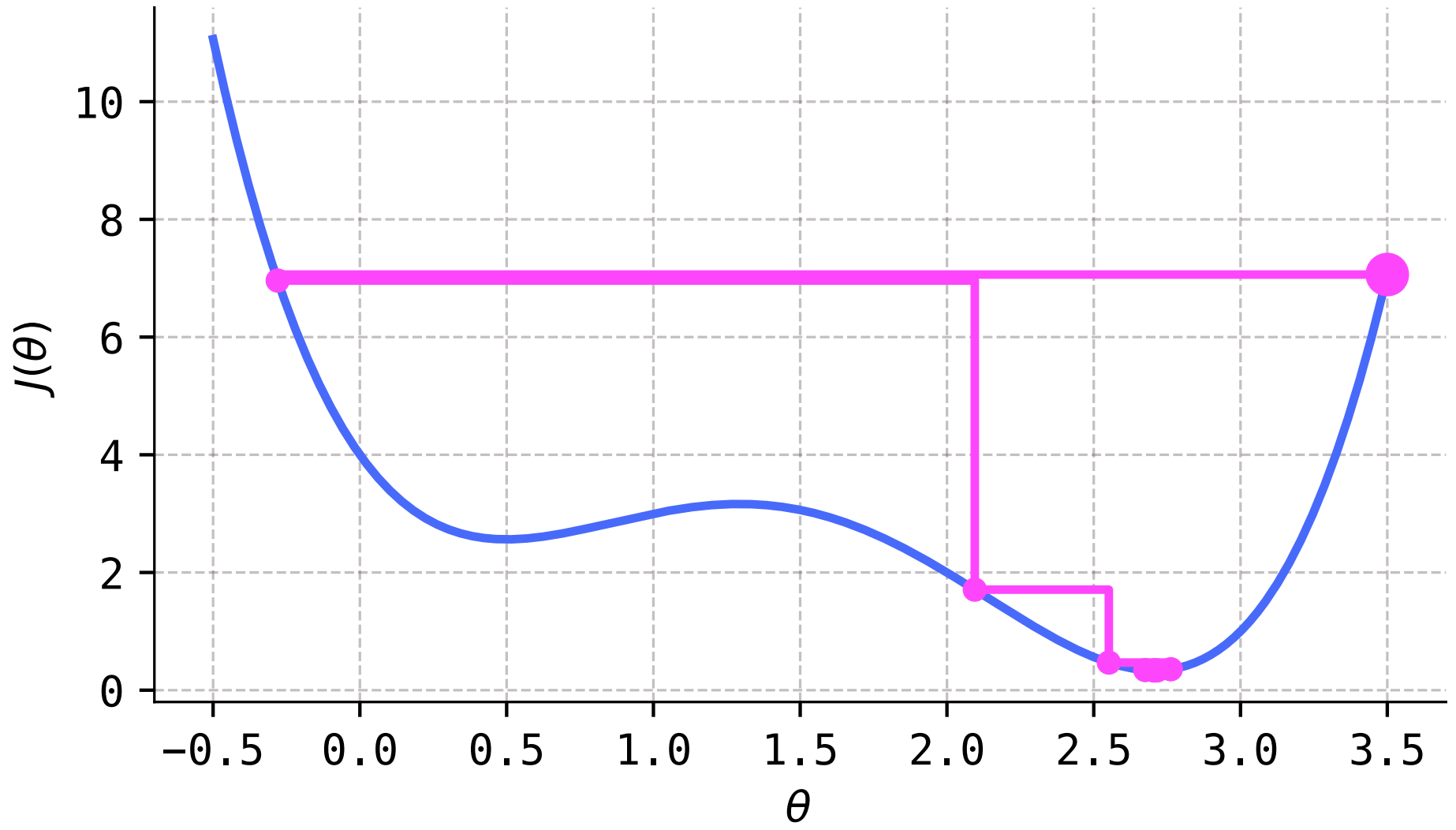
Do problems 2 and 3 on the worksheet.

Algorithm 11.2 (Single-variable gradient descent with learning rate decay)

Input: A differentiable objective function $J : \mathbb{R} \to \mathbb{R}$, an initial guess $\theta_0 \in \mathbb{R}$ for a local minimizer $\theta^\star$, a learning rate $\alpha > 0$, a decay rate $\beta \in [0, 1)$, and the number N of gradient steps.

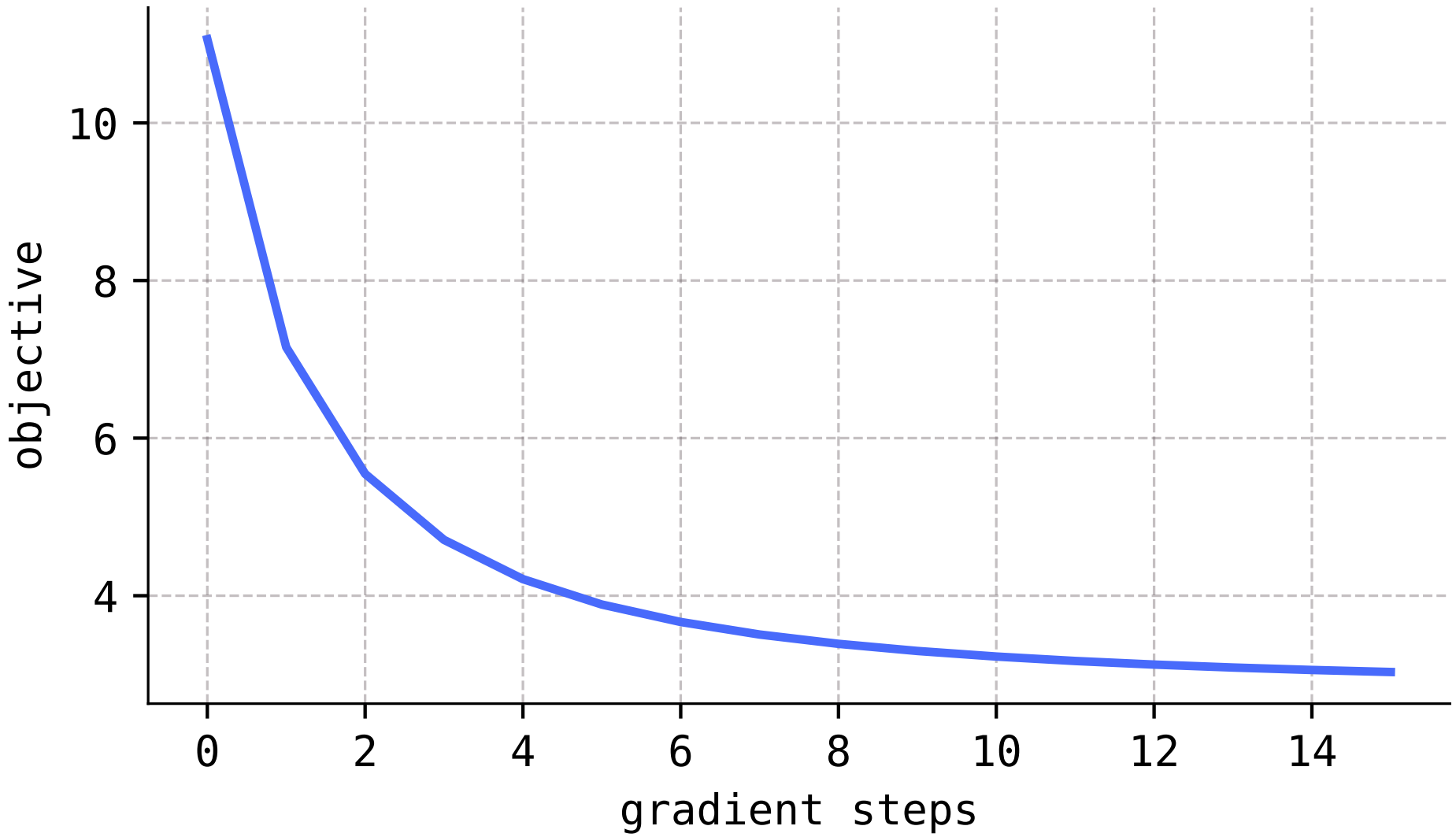
Output: An approximation to a local minimizer $\theta^\star$.

$\theta := \theta_0$
For t from 0 to $N - 1$, do:
    $\theta := \theta - \alpha(1 - \beta)^{t+1} J'(\theta)$
Return θ

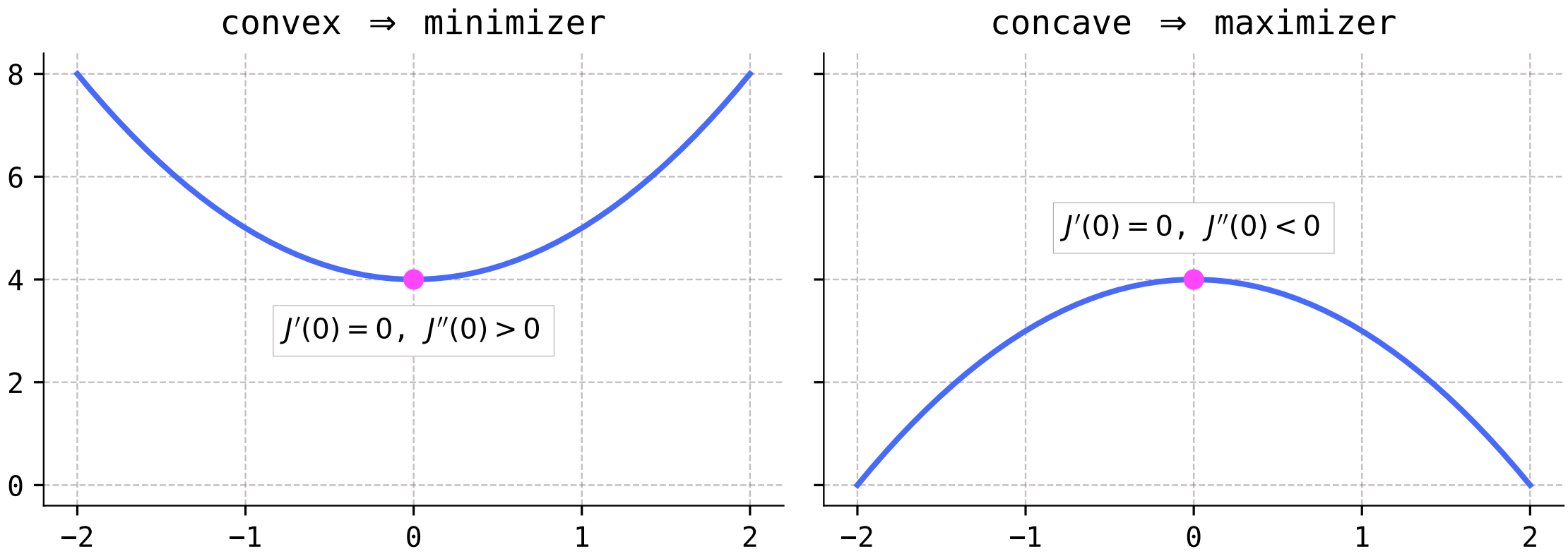

10
8
6
4
2
0
$J(\theta)$
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
θ

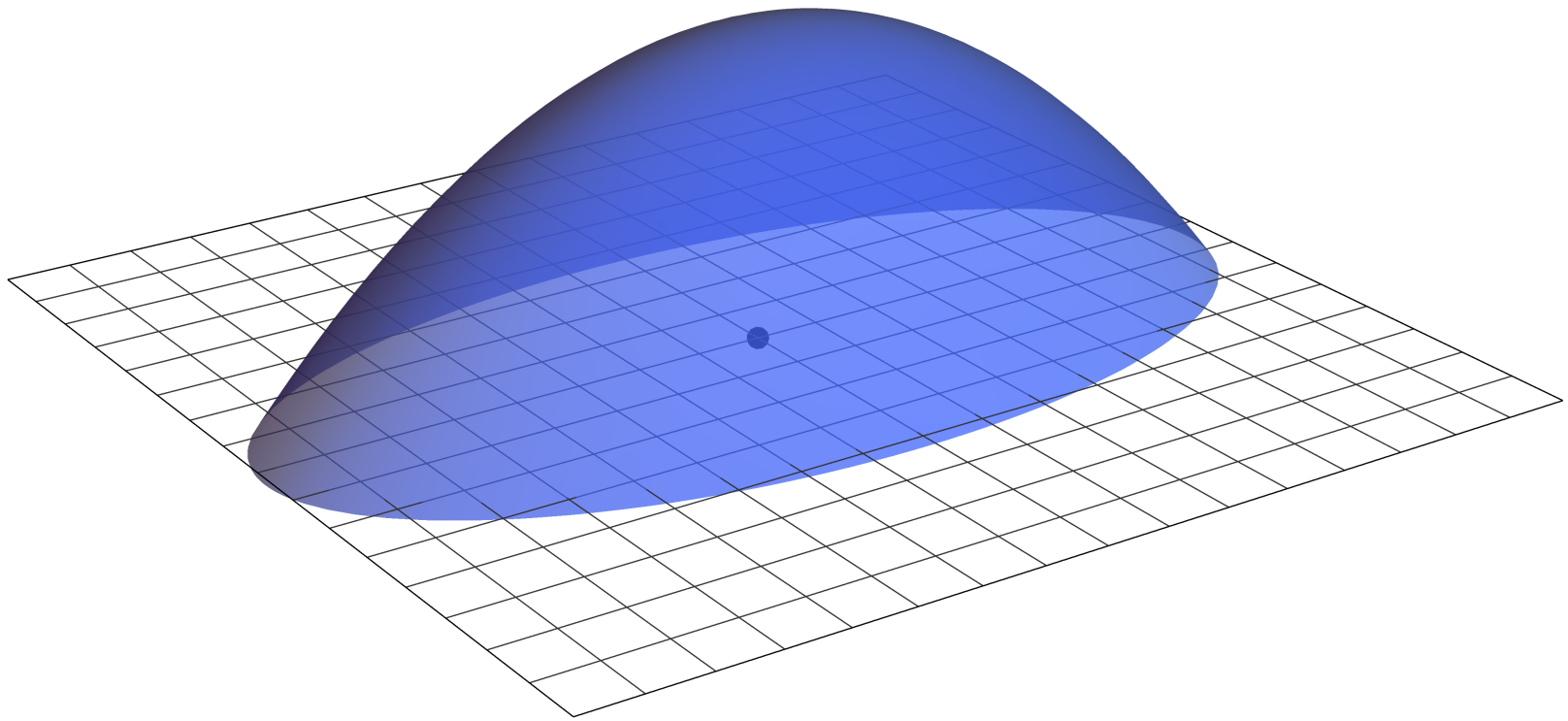
Problem Prompt

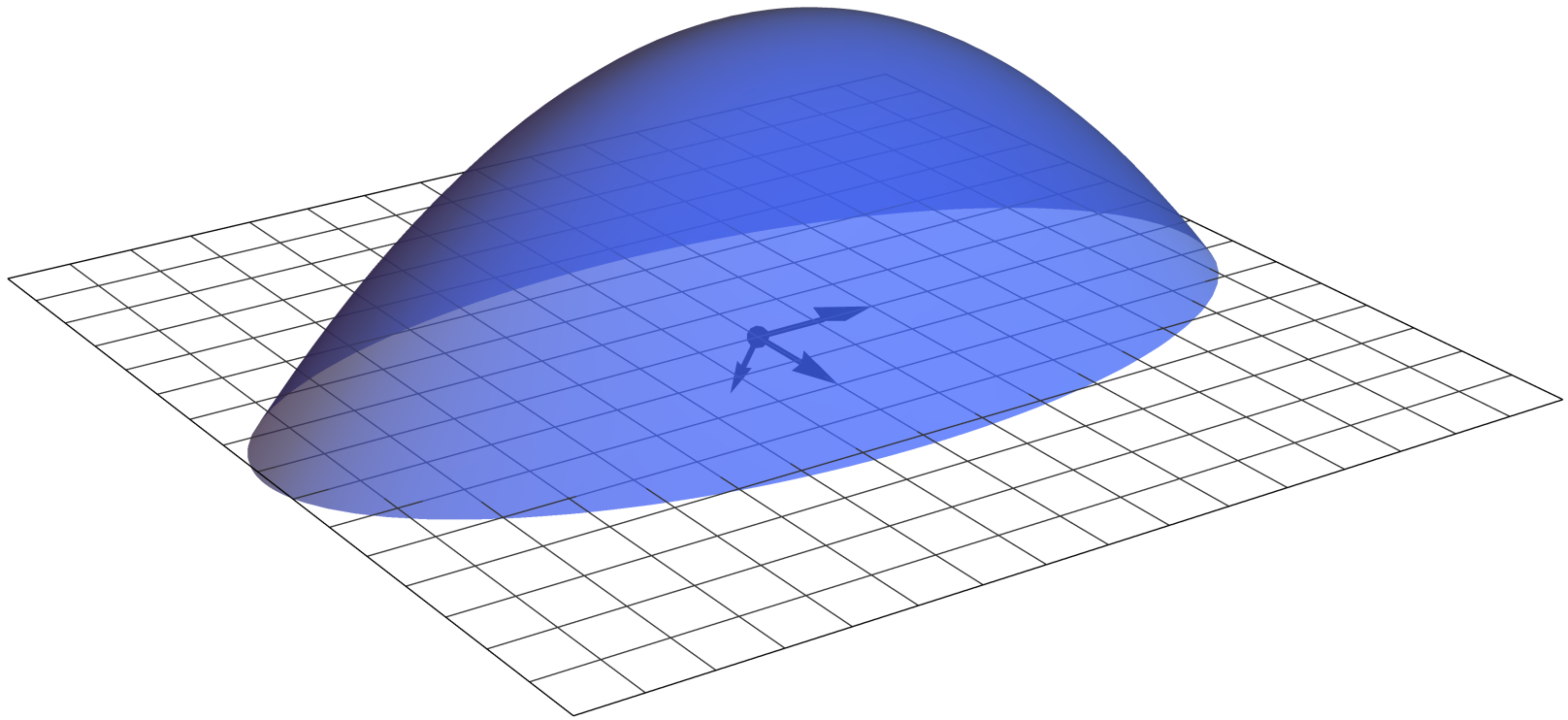
Do problem 4 on the worksheet.

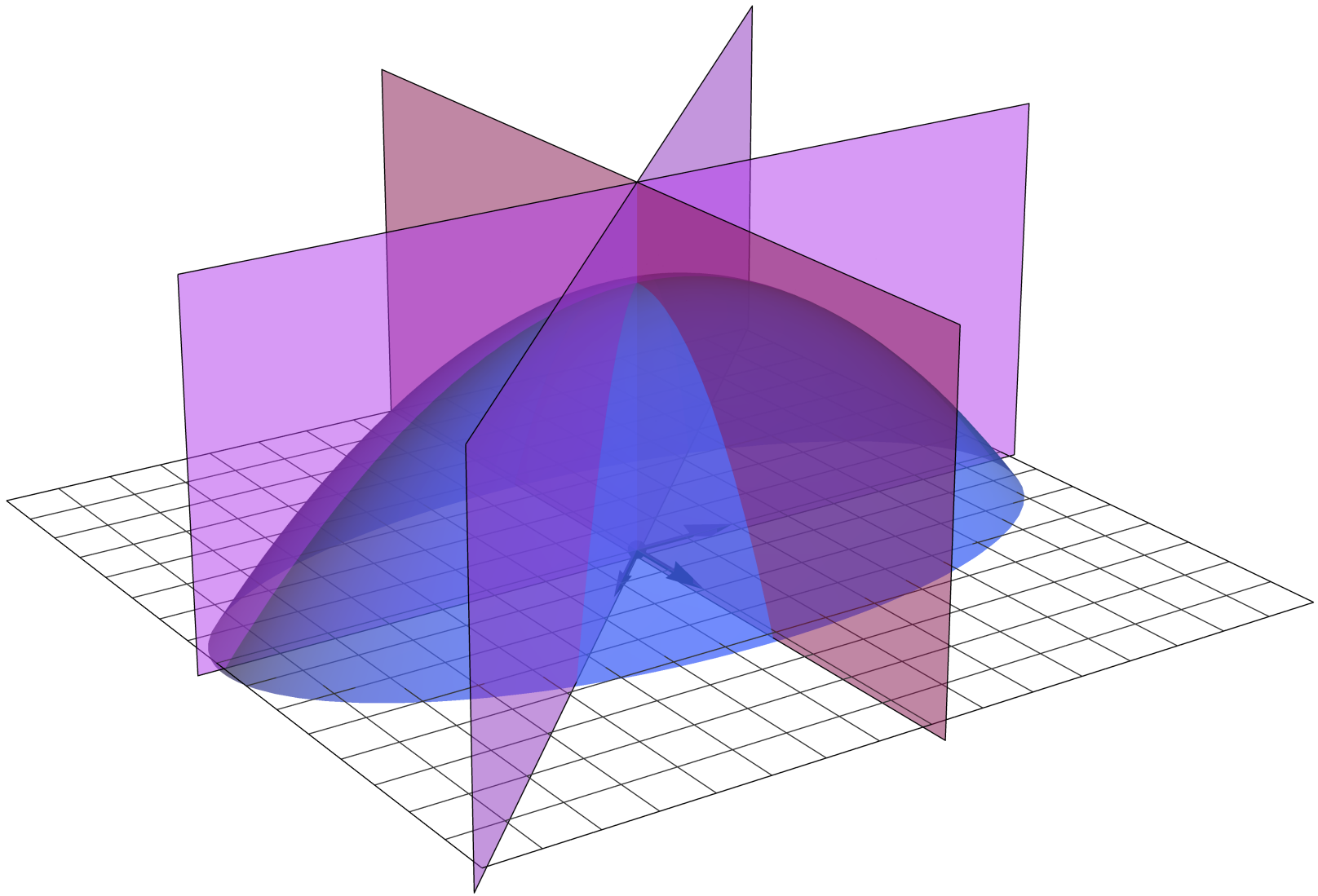

objective
10
8
6
4
0
2
4
6
8
10
12
14
gradient steps

11.2. Differential geometry


convex ⇒ minimizer
8
6
4
2
0
−2
−1
0
1
2
$J'(0) = 0,\ J''(0) > 0$
concave ⇒ maximizer
−2
−1
0
1
2
$J'(0) = 0,\ J''(0) < 0$







Definition 11.2

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2, $\boldsymbol{\theta} \in \mathbb{R}^n$ a point, and $\mathbf{v} \in \mathbb{R}^n$ a vector. We define the *directional first derivative of J at $\boldsymbol{\theta}$ in the direction $\mathbf{v}$* to be

$$J'_{\mathbf{v}}(\boldsymbol{\theta}) \overset{\text{def}}{=} \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} J(t\mathbf{v} + \boldsymbol{\theta}),$$

while we define the *directional second derivative* to be

$$J''_{\mathbf{v}}(\boldsymbol{\theta}) \overset{\text{def}}{=} \frac{\mathrm{d}^2}{\mathrm{d}t^2}\bigg|_{t=0} J(t\mathbf{v} + \boldsymbol{\theta}).$$

In this context, the vector $\mathbf{v}$ is called the *directional vector*.

Problem Prompt

Do problem 5 on the worksheet.

Definition 11.3

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2 and $\boldsymbol{\theta} \in \mathbb{R}^n$ a point. We define the *gradient vector* to be

$$\nabla J(\boldsymbol{\theta}) \overset{\text{def}}{=} \left[\frac{\partial J}{\partial \theta_i}(\boldsymbol{\theta})\right] = \begin{bmatrix} \frac{\partial J}{\partial \theta_1}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial J}{\partial \theta_n}(\boldsymbol{\theta}) \end{bmatrix} \in \mathbb{R}^n,$$

while we define the the *Hessian matrix* to be

$$\nabla^2 J(\boldsymbol{\theta}) \overset{\text{def}}{=} \left[\frac{\partial^2 J}{\partial \theta_i \partial \theta_j}(\boldsymbol{\theta})\right] = \begin{bmatrix} \frac{\partial^2 J}{\partial \theta_1^2}(\boldsymbol{\theta}) & \cdots & \frac{\partial^2 J}{\partial \theta_1 \partial \theta_n}(\boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 J}{\partial \theta_n \partial \theta_1}(\boldsymbol{\theta}) & \cdots & \frac{\partial^2 J}{\partial \theta_n^2}(\boldsymbol{\theta}) \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

Theorem 11.1 (Slopes, curvatures, and partial derivatives)

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2, $\boldsymbol{\theta} \in \mathbb{R}^n$ a point, and $\mathbf{v} \in \mathbb{R}^n$ a directional vector.

1. We have

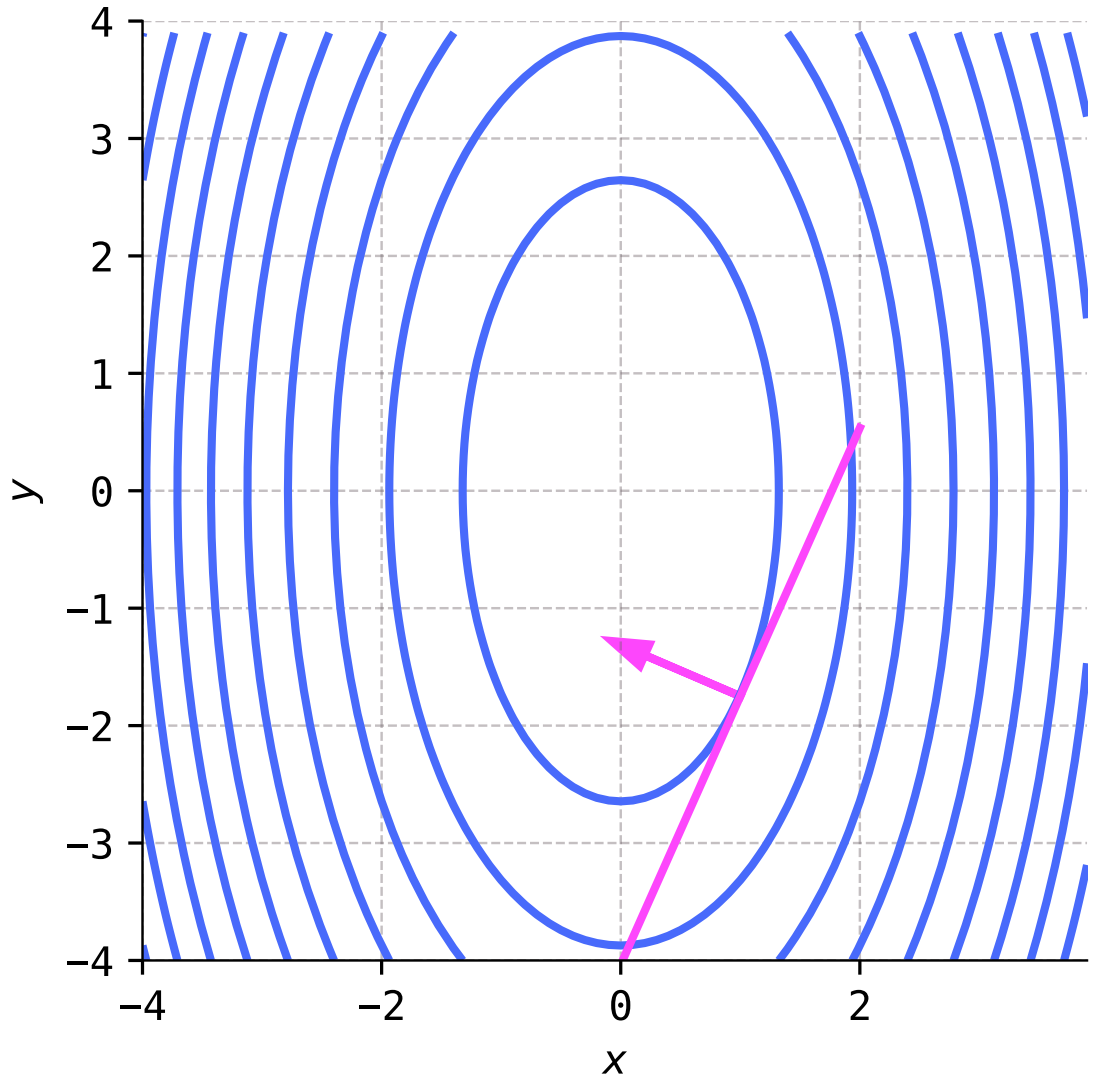
$$J'_{\mathbf{v}}(\boldsymbol{\theta}) = \mathbf{v}^\intercal \nabla J(\boldsymbol{\theta}).$$

2. We have

$$J''_{\mathbf{v}}(\boldsymbol{\theta}) = \mathbf{v}^\intercal \nabla^2 J(\boldsymbol{\theta}) \mathbf{v}.$$

Problem Prompt

Do problem 6 on the worksheet.


4
3
2
1
0
−1
−2
−3
−4
y
−4
−2
0
2
x

Theorem 11.2 (Properties of gradient vectors)

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2, $\boldsymbol{\theta} \in \mathbb{R}^n$ a point, and suppose the gradient vector $\nabla J(\boldsymbol{\theta})$ is nonzero.

1. The gradient vector $\nabla J(\boldsymbol{\theta})$ points in the direction of maximum rate of change.
2. The negative gradient vector $-\nabla J(\boldsymbol{\theta})$ points in the direction of minimum rate of change.
3. The gradient vector $\nabla J(\boldsymbol{\theta})$ is orthogonal to level surfaces.

Problem Prompt

Do problem 7 on the worksheet.

Theorem 11.3 (Second Derivative Test)

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2 and $\boldsymbol{\theta}^\star \in \mathbb{R}^n$ a stationary point.

1. If the Hessian matrix $\nabla^2 J(\boldsymbol{\theta}^\star)$ is positive definite, then $\boldsymbol{\theta}^\star$ is a local minimizer.
2. If the Hessian matrix $\nabla^2 J(\boldsymbol{\theta}^\star)$ is negative definite, then $\boldsymbol{\theta}^\star$ is a local maximizer.

Problem Prompt

Do problem 8 on the worksheet.

Theorem 11.4 (Eigenvalues, eigenvectors, and local curvature)

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2 and $\boldsymbol{\theta} \in \mathbb{R}^n$ a point with positive definite Hessian matrix $\nabla^2 J(\boldsymbol{\theta})$. Suppose we linearly order the eigenvalues of the Hessian matrix as

$$0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n. \tag{11.8}$$

Then:

1. The directional curvature $J''_{\mathbf{v}}(\boldsymbol{\theta})$ is maximized exactly when $\mathbf{v}$ lies in the eigenspace of λ_n, in which case $J''_{\mathbf{v}}(\boldsymbol{\theta}) = \lambda_n$.
2. The directional curvature $J''_{\mathbf{v}}(\boldsymbol{\theta})$ is minimized exactly when $\mathbf{v}$ lies in the eigenspace of λ_1, in which case $J''_{\mathbf{v}}(\boldsymbol{\theta}) = \lambda_1$.

Problem Prompt

Do problem 9 on the worksheet.

Definition 11.4

Let $\mathbf{A}$ be an $n \times n$ square matrix.

1. The *spectrum* of $\mathbf{A}$, denoted $\sigma(\mathbf{A})$, is the set of eigenvalues of $\mathbf{A}$.
2. The *spectral radius* of $\mathbf{A}$, denoted $\rho(\mathbf{A})$, is given by

$$\rho(\mathbf{A}) \overset{\text{def}}{=} \max_{\lambda \in \sigma(\mathbf{A})} |\lambda|.$$

3. If $\mathbf{A}$ is positive definite, the *condition number* of $\mathbf{A}$, denoted $\kappa(\mathbf{A})$, is the ratio

$$\kappa(\mathbf{A}) \overset{\text{def}}{=} \frac{\lambda_{\max}}{\lambda_{\min}}$$

of the largest eigenvalue of $\mathbf{A}$ to the smallest.

Problem Prompt

Do problem 10 on the worksheet.

11.3. Gradient descent in multiple variables

Algorithm 11.3 (Multi-variable gradient descent with learning rate decay)

Input: A differentiable function $J : \mathbb{R}^n \to \mathbb{R}$, an initial guess $\boldsymbol{\theta}_0 \in \mathbb{R}^n$ for a local minimizer $\boldsymbol{\theta}^\star$, a learning rate $\alpha > 0$, a decay rate $\beta \in [0, 1)$, and the number N of gradient steps.

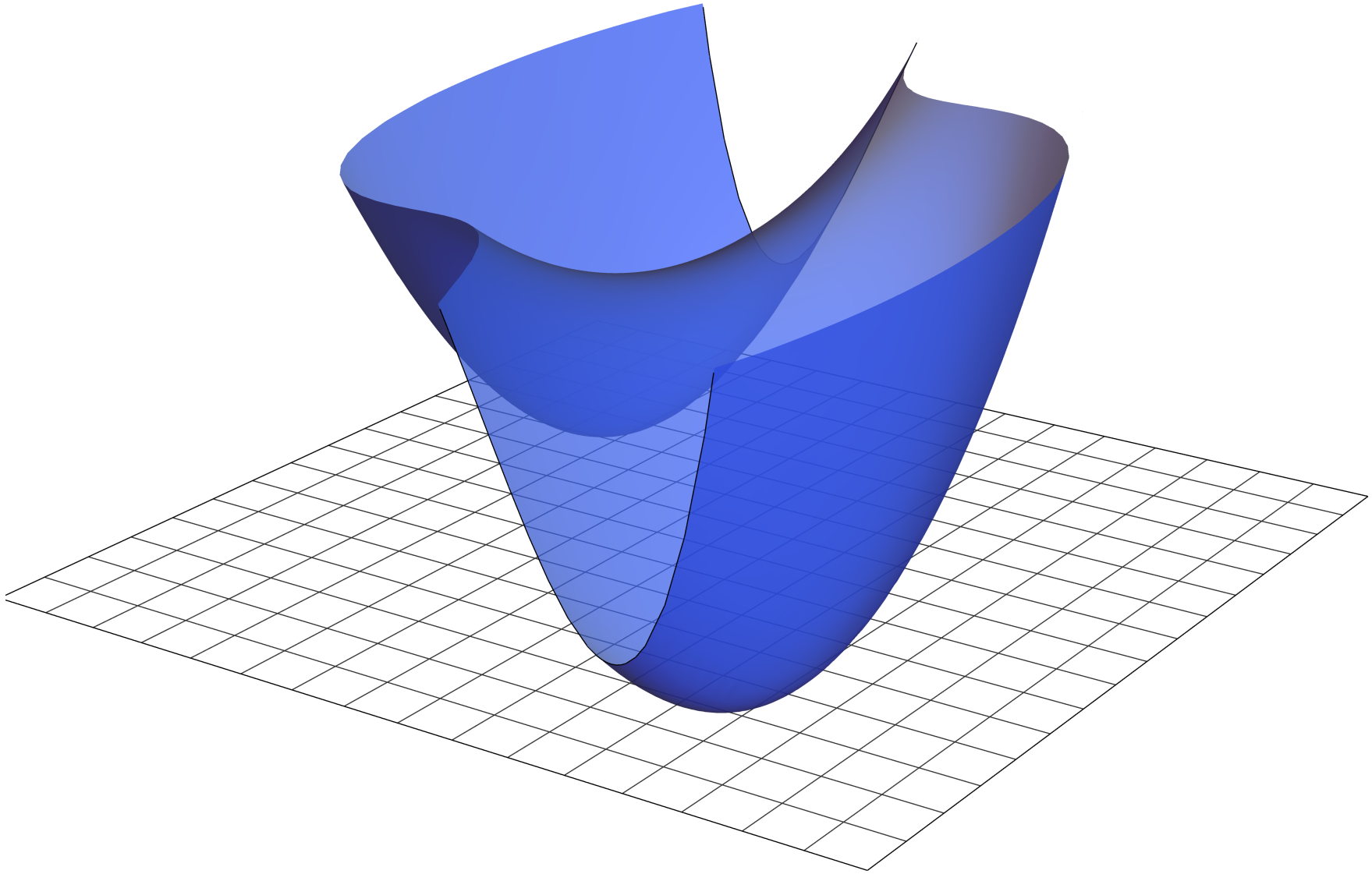
Output: An approximation to a local minimizer $\boldsymbol{\theta}^\star$.

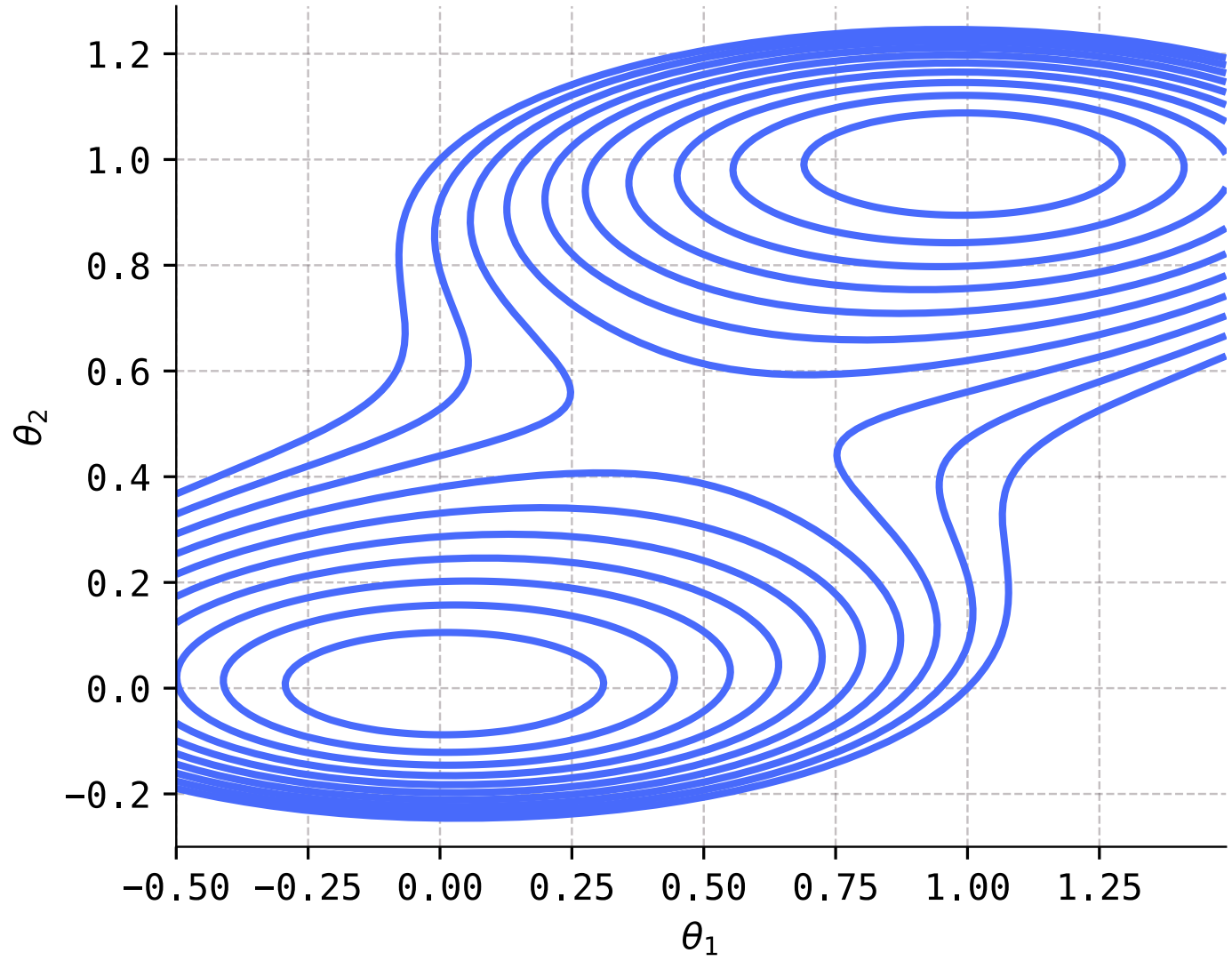
$\boldsymbol{\theta} := \boldsymbol{\theta}_0$

For t from 0 to $N - 1$, do:

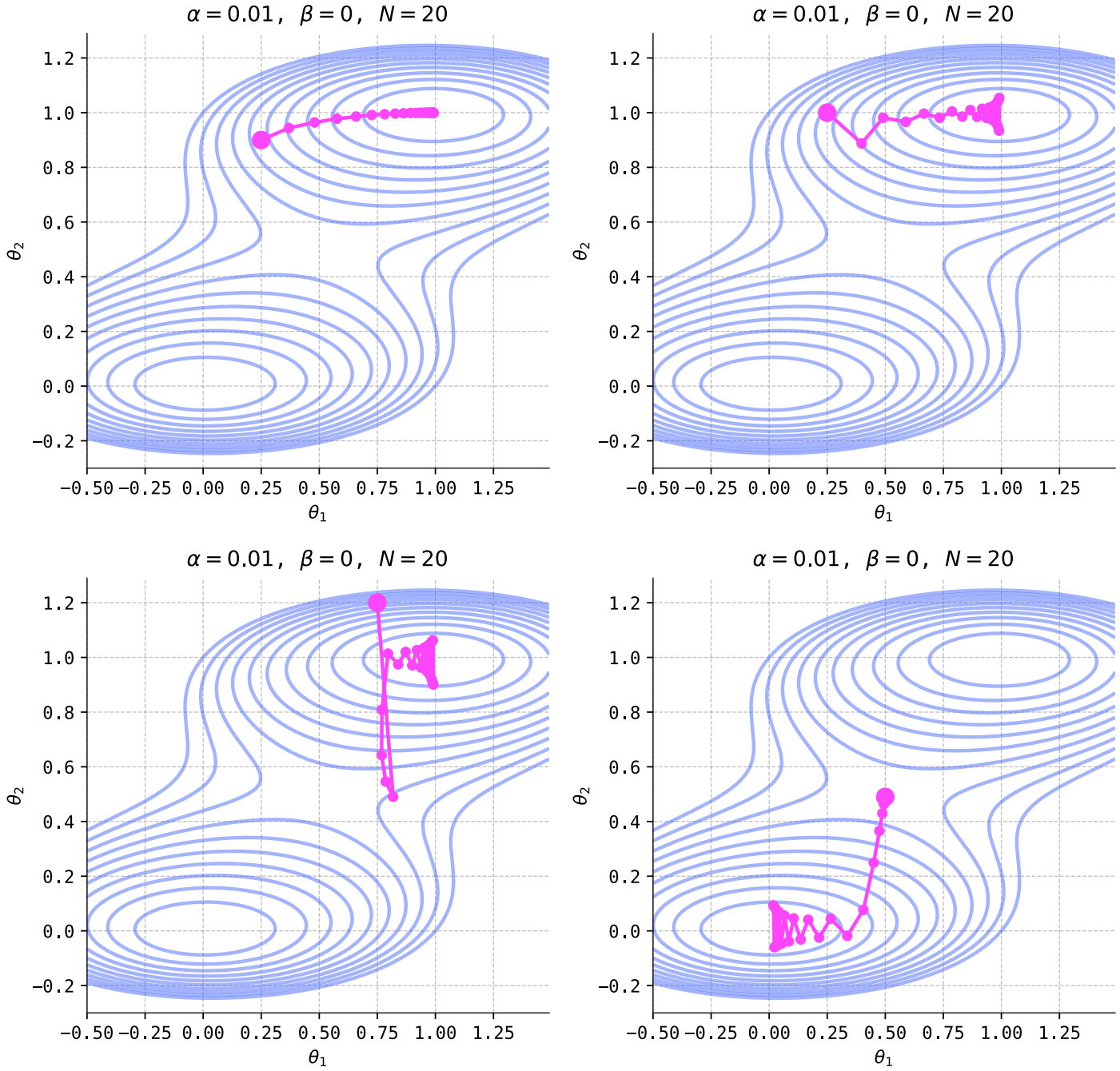
$\quad \boldsymbol{\theta} := \boldsymbol{\theta} - \alpha(1 - \beta)^{t+1} \nabla J(\boldsymbol{\theta})$

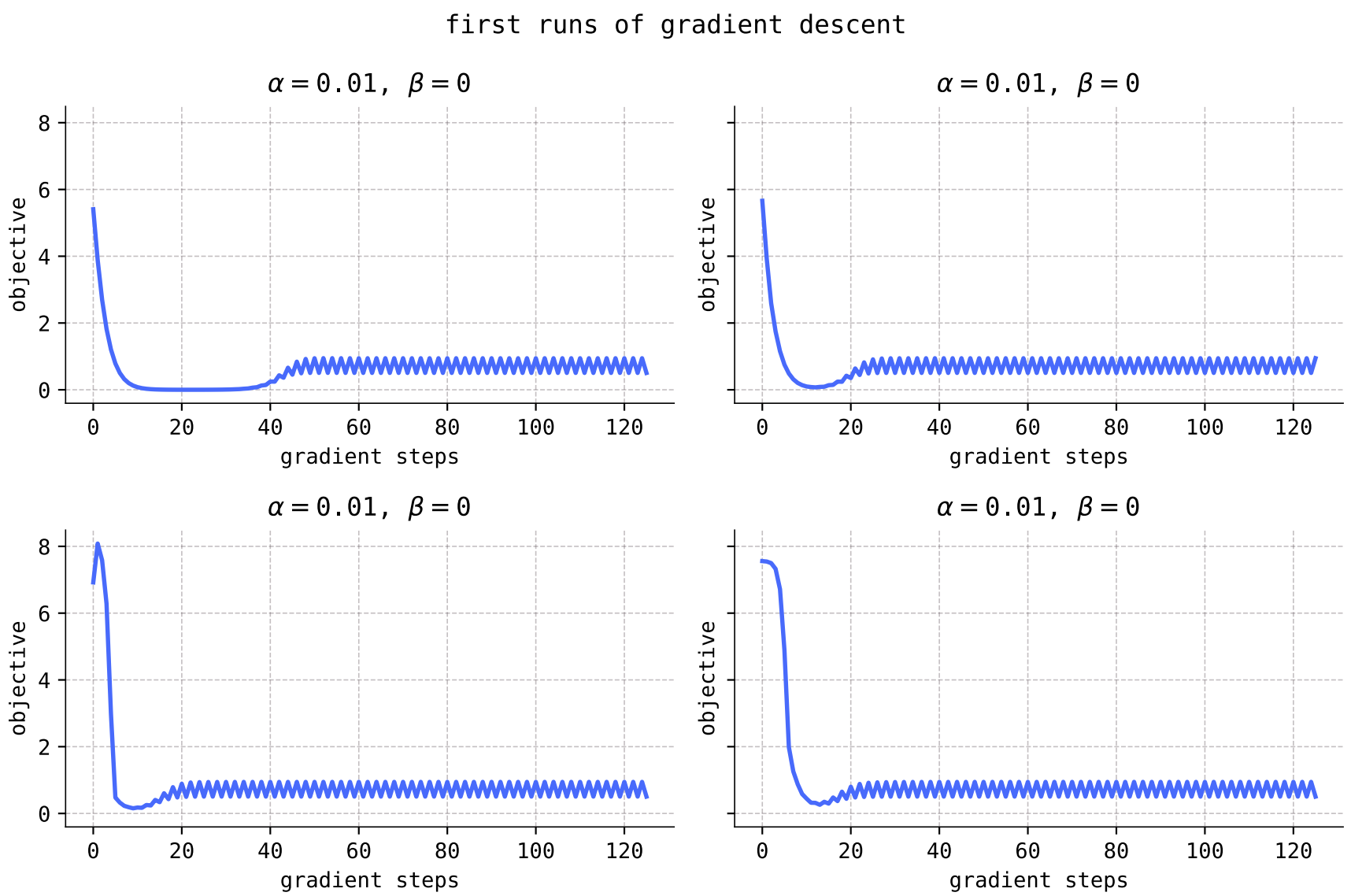
Return $\boldsymbol{\theta}$.

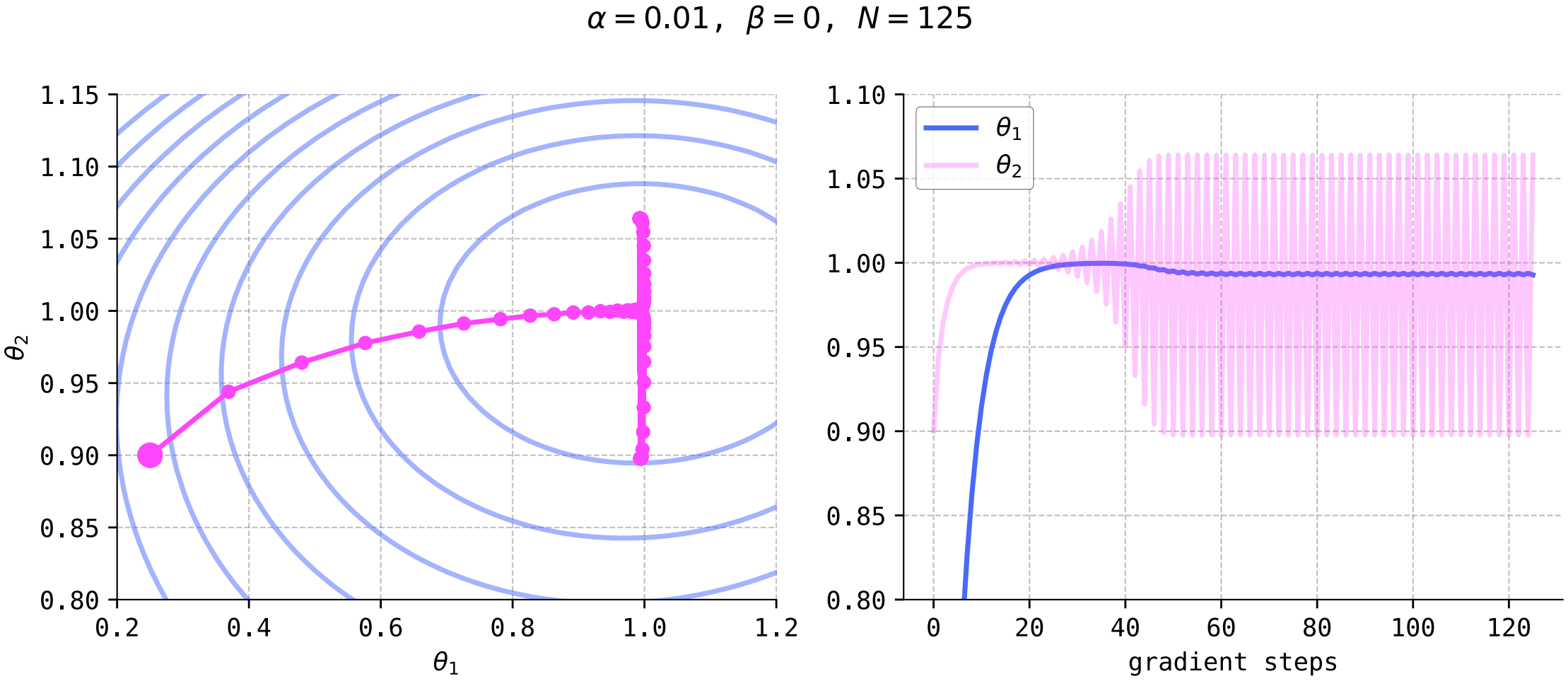



1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
θ_2
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25
θ_1

first runs of gradient descent




first runs of gradient descent
$\alpha = 0.01,\ \beta = 0$
$\alpha = 0.01,\ \beta = 0$
$\alpha = 0.01,\ \beta = 0$
$\alpha = 0.01,\ \beta = 0$
objective
gradient steps
0
2
4
6
8
0
20
40
60
80
100
120


$\alpha = 0.01,\ \beta = 0,\ N = 125$
1.15
1.10
1.05
1.00
0.95
0.90
0.85
0.80
θ_2
0.2
0.4
0.6
0.8
1.0
1.2
θ_1
1.10
1.05
1.00
0.95
0.90
0.85
0.80
θ_1
θ_2
0
20
40
60
80
100
120
gradient steps

Problem Prompt

Do problem 11 on the worksheet.

Theorem 11.5 (Curvature and gradient descent)

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a function of class C^2 and $\boldsymbol{\theta}^\star$ a local minimizer with positive definite Hessian matrix $\mathbf{H} = \nabla^2 J(\boldsymbol{\theta}^\star)$ with minimum and maximum eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$. For initial guesses $\boldsymbol{\theta}_0$ sufficiently near $\boldsymbol{\theta}^\star$ to allow a degree-2 Taylor polynomial approximation, the gradient descent algorithm with learning rate

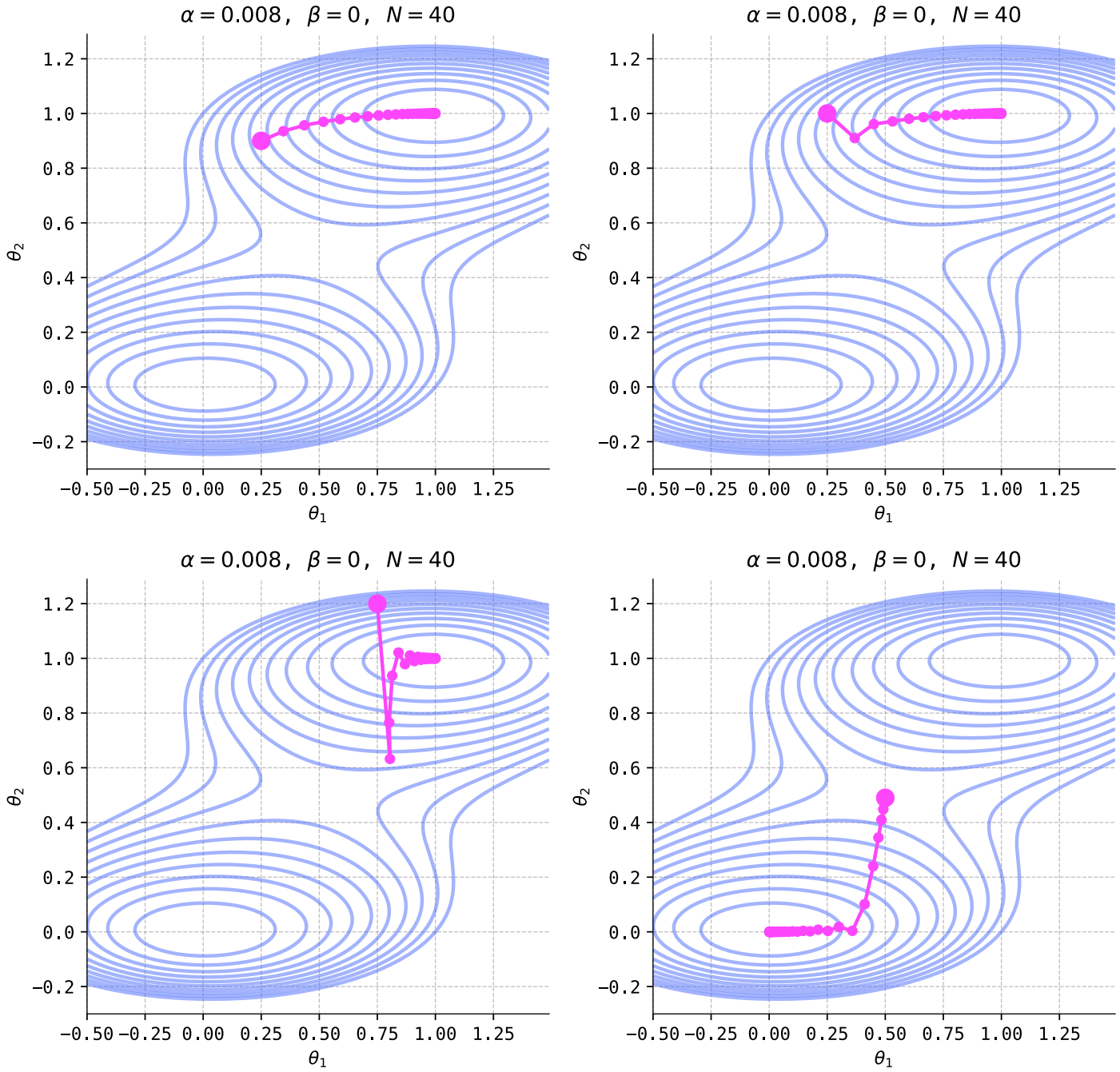
$$\alpha = \frac{2}{\lambda_{\min} + \lambda_{\max}} \quad \text{and} \quad \beta = 0$$

converges to $\boldsymbol{\theta}^\star$ exponentially fast, with

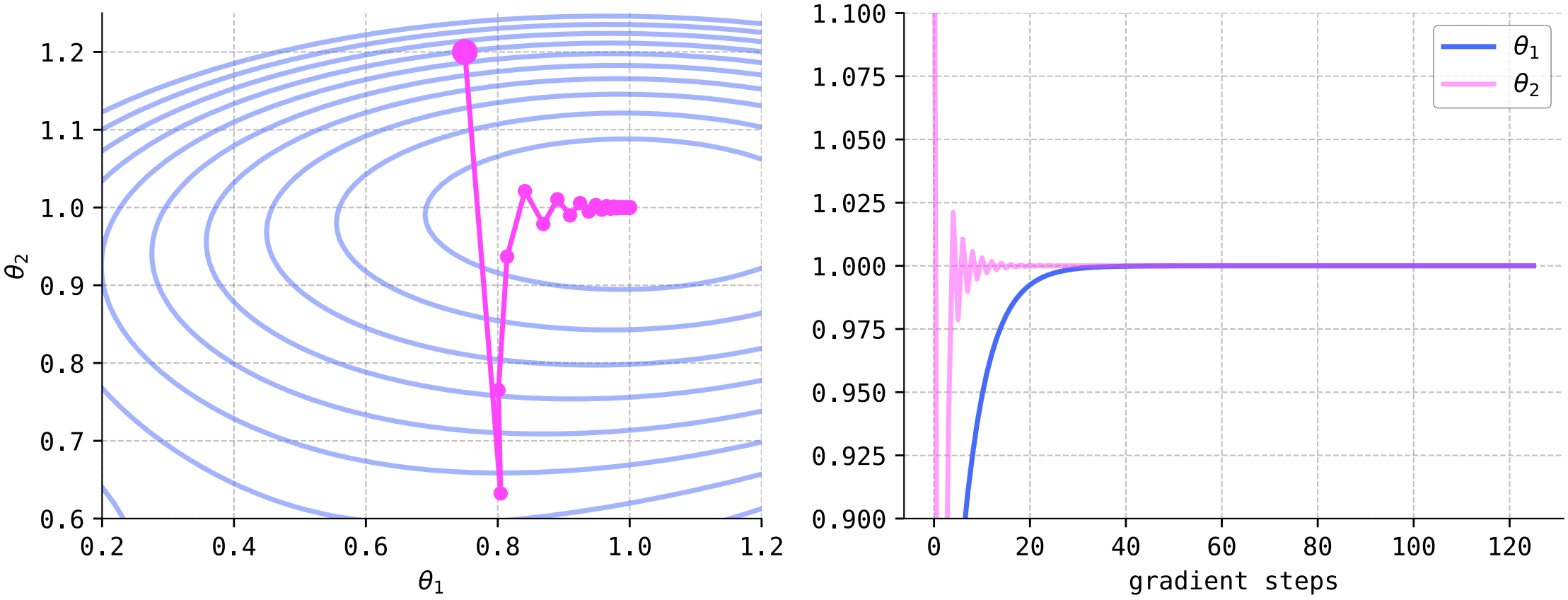
$$|\boldsymbol{\theta}_t - \boldsymbol{\theta}^\star| \leq \left(1 - \frac{2}{1 + \kappa(\mathbf{H})}\right)^t |\boldsymbol{\theta}_0 - \boldsymbol{\theta}^\star|$$

for each $t \geq 1$. Here, $\kappa(\mathbf{H})$ is the condition number of $\mathbf{H}$.

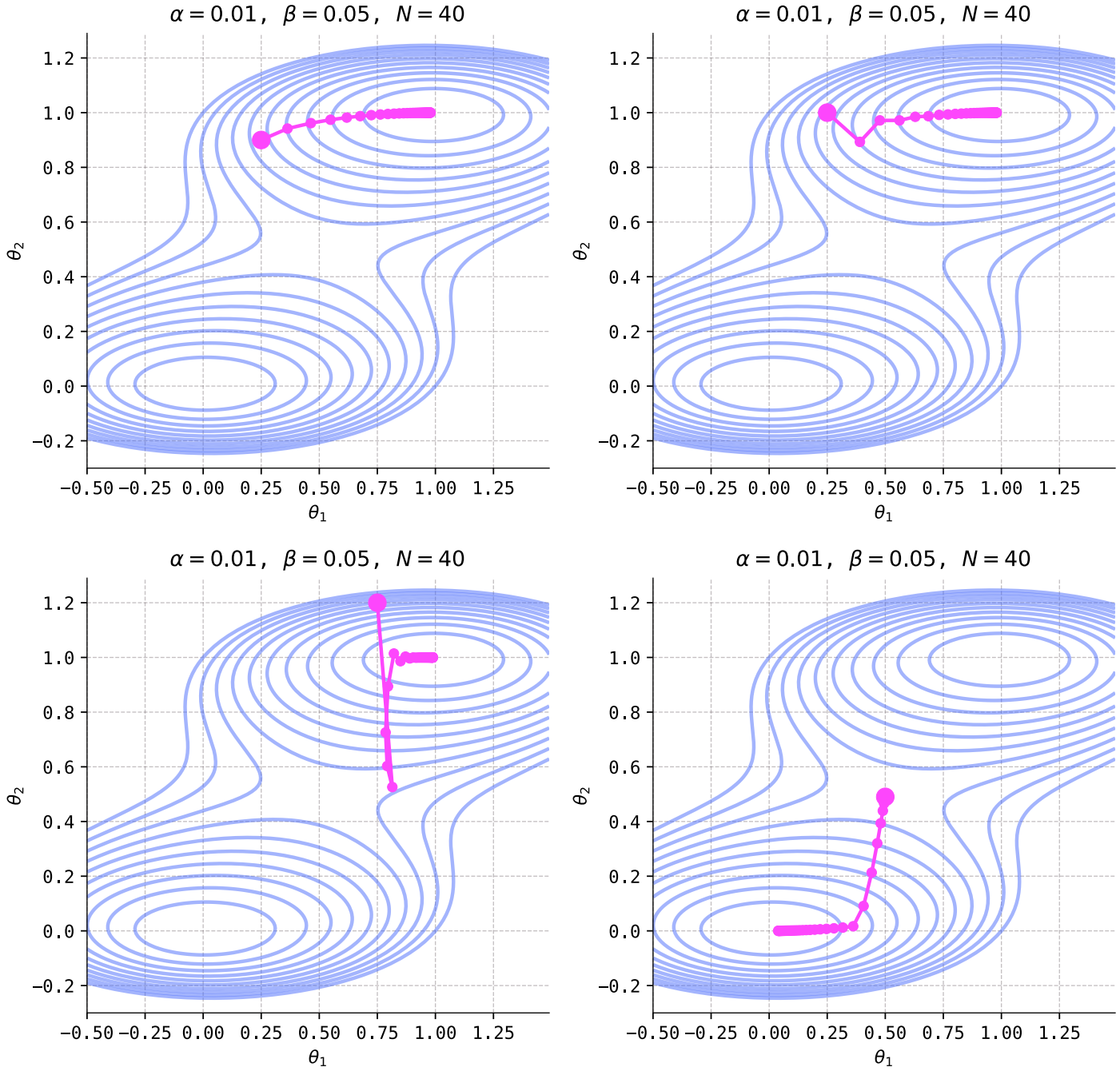
second runs of gradient descent with smaller learning rates

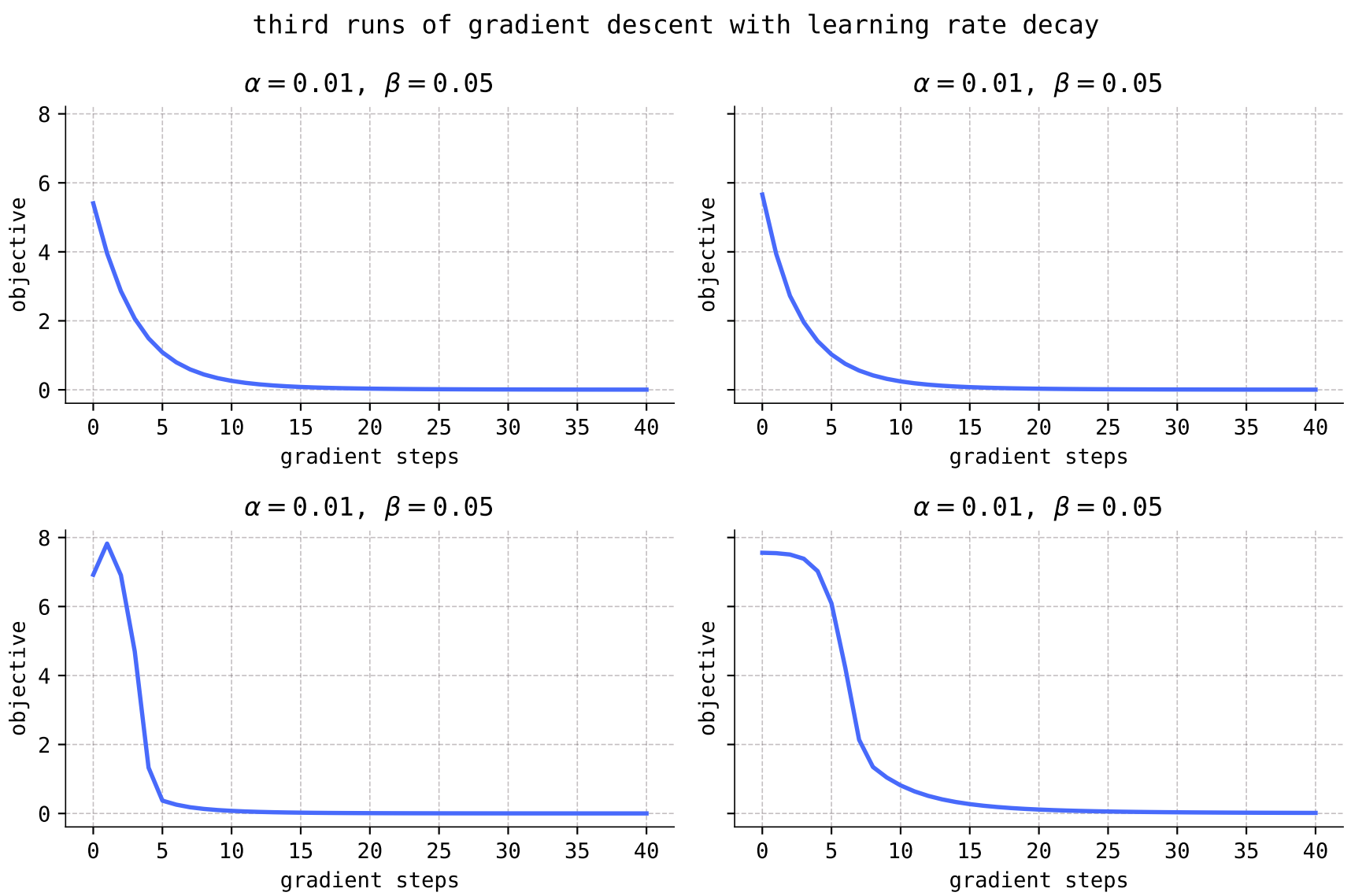


$\alpha = 0.008,\ \beta = 0,\ N = 125$



third runs of gradient descent with learning rate decay

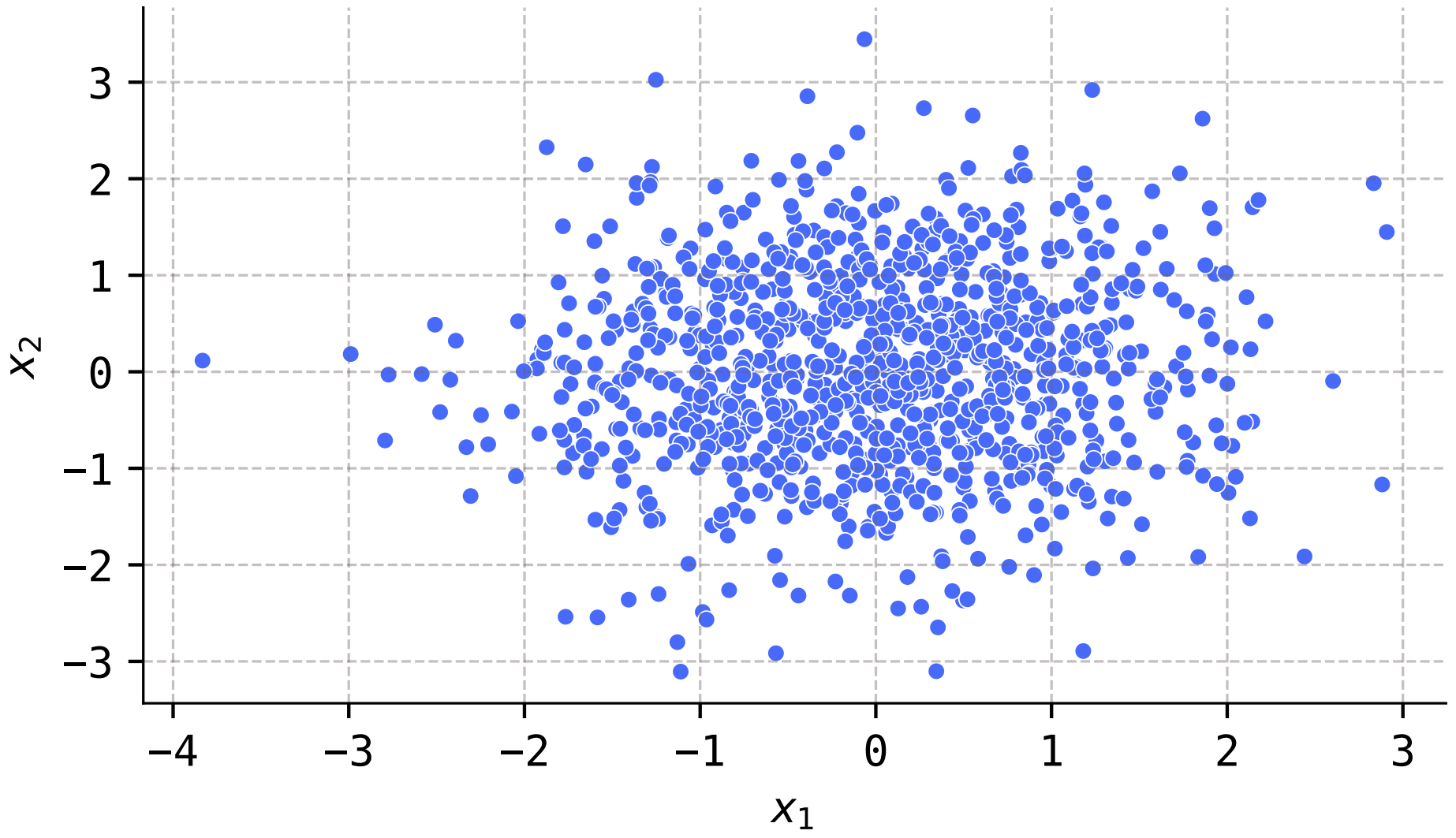



third runs of gradient descent with learning rate decay
$\alpha = 0.01,\ \beta = 0.05$
$\alpha = 0.01,\ \beta = 0.05$
$\alpha = 0.01,\ \beta = 0.05$
$\alpha = 0.01,\ \beta = 0.05$
objective
gradient steps
0
2
4
6
8
0
5
10
15
20
25
30
35
40

11.4. Stochastic gradient descent

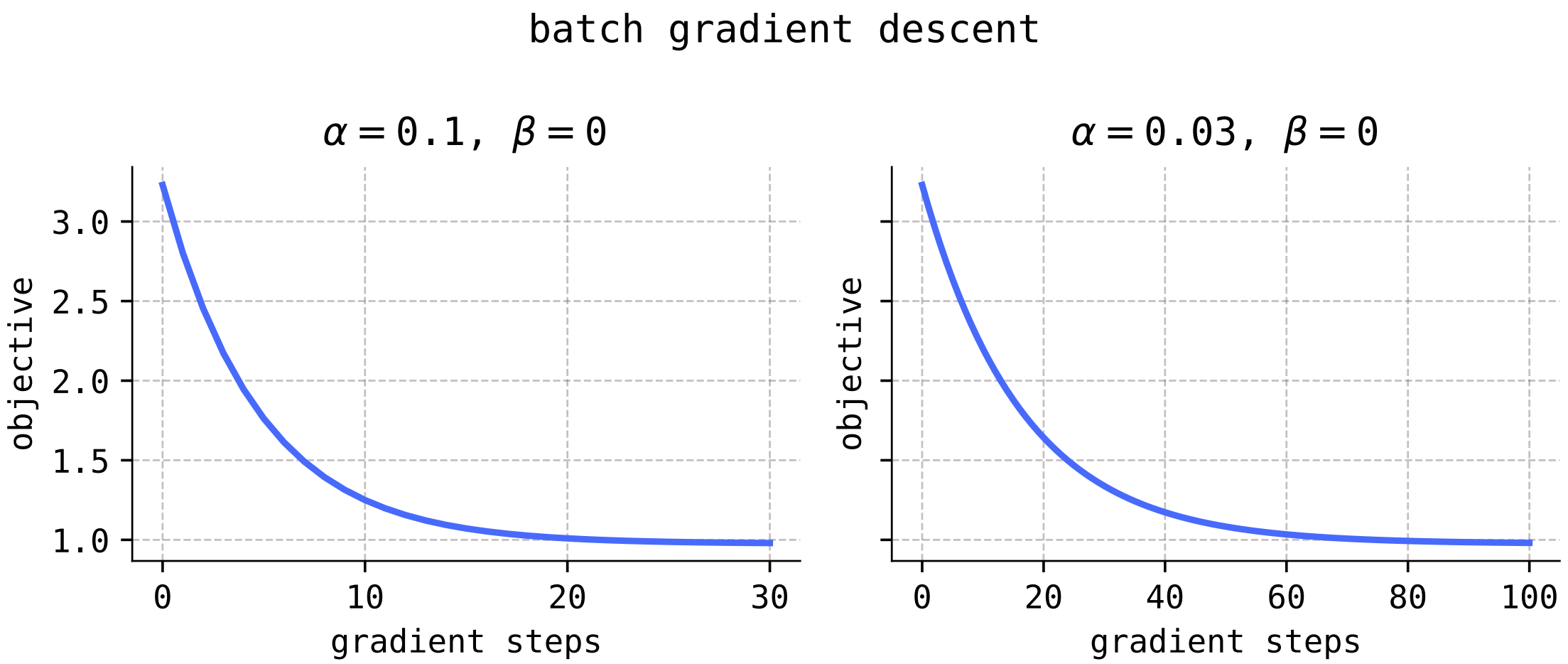
Definition 11.5

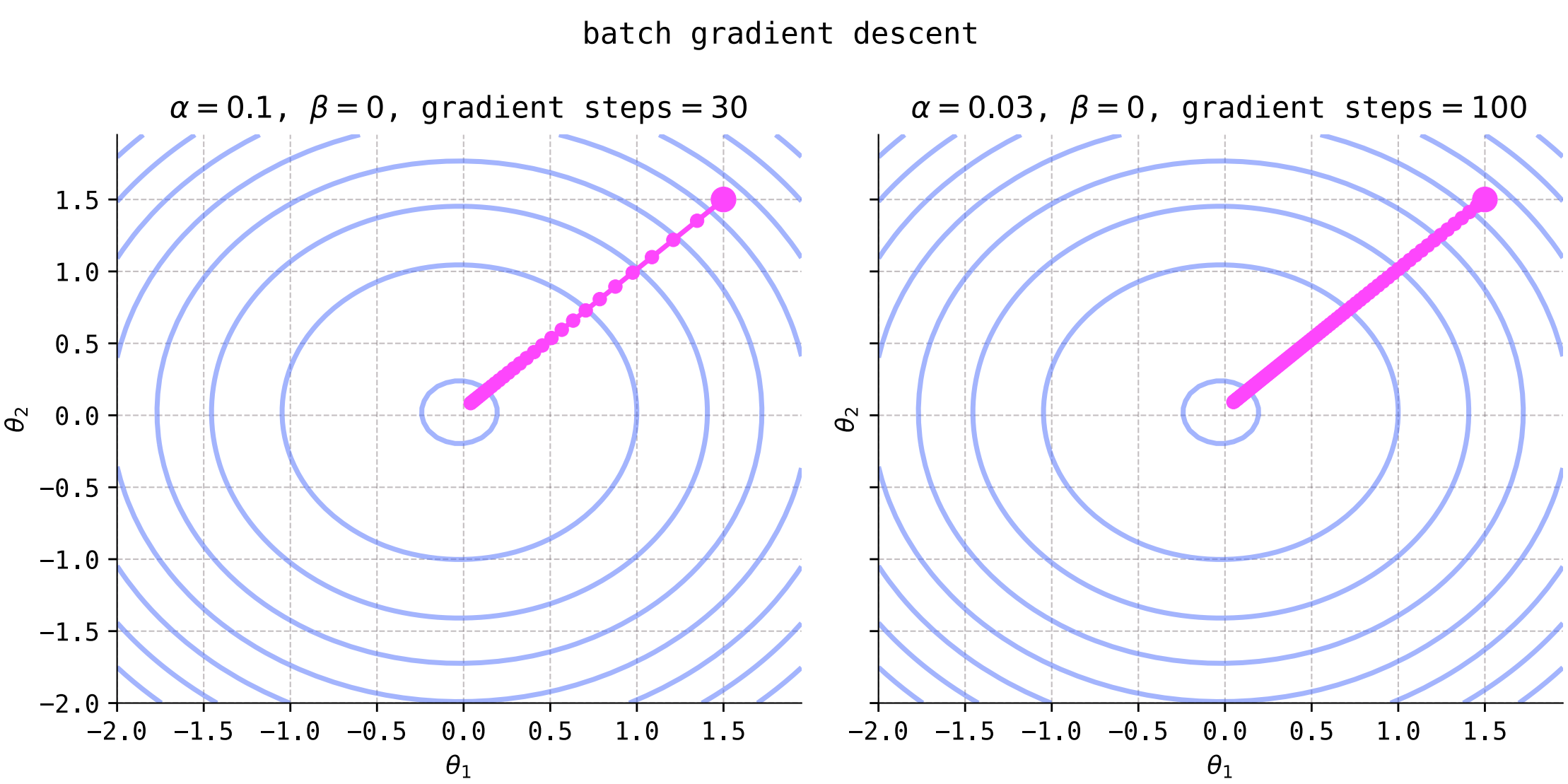
The *batch gradient descent algorithm* is the gradient descent algorithm applied to a stochastic objective function of the form (11.17).


3
2
1
0
−1
−2
−3
x_2
−4
−3
−2
−1
0
1
2
3
x_1

Problem Prompt

Do problem 12 on the worksheet.


batch gradient descent
$\alpha = 0.1,\ \beta = 0$
$\alpha = 0.03,\ \beta = 0$
objective
3.0
2.5
2.0
1.5
1.0
0
10
20
30
gradient steps
objective
0
20
40
60
80
100
gradient steps


batch gradient descent
$\alpha = 0.1,\ \beta = 0,$ gradient steps $= 30$
$\alpha = 0.03,\ \beta = 0,$ gradient steps $= 100$
θ_1
θ_2
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5

Algorithm 11.4 (Stochastic gradient descent with learning rate decay)

Input: A dataset $\mathbf{x}_1, \mathbf{x}_2 \ldots, \mathbf{x}_m \in \mathbb{R}^n$, a differentiable loss function $L : \mathbb{R}^{n+k} \to \mathbb{R}$, an initial guess $\boldsymbol{\theta}_0 \in \mathbb{R}^k$ for a minimizer $\boldsymbol{\theta}^\star$ of the stochastic objective function (11.18), a learning rate $\alpha > 0$, a decay rate $\beta \in [0, 1)$, and the number N of epochs.

Output: An approximation to a minimizer $\boldsymbol{\theta}^\star$.

$\boldsymbol{\theta} := \boldsymbol{\theta}_0$

$s := 0$

For t from 0 to $N - 1$, do:

Randomly partition the dataset into mini-batches $B_1, B_2, \ldots, B_p$

For each mini-batch B_j, do:

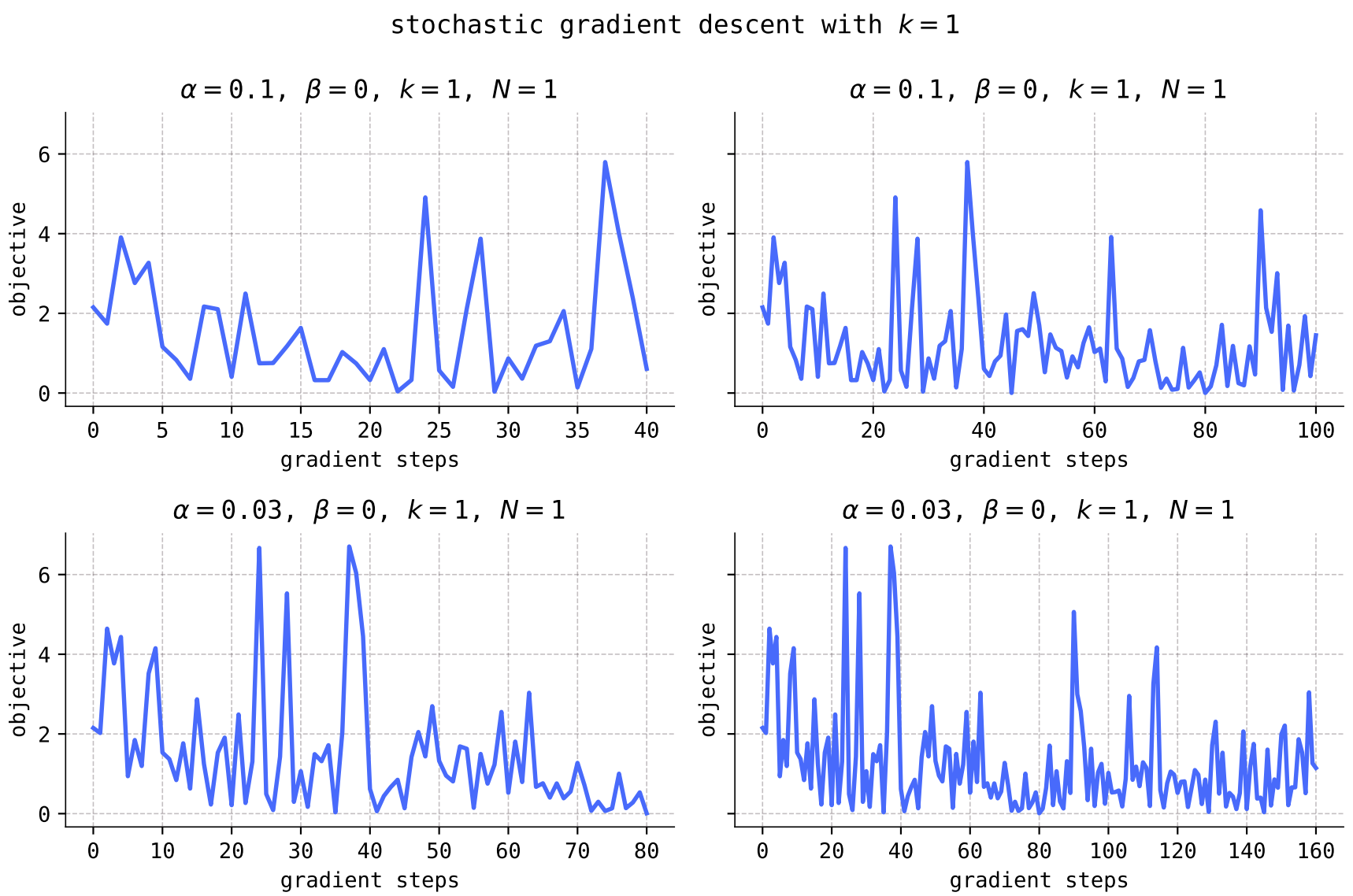
$$\boldsymbol{\theta} := \boldsymbol{\theta} - \alpha(1 - \beta)^{s+1} \frac{1}{|B_j|} \sum_{\mathbf{x} \in B_j} \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}, \mathbf{x})$$

$s := s + 1$

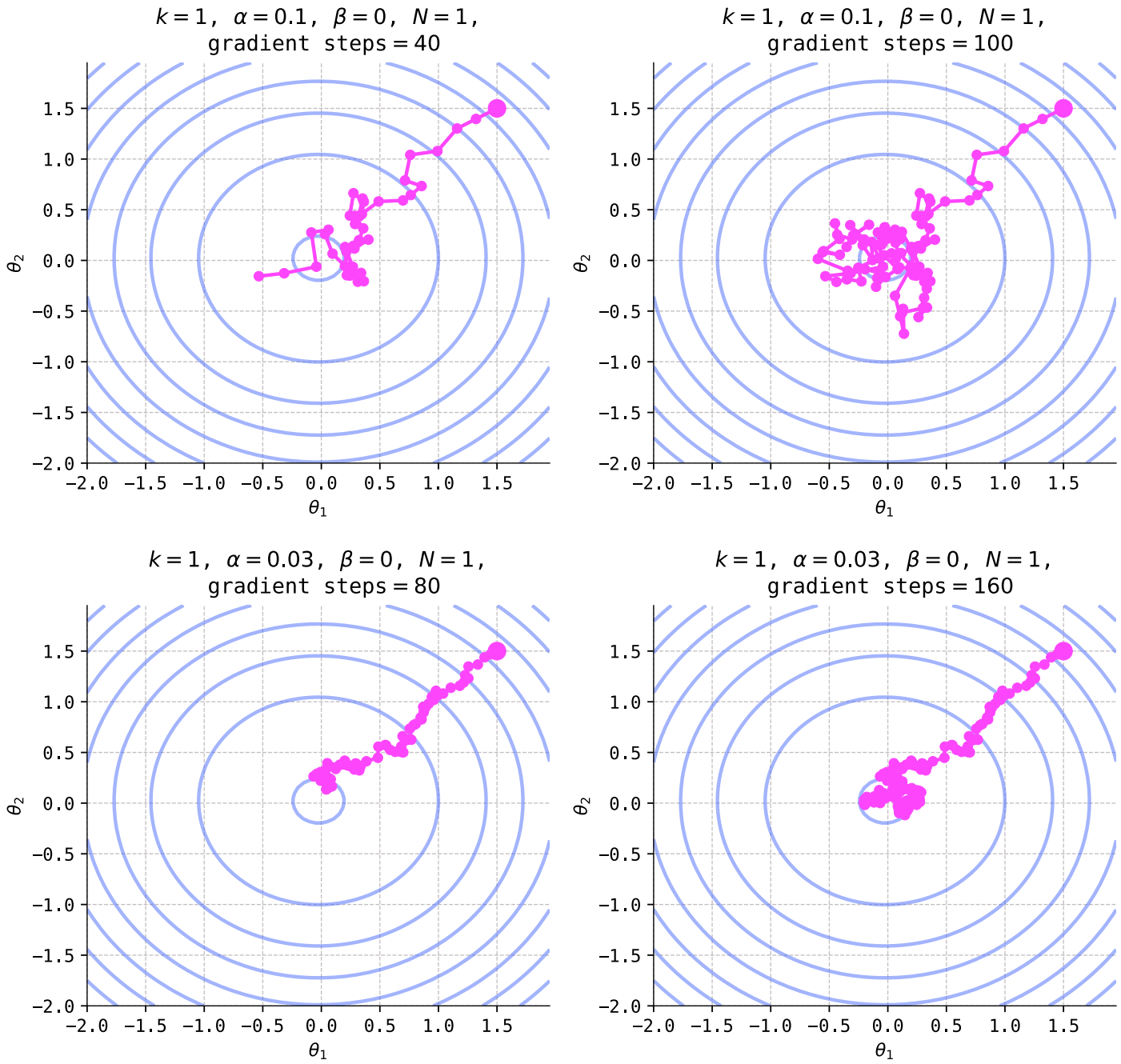
Return $\boldsymbol{\theta}$

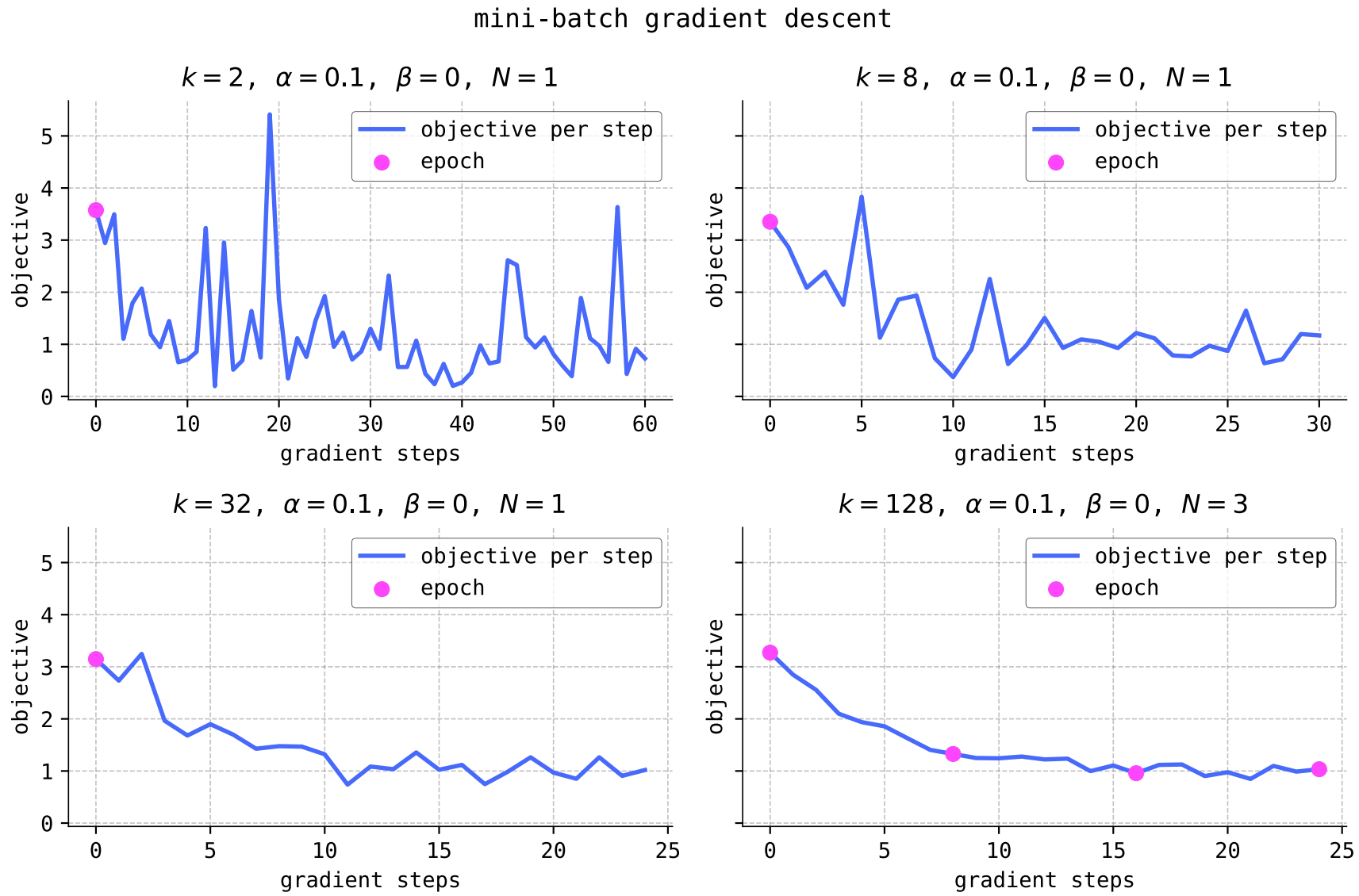
epoch number	output
0	$\boldsymbol{\theta}_0$
1	$\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_p$
2	$\boldsymbol{\theta}_{p+1}, \boldsymbol{\theta}_{p+2}, \ldots, \boldsymbol{\theta}_{2p}$
$\vdots$	$\vdots$
N	$\boldsymbol{\theta}_{(N-1)p+1}, \boldsymbol{\theta}_{(N-1)p+2}, \ldots, \boldsymbol{\theta}_{Np}$

epoch number	objectives
0	$J(\boldsymbol{\theta}_0)$
1	$J(\boldsymbol{\theta}_1), J(\boldsymbol{\theta}_2), \ldots, J(\boldsymbol{\theta}_p)$
2	$J(\boldsymbol{\theta}_{p+1}), J(\boldsymbol{\theta}_{p+2}), \ldots, J(\boldsymbol{\theta}_{2p})$
$\vdots$	$\vdots$
N	$J(\boldsymbol{\theta}_{(N-1)p+1}), J(\boldsymbol{\theta}_{(N-1)p+2}), \ldots, J(\boldsymbol{\theta}_{Np})$

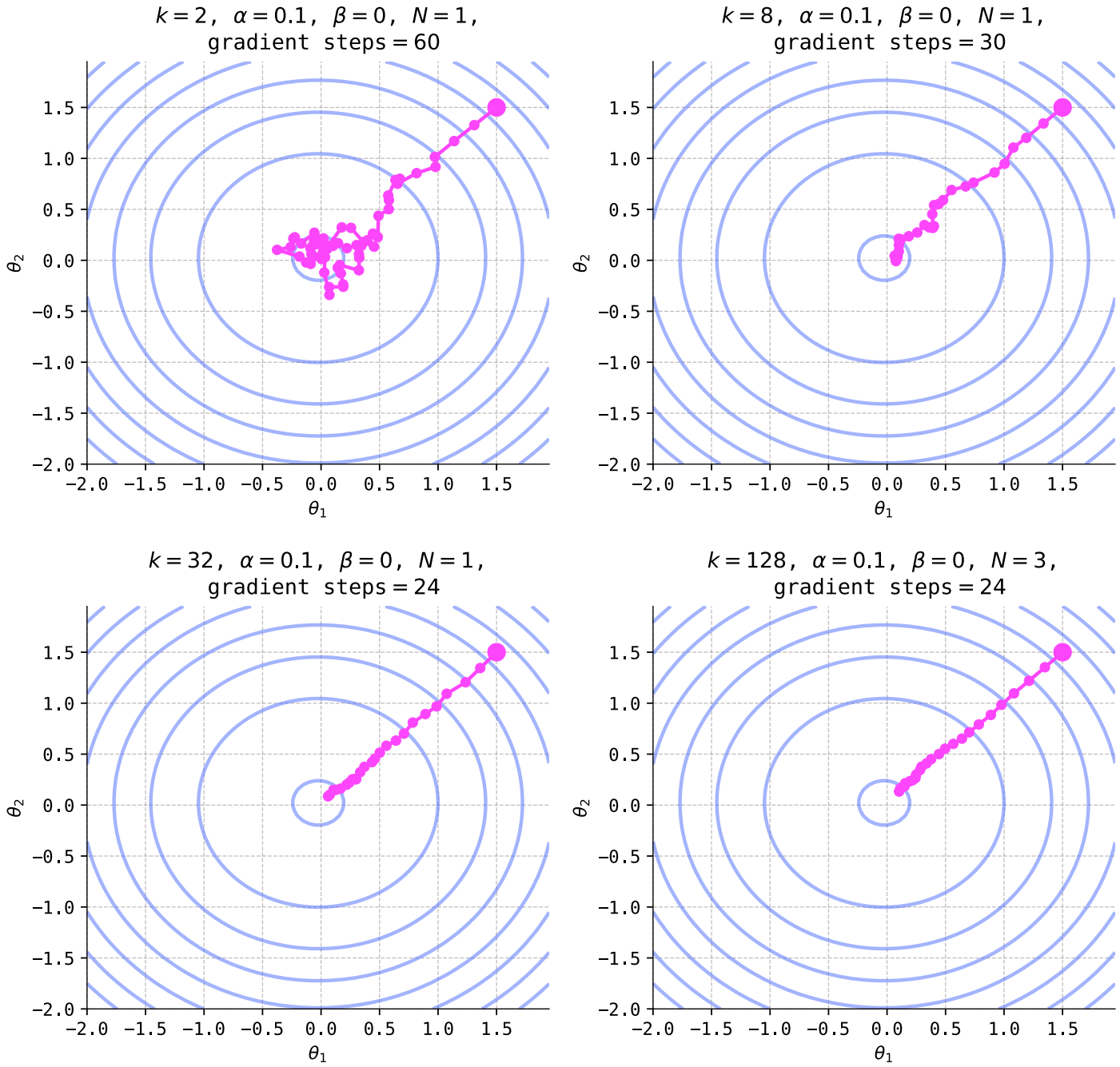

stochastic gradient descent with $k = 1$
$\alpha = 0.1,\ \beta = 0,\ k = 1,\ N = 1$
objective
gradient steps
$\alpha = 0.1,\ \beta = 0,\ k = 1,\ N = 1$
objective
gradient steps
$\alpha = 0.03,\ \beta = 0,\ k = 1,\ N = 1$
objective
gradient steps
$\alpha = 0.03,\ \beta = 0,\ k = 1,\ N = 1$
objective
gradient steps

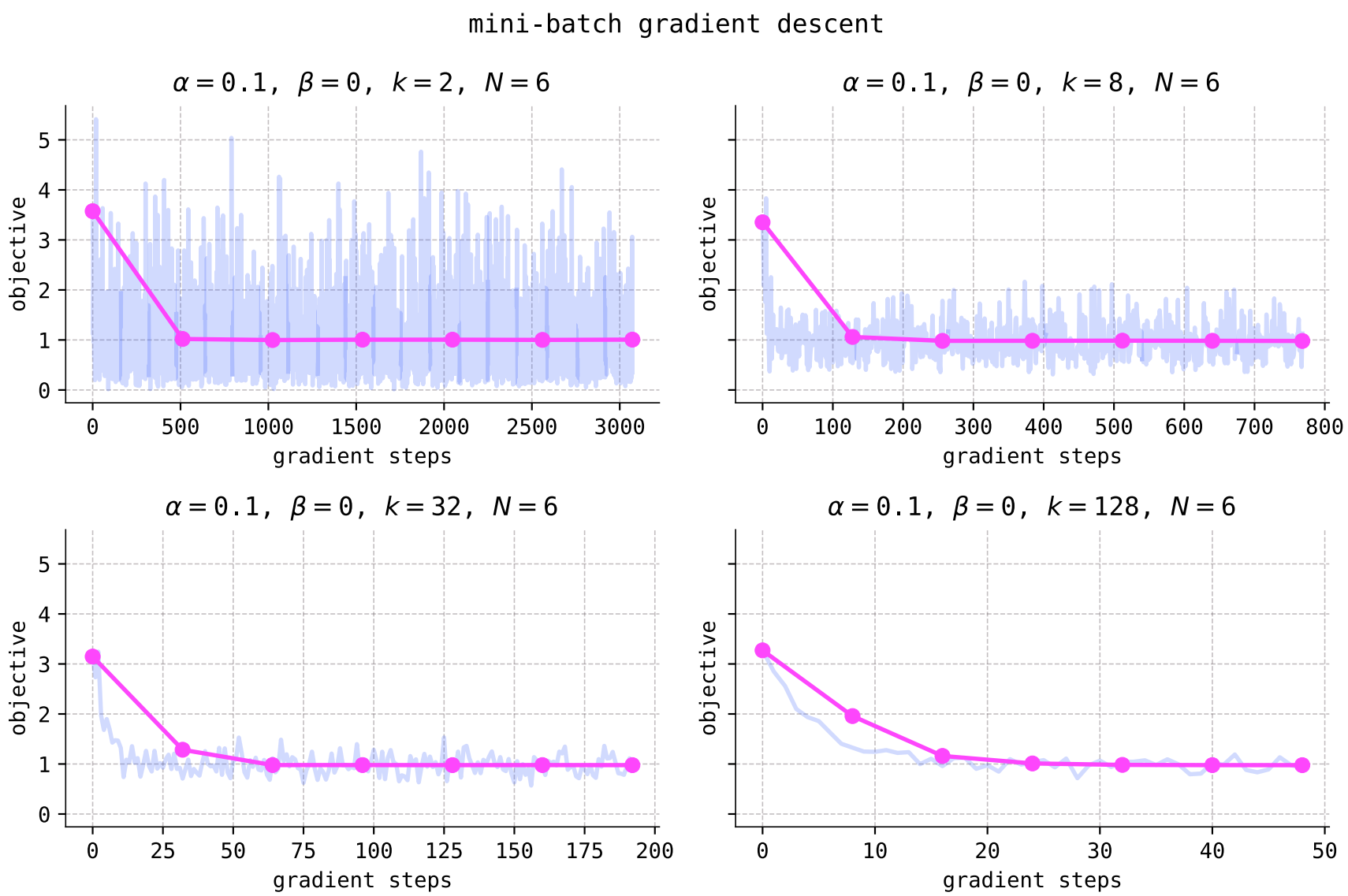
stochastic gradient descent with $k = 1$




mini-batch gradient descent
$k = 2,\ \alpha = 0.1,\ \beta = 0,\ N = 1$
$k = 8,\ \alpha = 0.1,\ \beta = 0,\ N = 1$
$k = 32,\ \alpha = 0.1,\ \beta = 0,\ N = 1$
$k = 128,\ \alpha = 0.1,\ \beta = 0,\ N = 3$
objective per step
epoch
objective
gradient steps

mini-batch gradient descent




mini-batch gradient descent
$\alpha = 0.1,\ \beta = 0,\ k = 2,\ N = 6$
$\alpha = 0.1,\ \beta = 0,\ k = 8,\ N = 6$
$\alpha = 0.1,\ \beta = 0,\ k = 32,\ N = 6$
$\alpha = 0.1,\ \beta = 0,\ k = 128,\ N = 6$
objective
gradient steps
0
1
2
3
4
5
0
500
1000
1500
2000
2500
3000
0
100
200
300
400
500
600
700
800
0
25
50
75
100
125
150
175
200
0
10
20
30
40
50