

- Filter
- Help Center
- ML models

Linear regression (70 min)

Linear regression (10 min)

Loss (10 min)

Interactive exercise: Parameters (5 min)

Gradient descent (10 min)

Hyperparameters (10 min)

Interactive exercise: Gradient descent (5 min)

Programming exercise (10 min)

Test your knowledge (10 min)

What's next

Logistic regression (35 min)

Classification (70 min)
- Data

Working with numerical data (85 min)

Working with categorical data (50 min)

Datasets, generalization, and overfitting (105 min)
- Advanced ML models

Neural networks (75 min)

Embeddings (45 min)

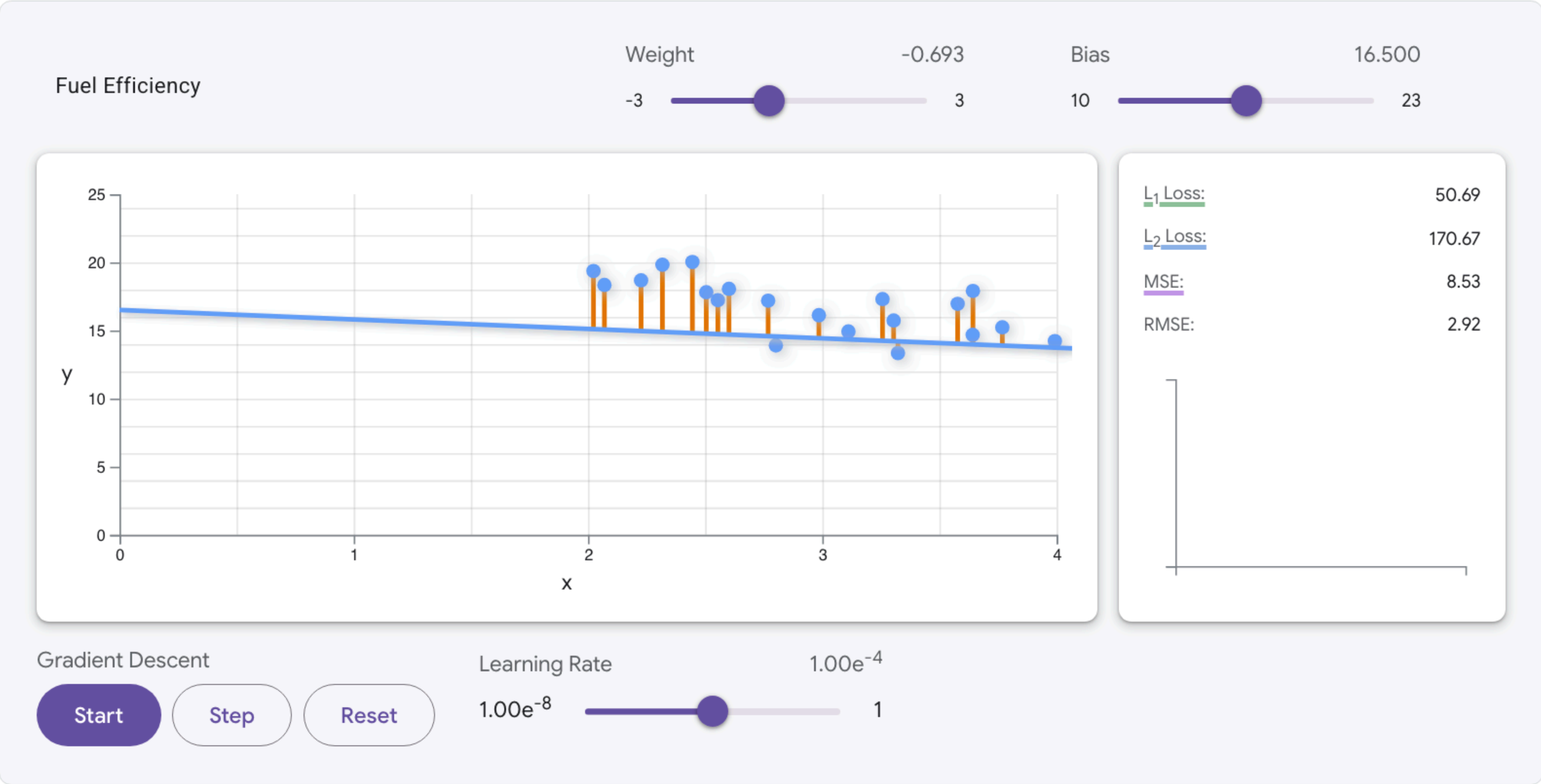
Linear regression: Gradient descent exercise

Send feedback

In this exercise, you'll revisit the graph of fuel-efficiency data from the [Parameters exercise](#). But this time, you'll use gradient descent to learn the optimal weight and bias values for a linear model that minimizes loss.

Complete the three tasks below the graph.

This interactive visualization can produce flashing visuals when set to a high Learning Rate, which may affect photosensitive individuals.



Task #1: Adjust the **Learning Rate** slider below the graph to set a learning rate of 0.03. Click the **Start** button to run gradient descent.

How long does the model training take to converge (reach a stable minimum loss value)? What is the MSE value at model convergence? What weight and bias values produce this value?

Click the plus icon to see our solution

When we set a learning rate of 0.03, the model converged after approximately 30 seconds, achieving a MSE of just under 3 with weight and bias values of -2.08 and 23.098 , respectively. This indicates we've picked a good learning rate value.

Note: The data points in the graph vary slightly each time you load the page, so your solutions here may be a little different than ours.

Task #2: Click the **Reset** button below the graph to reset the Weight and Bias values in the graph. Adjust the **Learning Rate** slider to a value around $1.10e^{-5}$. Click the **Start** button to run gradient descent.

What do you notice about how long it takes the model training to converge this time?

Click the plus icon to see the solution

After several minutes, model training still hasn't converged. Small updates to Weight and Bias values continue to result in slightly lower loss values. This suggests that picking a higher learning rate would enable gradient descent to find the optimal weight and bias values more quickly.

Task #3: Click the **Reset** button below the graph to reset the Weight and Bias values in the graph. Adjust the **Learning Rate** slider up to 1. Click the **Start** button to run gradient descent.

What happens to the loss values as gradient descent runs? How long will model training take to converge this time?

Click the plus icon to see the solution

Loss values fluctuate wildly at high values (MSE over 300). This indicates that the learning rate is too high, and model training will never reach convergence.

[Previous](#) [Next](#)
[Hyperparameters \(10 min\)](#) [Programming exercise \(10 min\)](#)

Was this helpful?



Send feedback

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see the [Google Developers Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2025-06-27 UTC.

Connect

Blog

Instagram

LinkedIn

X (Twitter)

YouTube

Programs

Google Developer Groups

Google Developer Experts

Accelerators

Women Techmakers

Google Cloud & NVIDIA

Developer consoles

Google API Console

Google Cloud Platform Console

Google Play Console

Firebase Console

Actions on Google Console

Cast SDK Developer Console

Chrome Web Store Dashboard

Google Home Developer Console

Google for Developers

Android

Chrome

Firebase

Google Cloud Platform

Google AI

All products

Terms | Privacy

Sign up for the Google for Developers newsletter

Subscribe

English