Filter

- Linear regression (70 min)
- Logistic regression (35 min)
  - Introduction (5 min)
  - Calculating a probability (10 min)
  - Loss and regularization (10 min)
  - Test your knowledge (10 min)
  - What's next
- Classification (70 min)

Data

- Working with numerical data (85 min)
- Working with categorical data (50 min)
- Datasets, generalization, and overfitting (105 min)

Info | Chat | API

Home > Products > Machine Learning > ML Concepts > Crash Course

Was this helpful? Send feedback

Logistic regression: Loss and regularization

On this page
Log Loss

Logistic regression models are trained using the same process as **linear regression** models, with two key distinctions:

- Logistic regression models use **Log Loss** as the loss function instead of **squared loss**.
- Applying regularization is critical to prevent **overfitting**.

The following sections discuss these two considerations in more depth.

Log Loss

In the Linear regression module, you used **squared loss** (also called L_2 loss) as the **loss function**. Squared loss works well for a linear model where the rate of change of the output values is constant. For example, given the linear model $y' = b + 3x_1$, each time you increment the input value x_1 by 1, the output value y' increases by 3.

However, the rate of change of a logistic regression model is *not* constant. As you saw in Calculating a probability, the **sigmoid** curve is s-shaped rather than linear. When the log-odds (z) value is closer to 0, small increases in z result in much larger changes to y than when z is a large positive or negative number. The following table shows the sigmoid function's output for input values from 5 to 10, as well as the corresponding precision required to capture the differences in the results.

input	logistic output	required digits of precision
5	0.993	3
6	0.997	3
7	0.999	3
8	0.9997	4
9	0.9999	4
10	0.99998	5

If you used squared loss to calculate errors for the sigmoid function, as the output got closer and closer to `0` and `1`, you would need more memory to preserve the precision needed to track these values.

Instead, the loss function for logistic regression is **Log Loss**. The Log Loss equation returns the logarithm of the magnitude of the change, rather than just the distance from data to prediction. Log Loss is calculated as follows:

$$\text{Log Loss} = \sum_{(x,y)\in D} -y \log(y') - (1-y) \log(1-y')$$

where:

- $(x, y) \in D$ is the dataset containing many labeled examples, which are (x, y) pairs.
- y is the label in a labeled example. Since this is logistic regression, every value of y must either be 0 or 1.
- y' is your model's prediction (somewhere between 0 and 1), given the set of features in x.

Regularization in logistic regression

Regularization, a mechanism for penalizing model complexity during training, is extremely important in logistic regression modeling. Without regularization, the asymptotic nature of logistic regression would keep driving loss towards 0 in cases where the model has a large number of features. Consequently, most logistic regression models use one of the following two strategies to decrease model complexity:

- L_2 regularization
- Early stopping: Limiting the number of training steps to halt training while loss is still decreasing.

Note: You'll learn more about regularization in the Datasets, Generalization, and Overfitting module of the course.

Key terms:

- Gradient descent
- Linear regression
- Log Loss
- Logistic regression
- Loss function
- Overfitting
- Regularization
- Squared loss

Help Center

Previous
Calculating a probability (10 min)

Next
Test your knowledge (10 min)

Was this helpful?
Send feedback