Generalization: Peril of Overfitting

Estimated Time: 10 minutes

This module focuses on generalization. In order to develop some intuition about this concept, you're going to look at three figures. Assume that each dot in these figures represents a tree's position in a forest. The two colors have the following meanings:

- The blue dots represent sick trees.
- The orange dots represent healthy trees.

With that in mind, take a look at Figure 1.

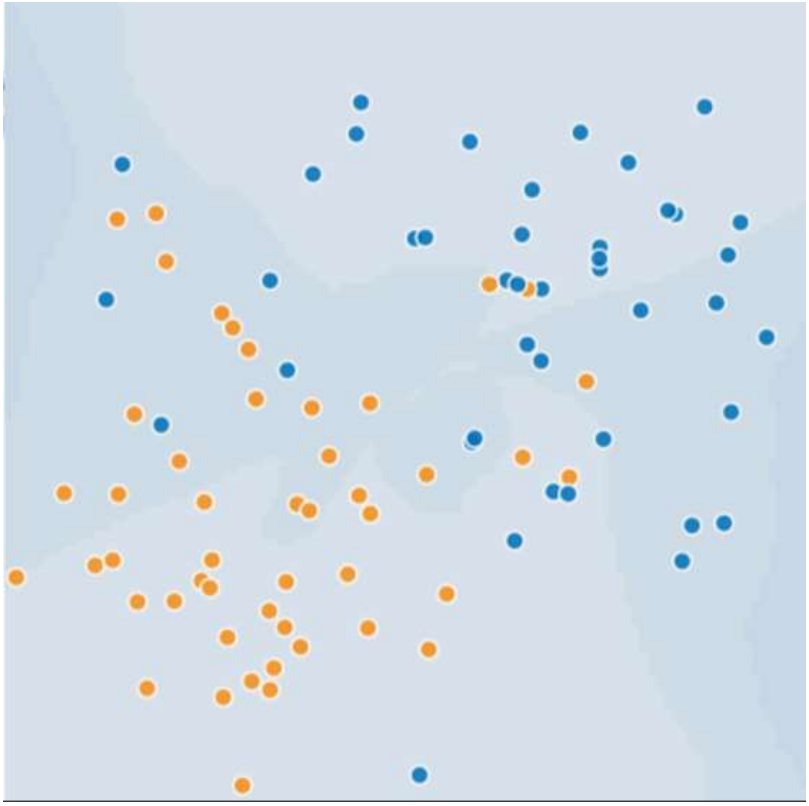

Figure 1. Sick (blue) and healthy (orange) trees.

Can you imagine a good model for predicting subsequent sick or healthy trees? Take a moment to mentally draw an arc that divides the blues from the oranges, or mentally lasso a batch of oranges or blues. Then, look at Figure 2, which shows how a certain machine learning model separated the sick trees from the healthy trees. Note that this model produced a very low loss.

Click the plus icon to see Figure 2.

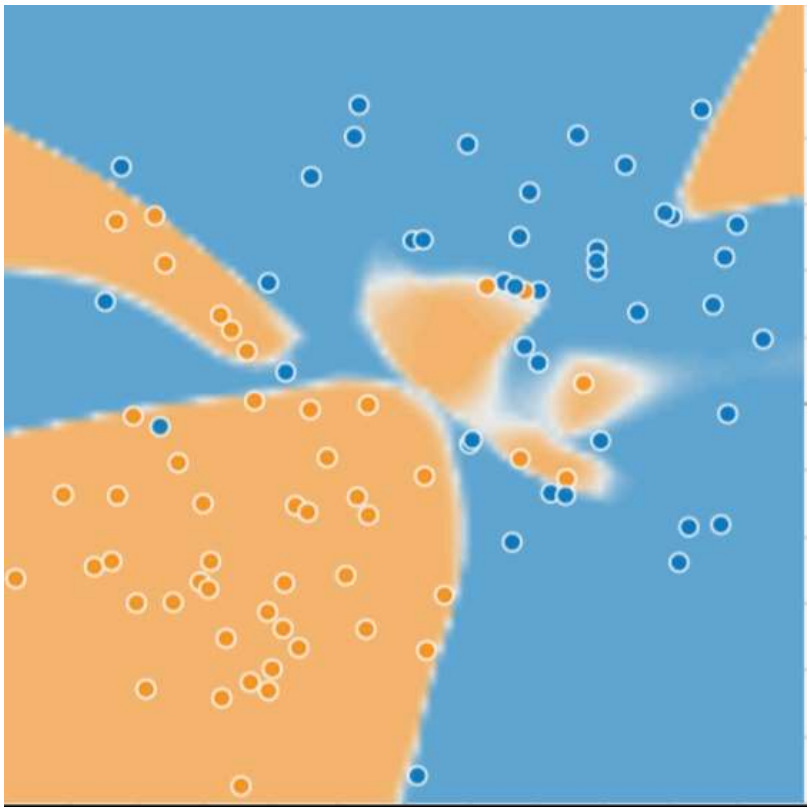

Figure 2. A complex model for distinguishing sick from healthy trees.

At first glance, the model shown in Figure 2 appeared to do an excellent job of separating the healthy trees from the sick ones. Or did it?

Low loss, but still a bad model?

Figure 3 shows what happened when we added new data to the model. It turned out that the model adapted very poorly to the new data. Notice that the model miscategorized much of the new data.

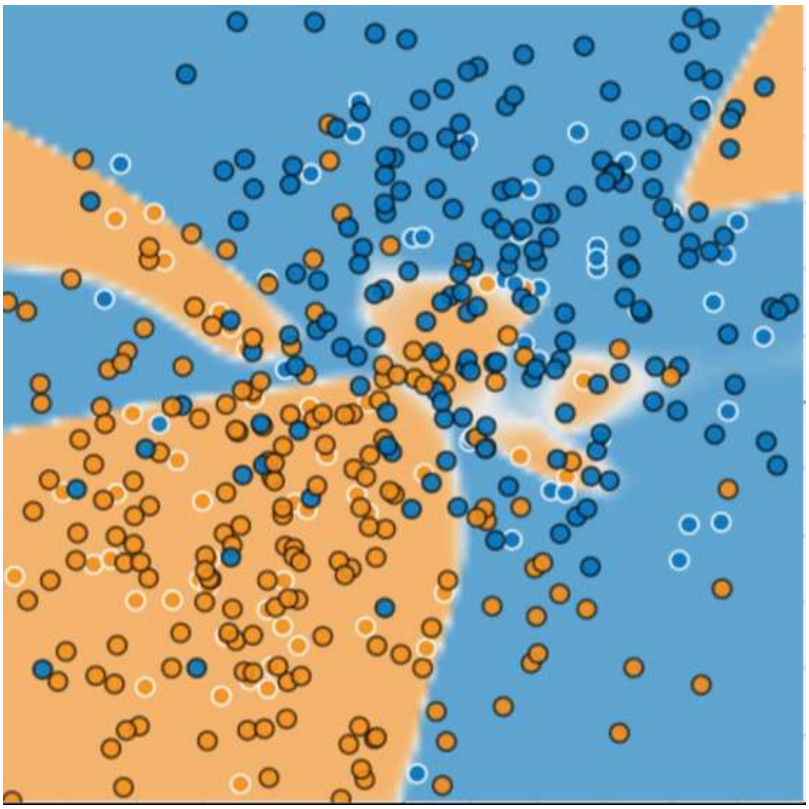

Figure 3. The model did a bad job predicting new data.

The model shown in Figures 2 and 3 **overfits** the peculiarities of the data it trained on. An overfit model gets a low loss during training but does a poor job predicting new data. If a model fits the current sample well, how can we trust that it will make good predictions on new data? As you'll see later on (/machine-learning/crash-course/regularization-for-simplicity/l2-regularization), overfitting is caused by making a model more complex than necessary. The fundamental tension of machine learning is between fitting our data well, but also fitting the data as simply as possible.

Machine learning's goal is to predict well on new data drawn from a (hidden) true probability distribution. Unfortunately, the model can't see the whole truth; the model can only sample from a training data set. If a model fits the current examples well, how can you trust the model will also make good predictions on never-before-seen examples?

William of Ockham, a 14th century friar and philosopher, loved simplicity. He believed that scientists should prefer simpler formulas or theories over more complex ones. To put Ockham's razor in machine learning terms:

> The less complex an ML model, the more likely that a good empirical result is not just due to the peculiarities of the sample.

In modern times, we've formalized Ockham's razor into the fields of **statistical learning theory** and **computational learning theory**. These fields have developed **generalization**

bounds--a statistical description of a model's ability to generalize to new data based on factors such as:

- the complexity of the model
- the model's performance on training data

While the theoretical analysis provides formal guarantees under idealized assumptions, they can be difficult to apply in practice. Machine Learning Crash Course focuses instead on empirical evaluation to judge a model's ability to generalize to new data.

A machine learning model aims to make good predictions on new, previously unseen data. But if you are building a model from your data set, how would you get the previously unseen data? Well, one way is to divide your data set into two subsets:

- **training set**—a subset to train a model.
- **test set**—a subset to test the model.

Good performance on the test set is a useful indicator of good performance on the new data in general, assuming that:

- The test set is large enough.
- You don't cheat by using the same test set over and over.

The ML fine print

The following three basic assumptions guide generalization:

- We draw examples **independently and identically** (**i.i.d**) at random from the distribution. In other words, examples don't influence each other. (An alternate explanation: i.i.d. is a way of referring to the randomness of variables.)
- The distribution is **stationary**; that is the distribution doesn't change within the data set.
- We draw examples from partitions from the **same distribution.**

In practice, we sometimes violate these assumptions. For example:

- Consider a model that chooses ads to display. The i.i.d. assumption would be violated if the model bases its choice of ads, in part, on what ads the user has previously seen.

- Consider a data set that contains retail sales information for a year. User's purchases change seasonally, which would violate stationarity.

When we know that any of the preceding three basic assumptions are violated, we must pay careful attention to metrics.

Summary

- Overfitting occurs when a model tries to fit the training data so closely that it does not generalize well to new data.
- If the key assumptions of supervised ML are not met, then we lose important theoretical guarantees on our ability to predict on new data.

Key Terms

- generalization (/machine-learning/glossary#generalization)
- prediction (/machine-learning/glossary#prediction)
- test set (/machine-learning/glossary#test_set)
- overfitting (/machine-learning/glossary#overfitting)
- stationarity (/machine-learning/glossary#stationarity)
- training set (/machine-learning/glossary#training_set)

Help Center (https://support.google.com/machinelearningeducation)

Previous
← Video Lecture (/machine-learning/crash-course/generalization/video-lecture)

Next
Video Lecture (/machine-learning/crash-course/training-and-test-sets/video-lecture) →

Last updated 2022-07-18 UTC.