Applied Machine Learning

Main Challenges of Machine Learning

Main Challenges of Machine Learning: Data

Insufficient Quantity of Training Data

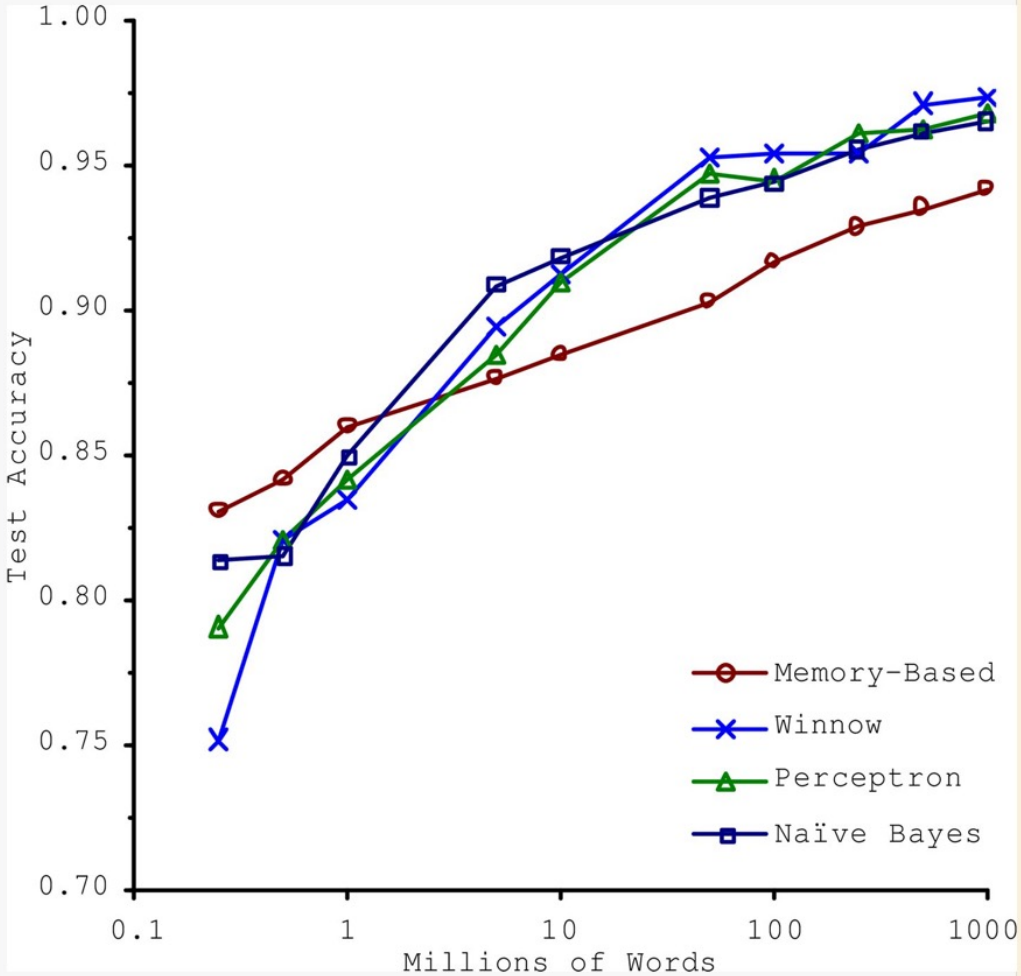


Figure 1-20. The importance of data versus algorithms[9]

Michele & Eric, 2001, Microsoft Research:

As the authors put it: "these results suggest that we may want to reconsider the trade-off between spending time and money on algorithm development versus spending it on corpus development."

The idea that data matters more than algorithms for complex problems was further popularized by Peter Norvig et al. in a paper titled "The Unreasonable Effectiveness of Data" published in 2009.[10] It should be noted, however, that small- and medium-sized datasets are still very common, and it is not always easy or cheap to get extra training data, so don't abandon algorithms just yet.

Non-representative Training Data

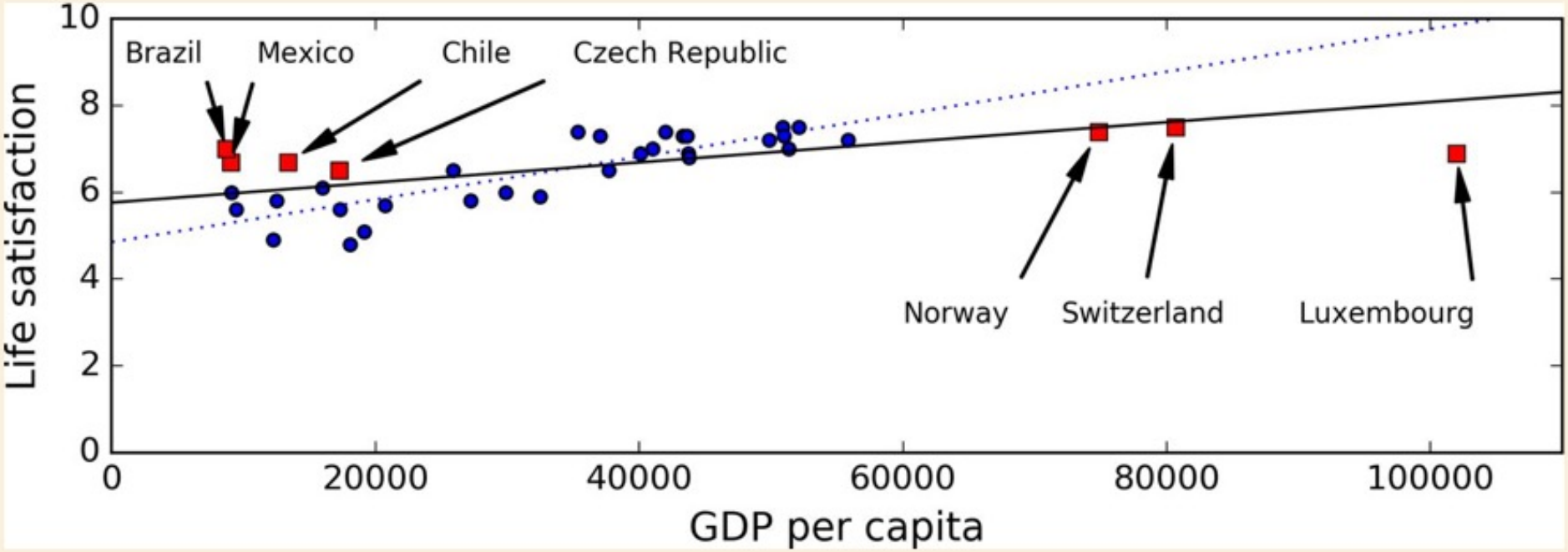
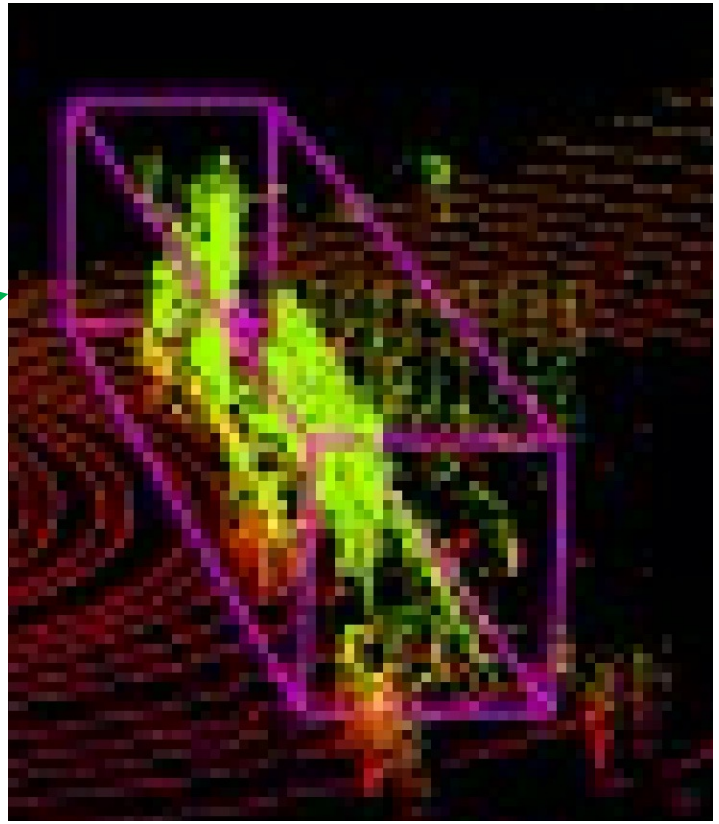


Figure 1-21. A more representative training sample

Poor-Quality Data



Demo MMVC Lab

Image from: https://www.extremetech.com/extreme/213517-a-laser-and-a-raspberry-pi-can-disable-a-self-driving-car

Irrelevant Features

Carefully choose features in your dataset

Photo by Gabe Photos, some rights reserved

Image cited from: https://machinelearningmastery.com/feature-selection-to-improve-accuracy-and-decrease-training-time/

Main Challenges of Machine Learning: Algorithms

Overfitting the Training Data

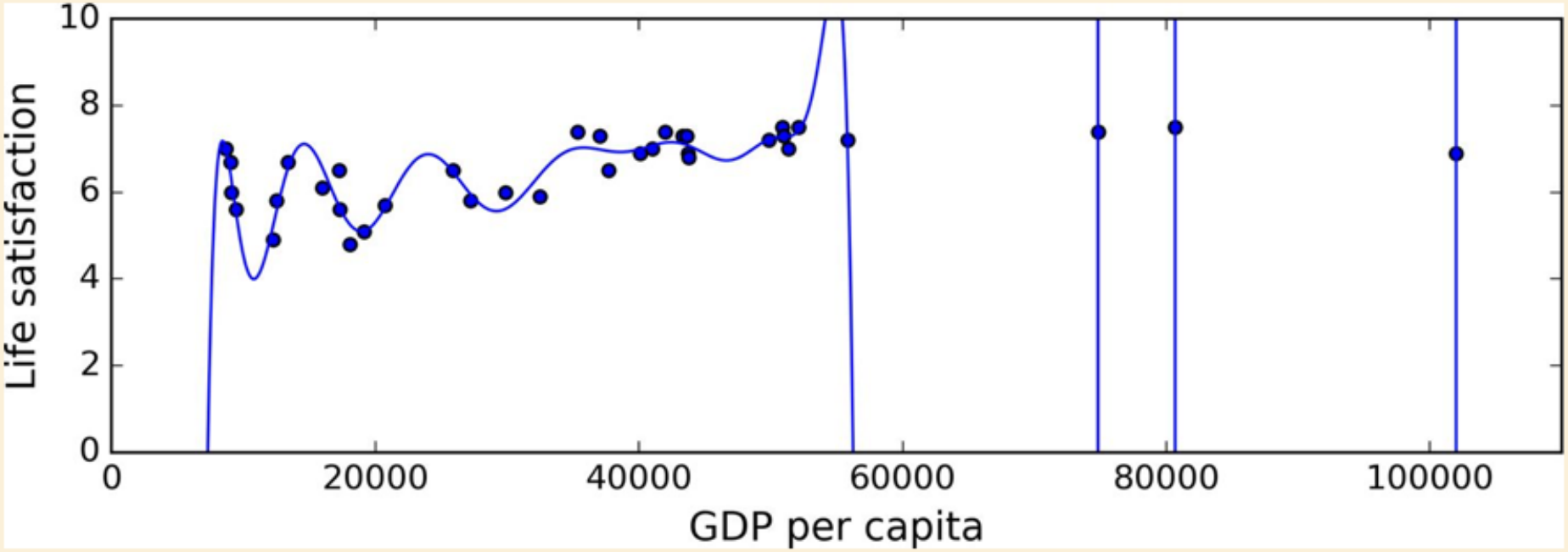


Figure 1-22. Overfitting the training data

WARNING

Overfitting happens when the model is too complex relative to the amount and noisiness of the training data. The possible solutions are:

- To simplify the model by selecting one with fewer parameters (e.g., a linear model rather than a high-degree polynomial model), by reducing the number of attributes in the training data or by constraining the model
- To gather more training data
- To reduce the noise in the training data (e.g., fix data errors and remove outliers)

Regularization Strategy

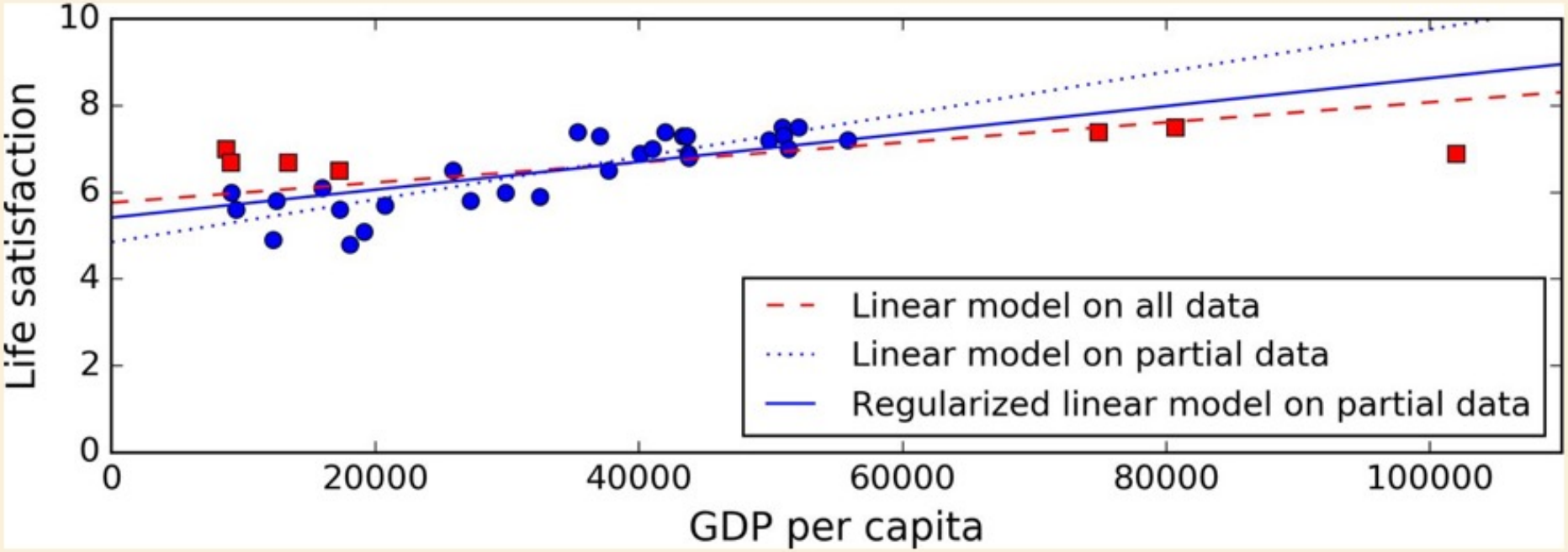


Figure 1-23. Regularization reduces the risk of overfitting

Under-fitting the Training Data

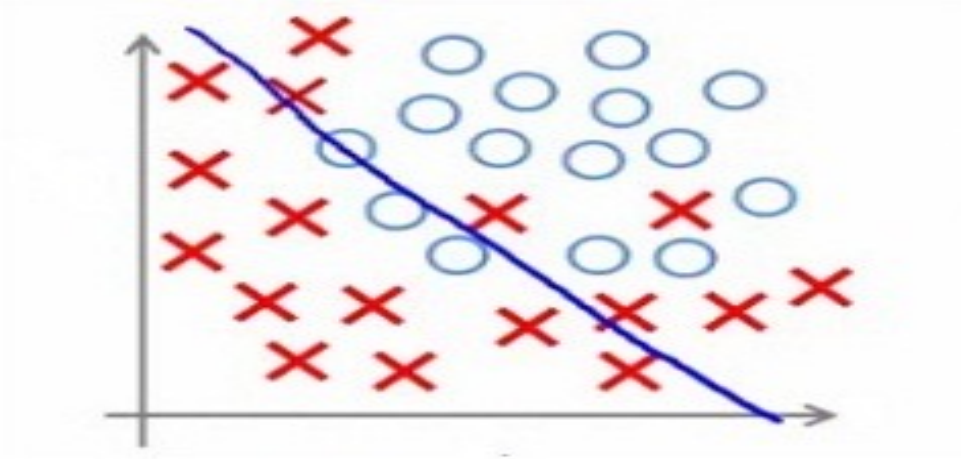


Image cited from: https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76

Fitting the Data

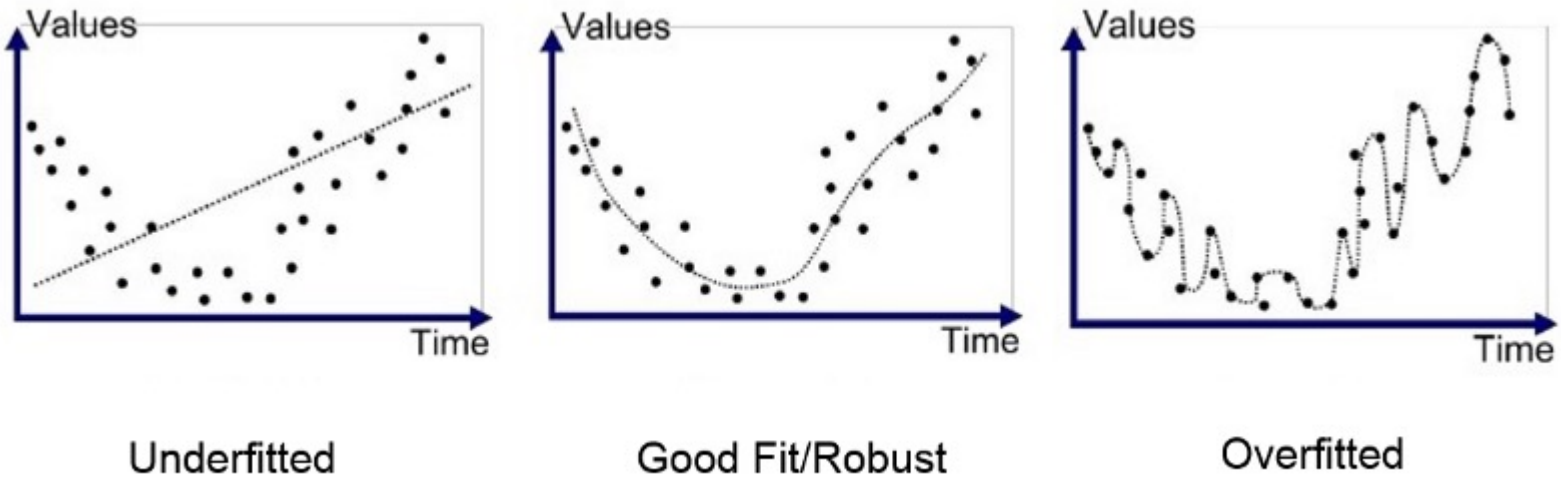


Image cited from: https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76

Summary:

By now you already know a lot about Machine Learning. However, we went through so many concepts that you may be feeling a little lost, so let's step back and look at the big picture:

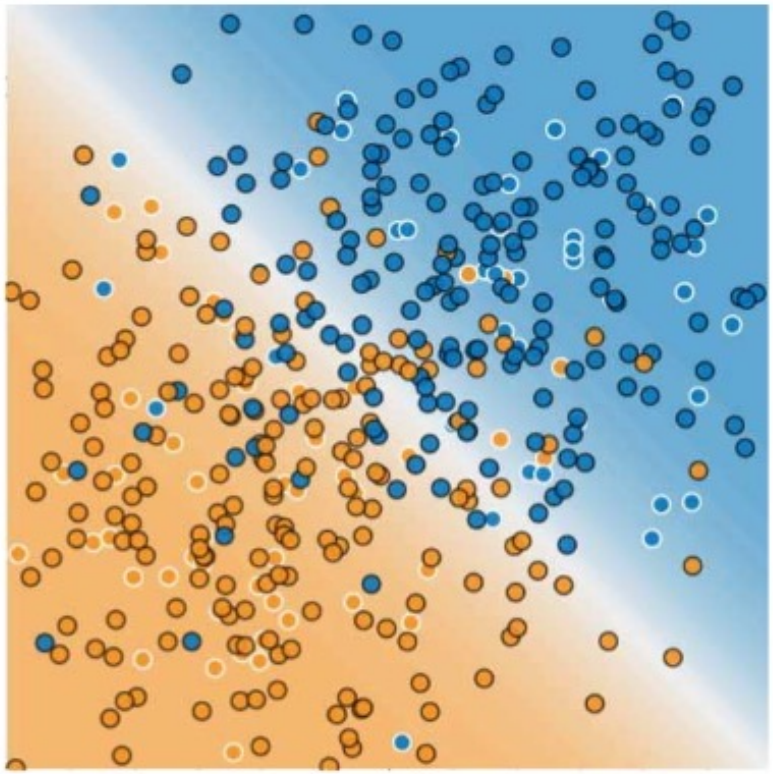
- Machine Learning is about making machines get better at some task by learning from data, instead of having to explicitly code rules.
- There are many different types of ML systems: supervised or not, batch or online, instance-based or model-based, and so on.
- In a ML project you gather data in a training set, and you feed the training set to a learning algorithm. If the algorithm is model-based it tunes some parameters to fit the model to the training set (i.e., to make good predictions on the training set itself), and then hopefully it will be able to make good predictions on new cases as well. If the algorithm is instance-based, it just learns the examples by heart and uses a similarity measure to generalize to new instances.
- The system will not perform well if your training set is too small, or if the data is not representative, noisy, or polluted with irrelevant features (garbage in, garbage out). Lastly, your model needs to be neither too simple (in which case it will underfit) nor too complex (in which case it will overfit).

Main Challenges of Machine Learning: Testing and Validating

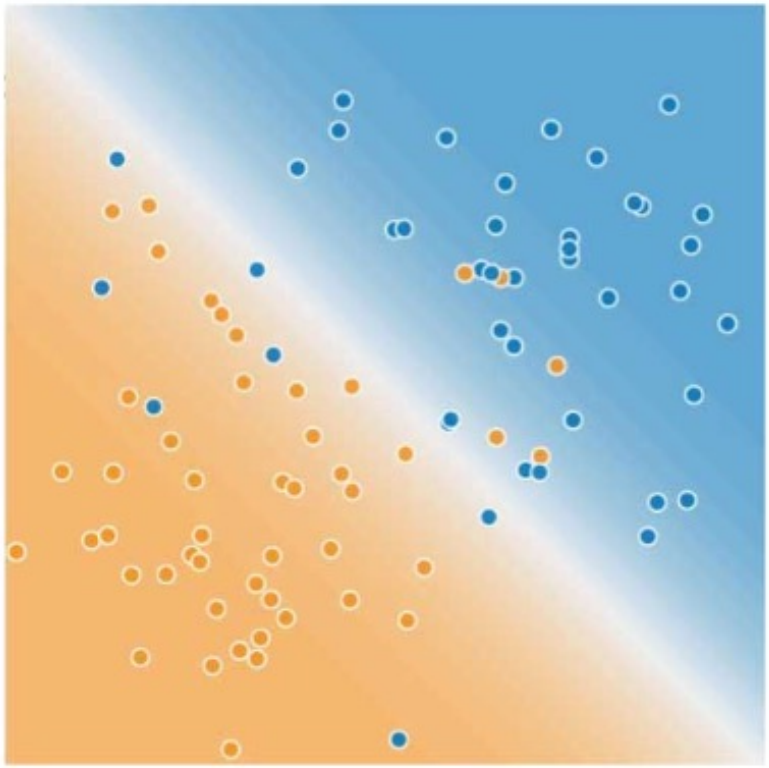
Split the Observed Data

Figure 1. Slicing a single data set into a training set and test set.

Split the Observed Data



Training Data



Test Data

Figure 2. Validating the trained model against test data.

Image cited: https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data

NO FREE LUNCH THEOREM

A model is a simplified version of the observations. The simplifications are meant to discard the superfluous details that are unlikely to generalize to new instances. However, to decide what data to discard and what data to keep, you must make *assumptions*. For example, a linear model makes the assumption that the data is fundamentally linear and that the distance between the instances and the straight line is just noise, which can safely be ignored.

In a famous 1996 paper,[11] David Wolpert demonstrated that if you make absolutely no assumption about the data, then there is no reason to prefer one model over any other. This is called the *No Free Lunch* (NFL) theorem. For some datasets the best model is a linear model, while for other datasets it is a neural network. There is no model that is *a priori* guaranteed to work better (hence the name of the theorem). The only way to know for sure which model is best is to evaluate them all. Since this is not possible, in practice you make some reasonable assumptions about the data and you evaluate only a few reasonable models. For example, for simple tasks you may evaluate linear models with various levels of regularization, and for a complex problem you may evaluate various neural networks.