

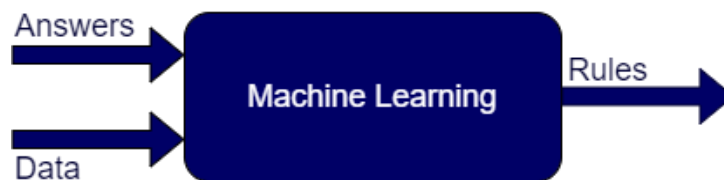
Finding Patterns

Traditional programming is where you explicitly figure out the rules that act on some data to give an answer like this:



Rules and data feed into traditional programming which then outputs answers.

Machine Learning changes this, for scenarios where you may not be able to figure out the rules feasibly, and instead have a computer figure out what they are. That makes the diagram look like this:



Answers and data feed into machine learning and from this rules emerge.

A computer takes the following steps to learn the rules - it makes a guess, then looks at the data to figure out how accurate the guess was, and then makes another guess and so on.

So, consider a given set of numbers:

X: -1, 0, 1, 2, 3, 4

And, consider another given set of numbers:

Y: -3, -1, 1, 3, 5, 7

Can you figure out the relationship between the two sets? There's a function that converts -1 to -3, 0 to -1, 1 to 1, 2 to 3, 3 to 5 and 4 to 7. Can you figure out the relationship? Think about it for a moment.

Often, people see that 0 is matched to -1, so assume Y is (something) times $X - 1$. Maybe they'll take a guess at something, and come up with 3.

Then fill in the gaps, if $Y = 3X - 1$, then

X : -1, 0, 1, 2, 3, 4

becomes

Y : -4, -1, 2, 5, 8, 11

This works only for 0, it fails for everything else. In ML terms, for this guess you can define the loss as 'high'. With what you learned from that, you might think, what if it's $Y = 2X - 1$?

Then, when you fill in the results for $Y = 2X - 1$, you'll get:

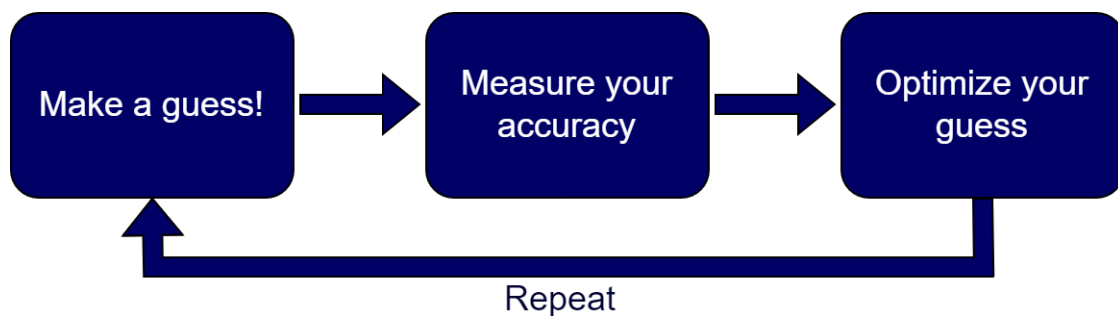
X : -1, 0, 1, 2, 3, 4

becomes

Y : -3, -1, 1, 3, 5, 7

...which matches your original data perfectly. The loss is zero!

You've just gone through this process:



Making a guess leads to measuring your accuracy which leads to optimizing your guess. This then gets repeated where you make a guess again and repeat the process.