INSTITUTE OF COMPUTING SCIENCE & MATHEMATICS

Schools Resources

How Computers Learn In an Uncertain World

How Do We Tell Computers What To Do?

Computers are good at following rules. Computer programs are rules and instructions for the computer to follow, so if you know how to perform a task, you should be able to write a program to allow a computer to perform that task too. This is how most computers are told what to do.

This approach works very well for a large number of the every day tasks that we rely on computers to perform. There are rules for calculating the balance on a bank account or playing an MP3 track and computers can easily be programmed to perform these tasks.

We Don't Always Know How Something is Done

Things get more difficult when we do not know the rules needed to perform a task. If we don't know how to do something, how can we tell a computer how to do it? Another problem arises when there is uncertainty about what the best action would be. When faced with uncertainty, humans are able to guess – make a judgement about what is most likely – but how do we allow a computer to make such judgements and make 'educated guesses'?

Computers That Can Learn

Solving both of these problems – allowing computers to perform tasks that we cannot program and allowing them to make guesses where there is uncertainty – is the challenge of a field of computing known as machine learning. The goal of machine learning is to enable computers to learn how to perform tasks without being programmed even when there is uncertainty about how best to perform a task.

Many games provide us with good examples of the kind of task a computer might be able to learn. Poker, for example has elements of chance, it requires judgement under uncertainty, and there is no agreed set of instructions to follow to ensure that you would win. For a computer to become a good poker player, it would need to learn from experience.

Could a computer learn to play poker?

Probability: Dealing With Uncertainty

The most common method of dealing with uncertainty is to use probability. Probability is often used to describe the expected outcome of some event. For example, if you toss a coin, then there are two possible outcomes: heads or tails. If you bet on a horse winning a race, there are two possible outcomes: you win or you loose. Probability is used to measure how likely each of the possible outcomes are.

Probability is like a score, where the higher the score, the more likely it is that the outcome will happen. Probabilities are always scored between 0 and 1, where a score of 0 means the outcome is impossible, a score of 0.5 means that the outcome is equally likely to happen as it is to not happen, and a score of 1 means the outcome will certainly happen. A probability of 0.8 means that the outcome will happen 8 times out of 10.

In our examples above, the probability of the outcome of the coin toss event being a head is 0.5, but the probability of you picking the winning horse might be 0.2.

It is easy to work out the probability of something happening if you have the opportunity to repeat the event over and over again and count how often each outcome occurs. This is one of the simplest ways of allowing a computer to learn. If I tossed a coin 100 times and got heads 55 times and tails 45 times, and showed each result to a computer, that computer could learn that the probability of getting a head is 55/100 = 0.55. Notice that this isn't quite the correct answer. If the coin is balanced, then the probability of getting a head should be 0.5

Experiment

You will need:

- 1 coin
- 1 piece of blu-tack or similar sticky substance with some weight

Method:

1. Stick the blu-tack to the tails side of the coin,
2. Toss the coin,
3. Use a piece of paper to note whether you got a head or a tail,
4. Repeat the previous 2 steps 20 times,
5. Count the number of heads and the number of tails you got, and divide each count by 20

This gives you the probability of getting a head or a tail with your weighted coin. Check that your two probabilities add up to 1. If they don't, you made a mistake with your division.

Now, see which outcome (heads or tails) has the highest probability. If you were to toss the coin again and this time you had to guess how it would land, which would you choose?

How often would you expect to be correct with this guess?

Pen and Paper Machine Learning

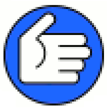
Perform the little experiment above and you will have done a simple piece of machine learning. Using nothing but experience and probability, you have learned the ability to predict whether your weighted coin will land heads or tails. You won't always be right with your prediction, but you will be right more often than you are wrong, which is sometimes all you can hope for.

Making Use of What We Know

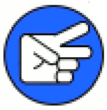
We don't always make guesses in the dark, quite often we have some knowledge that will help us make a guess. In these cases, we use the probability of an outcome given that we already know something that affects that outcome. We call this kind of probability **conditional**. For example we might say "The conditional probability of it raining, given that there is a clear blue sky is 0.05".

The next step in our trip into machine learning is to find out how to use conditional probabilities. For this, we will use the game of Rock, Paper, Scissors as an example.

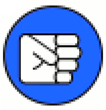
The game of Rock, Paper, Scissors is very simple. Each player picks one of the three objects (usually by making the appropriate hand shape on a count of three!) and these rules are applied to see who has won that round:



Paper wraps (beats) Rock



Scissors cut (beat) Paper



Rock blunts (beats) Scissors

The challenge of the game is to guess what your opponent will choose and pick the appropriate object to beat them. Several rounds of the game are played so the trick is to use your opponent's previous choices to help you guess what they will do next.

People find it quite hard to pick a sequence of perfectly random choices, so any pattern that a player develops could be learned by the opponent and used to win the game.

Hopefully, you can see where the conditional probabilities are used in this example. At each round of the game, each player must ask themselves the question, "What is the conditional probability of my opponent choosing Rock, Paper, and Scissors, given the choice they made on the last round?". Every time I chose Rock, let's say I choose Paper on the following round 8 times out of 10. That means the conditional probability of me picking Paper, given I chose Rock on the previous round, is 0.8. If you had learned that fact, you would pick Scissors and win 8 times out of 10.

You can see this principle in action on this web site:

http://www.cs.stir.ac.uk/~kms/schools/rps/index.php

The computer keeps track of the choices you make and always picks the action to beat the one that you are most likely to pick. Follow the instructions on the web site, play a few games and then discuss the following questions:

1. How easy is it to make sure your choices are totally random? What makes one pattern random and another pattern not random? Think about your answer in terms of probabilities.
2. The computer has no strategy other than picking the action to beat the one that you are most likely to pick. How could you use this fact to beat the computer?
3. As you play more and more rounds, the impact of each round you play gets smaller and smaller. Why is this, and what could the computer do to avoid this problem?