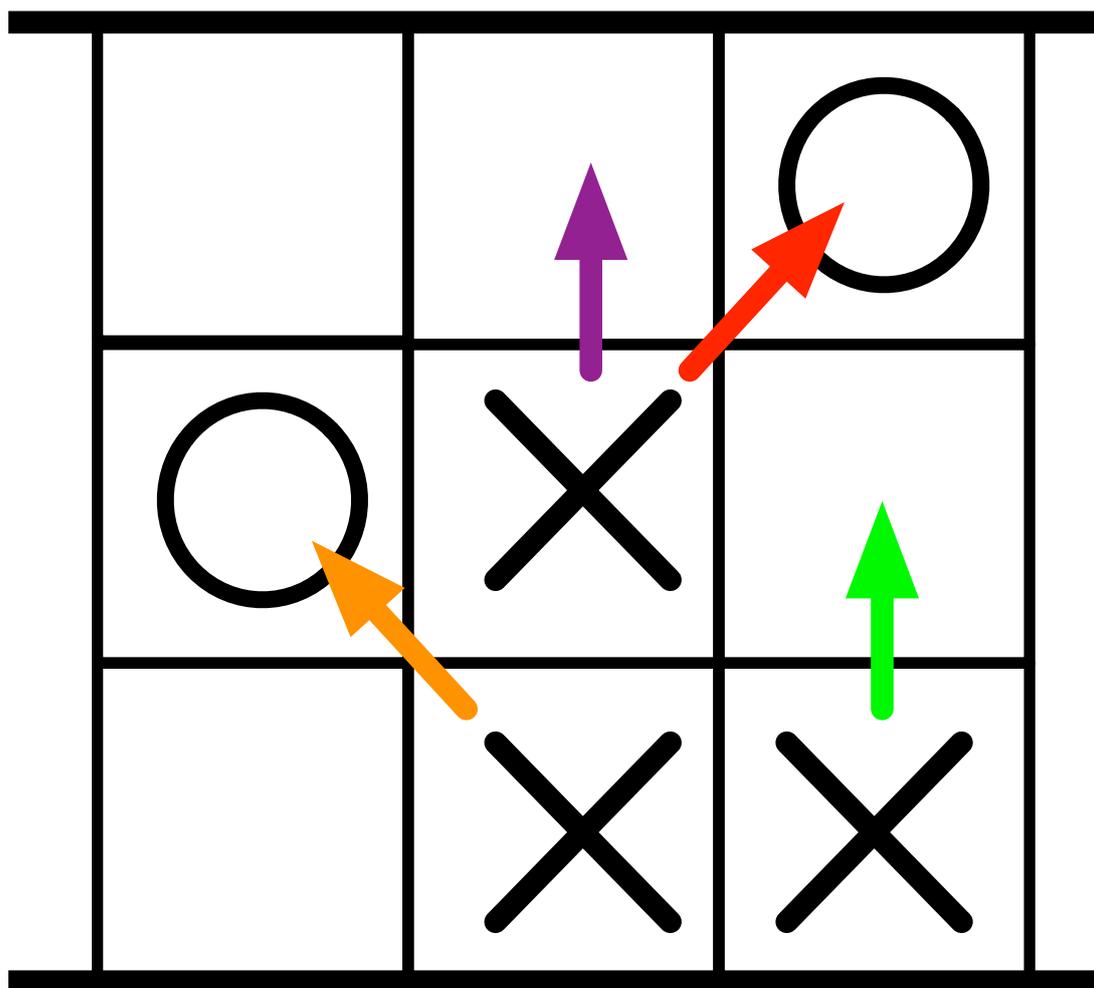


# CS4FN

A cs4fn / Teaching London Computing Special

## The Sweet Learning Computer

Making a machine that learns



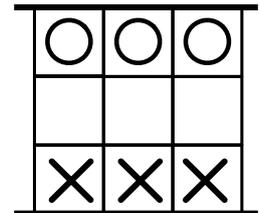
# The Sweet Learning Computer

How do machines learn? Don't they just blindly follow rules? You can build a machine just from cups and sweets that learns how to beat humans at simple games. It learns from its mistakes (because you eat it's sweets when it loses!)

Let's see how by building one to play Hexapawn.

## The Rules of Hexapawn

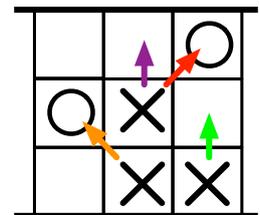
The game of Hexapawn is played on a 3x3 board. Three X and three O pieces are placed on the first and last rows of the board as shown to the right.



It's a little like playing a mini-game of chess with only pawns. To make a move you can either:

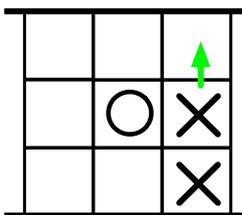
1. Move one place forward if nothing is there, or
2. Take an opponent's piece that's in the next place diagonally

For example, the board to the right gives the four possible moves for the player playing X from that position.

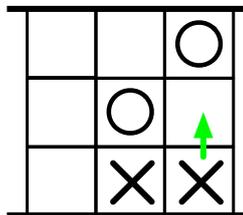


There are 3 ways to win. The examples below show winning moves by X in each way.

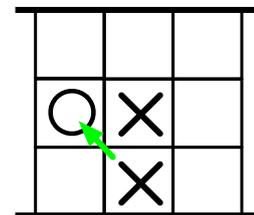
1. Getting a piece onto the last row.



2. Putting the other player in a position so that they can't move.



3. Taking all the other player's pieces

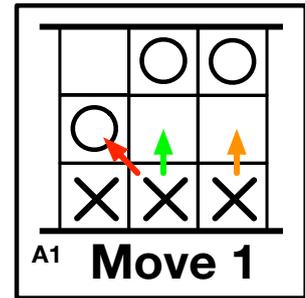


That's all there is to it. Play a few games to get the idea.

## Setting up a Sweet Learning Computer

To create the Sweet Learning Computer:

1. Stick each of the board position pictures on to a plastic cup.
2. Spread the cups out, grouping those for each move together so they are easy to find.
3. Place wrapped coloured sweets that match the coloured arrows shown on its board in to each cup. For example, the cup for position marked A1 (right) has a red, green, and orange sweet.
4. Put the pieces on the board in the start position (see the rules).



## Playing the Game

The Sweet Learning Computer plays second and is X. Whenever it is the computer's turn:

1. Find the cup corresponding to the game position.
2. If there are no sweets in the selected cup, the Sweet Computer resigns.
3. If there are sweets in the selected cup, then pick it up, cover the top with your hand, shut your eyes, shake the cup and take out a sweet at random.
4. Look at the colour of the sweet. Make the move shown by that coloured arrow. For example, if at board position A1 (above) and you pick an orange sweet, the machine's move is to push the right-most X a place forwards as shown by the orange arrow.
5. Place the sweet next to the cup to show the move made.

Note that on the first player's first move, playing down the left or right give symmetrical positions. As they are equivalent we've only provided one of the symmetrical sets of positions (those leading from position A1 above). That means for this version of the game the first player is only allowed to move the left or centre pieces on their first move. We've done this to speed up how quickly the machine learns. Once you understand how it works, extend the machine with the missing

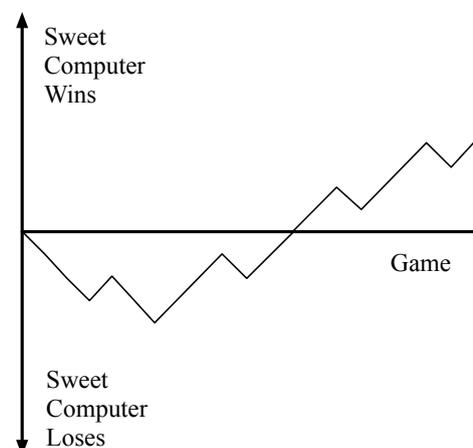
## Learning by making mistakes

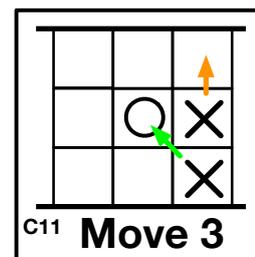
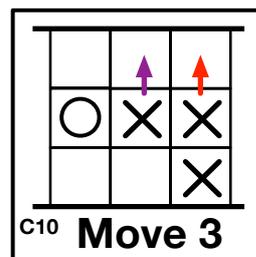
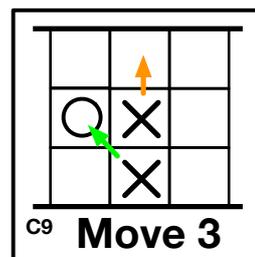
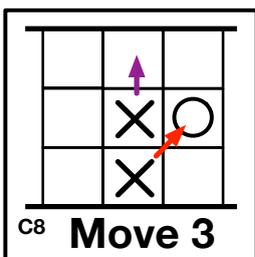
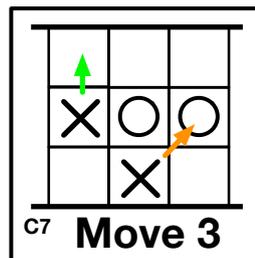
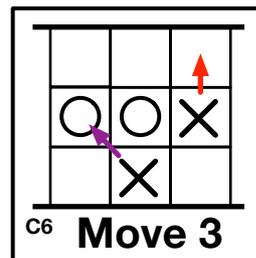
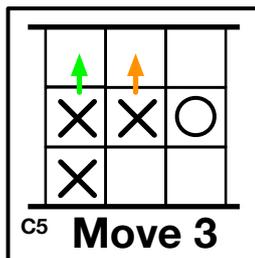
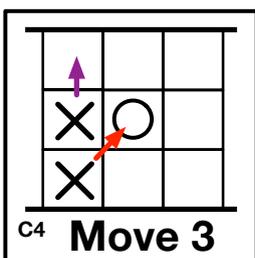
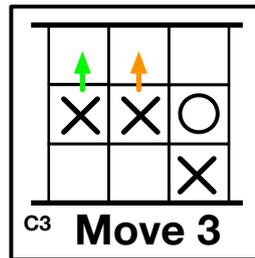
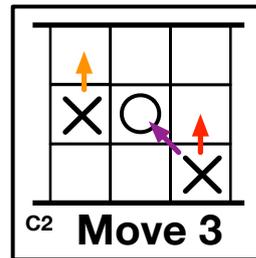
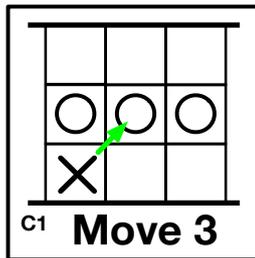
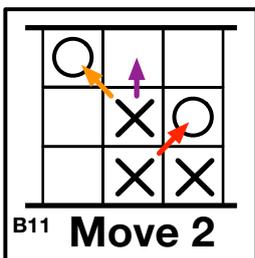
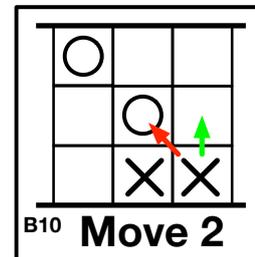
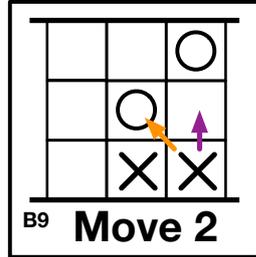
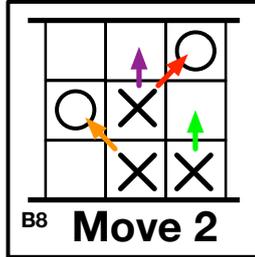
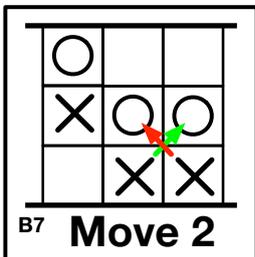
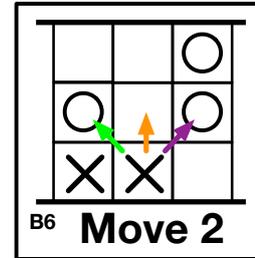
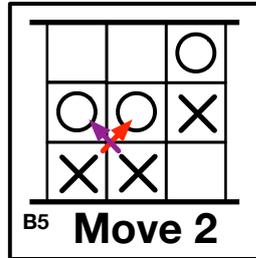
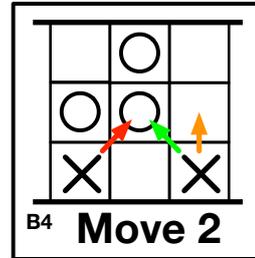
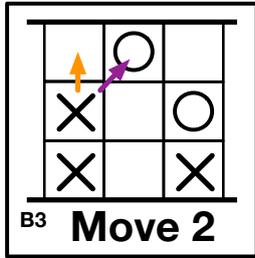
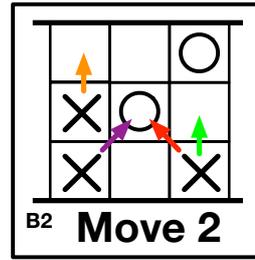
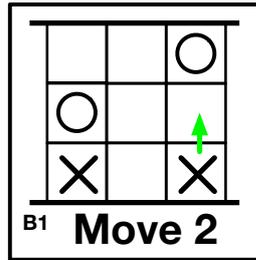
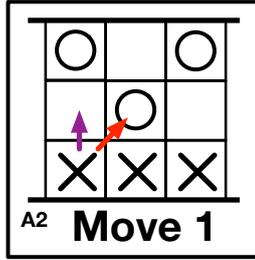
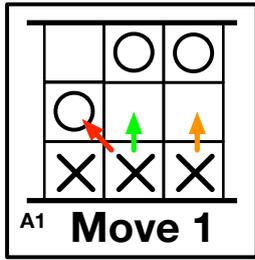
To allow the Sweet Learning Computer to learn from its mistakes you eat it's sweets as follows:

1. At the end of any game that it loses, eat the sweet corresponding to the last move it made.
2. Always place all the other sweets back in the cups they came from.

## What's going on

At the start, the Sweet Computer plays randomly so will lose a lot. Every time it does lose it will not make its last, losing, move again (as the sweet that stands for that move isn't there for future games). Eventually positions that always lead to a loss end up with no sweets so the machine resigns. The sweet in the previous move that led to it being in that position is then eaten...it starts to learn about bad second moves and eventually bad first moves. Over time it gets better and better, until it never makes a mistake as all the bad moves have gone. The sweets left represent moves for a perfect game. Plot a graph of when it wins and loses to see its progress learning.





SUPPORTED BY  
**MAYOR OF LONDON**



**More Machine Learning Fun at**  
**[www.cs4fn.org/machinelearning](http://www.cs4fn.org/machinelearning)**

Computer Science with a sense of fun: Sweet Computer Guide V2.1 (10 Jun 2016)  
Created by Peter McOwan and Paul Curzon Queen Mary University of London,  
Based on the Matchbox Computers of Donald Michie and Martin Gardner  
Teaching London Computing: [teachinglondoncomputing.org](http://teachinglondoncomputing.org)

