

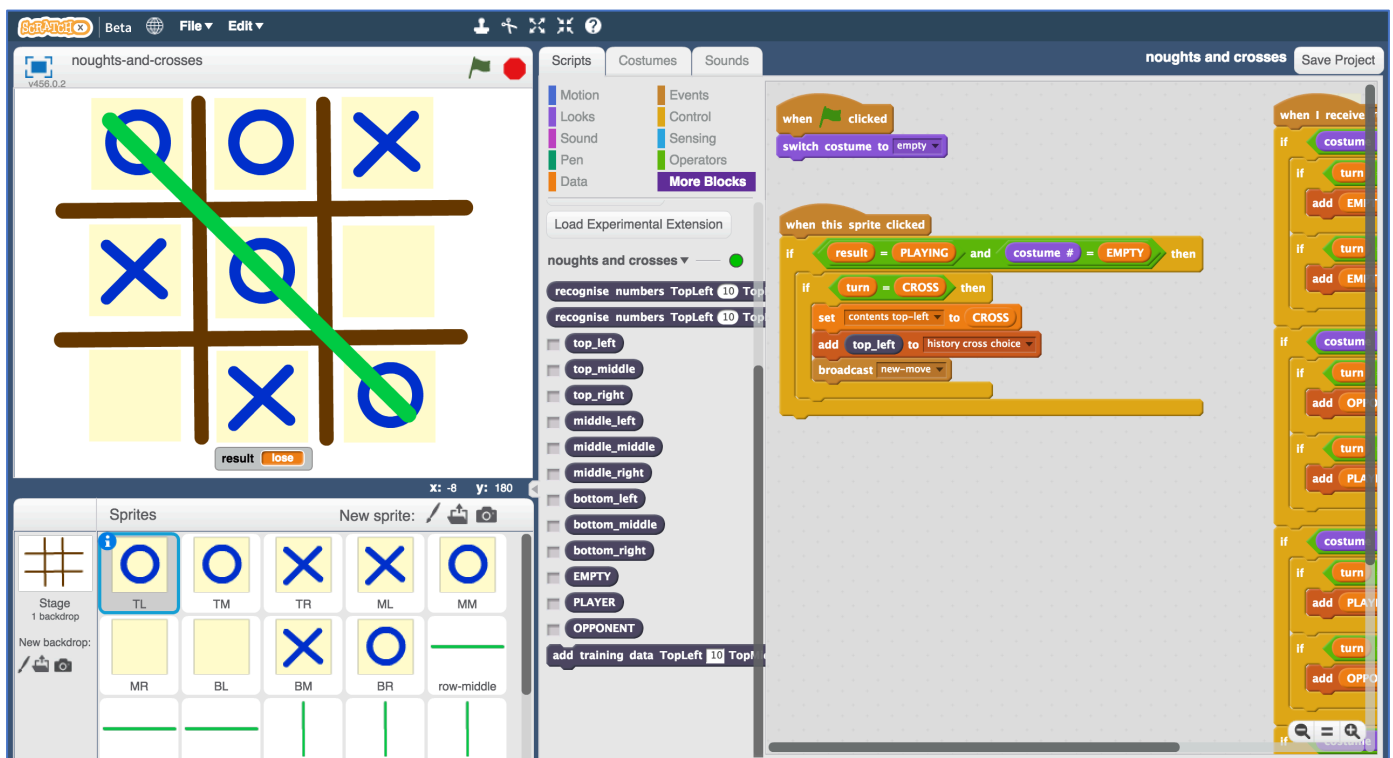


Noughts & Crosses

In this project you will create a noughts and crosses game in Scratch that is able to learn from how you play.

You won't give it instructions for how to play, or tell it what the objective or rules of the game are.

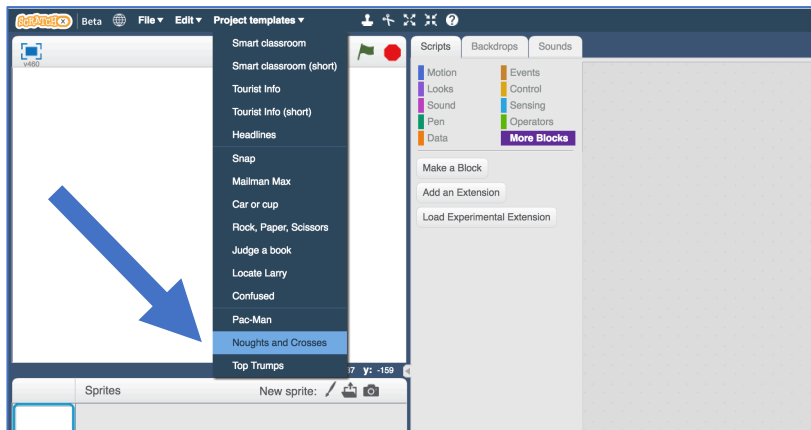
Instead, you'll show it examples of you playing the game. When it's seen enough examples to start trying to play for itself, you'll tell it when it beats you.



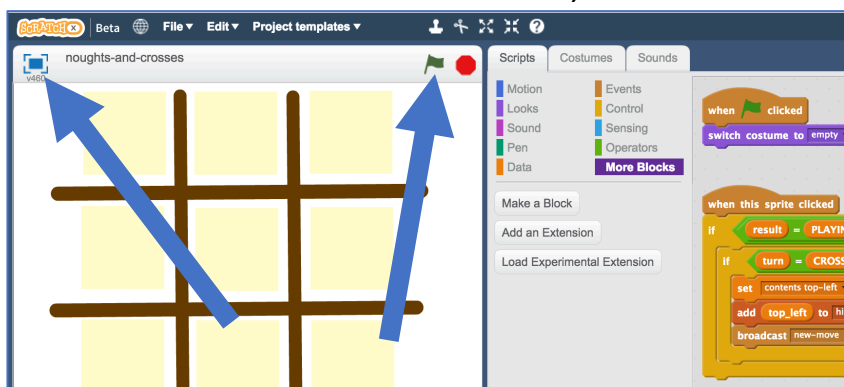
This project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

1. Go to <https://machinelearningforkids.co.uk/scratchx> in a browser.
2. Open the Noughts and Crosses project template

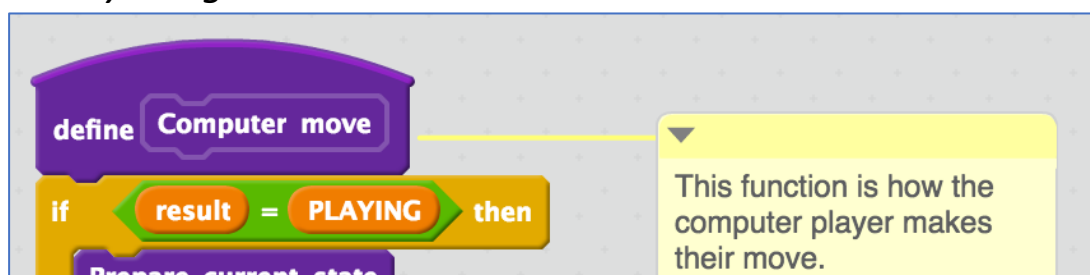
Click **Project templates** -> **Noughts and Crosses**



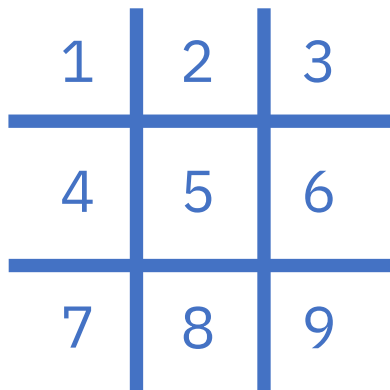
3. Click the **full-screen** button, and then click the Green Flag



4. Play a few games of noughts and crosses
You are CROSS (X), the computer is playing as NOUGHTS (O).
Click the green flag to start a new game, then click on the game board.
5. Can you see how the computer is choosing where to put its moves?
*When you've worked it out, look at the **Computer move** block in the **Stage***
Were you right?

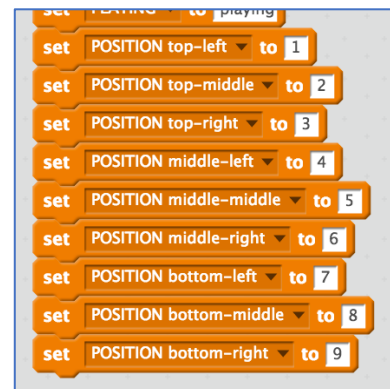


Representing noughts and crosses in Scratch



The positions of spaces on the noughts and crosses board are numbered from 1 to 9.

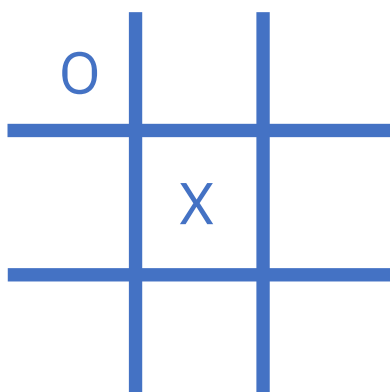
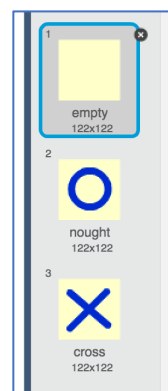
Data constants are used to make it easier to refer to them in scripts.



Empty = 1
O = 2
X = 3

An empty space is shown in costume 1.
A nought O is shown in costume 2.
A cross X is shown in costume 3.

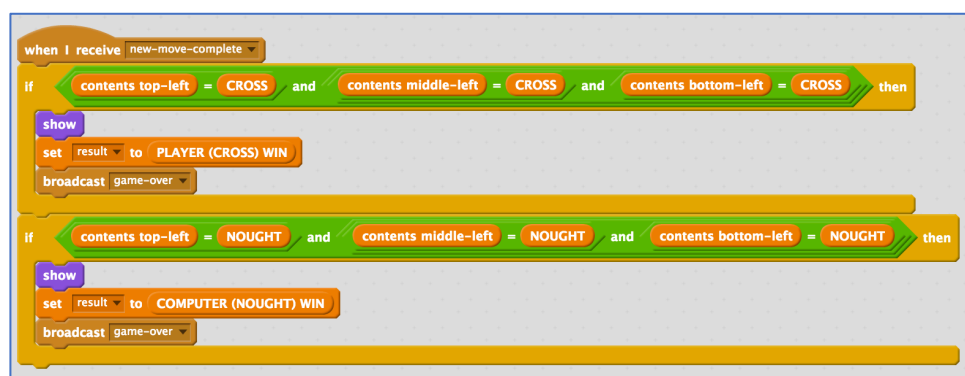
Data constants are used to make it easier to refer to these in scripts.



Variables are used to store the current state of the game.

For example, at this point:

contents top-left = 2
contents middle-middle = 3
contents bottom-right = 1



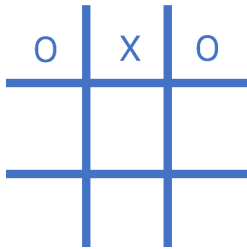
Each of the green row and column sprites check to see if someone has won.

This happens after every move.

What are you going to do?

You're going to train a computer to play noughts and crosses. You'll do this by showing it examples of how you play the game.

Imagine the board looks like this and it's X's turn.

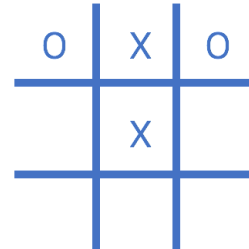


Imagine you decide to put your X in the centre space.

top-left	opponent
top-middle	player
top-right	opponent
middle-left	empty
middle-middle	empty
middle-right	empty
bottom-left	empty
bottom-middle	empty
bottom-right	empty

choice : middle-middle

Imagine the board looks like this and it's O's turn.



Imagine you decide to put your O in the bottom middle space.

top-left	player
top-middle	opponent
top-right	player
middle-left	empty
middle-middle	opponent
middle-right	empty
bottom-left	empty
bottom-middle	empty
bottom-right	empty

choice : bottom-middle

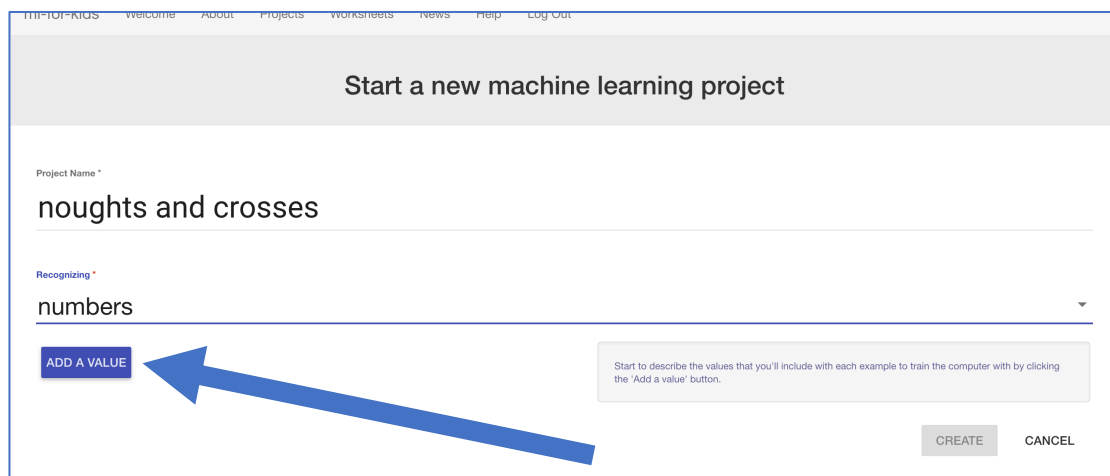
Using “opponent” and “player” instead of “nought” and “cross” means the computer can learn from both nought and cross moves.

You'll use examples of moves from the player that wins the game to train the computer.

If you (X) win, you'll use your moves as examples to train the computer. If the computer (O) wins, you'll use the computer's moves to train with.

These **examples of moves that lead to winning** will teach the computer how to play to win!

6. Close Scratch window then go to <https://machinelearningforkids.co.uk>
7. Click on **“Get started”**
8. Click on **“Log In”** and type in your username and password
If you don't have a username, ask your teacher to create one for you.
If you can't remember your password, ask your teacher to reset it for you.
9. Click on **“Projects”** on the top menu bar
10. Click on the **“+ Add a new project”** button.
11. Name your project **“noughts and crosses”** and set it to learn how to recognise **“numbers”**
12. Click **“Add a value”**



Start a new machine learning project

Project Name *

noughts and crosses

Recognizing *

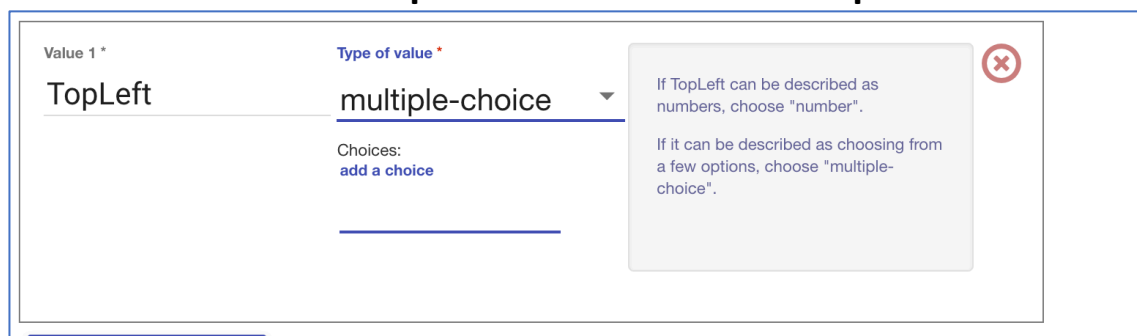
numbers

ADD A VALUE

Start to describe the values that you'll include with each example to train the computer with by clicking the 'Add a value' button.

CREATE CANCEL

13. Name a value **“TopLeft”** and make it **“multiple-choice”**



Value 1 *

TopLeft

Type of value *

multiple-choice

Choices:

add a choice

If TopLeft can be described as numbers, choose "number".

If it can be described as choosing from a few options, choose "multiple-choice".

- 14.** Type “EMPTY” into the “add a choice” box and press Enter
Type “PLAYER” into the “add a choice” box and press Enter
Type “OPPONENT” into the “add a choice” box and press Enter
These are the possible contents for the top-left space in the noughts and crosses board. It can be empty, or it can have the player’s own shape (cross) in it, or it can have the opponent’s shape in it (nought).

The screenshot shows the configuration for 'Value 1' with the label 'TopLeft'. The 'Type of value' is set to 'multiple-choice'. Below this, there are three choices: 'EMPTY', 'PLAYER', and 'OPPONENT', each with a small red 'x' icon. An 'add a choice' link is visible below the choices. To the right, there is a text box with the instruction: 'Type in another choice to use in your multiple-choice list, then press Enter.' and a red 'x' icon.

- 15.** Click “Add another value”

The screenshot shows the 'noughts and crosses' project configuration. The 'Recognizing' dropdown is set to 'numbers'. The 'Value 1' configuration is visible, showing 'TopLeft' as a 'multiple-choice' value with choices 'EMPTY', 'PLAYER', and 'OPPONENT'. A large blue arrow points from the bottom right towards the 'ADD ANOTHER VALUE' button at the bottom left. The 'CREATE' and 'CANCEL' buttons are at the bottom right.

- 16.** Call the next value “TopMiddle” and make it “multiple-choice”

The screenshot shows the 'noughts and crosses' project configuration with two values. The first value, 'TopLeft', is a 'multiple-choice' type with choices 'EMPTY', 'PLAYER', and 'OPPONENT'. The second value, 'TopMiddle', is also a 'multiple-choice' type. The 'ADD ANOTHER VALUE' button is at the bottom left.

17. Add “EMPTY”, “PLAYER”, and “OPPONENT” to “TopMiddle” as you did to “TopLeft”

18. Repeat for the other positions on a noughts-and-crosses board
Each example is the state of the board before a move that led to a win.
TopLeft, TopMiddle, TopRight,
MiddleLeft, MiddleMiddle, MiddleRight,
BottomLeft, BottomMiddle, BottomRight

*It's **very important** that you spell “EMPTY”, “PLAYER” and “OPPONENT” in the same way for all nine positions.*

19. Click “Create”

Project Name: noughts and crosses

Recognizing: numbers

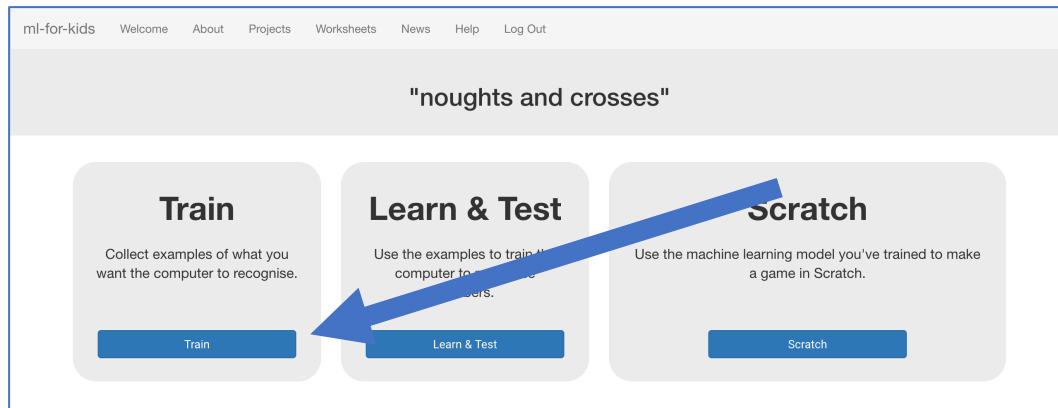
Value 1 *	Type of value *	Value 2 *	Type of value *	Value 3 *	Type of value *
TopLeft	multiple-choice	TopMiddle	multiple-choice	TopRight	multiple-choice
Value 4 *	Type of value *	Value 5 *	Type of value *	Value 6 *	Type of value *
MiddleLeft	multiple-choice	MiddleMiddle	multiple-choice	MiddleRight	multiple-choice
Value 7 *	Type of value *	Value 8 *	Type of value *	Value 9 *	Type of value *
BottomLeft	multiple-choice	BottomMiddle	multiple-choice	BottomRight	multiple-choice

ADD ANOTHER VALUE

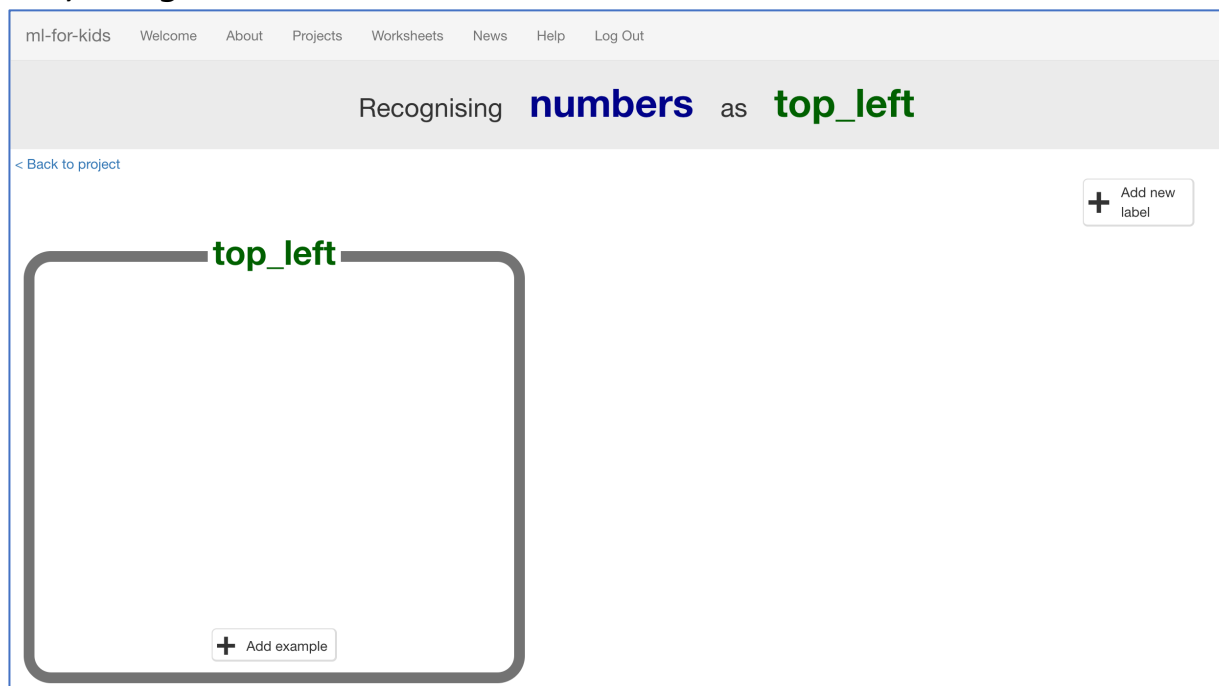
CREATE CANCEL

20. You should see “noughts and crosses” in your list of projects.
Click on it.

21. Click the “Train” button



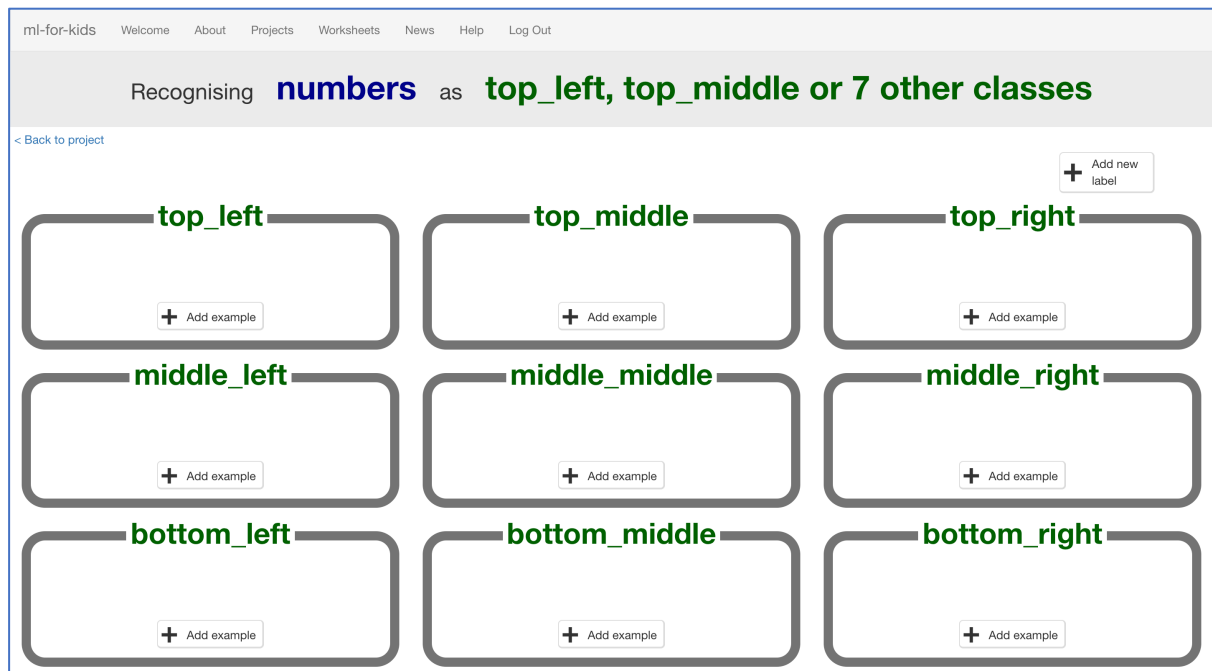
22. Click “+ Add new label” and create a label called “top left” *Examples of making a move in the top-left box (in games that lead to a win) will go in this bucket.*



23. Click “+ Add new label” again and create labels for the other eight spaces on the board.

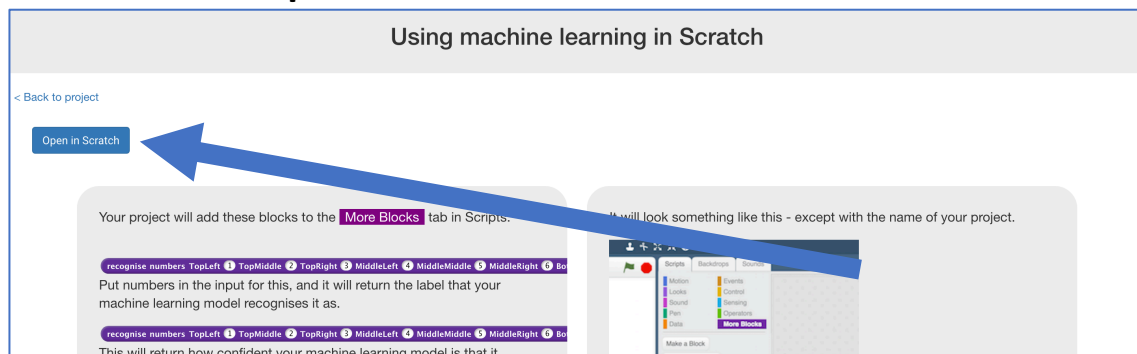
*“top middle”, “top right”,
“middle left”, “middle middle”, “middle right”,
“bottom left”, “bottom middle”, “bottom right”*

(see the next page for a picture)

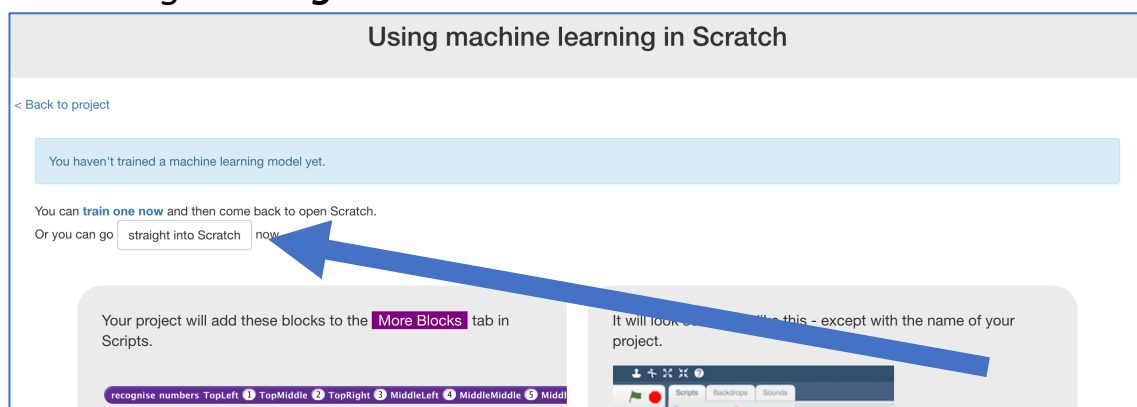


24. Click the “< Back to project” link then click **Scratch**

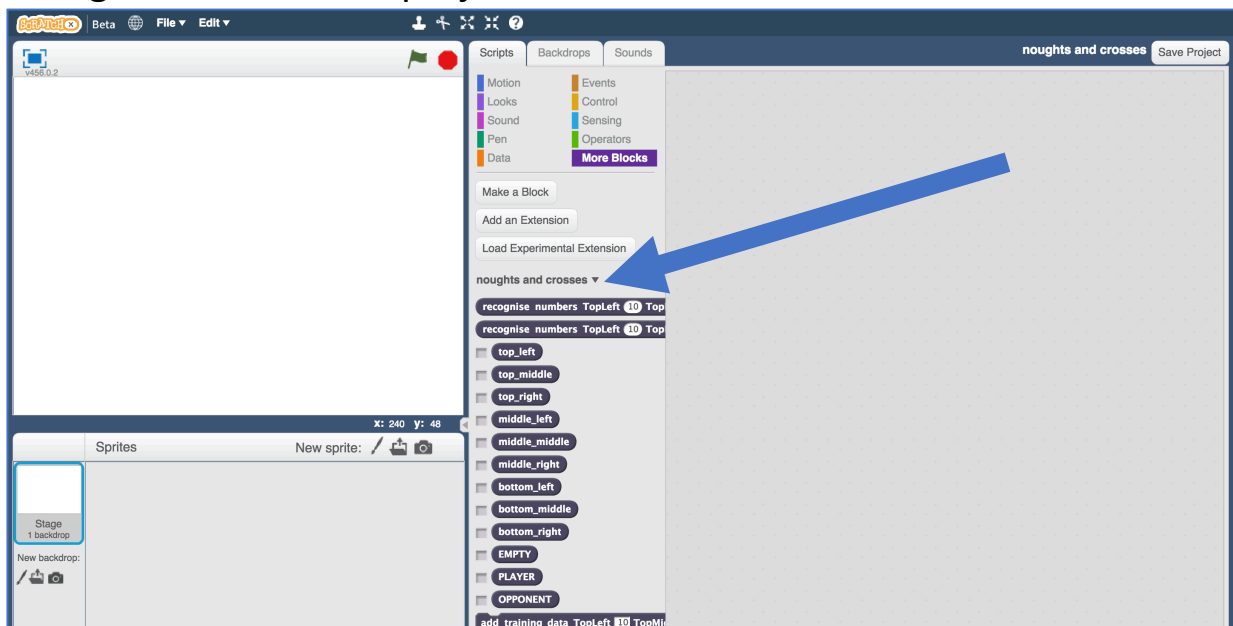
25. Click the **Open in Scratch** button



26. You’ll get a warning that you haven’t trained the computer yet
That’s okay – you’ll use Scratch to collect training examples.
 Click the go “**straight into Scratch**” now button.

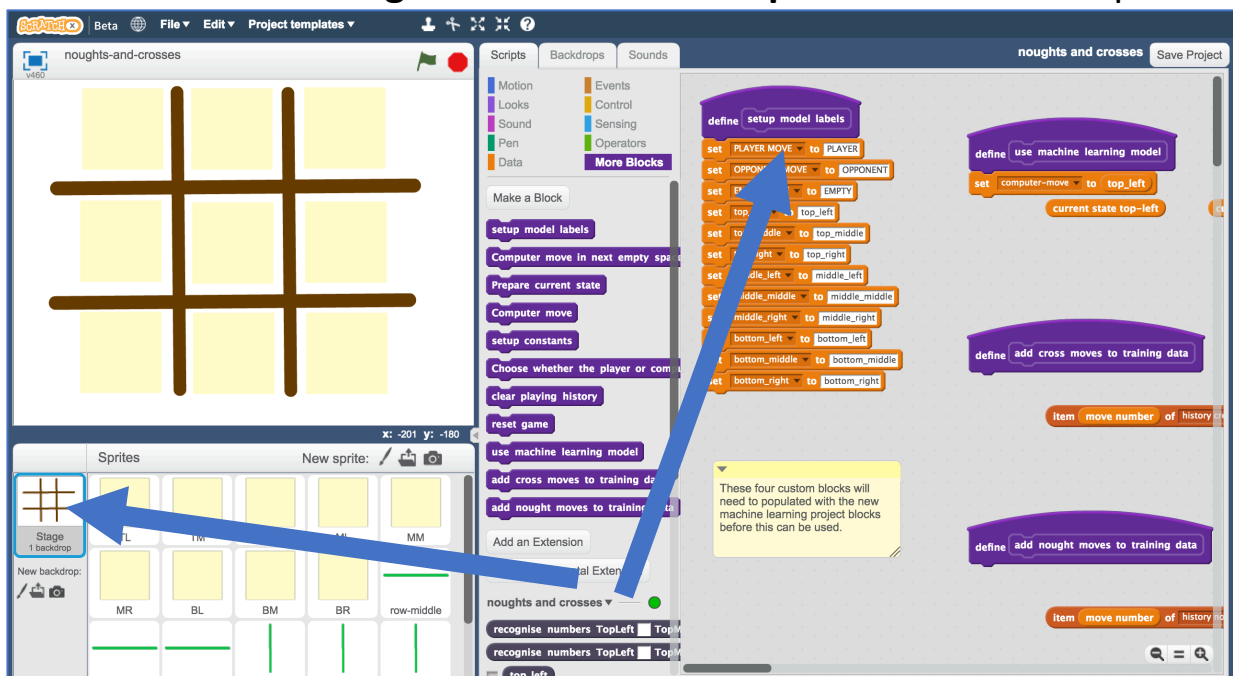


27. You should see new blocks in the “**More blocks**” section from your “noughts and crosses” project.



28. Open the Noughts and Crosses project template again.
Click **Project templates** -> **Noughts and Crosses**

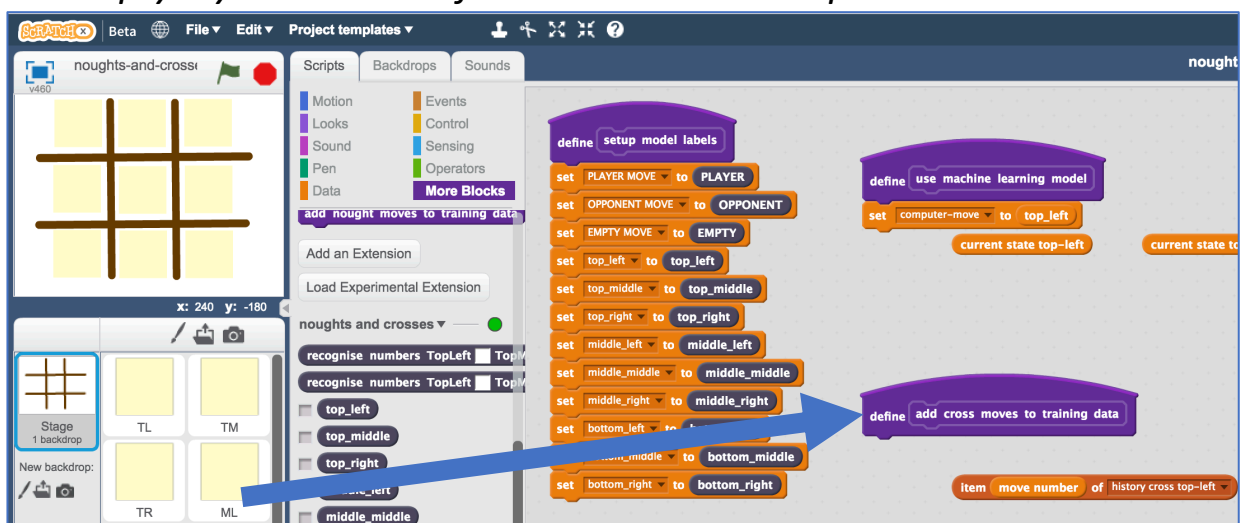
29. Click on the “**Stage**” and find the “**setup model labels**” script



- 30.** Modify the script to use your new project blocks
Update it to look like this, by copying in the dark-blue blocks from your project

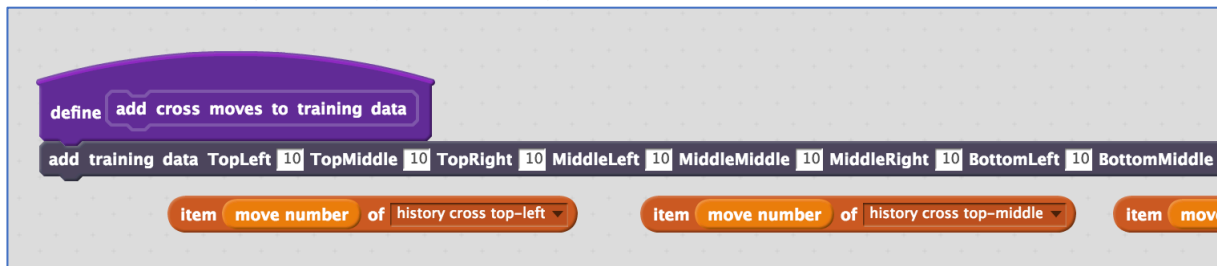


- 31.** Find the “add cross moves to training data” script
It’s empty – you’ll need to fill it in in the next step.



32. Add the “add training data” block

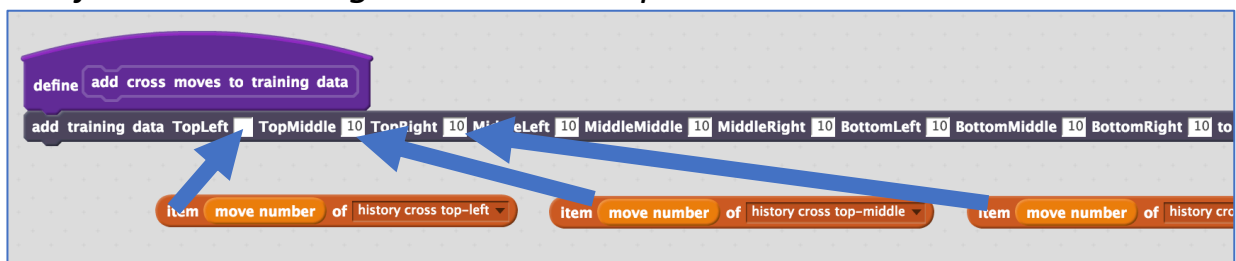
There are two that start with “add training data” – you want the second one, at the very end of the list of blocks, that ends in a “label” text box



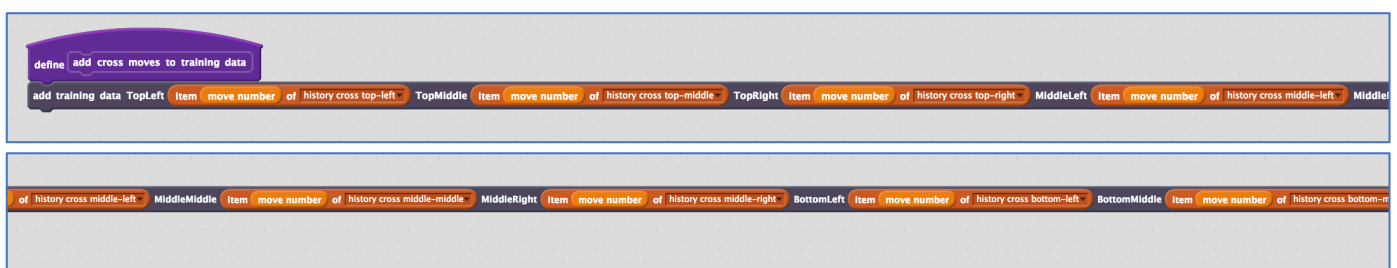
33. Copy the “item - move number” blocks into the spaces in the “add training data” block

The blocks are all ready for you underneath, and they are in the right order.

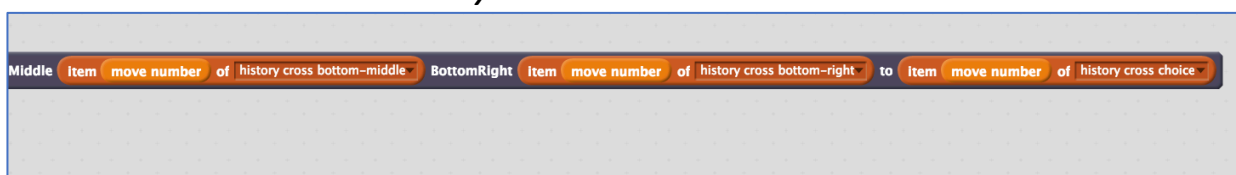
You just need to drag them into the spaces



It should look like this when you’re finished. The names should match up: TopLeft goes with “history cross top-left”, TopMiddle goes with “history cross top-middle”, etc.

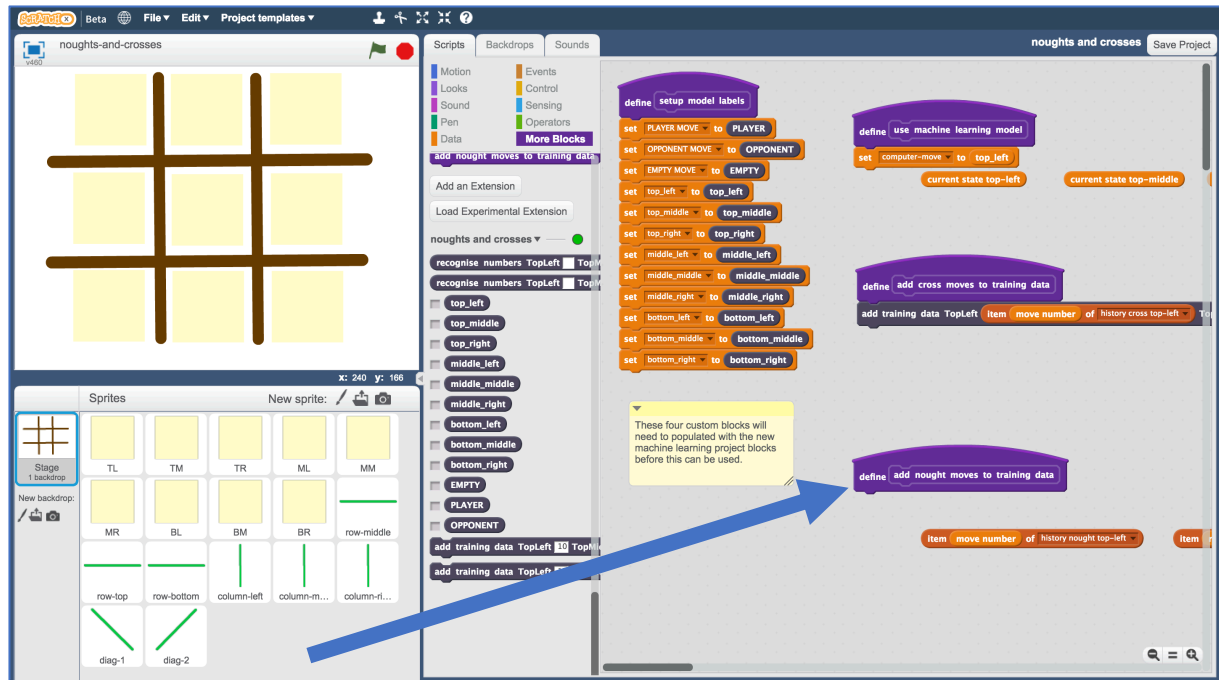


The last box has the “history cross choice” block.



This is a complicated step – so take your time and do it carefully!

34. Find the “add nought moves to training data” script

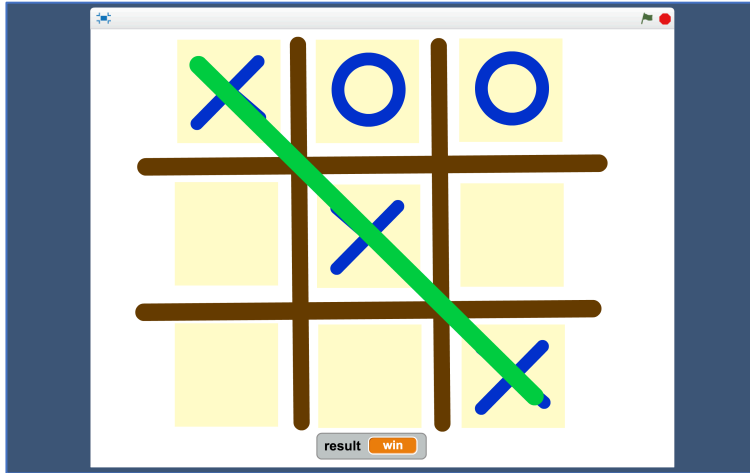


35. Add a “add training data” block and copy the move blocks into it in the same way as you did with the cross moves Remember to make sure that names in each block match the label to the left on the blue block. e.g. TopLeft goes with “history nought top-left”.

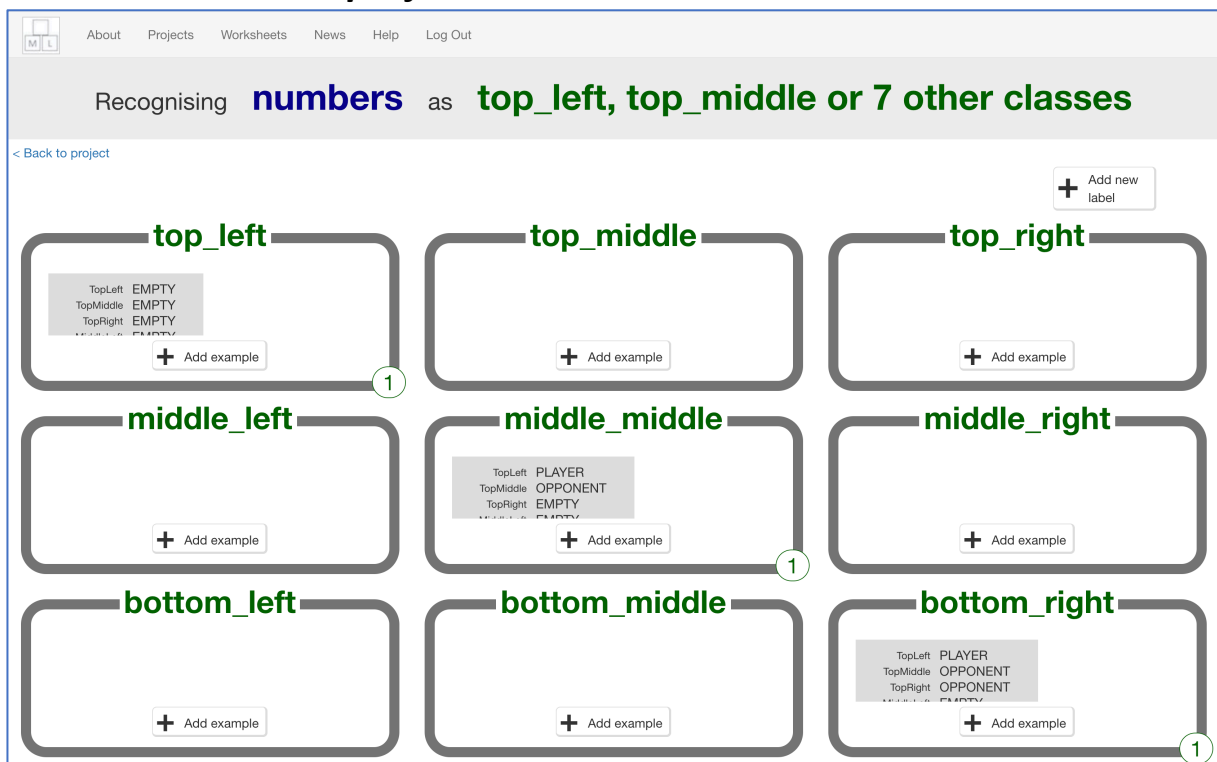


36. Save your project Click **File -> Save Project**

- 37.** Play one game against the computer
*Click on the **Green Flag** as you did before.*
Play in full-screen mode so you don't accidentally move sprites.



- 38.** Go back to the training page
Leaving the Scratch window open, go back to the training tool window.
Click the “< Back to project” link and then click “Train”



- 39.** Look at your training so far
Each item is a move made by the winning player.
The details in each item describe the state of the board at the time the winning player made that move.

- 40.** Go back to Scratch window
- 41.** Play several more games – you want lots of training examples to teach the computer how to play the game
When you think you've got examples of lots of different types of games, go back to the training tool window
- 42.** Click the “< **Back to project**” link. Click the “**Learn & Test**” button.
- 43.** If there is a “**Train new machine learning model**” button you can go to step **47**. Otherwise, carry on to step **44**.
- 44.** You don't have enough examples to train the computer yet
For the computer to know when it is a good idea to choose any space on the board, you need at least 5 examples of where you chose that space and ended up winning.
This page shows you how many examples you have so far. Look to see which one(s) you need more examples for.

Machine learning models

[< Back to project](#)

What have you done?

You have collected examples of numbers for a computer to use to recognise when numbers are top_left, top_middle or 7 other classes.

You've collected:

- 2 examples of top_left,
- 1 example of top_middle,
- 1 example of top_right,
- 1 example of middle_left,
- 3 examples of middle_middle,
- 0 examples of middle_right,
- 2 examples of bottom_left,
- 1 example of bottom_middle,
- 1 example of bottom_right

What's next?

Keep going!

Go back to the [Train](#) page and collect more examples for each of the labels.

The more you can get, the better it should learn, but you need at least five examples of each as an absolute minimum.

- 45.** Leave the “**Learn & Test**” window open.
Go back to the **Scratch** window.

46. Play more games.

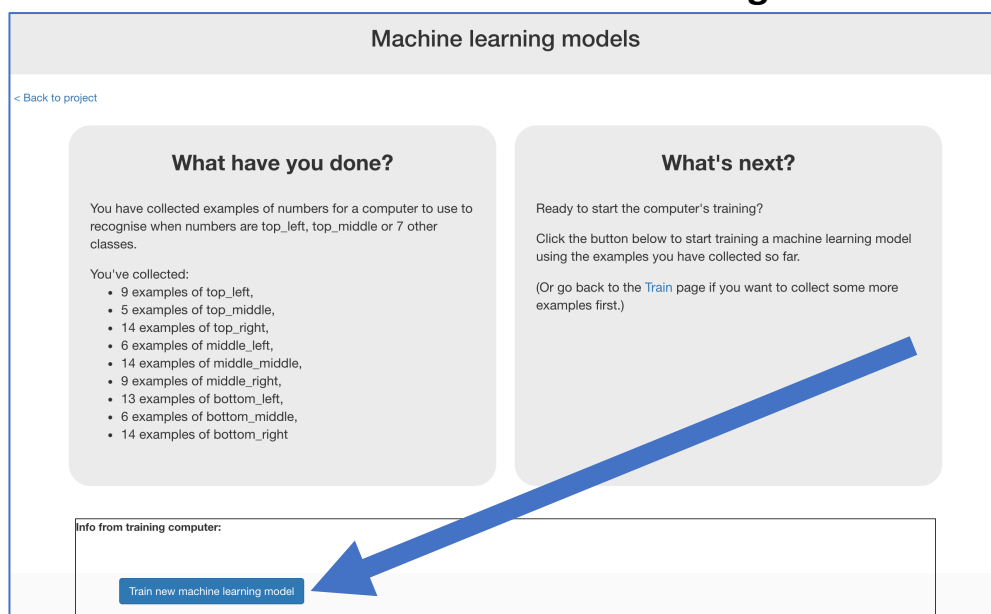
Try starting from a different position to get a variety of examples.

Try starting from positions that you know you need more examples of.

47. When you think you've got at least 5 examples of each space, go back to the **"Learn & Test"** window and **refresh** the page.

If there is still no "Train new machine learning model" button, you need to go back to step 44 and try again.

48. Click the **"Train new machine learning model"** button



What have you done so far?

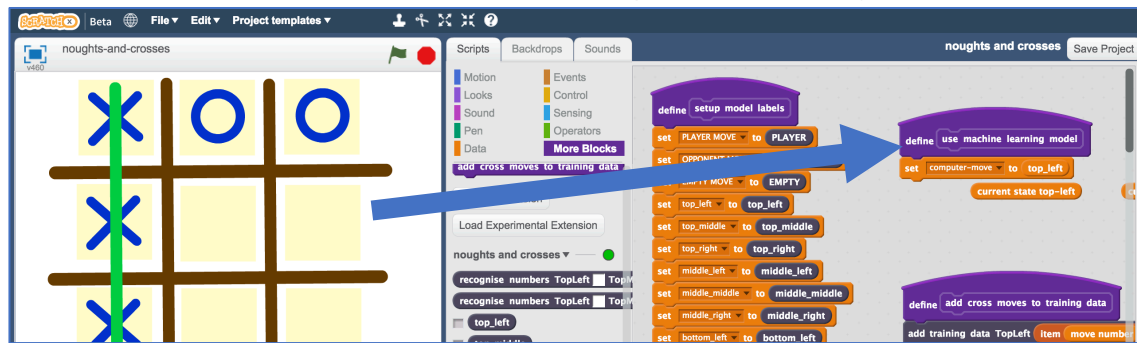
You're teaching a computer to play noughts and crosses.

So far, you've updated a Scratch noughts and crosses game so that it can collect examples of how you play and add them to a set of examples. And you've used those examples to train a machine learning "model".

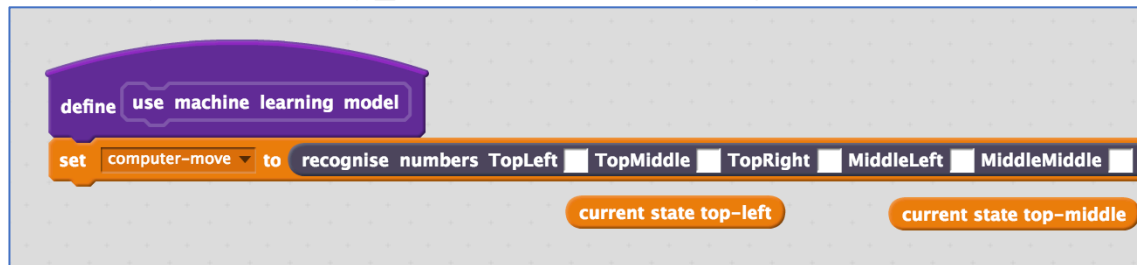
The next step is to use that model to let the computer decide what move to make – instead of just going for the next empty space every time.

49. Go back to the Scratch window

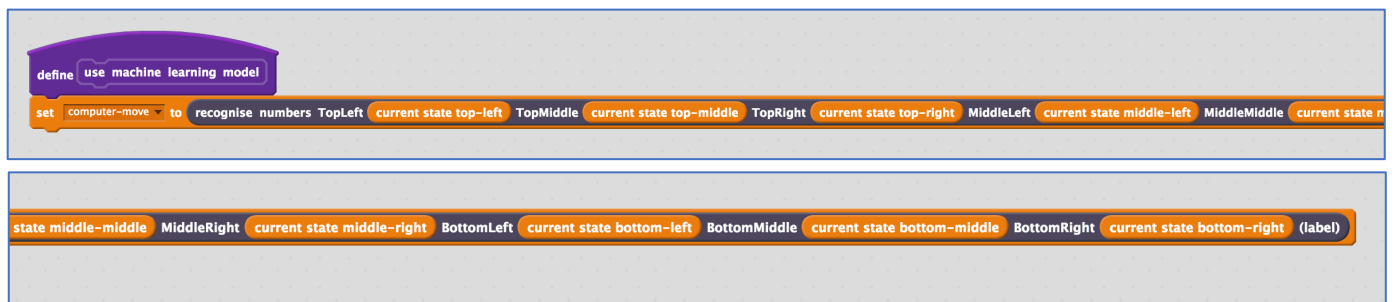
50. Find the “use machine learning model” script



51. Replace the top_left block with “recognise numbers ... label”



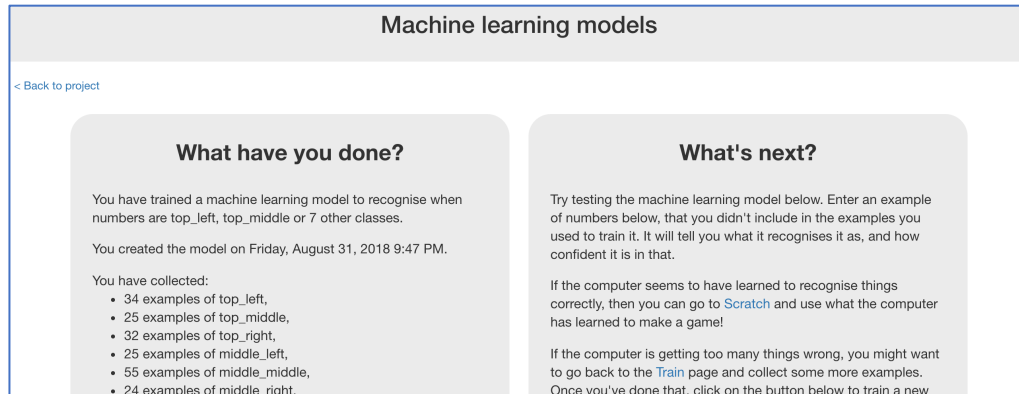
52. Copy the orange blocks underneath into the spaces in the “recognise numbers” block, similar to what you did before



53. Save your project
Click **File -> Save Project**

54. Play against the computer by clicking on the **Green Flag**
*Use full-screen to avoid moving sprites accidentally.
Avoid playing the same game over and over again. Choose different spaces to give the computer a variety of examples of how to play.*

- 55.** When it feels like you're playing the same games over and over, go back to the "Learn & Test" screen, and use the new training examples *Click the "Train a new machine learning model" button again*



- 56.** Go back to the Scratch window and play more games *Once you're struggling to come up with new games, click the "Train new machine learning model" button again. Repeat this a few times until your model starts getting good.*

What have you done?

You've trained a computer to play noughts and crosses.

You didn't have to describe the rules to the computer.

You didn't tell it that it should try to get three noughts in a row.

You didn't describe the difference between rows, columns or diagonals. (The rules are in the Scratch game, but that doesn't count – that wasn't used in the machine learning model).

Instead, you showed it how you play, by collecting examples of decisions that you made when you win.

When it makes decisions that leads to it winning, this is added to its training data, so it can be even more confident in that approach in future.

This is called "reinforcement learning" because when it does something good you are "reinforcing" this by rewarding it.

Tips

Don't be kind!

You might be tempted to go easy on the computer when you're playing against it, particularly when it's just starting to learn and is playing very badly.

For example, you might have two crosses-in-a-row next to a blank space and could win. But instead, you might feel sorry for it doing badly and put a cross somewhere else instead to give it a chance.

Don't.

It is learning from the way that you play. If you don't complete a three-in-a-row when you can, you will be teaching it that it should do that.

If you want it to get better quickly, **play as well as you can.**

Mix things up with your examples

Try to come up with lots of different types of examples.

For example, start from a different position on the board on every turn.

Did you know?

People have been learning about machine learning by training a computer to play noughts and crosses for decades!

One famous example was **Donald Michie** – a British artificial intelligence researcher. During World War II, Michie worked at Bletchley Park as a code breaker.

In 1960, he developed “**MENACE**” – the Machine Educable Noughts And Crosses Engine. This was one of the first programs able to learn how to play noughts and crosses perfectly.

As he didn’t have a computer he could use, Michie built MENACE using 304 matchboxes and coloured glass beads.

Each matchbox represented a possible state of the board – like the examples that you’ve been collecting in your training data.

He put beads in the matchboxes to show how often a choice led to a win – the number of beads in the matchbox was like the number of times an example shows up in one of the buckets you created for your training data.

