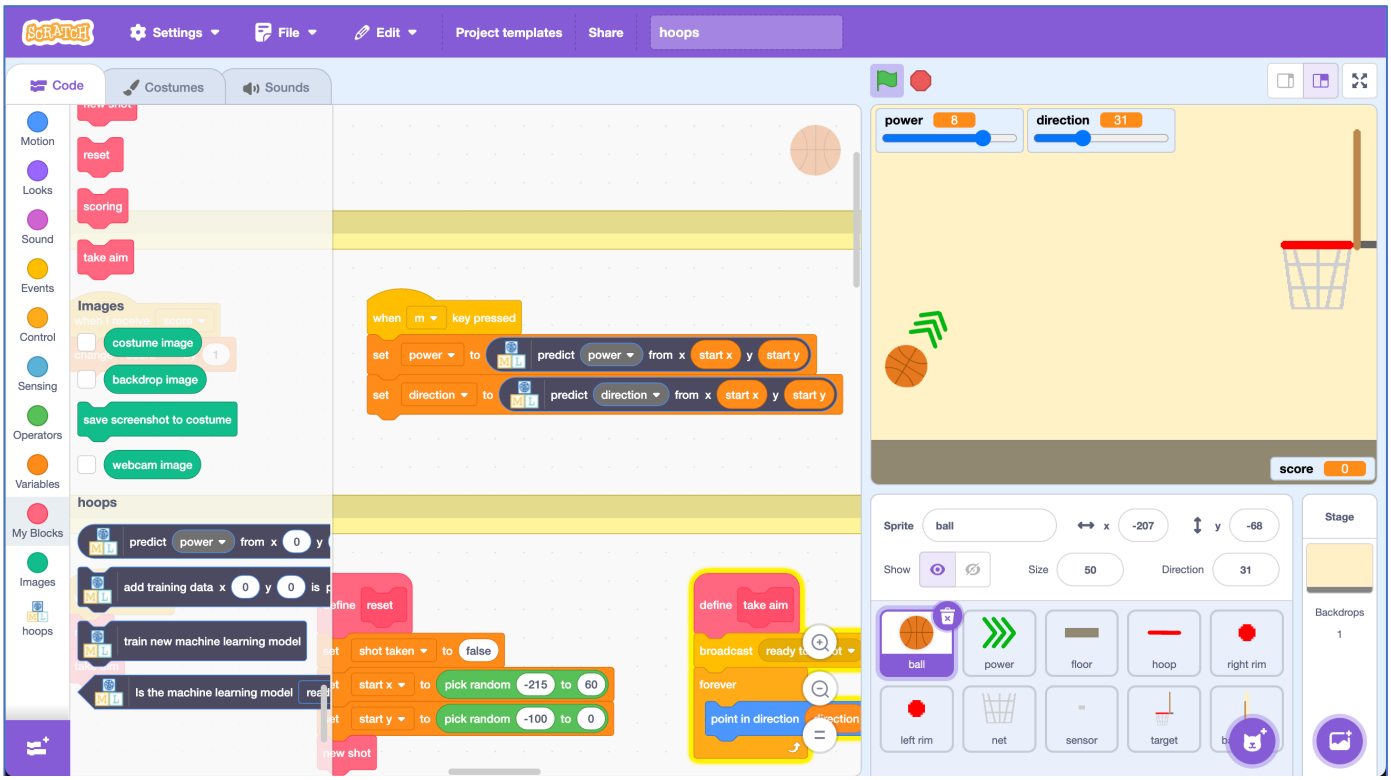
Hoops

In this project you will train a computer to play a simple arcade game.

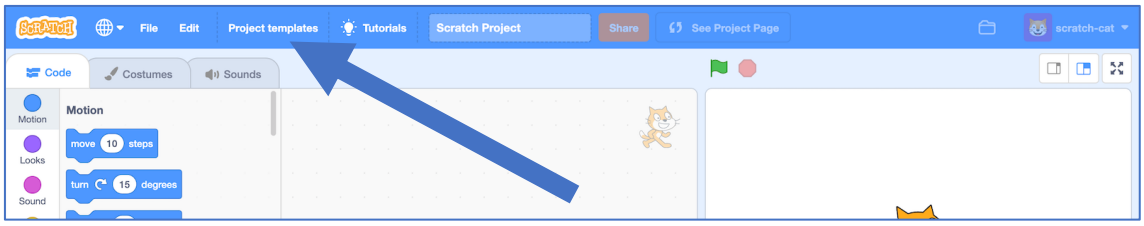
The game is based on shooting basketballs at a hoop.

You will teach the computer to be able to play this game by collecting examples of shots that score, so that it can learn to make predictions about the shot it should take.



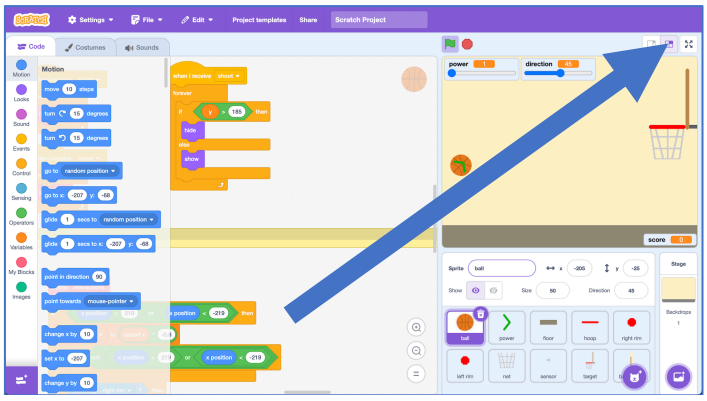
1. Go to https://machinelearningforkids.co.uk/scratch

2. Click on "**Project templates**"



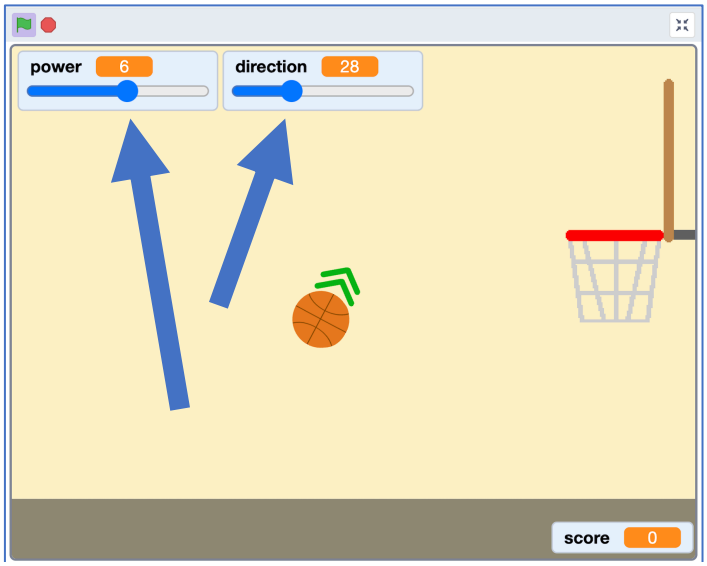
3. Click on the "**Hoops**" template

4. Click on the **full-screen** button



5. Click the **Green Flag** to move the ball to a new random location

6. Use the **power** and **direction** variable sliders to decide how you want to shoot the ball from this location



7. When you are ready, press **S** to **s**hoot

8. To improve on your shot, press **R** to **r**eset
This will put the ball back at the starting location, and let you change your decision about the power and direction

9. When you think you've got it right, press **S** to **s**hoot again
Did you score? If not, go to step 8 and try again until you get it in the hoop

10. Click on the **Green Flag** to try again with the ball in a new location
Remember:

Green Flag	Starts again, with the ball in a new location
S	**Shoot** the ball
R	**Reset** with the ball in the same place, so you can try again

What have you done so far?

You played a game in Scratch. Each time you play, the ball appears at a random location. The aim of the game is to shoot the ball into the hoop.

In this project, you are going to get the computer to predict what angle it should shoot at, and the power to use, based on the location of the ball.

You could do this by writing code to calculate the correct angle and power to launch at, based on the location. (If you have time, give this a try to compare!)

But, for this project, you're going to train the computer so that it learns for itself how to shoot hoops.

You'll collect examples of the game being played and use that to train a machine learning "model" that can predict the correct angle to fire the ball at.

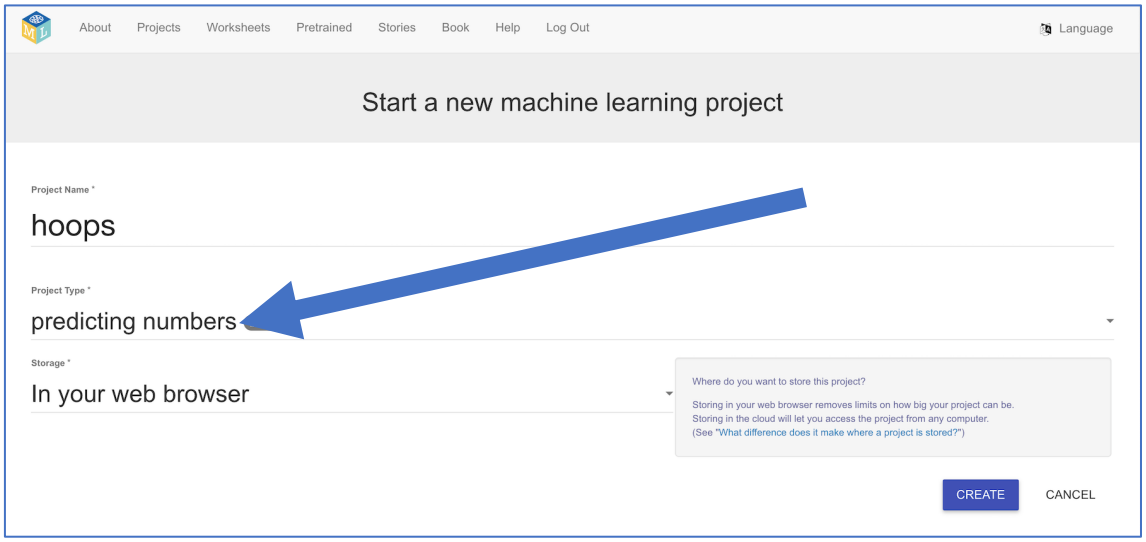
11. Go to https://machinelearningforkids.co.uk in a web browser

12. Click on "**Get started**"

13. Click on "**Try it now**"

14. Click the **"+ Add a new project"** button.

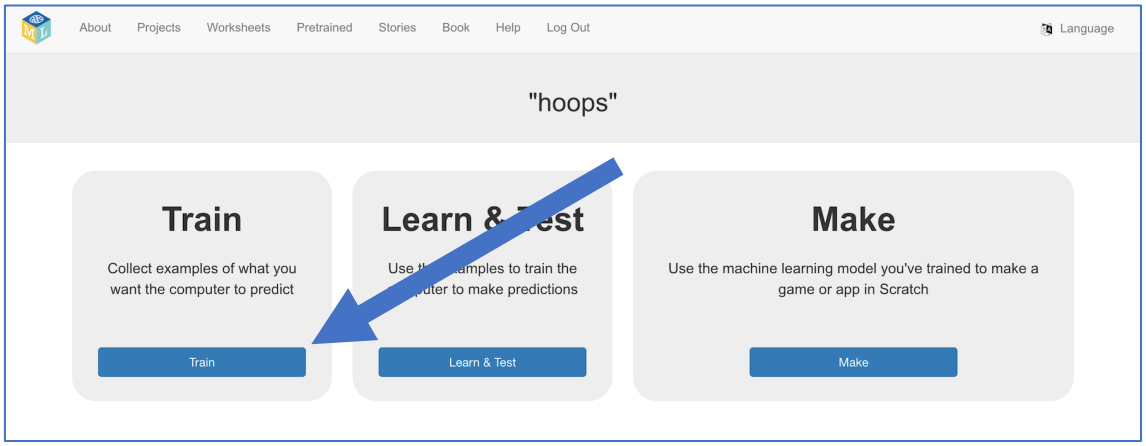
15. Name your project "hoops". Set it to learn to **predict numbers**.

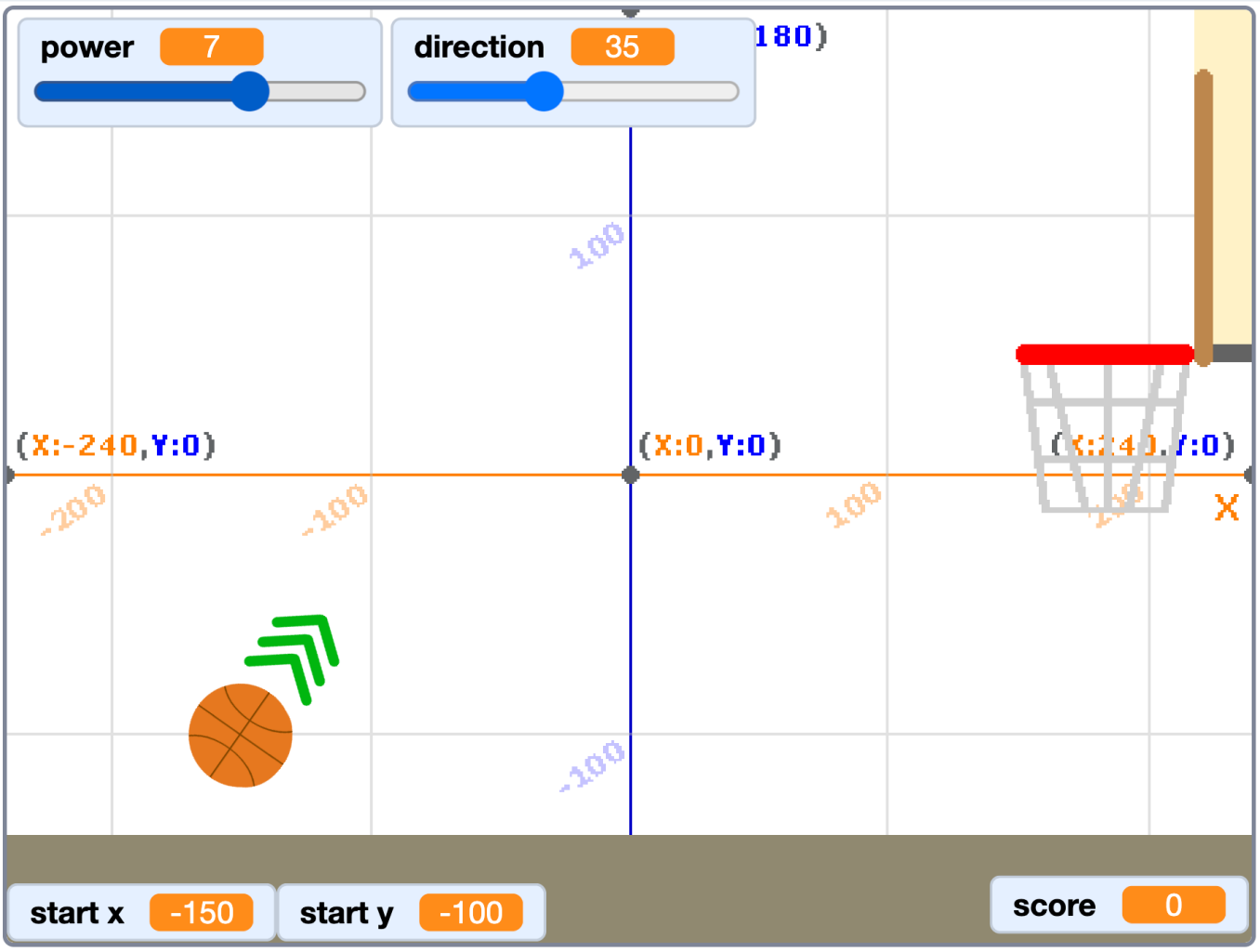


16. Click on "**Create**"

17. "**hoops**" will be added to your list of projects. Click on it.

18. Click the "**Train**" button





Values the computer will use to make a prediction:

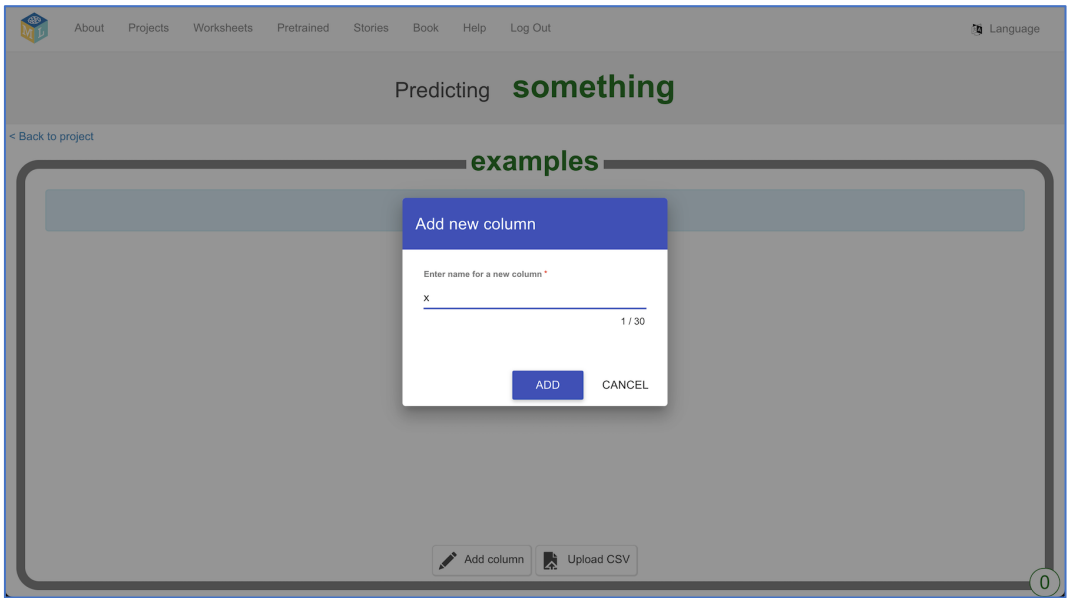
Name	What is it?	A positive number means...	A negative number means...	Example in screenshot
`start x`	x-coordinate of the starting location of the ball	The ball is on the **right side** of the stage	The ball is on the **left side** of the stage	-150
`start y`	y-coordinate of the starting location of the ball	The ball is at the **top** of the stage	The ball is at the **bottom** of the stage	-100

Values the computer will learn to predict:

Name	What is it?			Example in screenshot
`power`	speed to fire the ball	1 means the ball will be very **slow**	10 means the ball will be very **fast**	7
`direction`	angle to fire the ball	0 means the ball will go directly **up**	90 means the ball will go directly to the **right**	35

19. Click on **“Add column”**

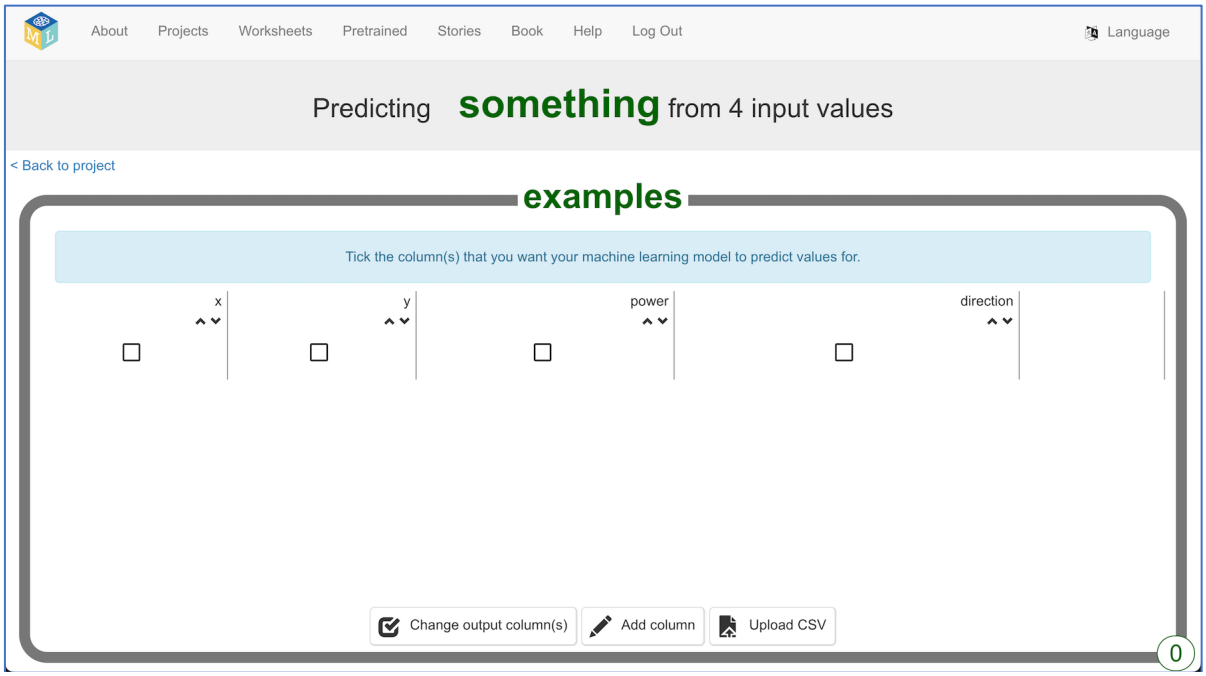
20. Type “x” and click “**Add**”



21. Click on “**Add column**”, type “y”, and click “**Add**”

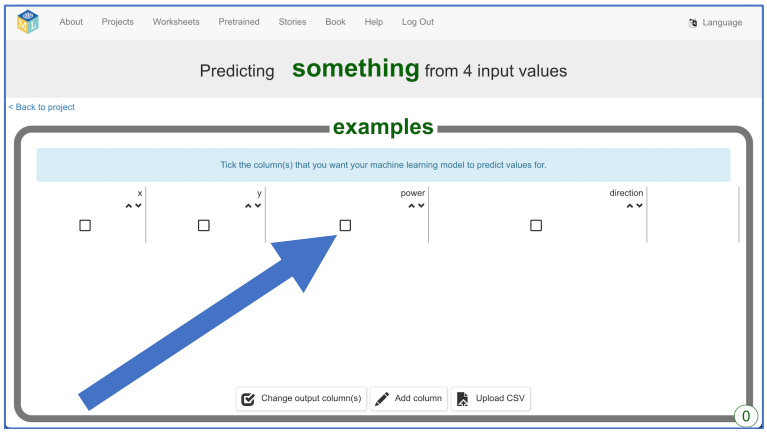
22. Click on “**Add column**”, type “`power`”, and click “**Add**”

23. Click on “**Add column**”, type “`direction`”, and click “**Add**”



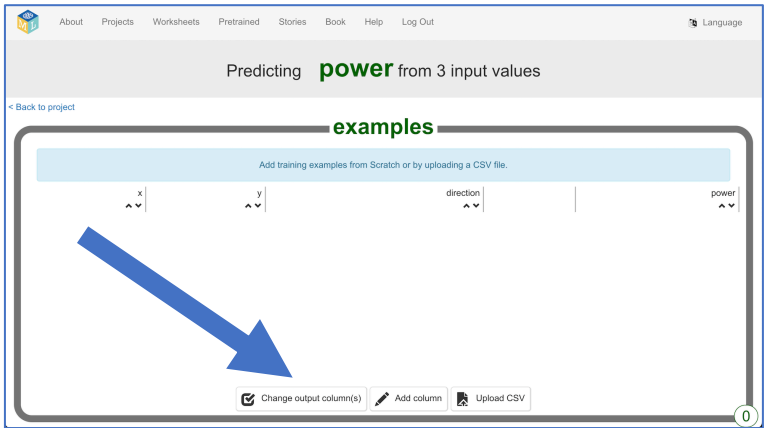
24. Click the tick box under “power”

This is to choose that the “power” value is one of the values that you want the computer to learn to predict



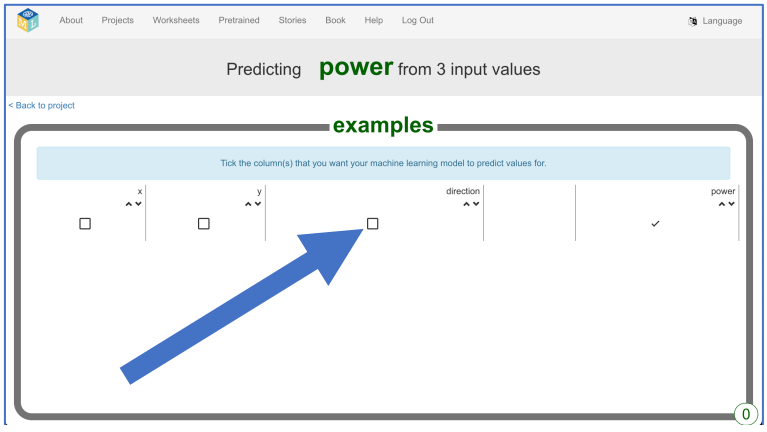
25. Click the **Change output columns** button

This is to let you add another value to the ones you want the computer to learn to predict



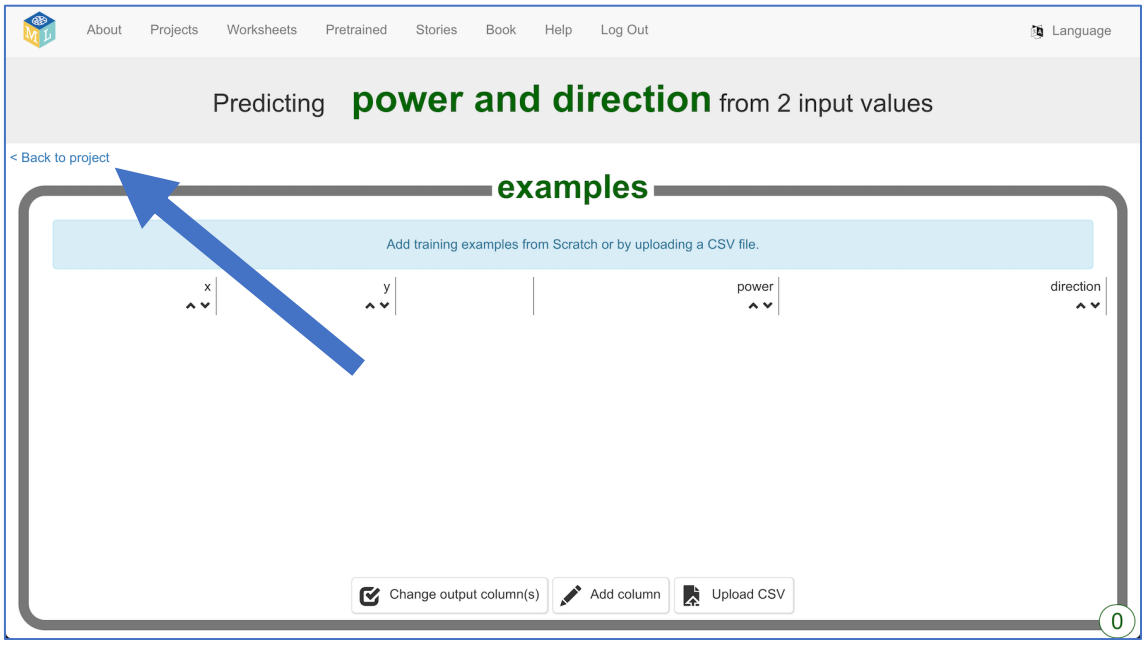
26. Click the tick box under “direction”

This is to choose that the “direction” value is the other value that you want the computer to learn to predict



27. Click “**Back to project**”

*Check your project looks like the screenshot before you continue. The title should say that your project will be predicting **power and direction** from 2 input values.*



28. Click the “**Make**” button

29. Click on “**Scratch 3**”

30. Click the “**straight into Scratch**” button

The page will warn you that you don’t have a machine learning model yet. That is okay.

You are going to use Scratch to collect the training examples you will use to train a model.

31. In the Scratch window that opens, click on “**Project templates**”

32. Open the “**Hoops**” template again

33. Find some empty space in the "**ball**" sprite and add this code

This means every time you score a hoop, the decision you made for power and decision will be added to your training data.
The computer will learn from all of the good shots you take.
Make sure you use the correct variables as shown below.

34. Click on the **full-screen** button

35. Click on the **Green Flag**

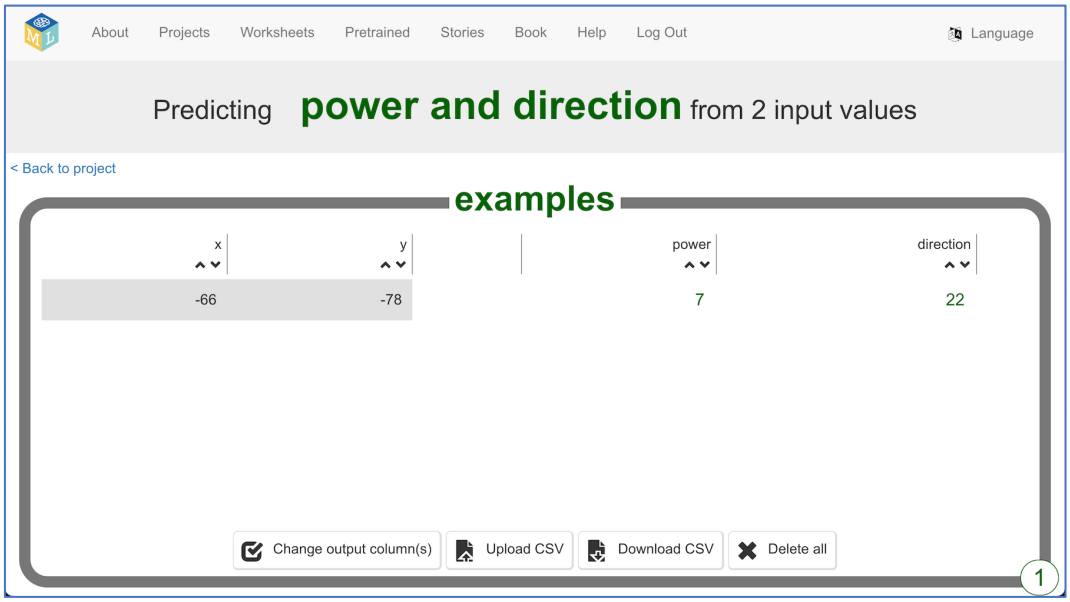
36. Play the game until you score a hoop

Remember:

Green Flag	Starts again, with the ball in a new location
S	**Shoot** the ball
R	**Reset** with the ball in the same place, so you can try again

37. In the training window, go back to the "**Train**" page

You should see the first example has been added to your training data

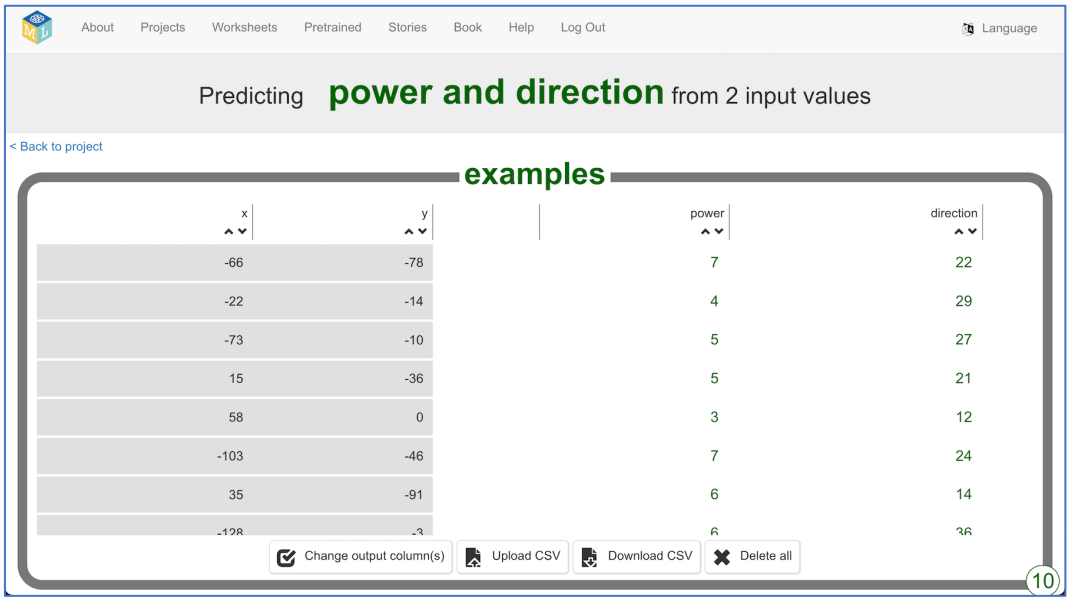


38. Go back to the Scratch window, and click on the **Green Flag** again

39. Repeat this to collect more training examples

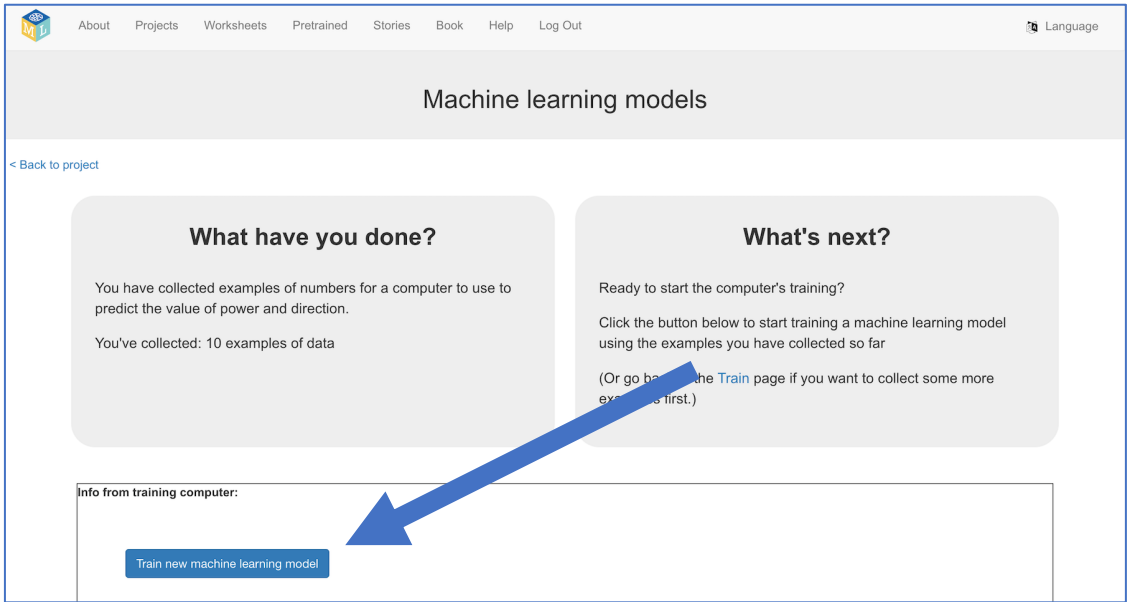
*The more examples you collect, the better your machine learning model should perform. Try to collect **at least ten** examples.*

40. Go back to the training window and refresh the Train page to review the examples you have collected



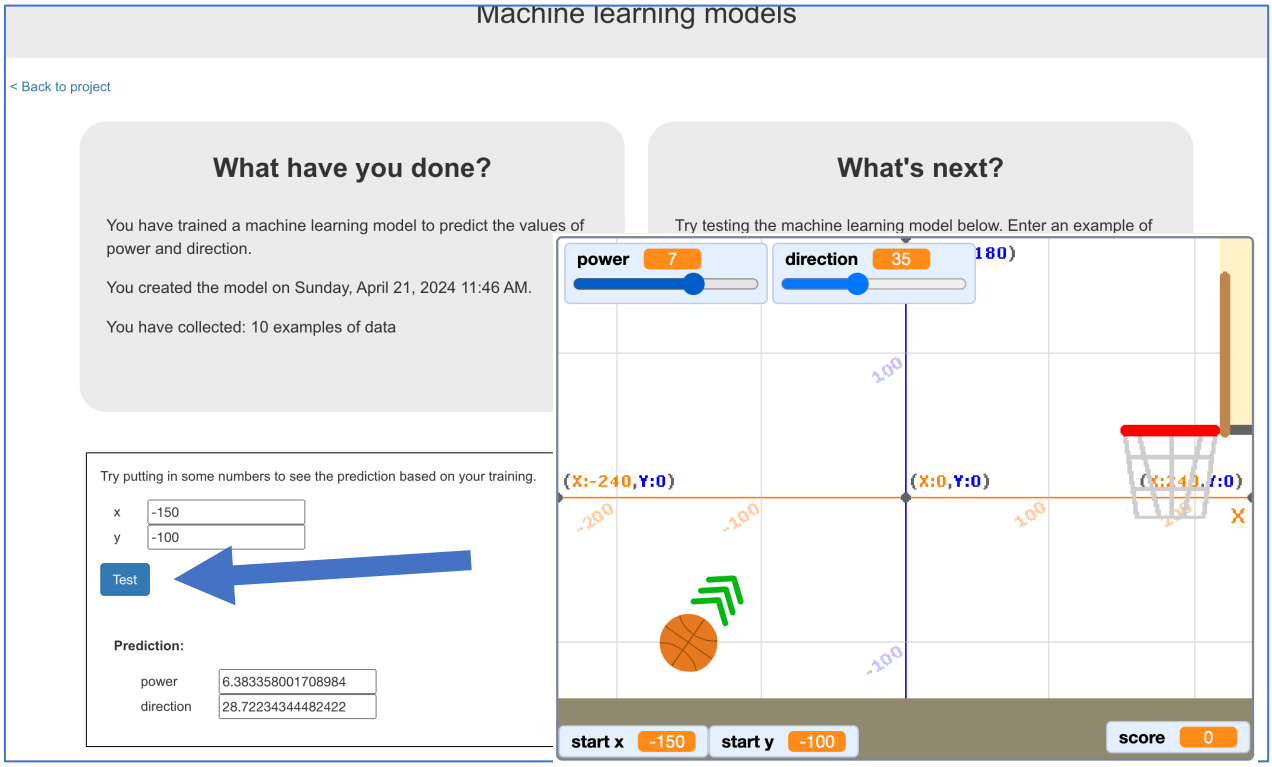
41. Train a machine learning model using your examples

*Click on "**Back to project**", then click the "**Learn & Test**" button. Then click on the "**Train new machine learning model**" button*



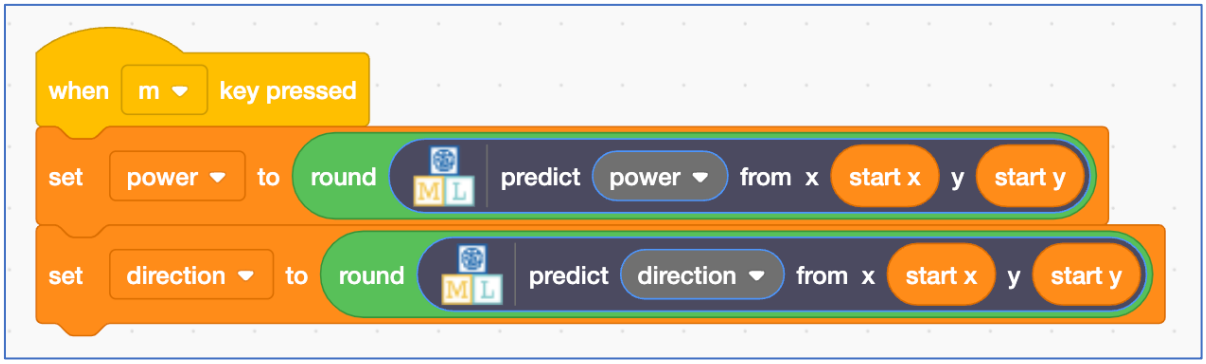
42. Try testing your model to see what it predicts

This is the prediction my model made for the values in my earlier diagram
Do you think your model is making good predictions?



43. In the Scratch window, find some empty space on the ball sprite to add this code.

You will need to exit full-screen mode to modify your code.
This means you will be able to press **M** *to get a recommendation from your machine learning model.*



44. Click on the **full-screen** button to go back into full-screen mode

45. Click on the **Green Flag**

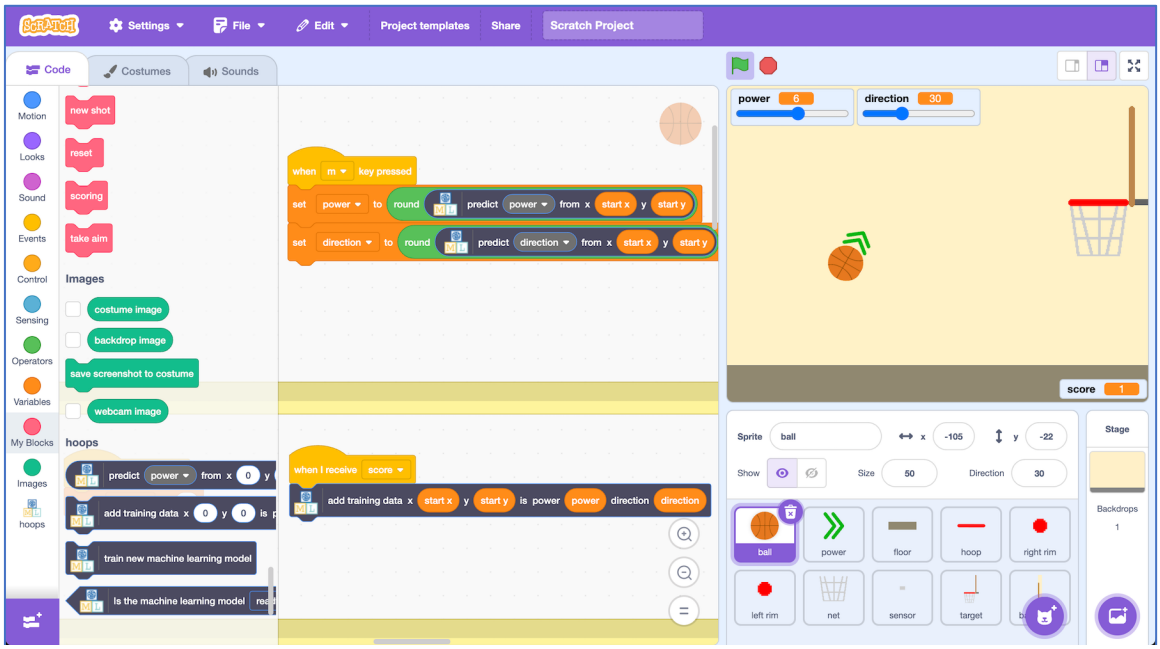
46. Try playing the game with help from your machine learning model

Remember:

Green Flag	Starts again, with the ball in a new location
M	Get the **Model** to choose power and direction values for you
S	**Shoot** the ball
R	**Reset** with the ball in the same place, so you can try again

If the machine learning model doesn't score a hoop,
*press **R**, change the values to what you think is right, then*
*press **S** to shoot again.*

How good is your machine learning model at shooting hoops?



What have you done so far?

You've trained a computer to play a game. Instead of working out the equation to fire the ball, you did it by collecting examples. These examples were used to train a machine learning model.

The computer learned from patterns in the examples. It used these to make predictions. The more examples it had to learn from, the better it will probably be.

47. You can go back to the **Learn & Test** page and click the **Train new machine learning model** button again

You still have the "add training data" blocks in your code, so you were collecting more training examples while you were testing your project. Training a new model now will hopefully be better at playing the game because it has more examples to learn from.

48. Go back to the Scratch window

49. Play again with help from your machine learning model

Have the recommendations improved?

Is this a good use of machine learning?

We use machine learning when we want computers to do things that would be complicated for us to write instructions for it to follow.

We avoid machine learning if the time it takes to collect training examples would be longer than just writing the instructions for how to do the task.

Compare the effort to collect the training examples to train the computer to play this game, with the effort it would've taken you to calculate the angle to fire at. Do you think this game is a good use of machine learning?

In this project, you were always in control of the shot that was taken. The machine learning model didn't replace you, but it was available to help you with recommendations. Do you think it's safer to use machine learning like this?

Ideas and Extensions

Now that you've finished, why not give one of these ideas a try?

Or come up with one of your own?

Import examples from other players

Training examples can be downloaded to a CSV file.

Sharing training examples with others can help a machine learning model to learn from the experiences of many players.

You can find an example at:
https://gist.github.com/dalelane/dffdf4bb3a397f016f3912ddcf573045
(Click on **Raw** to get the CSV file to save)