

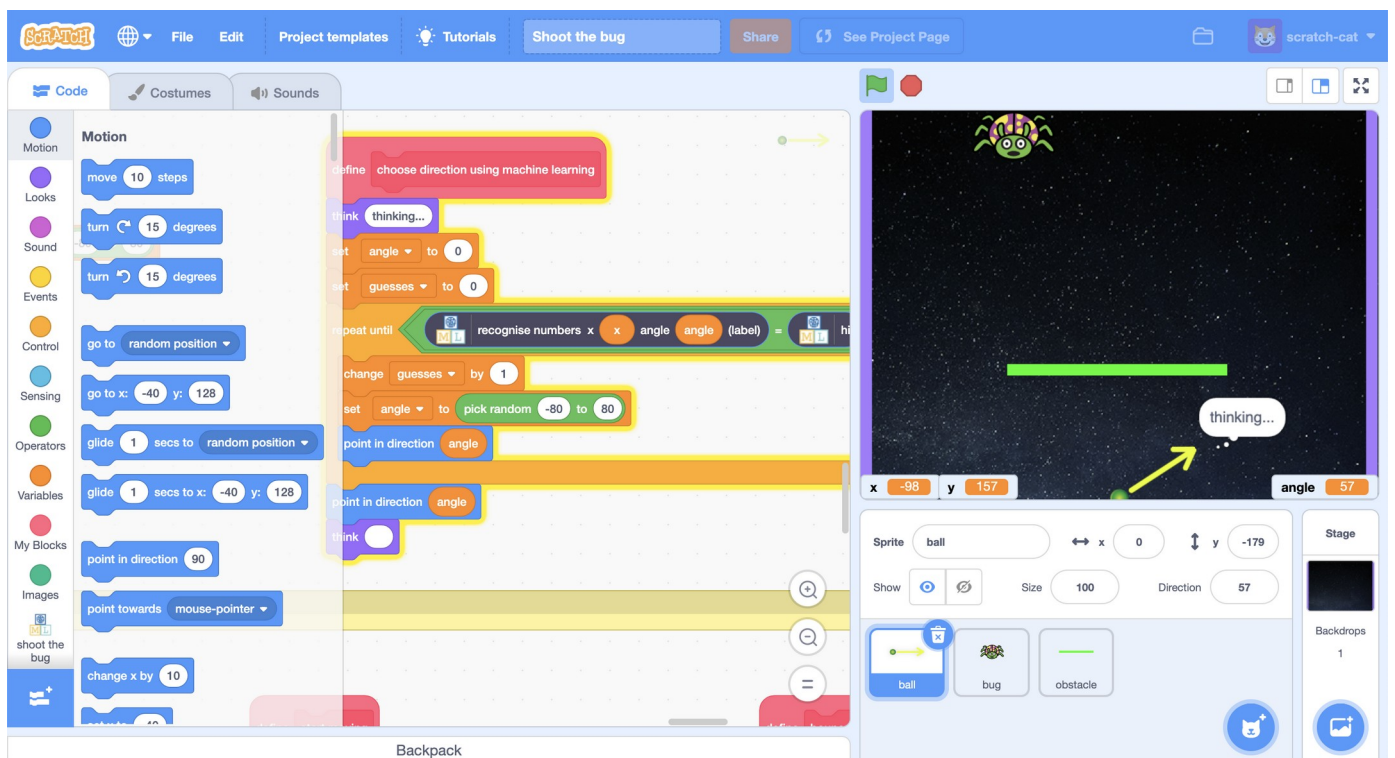


Dispara al bicho

En este proyecto, entrenarás a un ordenador para que juegue un juego de arcade simple.

El juego se basa en disparar bolas a un objetivo. No puedes apuntar al objetivo directamente porque hay una pared en el camino, por lo que debes hacer rebotar la pelota en una pared para hacerlo.

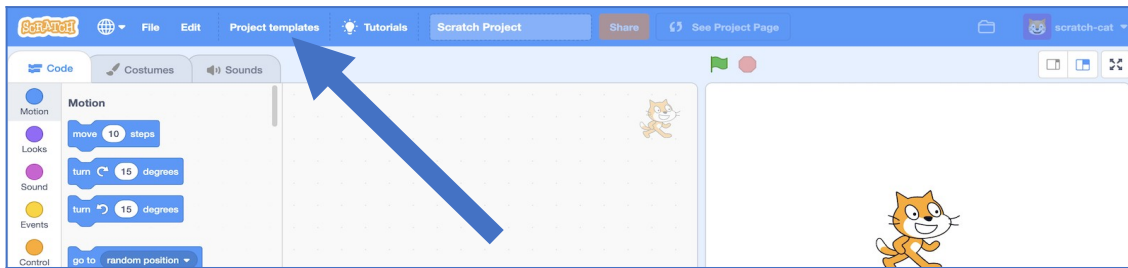
Enseñarás al ordenador a poder jugar este juego recolectando ejemplos de tiros que aciertan y fallan, para que pueda aprender a hacer predicciones sobre los tiros que puede tomar.



Esta hoja de proyecto está bajo licencia de Creative Commons Attribution Non-Commercial Share-Alike License
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

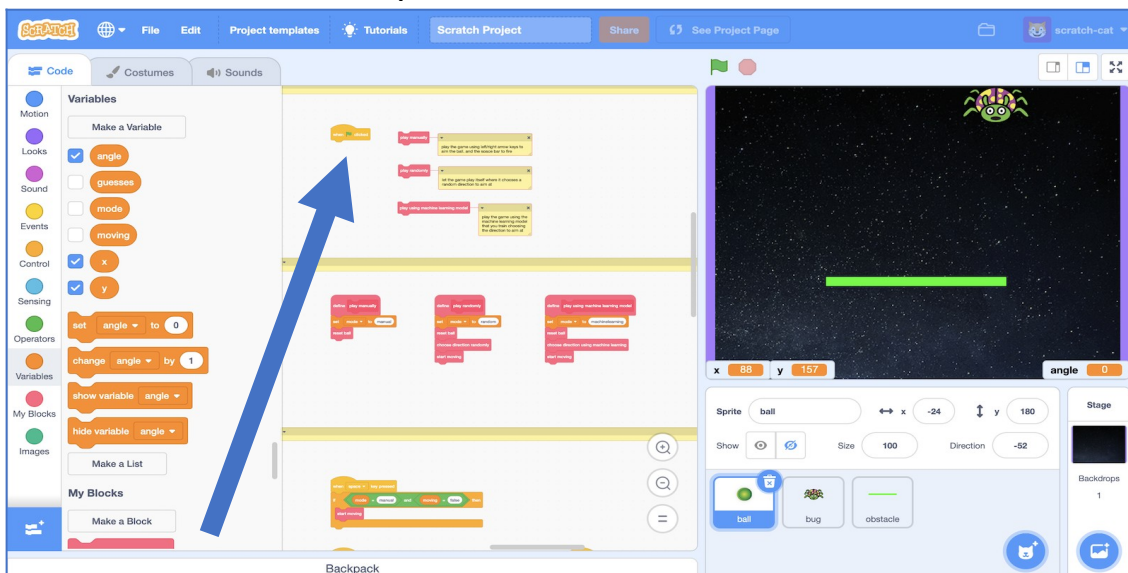
1. Ve a <https://scratch.machinelearningforkids.co.uk>

2. Haz clic en “Project templates”

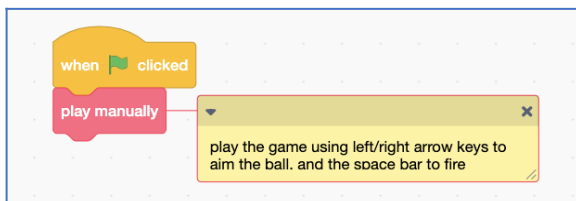


3. Haz clic en la plantilla “Shoot the bug”

4. Encuentra el bloque **Al hacer clic en la bandera verde**



5. Adjunta el bloque “play manually” al bloque de la bandera verde.



6. Haz clic en la bandera verde ¡e intenta disparar al bicho!
Usa las teclas de flecha para apuntar, luego presiona la barra espaciadora cuando estés listo.
Intenta jugar varias veces para acostumbrarte a cómo funciona el juego.

¿Qué has hecho hasta ahora?

Has jugado un juego en Scratch. Cada vez que juegas, el insecto se mueve a una ubicación aleatoria. El objetivo del juego es dispararle una pelota al insecto.

Hay un obstáculo en el camino, por lo que debe rebotar en una pared lateral para sortear el obstáculo.

Las coordenadas x,y del bicho se muestran en la pantalla del juego en la esquina inferior izquierda. El ángulo en el que lanzas la pelota se muestra en la esquina inferior derecha.

En este proyecto, harás que el ordenador decida en qué ángulo debe disparar, según la ubicación del bicho.

Puede hacer esto escribiendo código para calcular el ángulo correcto para lanzar, según la ubicación. (¡Si tienes tiempo, prueba esto para comparar!)

Pero, para este proyecto, vas a entrenar al ordenador para que aprenda por sí misma cómo dispararle al bicho.

Recopilarás ejemplos del juego que se está jugando y lo usarás para entrenar un "modelo" de aprendizaje automático que puede predecir si un tiro en un cierto ángulo acertará o fallará.

7. Ve a <https://machinelearningforkids.co.uk/>

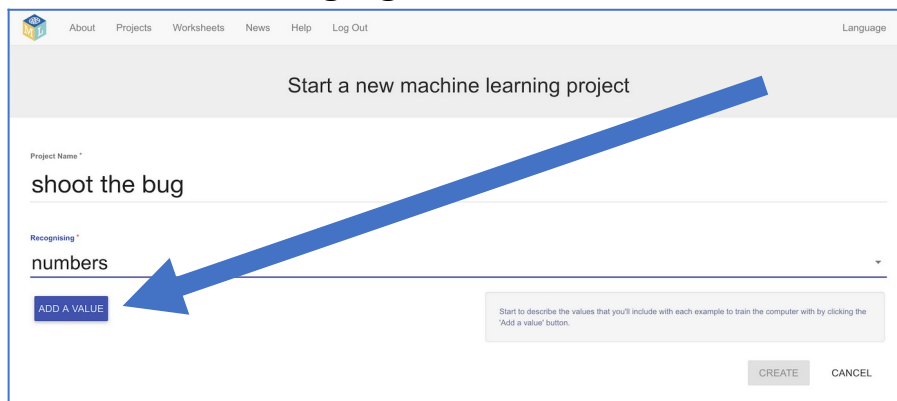
8. Haz clic en **Empezar**

9. Haz clic en **Prueba ahora**

10. Haz clic en el botón **Añadir un nuevo proyecto.**

11. Nombra a tu proyecto “Dispara al bicho” y lo estableces para aprender a reconocer **números**.

12. Haz clic en **Agregar un valor**



Start a new machine learning project

Project Name *

shoot the bug

Recognising *

numbers

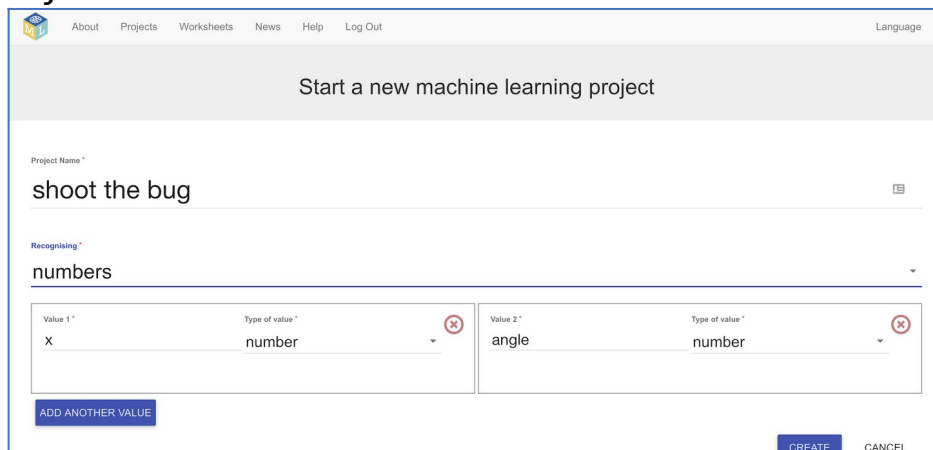
ADD A VALUE

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. Crea un valor **numérico** llamado “x”, luego haz clic en **Añadir otro valor**.

14. Crea un valor **numérico** llamado “ángulo”,
El formulario debería verse así ahora



Start a new machine learning project

Project Name *

shoot the bug

Recognising *

numbers

Value 1 *

x

Type of value *

number

Value 2 *

angle

Type of value *

number

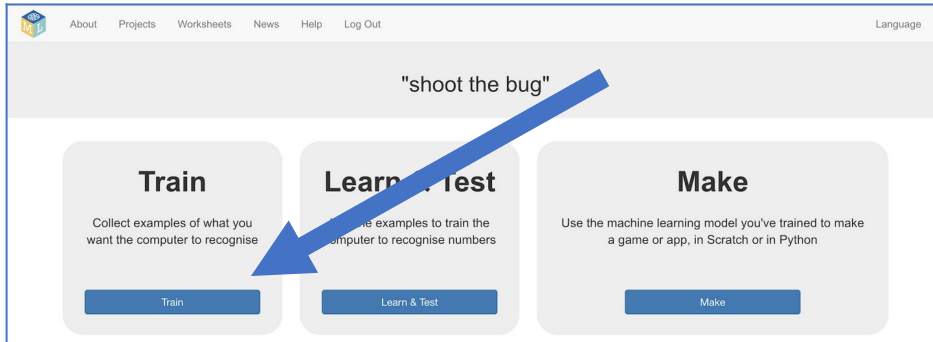
ADD ANOTHER VALUE

CREATE CANCEL

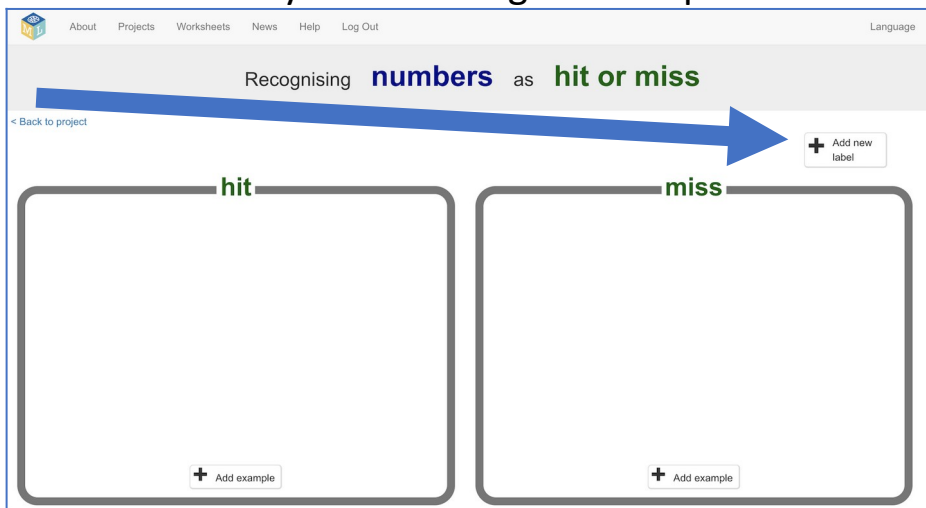
15. Haz clic en el botón **Crear**

16. Ahora deberías ver **Dispara al bicho** en la lista de tus proyectos. Haz clic en él.

17. Debe preparar los tipos de predicción que deseas que realice el ordenador. Haz clic en el botón **Entrenar**



18. Haz clic en **Añadir etiqueta** y llámala “golpe” Hazlo de nuevo y crea una segunda etiqueta llamada “error”



19. Haz clic arriba a la izquierda, en **Volver al proyecto**

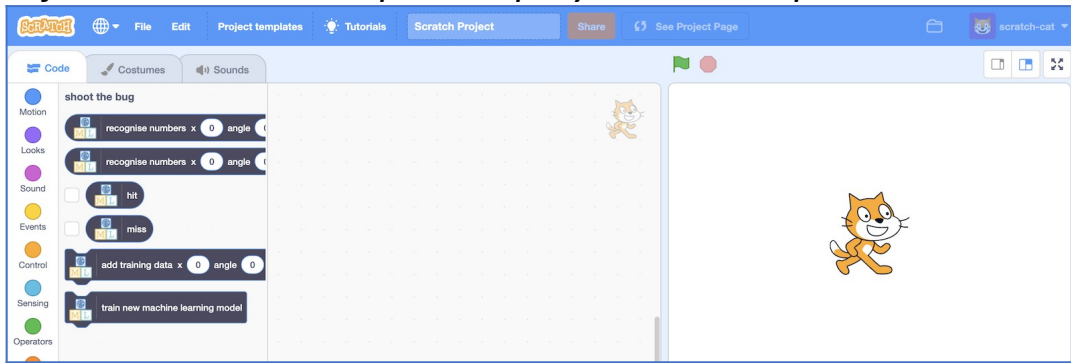
20. Haz clic en el botón **Crear**.

21. Haz clic en el botón **Scratch 3**

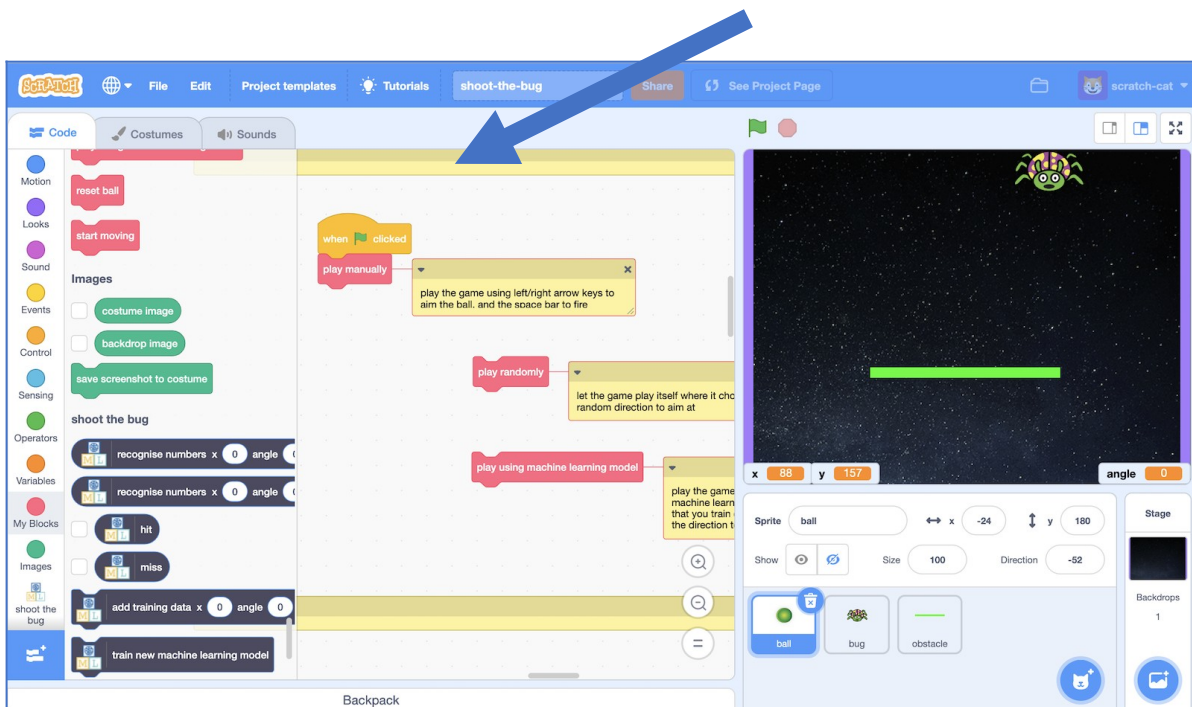
22. Haz clic en **Abrir en Scratch**

La página te advertirá que aún no has entrenado a un modelo, pero está bien, ya que primero usarás Scratch para recopilar ejemplos de

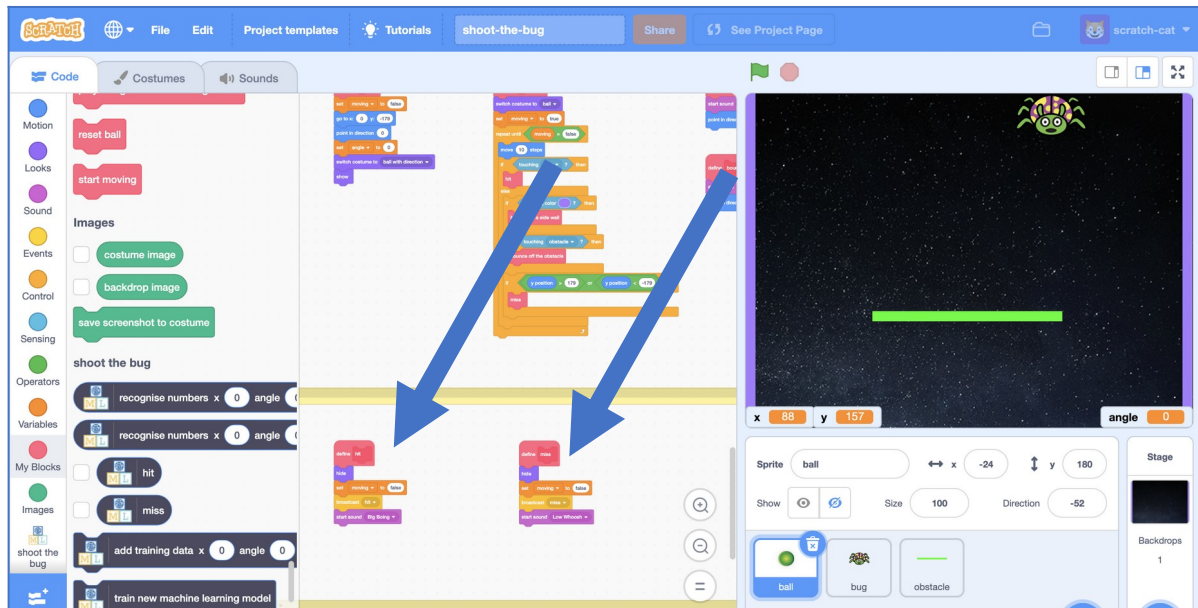
entrenamiento. Se abrirá Scratch, con bloques adicionales agregados a la caja de herramientas para tu proyecto de "disparar al bicho"



23. Haz clic en **“Project templates”** (Plantillas de proyectos).
24. Abre de nuevo la plantilla **“Dispara al bicho”**.
25. Conecta de nuevo **“play manually”** al bloque de **“al hacer clic en la bandera verde”**, como has hecho antes



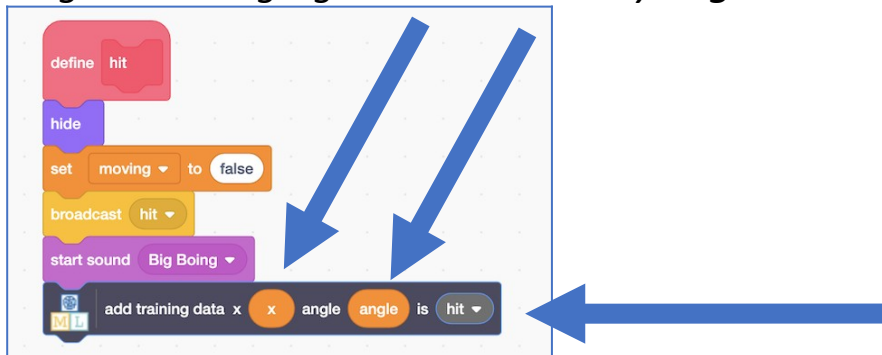
26. Encuentra la programación para “golpe” y “error”



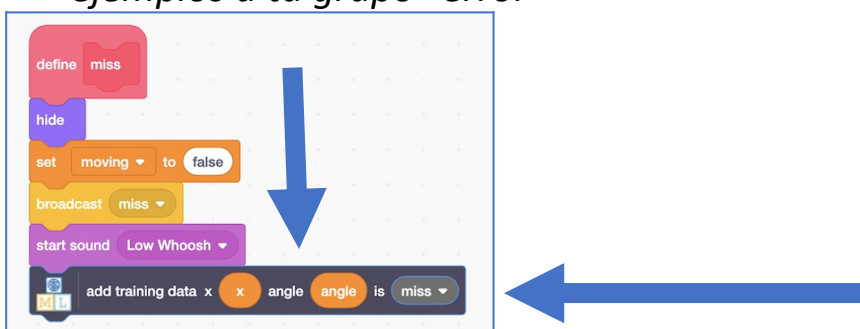
27. Agrega en bloque de “add training data” a la programación de “golpe”

Este será un ejemplo de entrenamiento para tu grupo de “golpe” cada vez que lances un tiro al bicho.

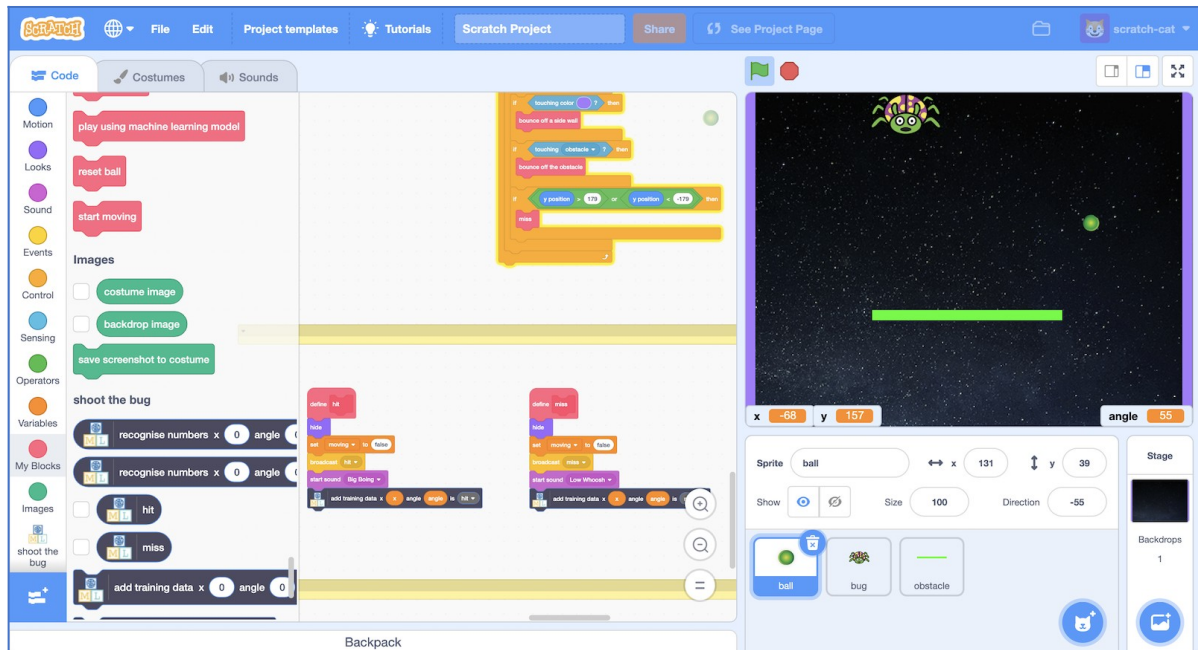
*Asegurate de agregar las variables **x** y **ángulo***



28. Agrega en bloque de “add training data” a la programación de “error”. Asegurate de actualizar la opción final a “error” para que agregue ejemplos a tu grupo “error”



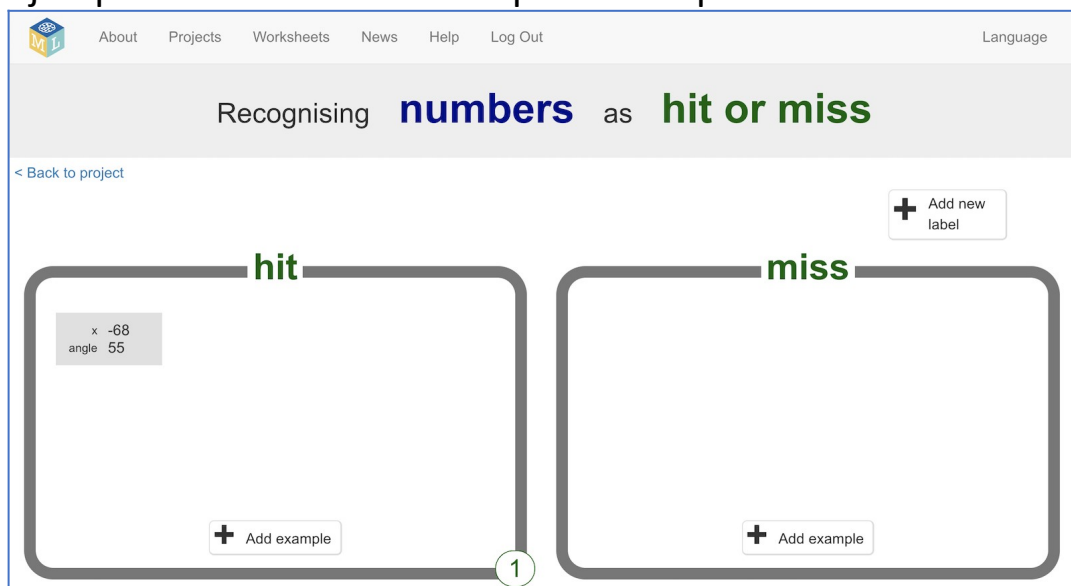
29. Haz clic en la **Bandera Verde** y prueba el juego. ¡Intenta golpear al bicho!



30. En la otra ventana del navegador web que aún está en la herramienta de aprendizaje automático, haz clic en el enlace **Volver al proyecto** en la esquina superior izquierda.

31. Haz clic en el botón **Entrenar**.

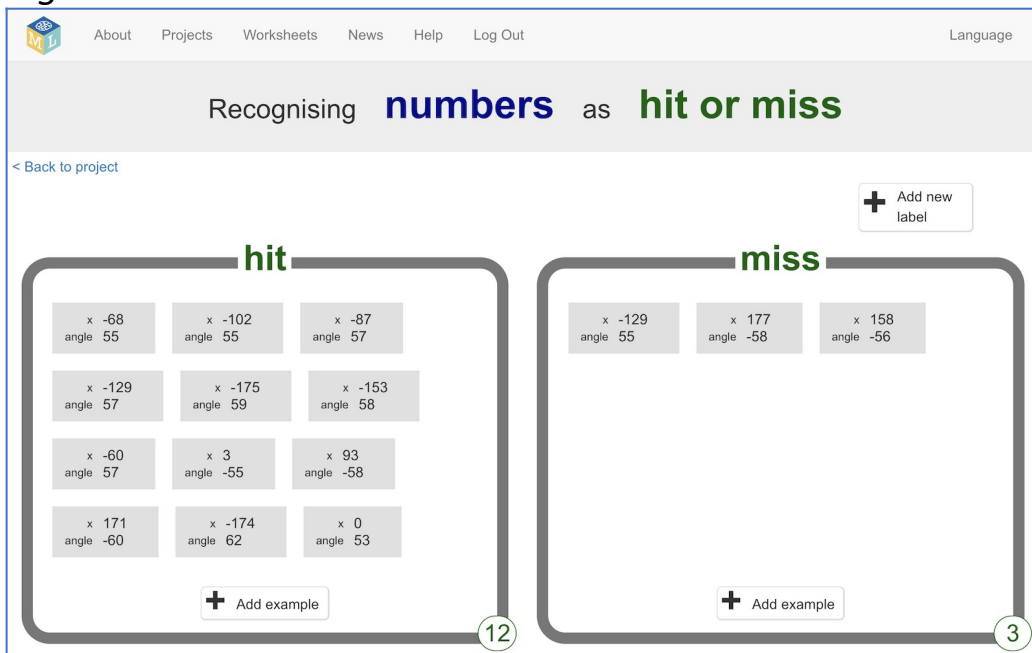
32. Verifica que el tiro que acabas de hacer se haya agregado a los ejemplos de entrenamiento que usarás para entrenar al ordenador.



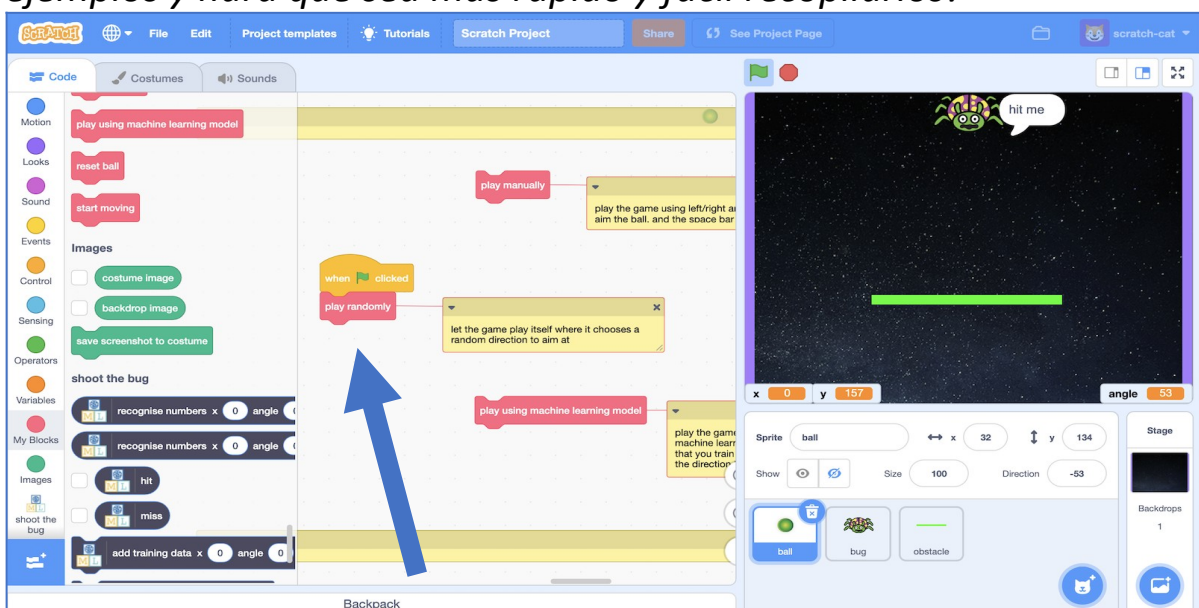
33. Vuelve a Scratch y vuelve a jugar **catorce** veces más.

Puede que te resulte más fácil jugar el juego en modo de pantalla completa.

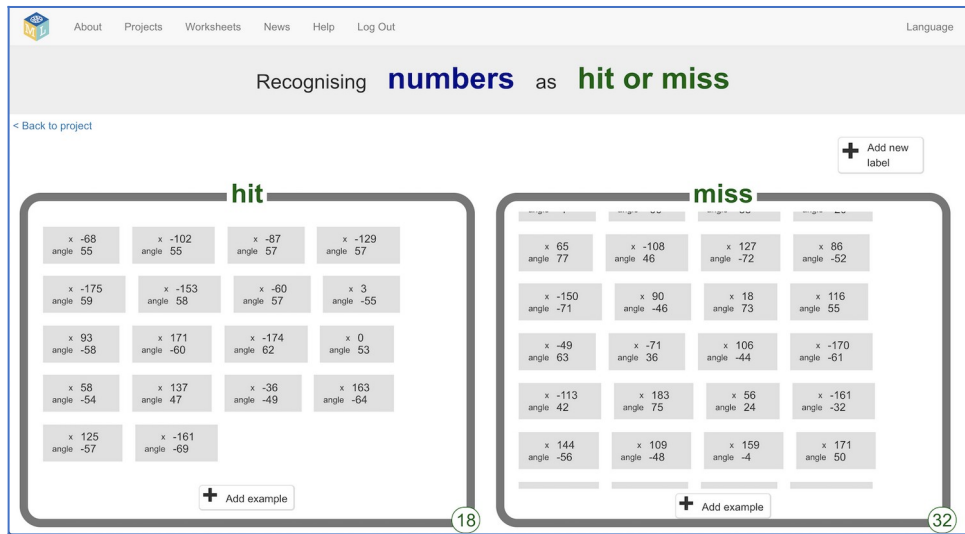
34. Comprueba cuántos ejemplos de entrenamiento has recopilado.
Intenta acertar tantos tiros como puedas, ¡pero no te preocupes si fallas algunos!



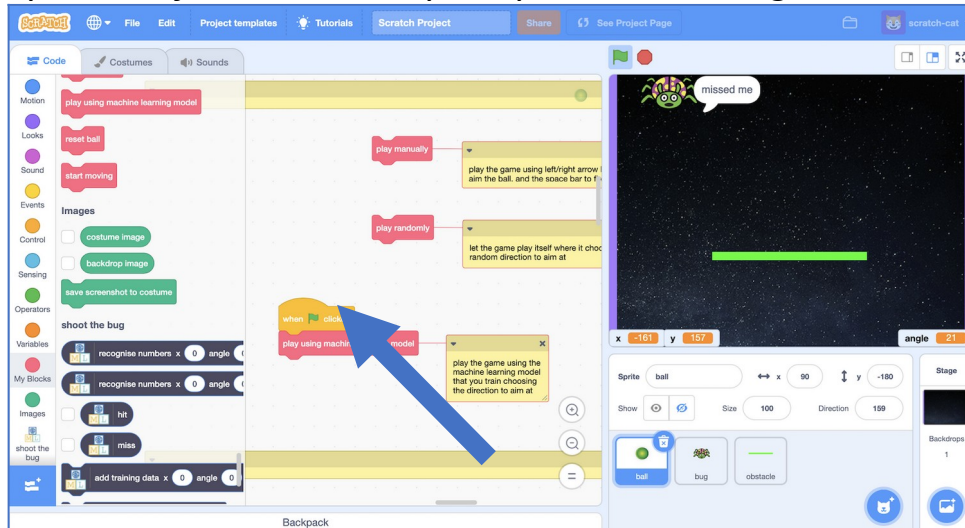
35. Actualiza la programación “al hacer clic en la bandera verde” para cuando uses “**play randomly**” (en lugar de “play manually”)
¡Usar ángulos aleatorios para tus ejemplos te dará mejor combinación de ejemplos y hará que sea más rápido y fácil recopilarlos!



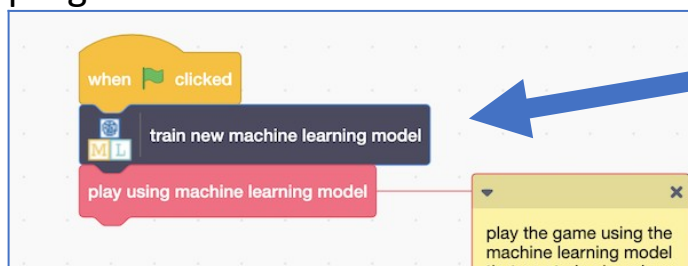
- 36.** Haz clic en la **Bandera Verde** para obtener otro ejemplo.
*Haz esto al menos unas **treinta** veces más*



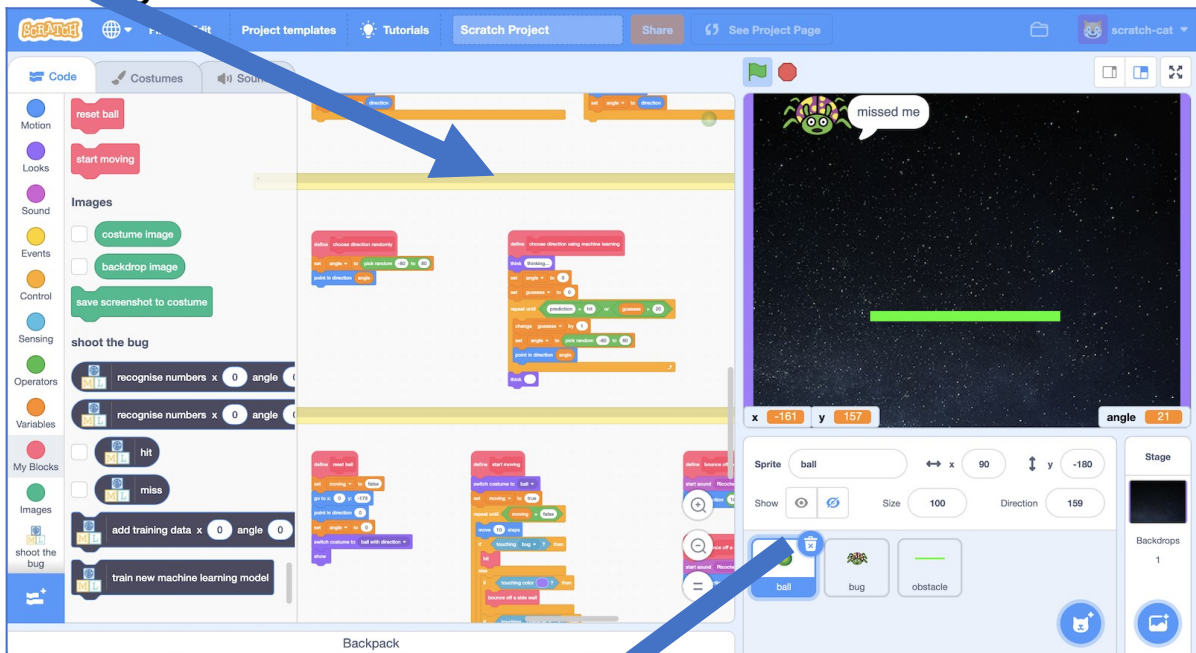
- 37.** Actualiza la programación “al hacer clic en la bandera verde” para cuando uses “**play using machine learning model**” (en vez de “play randomly”)
Debería tener suficientes ejemplos ahora para intentar usar un modelo de aprendizaje automático para predecir los ángulos correctos para disparar.



- 38.** Añade el bloque “**train new machine learning model**” block a la programación “al hacer clic en la bandera verde”

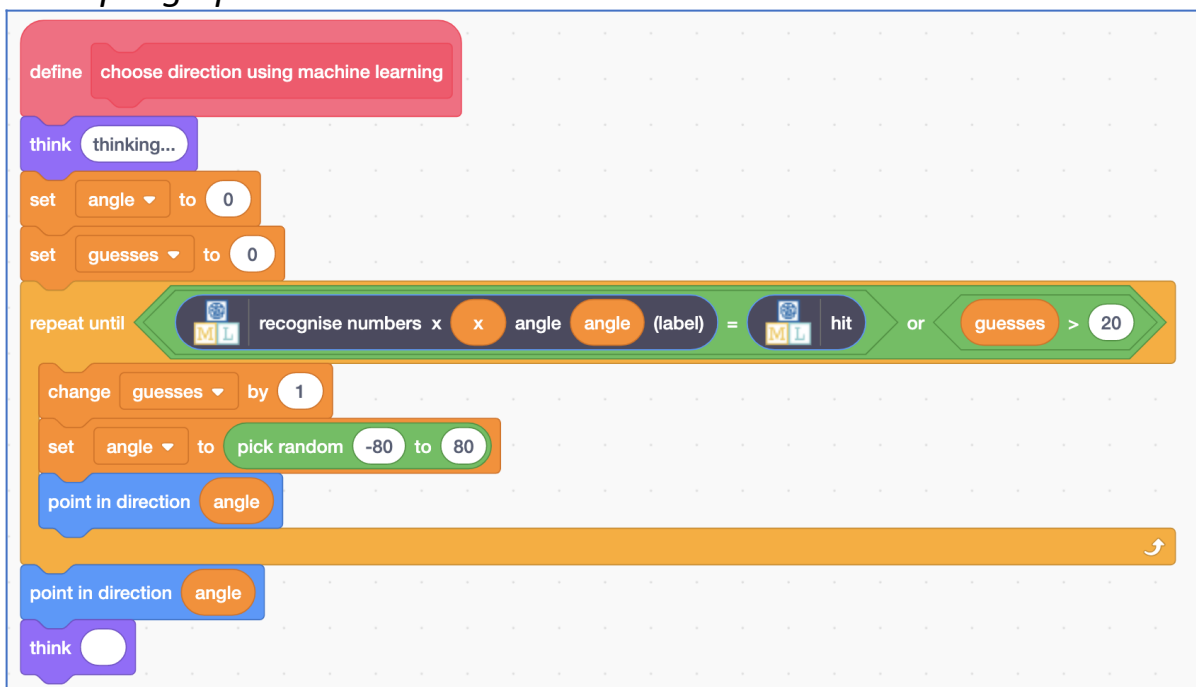


39. Encuentra la programación “definir **choose direction using machine learning**”



40. Actualiza la programación para usar tu modelo de aprendizaje automático.

Esto hará elecciones aleatorias para los ángulos a los que disparar, pero solo usa una elección aleatoria si tu modelo de aprendizaje automático cree que golpeará.



41. Haz clic en la **bandera verde** de nuevo.

¿Qué tan bueno es tu modelo de aprendizaje automático en la elección de ángulos que darán en el clavo?

¿Que has hecho hasta ahora?

Has comenzado a entrenar un ordenador para jugar un juego. En lugar de escribir reglas para poder hacer esto o resolver la ecuación para calcular el ángulo para disparar el proyectil, lo haces recopilando ejemplos. Estos ejemplos se utilizan para entrenar un modelo de aprendizaje automático.

El ordenador aprenderá de los patrones en los ejemplos. Los utilizará para hacer predicciones sobre si una ubicación y un ángulo darán como resultado un acierto o un error.

Debido a que todavía tienes los bloques "agregar datos de entrenamiento" en tu secuencia de comandos, todavía estás recopilando más ejemplos de entrenamiento cada vez que juegas. Esto significa que cuanto más tiempo dejes que tu modelo de aprendizaje automático juegue, mejor debería ser para jugar.

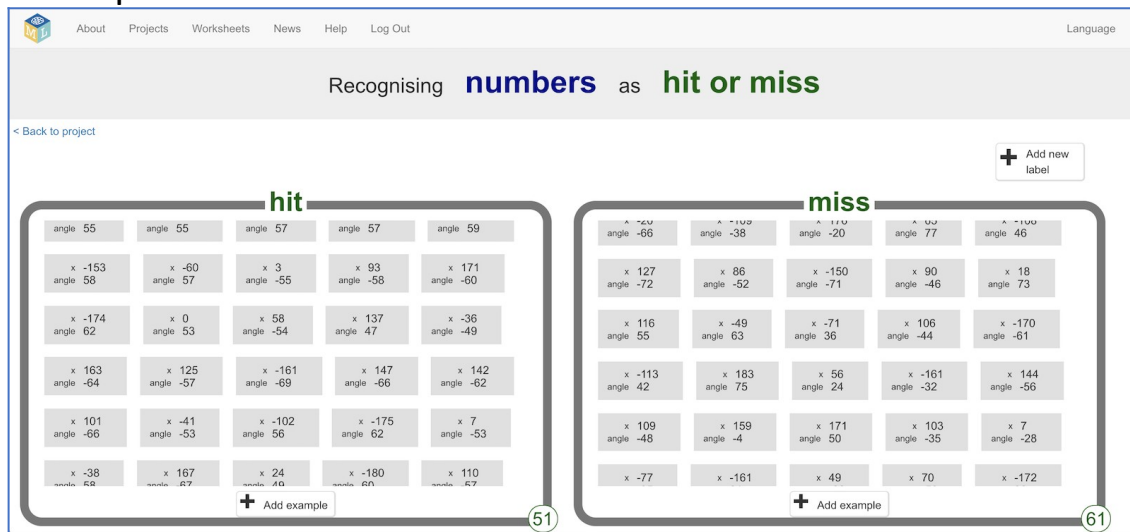
42. ¿Cuántas veces falla tu modelo de aprendizaje automático?

Si falla con demasiada frecuencia, puede deberse a que no le has dado suficientes ejemplos de aciertos.

Intenta volver a cambiar el juego al modo "jugar manualmente" y usa las teclas de flecha nuevamente. Recoge otros diez ejemplos de "golpe".

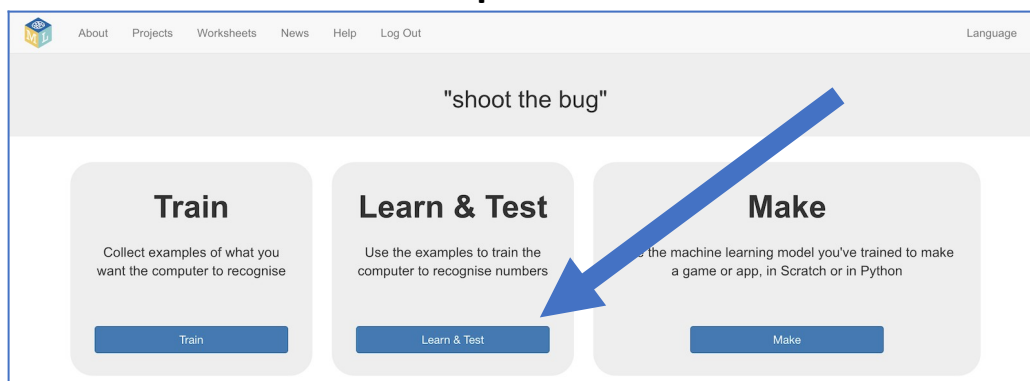
Luego, vuelve a cambiar a "jugar usando el modelo de aprendizaje automático" y mira si eso ayudó.

43. Sigue recopilando ejemplos de capacitación hasta que tu modelo de aprendizaje automático comience a ser bueno en el juego. ¿Cuántas llevas para tu modelo?



44. Haz clic arriba a la izquierda, en **Volver al proyecto**.

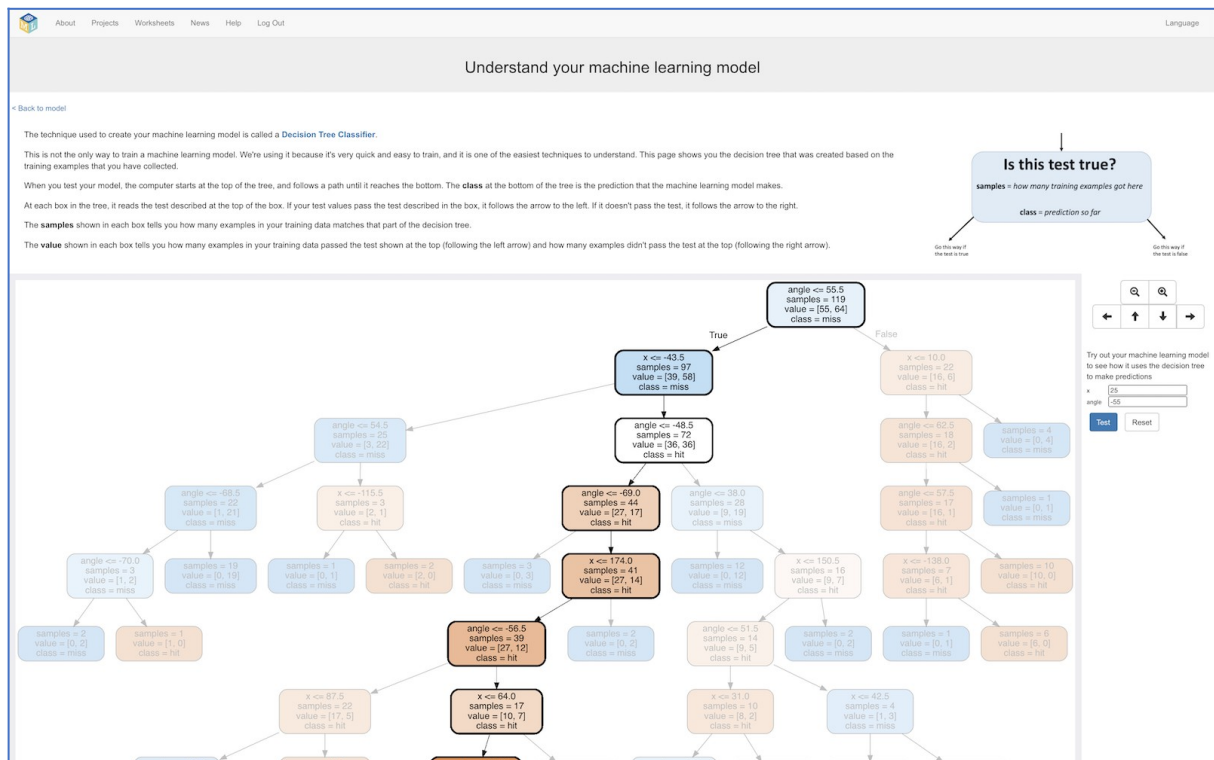
45. Haz clic en el botón **Aprender & Probar**



46. Haz clic en **Describe tu modelo**

Esta página te mostrará una imagen de tu modelo de aprendizaje automático.

*Lee la página para entender lo que significa. Intenta ingresar valores para la coordenada x del error y un ángulo para disparar, luego haz clic en **Probar** para ver cómo tu modelo de aprendizaje automático hace una predicción sobre lo que sucederá.*



47. Usa esta visualización y el juego en Scratch en modo "jugar manualmente" para ver qué predicciones está haciendo tu modelo de aprendizaje automático y si son correctas.

¿Qué has hecho?

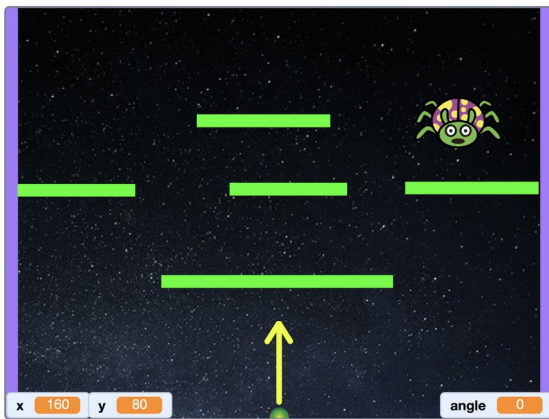
El tipo de modelo de aprendizaje automático que has entrenado es un "**clasificador de árbol de decisiones**". La visualización te permite ver cómo tu modelo hace predicciones. Es una buena manera de ver qué patrones encontró el ordenador en los datos de entrenamiento que recopiló.

¿Es este un buen uso del aprendizaje automático?

Usamos el aprendizaje automático cuando queremos que los ordenadores hagan cosas que son demasiado complicadas para que podamos escribir las instrucciones a seguir.

Evitamos el aprendizaje automático cuando el tiempo que lleva recopilar ejemplos de capacitación de una tarea sería más largo que simplemente escribir las instrucciones sobre cómo realizar la tarea.

Compara el esfuerzo de recopilar los ejemplos de entrenamiento para entrenar al ordenador para que juegue este juego, con el esfuerzo que te habría costado calcular el ángulo de disparo. ¿Crees que este juego es un buen uso del aprendizaje automático?



¿Qué pasaría si el juego se hiciera más difícil? ¿Qué pasaría si hubiera dos obstáculos para sortear? ¿O tres? ¿O cinco? ¿Qué pasaría si el bicho pudiera aparecer en alturas aleatorias, no solo en la parte superior?

Este tipo de cambios hacen que sea una tarea más difícil saber qué pasará con la pelota cuando disparas.

Las ecuaciones que necesitarías para calcular el ángulo correcto para disparar serían mucho más complicadas.

Esto hace que sea un mejor uso del aprendizaje automático que jugar el juego con un solo obstáculo.

(Pero es probable que se necesiten más ejemplos de entrenamiento para que el ordenador aprenda a jugar esto porque es una tarea más compleja que aprender a sortear un solo obstáculo. ¡Pruébalo y compruébalo tú mismo!)

Ideas y Extensiones

Ahora que has terminado, ¿por qué no pruebas una de estas ideas?

¿O inventar una propia?

Añadir obstáculos adicionales

Intenta hacer que el juego sea más desafiante agregando obstáculos adicionales a la pantalla del juego.

Deberás actualizar el script de "comenzar a moverse" para que la pelota sepa rebotar en tus nuevos obstáculos.

Usa las coordenadas x e y

Para reducir la cantidad de entrenamiento necesario, solo usamos una coordenada (coordenada x) y solo dejamos que el insecto se mueva hacia la izquierda/derecha.

Intenta hacer el proyecto nuevamente donde el bicho también puede moverse a una altura aleatoria (posición y). Deberás agregar un nuevo valor numérico para almacenar estas coordenadas cuando creas el proyecto de aprendizaje automático.

¡Hazlo competitivo!

Intenta agregar una variable para mantener la puntuación y mira si su modelo de aprendizaje automático puede obtener una puntuación más alta que la tuya.