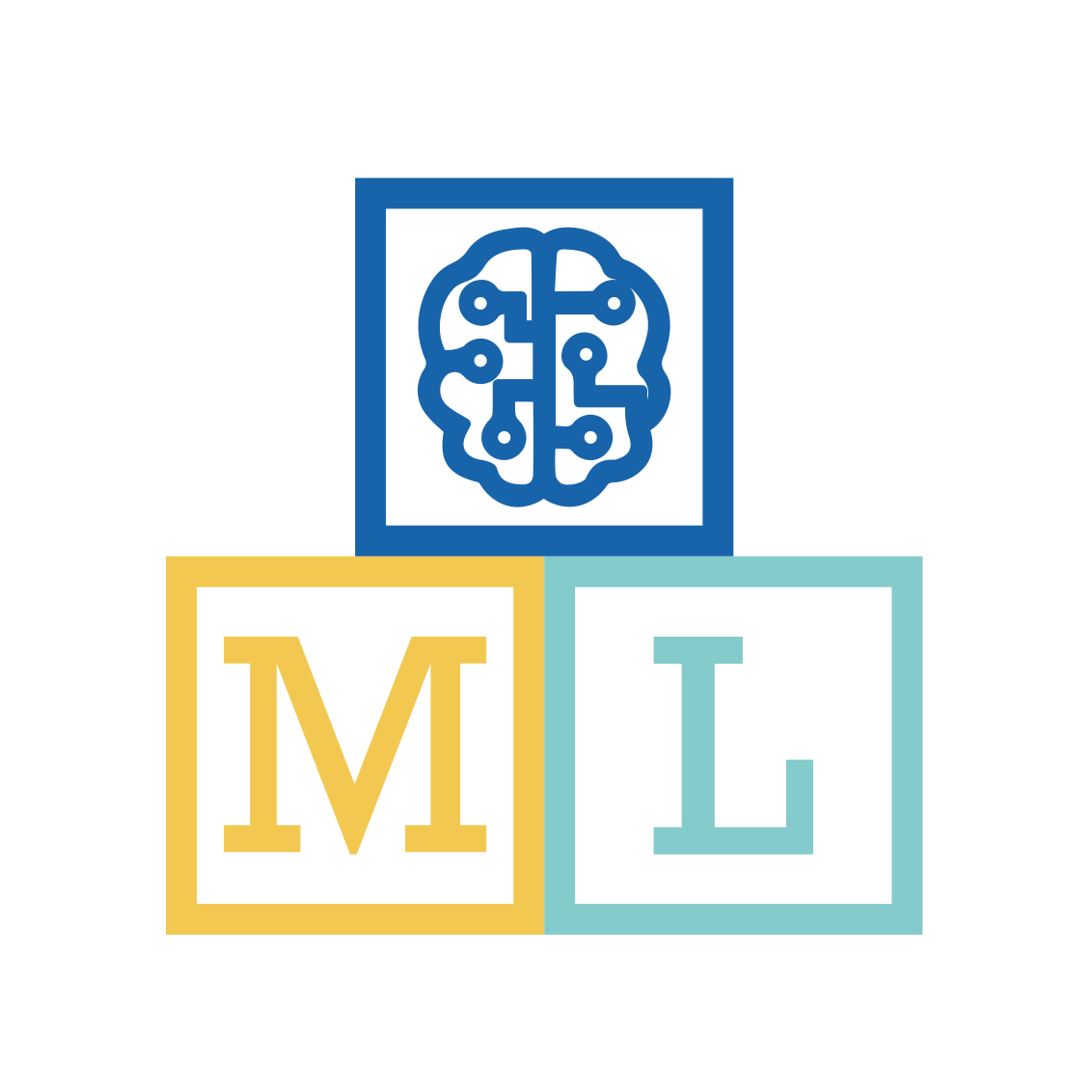
Journey to school



In this project you will train the computer to look for patterns in how your classmates get to school.

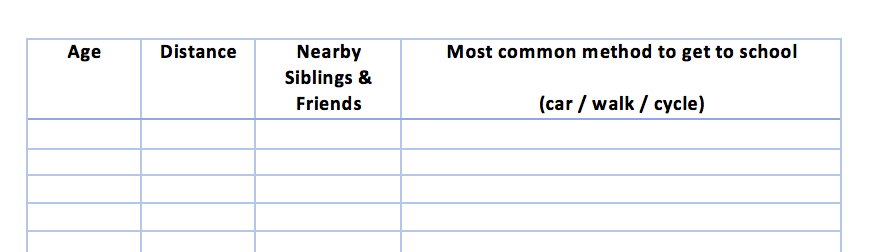
You’ll test this training by getting the computer to predict how different people travel to school.

A screenshot of a computer

AI-generated content may be incorrect.

Creative Commons LicenceThis project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License

http://creativecommons.org/licenses/by-nc-sa/4.0/

1. For this project, you will need to do a survey with your classmates. The more people you can ask, the better!
2. Write 2 or 3 questions you could ask your classmates that could affect how they travel to school. You need questions that they can answer with a number.   
   *For the rest of this worksheet, we’ll use:   
   \* Age (in years)  
   \* Distance (miles from home to school)  
   \* Number of siblings or school friends who live nearby  
   But choose your own values. Make sure that they are numbers, and pick things that could have something to do with their journey to school.*
3. Draw up a table to collect the results, and then go do your survey.   
   *Remember to ask how they travel to school as well as your questions.  
   The more children you ask, the better. If you can ask children from different classes and years, even better.*
4. Once you’ve collected answers from as many children as possible, it’s time to use this to train the computer.   
     
   Go to <https://machinelearningforkids.co.uk/> in a web browser and click on “**Get started**”.
5. 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 username or password, ask your teacher or group leader to reset it for you.*
6. Click on “**Projects**” on the top menu bar
7. Click on the **“+ Add a new project**” button.
8. Name your project “journey to school” and set it to learn how to recognise “**numbers**”  
   A screenshot of a computer

   AI-generated content may be incorrect.
9. Click “**Add a value**”, name it “age” and make the type “number”.   
   Do this again for a “number” value called “distance”.   
   Do it a third time for a “number” value called “friends”.   
   Click “**Create**” when it looks like the picture below. *Use the values you used in your survey, but* ***don’t include*** *the actual journey to school method. We’ll get to that next.*A screenshot of a computer

   AI-generated content may be incorrect.
10. You should now see “journey to school” show up in the list of your projects. Click on it.  
    A screenshot of a computer

    AI-generated content may be incorrect.
11. Click on the **Train** button to start giving your survey results to the computer.   
    A screenshot of a computer

    AI-generated content may be incorrect.
12. Click on “**Add new label**” and create a bucket called “car”.   
    Do it again and create a bucket called “walk”. Do it again for “cycle”.   
    *If you used different options in your survey, use those names instead.*   
    *A screenshot of a computer

    AI-generated content may be incorrect.*
13. Click on the “**Add example**” button in the “car” bucket and then type in the first survey result for someone who travels by car.   
    A screenshot of a computer

    AI-generated content may be incorrect.
14. Keep going until you’ve entered all the survey results.   
    A screenshot of a computer

    AI-generated content may be incorrect.
15. Click on the “**< Back to project**” link, and this time click on **Learn & Test** to use your examples.
16. Click on the **Train new machine learning model** button  
    *It should only take a few seconds to train.*   
    A screenshot of a computer

    AI-generated content may be incorrect.
17. Once training has finished, a Test box should appear.   
    Use this to test your model.   
    *Make up some values in the Test text boxes and press* ***Test****A screenshot of a computer test

    AI-generated content may be incorrect.*
18. Keep testing to try and work out the patterns that the computer has seen in your survey data.   
    *As you make the age higher, does the computer think walking becomes more likely than going by car?   
    If so, what’s the age where the computer seems to have seen the change?*
19. Click on the **Describe your model** button to see how the model is making predictions
20. Try testing your model again, and use the visualisation to understand how the model is making predictions  
    *If your survey data is complex, your visualisation might be larger or more complicated than the one in this screenshot.*  
    A diagram of a company

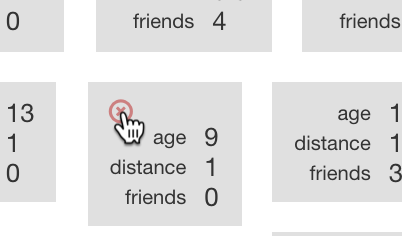
    AI-generated content may be incorrect.

**What have we done so far?**

You’ve trained a simple machine learning model using numbers from a travel survey.

You could use this model to make predictions for how likely someone is to walk to school if you know their age and how far they live from school.

But how accurate are the predictions?

1. Click on the “**< Back to project**” link and then go back into the **Train** page.
2. Choose **one** of the survey responses you entered earlier, and delete it.   
   **Make sure you write down all of the values for the example you are deleting before you delete it.** We’ll need the values in a minute.  
   *Hover the mouse pointer over it, until the red cross appears. Click on that red cross to delete the example.   
   *
3. Click on the “**< Back to project**” link and then go back into the **Learn & Test** page.
4. Click on **Train new machine learning model** again.  
   *You might need to scroll down – the button is at the bottom of the page.*
5. Once the new model has trained, enter the values that you deleted into the **Test** boxes.   
   You know what the right answer for this is – it’s the bucket that you deleted this example from.   
   Compare the computer’s prediction with the right answer.   
   Did the computer get it right?

**What have we done so far?**

You’ve tested your predictive machine learning model to see how accurate it is.

By deleting the example from the computer’s training first, it means you tested it by asking it for the answer to a question it hadn’t seen before.

(If you test it on an example that it has seen before – that it has had to learn from in it’s training – then you can’t really tell if the computer has learned how to work out the answer for itself.)

But how fair a test was it?

Did you choose a really easy example? (Like a student who lives a really really long way from school!)

Or did you choose a really hard example?

How could you have chosen an example to make it fair?

**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?

**Try using this in Scratch**

Can you think of a way to use the computer’s ability to predict how people travel to school in a game?

A screen shot of a social media message

AI-generated content may be incorrect.

**Try bigger tests**

Instead of just deleting one example from the training data and using that to test, try using more examples.

How many do you think you should use for testing?

If you use too few to test, you can’t be sure how good the computer is.

If you use too many to test, you’re reducing how many examples the computer has to actually learn from.

**Example survey results**

Not able to run a survey?

No problem! Here are the results from a small survey that we ran.

| **Age**  **(years)** | **Distance**  **(miles)** | **Friends & Siblings nearby** | **Normal journey to school** |
| --- | --- | --- | --- |
| 9 | 0.8 | 0 | car |
| 8 | 0.9 | 0 | car |
| 6 | 1.5 | 12 | car |
| 6 | 2 | 1 | car |
| 11 | 3 | 0 | car |
| 15 | 7 | 0 | car |
| 10 | 2 | 0 | car |
| 14 | 7 | 0 | car |
| 10 | 2.7 | 0 | car |
| 10 | 3.5 | 2 | car |
| 7 | 3.5 | 1 | car |
| 6 | 2.5 | 0 | car |
| 11 | 2.6 | 1 | car |
| 8 | 5 | 4 | car |
| 9 | 1.2 | 0 | car |
| 6 | 5 | 4 | car |
| 9 | 2.3 | 0 | car |
| 8 | 2.6 | 1 | car |
| 8 | 1.5 | 4 | car |
| 5 | 2.8 | 1 | car |
| 6 | 1.5 | 0 | car |
| 7 | 3 | 1 | car |
| 5 | 1 | 0 | car |
| 6 | 3 | 1 | car |
| 9 | 1 | 0 | car |
| 6 | 1.1 | 0 | car |
| 5 | 1 | 4 | car |
| 5 | 10 | 0 | car |
| 14 | 5 | 1 | car |
| 14 | 7 | 0 | car |
| 15 | 5 | 1 | car |
| 9 | 1 | 0 | car |
| 9 | 1.1 | 0 | walk |
| 12 | 0.9 | 4 | walk |
| 6 | 0.5 | 5 | walk |
| 7 | 1 | 1 | walk |
| 8 | 0.9 | 0 | walk |
| 13 | 1 | 0 | walk |
| 9 | 1 | 0 | walk |
| 11 | 1.4 | 3 | walk |
| 11 | 1.5 | 2 | walk |
| 9 | 0.9 | 0 | walk |
| 9 | 1.1 | 0 | walk |
| 9 | 1 | 1 | walk |
| 8 | 1.3 | 0 | walk |
| 9 | 1 | 0 | walk |
| 10 | 0.9 | 1 | walk |
| 9 | 1.2 | 0 | walk |
| 9 | 1 | 3 | walk |
| 11 | 1 | 0 | walk |
| 12 | 0.8 | 3 | walk |
| 9 | 0.5 | 1 | walk |
| 10 | 1.1 | 2 | walk |
| 12 | 1 | 3 | walk |
| 15 | 1.5 | 0 | walk |
| 12 | 1.2 | 3 | walk |
| 9 | 1.3 | 0 | walk |
| 10 | 0.1 | 6 | walk |
| 11 | 1.1 | 4 | walk |
| 9 | 1.25 | 1 | walk |
| 13 | 0.5 | 5 | walk |
| 7 | 1.25 | 1 | walk |
| 9 | 0.1 | 0 | walk |
| 9 | 1.4 | 1 | walk |
| 13 | 0.3 | 5 | walk |
| 15 | 0.3 | 3 | walk |
| 12 | 0.8 | 4 | walk |
| 5 | 0.8 | 4 | walk |
| 9 | 1.1 | 1 | walk |
| 9 | 1.5 | 4 | walk |
| 9 | 0.7 | 6 | walk |
| 8 | 0.6 | 4 | walk |
| 6 | 0.5 | 8 | walk |
| 16 | 1 | 10 | walk |
| 9 | 1 | 2 | walk |
| 12 | 0.2 | 8 | cycle |
| 10 | 2 | 1 | cycle |
| 11 | 0.8 | 1 | cycle |
| 10 | 1.1 | 0 | cycle |
| 13 | 1 | 1 | cycle |
| 9 | 1.1 | 0 | cycle |