Checking Your Work

Students consider the concept of trust and testing — how do we know if a particular analysis is trustworthy?

Lesson Goals	Students will be able to… - Create a subset of data to verify that a given transformation works as-advertised, using attributes of the transformation and the dataset.
Student-facing Lesson Goals	- Let's learn how to test the trustworthiness of a data analysis.
Prerequisites	- Simple Data Types - Contracts - Functions: Contracts, Examples & Definitions - Solving Word Problems with the Design Recipe - Contracts: Making Tables and Displays - Row and Column Lookups - Table Functions: Bringing it all Together
Materials	- PDF of all Handouts and Page - Trust but Verify Starter File - Lesson Slides - Printable Lesson Plan (a PDF of this web page)

Confirming Analysis *30 minutes*

Overview

Students learn how to create a *Testing Table*, which is small enough to reason about and can be used to test whether code does the right thing.

Launch

Samples are taken in Data Science and Computer Programming for different reasons. One of the main purposes of Data Science is to take a representative sample from a larger population, and use information from the sample to infer what's true about the whole population.

- Uber and Google are making self-driving cars, which use artificial intelligence to interpret sensor data and make decisions about whether a car should speed up, slow down, or slam on the brakes. This AI is trained on a lot of sample data, which it learns from. **What might be the problem if the sample data only included roads in California?**
- **Why might it be a bad thing to only test medicines on men** (or only on women), before prescribing them to the general public?

Testing Matters!

A good Testing Table should be *representative* of the population, and *relevant* to what's being analyzed. A good Testing Table should have…

- *At least* the columns that matter — whether we'll be ordering or filtering by those columns.
- Enough rows to include different circumstances that are relevant to the task at hand. For instance, if our code is supposed to extract certain cats from the animals table, our Testing Table should include at least one animal that's not a cat.
- Rows that aren't already sorted, if our analysis is supposed to sort for us.

Data scientists usually think in terms of samples that best serve the purpose of *performing inference*: Samples should be representative of the entire population, and large enough to get us fairly close to the truth about that population.

How can we trust that our code is correct?

You’ve already written lots of code to analyze data. Millions of lines of code just like yours are run on datasets every day. The results are used to tell us whether a drug is safe or not, whether someone should be put on the "no-fly" list, how much someone needs to pay for health insurance, and more. But programmers are only human, and everyone makes mistakes! And with so-called "A.I. Code Generators" out there writing more and more code for us, we need better and better ways of *verifying* that code does exactly what it claims to do!

Programmers need to think in terms of *Testing Tables* that best serve the purpose of verifying that their code does what it’s supposed to: The Tables should be designed to call attention to any imperfections in the code’s instructions.

Investigate

Testing Tables can also be used to *verify* that a certain analysis is correct.

An AI writes code that claims to filter out any shelter data to show only the cats.

- Could we test it using a Table that *already* contains only cats?
    - No! The AI’s code might do nothing at all and just return the Table it was given. It would give *the right answer for the wrong reason*! We need to find out if it actually removes non-cat Rows.
- Could we test it using a Table that has no cats at all?
    - No! The AI’s code might always return a table with no rows (regardless of species!), so giving it a Table without cats will give *the right answer for the wrong reason*! We need to find out if it actually keeps rows for cats.
- Could we test it using a Table that has only cats and dogs?
    - No! Maybe the AI’s code just removes dogs. We need to see if it removes other species as well.

Verifying that code does what it does is an important part of checking our work! That’s why writing examples is so valuable: it’s a chance to think about how the program *should* work, without worrying as much about *how* it works.

The AI writes a function called `fixed-cats` and *claims* that, given a table of animals, it produces a table with only fixed cats.

- Do you trust it? How could you test it?
- Which animals would you use in a Testing Table?

- Complete “Trust, but verify …”.
- Open the Trust but Verify Starter File. There are 3 versions of `fixed-cats`. Are they all correct? If not, which ones are broken?

An AI writes a function called `fixed-cats` and *claims* that, given a table of animals, it produces a table with only dogs five years or older.

- Do you trust it? How could you test it?
- Which animals would you use in a Testing Table?
- Turn to “Trust, but verify…” (2). Using the same Starter File, construct a Testing Table and figure out which (if any) of the functions are correct!

Synthesize

Complex analysis has more room for mistakes, so it’s critical to think about a Testing Table that allows us to trust that our code really does what it’s supposed to!

How would you check whether or not a facial recognition system was equally accurate for everyone?

When AI isn’t Intelligent… *20 minutes*

Launch

Law enforcement in many towns has started using facial-recognition software to automatically detect whether someone has a warrant out for their arrest. A lot of facial-recognition software, however, has been trained on sample data containing mostly white faces. Why might this be a problem?

Investigate

- Read "Summarizing US Congress Testimony on Artificial Intelligence", or watch this 10-minute video The Coded Gaze: Bias in Artificial Intelligence.
- Complete Can Software be Biased?

Synthesize

Discuss the article and/or video, revisiting the following questions:

- What are some concerns that experts and activists have raised about Artifical Intelligence?
- What are some solutions that would address these concerns?
- How would you test whether or not a facial recognition system was equally accurate for everyone?