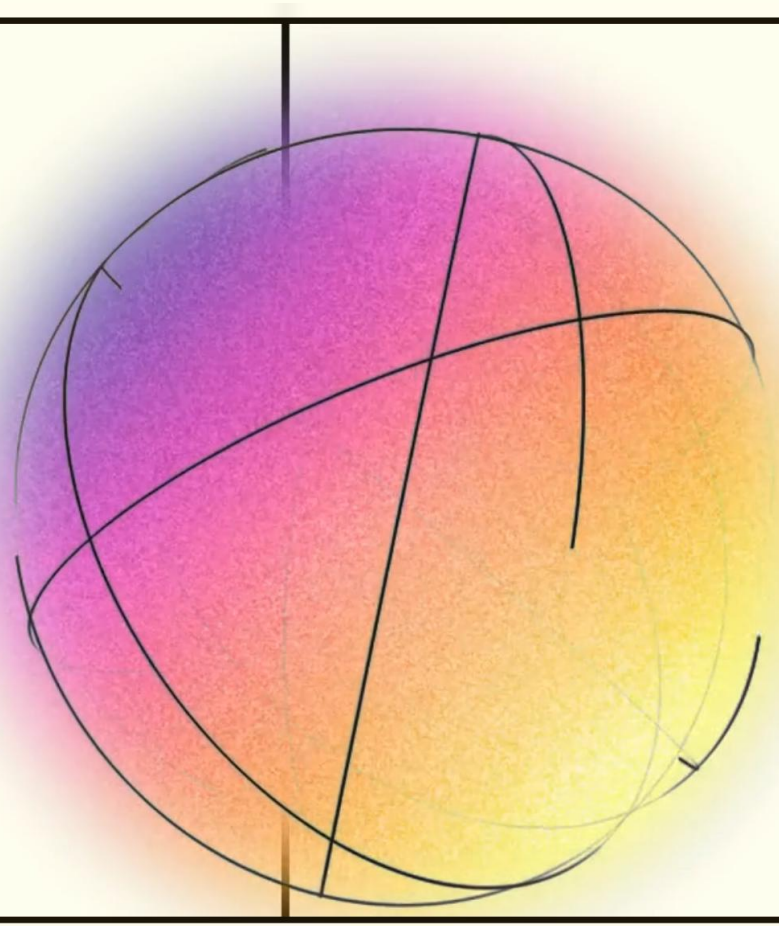
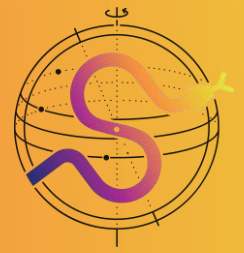
SciPy 2025



Develop Pythonic spreadsheets:
Running Python in and out of the grid

Sarah Kaiser
Sr. Cloud Developer Advocate @ Microsoft

Jim Kitchen
Sr. Software Engineer @ Anaconda



Agenda

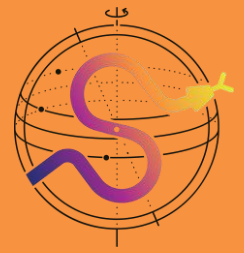
01	What is a spreadsheet really?
02	Intro to Python in Excel
03	Advanced Python in Excel
04	Advanced Analysis + Copilot for Python in Excel
05	Anaconda Toolbox and Code
06	Using `pandas` with spreadsheets
07	Customize spreadsheets with `openpyxl`

You will learn how to..

- Load and work with spreadsheets in Python
- Run Python directly in Excel
- Use AI tools like Advanced Analysis in Excel
- Use Anaconda Toolbox and Code in Excel

1. What is a spreadsheet really?

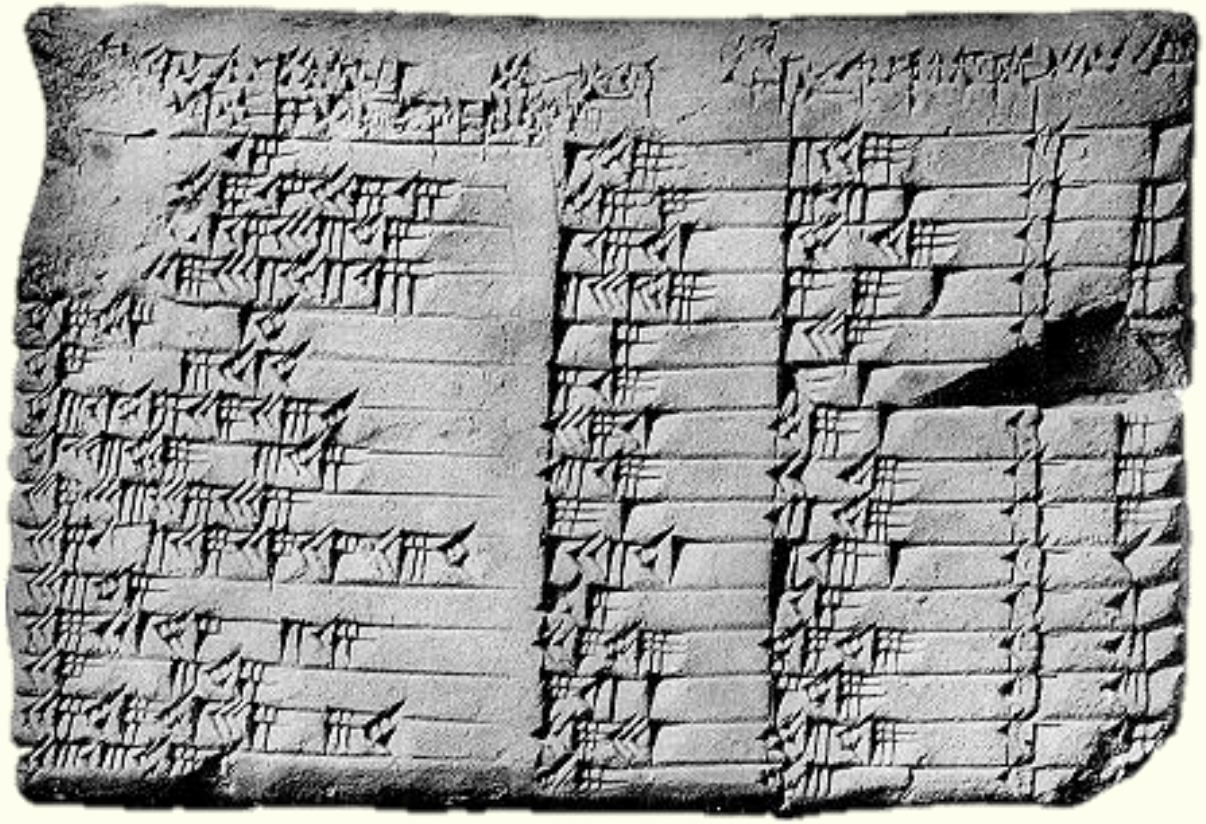
	A	B	C	D	E
1	sepal_length	sepal_width	petal_length	petal_width	species
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa



data.csv

```
Year,Make,Model,Length
1997,Ford,E350,2.35
2000,Mercury,Cougar,2.38
```

Tab(let)ular data



"Plimpton 322". *personal.math.ubc.ca*. Retrieved 2023-05-20.

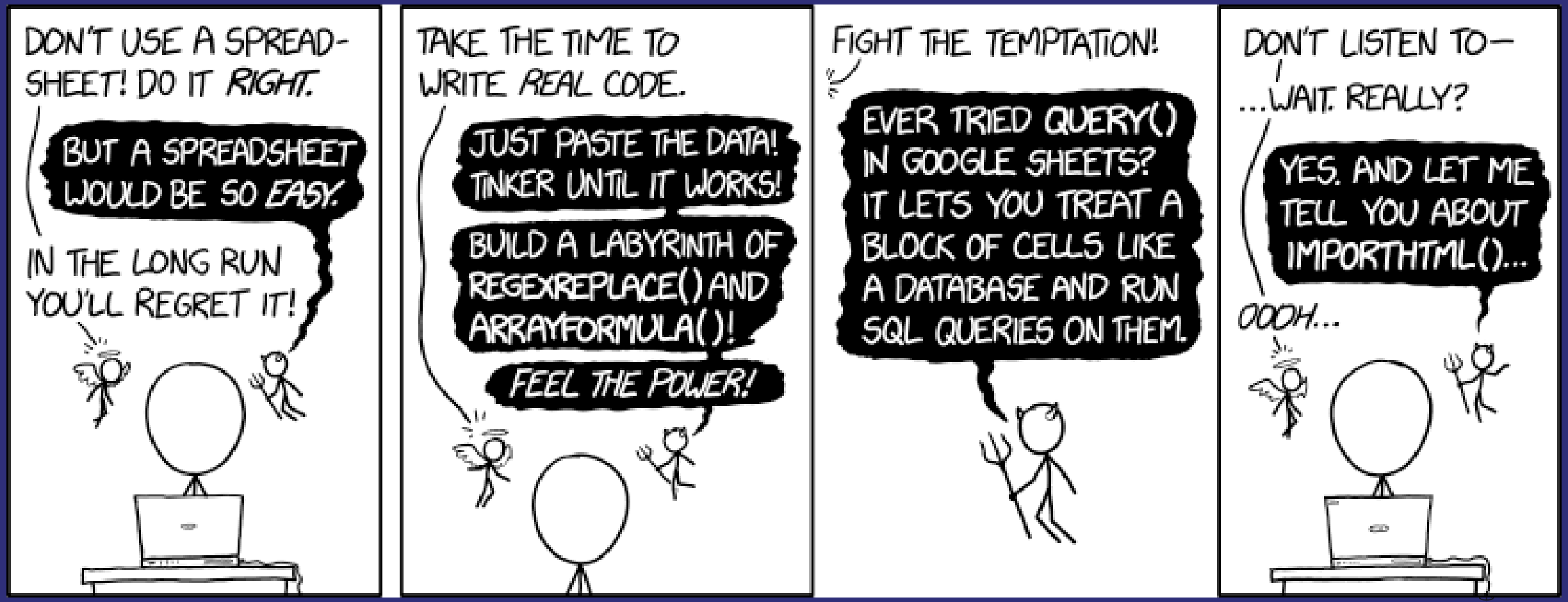
Tabular text formats

- csv, tsv, or other text delineated files
  - **Lowest common denominator format**
  - Human readable
  - Efficient storage
- Mainly used for storage, need other tools to analyze/plot

Tabular ➡️ Spreadsheets

- Applications that allow storing and manipulation of tabular data
  - ex. VisiCalc, Lotus 1-2-3, Excel, LibreOffice Calc, PySpread
- Automatic/real-time calculation alongside data
- “What-if” analysis

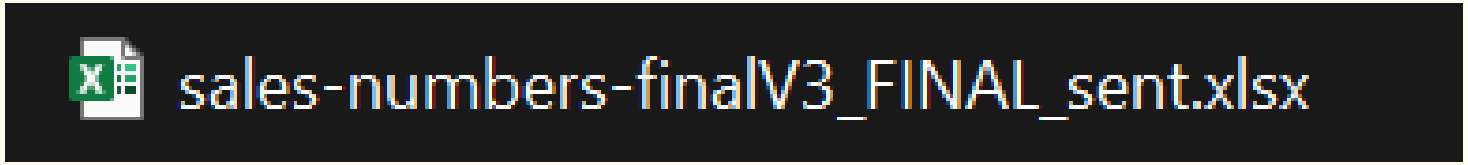
in other words… **Accessible Data Science**



https://xkcd.com/2180/

Spreadsheets: Limitations

- Harder to audit/version control
  - Can be harder to collaborate
- Potentially lots of function repetition
- More limited data sizes



	A	B	C
1	sepal length	sepal width	species
2	5.1	3.5	setosa
3	4.9	3	setosa
4	4.7	3.2	setosa
5	4.6	3.1	setosa

sheet1

Cell reference syntax

- B2
- A3:C3
- sheet1!C5
- C1
  - $ pins a reference, can do column, row or both

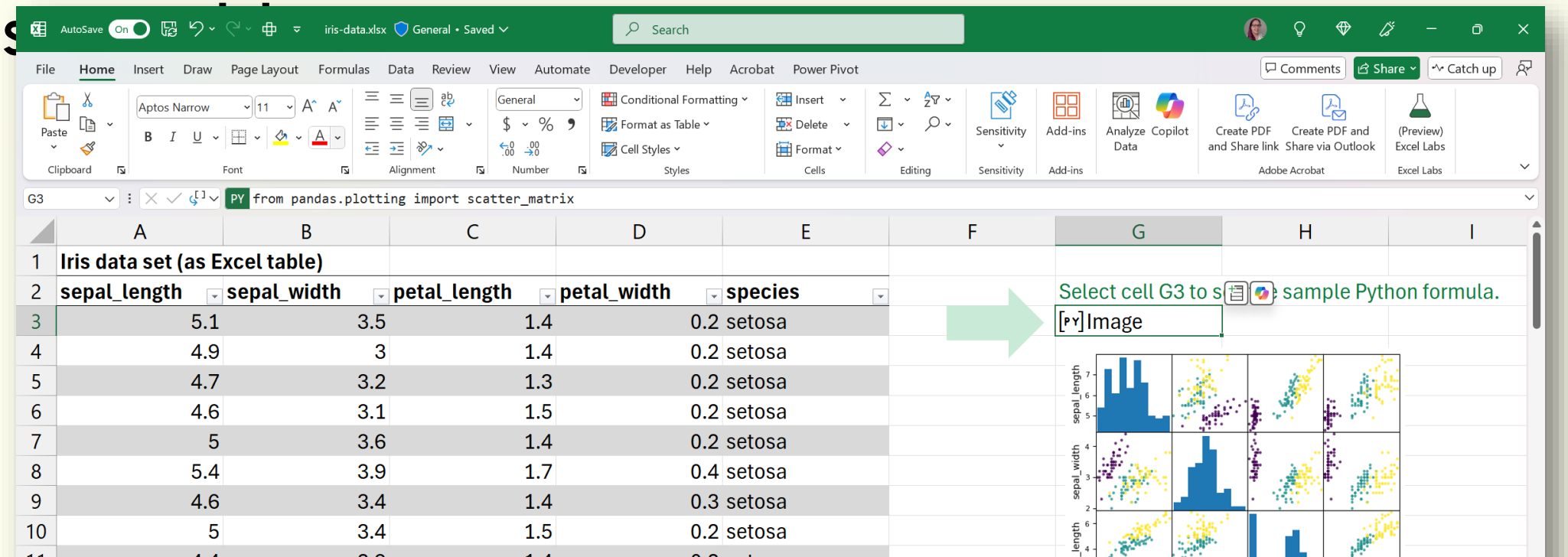
Historically, dates are hard…

- Excel incorrectly assumes that the year 1900 is a leap year - Microsoft Learn
  - Dates and Times — openpyxl documentation
- ISO 8601 is best pratice
  - 2025-05-12T23:38:55Z or 20250512T233855Z

tl;dr Be cautious if you are using dates prior to 1900!

Unwrapping an xlsx file

Spreadsheet documents are standardized, change extensions to *.zip and unpack an xlsx/odf

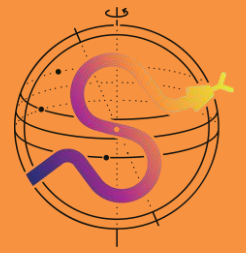


Excel Document Format

```
\IRIS-DATA-XLSX
├──docMetadata
├──docProps
├──xl
│   ├──drawings
│   │   └──_rels
│   ├──media
│   ├──richData
│   │   └──_rels
│   ├──tables
│   ├──theme
│   ├──worksheets
│   │   └──_rels
│   └──_rels
└──_rels
```

ODS Document Format

```
\IRIS-DATA-ODS
├──Configurations2
│   ├──accelerator
│   ├──floater
│   ├──images
│   │   └──Bitmaps
│   ├──menubar
│   ├──popupmenu
│   ├──progressbar
│   ├──statusbar
│   ├──toolbar
│   └──toolpanel
├──META-INF
├──Pictures
└──Thumbnails
```



2. Python in Excel: Use Python directly in spreadsheets

Macros refine data

- Spreads
- Macros
- Can hav
- VBA or

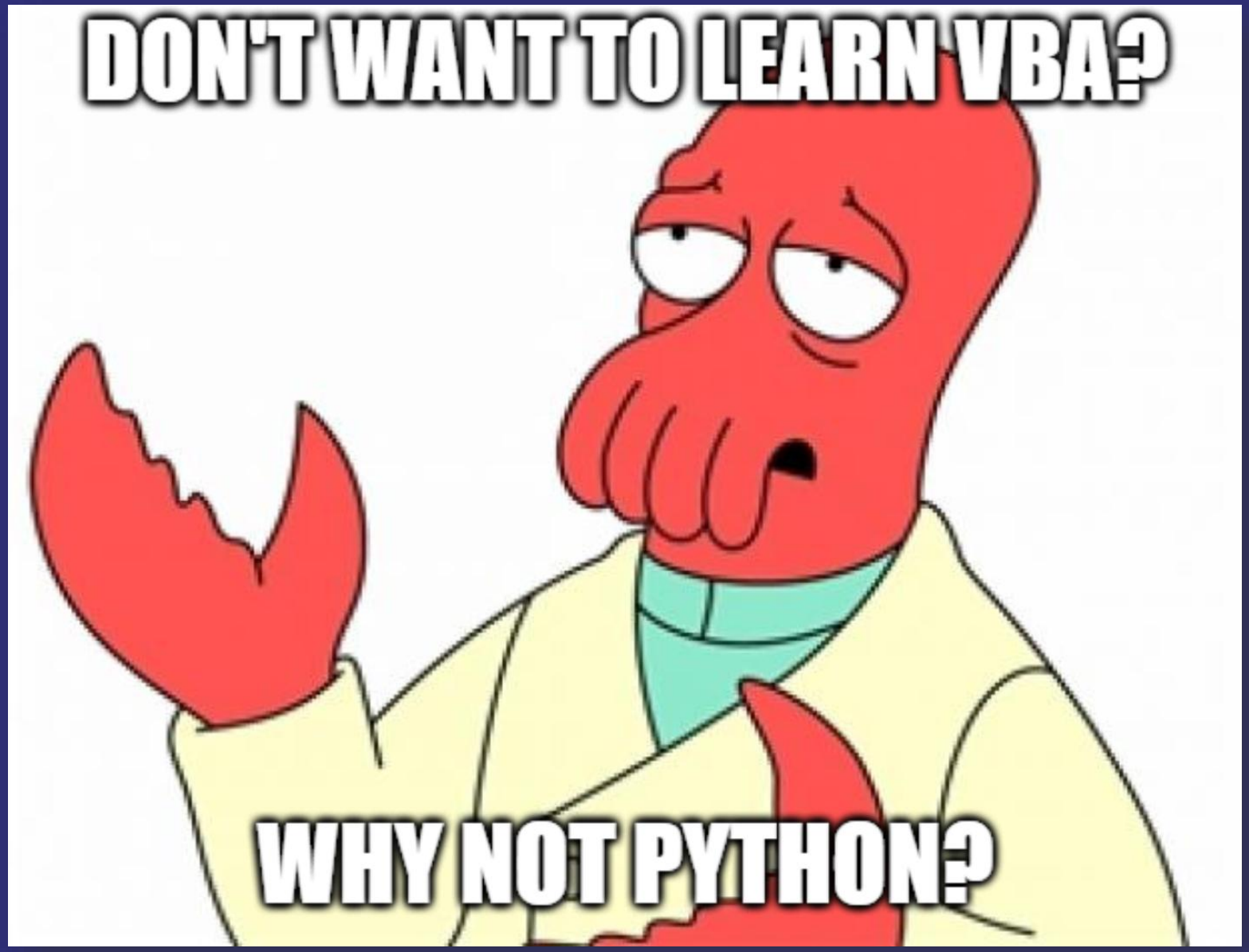
or functions

dvanced

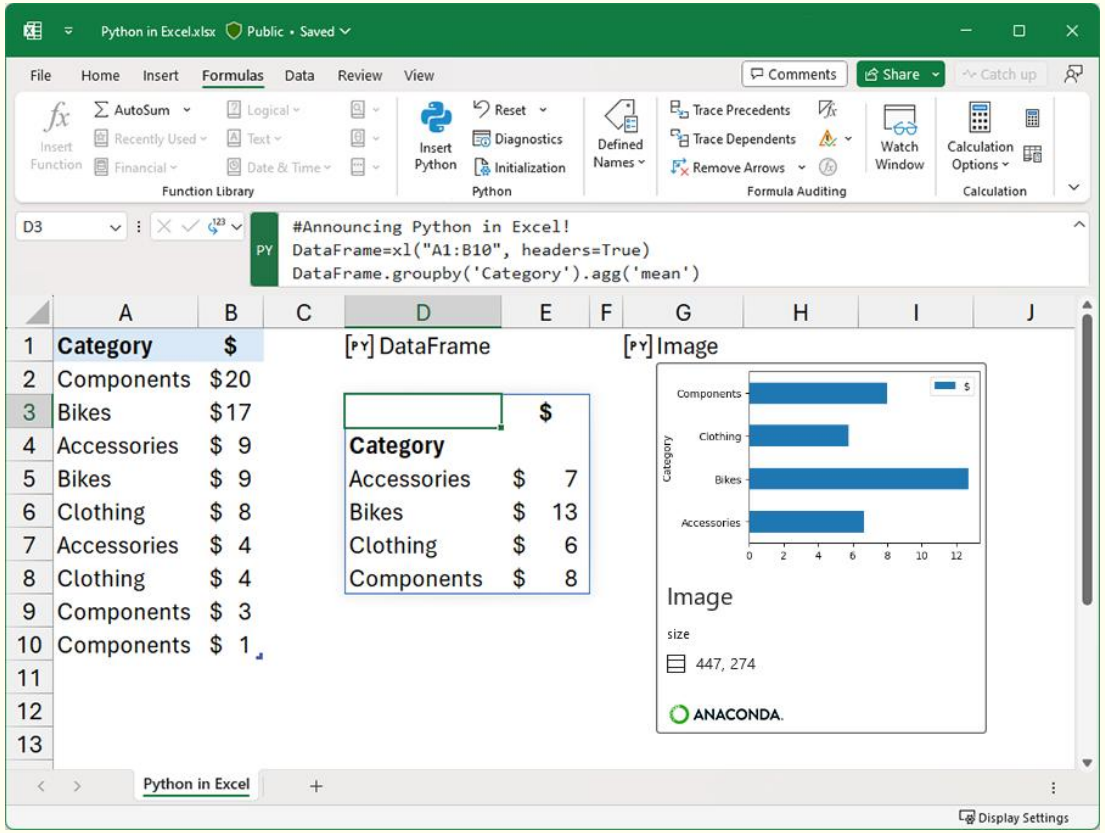
actions

all there

was for


DON'T WANT TO LEARN VBA?
WHY NOT PYTHON?

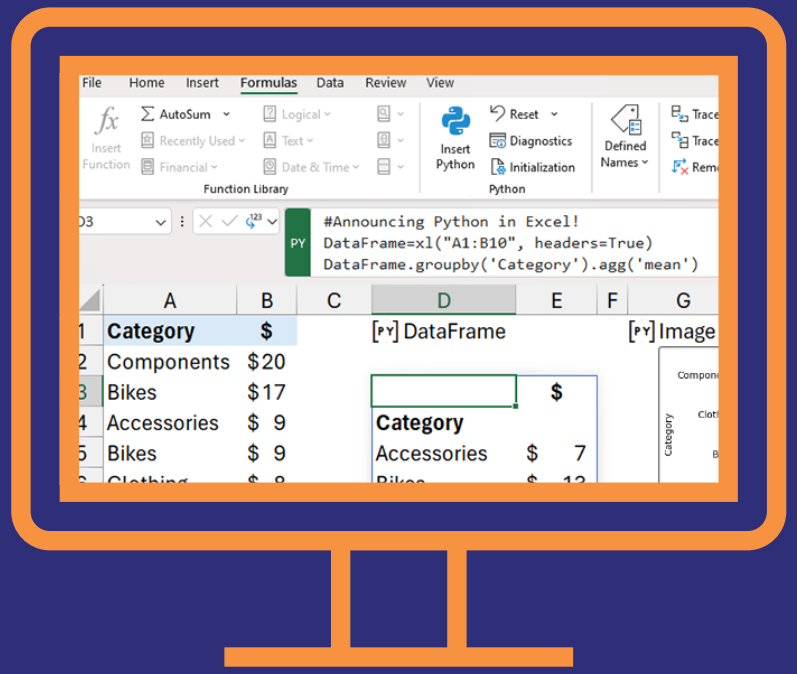
Python in Excel



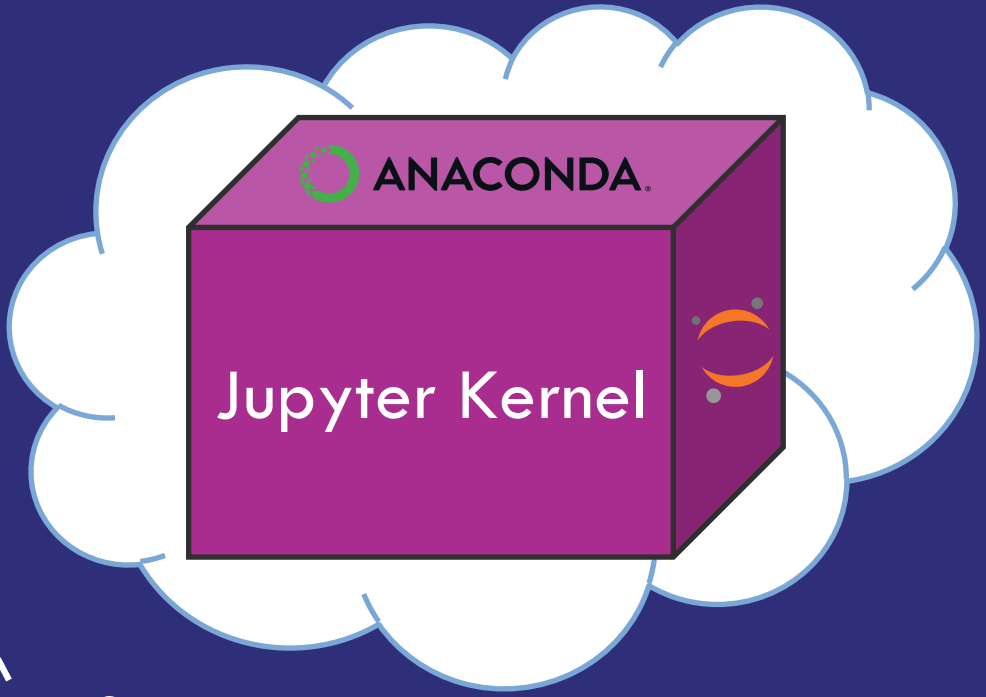
- Newish add-on feature of Excel that allows you run Python directly in the cell
- Add Python to existing workflows and analyses without creating new “projects”

How does it work?

Code cells are turned into a Jupyter notebook cells and run it in a cloud container



Data is exchanged with container from Excel via xl package



The container stay online as long as the workbook is open or until a timeout occurs.

TL;DR: 2D Jupyter Notebook running in an Anaconda distro in the cloud

Python in Excel tradeoffs

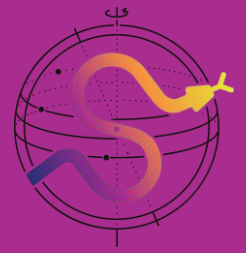
Advantages

- Highly vetted, secure environments to run code on the cloud
- Interoperates with other 365 experience tools like live editing, share links, and comments
- Copilot integration

Limitations

- Need network access to use Python and can't access internet in cloud container to get data
- Supports a small subset of the extensive Python package ecosystem
- Current data size limit (~100MB)

Let’s look at a demo!



Remote Execution

Python code runs on hypervisor isolated containers built on Azure Container Instances.

Trusted Packaging

The container has Python and a curated set of secured libraries provided by Anaconda.

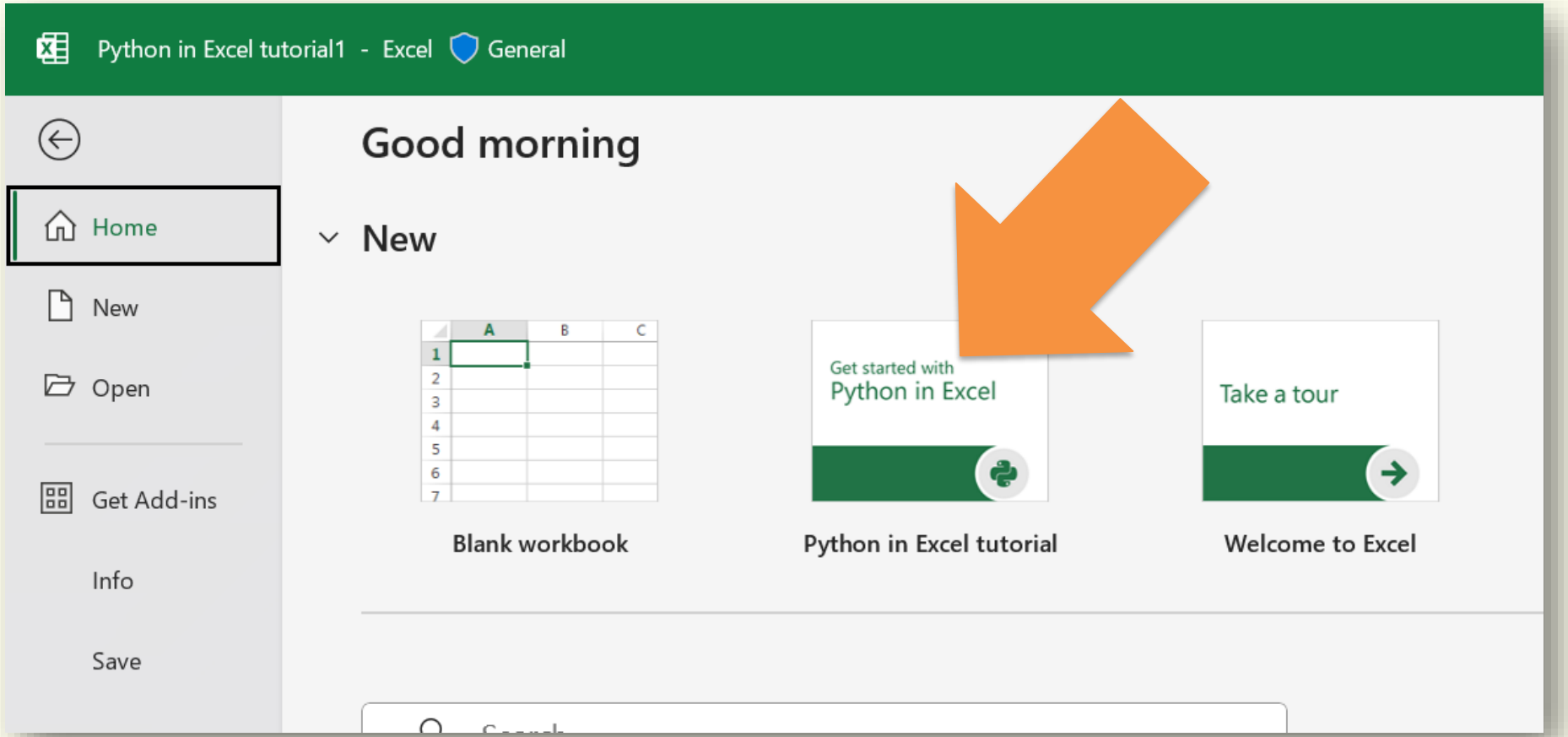
Secure Devices

No local disk access, user token, or network access.

Easy Collaboration

Connected to other tools like OneDrive, Advanced Analysis, sharing links, etc.

Fantastic built-in reference for all things Python in Excel

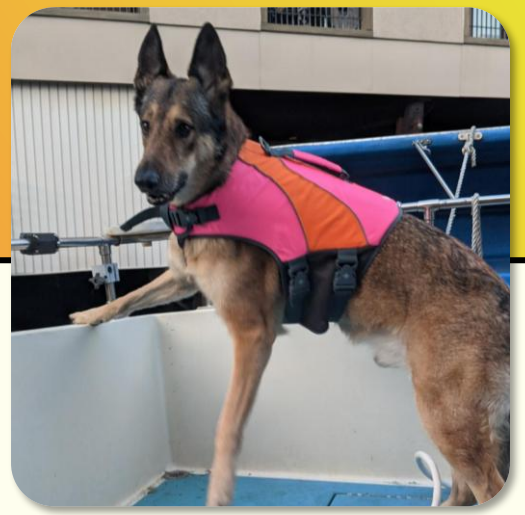


Exercise: Packages in Python in Excel

1. In the tutorial repo, download `2_python-in-excel.xlsx` which in Library Index sheet has a list of the packages in the current Python in Excel environment.

2. Pick one of the non-core libraries and in a new sheet create a code sample of using that library in Excel. Checking out the docs for the package is a good start!

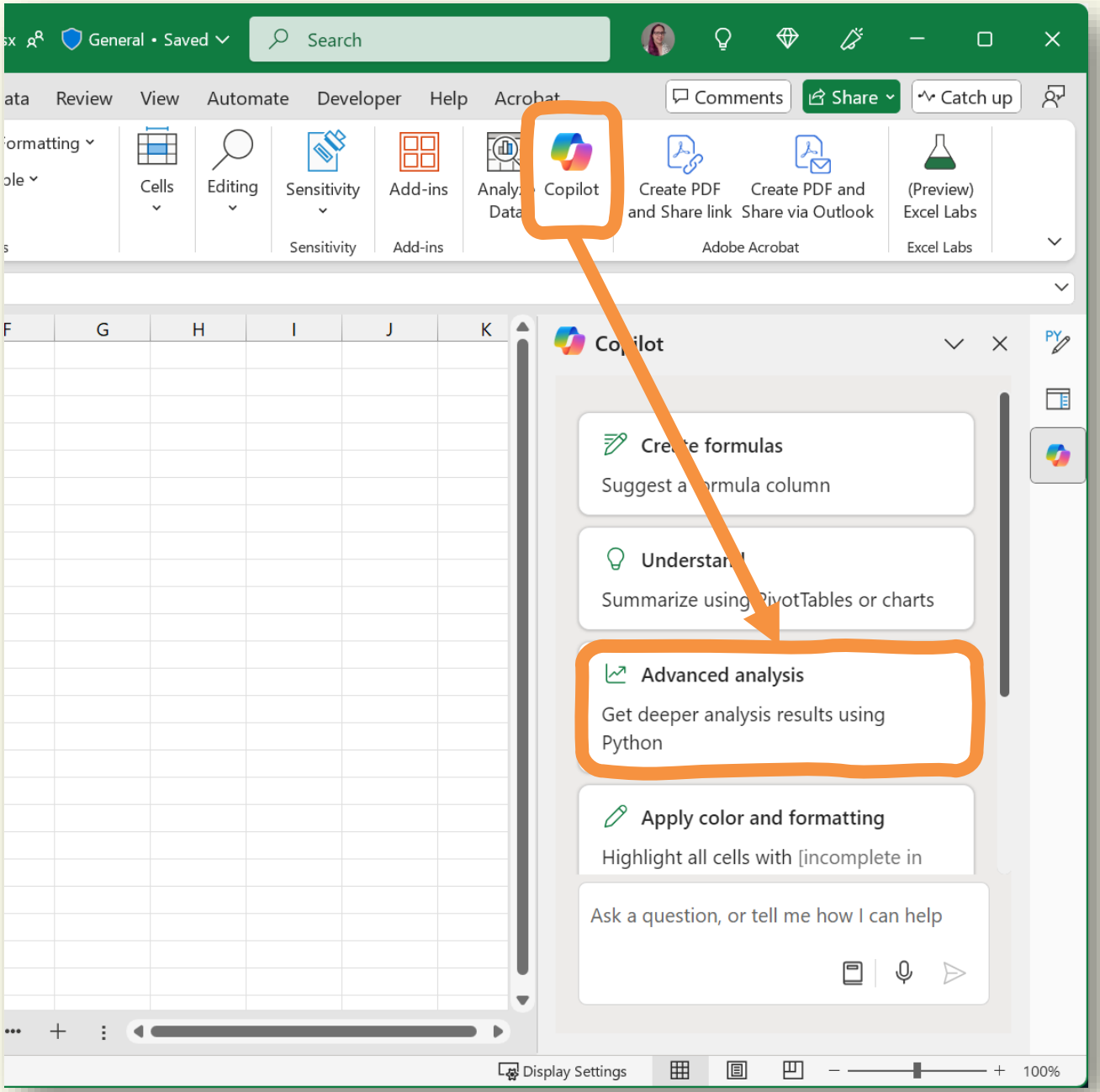
aka.ms/scipy-sheets



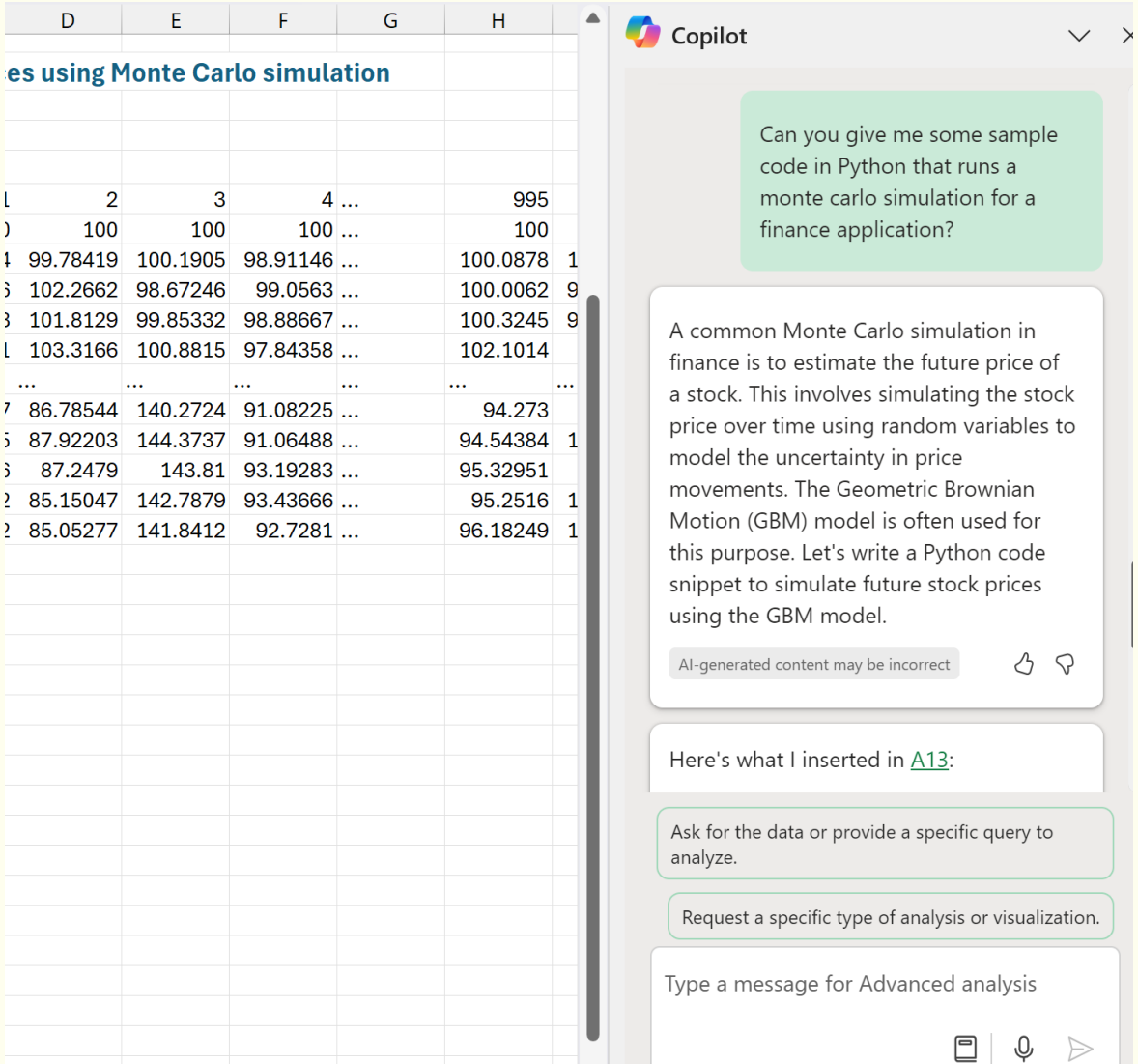
4. Advanced Analysis + Copilot for Python in Excel

Pair program your data analysis

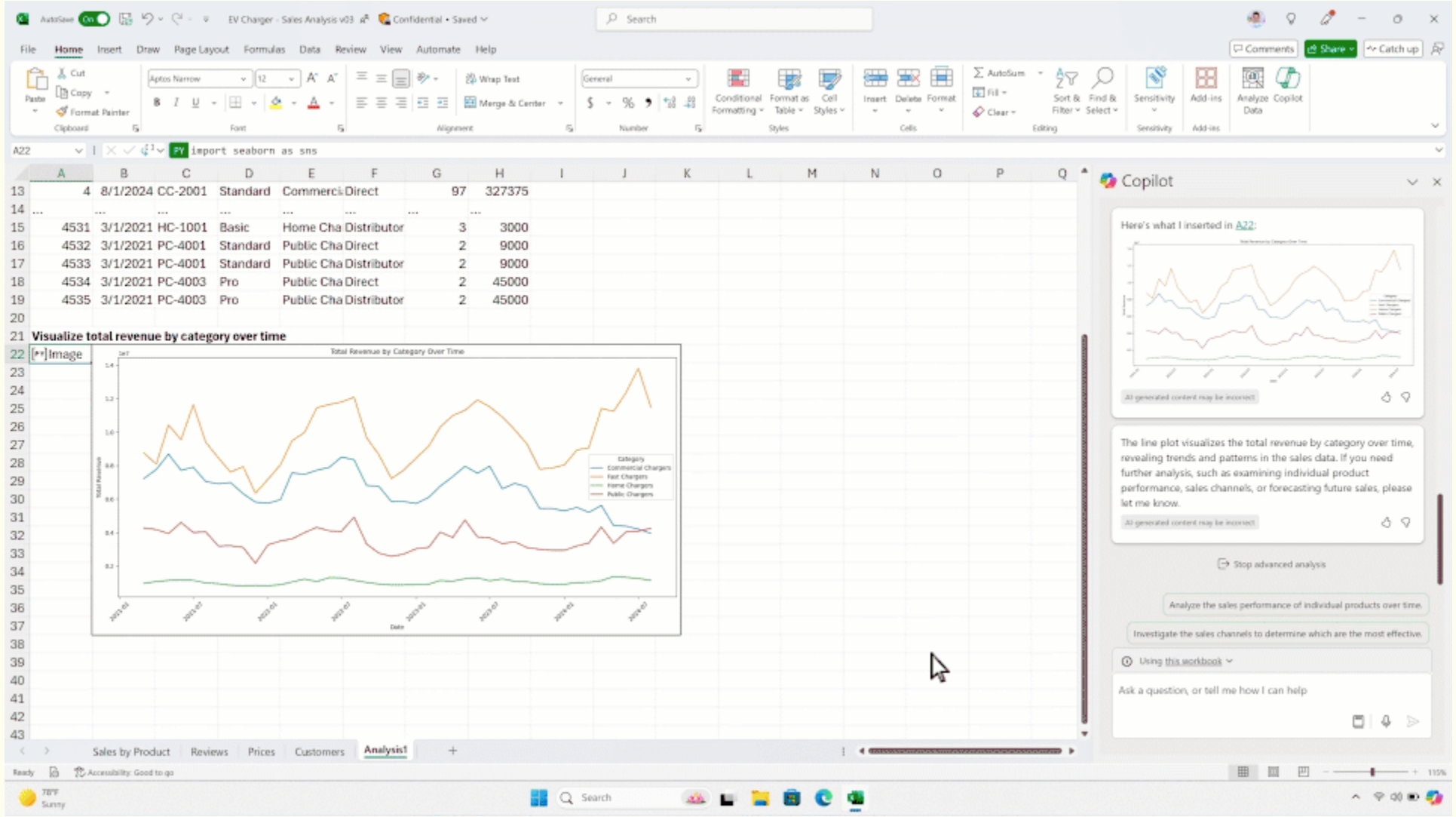
Use **Advanced Analysis** mode in **Copilot** which can generate Python code for you to start with!



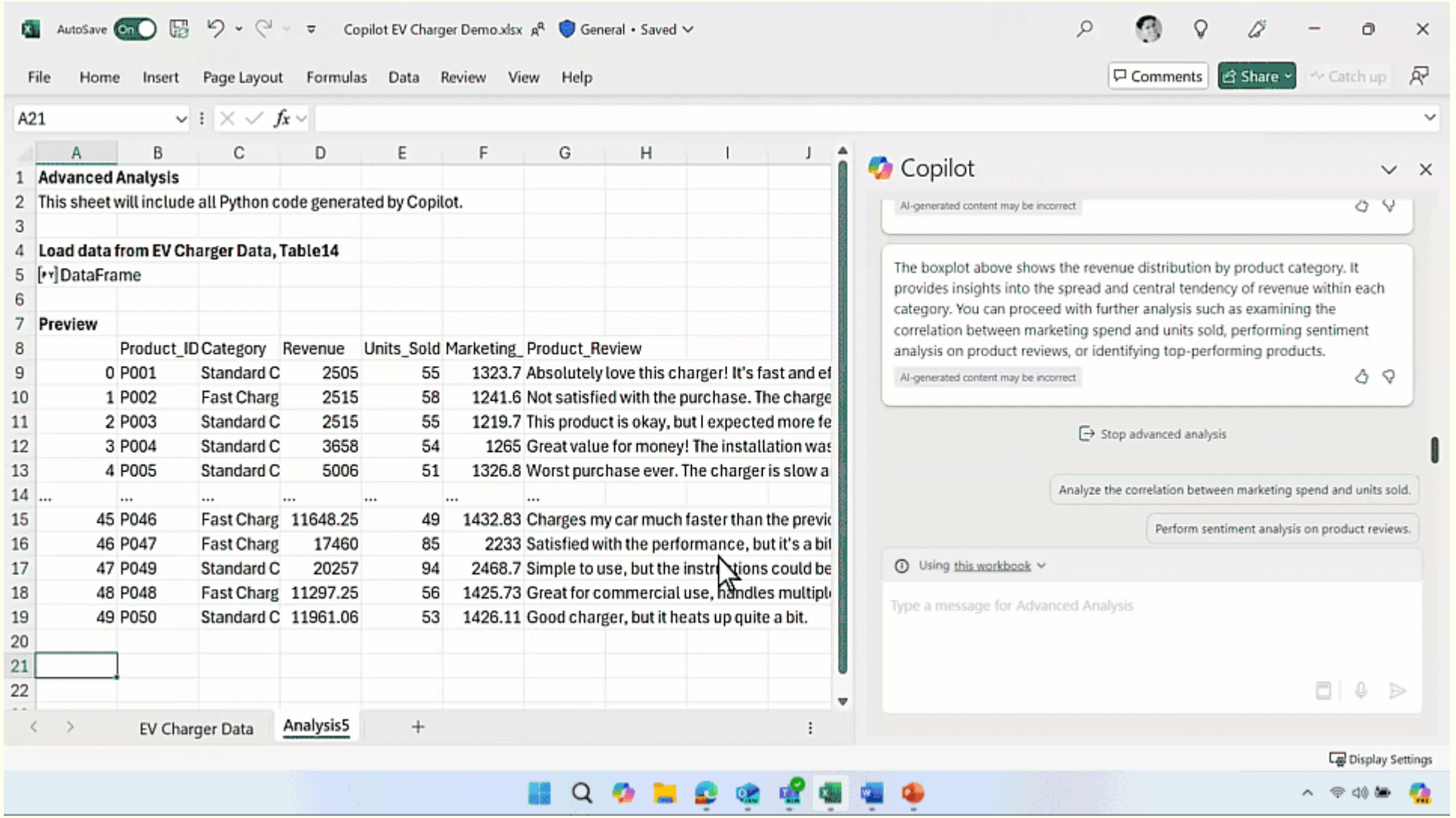
Ask for help starting a project



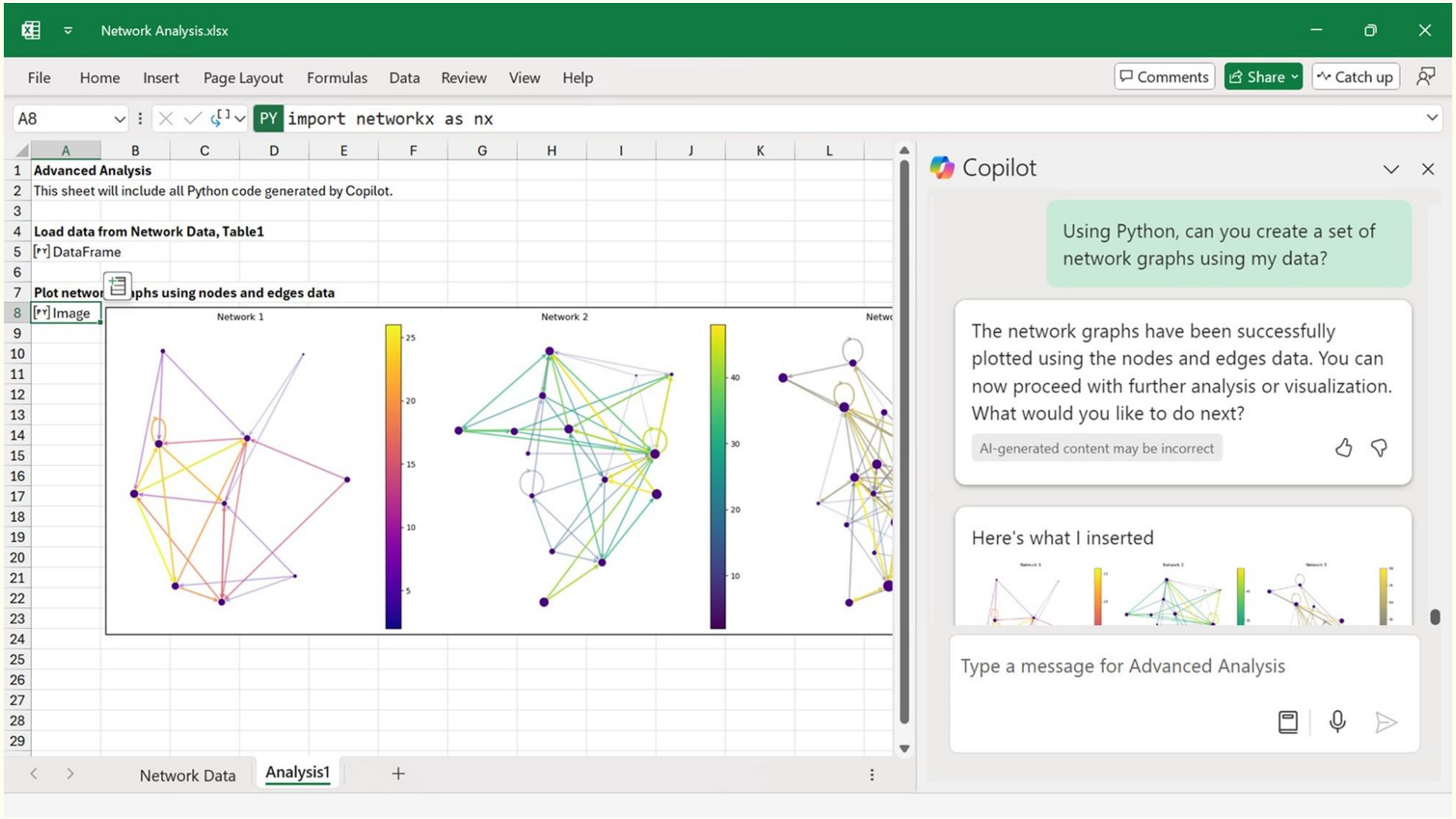
Ask for help forecasting and exending visuals



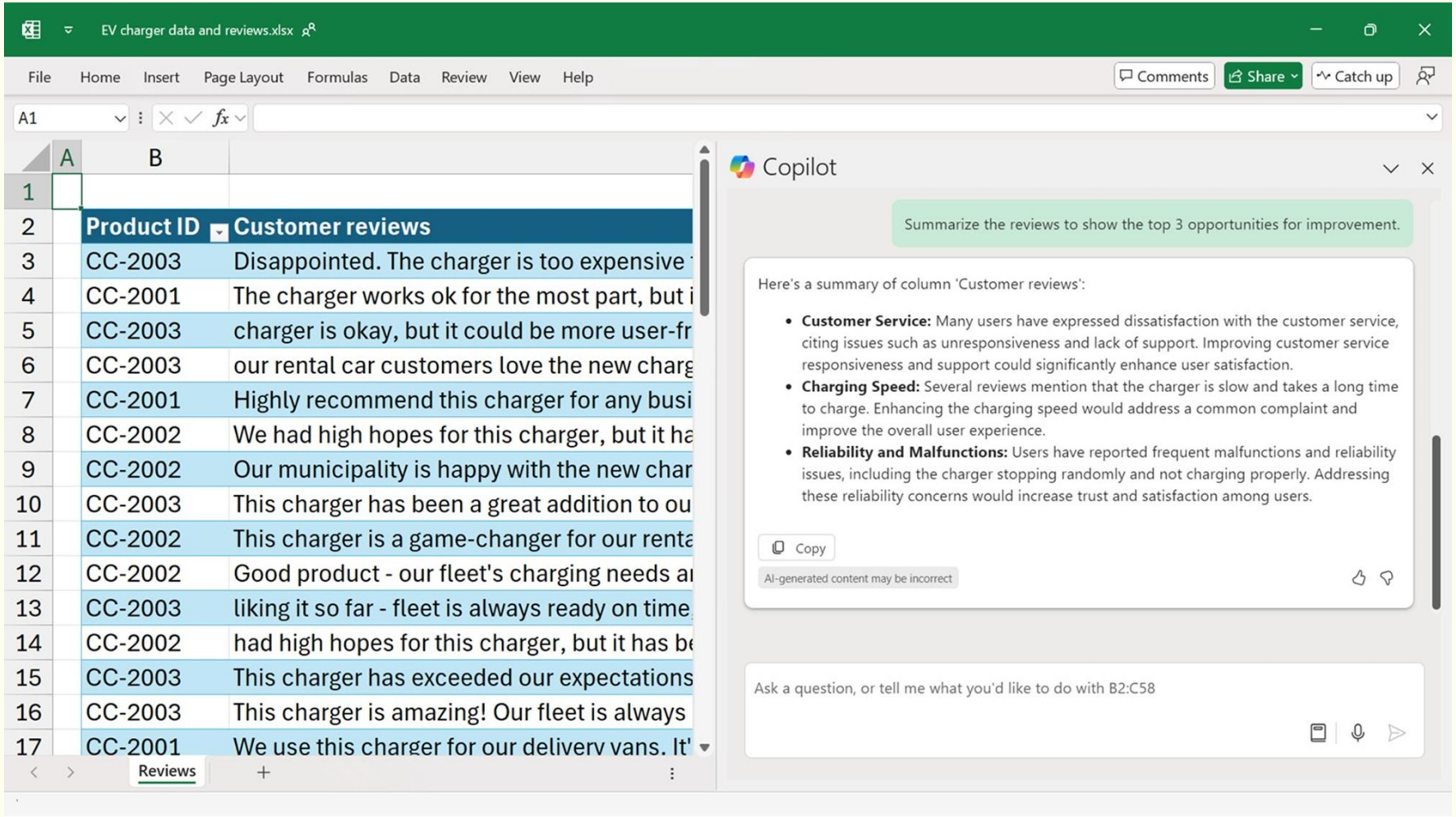
Ask for help creating visualizations…



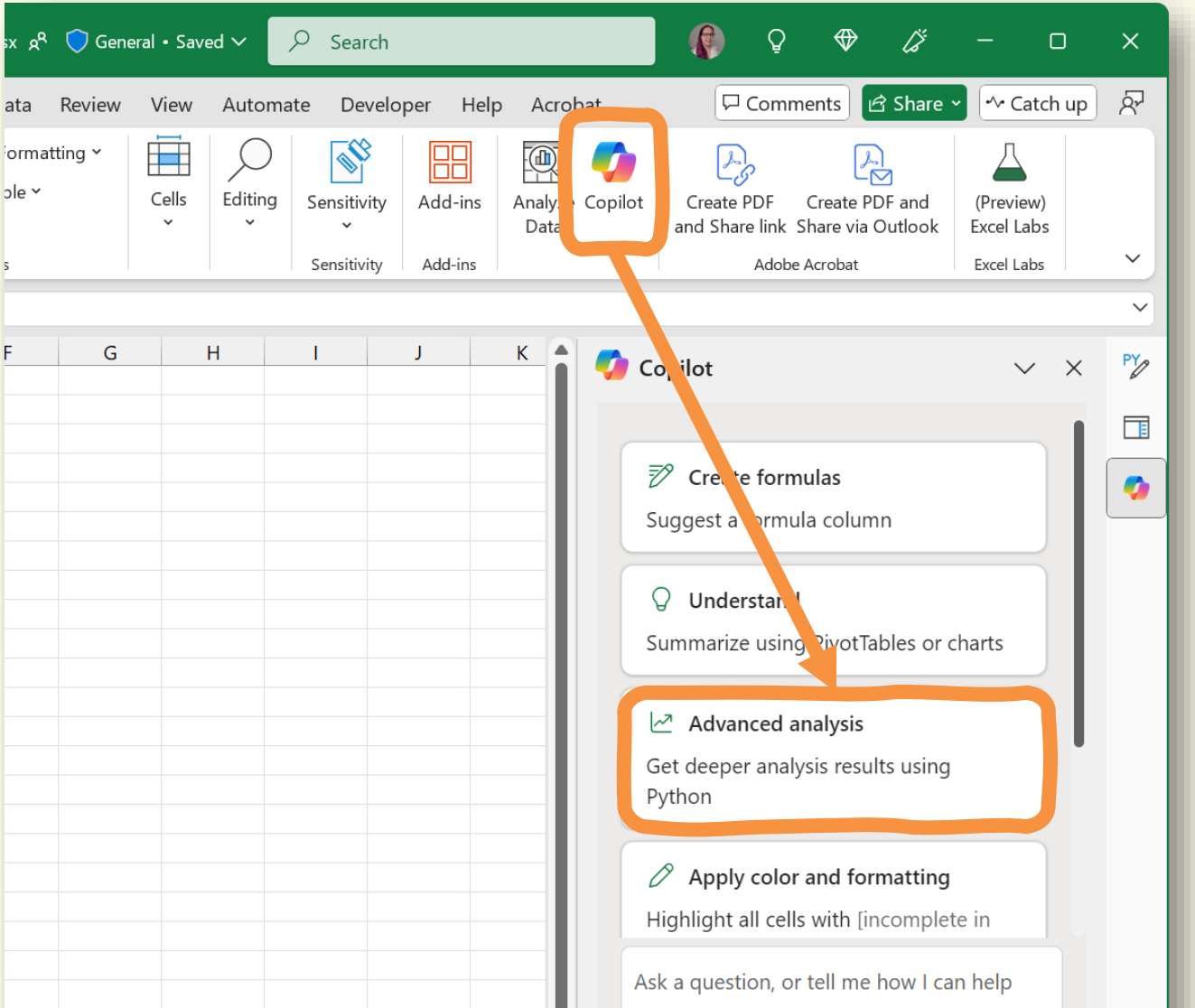
…even with pretty vague directions



Ask for summaries of data

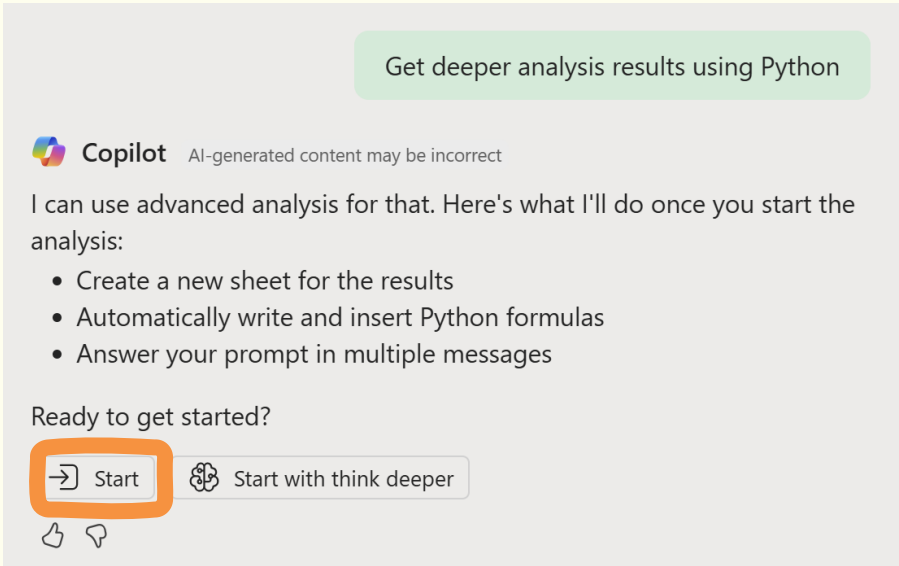


How do I start Advanced Analysis?

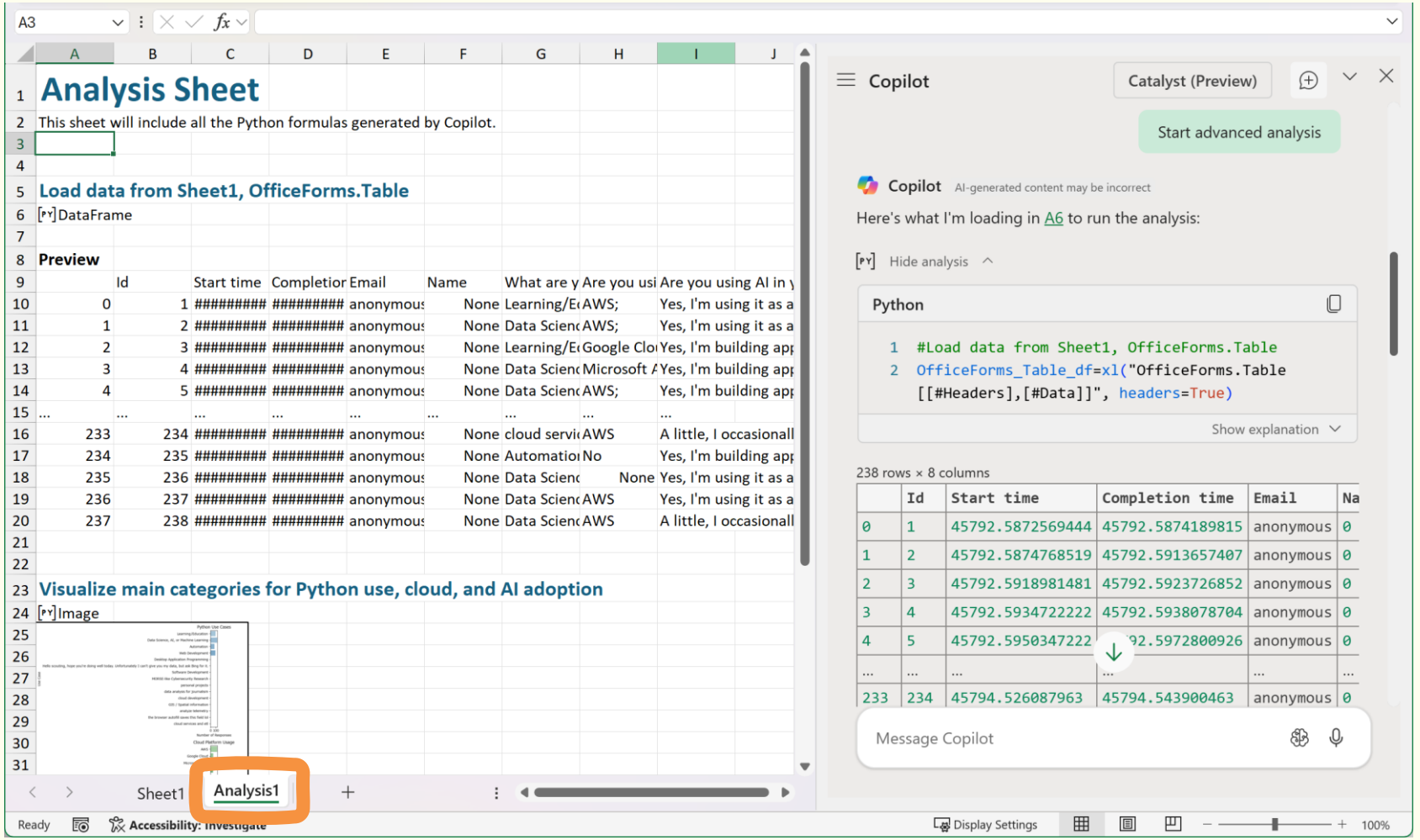


Or start with a prompt like:

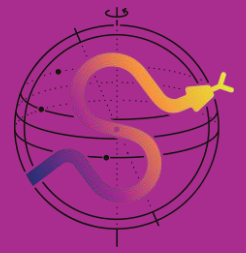
- "Analyze this data using Python"
- "Run advanced analysis using Python"



The Analysis sheet:



Let’s look at a demo!

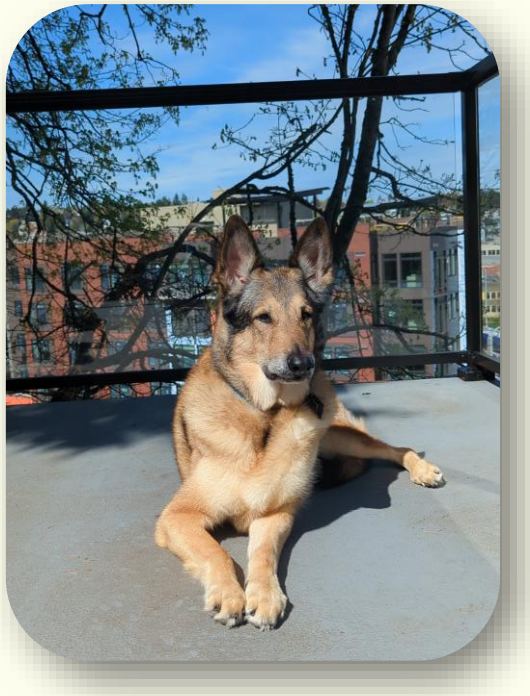


Exercise: Discover your Data

1. Using a data set of your own or one provided in the tutorial repo, use Advanced Analysis to help you make some sort of visualization of the data.
2. Then use the Copilot chat to learn more about the data by trying out some of it's suggested prompts.

aka.ms/scipy-sheets

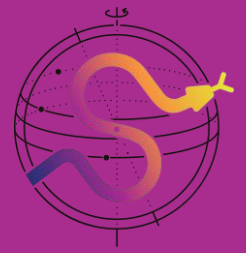
6. Spreadsheets with pandas



Pandas ↔️ Spreadsheet terms

Pandas	Spreadsheet
DataFrame	worksheet
Series	column
Index	row headings
row	row
NaN	empty cell

Let’s look at some code!

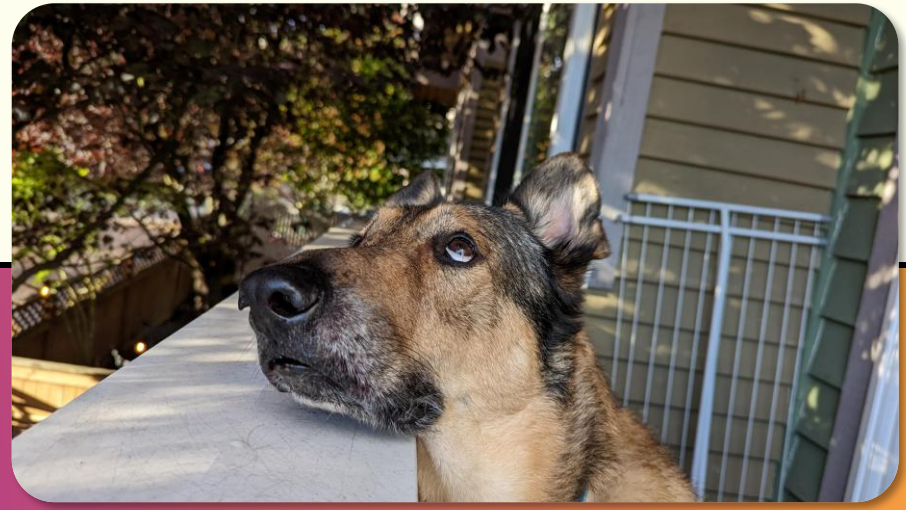


Exercise: Pandas and Sheets

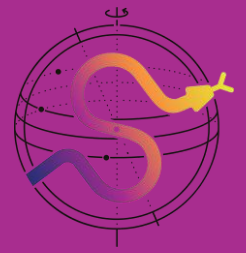
1. Take the `fish.csv` sample data and use `pandas` to load into Python (Jupyter notebook or script)
2. Clean the data in a way you think would be helpful (maybe parse columns, change value types, etc.). You can also ask Copilot in VS Code for suggestions.
3. Export the cleaned data back to a spreadsheet with `pandas`.

aka.ms/scipy-sheets

7. Build spreadsheets with `openpyxl`



Let’s look at some code!

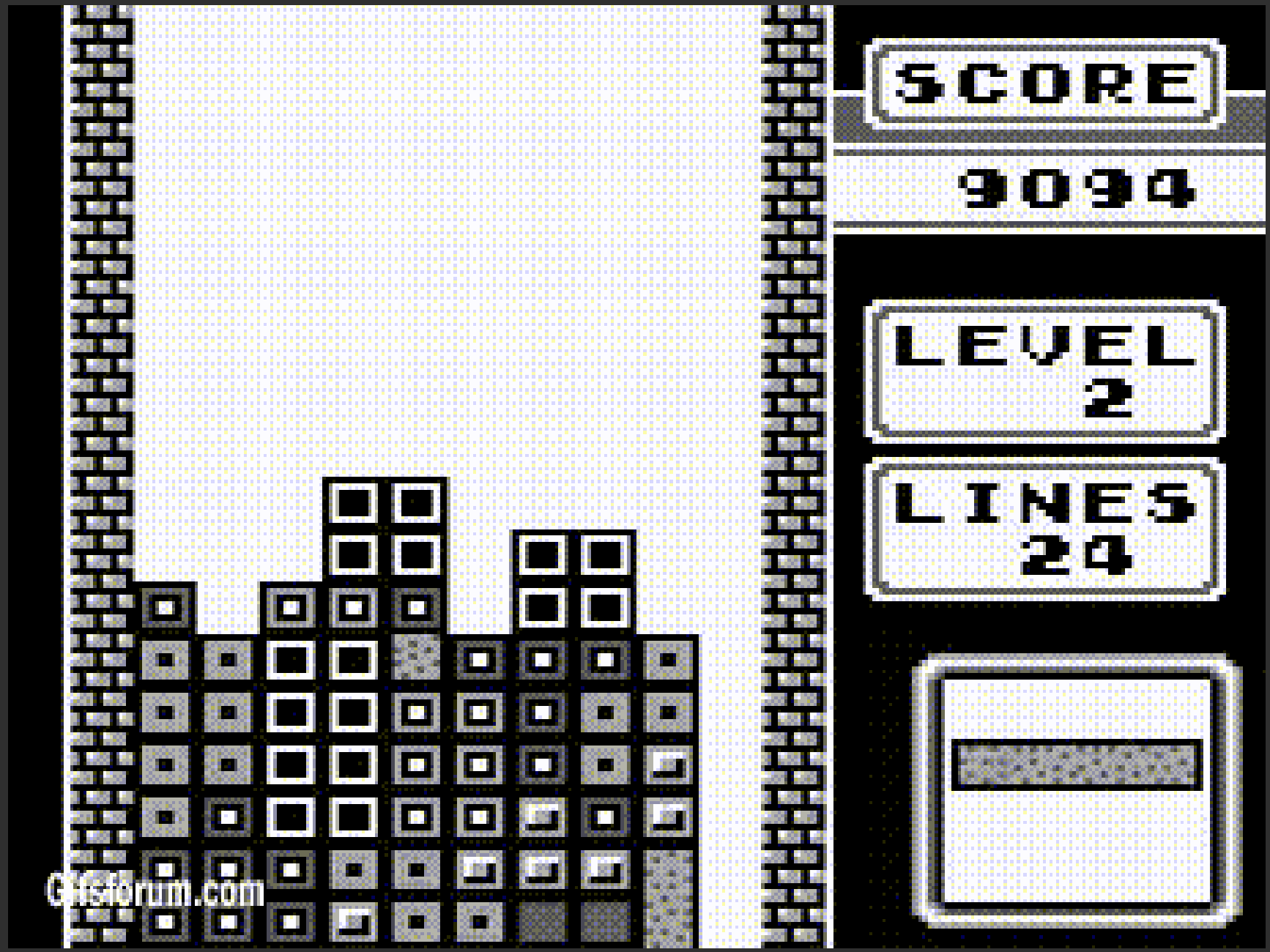


Exercise: Plots

1. Use your spreadsheet viewer of choice to add a plot to the fish spreadsheet from the `pandas` exercise.
2. Parse the spreadsheet with `openpyxl` and extract the image of the plot with Python.

aka.ms/scipy-sheets

=SUM(everything)


SCORE
9094
LEVEL
2
LINES
24

SCORE
12
EXCEL ESPORTS.
YES IT'S A THING

Exercise: Excel E-Sports challenge

In the tutorial repo, download/open the file `excel-esports-tetris.xlsx` for instructions.

You can either work in Python and export your answers in a spreadsheet or use Python in Excel/Anaconda Code to solve it. Anyone who finishes will get some limited-edition swag!

aka.ms/scipy-sheets

Wrap-up

- Load and work with spreadsheets in Python
- Run Python directly in Excel
- Use AI tools like Advanced Analysis in Excel
- Use Anaconda Toolbox and Code in Excel

What did we just do?

Find Sarah at:
@crazy4pi314 | sckaiser.com

Find Jim at:
@Jim22k on GitHub

Come hangout on our Discord:
https://aka.ms/azureai/discord

Ask the PyEx devs questions on GitHub:
https://github.com/microsoft/python-in-excel/issues

aka.ms/scipy-sheets

Thank You!