AI-Guided SIR Modeling Hackathon

Learning epidemic modeling with Python + ChatGPT

A beginner-friendly, hands-on workshop using real outbreak data

AMMnet Hackathon
March 12, 2026

What We Will Do Today

- Understand the basic idea of the SIR model
- Use ChatGPT to help write and explain code
- Simulate an epidemic outbreak
- Fit model parameters to real data
- Compare simple fitting methods
- Understand why validation matters
- No prior knowledge of epidemiology or LLMs is required

What Is ChatGPT?

- ChatGPT is a type of AI called an LLM (Large Language Model)
- It can understand prompts, generate text, write code, and explain ideas
- A simple mental model: it is a very powerful text-and-code assistant

How an LLM Works (Simple Version)

- It reads your prompt
- It predicts what response is likely to come next
- It generates text based on patterns learned from large amounts of data
- Good at: drafting code, explaining ideas, summarizing, suggesting next steps
- Not good at: guaranteeing correctness or replacing scientific judgment

How We Will Use ChatGPT Today

- We will use ChatGPT as a co-pilot for scientific work
- It can help write Python code, explain models, suggest fitting approaches, debug mistakes, and summarize results
- We still must check assumptions, inspect plots, verify units, and judge whether results make sense
- Key rule: ChatGPT is an assistant, not an oracle

The Five Components of an Effective AI Prompt

1. **Persona:** The specific role or identity the AI should assume.

2. **Context:** The background information, data, or parameters relevant to the task.

3. **Task:** The specific action or objective you want the AI to execute.

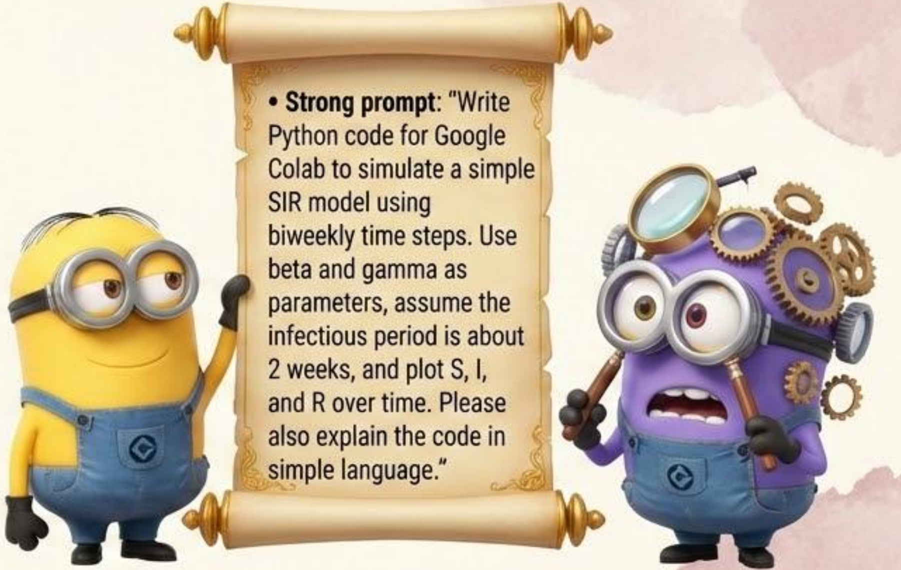
4. **Constraints/Verification:** Any limitations, assumptions, required outputs, units, or tools the AI must adhere to.

5. **Output Style:** The desired format for the response (e.g., code, summary, table, explanation, plot).

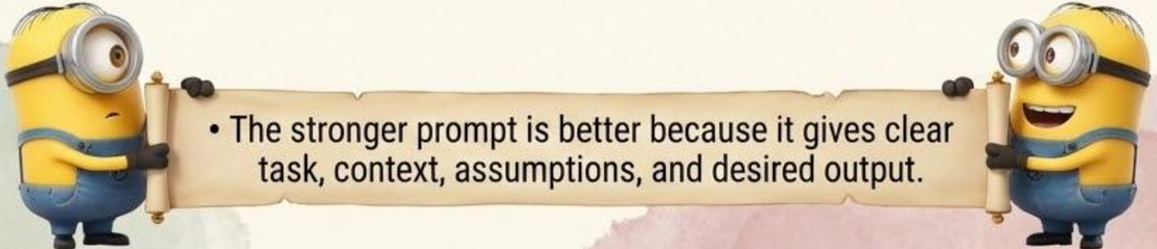
Formula for Prompting: Persona + Context + Task + Constraints/Verification + Output Style

Weak Prompt vs Strong Prompt

- Weak prompt: “Write SIR code.”



- **Strong prompt**: “Write Python code for Google Colab to simulate a simple SIR model using biweekly time steps. Use beta and gamma as parameters, assume the infectious period is about 2 weeks, and plot S, I, and R over time. Please also explain the code in simple language.”



- The stronger prompt is better because it gives clear task, context, assumptions, and desired output.

Prompting Rules for This Hackathon

- Be specific about units

- Say what the data represent

- Ask for simple readable code

- Ask ChatGPT to explain assumptions

- Ask for plots and sanity checks

- Revise the prompt if needed

- Ask for the next step, not the entire project at once

Common Prompting Mistakes

- Being too vague

- Forgetting units

- Not describing the data clearly

- Asking for too much at once

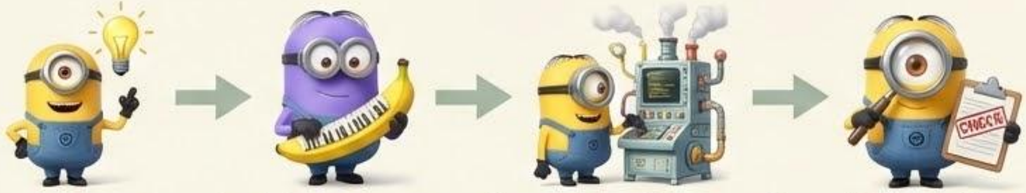
- Copying code without testing it

- Assuming the first answer is correct

- Rule for today: every generated result should be checked

How Today's Workflow Works

- We will repeat the **same learning cycle**:



Concept → Prompt → Code → Check

- **For each step:** ⚙️ learn one modeling idea, 💬 ask ChatGPT for help, run the code in Colab, 🖥️ inspect the result, 🛠️ and improve it if needed

Part 1 —
Meet the Data

The Dataset We Will Use

- Disease: Measles
- Location: Niamey, Niger
- Data type: Biweekly case counts
- Communities: A, B, C
- Important modeling detail: time is measured in biweeks
- The infectious period is approximately 2 weeks

First Task: Explore the Data

- Load the dataset
- Inspect its columns
- Make a simple plot
- Look for the outbreak pattern
- A model is only useful if we understand what was actually measured

ChatGPT in Action: Data Exploration

- Ask ChatGPT to help load the CSV
- Ask it to describe the columns
- Ask it to plot case counts over time
- Ask it to explain what we are seeing
- We must **verify**: column names, time variable, counts plotted correctly, no confusion between groups

Part 2 — First Modeling Insight

Early Outbreak Growth

- At the start of an outbreak, cases often rise quickly
- This early rise tells us something about transmission
- We can use early growth to get a rough estimate of how contagious the outbreak is
- This gives an early approximation of R_0

What Is R_0?

- R_0 (basic reproduction number) is the average number of people infected by one infected person in a fully susceptible population

- $R_0 > 1$ means the outbreak can grow

- $R_0 < 1$ means the outbreak tends to fade out

ChatGPT in Action: Estimating Early Growth

- Use ChatGPT to help fit a simple early growth trend

- Use ChatGPT to calculate a rough R_0 estimate

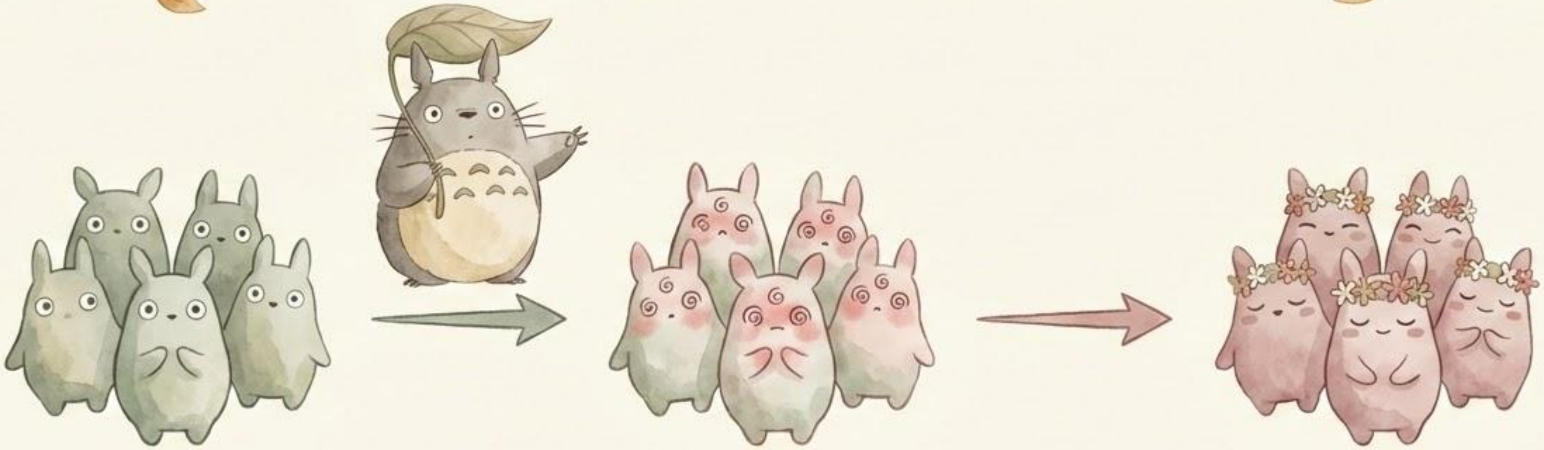
- Use ChatGPT to explain what the estimate means

- ChatGPT helps with implementation, but scientific interpretation still depends on us

Part 3 — Build the SIR Model

Constructing the Simulation

The Core SIR Idea



S = Susceptible

I = Infected

R = Recovered

People move from S to I through infection

People move from I to R through recovery

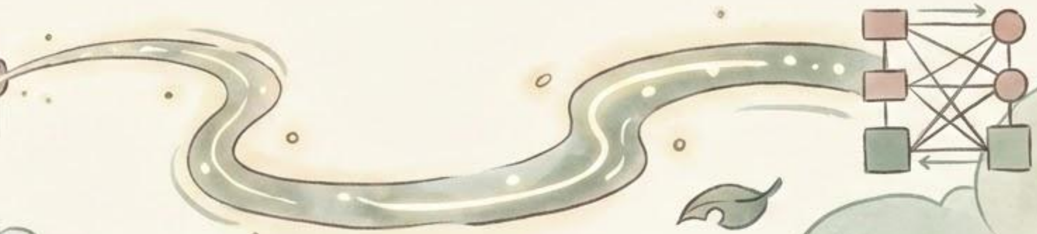
The Two Main Parameters

- β (beta) = transmission rate
- γ (gamma) = recovery rate
- These control how quickly infection spreads and how quickly infected people recover
- Because the data are biweekly, the model also uses biweekly time units

ChatGPT in Action: Writing the SIR Simulation

- Ask ChatGPT to write Python code for the SIR model
- Ask it to simulate the model over time
- Ask it to plot S, I, and R
- We check: total population stays consistent, infected curve rises and falls sensibly, units align with the dataset

Part 4 — Connecting Model to Data



A Critical Modeling Question

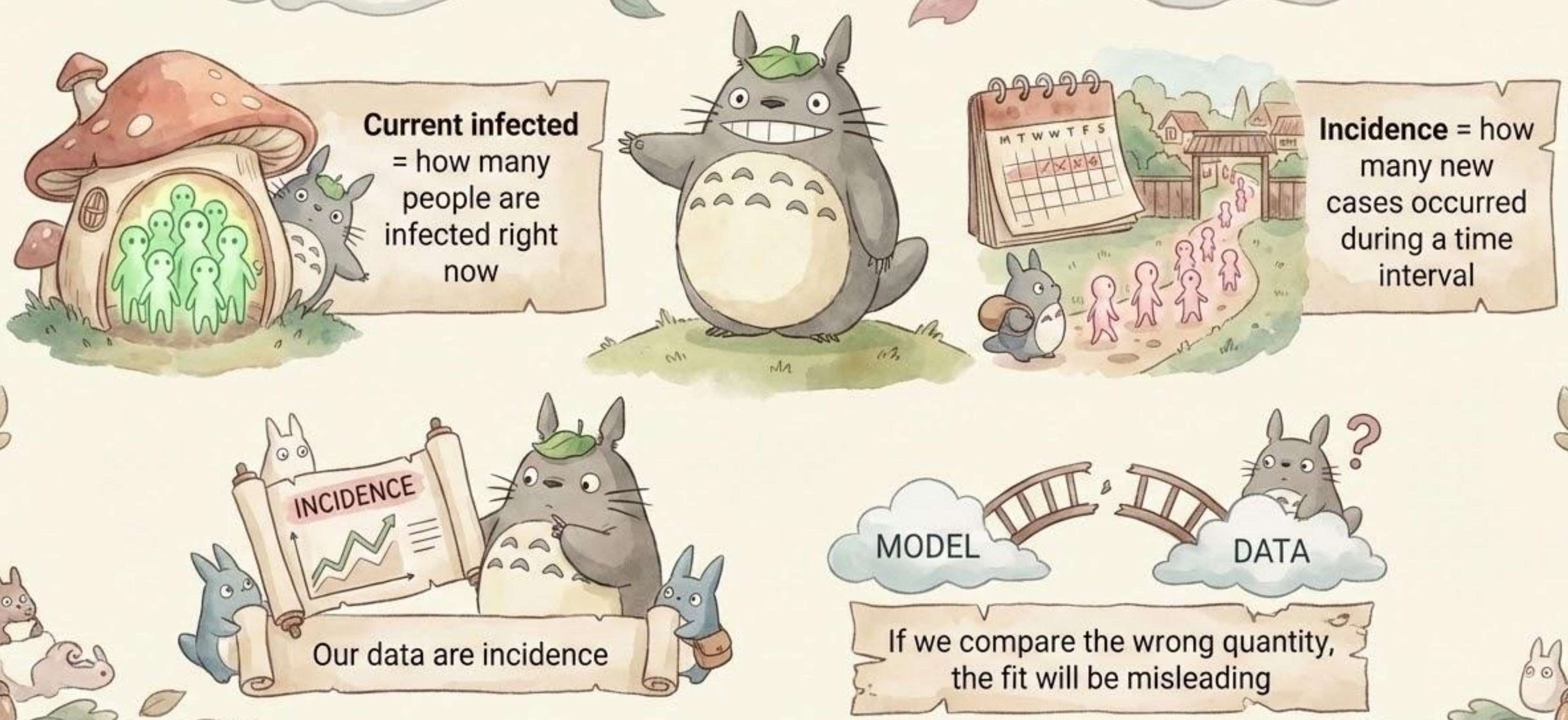
- The SIR model tracks how many people are currently in each compartment

- But the data give new cases per biweek

- These are not the same thing

- We must convert model output into something comparable to the observed data

Incidence vs Current Infected



ChatGPT in Action: Mapping Model Output to Cases

- Ask ChatGPT to convert model output into predicted biweekly cases
- Ask it to explain how that mapping works
- Ask it to produce a plot comparing predicted and observed values
- We verify: correct scale, same time units, same definition of case

Part 5 — Fit the Model to Data

Why We Need Parameter Fitting

- Until now, we have chosen parameters manually
- In real analysis, we want to estimate parameters from data
- That means finding values that make the model match the outbreak curve as well as possible

Method 1: Least Squares

Compare observed cases and predicted cases

Measure their difference

Choose parameters that make the total error small

This is intuitive and a good first fitting method

ChatGPT in Action: Least Squares Fitting

- Ask ChatGPT to define the objective function
- Ask it to run an optimizer
- Ask it to choose starting values
- Ask it to plot observed vs fitted cases
- We check whether the fitted curve matches the trend and whether parameter values are reasonable

Method 2: Poisson Likelihood

- Case counts are discrete, non-negative, and noisy
- A count-based statistical model is often more appropriate
- The Poisson likelihood assumes that the observed case counts are random realizations around the model-predicted incidence, scaled by a reporting (ascertainment) fraction.

ChatGPT in Action: Likelihood-Based Fitting

- Ask ChatGPT to write the Poisson likelihood
- Ask it to explain how it differs from least squares
- Ask it to explain how it differs from least squares
- Different fitting methods reflect different assumptions about the data

Compare the Two Approaches

- Compare least squares fit and Poisson likelihood fit
- Do they give similar trends?
- Do they produce different parameter values?
- Which assumptions seem more appropriate for count data?

Part 6 — Uncertainty and Interpretation

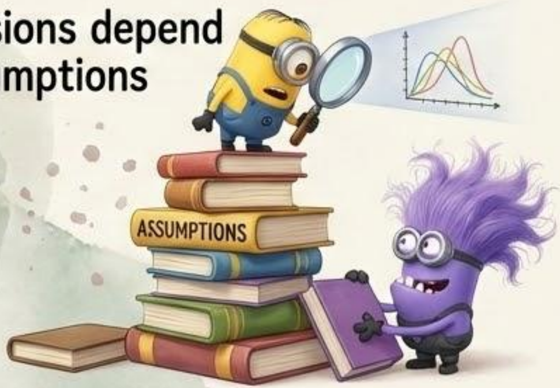
Why Uncertainty Matters

- A model fit is not just one perfect answer

- Multiple parameter values may fit the data

- Some parameters are more certain than others

- Conclusions depend on assumptions



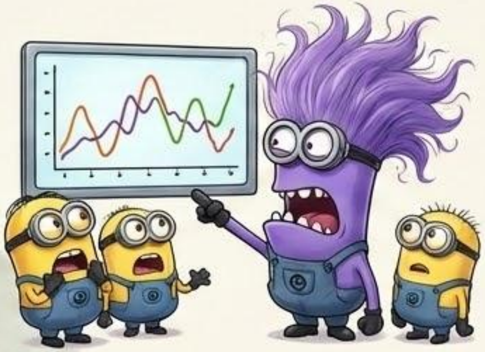
- We need to think in ranges, not just single numbers

Simple Uncertainty: Profile Likelihood

Vary one parameter

Refit the others

See how the fit changes

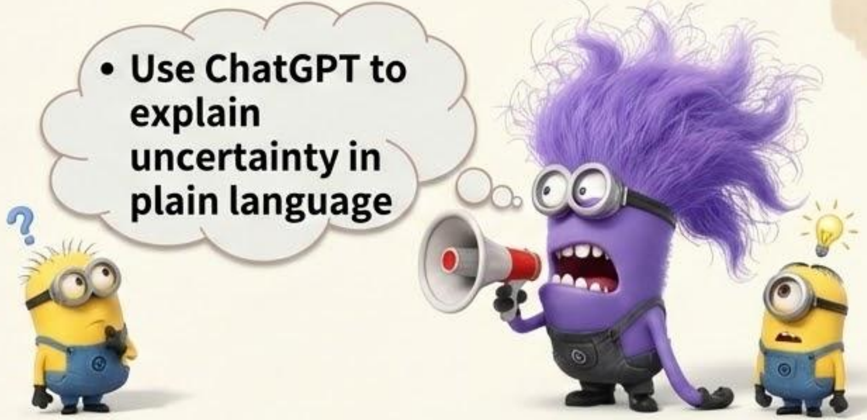


This shows which values are plausible and how sensitive the model is

ChatGPT in Action: Interpreting Results

- Use ChatGPT to summarize fitted parameters

- Use ChatGPT to explain uncertainty in plain language



- Use ChatGPT to describe limitations of the model

- Use ChatGPT to generate a short written report

Part 7 — Wrap-Up






What You Built Today

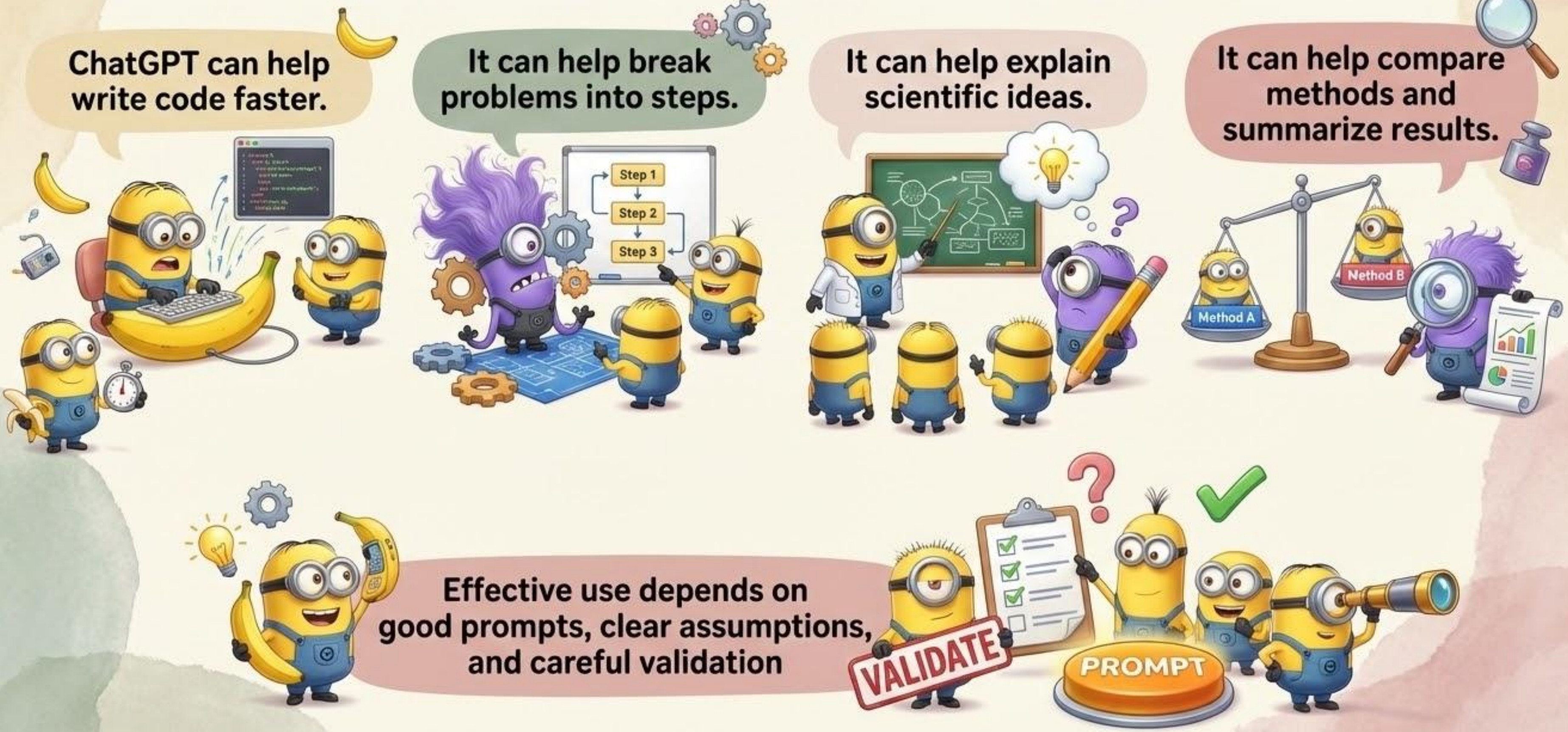
- A working SIR model

- At least one fitted parameter set

- An observed vs fitted plot

A basic understanding of uncertainty

What You Learned About AI



Key Takeaways

The SIR model gives a simple mechanistic view of epidemic spread

Inference connects that model to real outbreak data.

ChatGPT can speed up scientific work if you use it carefully

Best habit: prompt clearly, check units, plot often, validate everything.

Next Steps / Extensions

SEIR models
Stochastic epidemic models
Time-varying transmission

Bayesian inference

Richer outbreak datasets

You leave with a workflow you can reuse and extend

Thank you!