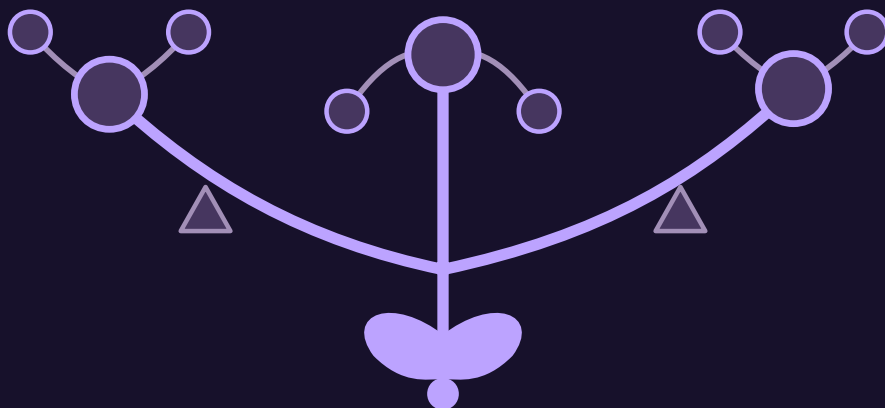




PRACTICAL AI

Create Useful Synthetic Evaluation Datasets

Design scenario families, reliable labels, and honest coverage claims



Sanjaay Gowda T S

FREE SAMPLE / INTRODUCTION + CHAPTER 1 / EDITION 1.0.0

sanjaaygowdats.ai

About this free sample

Create Useful Synthetic Evaluation Datasets

Sanjaay Gowda T S

First edition 1.0.0, 2026-10-13

Copyright 2026 Sanjaay Gowda T S. This is a labelled free sample, not the complete ebook. You may share this sample unchanged. It contains the cover, sample permissions, contents, introduction, Chapter 1 and a closing product link. The remaining chapters, worked solutions and companion files are not included. The full ebook text, diagrams and artwork have separate single-reader terms; this sample permission does not authorize sharing the full purchased PDF or EPUB.

The companion source code, tests and fictional data are separately MIT licensed. You may adapt or share that project with its included licence notice. MIT does not apply to the ebook text or book artwork. The books and companion projects have no technical activation requirement or buyer-specific watermark.

This first edition was prepared with AI assistance. The case studies are fictional teaching examples. Any measured outputs come from the supplied examples, not production customer data. The teaching scope, assumptions and limitations are explained in the chapters. Independent human technical, editorial and assistive-technology reader review is pending. Automated tests and accessibility checks do not establish certification or suitability for every use.

Run examples in a separate working folder using the supplied synthetic data. Review inputs and calculations before adapting the techniques to real operations. Model software, service access, subscriptions, ongoing updates and consulting are not included.

Author website: sanjaaygowdats.ai

Ebook terms: sanjaaygowdats.ai/ebooks/license

Inside this free sample

Start here.....	4
1. Choose the decision the dataset tests	5
Continue with the full guide	6

This free sample contains the introduction and Chapter 1 only. The closing link opens the complete guide. The companion project and worked solutions are supplied with the complete ebook.

Start here

A useful synthetic evaluation dataset makes a narrow claim: these designed cases test these specified behaviors under these conditions. It does not become representative of real users merely because it contains many rows. A thousand paraphrases of the same easy scenario may provide less diagnostic value than twenty carefully chosen decision boundaries.

This book constructs a fictional parcel-support dataset. Each family has explicit parcel facts, several handwritten wording variants, and an expected action derived from a small policy. The companion validates the facts, rejects normalized duplicate text, assigns an entire family to one partition, and emits deterministic JSON. It does not call a model or use real customer records.

The fictional policy has four actions. Delivered parcels receive `confirm_delivery`. Unknown status receives `ask_tracking`. An in-transit parcel fewer than three days late receives `wait`. An in-transit parcel at least three days late receives `escalate`. These are teaching rules, not a courier's actual policy. The input text is paired with structured facts; the expected action follows those facts rather than requiring an undocumented inference from the wording.

You need Python 3.10 or later and basic JSON familiarity. Run `python main.py --input example-input.json` inside project, then run `python -m unittest discover -s tests -v`. The expected output and eleven tests show exactly what the generator does. No synthetic row should be mistaken for an actual observed user interaction or a model-generated result.

By the end, you will have a scenario matrix, a family-based split design, a label-review process, and a release note that states where the dataset's evidence stops. The goal is an inspectable evaluation instrument, not a large file with an impressive name.

1. Choose the decision the dataset tests

Write an observable task

Begin with a decision that can be judged from available information. "Provide excellent parcel support" is too broad. "Choose one of four actions from a status and days-late value" is narrow enough to test. The second task can still matter: it checks whether an application respects a policy boundary instead of improvising an action.

Specify the input package. In this project, each evaluated case would provide a user wording variant and the structured parcel facts. If a future evaluation supplies only the text, it changes the task. Some variants do not state the status or exact lateness, so a text-only system cannot reliably derive the intended action. An evaluation must not withhold the evidence that its label assumes.

Write the expected behavior in a compact policy table:

Parcel facts	Expected action	Why the case exists
Delivered, zero days late	confirm_delivery	Positive status path
Unknown, zero days late	ask_tracking	Missing-information path
In transit, zero to two days late	wait	Below escalation boundary
In transit, three or more days late	escalate	At or above boundary

The fixture rejects nonzero lateness for delivered or unknown status. That restriction is a deliberate modeling simplification. A real parcel system may need a different representation, such as days beyond promise date even after delivery. If you adopt that broader representation, revise the policy and labels together rather than silently accepting contradictory facts.

Separate target behavior from output style

An action label evaluates a decision. A friendly sentence evaluates wording. A citation evaluates evidence attribution. These are different targets. This dataset labels the action only. It cannot establish politeness, factual explanation quality, privacy handling, or whether an escalation actually reached a human operator.

You can add other dimensions later, but name each and define its evidence. For example, a response may choose `escalate` correctly while falsely claiming that a ticket has already been filed. A decision-only metric would miss that defect. Keeping the scope explicit helps the evaluation owner see what additional tests are needed.

State the intended use and exclusions

Use this suite for early policy checks, parser tests, and regression cases around known boundaries. Do not use it to estimate production customer satisfaction or the prevalence of delayed parcels. The examples were selected by design and do not have a real traffic sampling process.

The distinction matters when presenting a result. "Passed all five supplied synthetic action cases" is supported by a correctly executed evaluation. "Handles parcel support reliably" requires a much broader evidence base. Your dataset card should include intended use before any score can be published.

Exercise 1

Write an observable task for a fictional library-renewal assistant. Define the input facts, the allowed action labels, and two capabilities the dataset will not evaluate.

Continue with the full guide

This free sample ends after the introduction and Chapter 1. The complete guide contains nine teaching chapters, worked exercise solutions, primary references and a separate Python companion project. The full PDF, reflowable EPUB and project files are supplied together.

Use the complete guide to work through the remaining exercises and inspect the supplied examples. The companion code and fictional data are separately MIT licensed; the complete ebook has the single-reader terms described on its product page.

The examples are teaching fixtures, not production customer results. Independent human technical, editorial and assistive-technology reader review remains pending. Software access, subscriptions, consulting and guaranteed results are not included.

[View the complete guide and purchase options.](#)

You may share this labelled free sample unchanged. Keep the full purchased PDF and EPUB private. Thank you for reading.