

Data Science, AI, and Machine Learning in Public Health using R

Part 1

January 2026

Presented By: Wronski Associates

Introduction: Directing AI Use in the Federal Government

MOVING BARRIERS TO AMERICAN LEADERSHIP IN ARTIFICIAL INTELLIGENCE

The White House | January 23, 2025

CDC's Vision for Using Artificial Intelligence in Public Health



Public Health
AUG. 22, 2025

CDC is committed to using artificial intelligence/machine learning for innovation, operational efficiency, and fighting infectious disease. CDC's artificial intelligence innovation approach includes investment areas, partnerships, workforce readiness, and guidance.

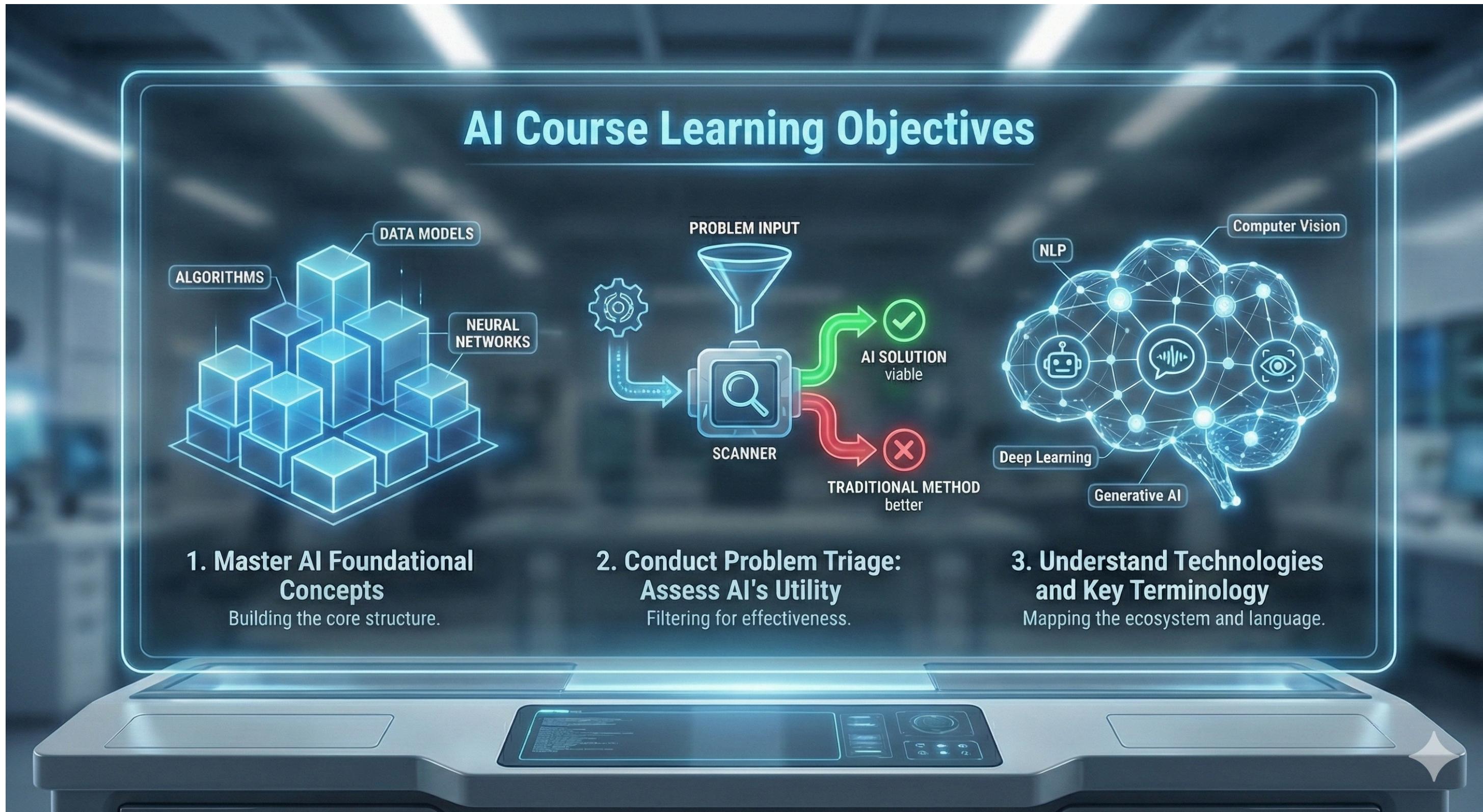
- **M-25-21:** "Accelerating Federal Use of AI through Innovation, Governance, and Public Trust" (April 3, 2025)
- **M-25-22:** "Driving Efficient Acquisition of Artificial Intelligence in Government" (April 3, 2025)
- **Executive Order 14319 -** "Preventing Woke AI in the Federal Government" (July 23, 2025)
- **America's AI Action Plan** (July 2025)

Source: The White House

To do what, exactly?



That is what we are here to figure out

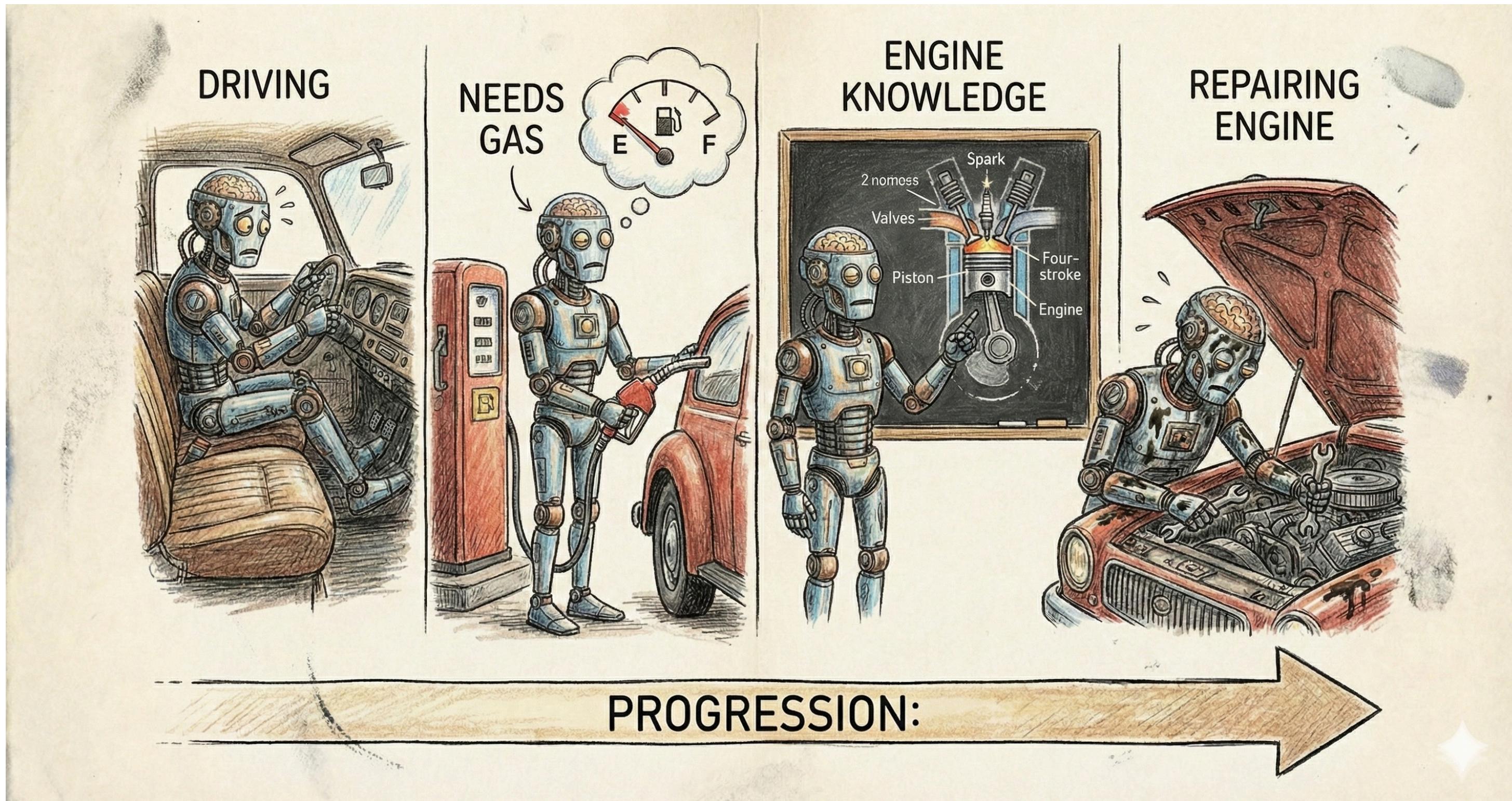


Prompt: Create an image related to AI. The image presents the learning objectives for a course. The learning objectives are: Learning AI Foundational Concepts, Problem Triage - Can AI be useful?, and Technologies and Terminology.

3 minute assignment

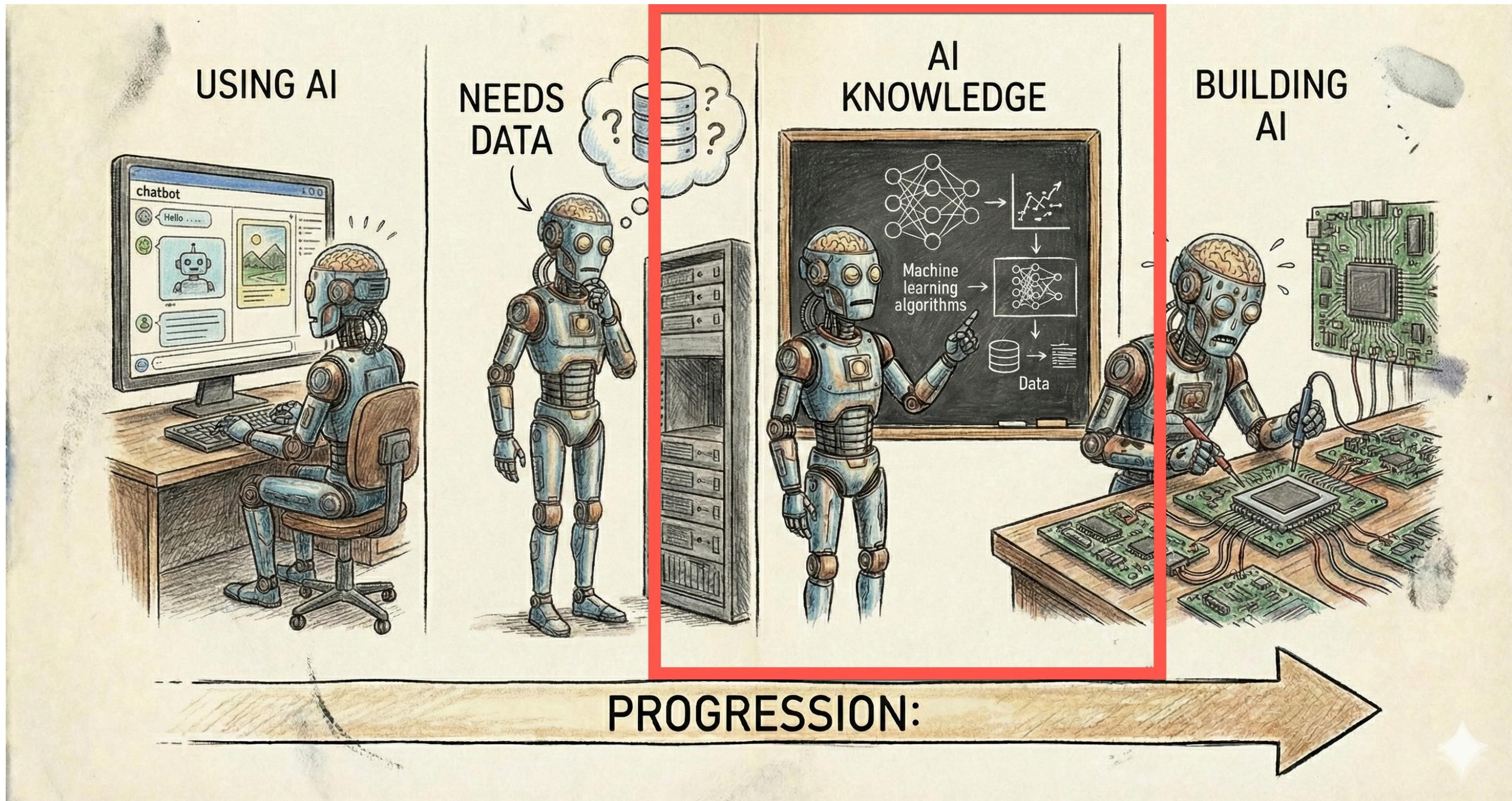
Think of a problem, opportunity, project that you have that could, potentially, be solved using AI.

Level of knowledge



Prompt: Use the robot, hand drawn style. Create an image of images that portray these ideas: driving, knowing the car needs gas, knowing, at a high level, how a combustion engine works, repairing a car engine. indicate there is a progression from the first to the last.

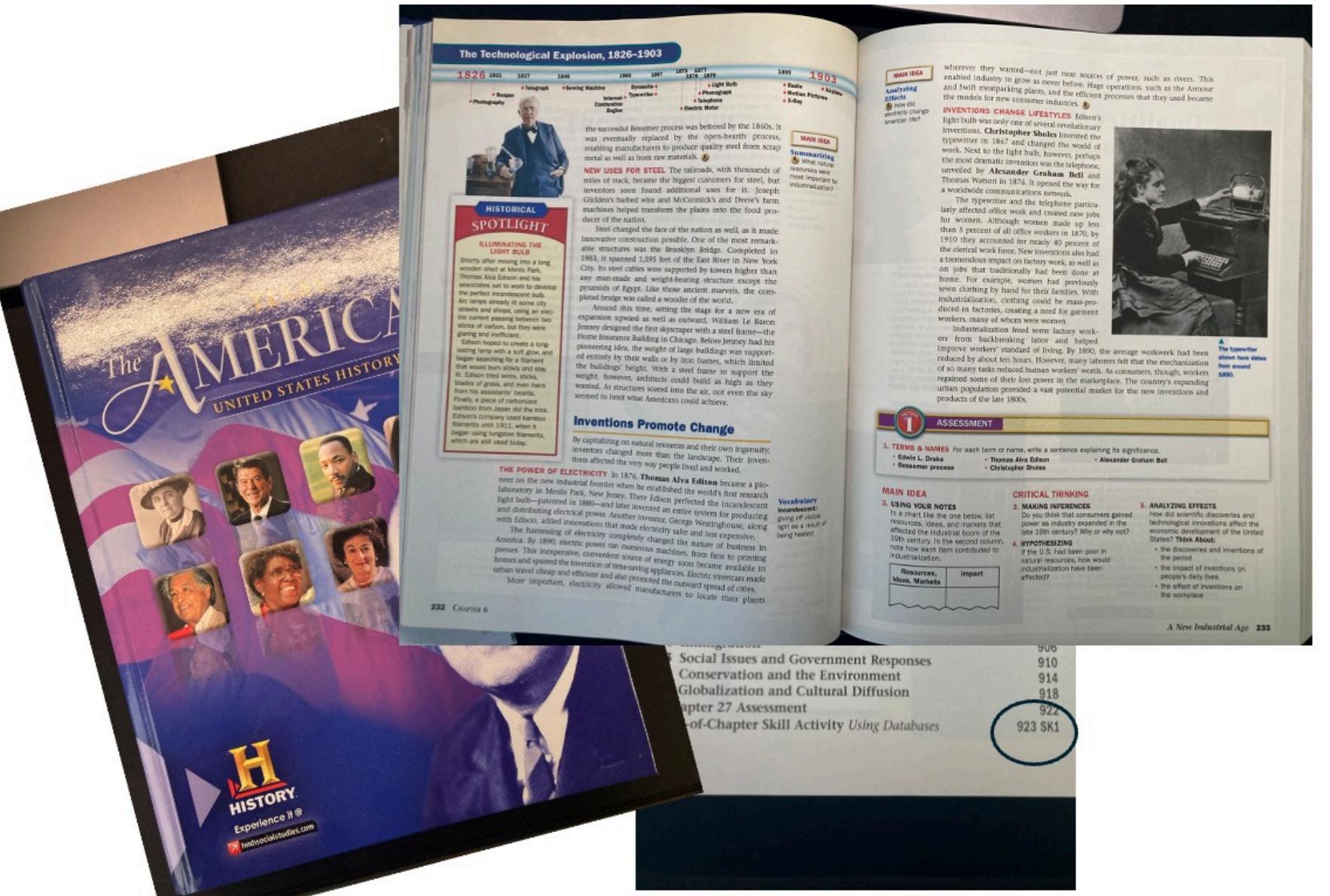
Level of knowledge



Prompt: Use the image you just created as a model to represent knowledge of AI

A quick working example

- I have 20 secondary school history textbooks.
- Each book is broken down into chapters. Each chapter has multiple sections.
- Each book has about 90 sections totaling around 1000 pages.
- I only want to read the sections that are pertinent to my study.



Creating a Computational Filter - The Input

PDF Topic Evaluator

Upload one or more PDFs and check if a specific topic is addressed.

🔍 Topic to evaluate (e.g. 'test-driven development')

national identity

🧠 Considerations to Analyze (one per line)

Values

National myths or symbols

Civic assumptions

Upload PDFs



Drag and drop files here

Limit 200MB per file • PDF

Browse files



Chapter 4 Section 4 Reading.pdf 4.3MB

X



Chapter 4 Section 3 Reading.pdf 7.0MB

X



Chapter 4 Section 2 Reading.pdf 2.3MB

X

Showing page 1 of 2

< >

Evaluate PDFs

Creating a Computational Filter - The Output

▼ Chapter 4 Section 4 Reading.pdf

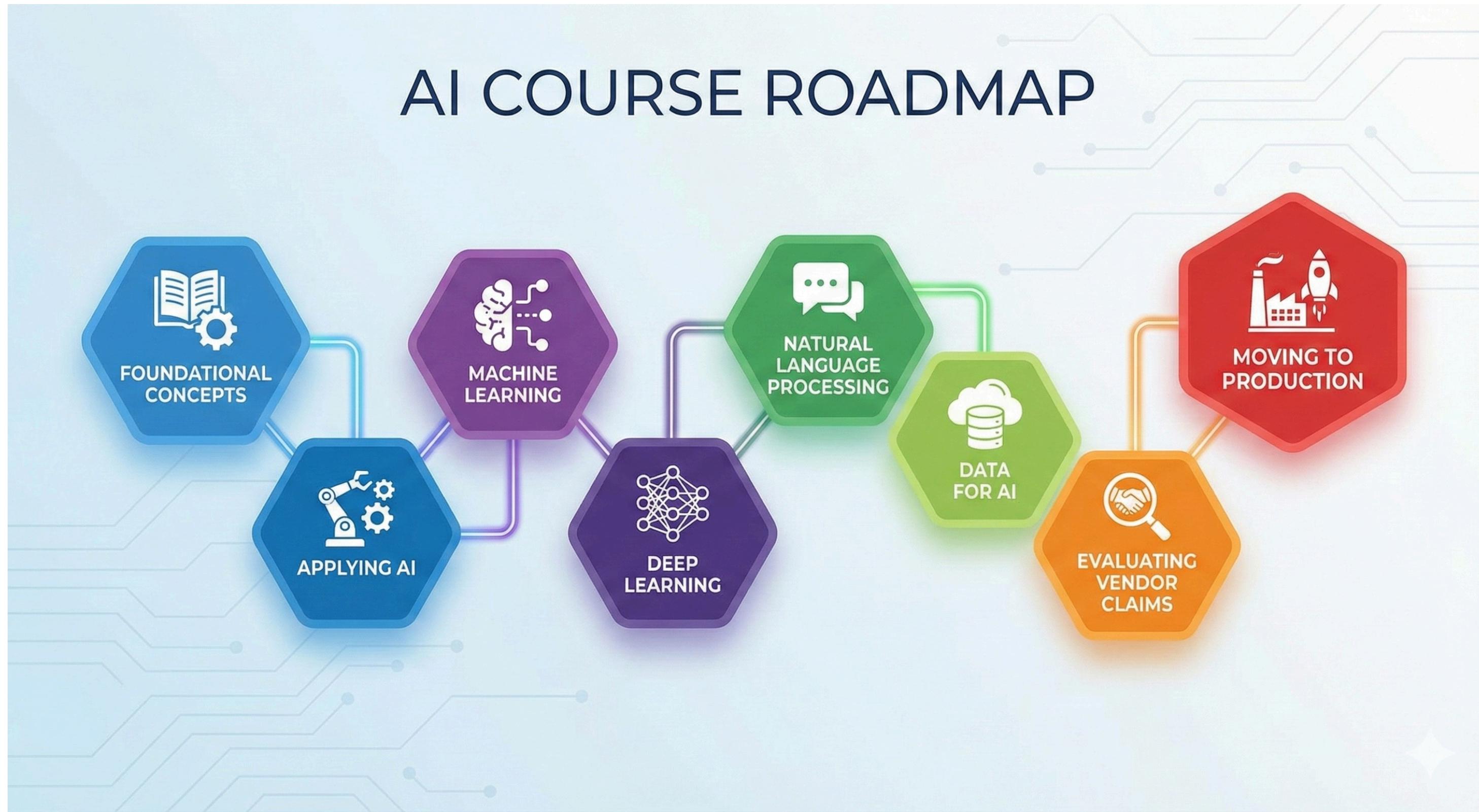
1. **Determination:** Yes, the document addresses the topic of "national identity."

2. **Analysis:**

- **Values:** The document reflects values such as liberty, courage, and patriotism. These are evident in the passages that discuss the contributions of women and soldiers during the war, emphasizing their dedication to the public good and the fight for freedom. The emphasis on liberty and resistance against tyranny are core values that contribute to the formation of a national identity.
- **National Myths or Symbols:** The document alludes to the mythic narrative of the American Revolution as a struggle for freedom and independence. The mention of the Treaty of Paris and the return of soldiers to their homes symbolizes the birth of a new nation and the establishment of its borders, which are key elements in the national identity of the United States.
- **Civic Assumptions:** The document assumes a shared commitment to the ideals of independence and self-governance. The references to the formation of a new government and the collective efforts of soldiers and civilians highlight the civic responsibility and unity necessary for nation-building.
- **Direct Quotes:**
 - "Born for liberty, disdaining to bear the irons of a tyrannic Government, we associate ourselves..."
 - "The courage of soldiers and civilians had made America's victory possible."

These elements collectively contribute to the construction and reinforcement of a national identity centered around the values and narratives of the American Revolution.

Course roadmap

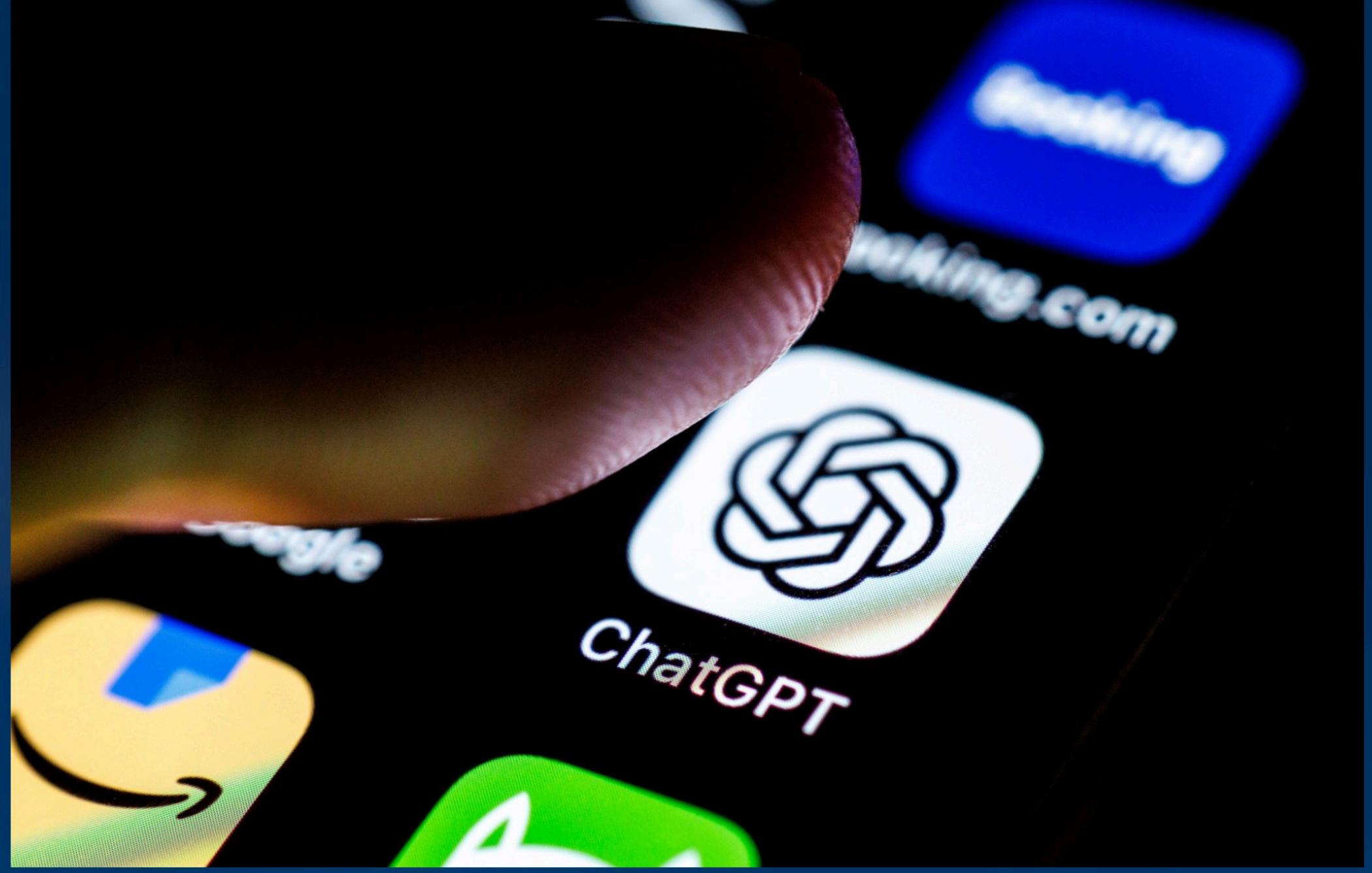


Prompt: Create an image related to AI. The image presents the topics, or roadmap, for a course. The topics to be covered in order are: Foundational concepts, applying AI, Machine Learning, Deep Learning, Natural Language Processing, Data for AI Evaluating Vendor Claims, Moving to Production.

RESEARCH

Is the politicization of generative AI inevitable?

Valerie Wirtschafter and Nitya Nadgir
October 16, 2025



<https://www.brookings.edu/articles/is-the-politicization-of-generative-ai-inevitable/>

Foundational Concepts: What is AI?

Artificial intelligence (AI) is a machine-based system that can, for a given set of human-defined objectives, make predictions, recommendations, or decisions influencing real or virtual environments. AI systems use machine- and human-based inputs to perceive real and virtual environments; abstract such perceptions into models through analysis in an automated manner; and use model inference to formulate options for information or action. ([15 U.S.C. 9401\(3\)](#)).

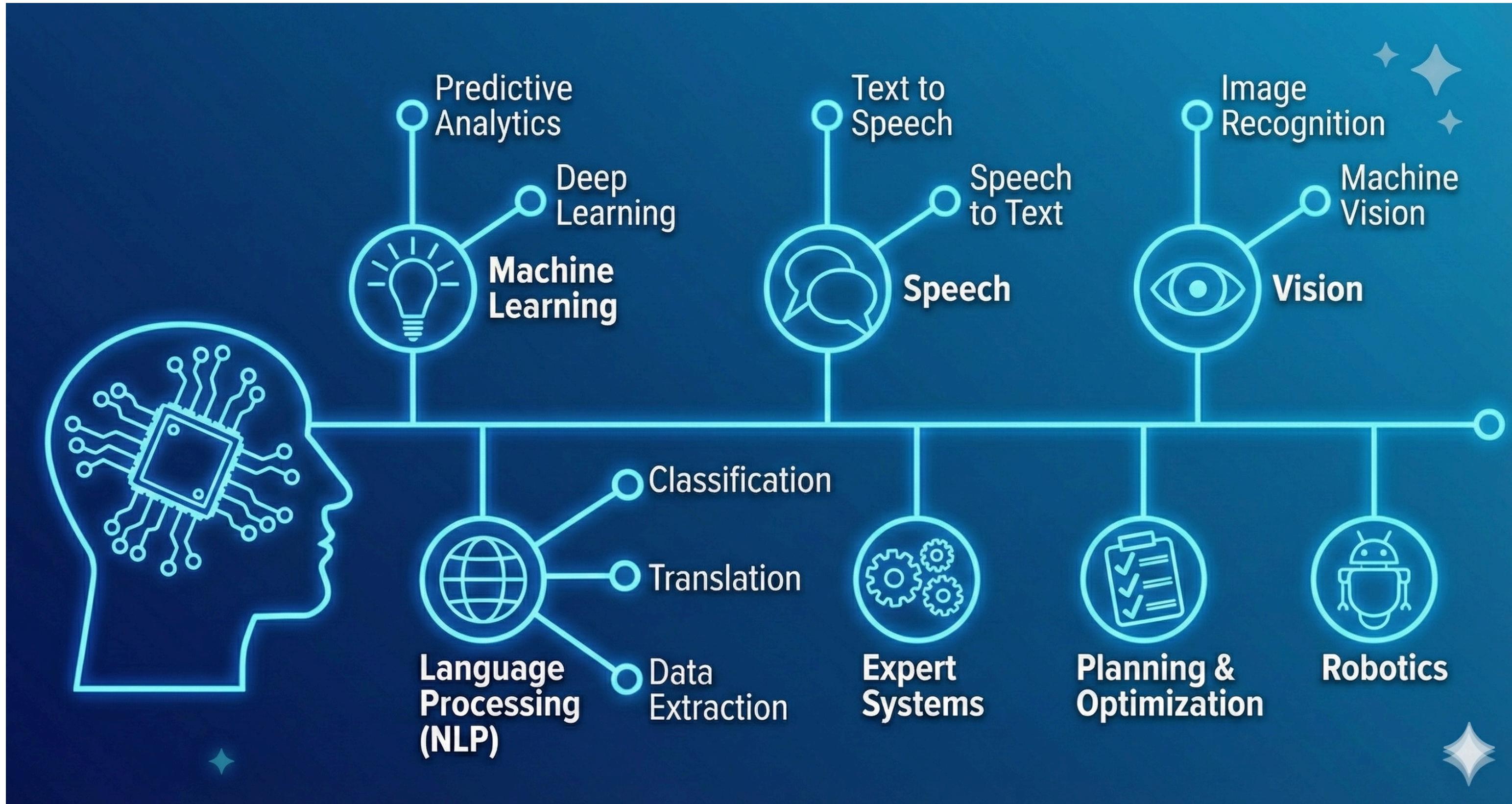
<https://www.cdc.gov/data-modernization/php/ai/cdcs-vision-for-use-of-artificial-intelligence-in-public-health.html>

Artificial Intelligence is **a set of computer techniques that let machines perform tasks that normally require human thinking**, such as recognizing patterns, making decisions, summarizing information, or predicting what might happen next.

Instead of following a strict list of rules, AI systems learn from examples, adjust themselves as they go, and improve when given more data.

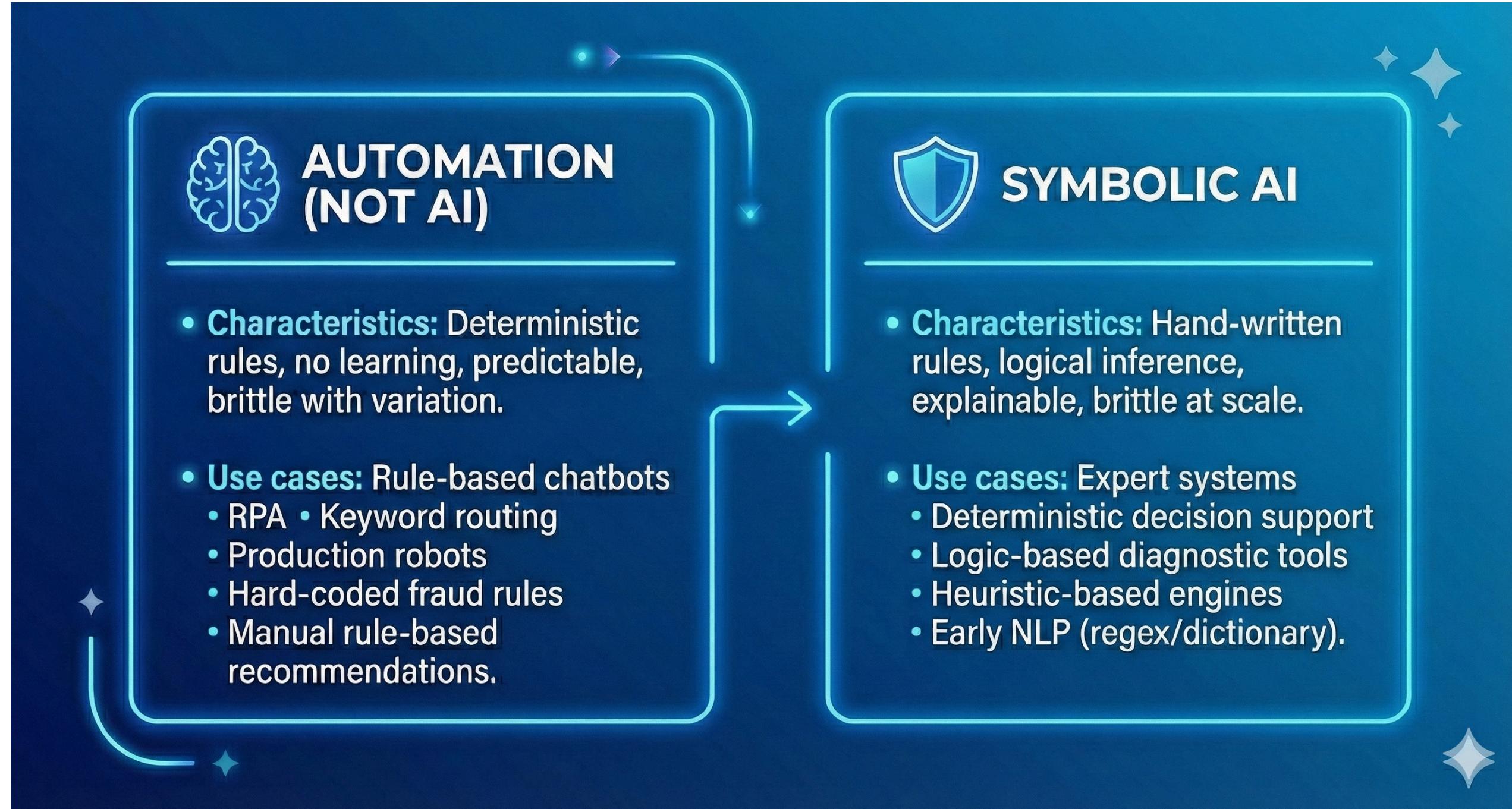
A Very Broad View of AI

Note: This slide is a confusing mixture of AI techniques, possible use cases and use cases that might not be modern AI.



It Probably Isn't AI

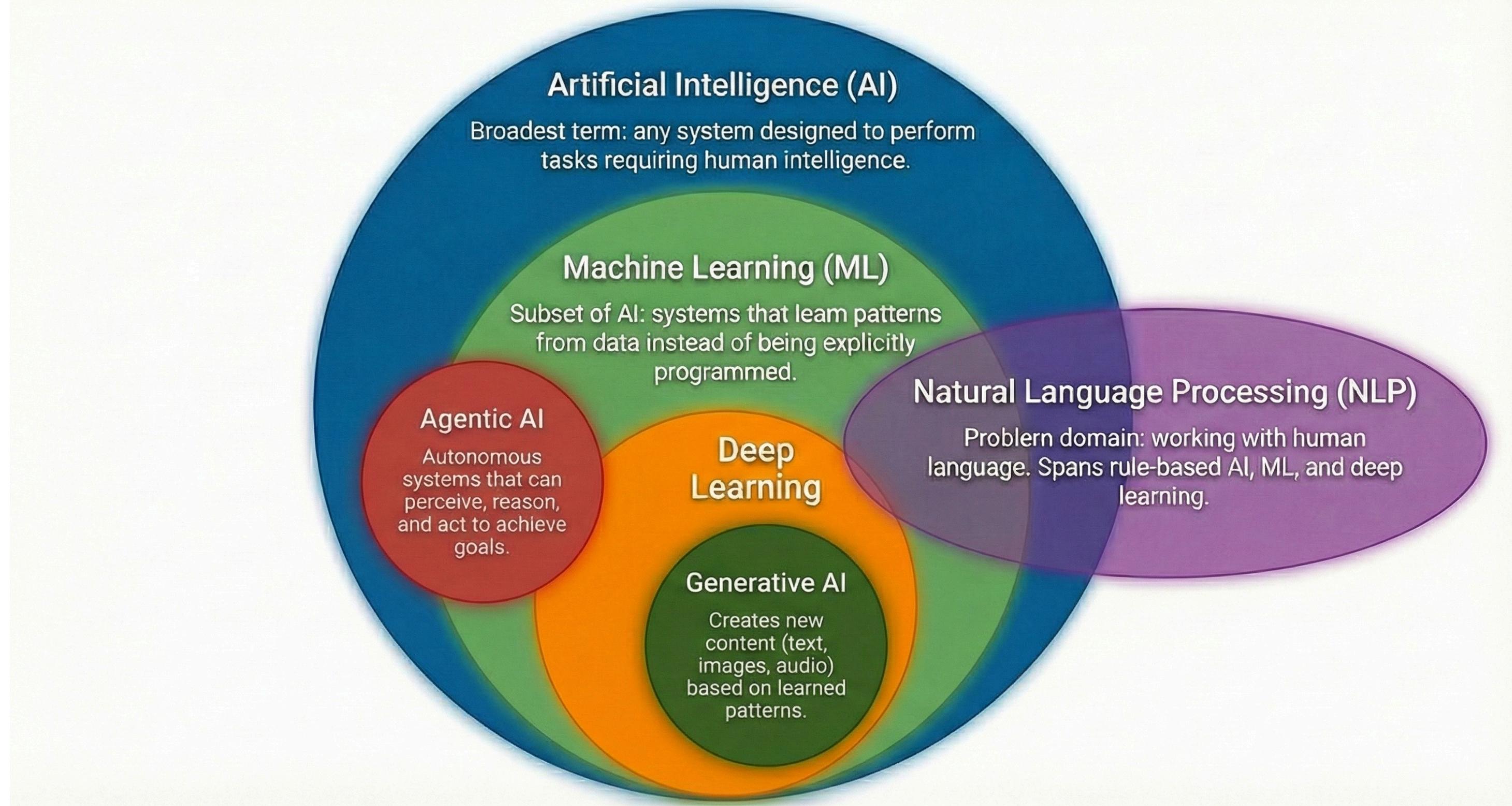
Some automation and expert systems use human-written if/then rules to imitate expert reasoning, while AI discovers its rules from data.



Prompt: Make the first item after 'Use Cases:' a bullet point. Rules-based chatbots is a bullet point. Expert systems is a bullet point. Make sure each bullet point is on a separate line.

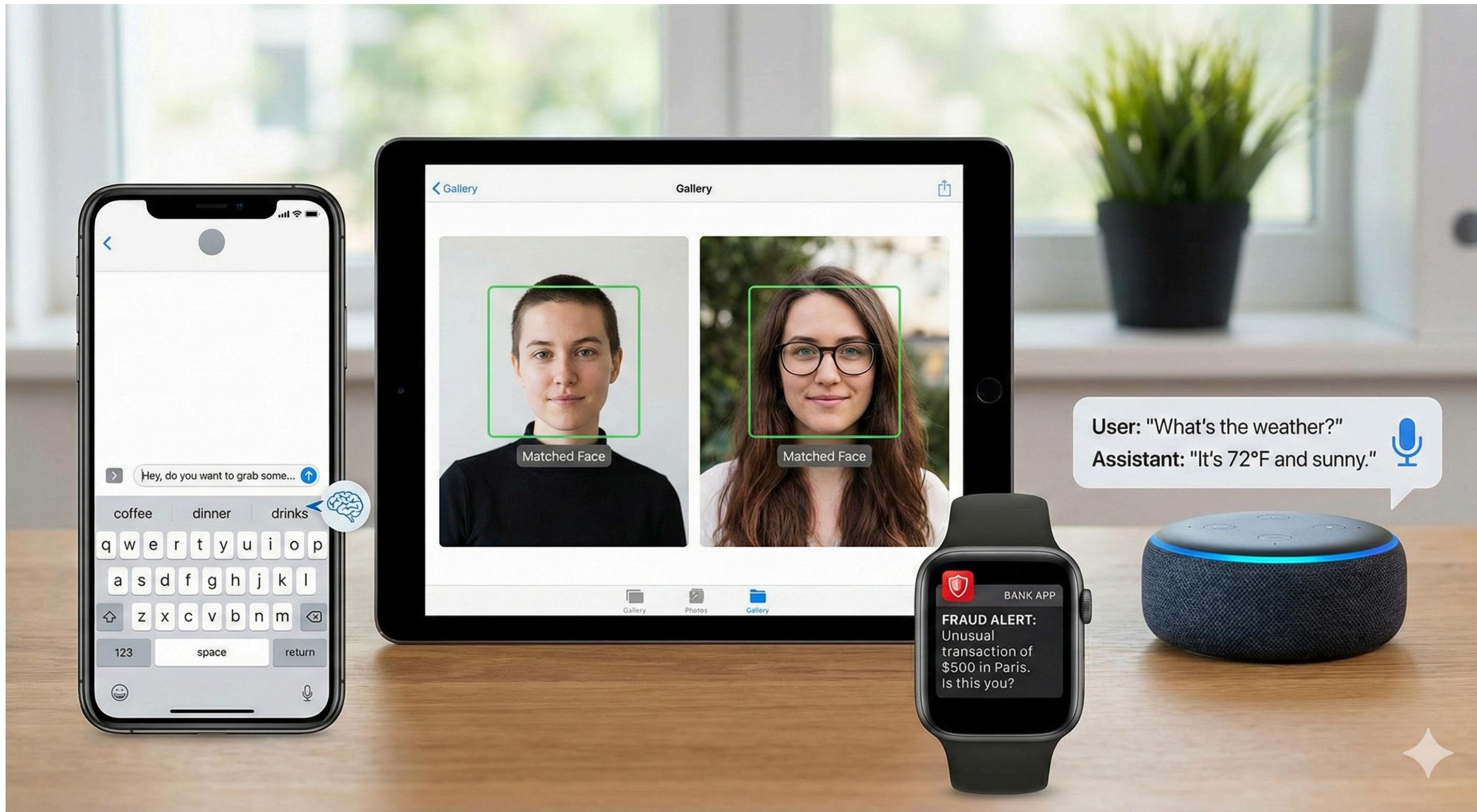
AI Subfields

AI is a field that includes multiple techniques (ML, deep learning, etc.).



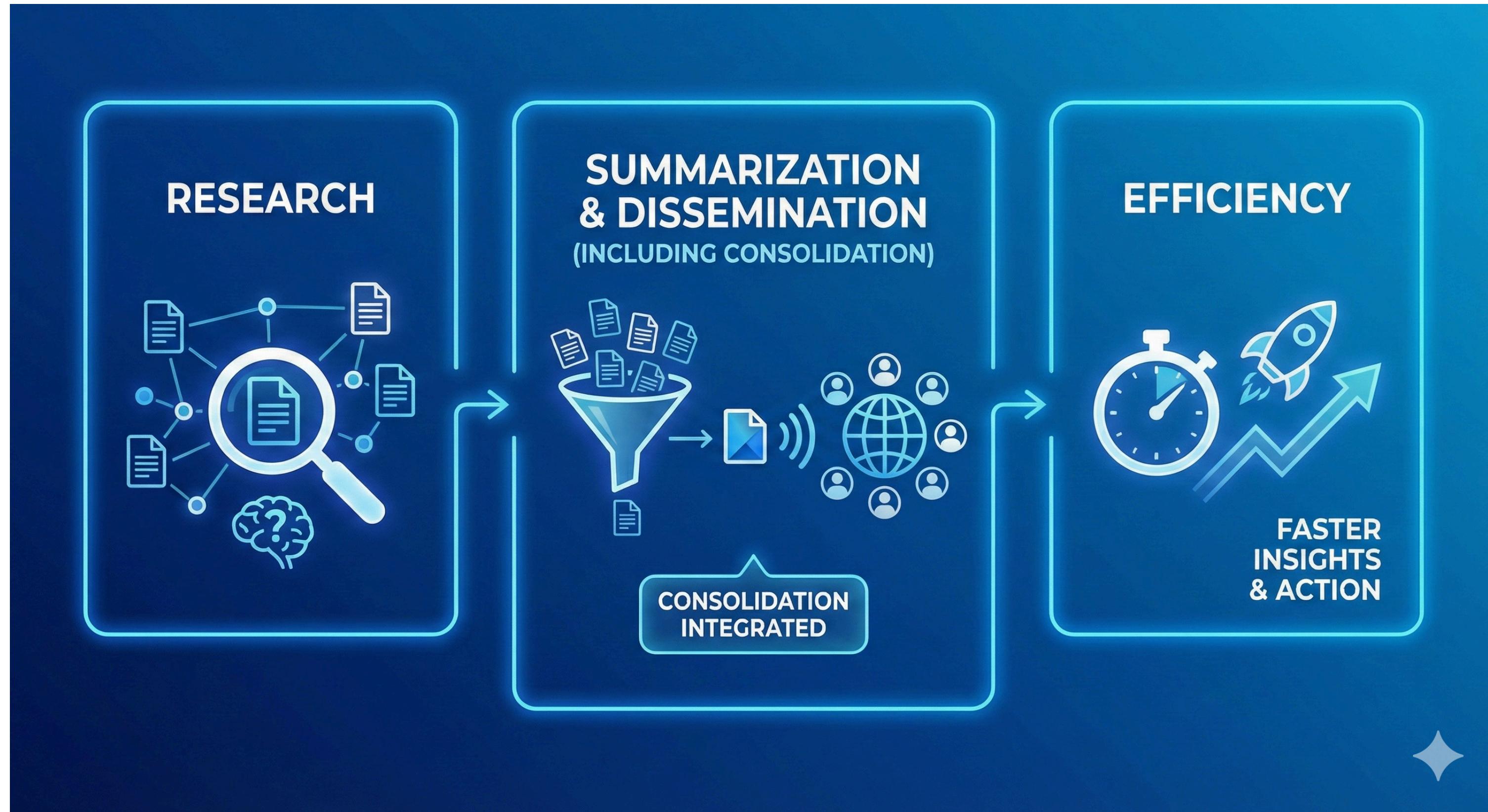
Note: NLP is a **problem domain** that leverages AI. Generative AI and Agentic AI are 'technologies'. They are systems built using the techniques.

Everyday Examples of AI



Prompt: Create an image that has images representing these ideas. Typing suggestions on your phone; the phone learns from previous words to predict the next one. Photo apps recognizing faces, even if the person changes their hair or angle. Fraud alerts from your bank, which notice unusual patterns in your spending. Chatbots or voice assistants, which understand your question and generate a helpful response.

Organizing AI by Use



Prompt: Create an image that has images representing these aspects of AI - Research, Consolidation, Summarization and Dissemination, Efficiency.

Examples: AI for Research in Public Health

Projects focused on developing predictive models, identifying risk factors, and generating new insights

#	Title/Project	Institution/Organization	Year	Description
1	US Diabetes Risk Prediction Using BRFSS Data	Multiple institutions	2024-2025	Analysis of 253,680 adult respondents using machine learning models (Extra Trees Classifier, Random Forest, XGBoost) to identify influential predictors of diabetes likelihood
2	Social Media Analysis for Mental Health	Research study	2022	Research showing social media discussions can predict mental health consultations on US college campuses, enabling earlier intervention

Examples: AI for Consolidation, Summarization & Dissemination in Public Health

Projects focused on extracting insights from large datasets, analyzing narratives, and synthesizing information for public health decision-making

#	Title/Project	Institution/Organization	Year	Description
3	BlueDot and HealthMap Surveillance Systems	BlueDot, HealthMap	Ongoing	AI-based epidemiological surveillance systems providing early warnings and real-time disease outbreak monitoring and visualization
4	Transforming Public Health Practice with Generative AI	US Health Departments	2024	Exploring how AI supports core public health functions including communications, organizational performance, and novel insights for decision-making

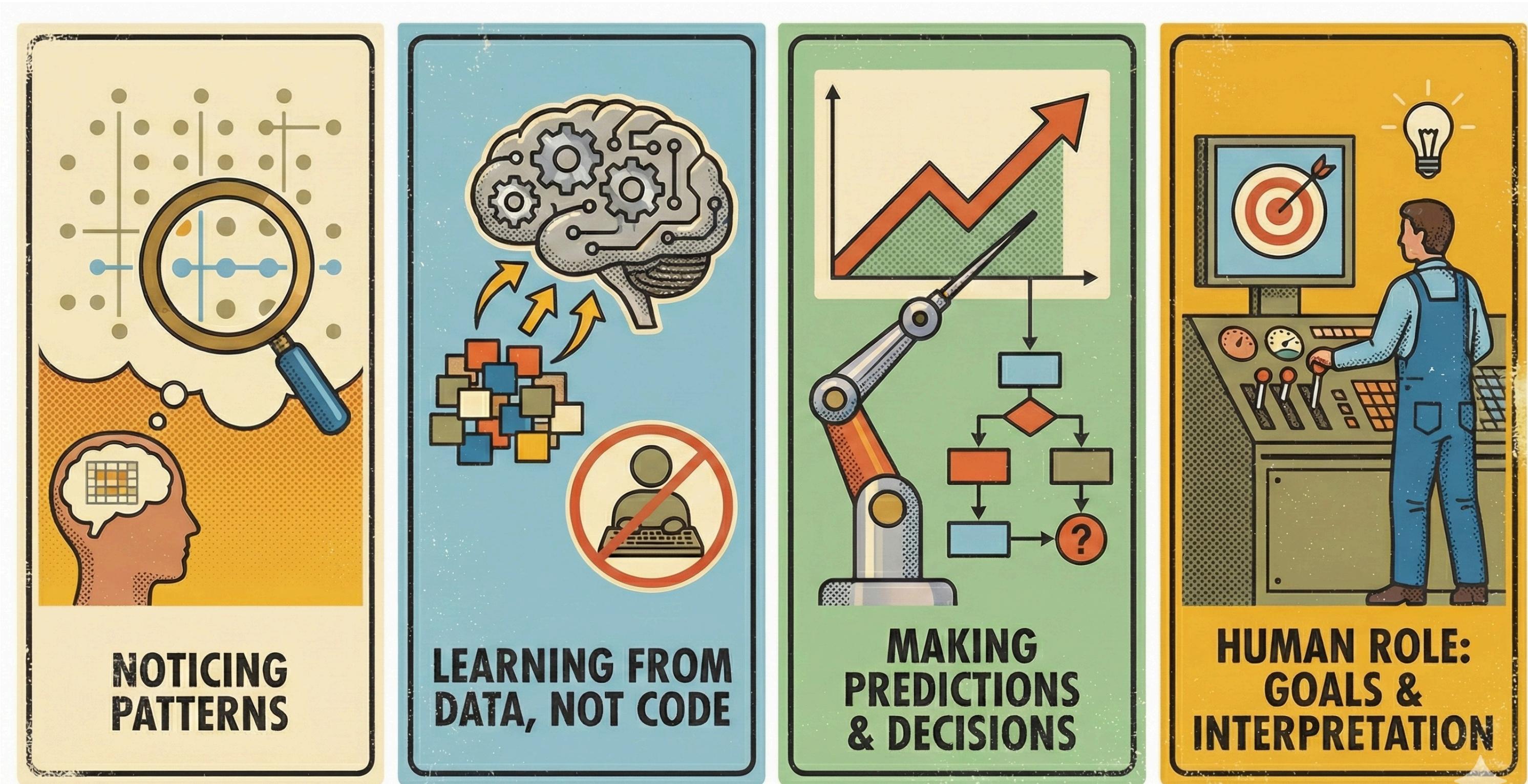
Examples: AI for Efficiency & Automation in Public Health

Projects focused on automating routine tasks, speeding up processes, and improving operational efficiency

#	Title/Project	Institution/Organization	Year	Description
5	Conversational AI for Vaccine Communication	Research review	2023	Systematic review showing chatbot studies measuring influence on vaccine attitudes found evidence of positive effects with no "backfire effects"
6	Motivational Interviewing-Oriented AI Digital Assistant	Research study (Hong Kong & US)	2022-2024	RCT with 177 participants testing an AI-driven chatbot with motivational interviewing techniques

Note Some projects could fit into multiple categories. Classification is based on the primary application or objective of each initiative.

The key idea: AI is not one thing; it is a collection of methods that let computers:



Prompt: Create an image containing images that represent these statements. AI is not one thing; it is a collection of methods that let computers: Notice patterns in data (like humans noticing trends). Learn from those patterns (without a programmer giving step-by-step instructions). Use that learning to make predictions or decisions. Humans still define the goal, set limits, and interpret the results.

A helpful analogy

Think of AI like teaching a new employee:

- You don't give them a rulebook for every situation.
- Instead, they watch examples, practice, and get feedback.
- Over time, they get better and faster at handling similar tasks.
- AI learns the same way, but with data instead of experience.

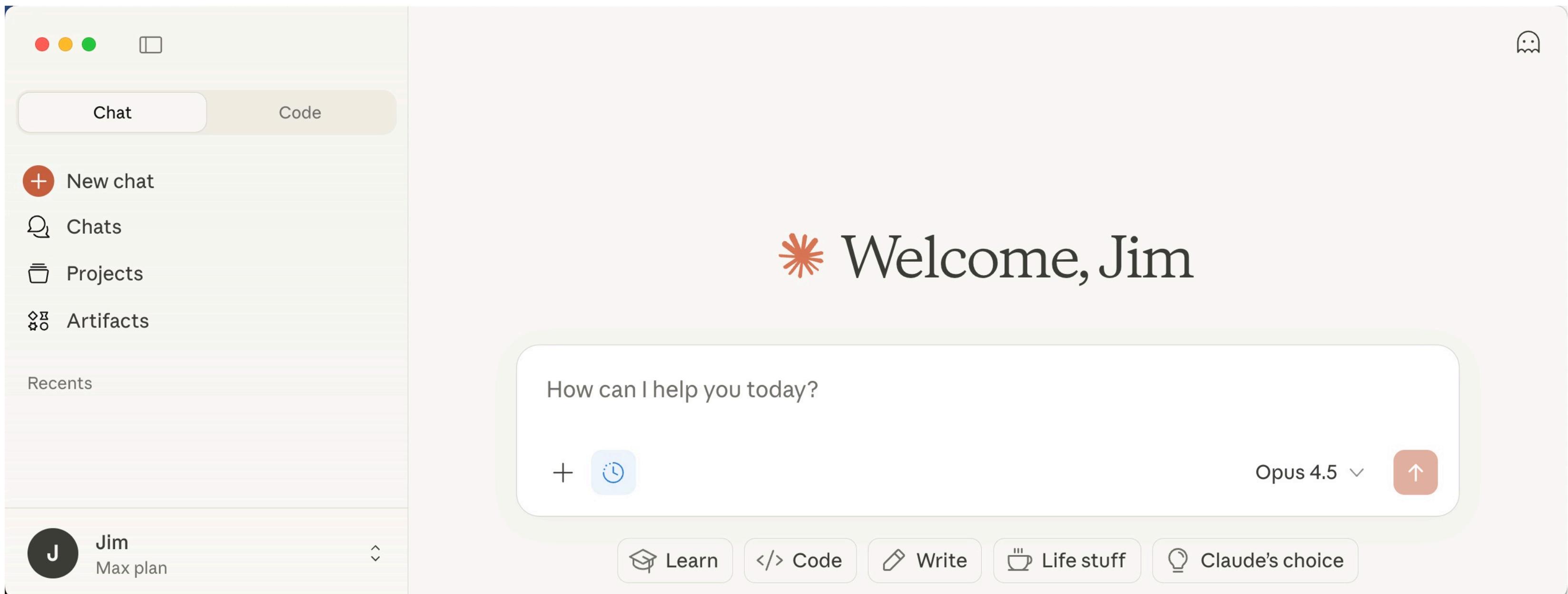
What AI is not

- AI is not magic; it is math and pattern recognition.
- AI is not a full replacement for human judgment; it needs oversight.
- AI does not understand the world the way humans do, even when it generates fluent language.

Major AI Subfields

Concept	Description
Artificial Intelligence (AI)	Broadest term: any system designed to perform tasks requiring human intelligence; includes both rule-based systems and learning-based systems; the umbrella category.
Machine Learning (ML)	Subset of AI: systems that learn patterns from data instead of being explicitly programmed. Requires structured data and engineered features; often interpretable. Examples include decision trees, random forests, logistic regression, and support vector machines.
Deep Learning	Subset of ML: neural networks with many layers that learn representations automatically. Works with unstructured data, needs large datasets and computation, and is often a “black box.”
Generative AI	Subset of Deep Learning: systems that create new content (text, images, code, audio, etc.) by learning patterns from data. Typically built using deep learning, especially transformer models.
Agentic AI	System behavior: AI that has autonomy to take actions toward a goal. Uses planning, decision-making, tool use, memory, and feedback loops.
Natural Language Processing (NLP)	Problem domain: working with human language. Spans rule-based AI, ML, and deep learning.

Claude HHS walkthrough



slide

How can I help you today?

+ ⏱

Learn Code Write ☕

Opus 4.5 ↑

Opus 4.5
Most capable for complex work ✓

Sonnet 4.5
Best for everyday tasks

Haiku 4.5
Fastest for quick answers

More models >

slide

How can I help you today?

+ ⏰

Add files or photos

Add to project >

Research

Web search ✓

Use style >

Connectors >

e Write Life stuff Claude's choice

Opus 4.5 ↑

The image shows the Opus 4.5 application interface. At the top, there is a blue header bar with the word "slide". Below the header is a light gray sidebar containing several icons and text labels. On the left side of the sidebar, there are two buttons: a tan one with a plus sign and a light blue one with a clock icon. Below these are five items listed with arrows pointing to the right: "Add files or photos" (with a paperclip icon), "Add to project" (with a folder icon), "Research" (with a magnifying glass icon), "Web search" (with a globe icon), and "Connectors" (with a grid icon). To the right of the sidebar, the main area has a white background. It features a large, rounded rectangular button with a light gray gradient. Inside this button, the text "How can I help you today?" is centered. Above this button are the two small icons from the sidebar. To the right of the main button, the text "Opus 4.5" is followed by a downward-pointing arrowhead inside a small orange rounded rectangle. Below the main button, there are four smaller rounded rectangles, each containing an icon and text: "Write" (pencil icon), "Life stuff" (coffee cup icon), "Claude's choice" (lightbulb icon), and a partially visible "e" (envelope icon). The entire interface has a clean, modern design with a focus on productivity and user interaction.

slide

read the data from this url and show me the first ten rows in a table.

https://raw.githubusercontent.com/jimcody2014/BA/refs/heads/main/diabetic_data1K.csv

^ 1 step

🌐 Fetched https://raw.githubusercontent.com/jimcody2014/BA/refs/heads/main/diabetic_data1K.csv [githubusercontent.com](#)

● Parsed CSV data and structured tabular display. ▾

Here are the first 10 rows of the diabetic data:

encounter_id	patient_nbr	race	gender	age	time_in_hospital	num_lab_procedures
2278392	8222157	Caucasian	Female	[0-10)	1	41
149190	55629189	Caucasian	Female	[10-20)	3	59
64410	86047875	AfricanAmerican	Female	[20-30)	2	11
500364	82442376	Caucasian	Male	[30-40)	2	44
16680	42519267	Caucasian	Male	[40-50)	1	51



slide

The screenshot shows a user interface for managing projects. On the left is a sidebar with the following items:

- Chat (selected)
- Code
- + New chat
- Chats
- Projects (selected)
- Artifacts
- Recents

At the bottom of the sidebar is a dark circular badge with a white letter "J" and the text "Jim Max plan".

The main area is titled "Projects" and features a search bar with the placeholder "Search projects...". A "Sort by" button is set to "Activity".

The projects listed are:

- Motivation** - Updated 2 days ago
- Do math** - Updated 2 days ago
- Claude to R** - Updated 4 days ago
- stage1 topic model** - Updated 6 days ago
- data generation** - Updated 6 days ago
- C2R - create data** - Updated 15 days ago

A vertical scroll bar is visible on the right side of the main content area.

slide

asciidoc

...



Reply...



Opus 4.5



CSS slide layout padding explanation

Last message 23 hours ago

CSS indentation for AsciiDoc markdown lists

Last message 23 hours ago

CSS text indent matching image alignment

Last message 1 day ago

CSS indentation for instructions and bullets

Last message 1 day ago

CSS padding strategy for markdown text

Last message 2 days ago

Memory

Only you



Purpose & context Jim is creating educational presentations using AsciiDoc converted to Reveal.js slides. The presentatio...

Last updated 12 hours ago

Instructions

I am converting asciidoc files to slide pdfs using reveal and Docker.

Files



1% of project capacity used

convert-to-slides.sh

18 lines

SH

command.txt

1 line

TXT

cdi-theme.yml

97 lines

YML

all-slides.adoc

23 lines

ADOC

slide

asciidoc / CSS indentation for AsciiDoc markdown lists ▾

all-sliders.css

CSS

Attached is the most current version on my .css. When I use this markdown, I
the lines below the image to be indented. This code does the indenting the
way I want it.

[.indent.space-above-3]

* *What is AI?* – The landscape: AI, Machine Learning, Deep
Learning, and NLP

* *Machine Learning Overview* – How machines learn from data:

Show more

Examined CSS compatibility and formatting inconsistencies between working and broken examples. ▾

Let me look at your CSS file to see what styling options are available and understand what
might be happening with the formatting.

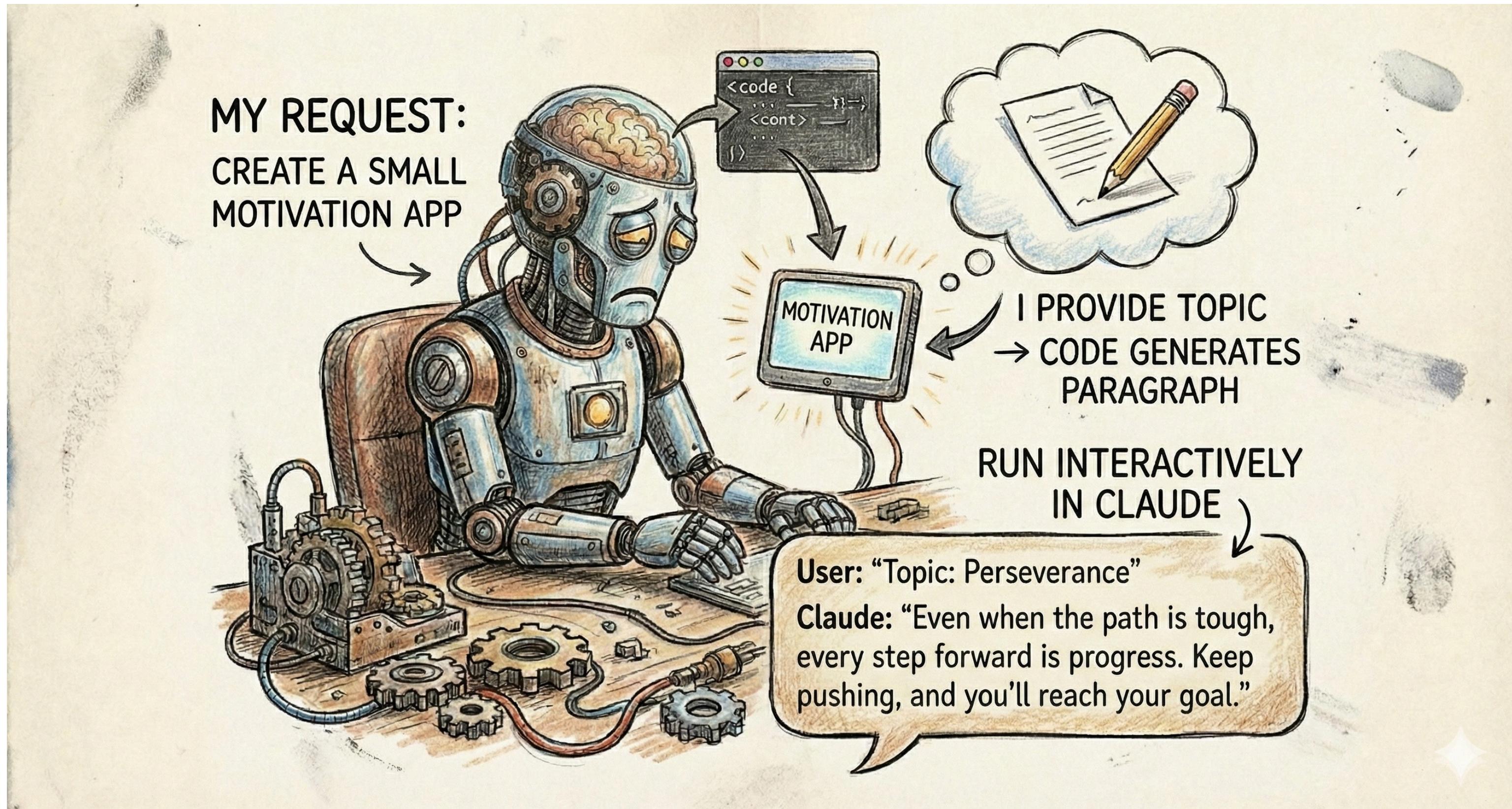
 Reviewing the CSS file to understand available indent styles and formatting options Show working file

Diagnosed block structure issue; prescribed open block wrapper solution.

⌚ 27s ▾

Looking at your CSS, I can see the `.indent` class is defined and works correctly. The issue
is with the **AsciiDoc structure**, not the CSS.

Motivation app



Prompt 1 - Getting started

Prompt

I need some motivation.

Devised warm, conversational encouragement for motivation request. ▼

I've got you.

Here's the thing: the fact that you're asking for motivation means something in you is still fighting, still reaching. That spark matters more than you might realize right now.

Whatever you're facing—whether it's a project that feels too big, a goal that seems far off, or just getting through the day—remember that progress doesn't have to be dramatic to be real. Small steps still move you forward. Showing up, even when it's hard, is an act of courage.

You don't have to feel motivated to take action. Sometimes action is what creates the motivation. Start with one thing. Just one. The momentum will follow.

What's weighing on you? I'm happy to get more specific if you want to talk through it.

Prompt 2

Prompt

How about asking me for a topic?

Fair point—I jumped ahead. What do you need motivation for?

waking up.

That's one of the hardest ones—because it's not just physical, it's the question underneath: *what am I getting up for?*

A few thoughts:

If it's about the morning battle itself: Your brain lies to you at 6 a.m. It says the bed is the only good thing and the day has nothing for you. It's wrong. The day hasn't even had a chance yet. Sometimes the win is just feet on the floor. That's it. Everything else comes after.

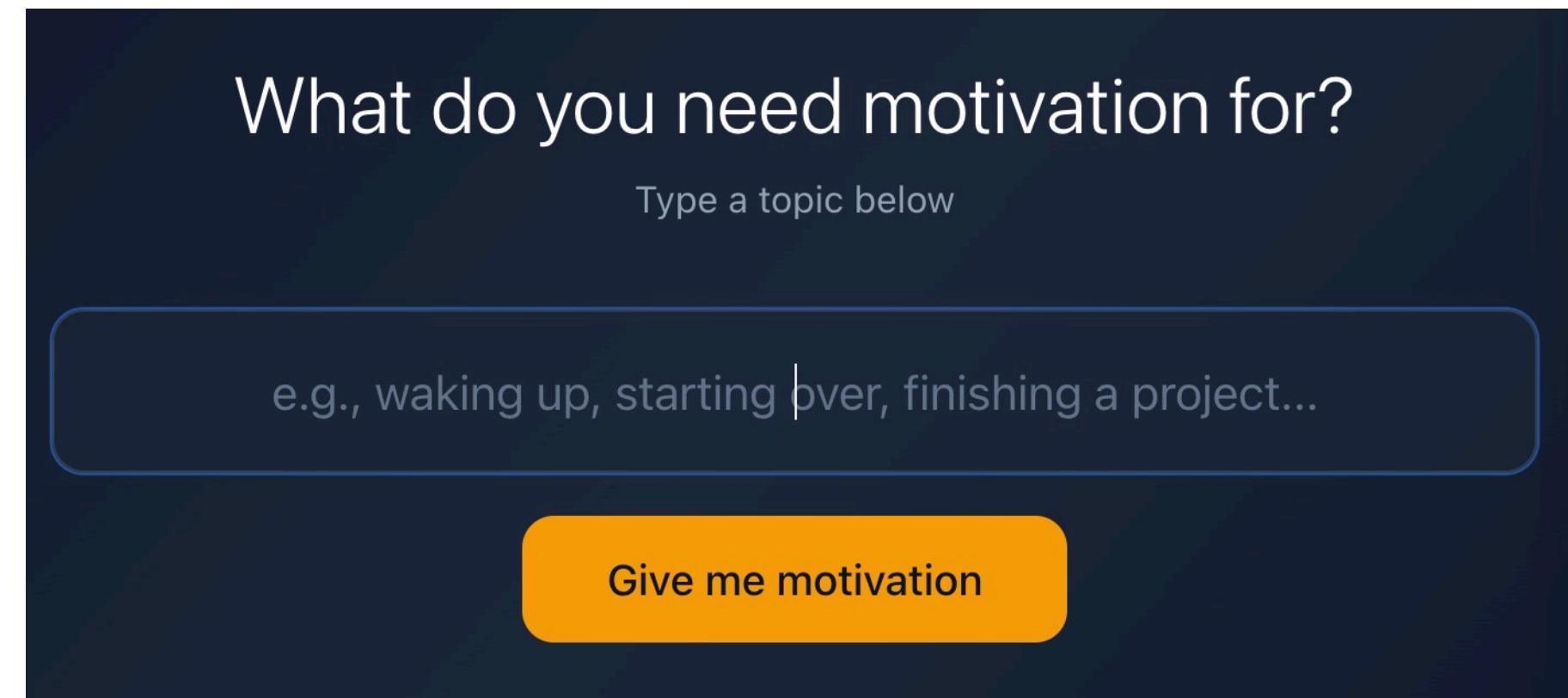
If it's something deeper: Not every day has to have a grand purpose waiting. Sometimes you wake up for small things—coffee, sunlight, curiosity about what might happen, a song you like, someone who'd notice if you weren't there. Those count. They're not small at all, actually.

Which side of it are you feeling?

Prompt 3

Prompt

Create an interface that will run within this chat. It should prompt me for a topic and respond with a 5-6 line motivational statement



Prompt 4

Prompt

You decide.

- Modify your motivation app. Give Claude instructions to shorten or lengthen the response, change the tone of the response, change colors, etc.

What do you need a little snarky motivation for?

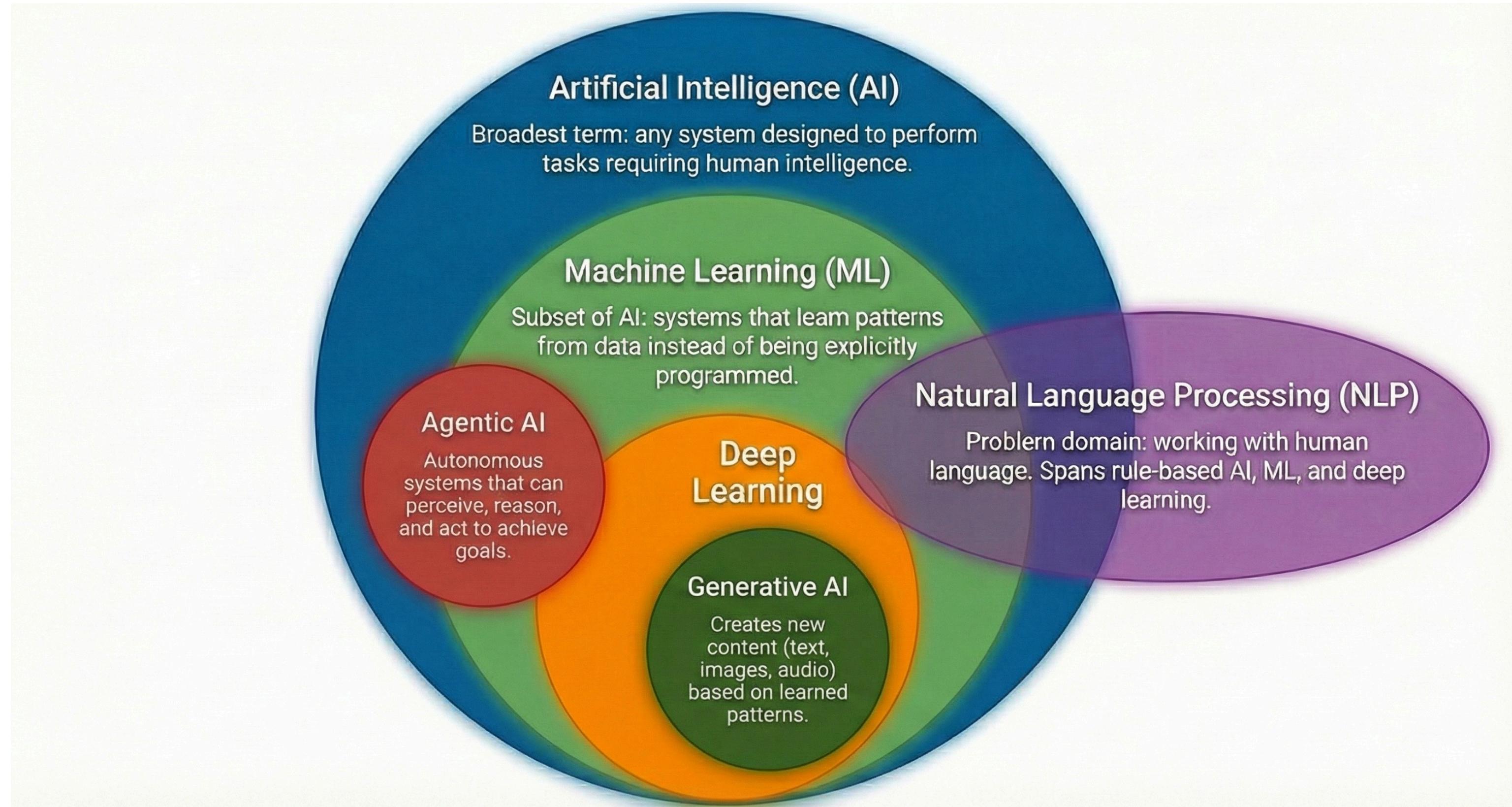
Type a topic below

e.g., waking up, starting over, finishing a project...

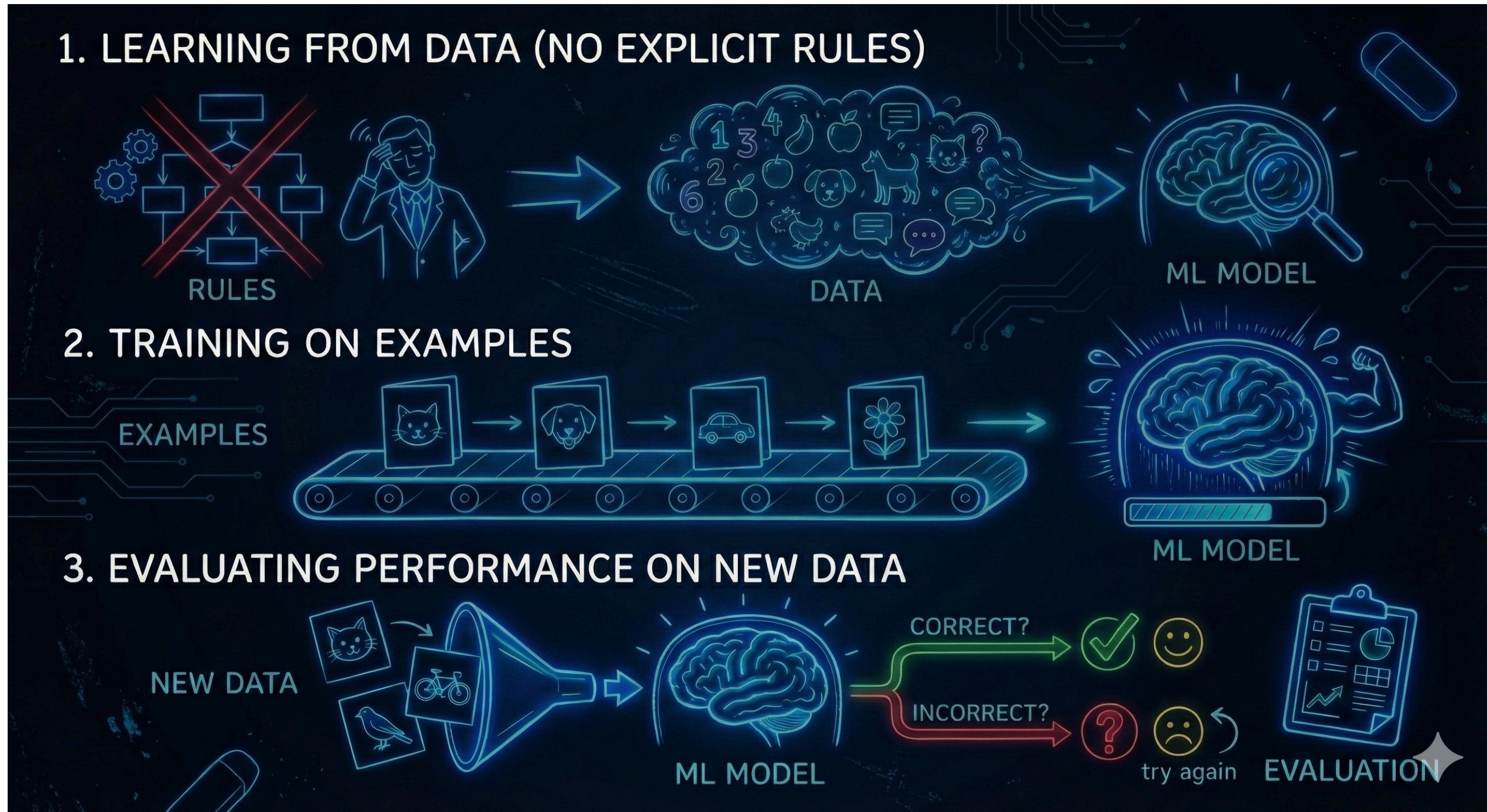
Give me motivation

Foundational Concepts: Machine Learning Overview

Machine learning (ML) is the engine that powers modern AI. Deep learning, NLP, and other advanced techniques all build on ML's core ideas. Understanding ML is the foundation for understanding all of modern AI.



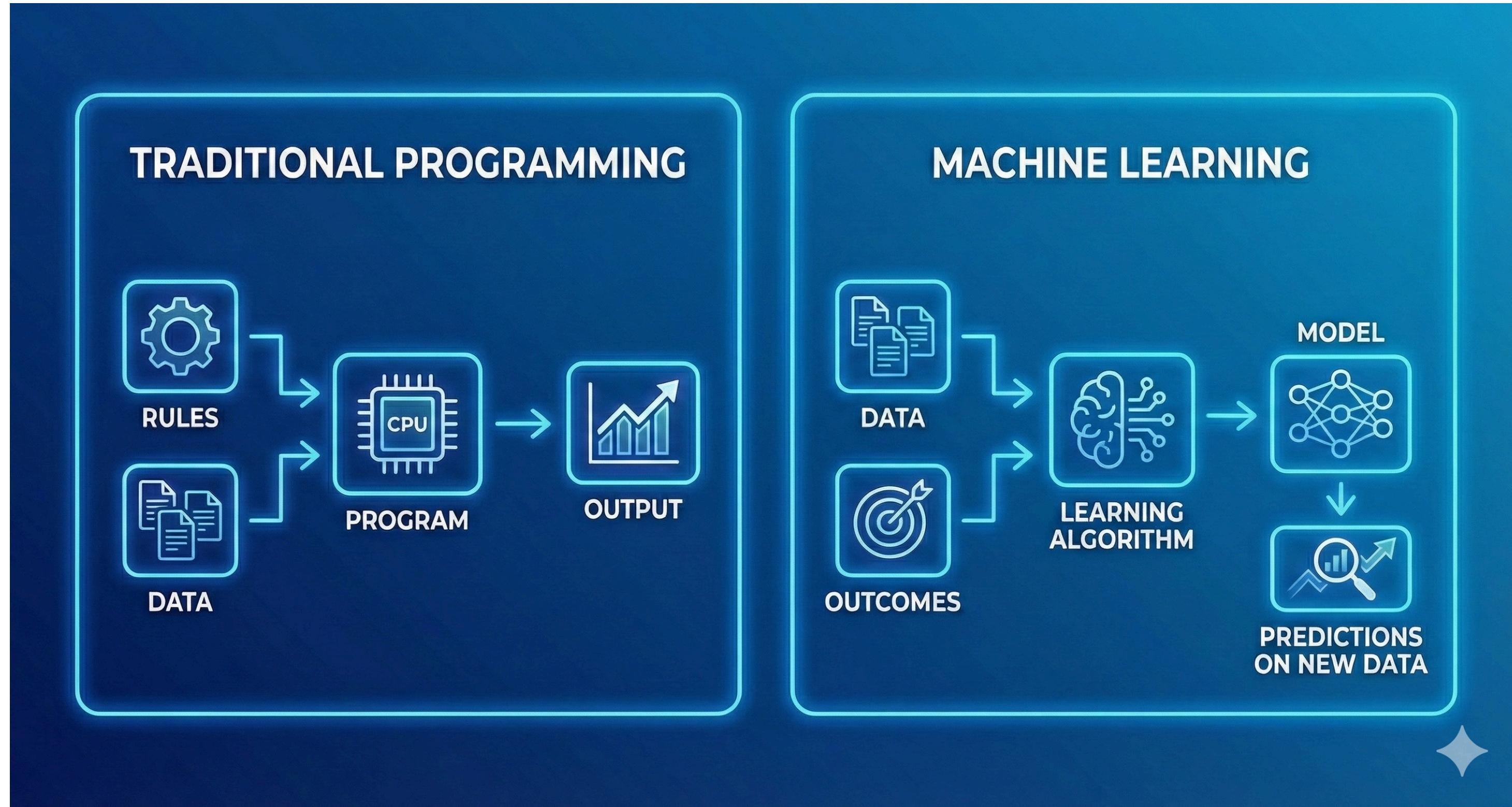
Machine Learning Core Concepts



Prompt: Create an image that portrays the three core concepts of Machine learning. Don't be word heavy. Use images. Create the image in a 'hand drawing' style. Use this as a basis for the image. ML provides the three core concepts that everything else builds on: Learning from data (not from explicit rules), Training on examples, Evaluating performance on new data. Follow-up: Can you make the images a little opaque so that the 1., 2. 3. text stands out more?

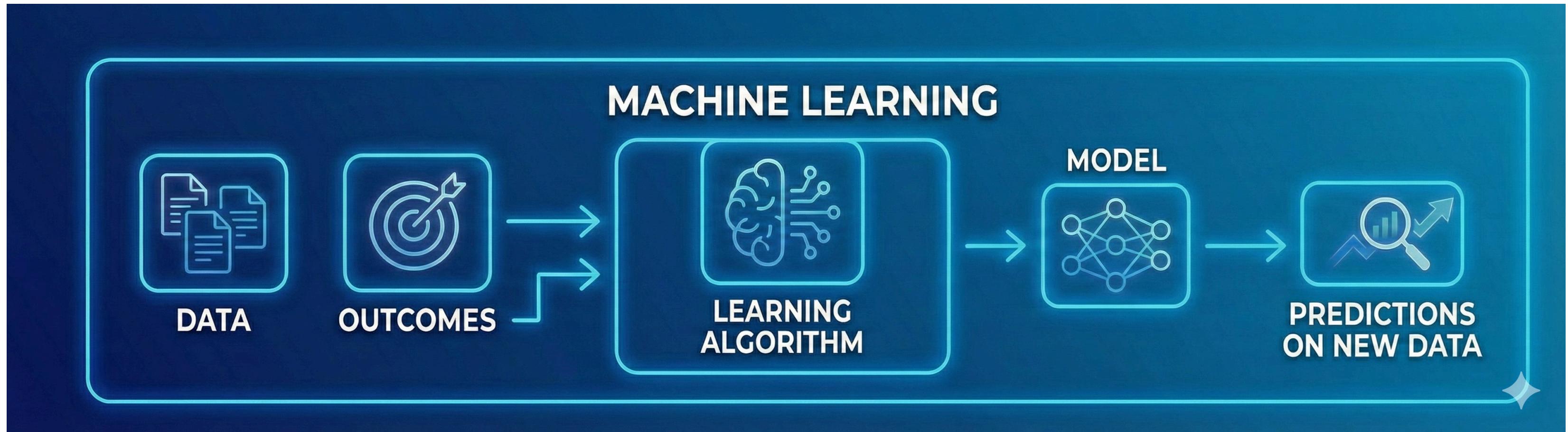
The Core Idea: Learning from Data

Machine learning is a way for computers to **learn patterns from data**, rather than being explicitly programmed with rules.



Prompt: I am discussing machine learning. create a Two-panel diagram: Left: Traditional programming (Rules + Data → Output) Right: Machine learning (Data + Outcomes → Learning Algorithm → Model → Predictions on new data). Use the attached file as a style guide. Follow-up: Leave the title of each box but remove the subtitle that describes the flow. Leave all else as is.

The Learning Process



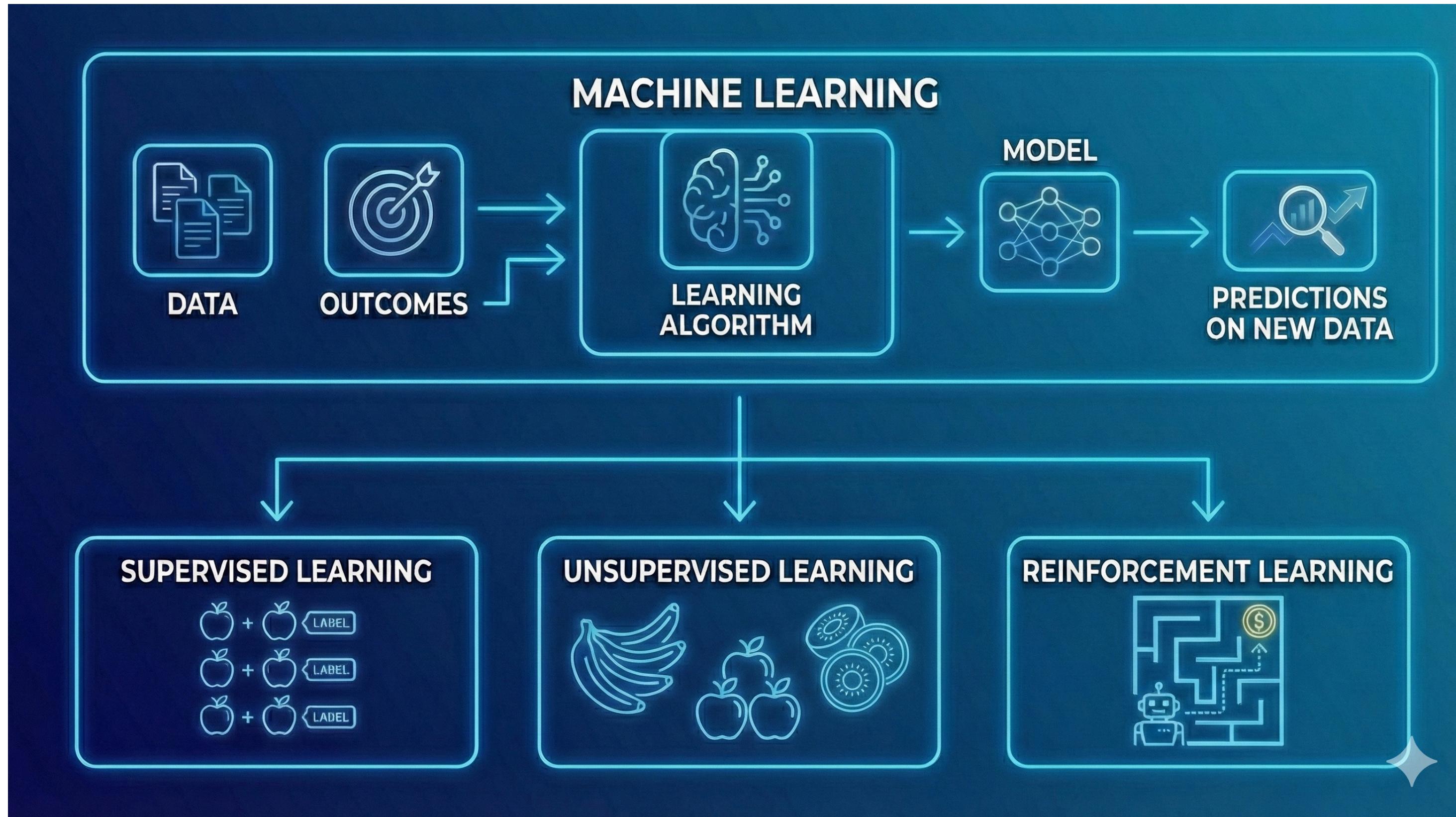
How learning works (intuition, not math):

- The algorithm makes a guess, checks how wrong it is, and adjusts.
- It repeats this process—thousands or millions of times—until it minimizes errors.
- Think of it like learning to throw darts: we throw, see where it lands, adjust our aim, throw again.

The train/test split:

- We can't test learning on the same examples we learned from (that's just memorization).
- So we split the data: learn from some (training data), test on the rest (test data).
- Performance on test data tells us if the model actually learned generalizable patterns.

ML Learning Paradigms



Supervised Learning

In supervised learning, the algorithm learns from **labeled** examples - data where we know the right answer.

encounter_id	patient_nbr	race	gender	age	time_in_hospital	num_lab_procedures	num_procedures	num_medications	number_outpatient	number_emergency	number_inpatient	A1Cresult	insulin	change	diabetesMed	readmitted
2278392	8222157	Caucasian	Female	[0-10)	1	41	0	1	0	0	0	None	No	No	No	NO
149190	55629189	Caucasian	Female	[10-20)	3	59	0	18	0	0	0	None	Up	Ch	Yes	>30
64410	86047875	AfricanAmerican	Female	[20-30)	2	11	5	13	2	0	1	None	No	No	Yes	NO
500364	82442376	Caucasian	Male	[30-40)	2	44	1	16	0	0	0	None	Up	Ch	Yes	NO
16680	42519267	Caucasian	Male	[40-50)	1	51	0	8	0	0	0	None	Steady	Ch	Yes	NO
35754	82637451	Caucasian	Male	[50-60)	3	31	6	16	0	0	0	None	Steady	No	Yes	>30
55842	84259809	Caucasian	Male	[60-70)	4	70	1	21	0	0	0	None	Steady	Ch	Yes	NO

- "Supervised" because the learning is being supervised with correct answers.
- The data includes both inputs (features) and the outcome (label) we want to predict.
- The algorithm learns the relationship between inputs and outcomes.
- Once trained, the model can predict outcomes for new cases when the answer is unknown.

Classification examples:

- Will this patient be readmitted? (Yes / No)
- Is this email spam? (Spam / Not spam)
- What type of request is this? (Complaint / Question / Compliment)

Regression examples:

- How many people will enroll next month? (a number)
- How long will this patient stay in the hospital? (days)
- What will the temperature be tomorrow? (degrees)

Unsupervised Learning

In unsupervised learning, there are no labels. The algorithm finds structure or patterns in the data on its own.

- "Unsupervised" because there's no right answer to learn from.
- The algorithm explores the data to find natural groupings, patterns, or simplifications.
- Useful when we don't know what we're looking for, or when we want to understand the structure of your data.

Application	What It Does	Example
Clustering	Groups similar cases together	Which patients have similar profiles? What types of complaints do we receive?
Dimensionality Reduction	Simplifies data with many variables	We have 200 variables—which combinations capture most of the information?

Clustering & Dimensionality Reduction

Clustering:

- Finds natural groupings in data without being told what the groups are.
- You might discover that your population falls into 4–5 distinct segments you didn't know existed.
- Useful for: customer segmentation, identifying outbreak clusters, grouping similar facilities.

Dimensionality reduction:

- When you have too many variables to work with, this technique reduces them to a smaller set.
- It finds combinations of variables that capture most of the important variation.
- Think of it as data simplification: 50 variables become 5 composite variables that retain most of the signal.
- Useful for: simplifying complex data before analysis, identifying which factors matter most.

Reinforcement Learning

Reinforcement learning is a third paradigm where an agent learns by taking actions and receiving feedback (rewards or penalties).

- Different from supervised (no labeled examples) and unsupervised (not just finding structure).
- An agent interacts with an environment, takes actions, and learns from the results.
- Good outcomes = rewards; bad outcomes = penalties.
- Over time, the agent learns a strategy (policy) that maximizes rewards.

Classic examples:

- Game-playing AI (learns to win by playing millions of games)
- Robotics (learns to walk by trial and error)
- Recommendation systems (learns what to show you based on your clicks)

Why it's not our focus:

- Less common in public health and government contexts (so far).
- Requires an environment where the agent can take actions and observe results.
- Most agency AI applications are supervised or unsupervised.

Why This Matters for Everything Else

The learning paradigms are core concepts. **Learning from data**, supervised vs. unsupervised, training and testing—appear in every AI application you'll encounter.

- Deep learning uses the same paradigms:
 - Supervised deep learning: image classification, speech recognition
 - Unsupervised deep learning: finding patterns in text or images
 - Reinforcement learning: game-playing AI, robotics
- NLP uses the same paradigms:
 - Supervised: classify emails as spam/not spam, sentiment analysis
 - Unsupervised: topic modeling, clustering documents by similarity
- This vocabulary transfers everywhere:
 - Training data, test data
 - Labels, features
 - Classification, regression, clustering
 - Overfitting (learning too much from training data)

The Three Paradigms at a Glance

A summary reference for the three learning paradigms.

Paradigm	What It Needs	What It Does	Common Tasks	Public Health Example
Supervised	Labeled data (inputs + outcomes)	Learns to predict outcomes	Classification, Regression	Predict readmission risk
Unsupervised	Unlabeled data (inputs only)	Finds structure or patterns	Clustering, Dimensionality reduction	Identify patient segments
Reinforcement	Environment + feedback	Learns strategy through trial and error	Game playing, Robotics	(Less common in public health)

Machine Learning Summary

Component	Key Takeaway
ML is the foundation	Everything else in AI builds on ML's core concepts
Learning from data	ML finds patterns through iteration, not explicit rules
Supervised learning	Learns from labeled examples; classification and regression
Unsupervised learning	Finds structure without labels; clustering and dimensionality reduction
Reinforcement learning	Learns from feedback; less common in public health
Why this matters	These concepts recur throughout deep learning and NLP
Three paradigms at a glance	Summary reference for quick recall

Foundational Concepts Exercises

- Topic 1: Was AI used to build the model that performs the task?
- Topic 2: What ML learning paradigm was used?

Topic 1: Was AI used to build the model that performs the task?

A system that analyzes chest X-rays to flag potential tuberculosis cases for radiologist review.

An eligibility determination system that checks if an applicant's income is below 138% of the federal poverty level and they reside in the state.

A model that predicts which restaurant establishments are most likely to have critical food safety violations, so inspectors can prioritize visits.

A chatbot that answers Medicaid beneficiaries' questions about their coverage, eligibility status, and how to find a provider.

Topic 2: What ML learning paradigm was used?

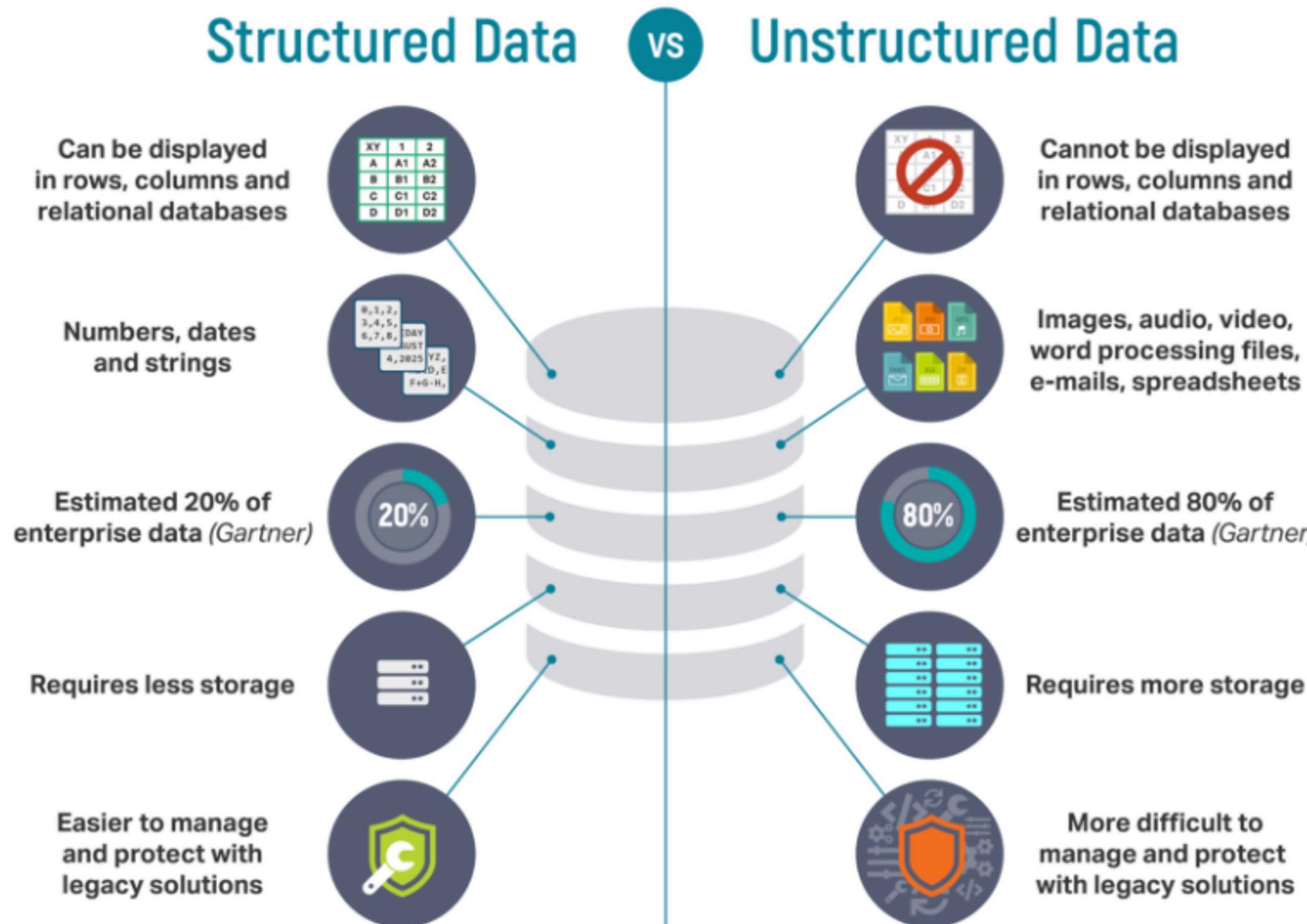
Predict which pregnant women enrolled in Medicaid are at high risk for preterm delivery so care coordinators can provide additional support.

Analyze emergency department visit data to discover natural groupings of patients with similar symptoms, demographics, and utilization patterns—without predefined categories.

Determine which WIC participants are likely to miss their recertification appointment based on past attendance, demographics, and enrollment history.

A simulation system that learns the optimal strategy for allocating limited vaccine doses across counties by trying different distribution approaches and observing outcomes over many simulated scenarios.

Foundational Concepts: Data Types



Structured vs. unstructured data

Data comes in different forms, and the form affects how easily ML can work with it.

- **Structured data:** Organized in rows and columns, where each column is a defined field with a consistent format.
 - Examples: Spreadsheets, databases, claims data, enrollment records
 - Each row is a case/record; each column is a variable
 - ML algorithms can work with this directly
- **Unstructured data:** No predefined format; the information is embedded in content that requires processing to extract.
 - Examples: Free-text notes, emails, PDFs, images, audio recordings
 - The meaning isn't in neat columns—it's in the words, pixels, or sounds

Implication for AI

Data Type

What It Means for ML

Structured

Most ML algorithms work directly; relatively straightforward

Unstructured

Requires extra processing before ML can use it

Extra processing for unstructured data:

- Text → Natural Language Processing (NLP) to extract meaning, categorize, or convert to structured features
- Images → Computer vision to identify objects, patterns, or classifications
- Audio → Speech recognition to transcribe, then NLP to analyze

Unstructured data isn't unusable—but it adds complexity, time, and specialized techniques. If someone proposes an AI project using free-text case notes or scanned documents, that's a signal the project will require more than basic ML.

Labeled vs. unlabeled data

A **label** is the outcome being predicted, not the column headers. Availability of labels determines what types of machine learning can be used.

What "label" actually means:

- The label is the outcome, target, or answer you want the model to learn to predict.
- In a dataset of patients, the label might be: "Was this patient readmitted within 30 days? Yes/No"
- In a dataset of emails, the label might be: "Is this spam? Yes/No"
- The label is what you're trying to predict for future cases where you don't know the answer yet.

Labeled data:

- You have both the inputs (features) AND the outcome (label) recorded for historical cases.
- Example: Patient records where you know who was readmitted and who wasn't.
- This enables supervised learning—the algorithm can learn the relationship between inputs and outcomes.

Unlabeled data:

- You have inputs but NOT the outcome.
- Example: Patient records, but readmission status was never tracked.
- This limits you to unsupervised learning—finding patterns or structure, but not predicting a specific outcome.

Where labels come from

Source	Example	Consideration
Recorded in normal operations	Claims data includes whether patient was readmitted	Best case—labels already exist
Created from historical decisions	"Fraud" label based on cases that were investigated	Labels reflect past decisions, which may be biased
Manually created for the project	Staff review 1,000 cases and label them	Time-consuming and expensive; need enough labeled examples
Doesn't exist	"Success" was never defined or tracked	Can't do supervised learning without creating labels first

Labels don't appear magically—someone or something recorded them. And the way labels were created matters. If "fraud" labels come from who got investigated (not who actually committed fraud), the model learns to predict who gets investigated, not who commits fraud.

Connection to learning paradigms

Data Situation	Learning Paradigm	What You Can Do
Have labels	Supervised	Predict outcomes (classification, regression)
No labels	Unsupervised	Find patterns, groups, structure (clustering, dimensionality reduction)

Volume: How much data is available?

ML learns from examples. More examples generally means better learning but what is "enough" depends on the problem.

- ML algorithms find patterns by seeing many examples. Too few examples, and the algorithm can't learn reliably.
- There's no magic number, but rough guidelines help set expectations.

Situation	Rough Minimum	Notes
Simple problem, structured data	Hundreds to low thousands	Logistic regression, simple decision trees
Moderate complexity	Thousands to tens of thousands	Random forests, gradient boosting
Complex patterns, many variables	Tens of thousands or more	May need more sophisticated approaches
Deep learning (images, text)	Often tens of thousands to millions	Data-hungry techniques

- **Too little data:** Model may memorize the training examples rather than learning generalizable patterns (overfitting).
- **Rare outcomes:** If you're predicting something that happens 1% of the time, you need enough data to have sufficient examples of that rare event. For example, if readmission happens 10% of the time and you have 500 records, you only have ~50 readmission cases to learn from.

Quality: Is the data any good?

Data quality issues can undermine or derail an AI project. Garbage in, garbage out.

- Real-world data is messy. It was usually collected for operational purposes, not for ML.
- Quality issues are the norm, not the exception. The question is how severe they are and whether they can be addressed.

Common Quality Issues

Issue	Example	Why It Matters
Missing values	30% of records have no income data	Model may learn wrong patterns or exclude too many cases
Inconsistent coding	"Diabetes," "DM," "Type 2 DM," "diabetic" all mean the same thing	Model treats them as different; patterns get diluted
Data entry errors	Birthdate of 1/1/1900; weight of 5,000 lbs	Outliers distort learning
Duplicates	Different IDs for the same patient	Inflates apparent data volume
Outdated information	Address from 10 years ago	May not reflect current reality
Inconsistent definitions	"Enroll date" means different things in different systems	Apples-to-oranges comparisons

Accessibility: Can you actually get to the data?

Just because data exists somewhere doesn't mean you can use it for an AI project. Accessibility barriers are common and often underestimated.

Common Accessibility Barriers

Barrier	Example	What It Means
Different systems	Data is in three different databases that don't talk to each other	Need to extract and link—may require technical work and approvals
Permissions	No permission to access the data	Need to request access; may take weeks or months
Data sharing agreements	Data belongs to another agency or partner	Need legal agreements that can take months
Privacy and legal restrictions	Data contains PHI, PII, or is subject to consent limitations	May need legal review or de-identification
Format barriers	Data is in paper files, scanned PDFs, or legacy systems	May require extraction or digitization
Political barriers	Data owner is uncooperative or protective	May be insurmountable without executive intervention

Five dimensions of data

Before proposing or evaluating an AI project, know what questions to ask about data.

Dimension	Key Question
Structure	Is the data structured (rows and columns) or unstructured (text, images, etc.)?
Labels	Do we have the outcome we want to predict recorded? Where did it come from?
Volume	How many records? How many examples of the outcome?
Quality	How complete and consistent is the data? What are the known issues?
Accessibility	Can we actually get to the data? What approvals are needed?

Working in Posit Cloud

Tasks:

- Create a posit.cloud account
- Install packages for other code (installs.R).
- Set an Anthropic API key (provided by instructor)
- Generate R code in Claude (math.R).
- Run the generated code (math.R) in posit.cloud

Create an account



[Log In](#) [Sign Up](#)

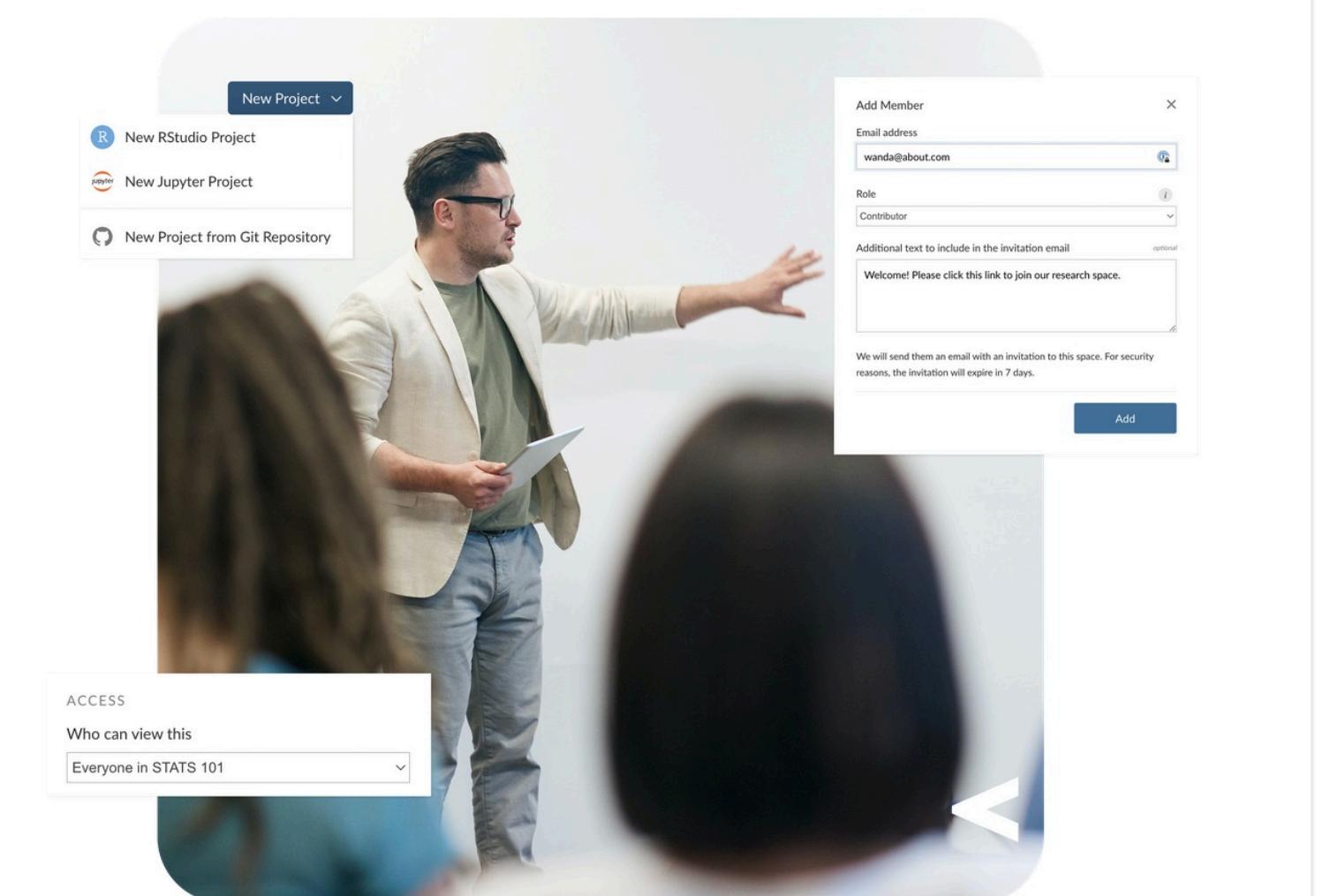
Friction free data science

Posit Cloud lets you access Posit's powerful set of data science tools right in your browser – no installation or complex configuration required.

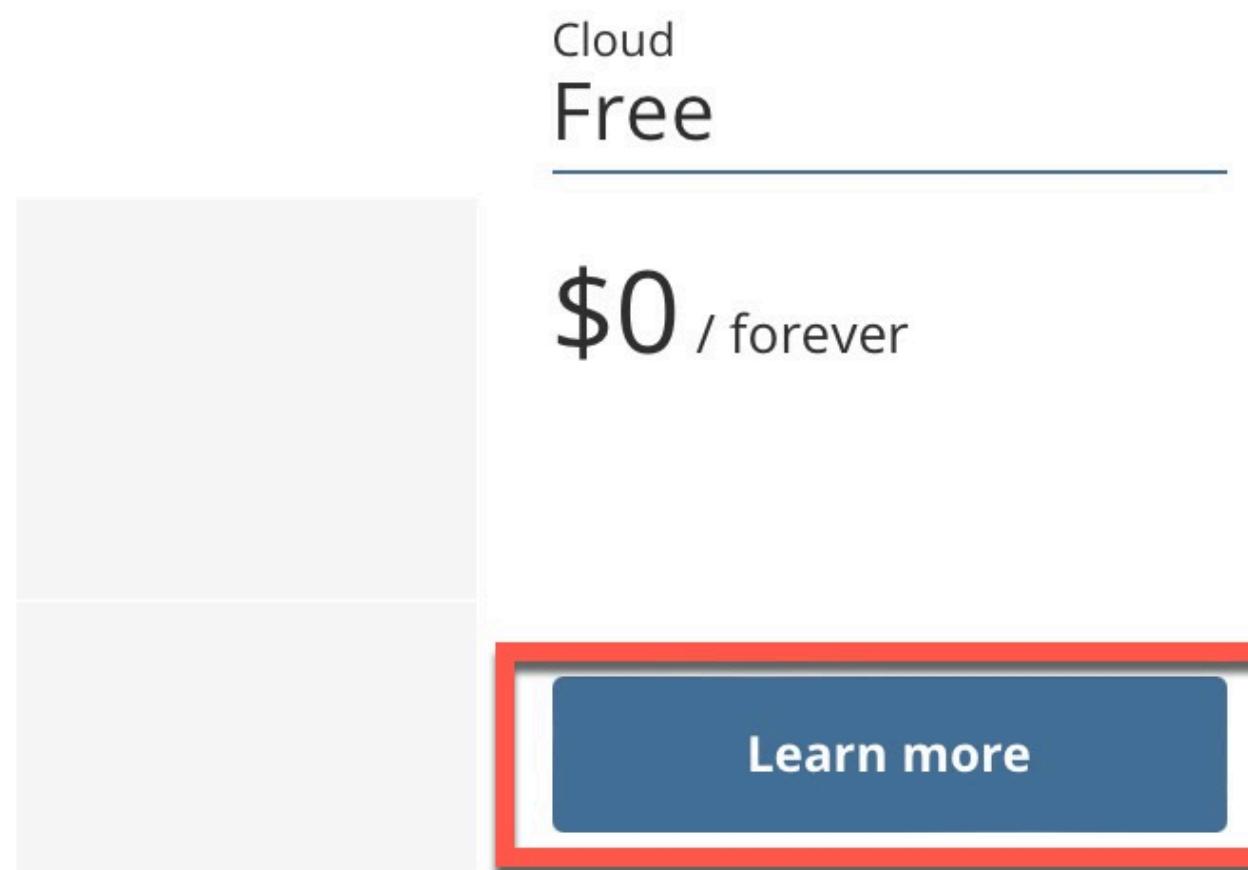
[GET STARTED](#)

[ALREADY A USER? LOG IN](#)

If you already have a shinyapps.io account, you can [log in](#) using your existing credentials.



Select the Free option



Enter account information

Sign Up for Cloud Free

\$0 / forever

Questions? [Contact Us.](#)

If you make limited, occasional use of Posit Cloud, have access to an organization account, or just want to explore Cloud to see if it's right for you, our Cloud Free plan is all you need.

[Sign Up](#)

Verify email

Verify Your Email

We sent a link to verify your email address to
[REDACTED] Please go to your email and click
that link to verify that we have your correct email
address - then return to this page and press
Continue.

[Continue](#)

Wrong email address? [Update your email](#)

Didn't get the email? You can [resend the link](#), but
you should also make sure our email didn't end up in
your spam folder.

Posit.cloud workspace

The screenshot shows the Posit.cloud workspace interface. The left sidebar contains navigation links for 'Spaces', 'Learn', 'Help', and 'Info'. The main content area is titled 'Your Workspace' and includes tabs for 'Content', 'Usage', and 'About'. The 'Content' tab is selected, showing a sub-section for 'Your Content' with options for 'New Project', 'Archive', and 'Trash'. A message at the top states: 'The ability to publish applications and documents to Posit Cloud is now removed for all users. Please visit [Posit Connect Cloud](#) to support your deployment needs.' The bottom of the page features the 'posit Cloud' logo and links for 'Terms' and 'Status', along with social media icons for Facebook, LinkedIn, Instagram, and GitHub.

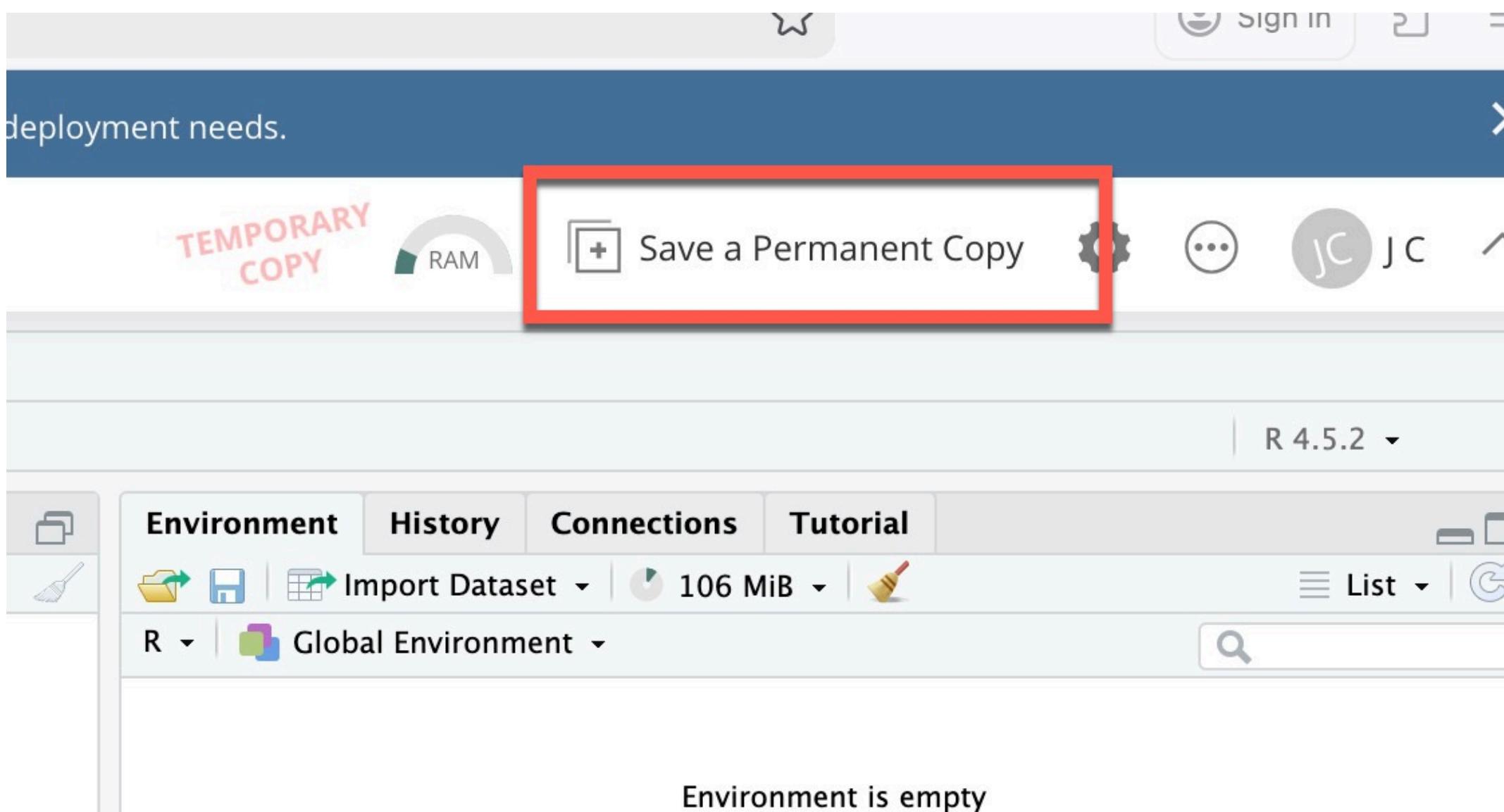
Copy the instructor project

While logged in to your account, paste this link into a url: <https://posit.cloud/content/11680866>

Copy in progress



Save as a permanent copy



Workspace layout

The screenshot displays the posit Cloud workspace interface with several panels:

- Scripting area:** The central workspace where code can be written and run.
- Console:** A panel at the bottom showing the R environment and current working directory.
- Environment & history:** A panel on the right showing the global environment and history.
- Files available:** A file browser panel showing files in the project directory.

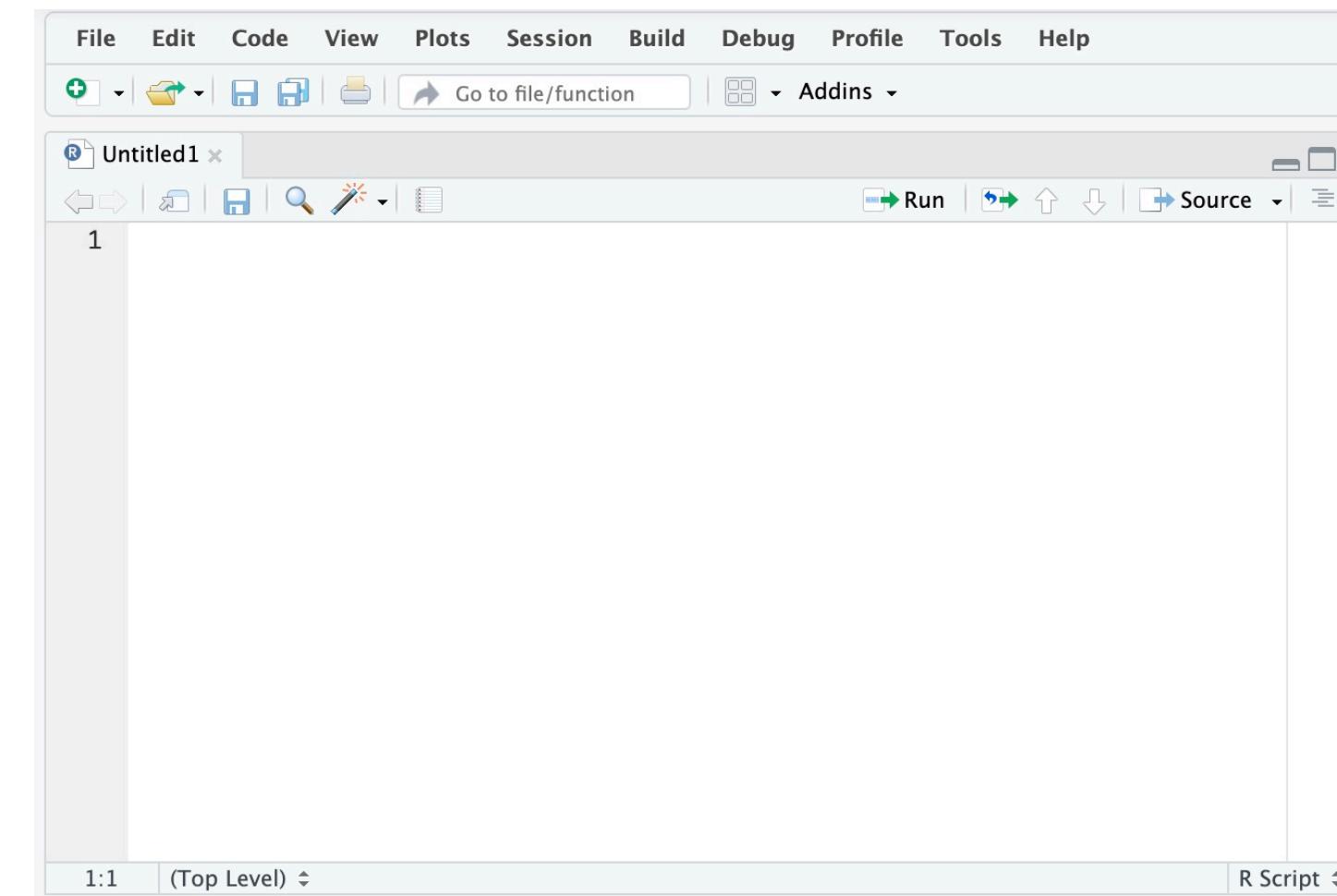
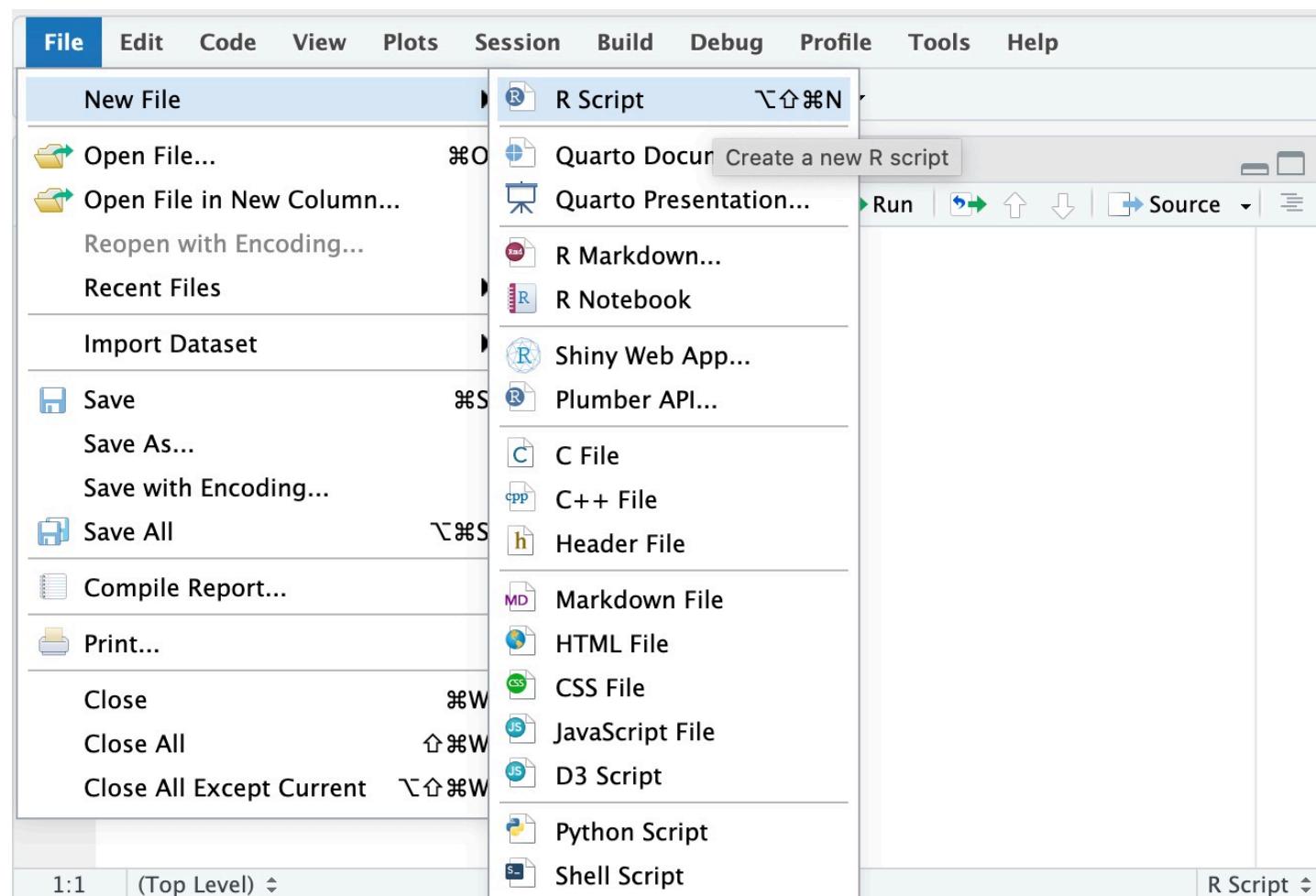
Annotations highlight specific features:

- Scripting area:** Labeled "Scripting area".
- Console for results & messages:** Labeled "Console for results & messages".
- Environment & history:** Labeled "Environment & history".
- Files available:** Labeled "Files available".

The workspace sidebar includes links for Spaces, Learn, Help, and Info.

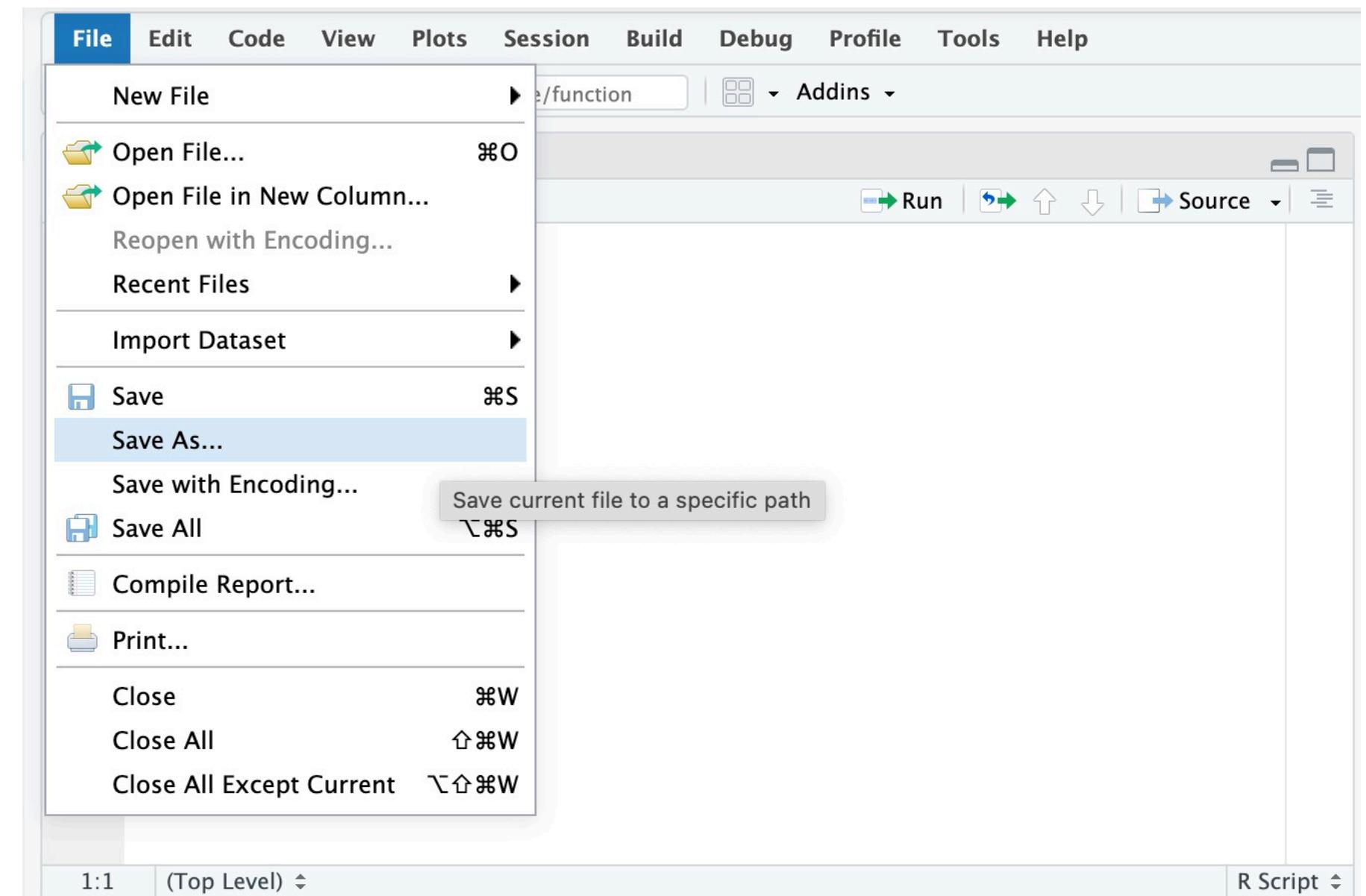
Name	Type	Size	Modified
.Rhistory	0 B	Dec 29, 2025, 3:28 PM	
project.Rproj	205 B	Dec 29, 2025, 3:28 PM	
math-backup.R	287 B	Dec 29, 2025, 3:30 PM	

New R Script



File → Save As

- From the Menu bar, select File → Save As
- Give the script as name (e.g., test1)
- Press the **Save** button
- The script will appear in the files list.



Install R packages

- In the Files section, click on **installs.R**
- This opens the install code.
- Press **Source** to run the installs. Packages need to be installed in your environment before they can be used.

The screenshot shows the RStudio interface. On the left, there is a code editor window titled "Untitled1" containing the following R code:

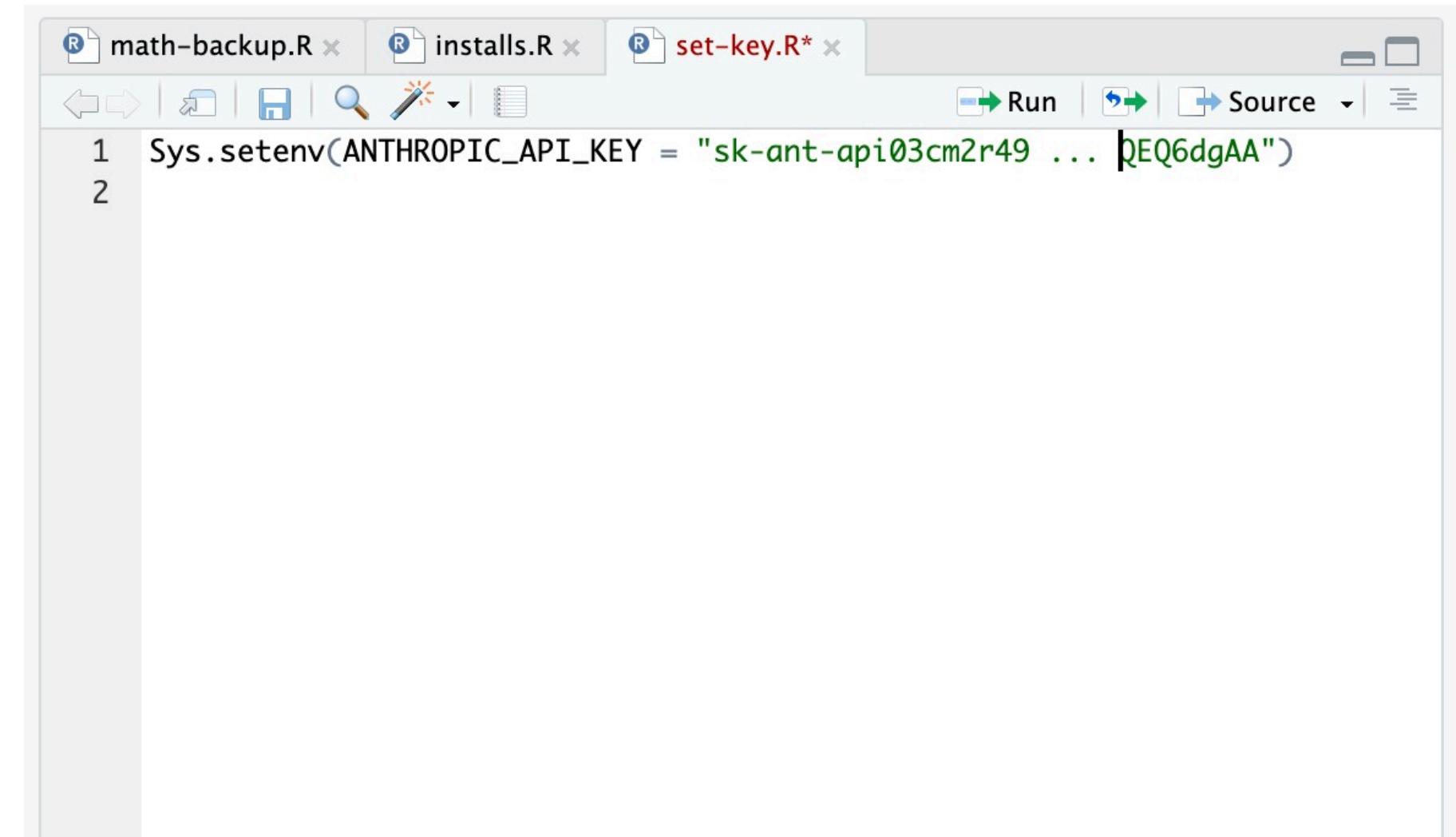
```
1 install.packages(c("httr", "jsonlite",
2                      "dplyr", "knitr", "ggplot2",
3                      "httr2", "rvest", "shiny",
4                      "bslib"))
5 install.packages("officer")
```

The "Source" button in the toolbar above the code editor is highlighted with a red oval. Below the code editor is a "Console" tab. To the right of the code editor is a file browser window showing a list of files in a project folder. The file "installs.R" is highlighted with a red arrow. The file browser includes columns for Name, Size, and Modified.

Name	Size	Modified
.Rhistory	0 B	Dec 7, 2025, 5:56 PM
agent-demo-backup.R	12.4 KB	Dec 10, 2025, 7:35 AM
agent-report-summary.R	20.2 KB	Dec 10, 2025, 7:36 AM
cdc_first_50_rows.csv	4.8 KB	Dec 10, 2025, 6:51 AM
installs.R	188 B	Dec 10, 2025, 8:07 AM
math-backup.R	262 B	Dec 9, 2025, 5:10 PM
motivation-backup.R	2.5 KB	Dec 9, 2025, 4:56 PM
motivation-shiny-backup.R	9.8 KB	Dec 10, 2025, 5:33 AM
project.Rproj	205 B	Dec 10, 2025, 7:35 AM
set-key.R	143 B	Dec 10, 2025, 6:39 AM
YRBSS-1-backup.R	1.7 KB	Dec 9, 2025, 4:56 PM
YRBSS-2-backup.R	11 KB	Dec 9, 2025, 4:52 PM
YRBSS-3-backup.R	9.4 KB	Dec 9, 2025, 5:11 PM
YRBSS-3-shiny-backup.R	10.2 KB	Dec 10, 2025, 5:23 AM
YRBSS-Report-backup.R	5.6 KB	Dec 9, 2025, 5:10 PM
YRBSS.csv	25.4 MB	Dec 10, 2025, 7:11 AM
weekly_report.docx	14.1 KB	Dec 10, 2025, 7:36 AM

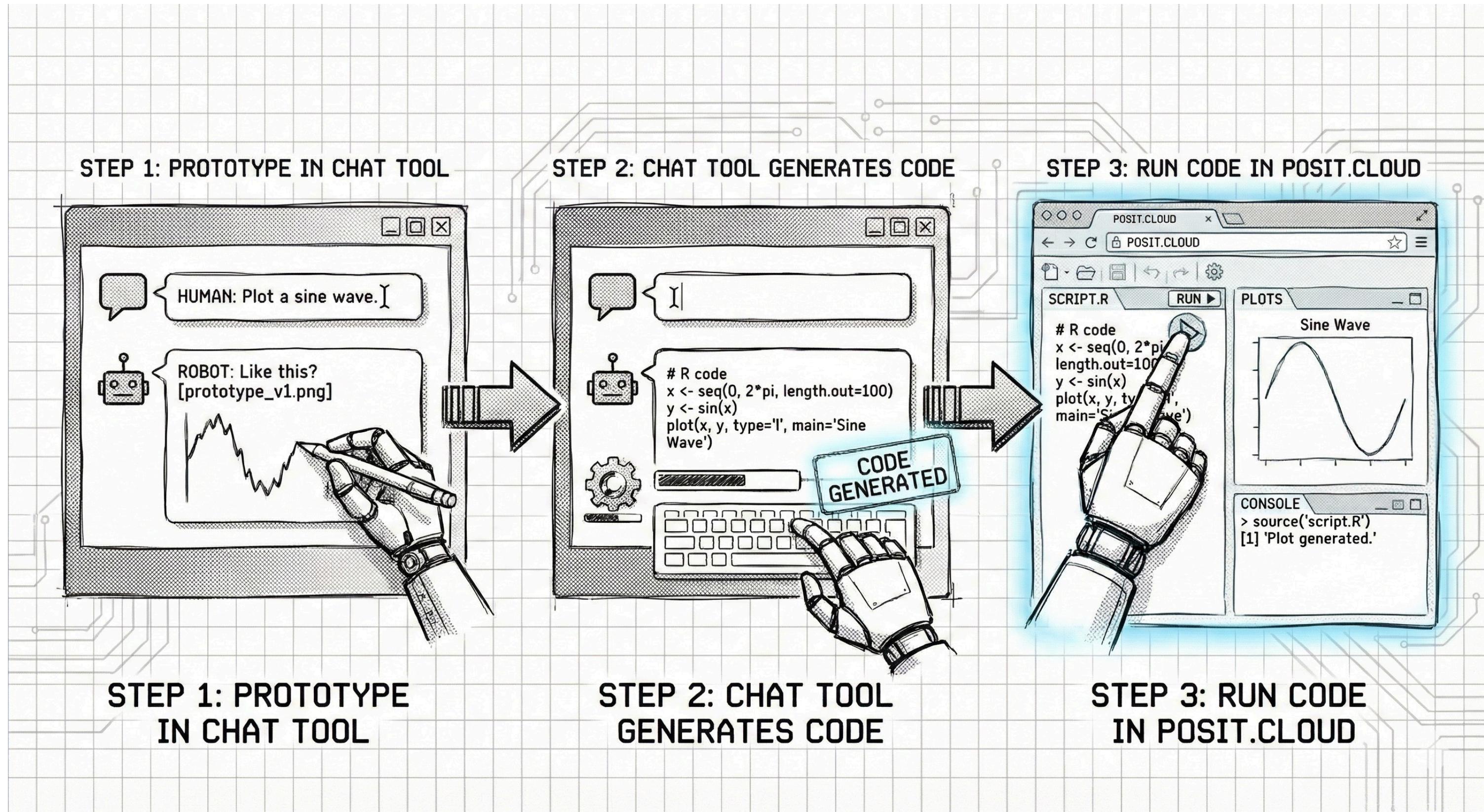
Set the Anthropic API key

- In the Files section, click on **set-key.R**
- This opens the set-key code.
- Put your Anthropic API key in between the double-quotes
 - This key needs to be in place in order to use Anthropic's (Claude) LLMs.
- Press **Source** to set the key.



```
1 Sys.setenv(ANTHROPIC_API_KEY = "sk-ant-api03cm2r49...QEQ6dgAA")  
2
```

From Chat to Code



Prompt: Use the robot hand drawn style. Create an image of images to portray these three steps: prototype in chat tool, chat tool generates code, run code in posit.cloud

Code from Claude

Prompt Instructions

1. I want to provide 2 numbers that you will then multiply together.
2. Generate R code that allows me to interactively enter 2 numbers that will be multiplied together. The results should be displayed in the console.
3. If needed -- Modify the code to use readline

Copy this code if Claude refuses to use readline.

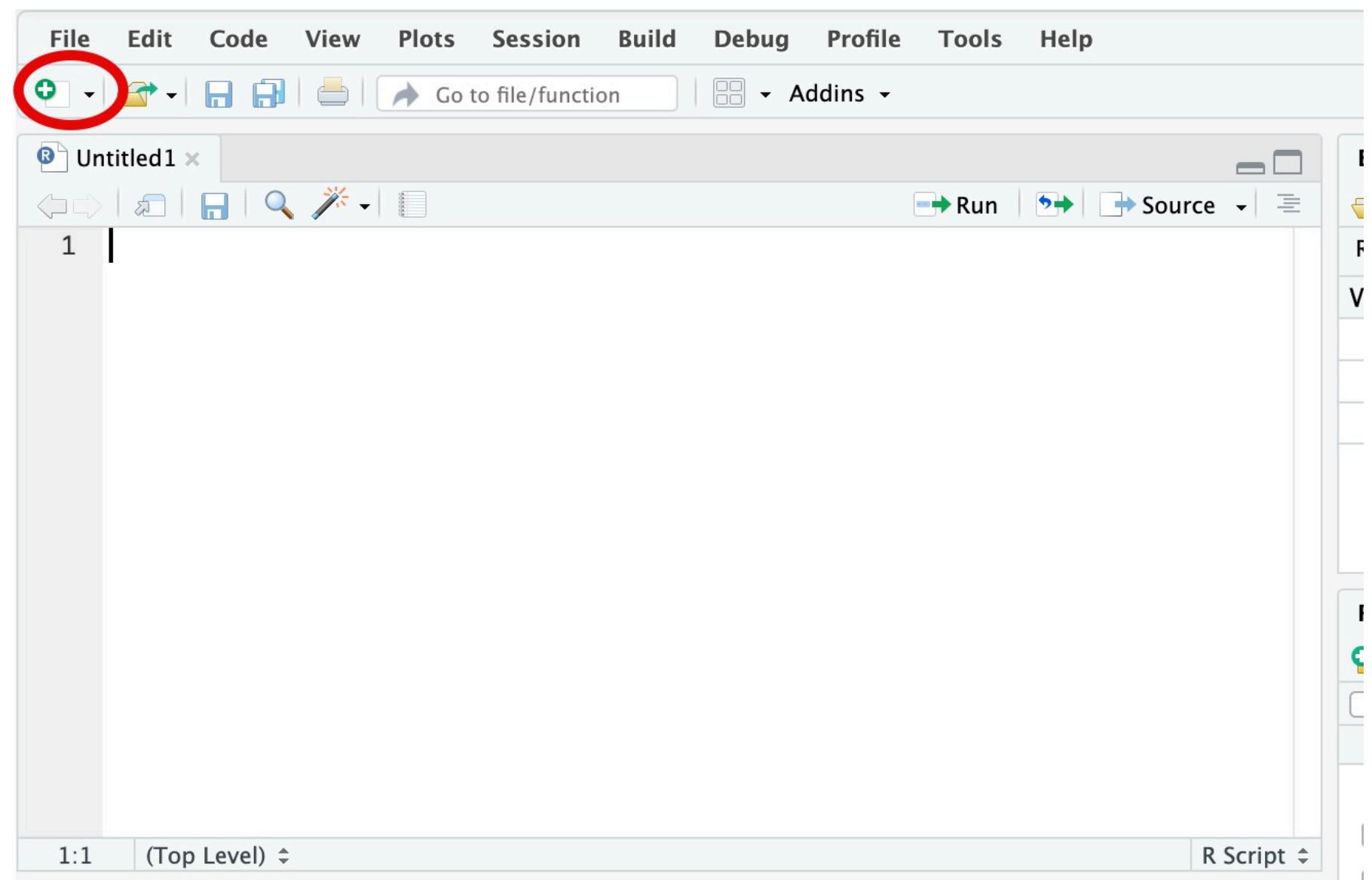
```
# Prompt user for two numbers
num1 <- as.numeric(readline(prompt = "Enter the first number: "))
num2 <- as.numeric(readline(prompt = "Enter the second number: "))
# Multiply the numbers
result <- num1 * num2
# Display the result
cat("The result of", num1, "*", num2, "=", result, "\n")
```

```
r
# Get two numbers from user input
num1 <- as.numeric(readline(prompt = "Enter first number: "))
num2 <- as.numeric(readline(prompt = "Enter second number: "))

# Multiply and display result
result <- num1 * num2
cat("Result:", num1, "*", num2, "=", result, "\n")
```

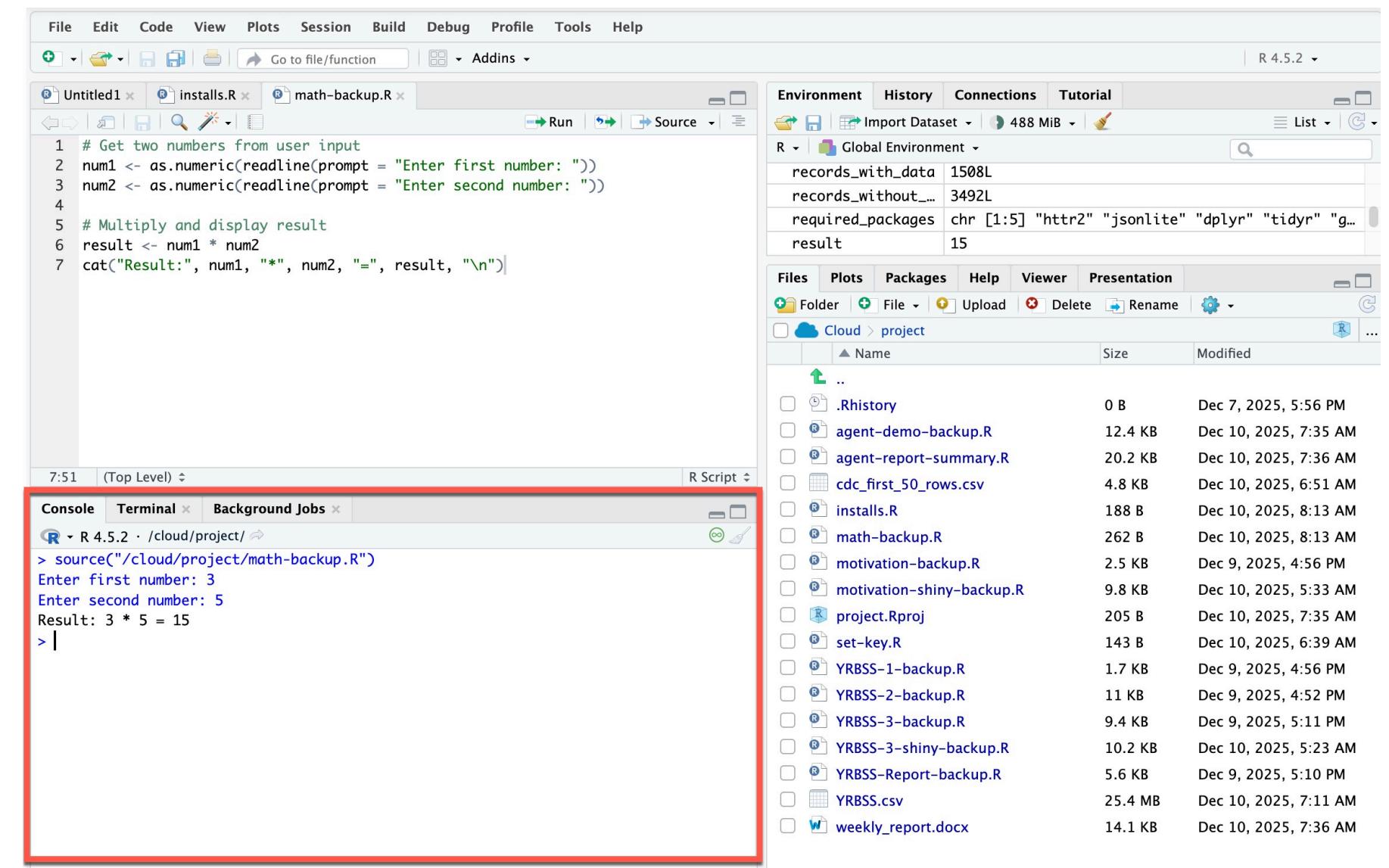
Create a new R script

- Copy the R code generated by Claude.
- In posit.cloud, select **R Script**. It is the first icon with a plus sign.
- Paste Claude code into posit.cloud
- File → Save As **Math**.
- Press **Source**



Enter values in the console

- Enter the values in the **Console**
- Review the result.



The screenshot shows the RStudio interface with the following components:

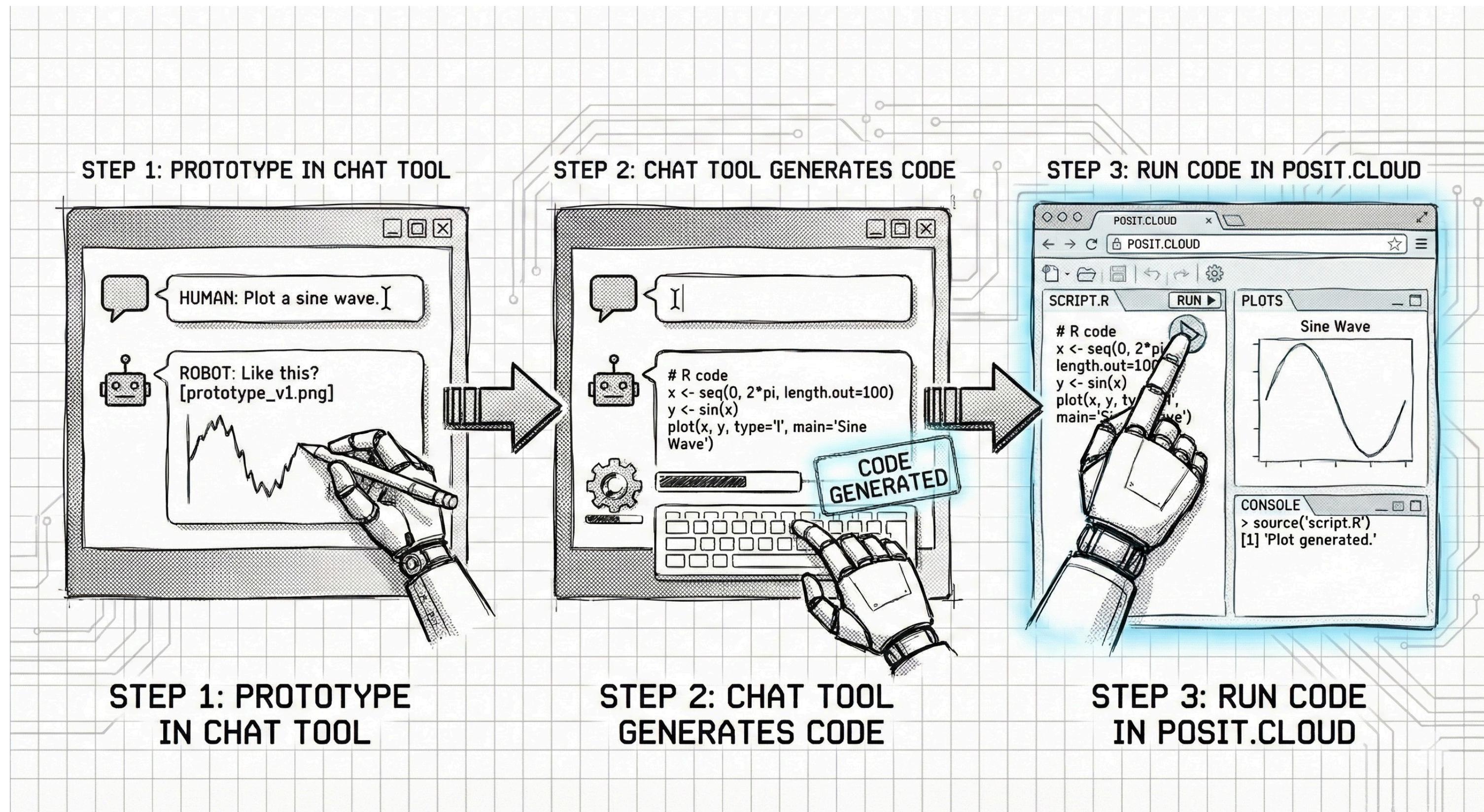
- File Bar:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Toolbar:** Go to file/function, Addins.
- Code Editor:** Untitled1, installs.R, math-backup.R. The math-backup.R code is:

```
1 # Get two numbers from user input
2 num1 <- as.numeric(readline(prompt = "Enter first number: "))
3 num2 <- as.numeric(readline(prompt = "Enter second number: "))
4
5 # Multiply and display result
6 result <- num1 * num2
7 cat("Result:", num1, "*", num2, "=", result, "\n")
```

- Environment Tab:** Global Environment, records_with_data (1508L), records_without_(... (3492L), required_packages (chr [1:5] "httr2" "jsonlite" "dplyr" "tidyR" "g...), result (15).
- Files Tab:** Cloud > project, listing files like .Rhistory, installs.R, motivation-backup.R, etc.
- Console Tab:** The tab is highlighted with a red box. It shows the R session output:

```
R > source("/cloud/project/math-backup.R")
> Enter first number: 3
> Enter second number: 5
> Result: 3 * 5 = 15
> |
```

From Chat to Code



Prompt: Use the robot hand drawn style. Create an image of images to portray these three steps: prototype in chat tool, chat tool generates code, run code in posit.cloud

Bring in data from an external source

Prompt Instructions

1. Read the data from this url and show the first 10 lines. Do not do anything else.

https://raw.githubusercontent.com/jmcody2014/BA/refs/heads/main/diabetic_data1K.csv

Hide steps

Thinking about retrieving and displaying initial dataset rows from a CSV file.

Fetched https://raw.githubusercontent.com/jmcody2014/BA/refs/heads/main/diabetic_data1K.csv githubusercontent.com ↗

Thought process

The user asked me to show the first 10 lines of the data. Let me extract just the first 10 lines from this CSV data.

Here are the first 10 lines of the data:

```
encounter_id,patient_nbr,race,gender,age,time_in_hospital,num_lab_procedures,nur
2278392,8222157,Caucasian,Female,[0-10),1,41,0,1,250.83,?,?,None,No,No,NO
149190,55629189,Caucasian,Female,[10-20),3,59,0,18,276,250.01,255,None,Up,Yes,>30
64410,86047875,AfricanAmerican,Female,[20-30),2,11,5,13,648,250,V27,None,No,Yes,NO
500364,82442376,Caucasian,Male,[30-40),2,44,1,16,8,250.43,403,None,Up,Yes,NO
16680,42519267,Caucasian,Male,[40-50),1,51,0,8,197,157,250,None,Steady,Yes,NO
35754,82637451,Caucasian,Male,[50-60),3,31,6,16,414,411,250,None,Steady,Yes,>30
55842,84259809,Caucasian,Male,[60-70),4,70,1,21,414,411,V45,None,Steady,Yes,NO
63768,114882984,Caucasian,Male,[70-80),5,73,0,12,428,492,250,None,No,Yes,>30
12522,48330783,Caucasian,Female,[80-90),13,68,2,28,398,427,38,None,Steady,Yes,NO
```

What will go into a summary table

Prompt Instructions

2. *What would you put into a summary table for this data?*

What would you put into a basic summary table?

Catalogued dataset variables and identified key summary statistics.



For this diabetic dataset, a basic summary table would typically include:

Dataset Overview:

- Total number of records (rows)
- Total number of variables (columns)
- Missing value counts

Numeric Variables (time_in_hospital, num_lab_procedures, num_procedures, num_medications):

- Count
- Mean
- Standard deviation
- Min / Max



Create a summary table

Prompt Instructions

3. Create a dataset overview and the mean, min and max for the numeric values.
4. Create the summary table and save it as a word document.

=====

DATASET OVERVIEW

=====

Total Records: 105
Total Variables: 16
Numeric Variables: 4
Categorical Variables: 12

=====

=====

NUMERIC VARIABLES SUMMARY

=====

Variable	Mean	Min	Max

time_in_hospital	4.8	1	14
num_lab_procedures	46.5	2	87
num_procedures	1.6	0	6
num_medications	15.3	1	32

=====

Summary table in a Word document

Diabetic Data Summary Table

Dataset Overview

Metric	Value
Total Records	105
Total Variables	16
Numeric Variables	4
Categorical Variables	12

Numeric Variables Summary

Variable	Mean	Min	Max
time_in_hospital	4.8	1	14
num_lab_procedures	46.5	2	87
num_procedures	1.6	0	6
num_medications	15.3	1	32

Code from Claude

Prompt Instructions

5. Convert this process to R code that writes out a word doc.

The screenshot shows a Jupyter Notebook interface. At the top, there's a toolbar with icons for file operations. Below it, a header bar says "url data / Reading CSV data from GitHub URL". The main area contains the following content:

- To run locally:**

```
r  
source("create_summary.R")
```
- A note: "The script reads directly from the GitHub URL, but you can change the path to a local file if preferred."
- A "Create summary" button with "Code · R" below it.
- An "Open in R" button.
- Artifacts** section:
 - "Create summary" (Code · R)
 - "Diabetic summary" (Document · DOCX)
- Project content** section:
 - "url data" (Created by you)

R isn't available on this system, but here's the R code that will create the Word document. I'll save it as a file you can download and run locally:

A slide titled "2 steps" with a "Presented file" section containing the following text:

```
Devised R script for generating Word documents.
```

Here's the R script that creates the Word document. It uses the `officer` and `flextable` packages which are the standard approach for creating Word docs in R.

Key components:

- 1. Packages:** `officer` (Word doc creation) + `flextable` (formatted tables)
- 2. Process:**
 - Read CSV data
 - Calculate summary statistics
 - Create formatted tables with `flextable()`
 - Build Word doc with `read_docx()` and `body_add_*` functions
 - Save with `print(doc, target = "filename.docx")`

Working in posit.cloud

Instructions

6. Create a new R Script.
7. Copy the code generated by Claude into the new, empty script
8. File → Save As 'diabetes-summary'
9. Press **Source** to run the code.
10. Download the Word document (*diabetic_summary.doc*) and examine it.

The screenshot shows the posit.cloud RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The main workspace contains two R script tabs: 'diabetes-summary-backup.R' and 'installs.R'. The 'diabetes-summary-backup.R' tab contains R code for generating a summary report. The 'installs.R' tab is currently active. The bottom right corner shows the R version as 'R 4.5.2'. The bottom panel features a 'Console' tab with the command 'source("./cloud/project/diabetes-summary-backup.R")' and its output: 'Document saved to: diabetic_summary.docx'. To the right of the workspace is a 'Files' panel showing a project structure with files like '.Rhistory', 'agent-demo-backup.R', 'agent-report-summary.R', 'bethany1.R', 'cdc_first_50_rows.csv', 'create-data.R', 'diabetes-summary-backup.R', 'diabetic_summary.docx', and 'installs.R'. The bottom right of the interface shows a file list with details such as name, size, and modified date.

Exercise

Instructions:

1. In the same project, begin a new chat.
2. Remind Claude not to create or run any code until you say it is ok to do so.

3. Ask Claude to:

- a. Use this url to read the data -

https://raw.githubusercontent.com/jimcody2014/BA/refs/heads/main/diabetic_data101k.csv

- b. This is the same data but has 101K rows.
- c. Display the first 10 rows of data as a table.
- d. Explain how it would go about identifying anything important or interesting in the data.
- e. Ask you 3 important questions before proceeding. Remind Claude to ask them one at a time.
- f. Once you are satisfied, ask Claude to proceed with its analysis.
- g. **Note:** Within the tool, code will be written using python. Don't worry, we will have Claude convert to R.

4. Tell Claude you want the code converted to R and you will run it locally.
5. Copy and paste the code into a new posit.cloud R Script. Save as 'diabetes-analysis'.
6. Test to see if it runs. Ignore any warning messages at the end

Create code that uses machine learning

Instructions:

1. Create a new project named **Using ML**. All chats for this walkthrough will be in this project.
2. Add an instruction to the project. (Adjust as you see fit)

a. I am a public health professional with little technical and programming experience.

3. Ask Claude to:

a. Read the data from this url and tell me how many rows are available. Do not generate any other code until I tell you it is ok to do so.

https://raw.githubusercontent.com/jimcody2014/BA/refs/heads/main/diabetic_data101k.csv

b. Using the downloaded data, give me 3-4 suggestions about how I can go about predicting whether the next patient we encounter will be re-admitted.

c. Are any of these approaches non-ml algorithms?

d. For the ML suggestions, give me a few bullet points for each describing pros and cons.

e. What do you need to know from me in order to use XGBoost?

i. Mine returned 4 questions

ii. If you do not understand what Claude is asking for, ask it for an explanation.

f. Proceed with building the model.

i. **This will take a few minutes.**

g. Convert this code to R. **Do not try to run the R code.** Prepare a file for me and I will run it locally.

h. Look at the code to make sure it is usig the url to get the data. If it is not, tell Claude to use the url.

4. Copy and paste the code into posit.cloud. Name the file predict-readmit. Run the code