



Why Choose One When We All Are Friends: The Power of Multiple AI Tools in Clinical Trials

PHUSE SDE 2024 – CARY

Hari Jampala

Disclaimer

The content of this presentation is for knowledge sharing and information purpose only. The content is developed through research process. The information presented here represents the work of author and should not be considered in general as a views of author's organization.





An accomplished Drug Development and Statistical Programming Professional and leader, currently serving as Senior Manager of Statistical Programming at a Fortrea. With deep expertise in statistical programming within the pharmaceutical industry and CROs, this subject matter expert (SME) excels in CDISC activities, regulatory submissions, and process automation. A proven leader, manage remote teams, drive cross-functional process improvements, and lead projects to successful completion. Equipped with a strong working knowledge of clinical trial phases, proficient in SAS R, Python, and PowerBI.

Harivardhan.jampala@fortrea.com

Embracing AI: From Caution to Collaboration in Statistical Programming

Industry Estimates: Overall, there may be an estimated 15,000 to 25,000 statistical programmers employed worldwide across both sectors. This number accounts for roles like biostatistics, data management, and other related functions that often require similar programming skills.

SPSS → SAS → Opensource(R, Python, Scala, Julia, java script) → AI → GENAI

Imagine what is the mindset of the programmers?

Help in changing the mindset to reshaping workflows rather than replacing the need for skilled programming professionals.

Enhanced Efficiency and Speed

Focus on Practical, Non-Coding Applications of AI

Emphasize AI as a Tool, Not a Replacement

Integrate AI into Existing Workflows Gradually

Promote Collaborative AI Development Projects

Highlight Career Growth Opportunities with AI

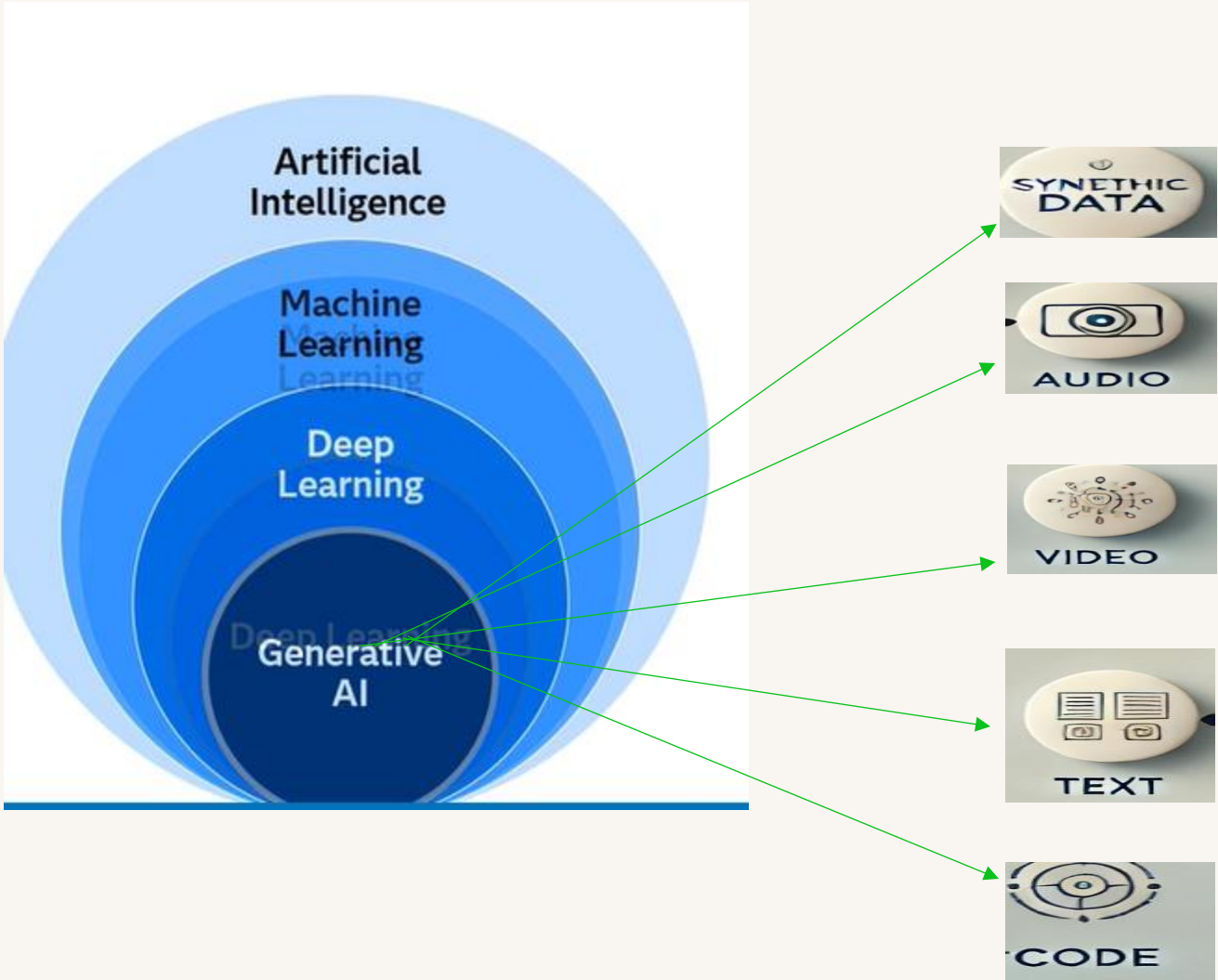
Asking them to use some Free and popular
LLMs/GenAI

Forecast,
Recommendation,
Identifications,
Detecting, Optimize

Generate
CONTENT/DATA

GENAI/ LLM

GENAI – Transforms the interaction between HI and AI



LARGE LANGUAGE MODEL (LLM)

LARGE = Learns from hundreds of billions of data

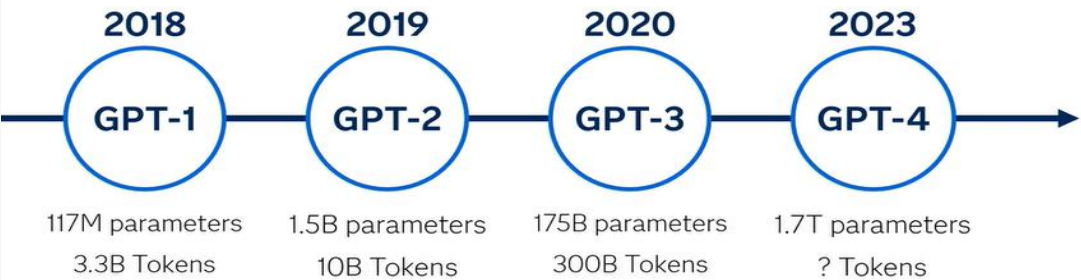
LANGUAGE = Understands human natural language



Timeline

GPT

Published by OpenAI in partnership with Microsoft



GENAI is much more than ChatGPT.

Opportunities

Clinical research

Dose range findings

Drug Discovery – Drug target identifications, selections, compound screening and design

Site selection, recruitment, retention, adherence,

Non-clinical – PK/PD and Dose range findings

Data collection, management, analysis

Clinical end point analysis

Manufacturing and post market safety monitoring-
advanced pharmaceutical manufacturing, post
market safety surveillance or pharmacovigilance

Biostats and Statistical programming:

CRF and edit check configurations

Validation and UAT

Data analysis and visualization

Statistical programming

Draft text like for SAP, CSR

Draft codes documentation or creation

Documentation and TMF

How? Your New Programming Partner: How AI Enhances, Not Replaces

HOW? Roadmap might be

- Understand Artificial Intelligence Basics
- Learn Math
- Learn Programming Language
- Learn Big Data
- Learn Data Science
- Learn Machine Learning Algorithms
- Learn Deep Learning Algorithms
- Learn Large Language Models



Highlight the Competitive Advantage



Encourage Small, Low-Stakes Experiments

LinkedIn -> 70% of this year list did not exist 20 years ago -> next 10 years ? -> 20 don't exist

Encourage Small, Low-Stakes Experiments

4. Scala (Spark)

In Scala, particularly with Apache Spark, you use `join` on DataFrames. This example assumes `df1` and `df2` are Spark DataFrames.

scala

Copy code

```
import org.apache.spark.sql.SparkSession
import org.apache.spark.sql.functions._

val spark = SparkSession.builder.appName("MergeExample").getOrCreate()
val merged_data = df1.join(df2, Seq("ID"), "inner")
```

5. Julia

In Julia, you can use the `DataFrames.jl` package, which has a `join` function for merging.

julia

Copy code

```
using DataFrames

# Assuming df1 and df2 are DataFrames
merged_data = innerjoin(df1, df2, on = :ID)
```

Summary

Language	Function/Method	Syntax
SAS	<code>MERGE</code> or <code>PROC SQL JOIN</code>	<code>MERGE df1(in=a) df2(in=b);</code>
R	<code>merge</code> or <code>dplyr::inner_join</code>	<code>merge(df1, df2, by = "ID")</code>
Python	<code>pandas.merge</code>	<code>pd.merge(df1, df2, on="ID")</code>
Scala	<code>Spark DataFrame.join</code>	<code>df1.join(df2, Seq("ID"), "inner")</code>
Julia	<code>DataFrames.innerjoin</code>	<code>innerjoin(df1, df2, on = :ID)</code>

Encourage Small, Low-Stakes Experiments

Gemini/ Bard

Gemini ▾

Try Gemini Advanced



- R and Python have extensive ecosystems of packages for data manipulation, making them highly flexible for complex merge operations. ▾
- SAS and Julia also offer powerful tools for data manipulation, but their ecosystems might be less extensive in some areas.

Additional Considerations:

- **Data Cleaning and Preparation:** Ensure that the data types of the join columns are consistent across datasets before merging.
- **Handling Duplicates:** Consider how to handle duplicate values in the join column. Some languages offer options to keep all, first, or last occurrences.
- **Performance Optimization:** For large datasets, consider techniques like partitioning or parallel processing to improve performance.

By understanding these core concepts and the specific syntax of each language, you can effectively merge datasets to create new insights from your data.

Check completed • [Understand the results](#)



Search related topics

- 🔍 Compare SAS, R, Python and Julia for merging 2 datasets
- 🔍 What is the comparison between SAS R and Python?
- 🔍 How to combine two datasets into one Python?

Ask Gemini



Other LLMs?

ctcae_v5_quick_reference_5x7.pdf

Source guide

Common Terminology Criteria for Adverse Events (CTCAE) v5.0 Publish Date: November 27, 2017

Introduction The NCI Common Terminology Criteria for Adverse Events is a descriptive terminology which can be utilized for Adverse Event (AE) reporting. A grading (severity) scale is provided for each AE term.

SOC System Organ Class (SOC), the highest level of the MedDRA1 hierarchy, is identified by anatomical or physiological system, etiology, or purpose (e.g., SOC Investigations for laboratory test results). CTCAE terms are grouped by MedDRA Primary SOCs. Within each SOC, AEs are listed and accompanied by descriptions of severity (Grade).

CTCAE Terms An Adverse Event (AE) is any unfavorable and unintended sign (including an abnormal laboratory finding), symptom, or disease temporally associated with the use of a medical treatment or procedure that may or may not be considered related to the medical treatment or procedure. An AE is a term that is a unique representation of a specific event used for medical documentation and scientific analyses. Each CTCAE v4.0 term is a MedDRA LLT (Lowest Level Term).

1 CTCAE v5.0 incorporates certain elements of the MedDRA terminology. For further details on MedDRA refer to the MedDRA MSSO Web site (<https://www.meddra.org/>).

Grades Grade refers to the severity of the AE. The CTCAE displays Grades 1 through 5 with unique clinical descriptions of severity for each AE based on this general guideline:

Grade 1 Mild; asymptomatic or mild symptoms; clinical or diagnostic observations only; intervention not indicated.

Grade 2 Moderate; minimal, local or noninvasive intervention indicated; limiting age-appropriate instrumental ADL*.

Grade 3 Severe or medically significant but not immediately life-threatening; hospitalization or prolongation of hospitalization indicated; disabling; limiting self care ADL.**

Grade 4 Life-threatening consequences; urgent intervention indicated.

Grade 5 Death related to AE.

A Semi-colon indicates 'or' within the description of the grade.

A single dash (-) indicates a Grade is not available. Not all Grades are appropriate for all AEs. Therefore, some AEs are listed with fewer than five options for Grade selection.

Grade 5 Grade 5 (Death) is not appropriate for some AEs and therefore is not an option.

Definitions A brief Definition is provided to clarify the meaning of each AE term. A single dash (-) indicates a Definition is not available.

Navigation Notes A Navigational Note is used to assist the reporter in choosing a correct AE.

Untitled notebook

Settings

Share

How does the CTCAE categorize adverse events based on their severity?

The Common Terminology Criteria for Adverse Events (CTCAE) categorizes adverse events based on severity using a 5-point grading scale, with Grade 1 being the least severe and Grade 5 the most severe 1 2. Here is a breakdown of the general guidelines for each grade:

- **Grade 1 (Mild):** Asymptomatic or mild symptoms; clinical or diagnostic observations only; intervention not indicated 2
- **Grade 2 (Moderate):** Minimal, local, or non-invasive intervention indicated; limiting age-appropriate instrumental Activities of Daily Living (ADL) 2 3
- **Grade 3 (Severe):** Severe or medically significant but not immediately life-threatening; hospitalization or prolongation of existing hospitalization indicated; disabling; limiting self-care ADL 2
- **Grade 4 (Life-Threatening):** Life-threatening consequences; urgent intervention indicated 2 4
- **Grade 5 (Death):** Death related to AE 4

It's important to note that not all adverse events have all 5 grades. Some AEs may have fewer grading options based on their nature and potential severity. For instance, Grade 5 (Death) may not be an option for all AEs 4.

The CTCAE provides detailed descriptions of each grade for every specific adverse event listed. These descriptions outline the clinical presentation and the level of intervention required for each severity level, providing a standardized framework for evaluating and documenting adverse events in clinical trials and practice. For example, looking at the adverse event "Anemia" under the "Blood and lymphatic system disorders" category, the CTCAE provides specific hemoglobin level ranges for each grade, ranging from Grade 1 (<LLN - 10.0 g/dL) to Grade 4 (Life-threatening consequences; urgent intervention indicated) 5.

The CTCAE's use of a standardized grading system for adverse event severity is crucial for:

- **Consistent Reporting:** This allows for reliable comparison of AE data across different clinical studies and settings.
- **Effective Communication:** It provides a common language for healthcare professionals to discuss and manage AEs.
- **Enhanced Patient Safety:** The standardized grading system enables timely recognition and intervention for serious AEs.



Save to note

How does the CTCAE categorize adverse events based on their system organ class?

What are the two primary types of layouts that can be used for p

Close Chat

4 sources

Start typing...



Notebook guide

NotebookLM may still sometimes give inaccurate responses, so you may want to confirm any facts independently.

Other LLMs?

Untitled notebook

 Add note  Select all

 Written Note

New Note

CTCAE grades adverse event

Other LLMs?

Source guide

- 2 Post
- 3 LinkedIn
- 4 Email
- 5 Print

Quick Links

- 1 eCTD Resources
- 2 Data Standards Catalog
- 3 Study Data Technical Conformance Guide

This page provides quick links to key guidances to support the submission of study data to FDA’s Center for Biologics Evaluation and Research (CBER), Center for Drug Evaluation and Research (CDER), Center for Devices and Radiological Health (CDRH), and Center for Veterinary Medicine (CVM) and provides a common site where guidances and technical documents related to study data standards are displayed together. Every guidance listed below may not apply to all centers. Each link provides more complete information on the document.

CDER and CBER, in collaboration with CDISC and PhUSE, has conducted preliminary testing of CDISC’s Dataset JSON message exchange standard. Initial results indicate potential use as a replacement for XPT v5. As such, CBER and CDER will conduct further testing to evaluate Dataset JSON’s capability to support the submission of regulatory study data. Results will be communicated, and we will engage relevant parties for input as we progress through this evaluation.

Guidances

- 1 (745A(a)) - Providing Submissions in Electronic Format Under Section 745A(a) of the FD&C Act
- 2 (eStudy) - Providing Submissions in Electronic Format - Standardized Study Data
- 3 (eCTD) - Providing Submissions in Electronic Format using the eCTD Specifications
- 4 (eCTD) - Submission Standards for eCTD v3.2.2 and Regional M1
- 5 (RWD) - Data Standards for Drug and Biological Product Submissions Containing Real-World Data
- 6 (ICSR) - Providing Regulatory Submissions in Electronic Format - IND Safety Reports
- 7 (BIMO) - Standardized Format for Electronic Submission for Marketing Applications Content for the Planning of Bioresearch Monitoring Inspections for CBER Submissions
- 8 (BIMO) - Standardized Format for Electronic Submission of NDA and BLA Content for the Planning of Bioresearch Monitoring Inspections for CDER Submissions

Resources for Guidance and Support:

The FDA provides numerous resources to assist researchers and sponsors in meeting data submission requirements:

- **Guidances:** Comprehensive documents providing instructions on various aspects of data submission, including electronic format requirements, standardized study data, and specific considerations for different types of submissions. 4
- **Technical Conformance Guide:** A detailed guide outlining technical specifications for data formatting, validation rules, and terminology standards. 1 3 4
- **Data Standards Catalog:** A repository of data standards and terminologies used by the FDA, aiding in the standardization of submitted data. 1 4

- **Technical Specifications:** Methods Templates, and data elements, such as Bioanalytical
- **Contact Information:** CDER and CBER, in collaboration with CDISC and PhUSE, has conducted preliminary testing of CDISC’s Dataset JSON message exchange standard. Initial results indicate potential use as a replacement for XPT v5. As such, CBER and CDER will conduct further testing to evaluate Dataset JSON’s capability to support the submission of regulatory study data. Results will be communicated, and we will engage relevant parties for input as we progress through this evaluation.

Benefits of Adhering

By adhering to FDA data

- **Enhance Data Quality:** Ensures the accuracy, completeness, and reliability of submitted data.
- **Facilitate Efficient Review:** Streamlines the regulatory review process, expediting the regulatory review and interpretation, leading to more
- **Support Meaningful Engagement:** Enables communication and collaboration among researchers, sponsors, and FDA
- **Improve Communication:** Facilitates the exchange of information between regulatory reviewers and sponsors, ensuring that their data is in the regulatory review process. By adhering to FDA data standards, sponsors can ensure that their submissions meet regulatory requirements.

The FDA’s emphasis on leveraging the available resources ensures that study data submissions meet regulatory requirements.

CDER and CBER, in collaboration with CDISC and PhUSE, has conducted preliminary testing of CDISC’s Dataset JSON message exchange standard. Initial results indicate potential use as a replacement for XPT v5. As such, CBER and CDER will conduct further testing to evaluate Dataset JSON’s capability to support the submission of regulatory study data. Results will be communicated, and we will engage relevant parties for input as we progress through this evaluation.

Guidances

- 1 (745A(a)) - Providing Submissions in Electronic Format Under Section 745A(a) of the FD&C Act
- 2 (eStudy) - Providing Submissions in Electronic Format - Standardized Study Data
- 3 (eCTD) - Providing Submissions in Electronic Format using the eCTD Specifications

Save to note

What are the two main formats for study data that the FDA uses?

What is the purpose of the FDA Validator Rules?

How can individuals submit comments?

Close Chat

2 sources

Start typing...



Notebook guide

NotebookLM may still sometimes give inaccurate responses, so you may want to confirm any facts independently.

Other LLMs?

Microsoft Phi3.5 - <https://ollama.com/library/phi3.5>

A lightweight AI model with 3.8 billion parameters with performance overtaking similarly and larger sized models.

Phi-3.5 models have a context length of up to 128K, allowing them to handle **long-context tasks effectively**. This can be beneficial for processing lengthy clinical research documents or patient records.

Phi -

3.5 models have shown superior performance in reasoning and math benchmarks compared to other small models, This makes them highly efficient and capable of handling complex tasks with reduced latency

Co-Pilots, Labs, and Tutors

Microsoft Co-pilot

Similarly, you can download a study protocol and start asking the study related questions by leveraging an internal AI engine?

- **Protocol Amendments:** Quickly identify the implications of proposed changes.
 - **Adverse Event Queries:** Ask specific questions about how to handle adverse events according to the protocol.
- Click to add first slide
- **Visit Schedules:** Retrieve details on patient visit schedules and procedures.
 - **Inclusion/Exclusion Criteria:** Clarify patient eligibility criteria efficiently.

Copilot

Create presentation from file
[General-Practical-Guide-to-Time-Sheets-in-Workday.docx](#)

Here's the outline for the presentation I'm generating:

- General Practical Guide to Time Sheets in Workday
- Introduction to Time Sheet Submission
- Accessing Workday for Time Sheets
- Single Entry Submission
- Multiple Entries Submission
- Adjustments and Corrections
- Time Sheet Adjustment Process
- Timesheet Template Guide
- Activities and Sub-activities
- Importance of Time Reporting
- Time Reporting Do's
- Time Reporting Don'ts
- Conclusion

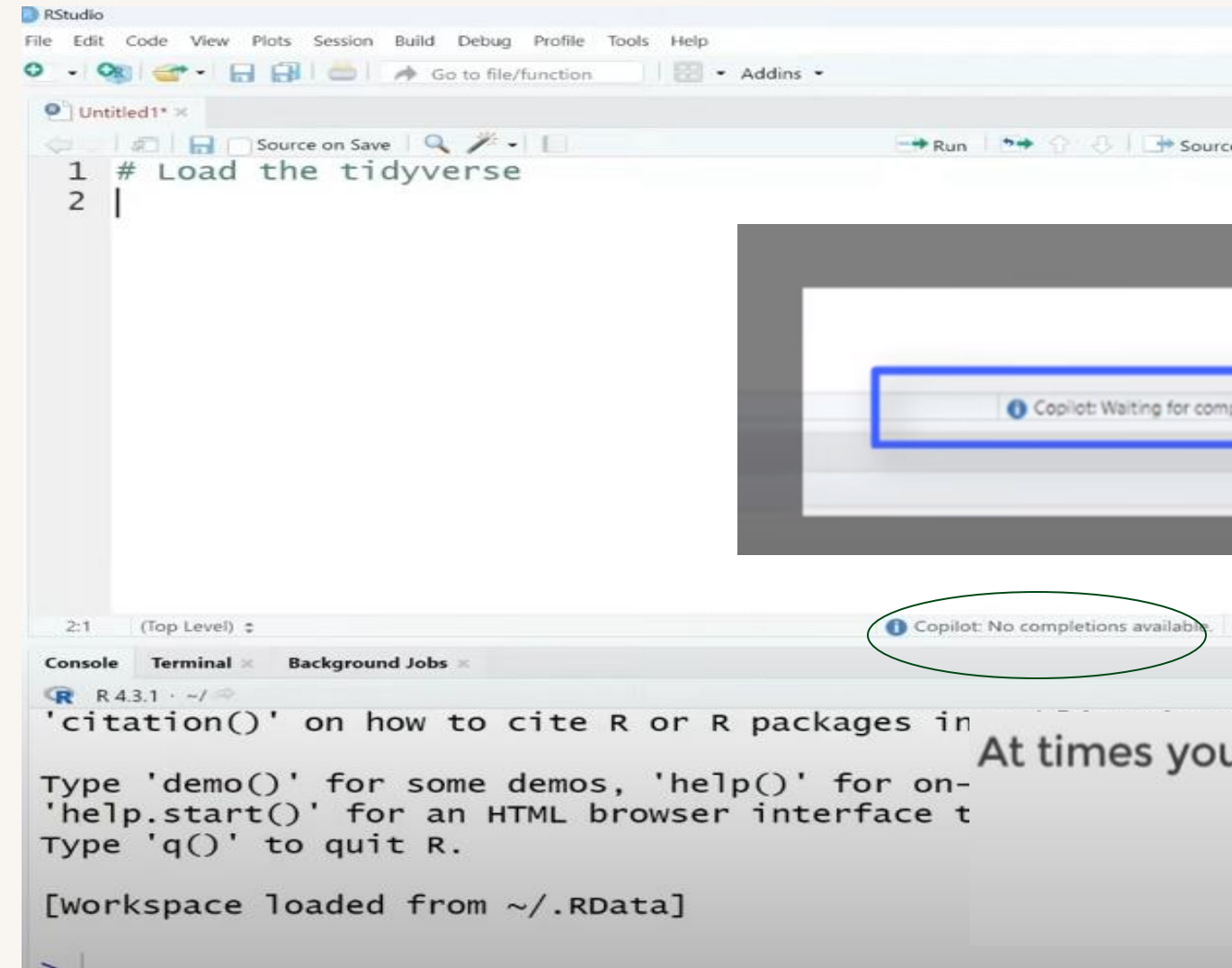
Generating slides (29%)...

Stop generating

Ask me anything about this presentation

Co-Pilots, Labs, and Tutors

Github Copilot



Copilot suggestions appear in grey
italics font
Press Tab key to accept or ENTER
to ignore

Environment is empty

Copilot: No completions available

At times you might not get suggestions immediately, you can try
Wait for few seconds or
Press ENTER or
try typing code

Co-Pilots, Labs, and Tutors

1 # Load the tidyverse
2 library(tidyverse)
3
4 library(chattr)
5 chattr_app()
6

Environment: Global Environment
Environment is empty

Files Plots Packages Help Viewer Presentation

Can you show me an example of a ggplot?

The best package for data visualization in R is ggplot2.

6.1 (Top Level) 1
Copilot: No completions available R Script 1

Console Terminal Background Jobs

R 4.3.1 ~ /
> chattr_app()
• Provider: Open AI - Chat Completions
• Path/URL: https://api.openai.com/v1/chat/completions
• Model: gpt-3.5-turbo
Loading required package: shiny
Warning: package 'shiny' was built under R version 4.3.2
Listening on http://127.0.0.1:6721

Files Plots Packages Help Viewer Presentation

show me an example of a ggplot2

Submit GPT 4 (OpenAI)

Sure, here is a simple example of a ggplot2 plot. This example uses the `mtcars` dataset which is built into R. It creates a scatter plot of `g` (miles per gallon) against `hp` (horsepower).

library(ggplot2)

ggplot(mtcars, aes(x = hp, y = mpg)) geom_point() +
labs(x = "Horsepower", y = "Miles per Gallon", title = "Scatterplot of MP

This code first calls `ggplot()` to create a new ggplot. The first argument is the dataset that we're using, `mtcars`. The second argument is the aesthetic `aes()`, which defines how variables in the data are mapped to visual properties. In this, we're mapping `hp` to the x-axis and `mpg` to the y-axis.

`geom_point()` adds a layer of points to the plot, creating a scatter plot.

Finally, `labs()` adds labels to the x-axis, y-axis, and the plot as a whole.

Co-Pilots, Labs, and Tutors

SAS CO pilot

Generate code

< 1/2 > Accept Tab Accept Word ⌨ → ...

```
/* Convert 'potassiumchar' to a numeric variable 'potassium' in kidneyTest */  
data kidneyTest;  
  set kidneyTest;  
  potassium = input(potassiumchar, best12.);  
  drop potassiumchar;  
run;
```

Add comments | Explain

data kidneyTest;
 set kidneyTest;
 potassium = input(potassiumchar, best12.);
 drop potassiumchar;
run;

Run Selected or All SAS Code ⌘⇧↵

Run All SAS Code F8

Run Region

Change All Occurrences ⌘F2

Format Document ⌘⇧F

Refactor... ⌘⇧R

Cut ⌘X

Copy ⌘C

Paste ⌘V

SAS Viya Copilot: Add comments

SAS Viya Copilot: Explain

SAS Viya Copilot: Generate code

Command Palette... ⌘⇧P

SAS and Generative AI

 SAS Viya

Data and AI Platform

Leverage LLMs at an Enterprise Level
SAS Viya enables orchestration of LLMs usage, integration with decisioning workflows, machine learning and data protection capabilities.



Copilots*

Accelerate AI Lifecycle Tasks
Leveraging Copilots that democratize ALC tasks, for increased user productivity - generate code, generate dashboards, run data quality checks.



AI Intelligent Assistants*

Execute Business/Industry Tasks
Empower individuals to do their job better and faster using AI assistants specialized in enterprise use cases across different industries.

SAS VIYA COPILOT: CHAT

 You

Explain the code.

 SAS Viya Copilot

The provided SAS code uses the PROC LIFETEST procedure to perform survival analysis on pharmaceutical data. The DATA statement specifies the input dataset "mycas.pharma" which contains the data for the analysis. The TIME statement defines the time-to-event variable as "Time_to_Event" and the event occurrence indicator as "Event_Occurred". By adding "(0)" after the variable names, it specifies that the dataset already contains the survival time and censoring information. The STRATA statement is used to stratify the data based on the "Treatment_Group" variable. This allows for separate survival analyses to be performed for each treatment group. The RUN statement signals the end of the PROC LIFETEST block. Overall, this code is used to analyze the survival times and events in the pharmaceutical data, stratifying the analysis by treatment group.

Co-Pilots, Labs, and Tutors

RTutor

Home

EDA

Report

More ▾

RTutor

Home

EDA

Report

About

Dataset: uploaded.

Reset

price बनाम carat का स्कैटर प्लॉट बनाएं। clarity द्वारा रंग बदलें

Submit

Settings

☐ Python

Ask a question on the code or statistics

AI generated code

Chunk #17

ggplot(df, aes(> geom_point(alp theme_minimal()

Results:

15000

audi a4 quattro

audi a4 quattro

Dataset: uploaded.

Reset

price बनाम carat का स्कैटर प्लॉट बनाएं। clarity द्वारा रंग बदलें

Submit

Settings

☐ Python

इस कोड को समझाइये

Chat with your tutor

इस कोड को समझाइये

यह कोड डायमंड डेटा को विश्लेषण करने के लिए उपयोग किया जा सकता है। इसमें ggplot2 पैकेज का उपयोग किया गया है जो रंगीन और ग्राफिकल विजुअलाइज़ेशन के लिए एक शक्तिशाली पैकेज है। यह डायमंड के कैरेट और मूल्य के बीच एक स्कैटर प्लॉट बनाता है, जहाँ clarity द्वारा रंग बदला जाता है। यह आपको डायमंड के कैरेट और मूल्य के बीच किसी भी संबंध या पैटर्न की पहचान करने में मदद कर सकता है।

इस कोड को समझाइये

यह कोड डेटा विश्लेषण के लिए रचा गया है। ggplot2 पैकेज का उपयोग करके एक स्कैटर प्लॉट बनाने के लिए है। इसमें आपके डेटा फ्रेम df का उपयोग किया जाता है। आपके डेटा में 'carat' और 'price' के बीच संबंध को देखना है।

Results:

ggplot(df, aes(x = carat, y = price, color = clarity)) + geom_point(alpha = 0.5) + theme_minimal()

price

15000

10000

5000

0

clarity

IF

S11

S12

VS1

VS2

VVS1

VVS2

compact

compact

24

Next

Co-Pilots, Labs, and Tutors

 Generate

Write a code for merging 2 datasets with a `innerjoin`



Close

< 1 of 1 >   [Use code with caution](#)

 # prompt: Write a code for merging 2 datasets with a `innerjoin`

```
import pandas as pd

# Sample data for dataset 1
data1 = {'ID': [1, 2, 3, 4, 5],
        'Name': ['Alice', 'Bob', 'Charlie', 'David', 'Eve'],
        'Age': [25, 30, 22, 28, 35]}
df1 = pd.DataFrame(data1)

# Sample data for dataset 2
data2 = {'ID': [2, 3, 5, 6, 7],
        'City': ['New York', 'London', 'Paris', 'Tokyo', 'Sydney'],
        'Country': ['USA', 'UK', 'France', 'Japan', 'Australia']}
df2 = pd.DataFrame(data2)

# Merge the two datasets using an inner join on the 'ID' column
merged_df = pd.merge(df1, df2, on='ID', how='inner')

# Print the merged dataset
merged_df
```



	ID	Name	Age	City	Country
0	2	Bob	30	New York	USA
1	3	Charlie	22	London	UK
2	5	Eve	35	Paris	France



Next steps: [Generate code with merged_df](#) [View recommended plots](#) [New interactive sheet](#)

CoLab offers Python code by default, and if you're interested, you can also integrate a SAS plugin to learn SAS code. There are resources available that explain how to add SAS. In the same session, you can ask questions related to both Python and SAS.

Power of Multiple AI Tools - AI Benefits in Clinical Statistical Programming

ClinicalTrials.gov

PubMed

These advancements can save time, reduce frustration, and enhance overall efficiency in clinical statistical programming. What do you think?

Real-Time Adverse Event Monitoring:

Understanding Relationships in Datasets:

1. Utilize the Rave database to live stream data, detecting any **abnormalities in adverse events** as they happen.
1. Map out **connections** between the Rave **database** and various dataset levels, as well as **CRF** (Case Report Form) forms, to gain a clearer understanding of data **relationships**.

Patient Journey Projections with GENAI:

Operational Efficiency with GENAI:

1. Employ General AI (GENAI) to simulate and predict **patient behavior and outcomes** using sigmoid methods, enhancing patient journey insights.
1. Utilize GENAI to reveal how each CRF form relates to others, and provide **keys** to connect each form, offering a **summary** of these connections.

Streamlined Utility Identification for Pharma/CROs:

Live Database Updates and Dashboards:

1. Implement a dashboard that updates the **data management** and **statistical programming** teams in real-time when the live database is updated, ensuring everyone is informed of the **latest changes** and key data points.
1. Contract Research Organizations (CROs) often juggle **multiple studies**, making it challenging to find **specific utilities**. Instead of relying solely on Subject Matter Experts (SMEs), create **Internal LLMs to provide scenario-based recommendations**. The model can suggest appropriate macros and reference studies, significantly reducing time and effort.

Provide reasoning and Compares potentially **impacted SOPs** with **new regulations**?

LLMs Can I create?

Dify

Star

50,713

Explore

Studio / Code Converter

Knowledge

Tools

Upgrade

Dify

Code Converter

COMPLETIONBASIC

Orchestrate

API Access

Logs & Ann.

Monitoring

Orchestrate

Prefix Prompt

Providing translation capabilities in multiple programming languages, translating the user's input code into the programming language they need. Please translate the following code snippet to {{Target_code}}: When the information entered by the user is not a code snippet, prompt: Please enter a valid code snippet. {{default_input}}

332

Variables

VARIABLE KEY	USER INPUT FIELD NAME	OPTIONAL	ACTIONS
Target_code	Language	☐	
default_input	default_input	☐	

Context

You can import Knowledge as context

Debug & Preview

USER INPUT FIELD

Fill in the value of the variable, which will be automatically replaced in the prompt words every time a question is submitted.

Language

Python

default_input

DATA merged_data:
MERGE dataset1 dataset2;
BY ID;
RUN;

Clear

RUN

Enable feature to enhance web app user experience

OUTPUT TEXT

merged_data = None
dataset1 = None
dataset2 = None

def merge_datasets():
 global merged_data, dataset1, dataset2
 merged_data = dataset1.merge(dataset2, on='ID')

merge_datasets()

Log

Copy

186 chars

©2024 Fortrea Inc. All rights reserved. Fortrea.com

21

Fortrea

Regulatory Authorities

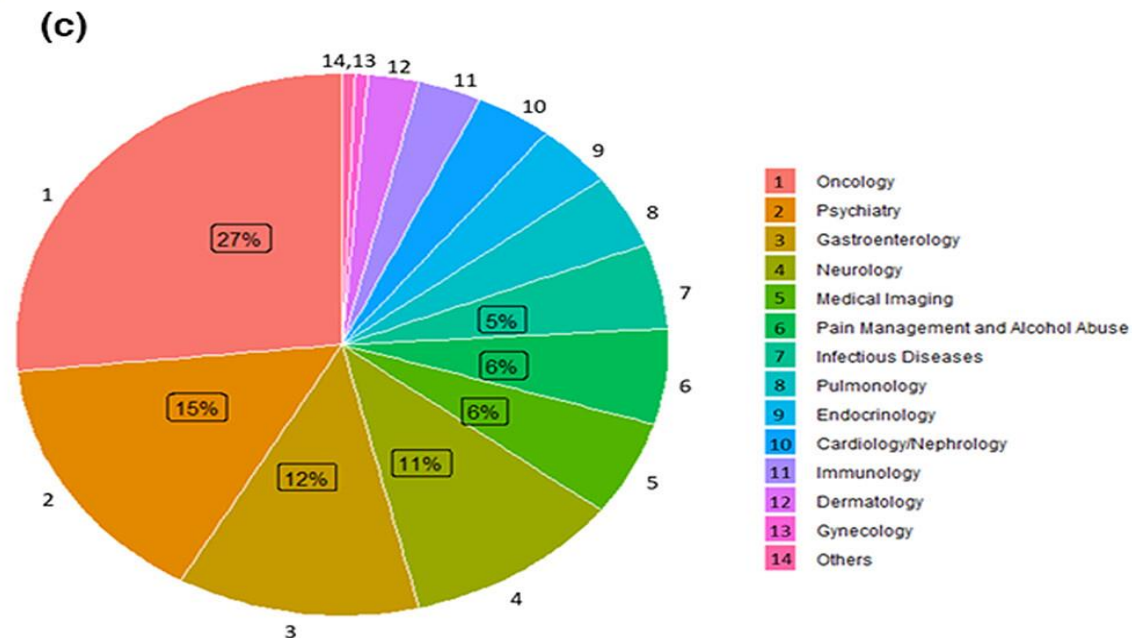
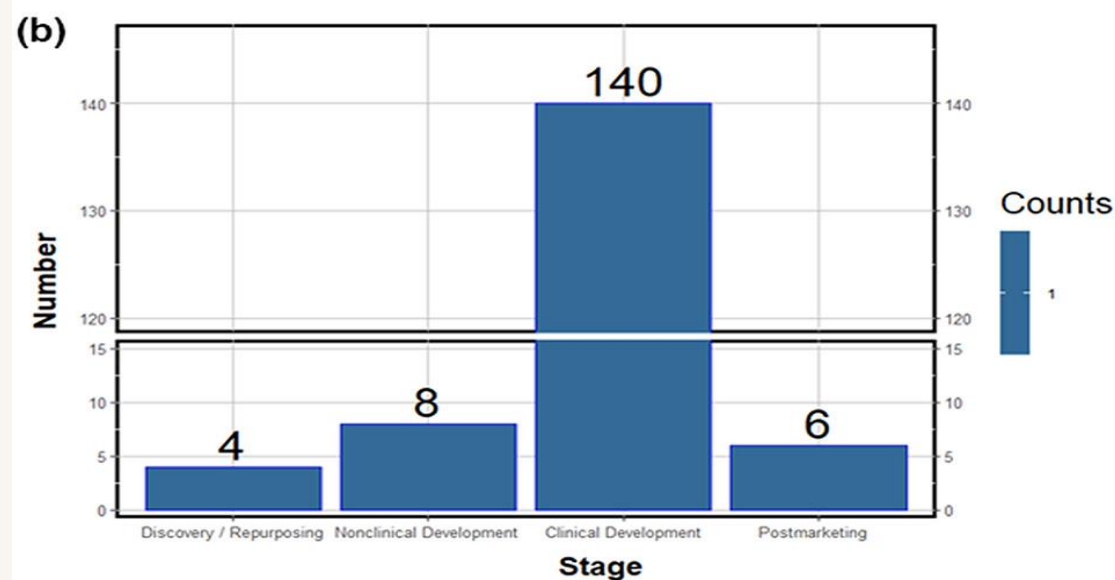
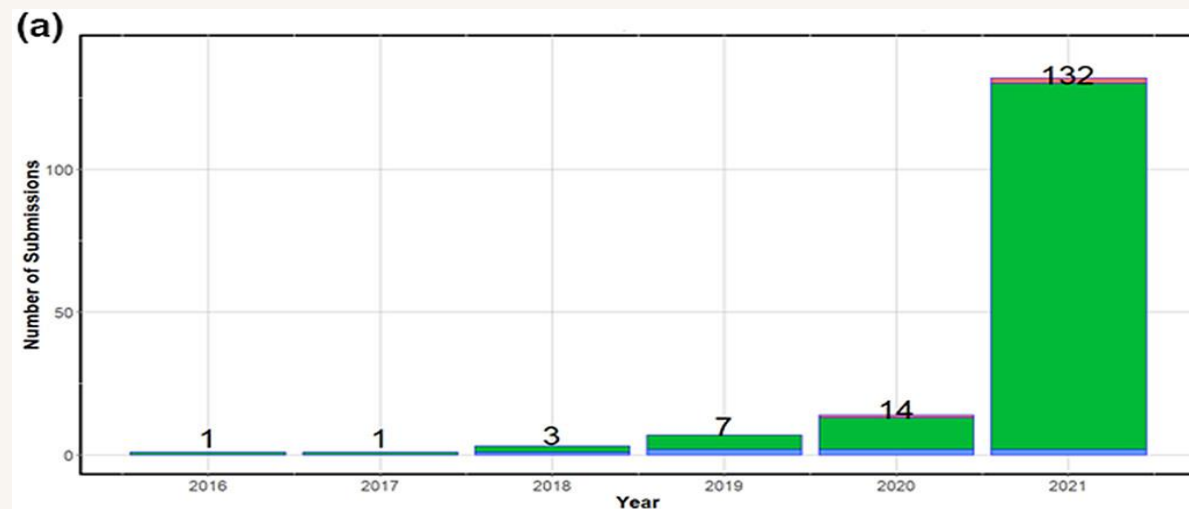
[FDA's drug center to consolidate AI efforts under single council](#)

[\(fiercebiotech.com\)](#) - **FDA's drug center to consolidate AI efforts under single council**

“The advent of novel AI applications, such as generative AI and large language models, is likely to expand opportunities for use of AI within CDER, including by non-technical staff. This will require increased AI education and coordination,” she said. “There has also been a rapid increase in CDER regulatory submissions incorporating AI, and the scope and impact of AI use in drug development are expanding.”

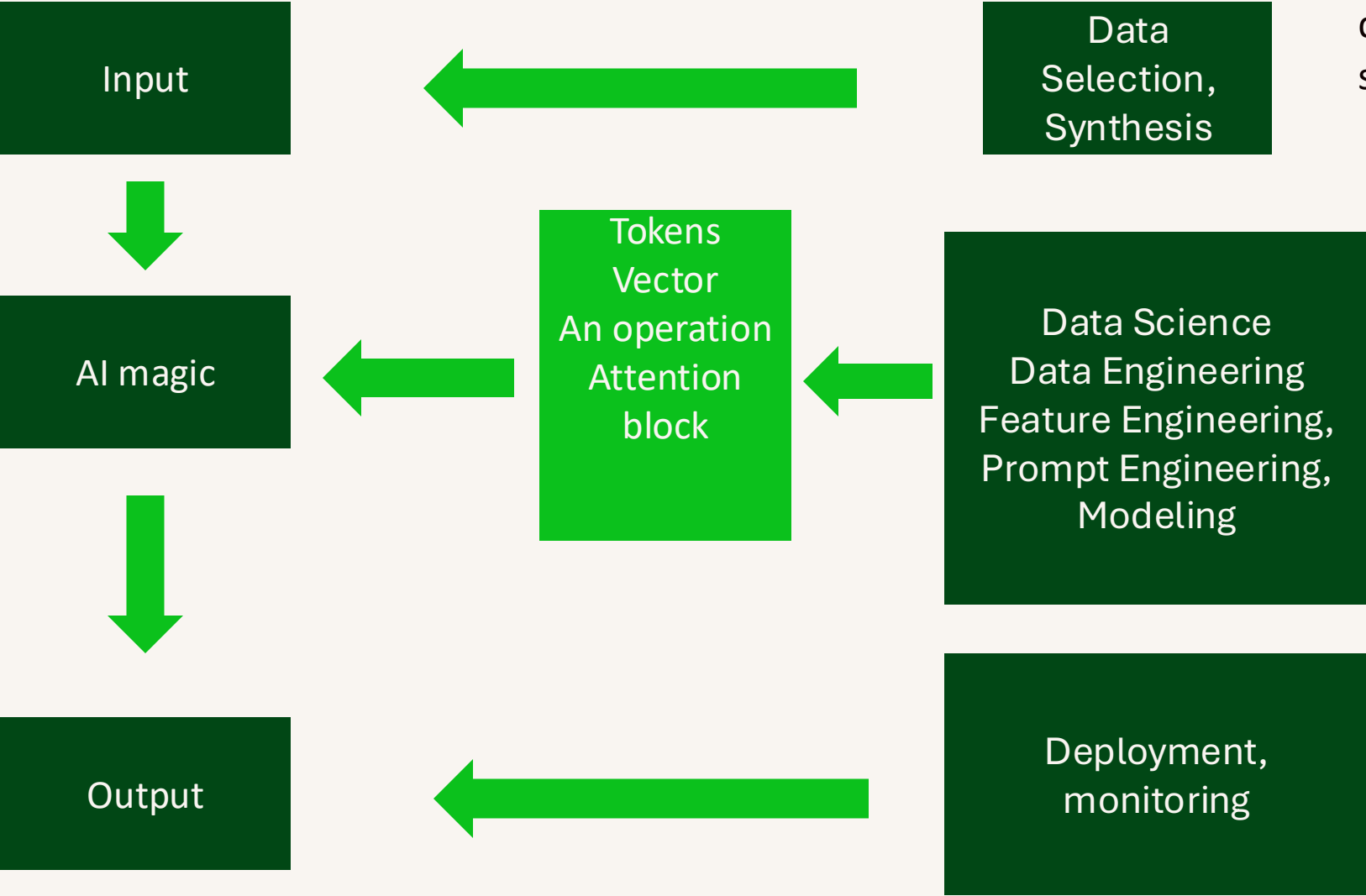
FDA's CDER has received >500 submissions with AI between 2016 to 2021

LANDSCAPE ANALYSIS OF THE APPLICATION OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING IN REGULATORY SUBMISSIONS FOR DRUG DEVELOPMENT FROM 2016 TO 2021



AI is not a Magic

Data is not true always- it will be true when you ask a question then you find something meaningful



[Foundational Understanding](#) - [AI vs GENAI with all other sources](#)

[Introductory Use Cases](#) - [Simple examples of Chatgpt, gemini, notbook LLM and some other LLMS](#)

[Hands-On Practice](#) - [Co-Pilots, Labs, and Tutors – Github, Microsoft, SAS and Google co-lab, R tutor](#)

[Experimentation](#) - [Creators your own LLM – DEFY](#)

[Regulatory Considerations](#)

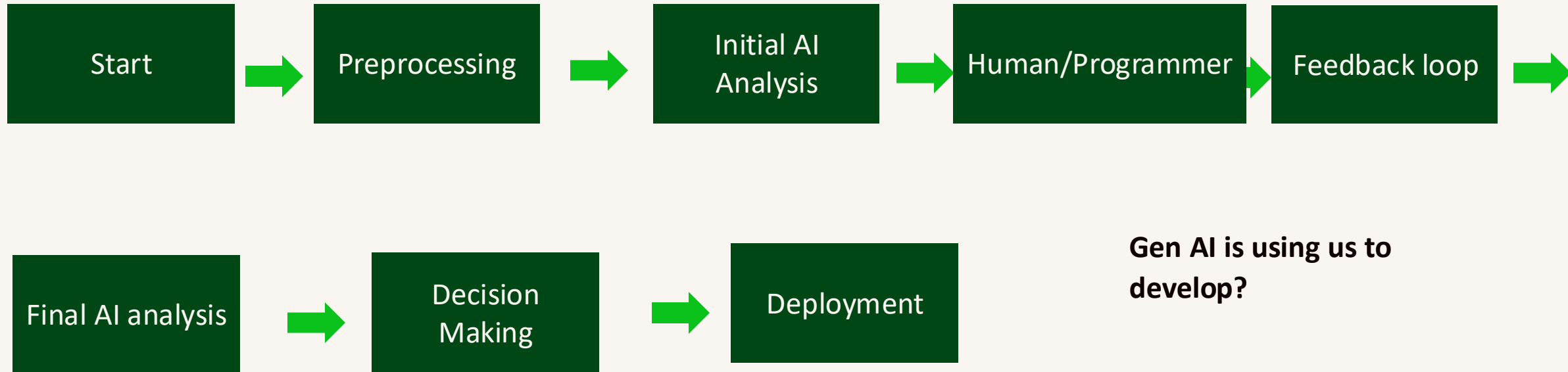
All of these applications require minimal programming effort but do rely on domain expertise, which we already possess as experienced programmers in the industry. So, why not leverage these tools?

Naturally, there are considerations when choosing between enterprise and public GPTs, such as data bias, data privacy, ethical use, and long-term ethical implications.

AI is everything ? NO – because it is Probabilistic system -> Support your day-to-day tasks

ChatGPT is great -> Intern - > better training or instructions -> it will produce a crap

What is the conclusion?



Human>Programmer-in-the-Loop Benefits:

- **Improved Accuracy:** Ensures AI outputs are accurate and contextually relevant.
- **Ethical Oversight:** Provides a human perspective to address ethical concerns.
- **Continuous Learning:** Enhances AI learning with human feedback, making models more robust.

Generative AI is not ready to operate without human oversight.

As programmers in the industry, our focus should shift from individual tools to the broader ideas and innovations they enable.

Thank you!

References

- GitHub Co-pilot: <https://docs.posit.co/ide/user/ide/guide/tools/copilot.html>
- Chatgpt in R : <https://www.r-bloggers.com/2024/05/how-to-get-chatgpt-in-r-with-chattr/>
- SAS plugin Google Co-lab: <https://communities.sas.com/t5/SAS-Programming/SAS-google-colab/td-p/759241>
- Dify : <https://cloud.dify.ai/app/4d60d622-ad72-4cb2-9d58-d1fb9f438b1a/configuration>
- ChatGPT: <https://openai.com/index/chatgpt/>
- Gemini :
https://gemini.google.com/?utm_source=google&utm_medium=cpc&utm_campaign=2024enUS_gemfeb&gad_source=1&gclid=Cj0KCQiA88a5BhDPARIsAFj595hqA82IPfYb9fkgBhcabgrXQz0WYW0wHZcNuLWR4QUSiSli04HhxaoaAjDYEALw_wcB&gclsrc=aw.ds
- Notebook LLM : <https://notebooklm.google/>
- Microsoft Co-Pilot: <https://copilot.microsoft.com/>
- Landscape Analysis of the Application of Artificial Intelligence and Machine Learning in Regulatory Submissions for Drug Development From 2016 to 2021: <https://ascpt.onlinelibrary.wiley.com/doi/full/10.1002/cpt.2668>
- Rtutor: <https://rtutor.ai/>
- Phi3.5 <https://ollama.com/library/phi3.5>