



# CM05: Unlocking the potential of Generative AI for the pharmaceutical industry – SAS AI-enabled assistant

Mark Lambrecht

Senior Director Health & Life Sciences at SAS



Jérôme Vetillard

Principal Industry Advisor  
Healthcare & Life Sciences at Microsoft





# Agenda

- What is Generative Artificial Intelligence ?
  - Different flavors of AI
  - How does it work ?
  - Gen(eral) AI *Versus* Gen(erative) AI
- Healthcare & Life Sciences possible use cases
  - Clinical Ambient intelligence with Nuance
  - Drug Discovery and Generative AI
  - Copilots
- Introducing new SAS development of a AI-enabled assistant
- Food for thoughts & Questions



# Introduction to Generative AI

# The world is changing

Industries are transforming rapidly

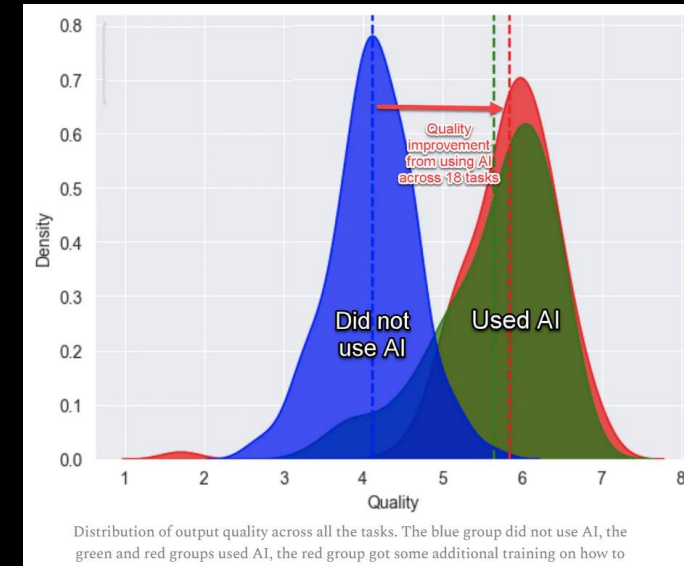
The drivers for economic growth are evolving

Technology is at an inflection point

# The AI technology is here

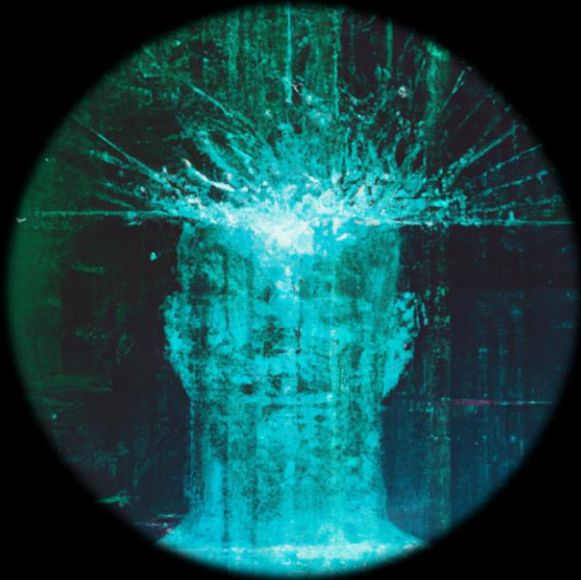
<b>Forbes</b> What ChatGPT And Generative AI Mean For Your Business?	<b>COMPUTERWORLD</b> Microsoft's new Teams Premium tier integrates with OpenAI's GPT-3.5
<b>MARKETS INSIDER</b> Nuance and Microsoft Announce the First Fully AI-Automated Clinical Documentation Application for Healthcare	<b>CNN BUSINESS</b> Real estate agents say they can't imagine working without ChatGPT now
<b>TheVerge</b> ChatGPT is now available in Microsoft's Azure OpenAI service	<b>USA TODAY</b> New Bing with ChatGPT brings the power of AI to Microsoft's signature search engine
<b>HEALTHCARE DIVE</b> Artificial intelligence could save healthcare industry \$360B a year Published Jan. 26, 2023	

# And the impact is real



Source: [Centaurus & Cyborgs](#)

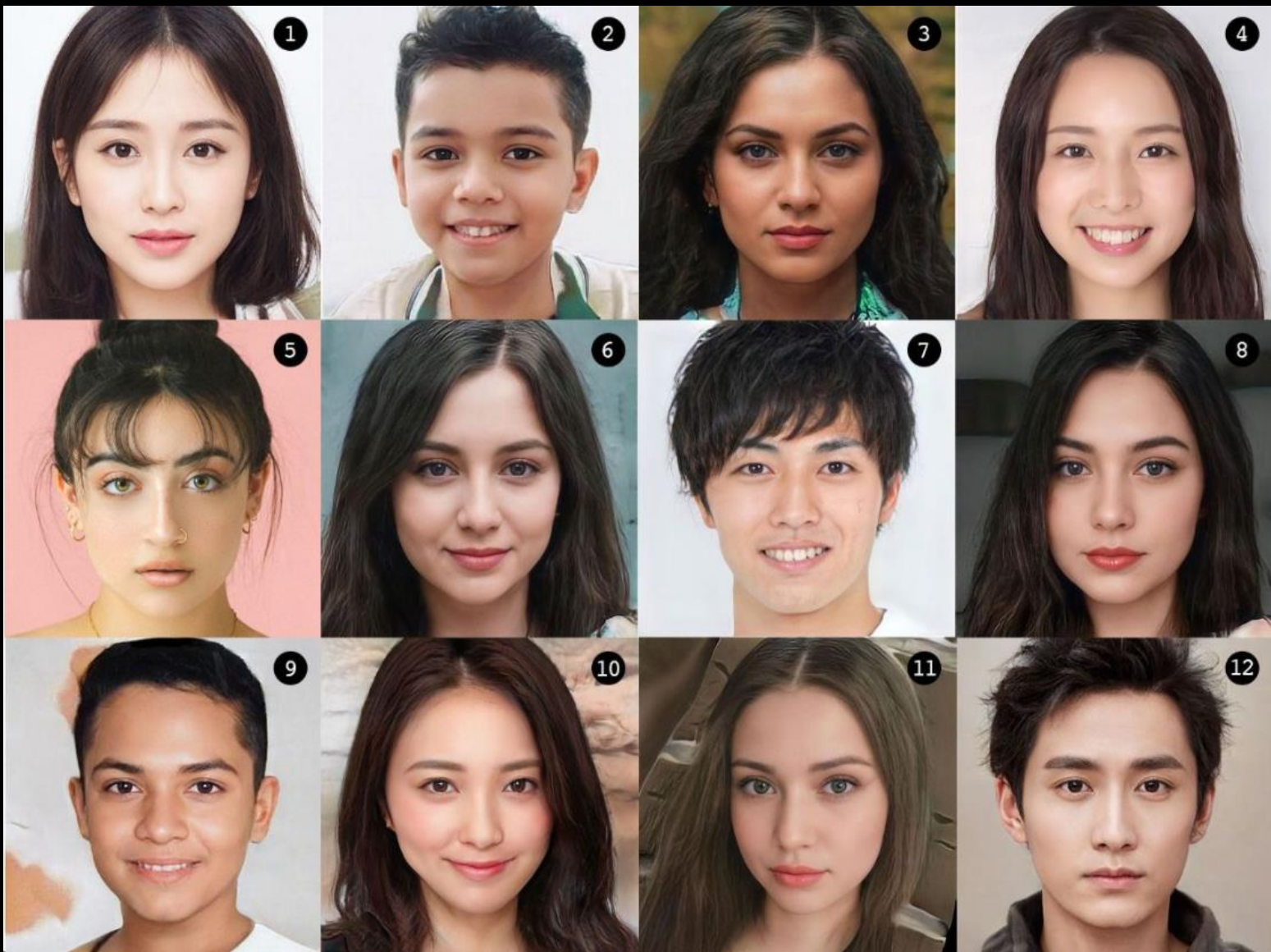
# Generative Artificial Intelligence



Generative AI algorithms learn the underlying patterns in a very large training dataset and can generate new instances that are statistically similar to the original data.

The most powerful generative AI algorithms are built on top of *foundation models* that are trained on a vast quantity of unlabeled data in a self-supervised way to identify underlying patterns for a wide range of tasks (*text, code, image, audio, video...*)

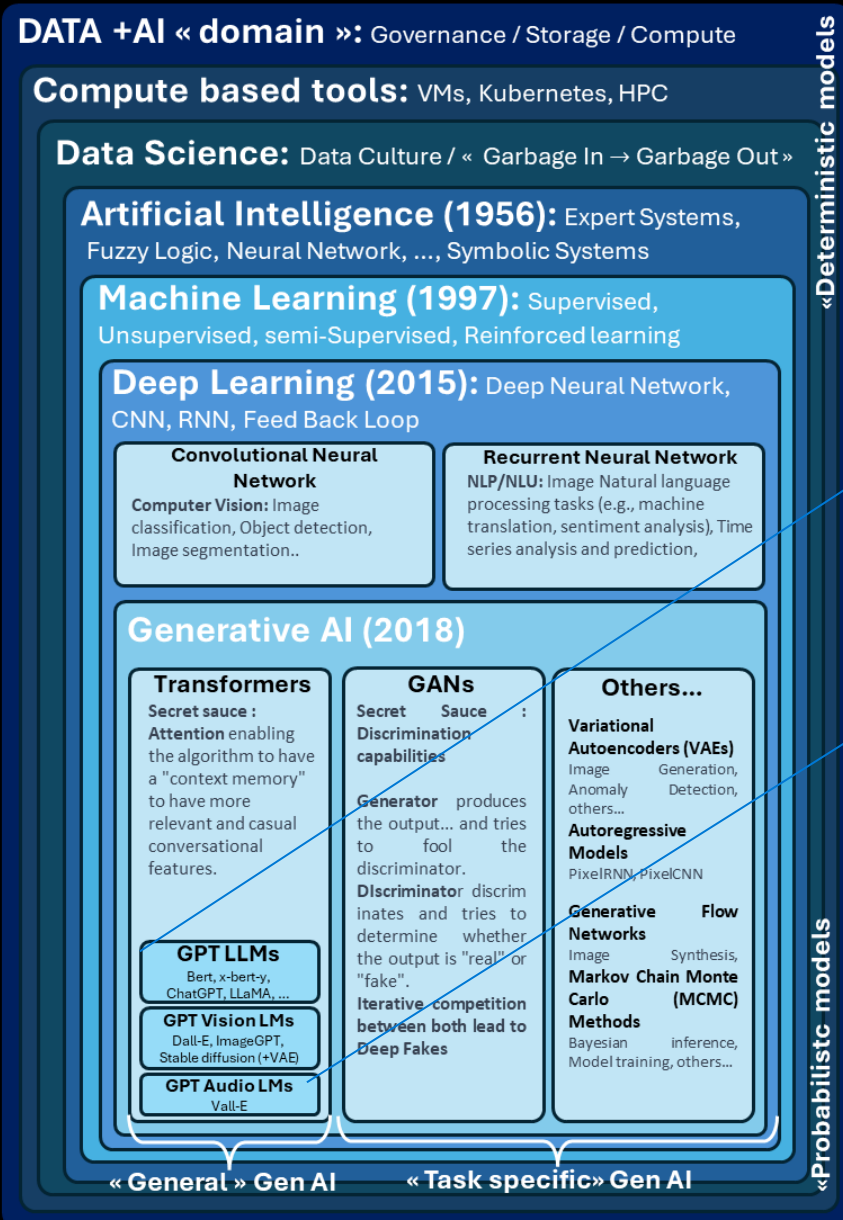
Can you beat the IA ?  
Identify the only one  
real face



<https://thispersondoesnotexist.com>



# Data + AI: Historical review of the landscape



Within Azure, OpenAI services are "just" a powerful family of AI-services offered

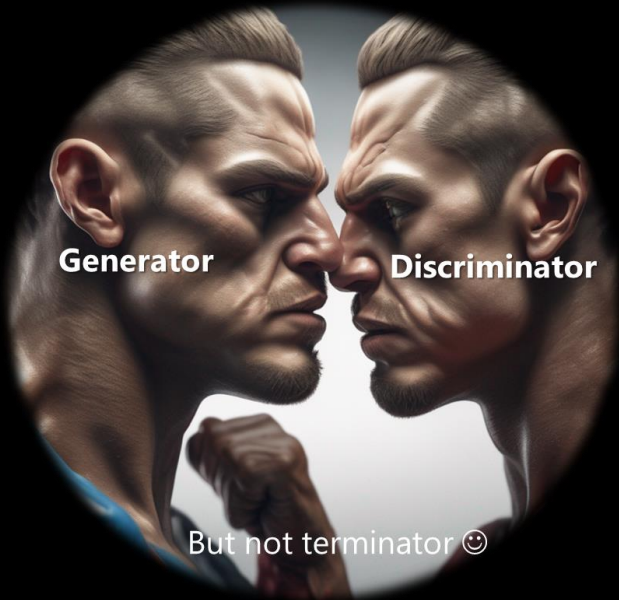


# Transformers & GANs: two pillars of Generative AI

## TRANSFORMERS



## GENERATIVE ADVERSARIAL NETWORKS (GANs)



Natural Language Understanding  
Natural Language Processing

Media Generation (Image/Video/Audio)  
Discrimination



# Generative AI: Potential Legal risks / Trustworthy AI



Generative AI (GAI) has taken the world stage, and it is here to stay. Advances in GAI are accelerating faster than either governing bodies or society have had time to reasonably address. In some categories of GAI, business value, risk, and ethics have yet to be reconciled. Resolving these three factors is critical.

SAS has long been a pioneer in, and a champion of, AI innovation. At SAS, we know that advances in technology don't exist in a vacuum and can have far-reaching effects and implications.

**SAS considers trustworthy and ethical AI to set the strategy and guardrails for all AI, and generative AI, innovation. Core to SAS DNA is to place human-centricity, customers, and “doing the right thing” first.**

SAS is committed to innovating responsibly while inspiring and empowering our customers, partners, and communities to do the same.



## Microsoft Responsible AI Principles:

### 1. Fairness

All AI systems should treat all people fairly.

### 2. Reliability & Safety

AI systems should perform reliably and safely.

### 3. Privacy & Security

AI systems should be secure and respect privacy.

### 4. Inclusiveness

AI systems should empower everyone and engage people.

### 5. Transparency

AI systems should be understandable.

### 6. Accountability

People should be accountable for AI systems.

## Microsoft Pledges Legal Protection for Customers Against AI-Generated Copyright Claims

Microsoft will provide legal protection to its commercial customers against third-party copyright infringement lawsuits, provided they adhere to Microsoft's established guardrails and content filters.

# Don't mistake Gen(eral) AI & Gen(erative) AI



Just Another Rather Very Intelligent System (JARVIS)

**JARVIS is a General AI**, multimodal, multipotent, augmented with robotic peripherals... It can handle a complex conversation and derive action plans for execution (it can calculate and build IronMan armors... and many other things).

**JARVIS is an ecosystem of specialized systems** with powerful orchestration

Speech to text + LLM+ Text to Speech would only be the NLP/NLU layer of JARVIS, the polite voice able to interact with Tony Stark (Conversational Language Understanding – CLU-)

**BUT.... IT IS POSSIBLE TO FUNCTIONALLY ENRICH LLM, STEPS TO MULTIMODAL AI**

## **EXTEND LLM CAPABILITIES VIA PLUGINS**

It is possible to extend the NLU/NLP core capabilities of LLM with « plugins ». Plugins provide new capabilities to LLM, leveraging their NLU/NLP capabilities to feed other specialized API (web search, PDF reader, computer vision...) which provide API-processed data back to the LLM.

## **PREPOPULATE THE PROMPT TO HAVE FOCUSED EMBEDDINGS**

You can « ground » the LLM with a « preprompt » telling the LLM how it should behave (role, context, expectations, tone guidance...).

Providing pairs of Question/Answer is also another way to « ground » the LLM and limit hallucinations.

## **RAG & FINE TUNING**

Retrieval Augmented Generation combines a retriever to the LLM. It will search documents, internet to enrich and generate more contextual embeddings along with the prompt itself.

**Fine-Tuning** is a re-training of the LLM with a more specialized dataset, but also modifying other parameters of the the training. Fine-Training can be used to augment the context memory (the attention) of the LLM.

# SAS and Large Language Models

Three distinct  
approaches

1

---

**SAS Viya to  
help build  
GAI apps.**

SAS Viya provides capabilities that you can use in your LLM app stack and toolchain.

2

---

**AI Assistants  
to help you  
use SAS  
tools.**

SAS will incorporate GAI to improve productivity with SAS Viya, further democratize analytics, and reduce cost of ownership.

3

---

**AI Assistants  
to help with  
industry-  
focused tasks.**

SAS will deliver industry- and domain-specific solutions that help organizations increase process efficiencies, optimize yield, or mitigate risks.



# Use Cases in Life Sciences

## Generative AI : most prominent HLS scenarios we are observing across the value chain

**Healthcare Automation:** Companies like *Anthropic*, *DataGen*, *data.world* are automating clinical data extraction, integration and analytics using generative models to standardize records and power evidence-based decisions. [Microsoft Nuance ambient clinical intelligence](#) is revolutionizing the way HCP and patient interpersonal relationship to produce gold clinical data.

**Healthcare Chatbots:** AI assistants use generative capabilities for question-answering, decision support and virtual consultation in areas like imaging, genomics, clinical trials. During the COVID crisis, health organizations have created 1,230 COVID-19 self-assessment bots based on the Microsoft Healthcare bot service, reaching 18 million individuals and serving more than 160 million messages. Now these bots are powered by Generative AI ([Azure Healthbot](#)).

**Drug Discovery:** AI startups like *Exscientia* and *DeepMind* are employing GANs and VAEs to generate novel molecular structures with desired properties, speeding up drug R&D. DeepMind's AlphaFold has also transformed protein structure prediction. [Azure Quantum Elements](#) speeds up certain chemistry simulations by 500,000 times, which is like compressing a year into one minute.

**Genomic Analysis:** Companies like *Anthropic*, *Insitro* and *TwoXAR* are leveraging generative language models to surface novel gene relationships and functional impacts from integrated omics datasets, advancing precision medicine. [Microsoft Genomics](#) can support your most demanding sequencing needs in a secured and scalable environment.

**Clinical Trial Optimization:** AI startups are developing techniques to synthesize prototype patient data (data augmentation & digital twins) and dynamically adapt trials based on outcomes, making them more efficient and personalized. [SAS Viya platform](#) concentrates all latest digital innovations needed to speed-up your clinical analytics.



# Generative AI : ambient intelligence to optimize the HCP/patient interaction (NUANCE)



- Ambient intelligence provided by NUANCE DAX capture with its NLP/NLU capabilities all the meaningful information during the anamnesis.
- It can complete the patient history in the EMR
- Empowering the HCP to be really focused on her/his interpersonal relationship with the patient.



## Captures a multi-party conversation ambiently

Clinicians engage in natural conversation with patients and other family members while the mobile app securely captures the conversation at the point of care, allowing clinicians to connect with patients rather than screens.



## Creates clinical documentation automatically

Converts encounter conversations into comprehensive specialty-specific clinical documentation summaries, and delivers the summary into the application and the Dragon Medical One desktop in seconds for clinician review and editing.



## Produces high-quality documentation

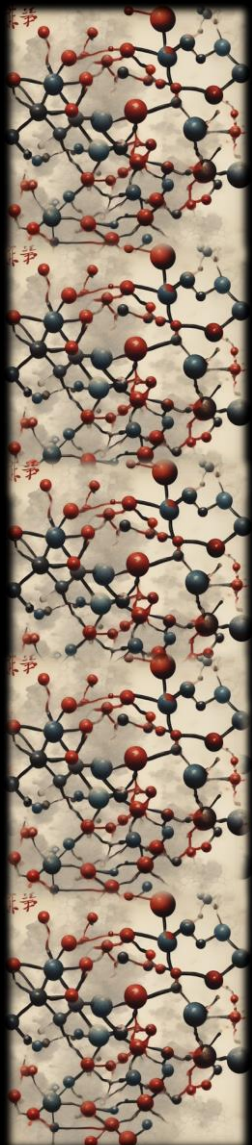
Uses the latest AI technologies and is trained on over 10M encounters to produce accurate documentation efficiently and consistently.



## Works seamlessly with electronic health records

Used in conjunction with Dragon Medical One which integrates with over 200 EHRs, streamlining implementation.

# Generative AI : Generation is not the big issue when dealing with Drug Discovery



Phase	Combinatory Chemistry	High-Throughput Screening	Hit-to-lead
<b>Process description</b>	Combinatory chemistry to generate large libraries of molecules (drug candidates)	High-Throughput Screening (HTS) to rapidly test a large number of compounds against specific biological targets	Hit-to-lead optimization involves modifying the initial hit molecule to improve its potency, selectivity, and pharmacokinetic properties.
<b>Currently...</b>	<p><b>Guided chemical synthesis patterns (Lengthy &amp; complex) either in solid or liquid phase</b></p> <p>High Throughput Parallel Synthesis (Robotic Diversity Oriented synthesis)</p>	<p><b>Automated (robotics) micro-well assays against compounds librairies (Very complex, lengthy, expensive)</b></p> <p>Assays against compound-librairies in microwells with robotics and computerized analytics.</p>	<p><b>Computerized assays, In-vivo, In-Vitro assays, PK/ADME...( Very complex, very lengthy, very expensive)</b></p> <p>Structure-Activity Relationship (SAR) Analysis, ADME (absorption, distribution, metabolism, and excretion), ...</p>
<b>With Gen AI...</b>	De Novo Molecule Design : variational autoencoders ( <b>VAEs</b> ), or generative adversarial networks ( <b>GANs</b> ) to generate structurally diverse and <b>synthetically feasible molecules</b> .	The GAN discriminator's ability to evaluate the quality of generated compounds can aid in selecting and prioritizing compounds for experimental validation. <b>Discrimination, not generation is the key point</b>	Generative AI models can propose new scaffolds or analogs based on the initial hit compound(s). <b>Generative AI models could</b> learn from large datasets of compounds with known absorption, distribution, metabolism, excretion (ADME), and toxicity properties



Introducing a  
SAS Copilot-like  
functionality

# Generative AI : SAS and Microsoft partner to add top notched Gen AI to the SAS Platform



## Could Github Copilot help us write SAS code?

Posted 07-31-2023 03:38 AM (551 views)

Has anyone done some research on github copilot? This AI plug-in (in VS Code) can support many mainstream programming languages, so can it also help us write SAS code? 😊

There is a demand for this kind of features !!!

But poor outcome « out of the box » !

FreelanceReinh JADE | LEVEL 19

Re: Extracting year and month from date9. date  
Posted 01-31-2023 09:04 AM (284 views) | In reply to PaigeMiller

@PaigeMiller wrote:  
I add that ChatGPT did not give you the best answer, it did not give you the easiest answer, and it gave you an answer that will cause difficulties if you need to perform arithmetic or logical operations on these month values (because these month values are now character, and you can't do math easily on character strings).

Not to mention the incorrect FORMAT statement it suggested, attempting to apply a numeric format to a character variable:

```

11  data example;
12      input date :date9.;
13      year_month = put(date, yymn6.);
14      format date date9.;
15      format year_month yymn6.;
16
17      -----
18              48
19
20  ERROR 48-59: The format $YYMNM was not found or could not be loaded.
21
22
23  16      cards;
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

NOTE: The SAS System stopped processing this step because of errors.  
WARNING: The data set WORK.EXAMPLE may be incomplete. When this step was stopped there were 0 observations and 2 variables.  
NOTE: DATA statement used (Total process time):  
real time 0.03 seconds  
cpu time 0.01 seconds

Hopefully, SAS & Microsoft partner together to bring SAS "Copilot" to you !!!

TECHZINE

HOME ARTICLES ▾ INSIGHTS ▾ INFORMATION

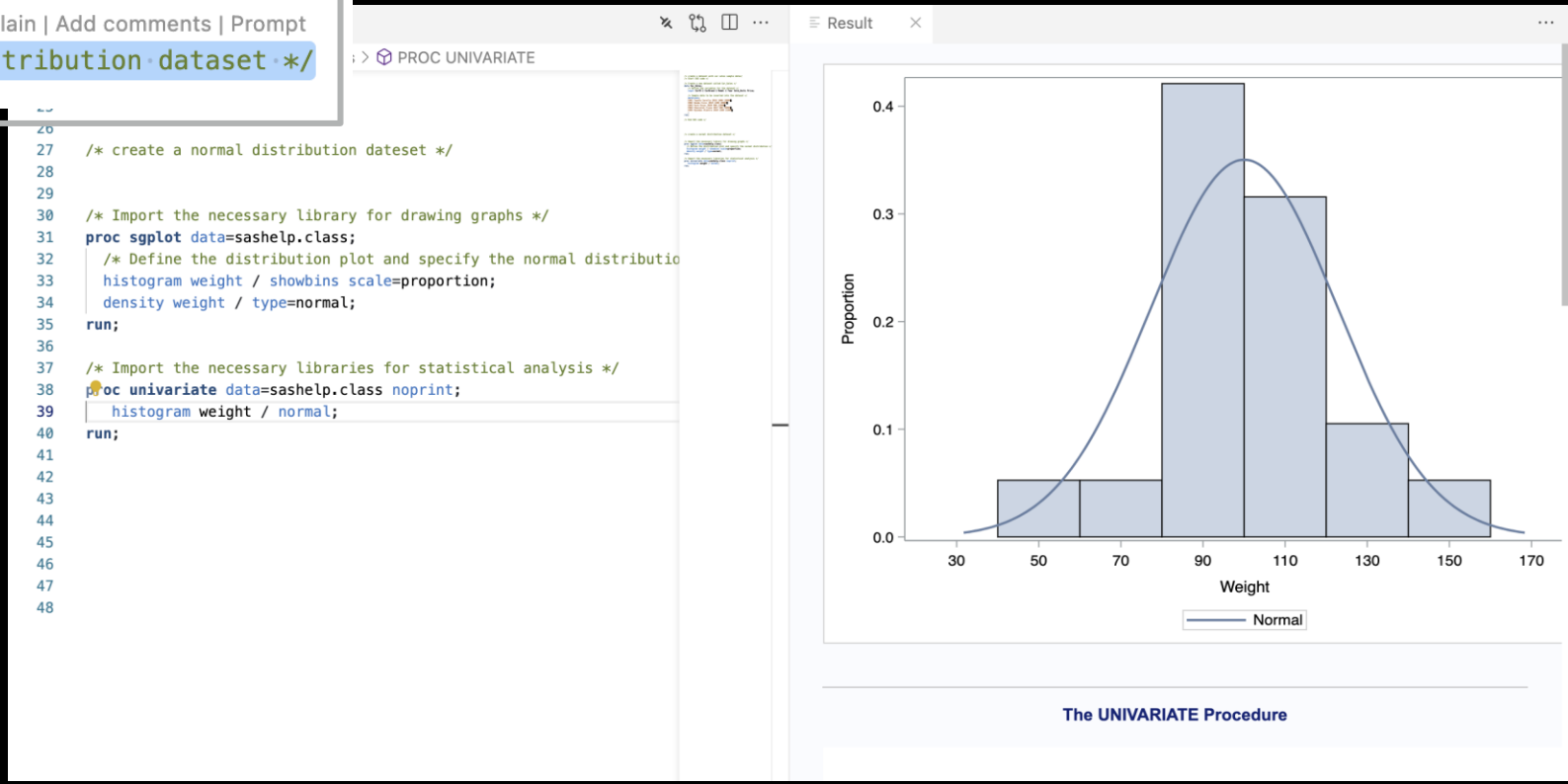
## SAS partners with Microsoft to bring generative AI to the enterprise



# SAS AI-enabled assistant for Code Gen

Type a comment to generate code

```
Copilot.sas U ●
tmp-sas-copilot > testcode > Copilot.sas > ...
1  🧠
   Brainstorm | Generate code | Explain | Add comments | Prompt
2  /* create a normal distribution dataset */
3
```





# SAS AI-enabled assistant for Code Gen

## Explain the code

**SAS VIVA COPILOT: CONVERSATION WINDOW**

You

Explain the code

**SAS Viva Copilot**

This SAS code is used to perform a statistical analysis and to create a graphical representation of the data.

In the first part of the code:

The `proc sgplot` procedure is used to create statistical graphics. The `data=sashelp.class;` statement is used to select the data set which is 'class' from the 'sashelp' library.

`histogram weight / showbins scale=proportion;` is instructing SAS to create a histogram of the 'weight' variable in the data set. The 'showbins' option allows for the visualization of the data bins in the histogram. The scale is set to 'proportion', meaning the y-axis reflects the proportion of total observations rather than the count itself.

`density weight / type=normal;` This line is instructing SAS to plot a normal density curve on the histogram for the 'weight' variable.

The `run;` statement is used to execute the procedure.

In the second part of the code:

The `proc univariate` procedure is used to provide summary statistics and to perform other univariate analysis. Again, the 'class' data set from the 'sashelp' library is specified by `data=sashelp.class;`. The 'noprint' option suppresses the usual printed output from the PROC UNIVARIATE procedure.

`histogram weight/normal;` This line is instructing SAS to produce a histogram of the 'weight' variable supplemented with a normal curve fit, which illustrates how data is distributed around the mean.

The second `run;` statement is used to execute the instructions outlined in the 'proc univariate' section of the code.

Clear conversation

Ask a question...

**Copilot.sas U**

```
tmp-sas-copilot > testcode > Copilot.sas > ...
25
26
27 /* create a normal distribution dataset and draw */
28
29
30 /* Import the necessary library for drawing graphs */
31 proc sgplot data=sashelp.class;
32 /* Define the distribution plot and specify the normal distribution */
33 histogram weight / showbins scale=proportion;
34 density weight / type=normal;
35 run;
36
37 /* Import the necessary libraries for statistical analysis */
38 proc univariate data=sashelp.class noprint;
39 histogram weight /
40 run;
41
42
43
44
45
46
47
48
```

PROBLEMS OUTPUT DEBUG

7 02:52:00 PM

88 ods html5;
NOTE: Writing HTML5 Body file...
89 /\* Set the seed for the random number generator \*/

SAS Log Friday,

Run Selected or All SAS Code F3  
Run All SAS Code F8  
Run Region  
Change All Occurrences  
Refactor... ^⇧R  
Share >  
Cut ⌘X  
Copy ⌘C  
Paste ⌘V  
SAS Copilot: Add comments  
SAS Copilot: Format  
SAS Copilot: Translate to SAS  
**SAS Copilot: Explain**  
SAS Copilot: Refine  
SAS Copilot: Find problems  
SAS Copilot: Analyze log  
SAS Copilot: See examples  
SAS Copilot: Brainstorm  
Command Palette... ⇧⌘P

# SAS AI-enabled assistant for Code Gen

See Proc examples

The image displays the SAS Viya Copilot interface, which is split into two main sections: a conversation window on the left and a code editor on the right.

**Conversation Window (Left):**

- SAS Viya Copilot** header.
- Text explaining the `sgplot` keyword: "The `sgplot` keyword in SAS development refers to the 'Statistical Graphics' procedure that produces single-cell graphs. This keyword is fundamental in creating various types of plots such as scatter plots, line plots, step plots, and histograms, etc. SG Procedures are designed to create high-quality graphics that are easy to use quickly and easily. An important feature about SG Procedures, including `sgplot`, is that they automatically handle many of the details that you had to specify in traditional SAS/GRAPH procedures. Moreover, it also accommodates group variables for enabling graph classification and makes it convenient for the user to customize graphs. Here are a few examples Basic SGPLOT
- Basic SGPLOT Example:**

```
proc sgplot data=sashelp.class;
  histogram height / binwidth=5;
run;
```
- Text: "In the above code, a histogram of the height variable from sashelp.class data is created. The `binwidth=5` option creates bins or categories with a width of 5 units. Using scatter plot:"
- Scatter Plot Example:**

```
proc sgplot data=sashelp.iris;
  scatter x=Sepallength y=PetalLength;
run;
```
- Text: "This code generates a scatter plot of SepalLength against PetalLength for the iris dataset. Using SGPLOT with a group:"
- Grouped Scatter Plot Example:**

```
proc sgplot data=sashelp.iris;
  scatter x=Sepallength y=PetalLength / group=Species;
run;
```
- A "Clear conversation" button and an "Ask a question..." input field with a send icon.

**Code Editor (Right):**

- File path: `tmp-sas-copilot > testcode > Copilot.sas > PROC SGPLOT`
- Code content:

```
25
26
27 /* create a normal distribution dataset */
28
29
30 /* Import the necessary library for drawing graphs */
31 proc sgplot data=work.sample2 noprint;
32   /* Define the normal distribution */
33   histogram height / binwidth=5;
34   density normal;
35 run;
36
37 /* Import the necessary library for drawing graphs */
38 proc univariate data=work.sample2 noprint;
39   histogram height;
40 run;
41
42
43
44
45
46
47
48
```
- A context menu is open over the `proc sgplot` line, listing actions like "Change All Occurrences", "Refactor...", "Share", "Cut", "Copy", "Paste", and several "SAS Copilot" options. The "SAS Copilot: See examples" option is highlighted.
- Execution output at the bottom:

```
real time    0.05 seconds
cpu time     0.05 seconds
```
- Log output at the bottom:

```
104 proc univariate data=work.sample2 noprint;
105 var Height;
ERROR: Variable HEIGHT not found.
106 output out=work.normal (drop=_TYPE_ _FREQ_) normal(blksize=10000 seed=52633) ranuni;
```

# SAS AI-enabled assistant for Code Gen

Refine the code

SAS VIYA COPILOT: CONVERSATION WINDOW

SAS Viya Copilot

Here is an optimized version of your SAS code:

```
/* Import the library only once */
libname mylib 'C:\sasdemo';

/* Perform statistical analysis before plotting */
proc univariate data=mylib.class noprint;
  var weight;
  histogram / normal;
run;

/* Then plot the graph with necessary labels and titles for
proc sgplot data=mylib.class;
  title "Histogram with Normal Curve";
  histogram weight / showbins scale=proportion fill;
  density weight / type=normal lineattrs=(color=red);
  xaxis label = 'Weight';
  yaxis label = 'Frequency';
run;
```

Here are the reasons why I made these changes:

- Importing Libraries: Instead of importing libraries multiple times, it's a good practice to do it once at the beginning for efficiency.
- Order of Operations: Perform the statistical analysis first before plotting for better workflow order.
- Variables: In the `proc univariate`, I specified the `var` statement to clearly indicate which variable is being analyzed.
- Labels and Titles: I added titles and labels on x-axis and y-axis to make the graph more understandable.
- Plotting: I added fill option to the histogram for better visual aspect and change the line color for the density plot to red for contrast.

Note: Please replace 'C:\sasdemo' with your library current location.

Clear conversation

Ask a question...

Copilot.sas U

```
tmp-sas-copilot > testcode > Copilot.sas > ...
25
26
27 /* create a normal distribution dataset and draw */
28
29
30
31 /* Import the necessary library for drawing graphs */
32 proc sgplot data=sasHELP.class;
33   /* Define the distribution plot and specify the normal distribution */
34   histogram weight / showbins scale=proportion;
35   density weight / type=normal;
36 run;
37 /* Import the necessary libraries for statistical analysis */
38 proc univariate data=sasHELP.class noprint;
39   var weight;
40   histogram / normal;
41 run;
42
43
44
45
46
47
48
```

Brainstorm | Generate code | Explain | Add comments | Prompt

- Run Selected or All SAS Code F3
- Run All SAS Code F8
- Run Region
- Change All Occurrences
- Refactor... ^⇧R
- Share >
- Cut ⌘X
- Copy ⌘C
- Paste ⌘V
- SAS Copilot: Add comments
- SAS Copilot: Format
- SAS Copilot: Translate to SAS
- SAS Copilot: Explain
- SAS Copilot: Refine
- SAS Copilot: Find problems
- SAS Copilot: Analyze log
- SAS Copilot: See examples
- SAS Copilot: Brainstorm
- Command Palette... ⇧⌘P

PROBLEMS OUTPUT DEBUG CONSOLE

7 02:52:00 PM

88 ods html5;

NOTE: Writing HTML5 Body file: sashtml3.htm

89 /\* Set the seed for the random number generator \*/

SAS Log Friday,

# SAS AI-enabled assistant for Code Gen

## Find problems

The screenshot displays the SAS Viya Copilot interface. On the left, a 'CONVERSATION WINDOW' shows a user asking to 'Find problems in the code'. The Copilot responds with a detailed analysis of the provided SAS code, identifying an error: 'The Proc sgplt isn't the valid procedure in SAS, the correct procedure is sgplot.' It provides the corrected code and an explanatory paragraph about the sgplot procedure. On the right, a code editor shows the original SAS code with a context menu open over a highlighted section. The menu includes standard editing actions like 'Cut', 'Copy', and 'Paste', as well as Copilot-specific actions such as 'SAS Copilot: Find problems', which is highlighted by a mouse cursor. The interface also shows a 'PROBLEMS' tab at the bottom, a 'Clear conversation' button, and a text input field for asking questions.

SAS VIYA COPILOT: CONVERSATION WINDOW

You

Find problems in the code

SAS Viya Copilot

Overall, the SAS code provided is largely correct with a minor issue. The Proc `sgplt` isn't the valid procedure in SAS, the correct procedure is `sgplot`.  
Corrected SAS code would be as follows:

```
/* Correct plot procedure */  
proc sgplot data=sashelp.class;  
/* Define the distribution plot and specify the normal dis  
  histogram weight / showbins scale=proportion;  
  density weight / type=normal;  
run;  
  
/* Import the necessary libraries for statistical analysis */  
proc univariate data=sashelp.class noprint;  
  histogram weight / normal;  
run;
```

The `sgplot` procedure is a powerful graphics procedure, capable of creating single-cell graphs of various kinds. It has numerous options for creating plots for visualizing your data. The `sgplot` procedure is easier to use and provides more options than the traditional plot procedure, making it a preferred choice for data visualization.

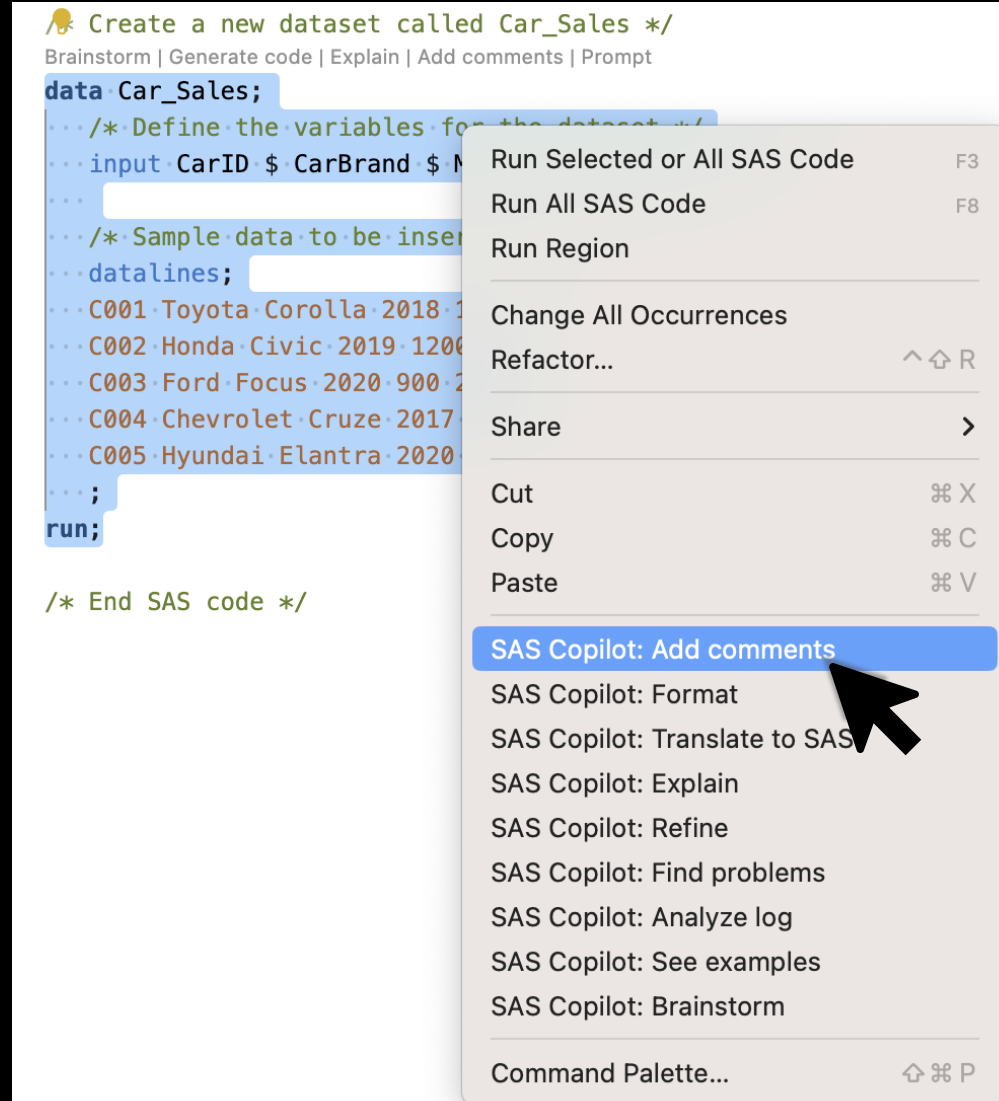
PROBLEMS OUTPUT DEBUG CONSOLE

02:52:00 PM The SAS System

88 ods html5;  
NOTE: Writing HTML5 Body file: sashtml3.htm  
89 /\* Set the seed for the random number generator \*/

# SAS Copilot-like functionality for Code Gen

Add comments to code



👤 Create a new dataset called Car\_Sales \*/  
Brainstorm | Generate code | Explain | Add comments | Prompt

```
data Car_Sales;  
  /* Define the variables for the dataset */  
  input CarID $ CarBrand $ M  
  /* Sample data to be inserted  
  datalines;  
  C001 Toyota Corolla 2018 1200  
  C002 Honda Civic 2019 1200  
  C003 Ford Focus 2020 900 2  
  C004 Chevrolet Cruze 2017  
  C005 Hyundai Elantra 2020  
  ;  
run;
```

/\* End SAS code \*/

- Run Selected or All SAS Code F3
- Run All SAS Code F8
- Run Region
- Change All Occurrences
- Refactor... ^ ⌘ R
- Share >
- Cut ⌘ X
- Copy ⌘ C
- Paste ⌘ V
- SAS Copilot: Add comments**
- SAS Copilot: Format
- SAS Copilot: Translate to SAS
- SAS Copilot: Explain
- SAS Copilot: Refine
- SAS Copilot: Find problems
- SAS Copilot: Analyze log
- SAS Copilot: See examples
- SAS Copilot: Brainstorm
- Command Palette... ⌘ ⌘ P



# SAS AI-enabled assistant for Code Gen



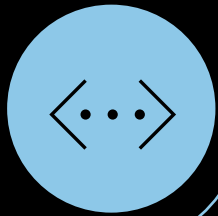
## User Interface

- **NOW** via Visual Source Code Extension
- **NEXT** Any code editor integrated throughout Viya



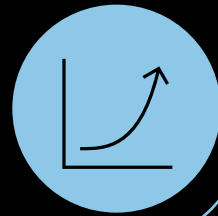
## Users

- Novice → Advanced SAS Programmers



## Use

- **NOW** Generate SAS code for a variety of data and analytical procedures
- **NEXT** Enable 9 → Viya migration
- **NEXT** Customize to industries



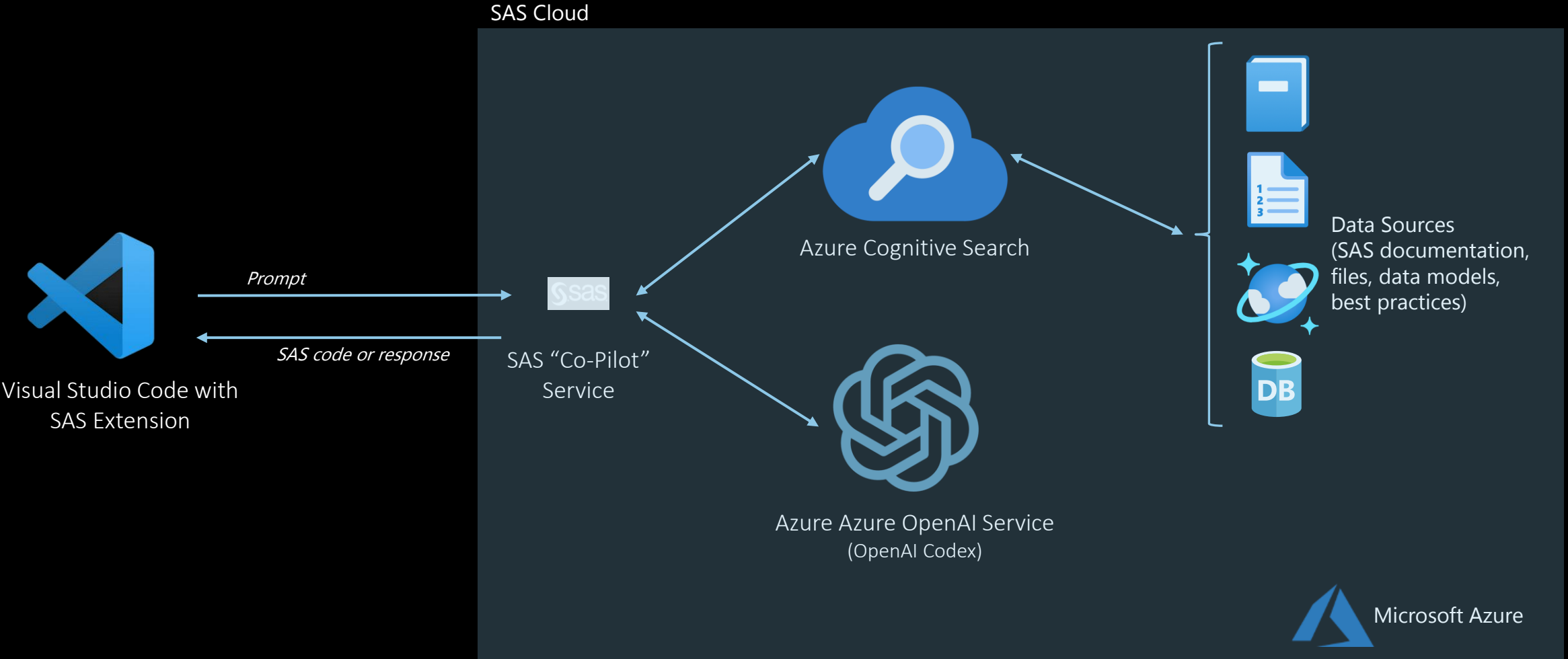
## Value

- Increased productivity and speed to market
- Decreased developer time spent on reviewing SAS documentation & syntax
- Increased discoverability of SAS capabilities
- Available via subscription service

# SAS AI-enabled assistant leveraging Azure & RAG

Available via connection to SAS Cloud on Azure.

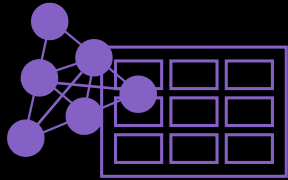
Benefits from the most secured Cloud environment and data privacy highest standards



# SAS and Generative AI

SAS includes these three AI/ML categories as part of its approach to generative AI:

## Synthetic Data Generation



## Digital Twin



## Large Language Models

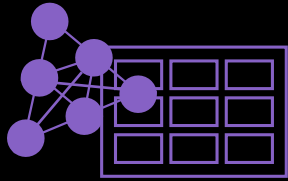


# SAS and Generative AI

SAS includes these three AI/ML categories as part of its approach to generative AI:

*In Clinical Trials...*

## Synthetic Data Generation



Useful when not enough data, such as with rare diseases

## Digital Twin



For virtual patients to simulate “what if” scenarios

## Large Language Models



Automate manual tasks, like building an FDA-regulated eCRF from a protocol

# Combining SAS and Microsoft's Azure OpenAI Service

Strategic: How do we transform clinical trial analysis and submission?

## AI-Enhanced SAS Life Sciences Analytics Framework

SAS strives to combine:

- OpenAI's GPT capabilities to process natural language queries and generate responses
- SAS's expertise in statistical computing environments for clinical research
- Dynamic embeddings built by SAS to tailor the results for clinical studies
- All on top of the enterprise-grade characteristics of Azure.



What are possible outcomes?

- Use GPT-3+ to turn a protocol into an auto-generated electronic case report form
- Analyze vast amounts of patient data to identify high-potential candidates during cohort and patient selection
- Review enrollment simulations for patient recruitment
- Use NLP to extract patient-reported outcomes.





# AI Assistants to help with industry-focused tasks

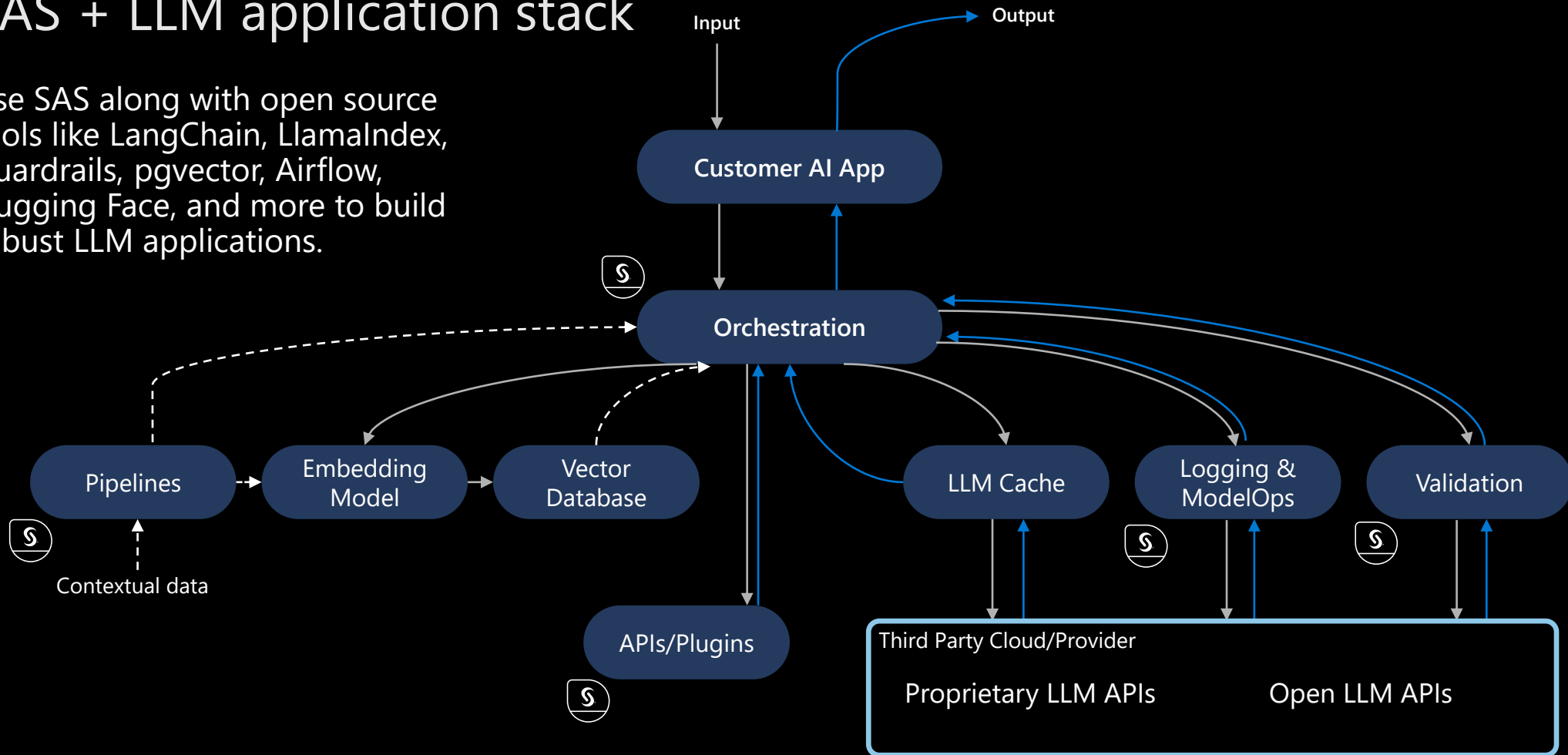
.....

SAS will deliver industry- and domain-specific solutions that help organizations increase process efficiencies, optimize yield, or mitigate risks.

# SAS Viya to help build gen AI apps

## SAS + LLM application stack

Use SAS along with open source tools like LangChain, LlamaIndex, Guardrails, pgvector, Airflow, Hugging Face, and more to build robust LLM applications.





Food for  
thought...

# Explainability of AI and LLMs : the lost fight ?

## **Principle #3: Transparency**

How AI works must be **easily explained** by those who create it and by those who use it.

**Problem:** To promote unbiased and inclusive AI, we must therefore be able to give the means to anyone (without being an expert) to be able to easily check its functioning and detect possible bias or manipulation.

**How can you trace the dynamic behavior of 175+ billions of parameters when it comes to decision making explainability ?**

*The chaos theory and emergence laws make the emergence of "eloquence" possible...*

*...but they also make explainability obsolete*

[\(41\) Artificial intelligence: Is it as "stupid" as a Galton board? | LinkedIn](#)

# Responsible AI and Sustainability ?

On the evening of Wednesday, December 2, Timnit Gebru, the co-lead of Google's ethical AI team, announced [via Twitter](#) that the company had forced her out.

“On the Dangers of **Stochastic Parrots: Can Language Models Be Too Big?**” lays out the risks of large language models—AIs trained on staggering amounts of text data. These have grown increasingly popular—and increasingly large—in the last three years. **They are now extraordinarily good, under the right conditions, at producing what looks like convincing, meaningful new text**—and sometimes at estimating meaning from language. But, says the introduction to the paper, “we ask whether enough thought has been put into the potential risks associated with developing them and strategies to mitigate these risks.”

## ARTIFICIAL INTELLIGENCE

# Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.



## Common carbon footprint benchmarks

in lbs of CO2 equivalent

Roundtrip flight b/w NY and SF (1 passenger)	1,984
Human life (avg. 1 year)	11,023
American life (avg. 1 year)	36,156
US car including fuel (avg. 1 lifetime)	126,000
Transformer (213M parameters) w/ neural architecture search	626,155

Chart: MIT Technology Review • Source: Strubell et al. • Created with Datawrapper

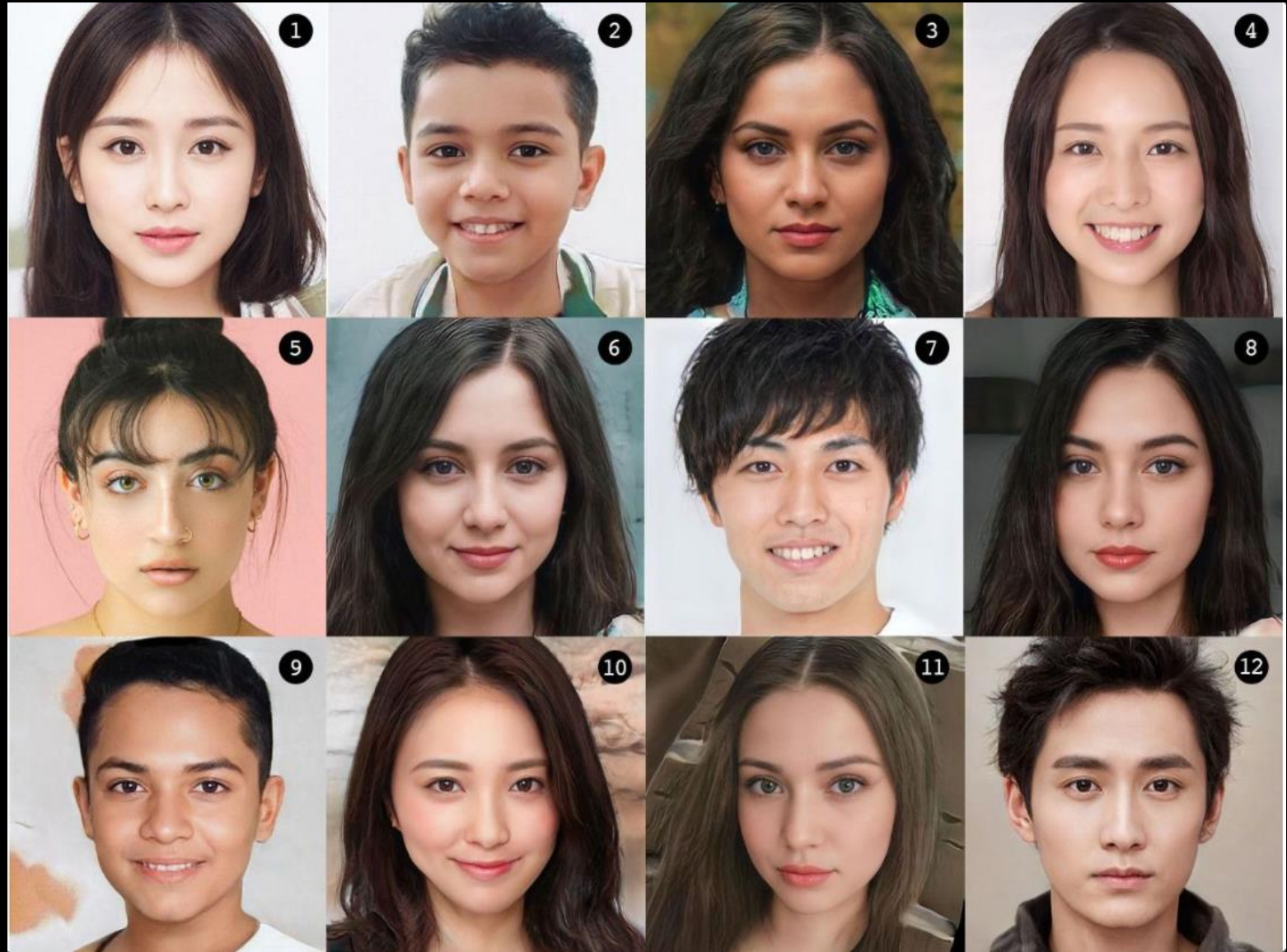
# Multi modality : The next frontier



- Generative AI has advanced in various domains, including image/video generation, voice synthesis, text synthesis, and even molecule design.
- The next frontier of generative AI is multi-modality, where models can generate and combine multiple types of data simultaneously.
- Multi-modality refers to the ability of AI models to generate outputs that incorporate diverse modalities seamlessly.
- Multi-modal generative AI combines different modalities to create cohesive and expressive outputs.
- It enables the synthesis of complex and integrated outputs, going beyond individual components.
- Applications of multi-modal generative AI include multimedia content generation, interactive storytelling, virtual environments, and personalized recommendations.
- Challenges include designing effective architectures, utilizing multi-modal datasets, and developing specialized training techniques.
- Some other philosophical, social, legal challenges will arise : intellectual property, deep fakes and mass control. Overcoming these challenges opens up exciting opportunities for AI in understanding and generating multi-modal data.



Did you make  
the right guess ?



<https://thispersondoesnotexist.com>





# Questions

