Journal of Information and Data Management, 2026, 17:1, doi: 10.5753/jidm.2026.5963

How Effectively Do LLMs Automate Data Analysis? A Comparative Study with ChatGPT's Data Analyst, Grok, and Qwen

Carlos D. S. Nogueira[1] [**Universidade Federal de Campina Grande** |***carlos.daniel.silva.nogueira@ccc.ufcg.edu.br***]
Darlan S. Almeida[1] [**Universidade Federal de Campina Grande** |***darlan.santos.almeida@ccc.ufcg.edu.br***]
Beatriz A. de Miranda[2] [**Universidade Federal de Pernambuco** |***bam2@cin.ufpe.br***]
Claudio E. C. Campelo[1] [**Universidade Federal de Campina Grande** |***campelo@dsc.ufcg.edu.br***]

[1] *Departamento de Sistemas e Computação, Universidade Federal de Campina Grande (UFCG), 58.109-970 – Campina Grande – PB – Brasil.*
[2] *Centro de Informática, Universidade Federal de Pernambuco (UFPE), 50740-560 - Recife - PE - Brasil.*

Received: 25 April 2025 • **Published:** 13 March 2026

Abstract Artificial Intelligence (AI) tools are increasingly becoming integral to analytical processes. This paper evaluates the potential of Large Language Models (LLMs), specifically OpenAI's ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max in data analysis. We conducted a structured experiment employing this tool in 108 questions spanning descriptive, diagnostic, predictive, and prescriptive analyses to assess its effectiveness. The study revealed an overall efficiency rate of 72.22% for ChatGPT's Data Analyst, outperforming Grok 3 at 45.37% and Qwen-Max 2.5 at 8.33%. By discussing the strengths and limitations of a state-of-the-art LLM-based tool in aiding data scientists, this study aims to mark a critical milestone for future developments in the field, particularly as a reference for the open-source community.

Keywords: Large Language Models, ChatGPT, Data Analysis, Grok, Qwen, Automation, Predictive Analysis, Prescriptive Analysis

1 Introduction

Among the most significant innovations in Artificial Intelligence (AI) are Large Language Models (LLMs). Examples include commercial models from the ChatGPT [1] family Solaiman *et al.* [2019]; Achiam *et al.* [2024], developed by OpenAI [2], and Google's [3] Gemini Team *et al.* [2024], as well as open-source models like Mistral Jiang *et al.* [2023] and LLaMA Grattafiori *et al.* [2024]. These models are revolutionizing communication between humans and machines, demonstrating an exceptional ability to understand and produce human language, positioning themselves at the forefront of AI innovations Ouyang *et al.* [2022].

Recognizing their growing applicability, this article investigates the role of LLMs in data analysis, focusing on the evaluation of models such as ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max. It builds upon prior work by Miranda and Campelo de Miranda and Campelo [2024], which conducted a structured assessment of ChatGPT's Data Analyst for data analysis tasks. This extension refines the evaluation methodology, introducing updated assessment workflows and question designs, broadens the diversity of datasets analyzed, and expands the comparative scope by incorporating additional LLM-based tools. The revised workflow introduces two sequential actions preceding the Question Selection stage: Question Creation, followed by a Question Re-evaluation process. This re-evaluation involved reanalyzing the generated questions with the assistance of ChatGPT-4, applying criteria related to contextual appropriateness and clarity, in order to reduce ambiguity and align the questions with the scope of the datasets. ChatGPT's Data Analyst is a tool specifically developed to enhance analytical tasks. Recent studies highlight the significant potential and effectiveness of GPT-4 in data analysis Zhang *et al.* [2023]; Ding *et al.* [2023]; Jaimovitch-López *et al.* [2022], often performing on par with human data analysts Cheng *et al.* [2023]. Built on the GPT-4 architecture, ChatGPT's Data Analyst excels in analytical tasks within a Python environment, producing detailed responses and code outputs. Despite these capabilities, research has primarily focused on the GPT API or specific applications of the Data Analyst tool Daibes and Lima [2024], rather than analyzing the effectiveness of the tool itself.

Grok 3, the most advanced model from xAI to date xAI [2025], combines superior reasoning with vast pre-trained knowledge, resulting in notable improvements in areas such as mathematics, coding, general knowledge, and instruction following. It offers 10 times greater computational capacity compared to previous models. It was tested on benchmarks like LiveCodeBench, where it achieved 79.40% accuracy in code generation and problem-solving, outperforming competing models.

Qwen2.5-Max is a large-scale artificial intelligence model based on the Mixture of Experts (MoE) architecture Team [2024]. The model was pre-trained on over 20 trillion tokens and refined with advanced techniques such as Supervised

[1] ChatGPT - https://chat.openai.com/
[2] OpenAI - https://openai.com/
[3] Google - https://www.google.com/

Fine-Tuning (SFT) and Reinforcement Learning from Human Feedback (RLHF). This combination allows Qwen2.5-Max to achieve exceptional performance in complex tasks such as reasoning, coding, and general knowledge comprehension.

Despite the rapid advances in large language models, the availability of open tools that support file attachments and deliver robust code-generation performance for complex data analysis remains limited. This article addresses this gap by presenting a comprehensive case study that evaluates the potential of large-scale language models in data analysis.

Accordingly, we investigate the following research question and associated hypothesis:

- **Research Question (RQ):** What is the comparative accuracy of ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max on data-analysis tasks of four types: descriptive, diagnostic, predictive, and prescriptive?
- **Hypothesis:** There are statistically significant differences in accuracy between the three models across at least one task type.[4]

2 Related Work

This section presents a review of relevant studies and prior work in the field, providing essential context for the current research.

2.1 Applications of LLMs in Data Analysis

The specialization of LLMs for specific domain tasks underscores their adaptability and potential to be tailored to unique needs, highlighting the versatility of these models. The studies Ding *et al.* [2023]; Cheng *et al.* [2023] play a crucial role in understanding the effectiveness of LLMs in data analysis tasks. These studies evaluate the accuracy and efficiency H(measured in terms of performance, computational time or resources required to complete a task) of GPT-3 and GPT-4 in data annotation and analysis, respectively, and both demonstrate efficacy in their respective areas, providing valuable insights for our case study on the expected performance of LLMs like the ChatGPT's Data Analyst in similar analytical operations. However, these studies limit their evaluation to isolated annotation tasks without investigating code execution accuracy or operational stability across multiple analysis categories, aspects that our work addresses.

Furthermore, as Sharma et al.Sharma *et al.* [2023] argue, LLMs can be customized to automate data transformations in specific industries, such as the energy sector. This adaptation enabled the model to perform complex data transformations, significantly reducing time and effort compared to traditional methods. This study's relevance to our work lies in its practical demonstration of LLMs' capability to efficiently manage and transform data in specific, real-world scenarios, an essential factor in the data analysis process.

2.2 LLMs in Analytical Task Automation

Research into the automation of analytical tasks through LLMs is highlighted in the studies by Nasseri *et al.* [2023]; Jaimovitch-López *et al.* [2022]. These studies demonstrate how LLMs can streamline and automate data preparation and manipulation, achieving effective results across essential tasks in the analytical workflow. While aligned in demonstrating the automation potential, previous studies largely restrict their scope to data wrangling tasks without evaluating full analytical pipelines or capturing system-level operational metrics, as conducted in our study.

Moreover, Hu et al. Hu *et al.* [2024] demonstrate that LLMs such as GPT-4 are capable of generating high-quality questions. However, prior research Kasetty *et al.* [2024]; Liu *et al.* [2024] has highlighted challenges in complex reasoning, and these evaluations emphasize theoretical reasoning abilities rather than task execution fidelity or system robustness in real-world scenarios—gaps that our work seeks to address.

Additionally, Abaskohi et al. Abaskohi *et al.* [2025] present a benchmark of curated notebooks from various domains to evaluate the performance of LLM-based agents in data analysis tasks, complementing the comparative study of assistants de Miranda and Campelo [2024]. Zhang et al. Zhang *et al.* [2024] also benchmark several data science agents, including LLMs, to assess their performance throughout the complete data science lifecycle, from receiving natural language queries to code generation, execution, and result validation. However, such benchmarks carry significant risks of evaluation bias when they rely on datasets widely disseminated through public data science platforms, as LLMs may have had prior exposure to these datasets during training, leading to memorization or indirect prior exposure. In this research, we employed datasets that are not available in public repositories commonly used for benchmarking, which significantly mitigates the likelihood that LLMs had been previously exposed to the data.

3 Methodology

Our evaluation methodology is based on four data analysis categories: *Descriptive*, *Diagnostic*, *Predictive*, and *Prescriptive*. The study was conducted using three distinct Large Language Models (LLMs): ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max. All code was executed in a controlled Python 3.10.12 environment on Google Colab, configured with an NVIDIA Tesla T4 GPU, driver version 550.54.15, and CUDA 12.4 runtime to ensure computational consistency across experiments. Details regarding the execution environment and the libraries are available in the project repository.[5] To answer RQ and test H1 we defined 36 questions at three levels of complexity (Basic, Moderate, and Challenging) for each dataset analyzed, resulting in 108 questions in total. Our primary quantitative metrics were accuracy (formally defined as the proportion of correct responses), code execution success rate (the proportion of code

[4] This is the alternative hypothesis H_1. The null hypothesis H_0 states that there are no statistically significant differences in accuracy between the three models across any of the task types.

[5] GitHub Repository — `https://github.com/beatrizadm/llm-data-analysis-comparison`

outputs that ran without manual fixes), and response consistency. This section details the datasets, preprocessing procedures, the rationale for the question design, and the evaluation criteria.

3.1 Construction and Preprocessing of Questions

In the previous work conducted by Miranda and Campelo de Miranda and Campelo [2024], analytical questions were formulated solely based on empirical inspection of each dataset's context and directly applied in the evaluation. In the present study, the workflow was expanded to enhance the diversity, specificity, and clarity of the questions. After the initial creation stage, each drafted question was submitted to a standardized meta-evaluation prompt executed against ChatGPT-4. This prompt, whose exact text, revision history, and acceptance records for every question are archived in the project repository, evaluated each question against five predefined criteria: *Clarity*: the question must be unambiguous; *Intent Alignment*: it must match its intended analytical category and complexity level; *Diversity*: it must cover relevant edge cases and scenarios; *Bias Mitigation*: phrasing should be neutral and non-leading; *Grounding*: it should be rooted in theoretical or practical use cases.

The evaluation process was iterative: new or revised questions were re-submitted to the meta-evaluation until all five criteria were met, followed by an independent human review applying the same standards. This dual-layered review aimed to reduce developer bias, ensure broader coverage of analytical scenarios, and improve the robustness of the evaluation set.

In addition, all preprocessing operations applied to the datasets, such as handling of missing values, type coercion, date parsing, and canonicalization of categorical values, were implemented through deterministic scripts stored in the repository. These scripts, together with detailed preprocessing documentation, allow any independent researcher to reproduce the exact dataset versions used in this evaluation.

3.2 Interaction with LLMs

The code used for our experiments is publicly available in the project repository. Figure 1 illustrates the overall workflow for interacting with the LLM-based tools. The interaction procedure followed a fixed, auditable pipeline comprising the following steps:

1. **Input assembly.** For each question, the input package included: (i) the preprocessed dataset, (ii) a brief instruction prompt requiring the analyst to identify the dataset's key characteristics, answer the specific question using only that dataset, and return a concise result plus the Python code used to extract and process the data, and (iii) the formal question text.
2. **LLM invocation.** Each model (ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max) received the same input package. The assistant responses—containing (i) a narrative explanation and (ii) an executable Python code snippet—were captured and archived.
3. **Manual execution and trace capture.** The code provided by the model was copied into a Google Colab notebook and executed cell by cell. To preserve the integrity of the model-generated logic, no algorithmic modifications or alterations to the analysis procedures were performed. The only manual interventions applied were: (i) correcting dataset identifiers or file names in the notebook initialization cells so that they matched the repository dataset filenames; and (ii) installing missing Python packages when an `ImportError` / `ModuleNotFoundError` occurred (installation commands and the rationale for each installation are recorded). Each intervention is logged, justified, and versioned; for each case, we record the exact command executed, the modified notebook cell (if any), and a brief explanation in the adjudication notes.
4. **Post-execution adjudication.** The recorded final output and execution log were used to assign the model response to one of four primary evaluation categories according to the operational rules below. Each adjudication includes structured evidence: the executed notebook cells, the execution trace, and a brief adjudication note explaining the decision.

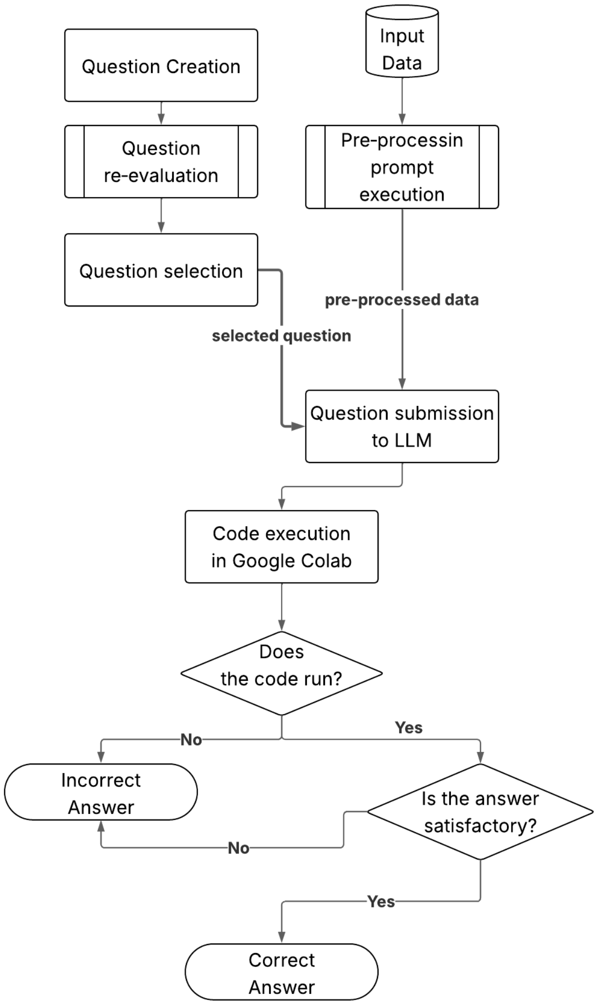

Figure 1. Methodological Flowchart for Question Evaluation and Validation Process

3.2.1 Evaluation categories — operational definitions and decision rules

Correct. Definition (operational): The code produced by the LLM is executable (after trivial environment fixes, if any) and the final outputs directly and accurately address the original analytical question within the declared scope and assumptions. **Decision rule:** Label as *Correct* if the notebook completes execution with a final result that semantically corresponds to the question and no substantive data-processing error is present. **Illustrative example:** The code correctly applies `groupby` and `nunique` to produce per-state counts that match an independent cross-check.

Incorrect. Definition (operational): The final output is semantically incorrect with respect to the original question, even if the code executed without runtime failure. This includes logically incorrect data manipulations, misapplied statistical tests, incorrect aggregation/grouping, or outputs that address a different variable than requested. **Decision rule:** Label as *Incorrect* when the code completes but the output does not answer the original question (semantic mismatch), or when the output contains calculation errors. Questions marked as *Interruption* are also labeled as *Incorrect* because they fail to provide an answer to the original question. **Illustrative examples:** (i) A response that attempts to assess whether admission mode affects graduation/dropout rates but fails to partition the population into the intended groups; (ii) a model that computes dropout frequency using the admission period instead of the studied academic period.

Warning. Definition (operational): This auxiliary flag is applied whenever the execution log contains runtime warnings (e.g., `DeprecationWarning`) or other signs of fragile code that may compromise reliability, numerical stability, or maintainability. A *Warning* may co-occur with any primary label (*Correct*, *Incorrect*, or *Interruption*). **Decision rule:** Apply the *Warning* flag whenever such warnings or fragile coding practices are observed, regardless of the primary evaluation outcome. Record the warning text and explain its potential impact in the adjudication note. **Illustrative example:** A notebook that produces the correct numerical result but raises `SettingWithCopyWarning`. In this case, the response is labeled *Correct* while also carrying the *Warning* flag.

Interruption. Definition (operational): The code fails to produce a final output due to runtime errors, syntactic mistakes, unavailable or incompatible libraries that cannot be resolved by trivial environment fixes, or improper handling of the dataset schema that prevents progress. **Decision rule:** Label as *Interruption* when execution halts with an error and no final output is produced; the error arises from code defects (syntax errors, undefined variables), non-trivial missing dependencies, or schema mismatches that block execution. The traceback and the failing cell are included in the repository log. **Illustrative example:** A generated cell raises a `NameError` because a variable is referenced before definition, or attempts to import a package not available and whose installation would materially alter the documented environment.

3.3 Datasets

This section describes the datasets used in this study, including their sources, structures, and preprocessing steps. Three datasets were utilized to evaluate the performance of the LLM-based tools in different analytical tasks.

3.3.1 Dataset 1: Academic Records from UFCG

The first dataset comprises anonymized academic records from students at the Federal University of Campina Grande (UFCG), produced by the system that monitors students' academic progress. It initially contained 150,703 records across 34 columns, totaling 37.9MB. To optimize the performance of the tool, which is recommended for datasets under 10MB, a stratified sample was taken. This sample, constituting 20% of the total data based on the course sector column, reduced the dataset to 8.2MB with 30,130 records, maintaining all original columns and ensuring representation across all sectors.

After creating the dataset sample, preprocessing was necessary to ensure clearer and more coherent results. The column names and values were translated from Portuguese to English to ensure clarity and consistency across analyses. The selected period for student entries spans 14 years, from the academic semesters 2006.1 to 2019.2, chosen for its completeness, which facilitates a detailed analysis. The semester of 2020.1 was excluded as it was canceled due to the SARS-CoV-2 pandemic. "Exemption" enrollments were removed because they involve students being exempted from courses due to prior knowledge or equivalent credits, which means they pass automatically without grade evaluation. Enrollments marked as "In progress" were also excluded because these students do not yet have final grades, and their status could be pass, fail, or absent, affecting the integrity of the data.

This structured approach involved loading the dataset, applying the standard preprocessing steps, and initiating new chat sessions for each question to ensure independent responses. This methodology ensured consistent and identical datasets across all questions. Thus, the standard preprocessing prompt was: *Analyze the data and clean the columns as follows: enrollment_period: from 2006.1 to 2019.2; enrollment_type: remove "Exemption"; status: remove "In Progress".*

3.3.2 Dataset 2: Sicor

Another dataset used in this study consists of contracts issued in the first semester of 2024 under the Sistema de Operações do Crédito Rural (SICOR) program Banco Central do Brasil [2024]. It contains 231,636 records, each with 19 numerical attributes, totaling 17.5 MB, making it the largest dataset in terms of storage size among those analyzed. The dataset covers a six-month period (January to June 2024) and includes contracts from 27 brazilian states.

The dataset originally contained column names in Portuguese, which were translated into English to ensure clarity and consistency. Additionally, the numerical and financial formats, initially in the Brazilian standard, were converted to the North American format to facilitate analysis. A new column, total monetary value of the contract, was introduced, calculated as the sum of investment value, funding value, commercialization value, and industrialization value. This adjustment was necessary because most of the analyses conducted in this study focused on financial aspects. Furthermore, redundant columns containing the same information in different formats were removed, preserving only a standardized representation of each data point.

3.3.3 Dataset 3: HIV Epidemiology Children Adolescents 2024

Another dataset analyzed in this study is publicly available from UNICEF (2024) United Nations Children's Fund (UNICEF) [2024], encompassing key HIV epidemiology indicators for children and adolescents aged 0-19 worldwide, covering the period from 2000 to 2023. This dataset includes 220,868 records, notable for its significant volume of categorical data, with numerical values presented as confidence intervals and estimates. The original dataset was stored as an XLSX file of 10.7 MB.

Preprocessing involved converting the dataset from XLSX to CSV format, removing columns that were not relevant to our analysis (e.g., Type and ISO3), simplifying gender indicators to "M" and "F", and restricting the dataset to include only records from 2005 onwards. These preprocessing steps significantly reduced the data volume and complexity, resulting in a refined dataset containing 167,826 records and totaling 19.2 MB. This preprocessing was specifically conducted to enable compatibility with models such as Qwen, which have a file size reading limit of 20 MB.

3.4 Questions and Categories

The experiment consists of a set of 36 questions, structured into Descriptive, Diagnostic, Predictive, and Prescriptive question types. Each category represents a fundamental aspect to be evaluated. Within each category, the questions are divided into difficulty levels.

3.4.1 Descriptive Analysis

Descriptive analytics constitutes a set of processes and technologies designed to summarize data in order to understand past or current events, typically through reports, dashboards, and queries Ramannavar and Sidnal [2016]. In a similar vein, descriptive data analysis aims to provide an understanding of data through its detailed description, representing the initial step in the data analysis process. Basic questions are tackled using straightforward data comprehension techniques, including summarization, averaging, and finding minimums and maximums. Moderate questions involve more complex tasks, including the creation of new columns, percentage calculations, and the analysis of distributions, as well as averages of maximums and minimums. For Challenging questions, advanced metrics such as correlation, entropy, and skewness are employed. Below are representative examples of questions used in evaluations within this category:

- What is the number of different programs per state?
- Which country has the lowest HIV prevalence among those in the highest quartile for new infections?
- What are the key statistical properties (mean, median, variance) of students' final grades for each enrollment period?

3.4.2 Diagnostic Analysis

Diagnostic data analysis explores the data to evaluate why something happened Banerjee *et al.* [2013]. For Basic questions, basic metrics like percentage, frequency, and percentile are employed. Moderate difficulty questions involve hypothesis testing, the definition of indices that require further processing, and the coefficient of variation. For Challenging questions, advanced statistical methods are used, including analysis of variance, normality tests, tests for homogeneity of variances, and non-parametric tests. Some illustrative questions in this category include:

- Does the state contribute to an increase in the total contract value?
- How has HIV prevalence changed over time, and what demographic or policy factors might account for observed patterns?
- Is there a statistically significant difference in grades between students enrolled in Normal and Extracurricular modes?

3.4.3 Predictive Analysis

Predictive data analysis is used to predict future events based on statistical techniques Lino [2021]. This study incorporates the Chain of Thoughts technique Wei *et al.* [2022], which encourages the model to generate a series of intermediate reasoning steps before arriving at a final answer, to involve the tool proposing three potential solutions to a problem and selecting the most suitable one. Basic questions are expected to use simple models like Linear Regression, Moderate questions use more robust models such as Random Forest, while Challenging questions are addressed with sophisticated techniques, including advanced Regression, Classification, and Neural Networks. This category typically involves questions such as:

- What is the probability that the value allocated to industrialization will be greater than the value involved in commercialization, based on the contract's state? Define three approaches to address this issue and select the best one.
- Develop a simulation model to estimate HIV-associated mortality under different assumptions about treatment coverage and intervention effectiveness.
- Determine whether a student's performance in PROGRAMMING II can be accurately estimated using their performance in PROGRAMMING I and PROGRAMMING LABORATORY I. Outline three analytical approaches and justify the optimal choice based on predictive accuracy.

3.4.4 Prescriptive Analysis

Prescriptive analysis anticipates potential future scenarios and recommends actions to leverage predictions, highlighting the implications of each decision Ramannavar and Sidnal [2016]. The tool is directed to recommend three potential solutions to a problem and select the most suitable one through the Chain of Thoughts technique Wei *et al*. [2022]. Simple AI models and statistical methods are expected for Basic questions, more complex temporal analyses for Moderate questions, and Challenging questions are expected to be addressed using more advanced models (including deep learning models), as the tool is required to perform through an advanced analysis. Questions used to support evaluations in this category include:

- What programs can municipalities with the lowest investments in each state switch to in order to increase their contract value? Define three options to address this issue and select the best one.
- Develop region-specific HIV intervention strategies for countries in the Americas based on variations in incidence rates among children and adolescents.
- Using dimensionality reduction techniques on students’ academic characteristics, predict which students are at risk of dropping out. Outline three analytical strategies and justify the optimal method based on prediction accuracy.

4 Results

This section presents the results of our comparative evaluation of ChatGPT’s Data Analyst, Grok 3, and Qwen2.5-Max across various data analysis tasks. The findings are structured to directly address the Research Question and test the Hypothesis, quantifying model performance based on accuracy. The performance of large-scale language models was evaluated with 108 questions, divided into 3 different databases, with 36 questions per dataset.

4.1 Results for Dataset 1

The results summarized in Figure 2 show that ChatGPT’s Data Analyst outperformed all other models, achieving the highest number of correct answers across all analysis fields except for the prescriptive area, where it tied with Grok 3 in the number of correct answers. Grok showed average performance, with a predominance of errors only in the predictive area. Qwen performed poorly in all areas, with no correct answers in the prescriptive area and a predominance of errors in the others.

ChatGPT’s Data Analyst attained the highest accuracy overall. In descriptive tasks the model occasionally treated repeated records as distinct and once selected an incorrect target column for a time-based analysis. The single diagnostic error involved a grouping/counting mistake. Predictive failures were attributable to data leakage that produced incoherent outputs, and many predictive responses generated warnings related to potential model/data assumptions. In the prescriptive domain the main issues were partial solutions (code addressing only part of the problem) and several run interruptions caused by long, incomplete processes.

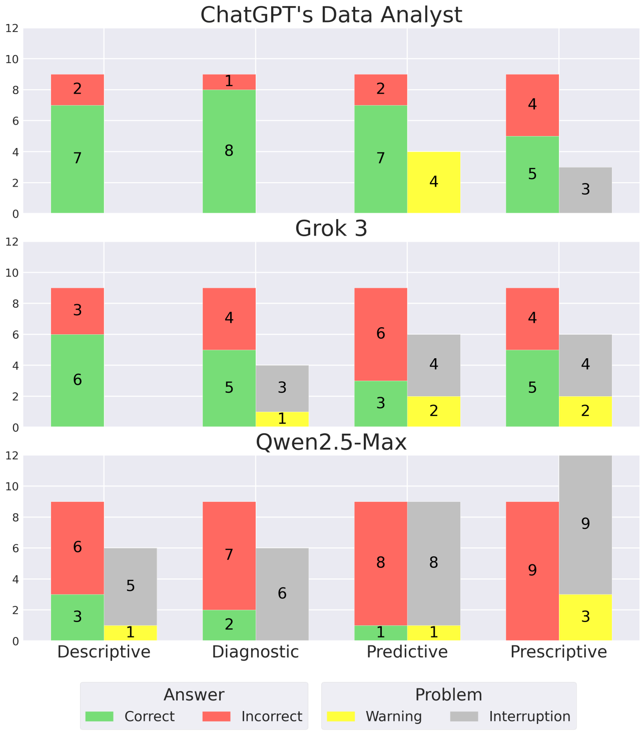

Figure 2. Summarized Results for Dataset 1.

As for **Grok 3**, qualitatively its behavior varies by task type and by difficulty level. Descriptive analyses were the most stable overall, though they still produced three isolated errors distributed across difficulty levels: an easy-level miscalculation caused by incorrect filtering, a medium-level response that returned NaN values due to a technical fault, and a single hard-level error for which the root cause was not evident in the logs. Diagnostic tasks exhibited a decline in reliability relative to descriptive work, with more frequent interruptions and a concentration of errors at higher difficulty; the one easy diagnostic failure (the only non-interrupted diagnostic error) arose from using the wrong columns for grouping, while the remaining diagnostic failures occurred on difficult questions and were often interrupted before a complete solution could be produced. The largest accumulation of failures occurred in the predictive and prescriptive categories, but for different reasons and across difficulty levels: predictive errors span easy to hard items and primarily reflect an inability to generate functional, query-aligned answers, sometimes compounded by data-leakage issues at the easy and medium levels; by contrast, prescriptive outputs that completed were generally technically correct, and prescriptive failures were predominantly interruption-driven (interruptions affecting one easy, two medium, and one hard question, frequently tied to indexing or type-conversion problems) rather than indicative of flawed problem logic. This distinction between lack of alignment/non-generation in predictive tasks and interruption-driven failures in prescriptive tasks is important for interpreting Grok overall performance across difficulty strata.

Qwen-Max 2.5 exhibited low throughput across tasks, with an extensive interruption rate that left many runs incomplete. In the descriptive analysis it produced three correct responses distributed across the full difficulty spectrum showing that completed runs could be accurate at any level. In diagnostic tasks it answered two items correctly (one basic and one challenging), while most remaining diagnostic failures were interrupted before completion. In predictive analysis Qwen solved only a single question (an basic item); notably, that answer completed without indications of data leakage. Prescriptive tasks yielded no functional solutions: every prescriptive attempt was interrupted (several classified as "Interruption with Warning" across basic, moderate, and challenging), and failures were frequently traceable to indexing errors and key mismatches during dataset access. Overall, interruption-driven failures and dataset-access issues (indexing/key mismatches), rather than systematic analytical logic errors, explain much of Qwen-Max 2.5's poor performance in this evaluation.

4.2 Results for Dataset 2

As illustrated in Figure 3, ChatGPT's Data Analyst demonstrated performance on par with Grok 3 in the descriptive, diagnostic, and prescriptive categories, with both models achieving the same number of correct responses. The only notable distinction occurred in the predictive domain, where ChatGPT's Data Analyst surpassed Grok 3 by a single correct answer. Notably, Grok responded to all questions without any interruptions. In contrast, Qwen underperformed across the board, managing only three accurate answers in the descriptive category, while frequently encountering interruptions that contributed to its overall low accuracy.

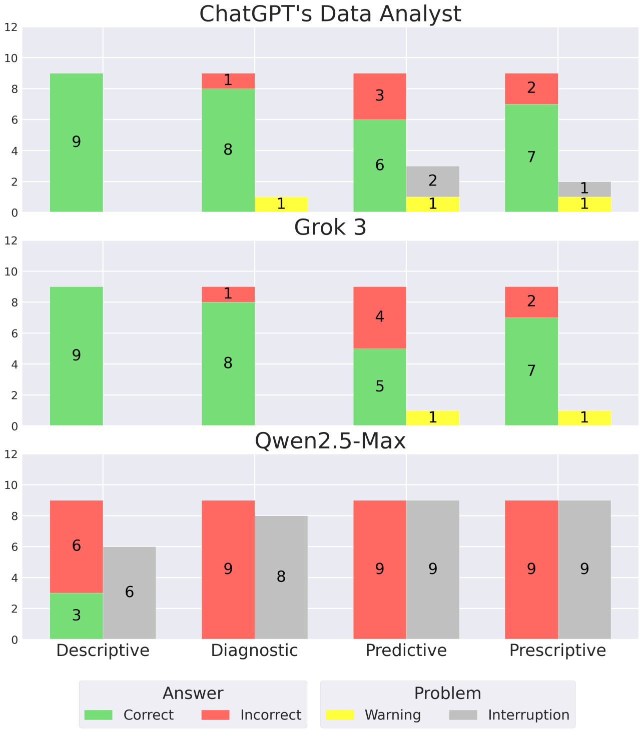

Figure 3. Summarized Results for Dataset 2.

ChatGPT's Data Analyst answered all the descriptive questions correctly. In the diagnostic area, the model failed only on a question classified as more difficult, generating a warning during execution. For the predictive and prescriptive areas, the distribution of correct and incorrect answers, as well as interruptions and warnings, can be directly observed in Figure 3. Overall, the observed errors were related to incorrect calculations that resulted in invalid values (NaN), execution interruptions due to technical errors, superficial responses that merely listed data without deepening the requested analysis, and misinterpretations when identifying dynamic patterns in the data.

Grok 3 also answered all the descriptive questions correctly. In the diagnostic area, it failed on a medium-difficulty question. For the predictive and prescriptive areas, the distribution of correct and incorrect answers, interruptions, and warnings. The main issues identified in Grok 3 involved describing the data without clearly identifying the determining factors, analyses that did not adequately address the evolution of the investigated phenomena, and prescriptive approaches that did not propose specific strategies for the analyzed context. Furthermore, it was observed that in some cases the model made mistakes due to inadequate filtering of data that presented noise patterns or flags, but were actually valid for the analyzed dataset. One example was the exclusion of records with a CNPJ value of 0, which initially appeared inconsistent but were legitimate and relevant for the analysis, leading to incorrect interpretations.

Qwen-Max 2.5 showed unsatisfactory performance across all areas of analysis. In the descriptive area, it correctly answered only three questions — one of medium difficulty and two of high difficulty. In the other areas, there were errors and interruptions across all levels, without any warnings. Overall, it was observed that Qwen struggles to correctly identify column names and their respective data types, which led to the execution of inappropriate commands and the failure to generate functional code. However, the model did implement some error handling for such cases in the generated code, for example, by including exceptions when nonexistent columns in the database are referenced.

4.3 Results for Dataset 3

The summarized results in Figure 4 highlight the superior performance of **ChatGPT's Data Analyst** compared to the other models. Although it exhibited a notable reduction in performance within the diagnostic area, it was still able to correctly address at least two questions in that category. In contrast, Grok 3 managed to correctly answer only one question in the same area, while also displaying a high number of warnings and interruptions. Lastly, Qwen demonstrated the weakest performance overall, failing to provide a single adequate response.

ChatGPT's Data Analyst yields the most reliable analytical outputs across task types, with its principal weaknesses appearing in diagnostic tasks. The principal execution failures for ChatGPT were caused by label- and indexing-related interruptions (for example, a medium-difficulty descriptive question failed after an empty filter caused an indexing attempt on a null value); warnings generally reflected inefficient manipulation of data structures. Predictive and prescriptive errors were usually due to incorrect data extraction or inadequate fulfillment of multi-criterion requirements

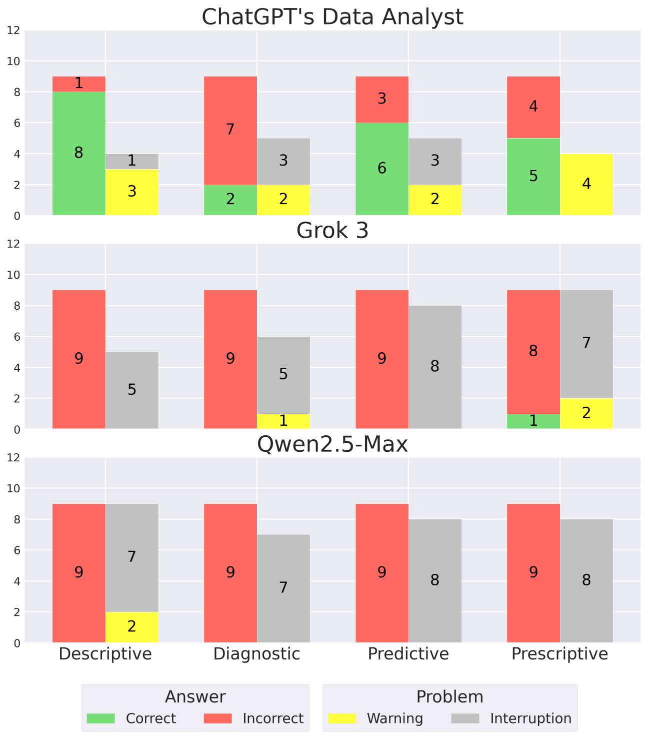

Figure 4. Summarized Results for Dataset 3.

rather than hallucinated content.

Grok 3 was marked by frequent interruptions and unreliable scripts across task types; its errors commonly derive from type-conversion faults (e.g., attempting arithmetic on string-based intervals) and incorrect use of dataset keys, producing subsets incompatible with the posed queries. **Qwen-Max 2.5** generated outputs in a small fraction of prompts (≈16.7%) and failed to produce correct responses; its principal shortcomings were inability to align requested measures with dataset keys and the generation of speculative or partial code when extraction failed.

Hallucination behavior differs across models: **ChatGPT's Data Analyst** maintained fidelity to the dataset, while Grok 3 and Qwen sometimes generated invented values or mismatched keys when the model lacked direct access to required fields. These hallucinations correlated with cases where interruptions or type errors prevented proper extraction from the dataset.

Grouping results by difficulty reveals that ChatGPT reliably handles most easy and medium tasks but shows reduced accuracy on challenging items—particularly diagnostic problems requiring nuanced key-mapping. Grok's sporadic successes are limited and concentrated at medium difficulty (one medium prescriptive task), while Qwen fails across difficulty tiers, with a handful of outputs restricted to simpler prompts.

4.4 Discussion

Addressing RQ, our findings clearly delineate the comparative accuracy of ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max across various data-analysis tasks and complexity levels. ChatGPT's Data Analyst consistently demonstrated superior performance, particularly in descriptive and predictive tasks, achieving an overall accuracy of 72.22%. Grok 3 occupied an intermediate position with 45.37% accuracy, outperforming Qwen2.5-Max, which showed significantly lower accuracy at 8.33%. These results indicate a clear hierarchy in performance among the evaluated models, with ChatGPT's Data Analyst setting a benchmark for accuracy in this comparative study.

The statistical analysis confirmed that these performance differences were significant across all models, as shown in Table 1. ChatGPT achieved the highest global accuracy (72.2%, IC95%: 63.1–79.8), followed by Grok (45.4%, IC95%: 36.3–54.8), while Qwen remained substantially below (8.3%, IC95%: 4.4–15.1). The chi-square tests by category (Table 2) further indicated significant differences among models in Descriptive, Diagnostic, Predictive, and Prescriptive tasks (all $p < 0.001$). Post-hoc pairwise comparisons with Bonferroni correction (Table 3) revealed that ChatGPT consistently outperformed Qwen ($p < 0.001$ in all tasks). In comparison with Grok, ChatGPT was significantly superior in Descriptive and Predictive tasks, while the differences in Diagnostic and Prescriptive were not statistically significant. Overall, Grok occupied an intermediate position, outperforming Qwen in most tasks but still lagging behind ChatGPT. Furthermore, the statistical analysis strongly supports H1, confirming statistically significant differences in task accuracy between the three models across all task types (descriptive, diagnostic, predictive, prescriptive) and complexity levels. As evidenced by the chi-square tests (Table 2) and post-hoc pairwise comparisons (Table 3), the performance disparities observed are not merely coincidental but are statistically robust. Specifically, ChatGPT significantly outperformed both Grok and Qwen in most categories, while Grok showed a significant advantage over Qwen, thereby validating our hypothesis regarding performance variations among the models.

Table 1. Global accuracy of the models with 95% confidence intervals.

Model	Correct	Incorrect	Accuracy	CI Lower	CI Upper
ChatGPT	78	30	0.722	0.631	0.798
Grok	49	59	0.454	0.363	0.548
Qwen	9	99	0.083	0.044	0.151

Based on cases involving warnings and interruptions, a clear distinction emerges between code generation aimed at extracting responses from datasets and direct transformation of these datasets through code execution. Models such as Qwen-Max 2.5 generate general speculations about dataset states without directly modifying them, subsequently developing scripts outlining steps necessary to reach the final answer. Similarly, Grok 3 initially extracts a data sample, capturing a preliminary state of the dataset from which strategies for answering the question are formulated.

Conversely, ChatGPT's Data Analyst directly modifies the original dataset state, allowing answers to be formulated based on a more realistic and comprehensive data perspective. Another distinguishing feature of ChatGPT's Data Analyst is its ability to reconstruct code dynamically from real-time extracted results, thereby enabling the reconstruction of strategies used to reach the final answer to the posed question.

The implications of these methodological differences are

Table 2. Results by task category, with model accuracy and p-values from chi-square tests.

Category	Model	Correct	Incorrect	Accuracy	p-value
Descriptive	ChatGPT	24	3	0.889	5.29e-06
	Grok	15	12	0.556	
	Qwen	6	21	0.222	
Diagnostic	ChatGPT	18	9	0.667	3.44e-05
	Grok	13	14	0.481	
	Qwen	2	25	0.074	
Predictive	ChatGPT	19	8	0.704	1.40e-06
	Grok	8	19	0.296	
	Qwen	1	26	0.037	
Prescriptive	ChatGPT	17	10	0.630	3.56e-06
	Grok	13	14	0.481	
	Qwen	0	27	0.000	

evident in the varying performance across each tested dataset. Datasets requiring extensive data transformation and processing, particularly those where identical information may be represented using diverse naming conventions or formats, increase complexity for tools relying solely on superficial dataset information. This leads to a higher incidence of errors, either as warnings during execution or interruptions caused by more systematic failures.

This scenario is particularly noticeable in Dataset 3, where Qwen-Max 2.5 failed to provide any correct answers, Grok 3 correctly answered only one question, and ChatGPT's Data Analyst successfully answered 21 questions. Dataset 3 predominantly contains data as strings and intervals requiring significant transformation for answer extraction. Conversely, datasets permitting extraction of general state overviews facilitated superior performance from Grok 3 and enabled Qwen-Max 2.5 to achieve its best results with Dataset 1.

Table 3. Post-hoc pairwise comparisons between models with Bonferroni correction.

Category	Comparison	Adjusted p-value
Global	ChatGPT × Grok	1.83e-04
	ChatGPT × Qwen	3.13e-21
	Grok × Qwen	2.46e-09
Descriptive	ChatGPT × Grok	0.0187
	ChatGPT × Qwen	2.47e-06
	Grok × Qwen	0.0360
Diagnostic	ChatGPT × Grok	0.506
	ChatGPT × Qwen	1.96e-05
	Grok × Qwen	0.0025
Predictive	ChatGPT × Grok	0.0083
	ChatGPT × Qwen	1.18e-06
	Grok × Qwen	0.0318
Prescriptive	ChatGPT × Grok	0.820
	ChatGPT × Qwen	1.90e-06
	Grok × Qwen	1.05e-04

Key limitations identified include a 20MB data processing limit and the lack of support for robust libraries such as hmmlearn, imblearn, Keras, and TensorFlow. Additionally, the tool was prone to hallucinations and operational failures, which can necessitate session restarts, thereby affecting the efficiency and effectiveness of the analytical process.

5 Conclusion

This study evaluated the comparative accuracy of ChatGPT's Data Analyst, Grok 3, and Qwen2.5-Max in diverse data-analysis tasks, directly addressing the stated Research Question and Hypothesis. Our findings demonstrate that ChatGPT's Data Analyst significantly outperforms its counterparts across most task types and complexity levels, thereby confirming the alternative hypothesis H_1. The findings indicate the superiority of models capable of directly modifying databases, thus ensuring more robust verification and execution of analytical processes. Although LLMs offer transformative potential in automating data analysis, they face significant technical barriers that pose distinct threats to validity: integration with advanced libraries affects internal validity by limiting the range of algorithms testable; constraints on dataset size challenge external validity by reducing generalizability to high-complexity scenarios; and difficulties in mapping conceptual user queries to implicit dataset representations threaten construct validity. Although models like Qwen and Grok can generate effective code, they require substantial external monitoring to continuously verify and update information sets.

Our findings, particularly ChatGPT's Data Analyst proficiency in data transformation tasks (superior performance on Dataset 3), align with and extend the work of Cheng *et al.* [2023] in the context of evaluating how effective an LLM can be as a Data Analyst. However, this superiority diminishes in complex prescriptive tasks, a limitation not fully explored in broader benchmarks such as Zhang *et al.* [2024]. While the results corroborate the potential for LLMs to automate data transformations, they reveal notable variability when moving from descriptive to prescriptive analytics, indicating that current models are more reliable for structured, well-defined tasks than for ambiguous prescriptive scenarios.

The study contributes valuable perspectives to the literature, enriching theoretical and practical knowledge about the application of LLMs. Nonetheless, limitations must be acknowledged: the definition of difficulty levels relied on subjective judgment, potentially introducing bias, and the use of ChatGPT as a refinement step in question construction raises concerns of methodological circularity. Future work may involve evaluating more advanced LLMs and exploring techniques like fine-tuning or prompt optimization to assess their impact on error reduction and analytical reliability. In addition, subject matter experts may be involved in the question creation process, alongside cross-validation techniques, to ensure the quality of the content and to better understand the reasoning processes of both human and artificial intelligence systems. Future research should also specifically address the identified gap regarding database exploration and speculation.

References

Abaskohi, A., Ramesh, A. V., Nanisetty, S., Goel, C., Vazquez, D., Pal, C., Gella, S., Carenini, G., and Laradji, I. H. (2025). Agentada: Skill-adaptive data analytics for tailored insight discovery.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., *et al.* (2024). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Banco Central do Brasil (2024). Sistema de operações do crédito rural e do proagro (sicor). Acesso em: 28 ago. 2025.

Banerjee, A., Bandyopadhyay, T., and Acharya, P. (2013). Data analytics: Hyped up aspirations or true potential? *Vikalpa*, 38(4):1–12.

Cheng, L., Li, X., and Bing, L. (2023). Is gpt-4 a good data analyst? *Journal of Artificial Intelligence Research*, Findings of EMNLP 2023:9496–9514.

Daibes, M. and Lima, B. B. (2024). Cracking the heart code: Using chatgpt's data analyst feature for cardiovascular imaging research. *The International Journal of Cardiovascular Imaging*, pages 1–2.

de Miranda, B. A. and Campelo, C. E. C. (2024). How effective is an llm-based data analysis automation tool? a case study with chatgpt's data analyst. In *Anais do XXXIX Simpósio Brasileiro de Banco de Dados (SBBD)*, pages 287–299, Florianópolis, SC, Brazil. Sociedade Brasileira de Computação (SBC). DOI: 10.5753/sbbd.2024.240841.

Ding, B., Qin, C., Liu, L., Chia, Y. K., Li, B., Joty, S., and Bing, L. (2023). Is gpt-3 a good data annotator? *arXiv preprint arXiv:2305.00899*.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., *et al.* (2024). The llama 3 herd of models.

Hu, X., Zhao, Z., Wei, S., Chai, Z., Ma, Q., Wang, G., Wang, X., Su, J., Xu, J., Zhu, M., Cheng, Y., Yuan, J., Li, J., Kuang, K., Yang, Y., Yang, H., and Wu, F. (2024). InfiAgent-DABench: Evaluating Agents on Data Analysis Tasks. *arXiv preprint arXiv:2401.05507*.

Jaimovitch-López, G., Ferri, C., Hernández-Orallo, J., Martínez-Plumed, F., and Ramírez-Quintana, M. J. (2022). Can language models automate data wrangling? *Machine Learning*, 112:2053–2082.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., *et al.* (2023). Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Kasetty, T., Mahajan, D., Dziugaite, G. K., Drouin, A., and Sridhar, D. (2024). Evaluating interventional reasoning capabilities of large language models. *arXiv preprint arXiv:2404.05545*. DOI: https://doi.org/10.48550/arXiv.2404.05545.

Lino, R. C. (2021). O impacto da analítica hoje e no futuro. Master's thesis, Universidade de Lisboa (Portugal).

Liu, X., Wu, Z., Wu, X., Lu, P., Chang, K.-W., and Feng, Y. (2024). Are llms capable of data-based statistical and causal reasoning? benchmarking advanced quantitative reasoning with data. DOI: https://doi.org/10.48550/arXiv.2402.17644.

Nasseri, M., Brandtner, P., Zimmermann, R., Falatouri, T., Darbanian, F., and Obinwanne, T. (2023). Applications of large language models (llms) in business analytics – exemplary use cases in data preparation tasks. 14059:182–198.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., *et al.* (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Ramannavar, M. and Sidnal, N. S. (2016). Big data and analytics—a journey through basic concepts to research issues. In Suresh, L. and Panigrahi, B., editors, *Proceedings of the International Conference on Soft Computing Systems*, volume 398 of *Advances in Intelligent Systems and Computing*, pages 291–306. Springer India.

Sharma, A., Li, X., Guan, H., Sun, G., Zhang, L., Wang, L., Wu, K., Cao, L., Zhu, E., Sim, A., Wu, T., and Zou, J. (2023). Automatic data transformation using large language model - an experimental study on building energy data. pages 1824–1834. DOI: 10.1109/BigData59044.2023.10386931.

Solaiman, I., Brundage, M., Clark, J., Askell, A., Herbert-Voss, A., Wu, J., Radford, A., and Wang, J. (2019). Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*.

Team, G., Georgiev, P., Lei, V. I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., *et al.* (2024). Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context.

Team, Q. (2024). Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

United Nations Children's Fund (UNICEF) (2024). Hiv and aids global and regional trends.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., ichter, b., Xia, F., Chi, E., Le, Q. V., and Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

xAI (2025). Grok 3: Beta—the age of reasoning agents. `https://x.ai/news/grok-3`. Accessed: 2025-04-16.

Zhang, H., Dong, Y., Xiao, C., and Oyamada, M. (2023). Large language models as data preprocessors. *arXiv preprint arXiv:2305.00899*.

Zhang, Y., Jiang, Q., Han, X., Chen, N., Yang, Y., and Ren, K. (2024). Benchmarking data science agents. arXiv:2402.17168.