Simulating Tabular Datasets through LLMs to Rapidly Explore Hypotheses about Real-World Entities

Miguel Zabaleta, Joel Lehman[1]

[1]Stochastic Labs
mzabaletasar@gmail.com, lehman.154@gmail.com

Abstract

Do horror writers have worse childhoods than other writers? Though biographical details are known about many writers, quantitatively exploring such a qualitative hypothesis requires significant human effort, e.g. to sift through many biographies and interviews of writers and to iteratively search for quantitative features that reflect what is qualitatively of interest. This paper explores the potential to quickly prototype these kinds of hypotheses through (1) applying LLMs to estimate properties of concrete entities like specific people, companies, books, kinds of animals, and countries; (2) performing off-the-shelf analysis methods to reveal possible relationships among such properties (e.g. linear regression); and towards further automation, (3) applying LLMs to suggest the quantitative properties themselves that could help ground a particular qualitative hypothesis (e.g. number of adverse childhood events, in the context of the running example). The hope is to allow sifting through hypotheses more quickly through collaboration between human and machine. Our experiments highlight that indeed, LLMs can serve as useful estimators of tabular data about specific entities across a range of domains, and that such estimations improve with model scale. Further, initial experiments demonstrate the potential of LLMs to map a qualitative hypothesis of interest to relevant concrete variables that the LLM can then estimate. The conclusion is that LLMs offer intriguing potential to help illuminate scientifically interesting patterns latent within the internet-scale data they are trained upon.

Code —
https://github.com/mzabaletasar/llm_hypoth_simulation

1   Introduction

While science often involves generating new data to explore hypotheses, we likely underappreciate what possible insights hide in plain sight within the vast expanse of already-existing data. One problem is that it takes significant human effort to unearth and clarify hypotheses from diverse data sources. For example, there exist many written biographies, which in aggregate may speak to important patterns of the human condition, e.g. how and if aspects of childhood experience relate to adult life choices or relationship, or how personality and mental health interact. However, such information is unstructured and potentially spread across many different texts; for many questions of interest, no one has yet made the effort to curate a specific dataset from such diverse sources.

To explore these kinds of questions quantitatively within existing data requires: (1) seeking quantitative variables that are indicative of more qualitative properties of interest (e.g. how many adverse childhood experiences, or ACEs (Boullier and Blair 2018) a specific person experienced, or estimating their OCEAN personality traits (Roccas et al. 2002)); and (2) sifting through diverse unstructured collections of text to ground or estimate those quantities (e.g. reading several biographies of a figure to count their ACEs). To approach both steps manually often requires significant labor, domain expertise, and trial and error. As a result of these costs, we do not thoroughly mine what lies latent within existing data.

Interestingly, large language models (LLMs) are trained over an enormous corpus of human cultural output, and continue to advance in their capabilities to answer arbitrary queries about specific entities. Thus, the main idea in this paper is to leverage LLMs for quick-and-dirty explorations of hypotheses about real-world entities (like people, countries, books, and activities). In particular, given a high-level hypothesis (such as "Do horror writers have worse childhoods than other authors?"), an LLM can (1) suggest quantitative variables to ground such a hypothesis that are plausibly within its training corpus (e.g. "Did this person's parents get a divorce"), (2) generate a list of concrete entities (e.g. 100 well-known horror writers and 100 well-known writers of other genres), and (3) estimate the concrete variables for each entity (e.g. "Did Steven King's parents get a divorce?").

In this way, from an initial rough idea, an LLM can generate an approximate artisanal dataset, providing a preliminary way of exploring the hypothesis. The hope is that this estimation, while not perfect, could serve as an accelerant for active brainstorming, and could fit into a larger pipeline of science. For example, correlations between variables could also be automatically calculated in the simulated dataset, and if a strong and interesting correlation is found, it could motivate the effort to curate by hand a validated dataset, or to gather new data in service of the hypothesis (e.g. a more controlled survey of aspiring writers and their ACE scores).

Because this kind of data generation (for a moderate-sized dataset) is inexpensive and fast, it can enable faster iteration cycles of hypothesis brainstorming and debugging.

The experiments in this paper focus mainly on step (3) above (e.g. estimating concrete properties of concrete entities), although they also touch on steps (1) and (2). In particular, we find across several domains that indeed LLMs can generate useful datasets about real-world entities, and that such datasets increase in fidelity with model scale. We also show in a preliminary experiment that LLMs can also translate high-level hypotheses into concrete variables, and that (perhaps unsurprisingly) they are adept at creating lists of entities appropriate for exploring a hypothesis (e.g. a list of horror writers).

The conclusion is that LLM pipelines may provide novel ways for quickly exploring hypotheses related to real-world entities, helping us to better leverage and understand the oceanic data already generated by humanity.

2 Background

LLMs as simulators. Several previous studies have demonstrated the potential for LLMs to act as simulators, often focusing on human behaviors or responses. For instance, (Argyle et al. 2023) demonstrate that LLMs can represent diverse human subpopulations and simulate survey result probabilities based on demographic information, such as predicting voting behavior given race, gender, and political affiliation. Similarly, other works leverage LLMs to replicate human behavior experiments, showing that LLMs can reproduce well-established findings from prior human subject studies (Aher, Arriaga, and Kalai 2023); and others simulate user satisfaction scores to optimize a dialogue system (Hu et al. 2023).

Our work aims to generalize beyond human-centered applications by focusing on simulating the properties of any class of specific entities, such as animals and countries (although we also include an experiment about athletes). Our focus is different as well: most previous studies explore simulations of human behavior for experimental replication, while we aim to use LLMs as a tool for quickly simulating datasets that can inform the exploration of broader scientific hypotheses in an efficient, exploratory manner.

Similarly, (Cohen et al. 2023) demonstrate the potential for extracting structured knowledge from LLMs to build knowledge graphs, which supports the idea that LLMs will be useful tools for simulating datasets on the fly. We build upon this idea to generate synthetic data for exploring novel relationships and hypotheses.

More broadly, synthetic data generation has been widely studied for its ability to improve machine learning models, address privacy concerns, and augment datasets (Lu et al. 2024b). However, most applications focus on tasks like model enhancement or privacy-preserving data generation, rather than on hypothesis-driven exploration. Recent work has begun to explore the use of LLMs to generate synthetic datasets, but most often with the aim to increase the performance of LLMs rather than to enable rapid hypothesis testing.

Hypothesis Generation. LLMs are increasingly being applied for hypothesis generation, with approaches generally falling into three categories: text-based, data-driven, and hybrid methods.

Text-based approaches leverage LLMs to synthesize hypotheses directly from given textual data. For example, (Tong et al. 2024) explore generating psychological hypotheses from academic articles. Their method relies on extracting a causal graph from the corpus of literature for hypothesis generation. Data-driven approaches focus on uncovering patterns in structured datasets. For instance, (Zhou et al. 2024) extracts hypotheses from labeled data, enabling automated discovery of insights. However, this reliance on existing datasets poses challenges when suitable labeled data is unavailable, restricting its scope in exploratory or novel domains. Hybrid approaches combine insights from both literature and data. (Xiong et al. 2024) demonstrates how LLMs can integrate knowledge from text and structured data to propose hypotheses.

In contrast to these approaches, our work focuses not on generating hypotheses directly, but on simulating datasets from which hypotheses can be explored. By leveraging LLMs as simulators of the properties of concrete entities, we enable a structured and data-driven pathway to hypothesis prototyping, mitigating the pitfalls of forgetting and compounding errors observed in direct hypothesis generation (Liu et al. 2024). Furthermore, in domains where hallucination poses a significant challenge, we apply a self-correction mechanism (Madaan et al. 2023) to improve simulation quality, which in future work could be further addressed with retrieval-augmented generation.

3 Approach

The overarching ambition in this paper is to move towards automating more of the process of exploring interesting and important patterns latent within existing internet-scale data, to advance our scientific understanding of the world and make the most of the data society has already generated. One significant obstacle to prototyping a hypothesis within society-scale data is to curate a dataset by hand that can reveal evidence about the hypothesis, which requires sifting through many data sources and carefully translating unstructured data into structured, quantitative tables.

The general approach in this paper to avoid that cost, is to generate approximate tabular datasets by querying LLMs. Such tabular data is a powerful way of exploring patterns, where we can consider each row as an entity, and each column as a property of that entity. The idea is that training data for LLMs implicitly includes many properties of real-world entities of scientific interest, like people, animals, activities, and countries. Information about a particular entity may be spread across many different documents and contexts, and usefully centralized into the weights of the LLM through the training process (Cohen et al. 2023).

This naturally leads to a simple approach to simulate an artisanal tabular dataset fit to explore a particular hypothesis. First, we consider the case where an experimenter provides a list of entities (e.g. like particular animals) and properties
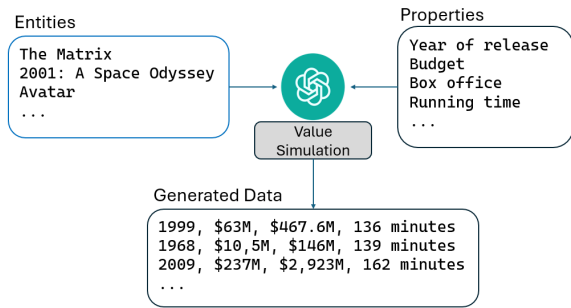
Figure 1: LLM-driven Dataset Simulation. Given a list of entities and properties, the method is to call an LLM for each combination of entity and property to estimate the value of the property for that entity.

(e.g. whether they lay eggs, or have wings): Then, the approach is to simply query the LLM to estimate each property for each entity (see Figure 1). We call this method **LLM-driven Dataset Simulation**. Our first set of experiments explores the ability of LLMs to simulate datasets with reasonable fidelity, i.e. whether the relationships among variables in the simulated dataset reflect those in a human-validated dataset.

To further automate the process of exploring hypotheses, we can use LLM-driven Dataset Simulation as a building block, and additionally apply LLMs to help translate a qualitative high-level hypothesis into the rows and columns of the tabular dataset (e.g. for it to also create the list of real-world entities, and the list of properties of those entities relevant to explore the hypothesis). We call this method **Hypothesis-driven Dataset Simulation**, and this broader pipeline is shown in Figure 2. The idea is that an experimenter can describe the hypothesis they want to explore, and a chain of LLM calls can orchestrate creating the rows and columns of the dataset, as well as to simulate the dataset itself. In more detail about the components of the Hypothesis-driven Dataset Simulation pipeline:

Prompt Generation. The first stage involves generating the system and user prompts required to estimate property values. An LLM is prompted with the experimenter-provided hypothesis description (and optionally the desired number of properties) along with a one-shot example, and is directed to produce (1) a system prompt defining the role the LLM should adopt, and (2) a user prompt specifying the task for the LLM (i.e. to generate a list of key properties given the hypothesis description). These generated prompts are used to guide the subsequent stages of property generation. *Note that examples of each of the prompts in this and following sections can be found in the appendix.*

Property Simulation. Using the generated prompts, another LLM call simulates property descriptions in free-form text, such as Property Name: "Average Happiness Level", Description: "The average self-reported happiness level of individuals in this entity.", Possible Values: [0-10].

Property Parsing. To structure the simulated properties, an LLM extractor parses the free-form text into a consistent format: property-name: [description, possible values or ranges]. This structured format is then combined with the list of entities to prompt the LLM for property values.

Self-Correction. After simulating property values, an optional self-correction step ensures robustness (Madaan et al. 2023). The LLM is prompted to evaluate the accuracy of each value given the property description and range, provide reasoning for its assessment, and output a corrected (or confirmed) value. The aim is to improve the reliability and consistency of the simulated dataset.

4 LLM-driven Dataset Simulation Experiments

The first set of experiments explore LLMs' ability to simulate useful tabular datasets, given a list of entities and properties. We begin with a simple dataset of binary characteristics of animals as a didactic toy example that we expect to be well within the capabilities of LLMs; we also explore a more difficult domain that involves specific demographic properties of countries, and complicated constructed indicators (e.g. of how egalitarian a country is), where it is less clear that LLMs would be adept, as a way of probing the limits of this technique. Note that in these experiments we use existing ground-truth datasets as a grounded proxy for the situation of real interest, e.g. to simulate novel datasets; while there is some risk of LLMs memorizing these datasets (as LLMs are at least aware of the Zoo dataset, described next), we find in later experiments that the method does indeed generalize to novel datasets.

4.1 Zoo Domain

Description. In this experiment, we assess the ability of LLMs to simulate the well-known "Zoo Dataset" from the UCI Machine Learning Repository (Forsyth 1990). This dataset consists of 101 animal instances (e.g. vampire bat, aardvark), each characterized by 16 binary features (e.g. hair, feathers, teeth) and a categorical target variable representing the animal's type (e.g. mammal, insect). Our aim is to determine whether LLMs can replicate this dataset accurately. Note that the LLM is conditioned only on the plain-text names of the animals and features.

Motivation. We chose this dataset as a first exploration because of its intuitive simplicity. It is clear that LLM training should include simple biological features of animals within it, and thus this provides an easy-to-understand sanity check for the approach. The Zoo domain also illustrates how LLMs can be applied to biological or ecological datasets, offering potential for hypothesis generation in specialized fields.

Experiment setting. To assess the accuracy of individual simulated binary properties, we compared the outputs of the LLMs to the ground-truth dataset. The quality of properties was evaluated using accuracy as the primary metric for both animal features (independent variables) and animal type (dependent variable). We used GPT-4o-mini and the prompting
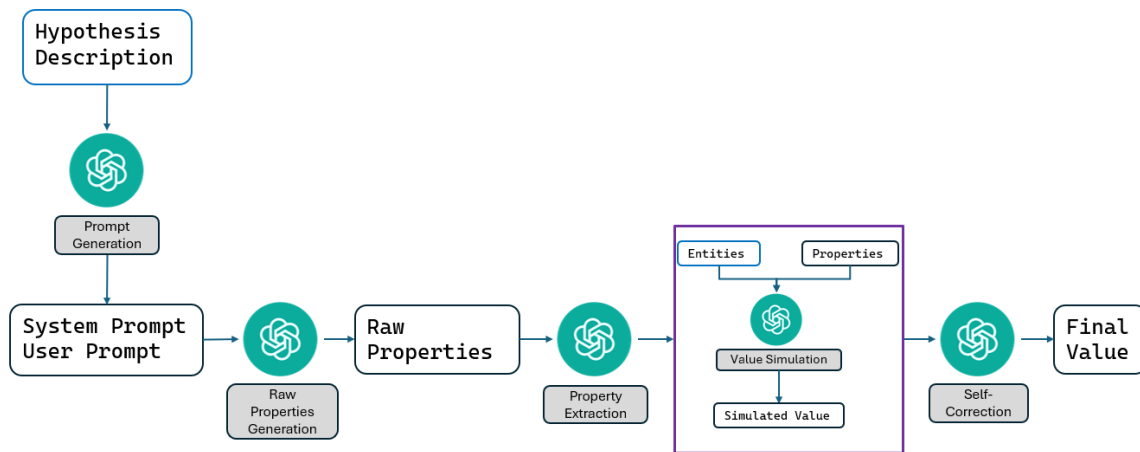
Figure 2: Architecture of the hypothesis-driven simulation. The pipeline starts with a description of the hypothesis to explore, followed by the prompts that will generate the raw properties. After extracting the properties, a datapoint is then generated for each property of each entity, which undergoes an optional self-correction prompt.

strategy of directly querying property values in a Pythonic dictionary format.

We also evaluated the utility of simulated datasets for exploratory data analysis and hypothesis testing. This process emulated a typical scientific workflow: a standard analysis model, such as linear or logistic regression, was trained on the simulated training data and then run on unseen simulated validation data. The predictions on the (simulated) validation set were then compared to real-world validation labels to assess performance. The idea is to get a sense of how well an analysis method applied on simulated data captures the same patterns as in the real data.

To quantify how closely the simulated data approximates real-world patterns, we introduce a **Simulation Error Gap metric**. This metric measures the difference in generalization error between models trained on simulated data and the upper-bound performance achieved by fitting models on ground-truth training data. A smaller Simulation Error Gap reflects a higher fidelity of the simulated data in capturing the underlying relationships within the real-world dataset. In this domain, a logistic regression model was trained on 70% of the data, and generalization error was measured by accuracy on the ground-truth held-out data.

Simulation Fidelity of Properties.   Overall, the results indicate that the simulator effectively models binary properties in the domain. As shown in Figure 3, the average accuracy across all properties is 0.923, suggesting that the simulated data closely approximates the characteristics of the real data. Much of the remaining error is due to an ambiguous property called "catsize", which highlights that an LLM requires a clear semantic description of the property to be simulated.

Asking the LLM to Instead Directly Output Correlation Coefficients.   As a control experiment, we tested the direct generation of hypotheses by asking the LLM to estimate correlations between each independent variable and each class (e.g. animal type). In particular, the Matthews correlation
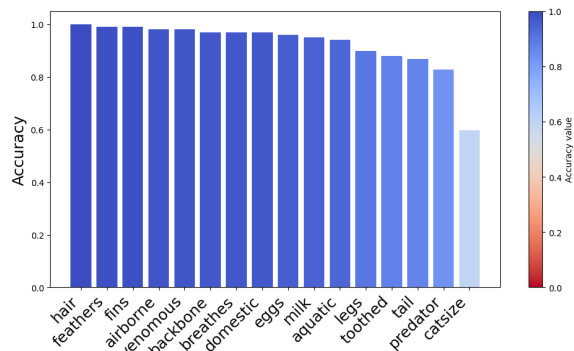

Figure 3: Simulation accuracy for properties in the Zoo domain. Shown are how accurately the LLM is able to simulate each property in the Zoo domain across all the animals in the dataset. Accuracy is generally high, although the LLM understandably struggles with the ambiguous variable name "catsize". The conclusion is that the approach is viable, although it is important to give the model sufficient context about the property it is to simulate.

coefficient, which is appropriate for binary variables and multi-class output. Interestingly, we found an average absolute difference of 0.321 between the LLM's estimations and the real coefficients, highlighting the limited capabilities of LLMs to be used as direct estimators of relationships between variables even in quite simple domains.

Training Classifiers on Simulated Data. Perhaps unsurprisingly, in simulating class labels in this dataset (e.g. mapping an animal name to its type—such as mammals, birds, and reptiles), the simulator performs very well, achieving perfect accuracy.

To further assess the simulator's utility for predictive modeling, we trained a logistic regression model on the simulated data and evaluated it on real validation data. The model achieved an accuracy of 0.833 when trained on the simulated data, compared to an accuracy of 0.933 on real data. This resulted in a simulation error gap of 0.1, indicating a modest difference between the simulated and real data for this particular predictive task.

In summary, these results serve as some grounding evidence that LLMs can simulate datasets with reasonable fidelity. The next experiment explores a more difficult domain.

4.2 Countries Domain

Description. The dataset in this experiment is designed to explore how demographic features of countries correlate with their Egalitarian Democracy Index (EDI) scores; the EDI is an index that combines information on voting rights, the freedom and fairness of elections, freedoms of association and expression, as well as the extent to which the protection of rights, access to power, and distribution of resources is equal (Sigman and Lindberg 2019). It ranges from 0 to 1 (most democratic). Our reference dataset combines various indicators, such as population statistics across age groups and genders from the World Bank (The World Bank 2024), with EDI scores from Our World in Data (V-Dem 2022), all from 2022. For example, properties include metrics like 'Population ages 60-64, male (% of male population)', 'Regulatory Quality: Percentile Rank, Upper Bound of 90% Confidence Interval', and 'Political Stability and Absence of Violence/Terrorism: Number of Sources.' The goal is to test whether LLMs can simulate tabular data that reflects real-world patterns, facilitating rapid hypothesis testing in more complicated settings.

Motivation. This dataset was chosen because it is more specialized and requires estimating continuous variables with various ranges, and requires the LLM to handle ambiguous property names (e.g. 'Regulatory Quality: Percentile Rank, Upper Bound of 90% Confidence Interval'). In contrast to the simplicity of the Zoo domain, this provides a more challenging environment to further develop and test dataset simulation techniques. It also highlights how dataset simulation may be useful for hypothesis generation in areas of economics and policy.

Experiment setting. After pre-processing (see Appendix A.1), a random sample of 50 countries is selected from a pool of 155 countries, and 10 random properties are chosen from a pool of 120 properties. In the Countries Domain, the quality of predictor variables was evaluated using correlation with actual values, while the Egalitarian Democracy Index was assessed using Mean Absolute Error (MAE).

For the analysis method, we trained a linear regression model on 80% of the simulated data, and the generalization error of the model was measured by Median Absolute Error (MedAE), as in contrast to the Zoo domain, the dependent variable (EDI) is continuous.

Explorations to Increase Simulation Fidelity. In this more challenging domain, we explored several techniques to improve simulation performance. One approach was to condition the dependent variable (EDI) on the previously simulated property values for a particular country. Another approach was to take certain complex properties, such as demographic percentages, and use few-shot learning strategies to help ground out the variable's range. Interestingly, despite extensive experimentation (e.g., conditioning on outliers or randomly-chosen data points), no consistently superior approach was identified.

Impact of Model size. We tested three models: Llama3-8B, Llama3-70B, and GPT-4o-mini. GPT-4o-mini consistently outperformed the others, producing the most accurate and contextually relevant simulations. As a result, GPT-4o-mini was used for all subsequent experiments in other domains. Table 1 shows the impact of model size on simulation quality, measured by average correlation between simulated and real data points, Mean Absolute Error in simulated EDI (EDI MAE), and simulation error gap. The models compared include LLaMA-3-8b, LLaMA-3-70b, and GPT-4o-mini. As seen in Table 1, performance improves across all metrics as model size increases. Further, Figure 4, visualizes the fidelity of predictive models increasing across different model sizes.

Model	Average Corr.	EDI MAE	Sim. Error Gap
LLaMA-3-8b	0.221	0.134	0.064
LLaMA-3-70b	0.644	0.189	**0.036**
GPT-4o-mini	**0.738**	**0.119**	**0.036**

Table 1: Simulators' performance by model size

In summary, these results highlight that larger models significantly improve simulation quality. Increased model size leads to higher correlation with real-world data, reduced error gaps, and more accurate predictions of EDI values, making larger models more reliable for hypothesis generation.

Impact of Prompting strategy. We examined two key factors in prompting: prompt style and output format. For prompt style, we explored prompts that were direct queries for specific data points ("Make your best guess about this value..."), with prompts that told the LLM it was an expert and was tasked to complete a report about the property at hand ("You are an expert historian. Complete the following document..."). Output format compared structured formats
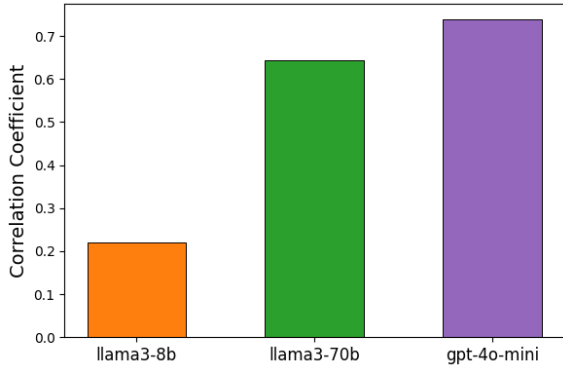
Figure 4: Correlation coefficients by model size for Countries domain. Shown are how well the simulated properties correlate with the ground-truth properties across all entities. The conclusion is that the fidelity of the simulations improves with model scale and capability.

(e.g., Python dictionaries) with instead outputting an answer in natural language. This analysis reveals how different presentation styles affect data consistency and realism in simulations.

The prompting strategy that proved most effective in this experiment involved directly querying property values and using a Pythonic dictionary format to structure the data. This strategy was adopted for all subsequent experiments. Table 2 shows the effect of different prompting strategies on simulation quality.

Prompting	Average Corr.	EDI MAE	Sim. Error Gap
direct-descriptive	0.738	**0.119**	0.036
direct-structured	**0.770**	0.132	**0.011**
report-descriptive	0.394	0.171	0.087
report-structured	0.253	0.160	0.153

Table 2: Simulators' performance by prompting strategy in the Countries domain.

The direct-structured strategy, where values are requested in a structured, Pythonic dictionary format, consistently outperforms the other strategies, achieving the highest average correlation (0.770) and the lowest simulation error gap (0.011). The direct-descriptive strategy, which asks for values as free-text words, also performs well, achieving an average correlation of 0.738, an EDI MAE of 0.119, and a simulation error gap of 0.036. These results suggest that asking for data in a structured format (as in direct-structured) leads to more precise simulations, while the direct-descriptive strategy still provides reliable results. In contrast, the report-descriptive and report-structured strategies show significantly weaker performance. Both report-based strategies involve completing partial data rather than requesting full values, leading to less accurate and more inconsistent simulations. These patterns are further illustrated in Figure A.1, which compares the average correlation metrics across the different prompting strategies.

Asking the LLM to Instead Directly Output Correlation Coefficients. Similar to the Zoo domain, we also tasked an LLM with directly estimating the correlations between some of the properties and the EDI. The results showed that, on average, there was a 0.483 difference between the correlations suggested by the LLM and the real correlations in the data. This discrepancy is significantly larger than the differences observed when using our simulation-based method (as illustrated in the figure in the Appendix A.3), further highlighting the advantages of simulating data before analyzing patterns.

5 Towards Hypothesis-Driven Dataset Simulation

The previous experiments explored the ability of LLMs to simulate datasets in a controlled setting where ground-truth data was available (e.g. by having the LLM simulate existing datasets). In this section, we move more towards the setting of direct interest, where we want to explore a hypothesis but do not have a pre-existing dataset. We also experiment here with greater LLM autonomy: In addition to having the LLM simulate the data, we also have it map from a high-level hypothesis to the properties worth simulating to explore it. Further experiments explore having the LLM also generate the list of entities of interest (e.g. particular sports figures in this case). In this way, we move more towards having an LLM assistant that can help an experimenter quickly brainstorm and explore potential hypotheses.

Description. In this section, we evaluate the ability of LLMs to generate datasets based on qualitative hypotheses. Specifically, we explore the relationship between an athlete's sport type (team vs. individual), the number of major injuries (lasting over two months), and peak performance age. The system receives a prompt outlining the hypothesis along with a list of 40 athletes (20 soccer players and 20 tennis players). The simulator was provided only with the hypothesis and a list of entities, from which it generated data corresponding to the key properties mentioned. For validation purposes, ground-truth values for the number of injuries were collected from Tennis Explorer for tennis players and Transfermarkt for soccer players, while the peak performance age was sourced using Perplexity (Perplexity 2024) (as a proxy for exhaustive Google searches).

To justify our use of Perplexity for sourcing the peak performance age, we conducted spot checks comparing it to direct LLM queries (e.g., asking ChatGPT). Specifically, we asked both systems for the place of birth of 20 lesser-known soccer players from the Spanish soccer league. While ChatGPT accurately identified only 10 out of 20, Perplexity correctly retrieved all 20 places of birth. This significant difference (Fischer's exact test; $p < 0.001$)) is likely due to Perplexity's use of Retrieval-Augmented Generation (RAG), which enhances factual accuracy by grounding the inferences of the LLM in externally retrieved data (Shuster et al. 2022; Ren et al. 2023).

Motivation. This task tests whether LLMs can simulate data for hypotheses that would be time-consuming to col-

lect in the real world. Information such as the number of injuries or peak performance age is often scarce, so generating these values synthetically could accelerate hypothesis testing. This experiment demonstrates the potential of LLMs to convert high-level qualitative ideas into structured, usable data, making it easier to explore relationships in data-sparse fields.

Experiment setting. The quality of the simulated properties were evaluated using correlation and MAE with the real (hand-gathered) data points as metrics. A linear regression model was fitted as the analysis method, and MAE was used to measure the generalization error (on 20% of the simulated data). GPT-4o-mini and directly querying for property values in a Pythonic dictionary format were used in this experiment.

LLMs for Hypothesis Mapping. The results indicate that the LLM-based simulator was successful in identifying and generating relevant quantitative properties implied in the hypothesis. For instance, the simulator accurately mapped the age of peak performance and the total number of major injuries to corresponding simulated values that were highly aligned with real-world data. The correlation coefficients between simulated and actual data were 0.625 for age of peak performance and 0.581 for total number of major injuries, suggesting that the LLM effectively captured the underlying relationships specified in the hypothesis.

To assess the accuracy of the simulated data, we also calculated the simulation error gap, which quantifies the discrepancy between the simulated data and actual data. The error gap was found to be 1.325 MAE, indicating that the LLM's output was relatively close to the actual data, but there was still some room for improvement in accuracy.

Figures 5, B.2, and B.3 further illustrate the performance of the simulator. Figure 5a and Figure 5b show scatter plots comparing simulated and actual values for the key properties (age of peak performance and total amount of injuries). The alignment between the two datasets is strong, especially for age. Figures B.2 and B.3 present line plots comparing simulated and actual values over a range of data points, with both plots showing that the LLM's simulated values are closely aligned with actual data.

Impact of Self-Correction. The introduction of self-correction (Madaan et al. 2023) into the simulation pipeline led to measurable improvements in the simulator's performance. Specifically, the correlation coefficients for the two key variables—age of peak performance and total number of major injuries—were higher when self-correction was applied. Without self-correction, the correlation for age was 0.570, and for injuries, it was 0.557. However, with self-correction, these correlations improved to 0.625 for age and 0.581 for injuries, demonstrating the effectiveness of self-correction in refining the LLM's output.

Additionally, the simulation error gap modestly decreased with the application of self-correction. Without self-correction, the error gap was 1.791 MAE, but with self-correction, it was reduced to 1.325 MAE. While this reduction highlights some benefit, the overall improvement is
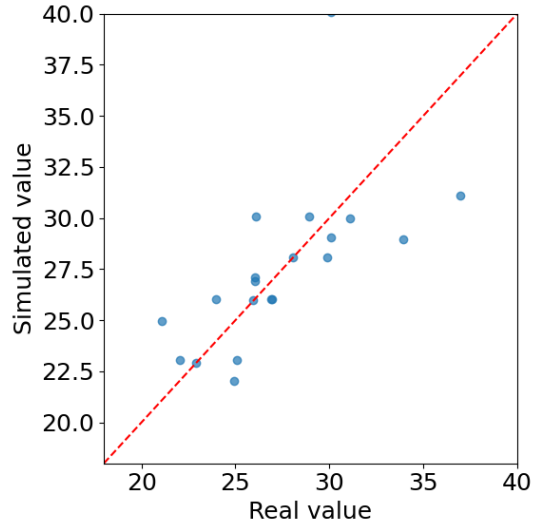
(a) Scatter plot for peak performance age.
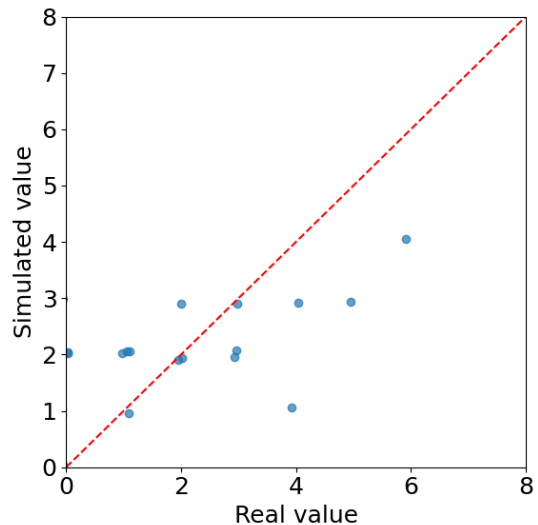
(b) Scatter plot for total major injuries.

Figure 5: Scatter plots comparing simulated and real values for peak performance age and total major injuries. Dashed red line indicates perfect correspondence between real and simulated values.

relatively small. Moreover, as illustrated in Figure B.1, the error bars for the generalization errors—both with and without self-correction—overlap significantly. This limited improvement is likely due to the inadequacy of the real data, both in terms of the properties and sample size. Specifically, the properties used in the simulations may not adequately explain the labels, thereby constraining the potential impact of self-correction. This aligns with the observation that, despite larger improvements in the correlations between simulated and real properties, these gains did not translate into a correspondingly significant reduction in the Simulation Error Gap.

LLMs to generate lists of entities   To further evaluate the ability of LLMs to generate entities themselves, we performed an additional experiment. We prompted the LLM to generate two lists: one of 20 well-known soccer players and another of 20 less-known soccer players. For validation, we randomly selected 20 pairs of players (one from each list) and compared their relative popularity using Google Trends scores as a proxy for notoriety. The results clearly showed that the LLM was capable of differentiating between well-known and less-known soccer players, with significantly higher trend scores for the well-known list ($p < 0.001$). This finding highlights the LLM's ability not only to simulate data but also to generate entities that align with specific qualitative criteria.

In summary, the combination of LLMs for hypothesis mapping, entity generation, and dataset simulation provides a viable framework for using AI to generate reasonably accurate dataset prototypes for hypothesis testing and exploration.

6   Discussion and Conclusion

The experiments in this paper highlight the potential for LLMs to translate high-level descriptions of hypotheses into approximate datasets, ones that can be used to quickly iterate towards interesting latent patterns in existing data. The hope is to empower experimenters to more easily sift through the space of hypotheses by lowering the cost of gathering a dataset by hand. In practice, after discovering an interesting hypothesis, the experimenter will still likely need to either curate a grounded dataset, or perform a real-world experiment, to generate a scientifically validated result.

This kind of method of course has its limitations, as it depends on the estimation abilities of LLMs, which will vary with how well the LLMs' dataset covers the entities as properties of interest, as well as the overall capabilities of the LLM itself. One interesting phenomenon to note is that in the Countries domain, simulating data and then analyzing the relationships among that data performed better than asking the LLM directly to estimate relationships among variables (without simulating the data). In other words, while the information about the variables was latent within the LLM (as it could be simulated), externalizing that information to run outside analysis upon it, yielded further insights. Such improvement relates to the general idea of LLMs iterating upon their own outputs as a way of generating further useful synthetic data.

While the approach here directly queries an LLM, another interesting direction is to employ a more agentic pipeline to actively construct a grounded dataset. That is, LLMs that can browse the web and write code could piece together existing datasets, or attempt to actively ground each data point in reliable sources (e.g. similar to how Perplexity was used to approximate ground truth in the final domain). Such an approach, if it worked well, might present another point in the trade-off between (1) cost and speed, and (2) dataset fidelity: e.g. it would gain fidelity but require more complex chaining of LLM calls.

More broadly, a grander ambition is to create an open-ended system (Stanley, Lehman, and Soros 2017) that could continually discover new, interesting patterns in data. The second set of experiments represents a step in this direction, where the experimenter supplies the high-level hypothesis, which is then translated into the rows and columns of a dataset, which is then simulated. But this could be taken further, where a user instead supplies a more broad question of interest, e.g. "What are interesting patterns of human behavior that can be discerned from biographies of historical figures," and the system itself continually searches for unexpected and interesting patterns by simulating and analyzing datasets. This is related to other directions that attempt to apply LLMs towards open-ended creativity (Lu et al. 2024a; Lehman et al. 2023).

While the approach here works with simulating specific real-world entities (like countries, athletes, and animals), it is also interesting to consider automated creation of datasets that relate to simulations of people through LLMs (Argyle et al. 2023; Aher, Arriaga, and Kalai 2023). Indeed, the work here started with that direction (to explore hypotheses related to whether people with different e.g. OCEAN personality scores would benefit from different leisure activities). There are interesting technical challenges to consider, such as modeling *distributions* of people and their responses (e.g. the distribution of people with a high openness score, and the distribution of their favorite activities), rather than discrete properties of singular entities as in this paper. Such research is an interesting direction of future work that can build off the foundation established in this paper.

Finally, it is interesting to consider the possibilities for novel kinds of ML algorithms opened up by the ability to simulate new features and datasets on the fly. That is, classic tabular learning algorithms (like decision trees) are typically applied to fixed datasets; yet, LLMs open up the possibility of dynamically expanding the feature set as learning progresses. Future work will explore extensions of decision trees that start from a minimal dataset (perhaps only consisting of entities and the dependent variable), and through human-computer interaction, gradually build the dataset as the learning algorithm proceeds; the decision tree algorithm itself becomes more open-ended in its unfolding.

In conclusion, this paper described the potential of using LLMs to simulate datasets about real-world entities, in service of accelerating the exploration of hypotheses about them. Overall, this research points towards the possibility of fully automated systems for automated discovery of knowledge aggregated from the vast cultural output of humans:

What exciting patterns (about us, and about the world) lie waiting for us to distill from the ever-growing ocean of civilization-scale data?

References

Aher, G.; Arriaga, R. I.; and Kalai, A. T. 2023. Using Large Language Models to Simulate Multiple Humans and Replicate Human Subject Studies. arXiv:2208.10264.

Argyle, L. P.; Busby, E. C.; Fulda, N.; Gubler, J. R.; Rytting, C.; and Wingate, D. 2023. Out of One, Many: Using Language Models to Simulate Human Samples. *Political Analysis*, 31(3): 337–351.

Boullier, M.; and Blair, M. 2018. Adverse childhood experiences. *Paediatrics and Child Health*, 28(3): 132–137.

Cohen, R.; Geva, M.; Berant, J.; and Globerson, A. 2023. Crawling the Internal Knowledge-Base of Language Models. arXiv:2301.12810.

Forsyth, R. 1990. Zoo. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5R59V.

Hu, Z.; Feng, Y.; Luu, A. T.; Hooi, B.; and Lipani, A. 2023. Unlocking the Potential of User Feedback: Leveraging Large Language Model as User Simulators to Enhance Dialogue System. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23. ACM.

Lehman, J.; Gordon, J.; Jain, S.; Ndousse, K.; Yeh, C.; and Stanley, K. O. 2023. Evolution through large models. In *Handbook of Evolutionary Machine Learning*, 331–366. Springer.

Liu, H.; Zhou, Y.; Li, M.; Yuan, C.; and Tan, C. 2024. Literature Meets Data: A Synergistic Approach to Hypothesis Generation. arXiv:2410.17309.

Lu, C.; Lu, C.; Lange, R. T.; Foerster, J.; Clune, J.; and Ha, D. 2024a. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*.

Lu, Y.; Shen, M.; Wang, H.; Wang, X.; van Rechem, C.; Fu, T.; and Wei, W. 2024b. Machine Learning for Synthetic Data Generation: A Review. arXiv:2302.04062.

Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; Gupta, S.; Majumder, B. P.; Hermann, K.; Welleck, S.; Yazdanbakhsh, A.; and Clark, P. 2023. Self-Refine: Iterative Refinement with Self-Feedback. arXiv:2303.17651.

Perplexity. 2024. Perplexity AI: Explore Answers and Insights with AI-powered Search.

Ren, R.; Wang, Y.; Qu, Y.; Zhao, W. X.; Liu, J.; Tian, H.; Wu, H.; rong Wen, J.; and Wang, H. 2023. Investigating the Factual Knowledge Boundary of Large Language Models with Retrieval Augmentation. *ArXiv*, abs/2307.11019.

Roccas, S.; Sagiv, L.; Schwartz, S. H.; and Knafo, A. 2002. The big five personality factors and personal values. *Personality and social psychology bulletin*, 28(6): 789–801.

Shuster, K.; Komeili, M.; Adolphs, L.; Roller, S.; Szlam, A.; and Weston, J. 2022. Language Models that Seek for Knowledge: Modular Search & Generation for Dialogue and Prompt Completion. In *Conference on Empirical Methods in Natural Language Processing*.

Sigman, R.; and Lindberg, S. I. 2019. Democracy for all: conceptualizing and measuring egalitarian democracy. *Political Science Research and Methods*, 7(3): 595–612.

Stanley, K. O.; Lehman, J.; and Soros, L. 2017. Openendedness: The last grand challenge you've never heard of. *While open-endedness could be a force for discovering intelligence, it could also be a component of AI itself.*

The World Bank. 2024. World Development Indicators. Data filtered to include only 2022 values. Accessed: 2024-08-04.

Tong, S.; Mao, K.; Huang, Z.; Zhao, Y.; and Peng, K. 2024. Automating psychological hypothesis generation with AI: when large language models meet causal graph. *Humanities and Social Sciences Communications*, 11(1).

V-Dem. 2022. Egalitarian Democracy Index – (best estimate, aggregate: average). Retrieved August 04, 2024. Processed by Our World in Data. Original data: V-Dem Country-Year (Full + Others) v14.

Xiong, G.; Xie, E.; Shariatmadari, A. H.; Guo, S.; Bekiranov, S.; and Zhang, A. 2024. Improving Scientific Hypothesis Generation with Knowledge Grounded Large Language Models. arXiv:2411.02382.

Zhou, Y.; Liu, H.; Srivastava, T.; Mei, H.; and Tan, C. 2024. Hypothesis Generation with Large Language Models. arXiv:2404.04326.

A Countries Domain

A.1 Pre-processing steps

1. Filtering feature database for year 2022

2. Remove features that contain 'Standard Error'. For example, 'Rule of Law: Standard Error' seems a bit too unclear what it means

3. Filter for common countries (egalitarian index and demographic features come from different data sources)

4. Remove demographic features that are not present in all countries

5. Randomly sample N countries and K features from that (N=50, K=10)

A.2 List of countries

```
 1  Eswatini
 2  Mongolia
 3  Trinidad and Tobago
 4  Madagascar
 5  Estonia
 6  Mauritania
 7  Germany
 8  Guinea-Bissau
 9  Ethiopia
10  Canada
11  Kazakhstan
12  Colombia
13  Eritrea
14  Somalia
15  Haiti
16  Brazil
17  Paraguay
18  Mali
19  Georgia
20  Sweden
21  Czechia
22  Myanmar
23  Guyana
24  Cyprus
25  El Salvador
26  Indonesia
27  Montenegro
28  Bolivia
29  Kenya
30  New Zealand
31  Dominican Republic
32  Sudan
33  Tanzania
34  Bahrain
35  Solomon Islands
36  Thailand
37  Romania
38  Mauritius
39  Peru
40  Morocco
41  India
42  Zambia
43  Philippines
44  Togo
45  Djibouti
46  Barbados
47  Zimbabwe
48  Central African Republic
49  Portugal
50  Malawi
51  Chile
52  Sao Tome and Principe
53  Gabon
54  Switzerland
55  Jamaica
56  Sierra Leone
57  Lesotho
58  Nicaragua
59  Malta
60  Honduras
61  Norway
62  Senegal
63  Afghanistan
64  Lebanon
65  Mexico
66  Singapore
67  Niger
68  Iraq
69  United Kingdom
70  Papua New Guinea
71  Saudi Arabia
72  Belarus
73  Seychelles
74  Ireland
75  Fiji
76  Pakistan
77  Uganda
78  France
79  Burundi
80  Bosnia and Herzegovina
81  Maldives
82  Benin
83  Vanuatu
84  Liberia
85  Qatar
86  Uzbekistan
87  Kuwait
88  South Africa
89  Finland
90  Libya
91  Austria
92  Chad
93  Oman
94  United Arab Emirates
95  Namibia
96  Belgium
97  Guatemala
98  Kosovo
99  Ecuador
100 Slovenia
101 Poland
102 Bhutan
103 Turkmenistan
104 Burkina Faso
105 Cuba
106 Cambodia
107 Moldova
108 Spain
109 United States
110 Cote d'Ivoire
111 Serbia
112 Croatia
113 South Sudan
```

114 Netherlands
115 Guinea
116 Latvia
117 Japan
118 Algeria
119 Albania
120 Hungary
121 Luxembourg
122 Uruguay
123 Armenia
124 Greece
125 Bulgaria
126 Suriname
127 Nigeria
128 Angola
129 Jordan
130 Azerbaijan
131 China
132 Ghana
133 Denmark
134 Comoros
135 Malaysia
136 Italy
137 Lithuania
138 North Macedonia
139 Tajikistan
140 Mozambique
141 Panama
142 Ukraine
143 Israel
144 Sri Lanka
145 Australia
146 Equatorial Guinea
147 Bangladesh
148 Tunisia
149 Cameroon
150 Iceland
151 Argentina
152 Rwanda
153 Nepal
154 Costa Rica
155 Botswana

A.3 List of features

1 Population ages 00-04, male (% of male population)
2 Population, male (% of total population)
3 Population ages 65 and above, female (% of female population)
4 Regulatory Quality: Percentile Rank
5 Population ages 40-44, male (% of male population)
6 Regulatory Quality: Estimate
7 Population ages 0-14, female
8 Population ages 60-64, female (% of female population)
9 Survival to age 65, female (% of cohort)
10 Population ages 15-64, total
11 Rule of Law: Estimate
12 Government Effectiveness: Percentile Rank
13 Population ages 15-64, male (% of male population)
14 Population ages 65-69, male (% of male population)
15 Population ages 05-09, female (% of female population)
16 Birth rate, crude (per 1,000 people)
17 Mortality rate, under-5 (per 1,000 live births)
18 Population ages 20-24, male (% of male population)
19 Population ages 65 and above, total
20 Adolescent fertility rate (births per 1,000 women ages 15-19)
21 Population ages 10-14, female (% of female population)
22 Sex ratio at birth (male births per female births)
23 Life expectancy at birth, male (years)
24 Regulatory Quality: Number of Sources
25 Number of deaths ages 20-24 years
26 Number of deaths ages 10-14 years
27 Population ages 70-74, male (% of male population)
28 Population ages 40-44, female (% of female population)
29 Probability of dying among adolescents ages 15-19 years (per 1,000)
30 Population ages 0-14, female (% of female population)
31 Voice and Accountability: Percentile Rank
32 Mortality rate, neonatal (per 1,000 live births)
33 Population ages 50-54, male (% of male population)
34 Population ages 75-79, male (% of male population)
35 Population ages 55-59, female (% of female population)
36 Population ages 0-14, total
37 Population ages 45-49, male (% of male population)
38 Population ages 50-54, female (% of female population)
39 Population ages 55-59, male (% of male population)
40 Voice and Accountability: Percentile Rank, Upper Bound of 90% Confidence Interval
41 Mortality rate, under-5, female (per 1,000 live births)
42 Population ages 0-14, male (% of male population)
43 Population ages 65 and above, male (% of male population)
44 Mortality rate, infant (per 1,000 live births)
45 Population ages 45-49, female (% of female population)
46 Population ages 30-34, male (% of male population)
47 Population ages 70-74, female (% of female population)
48 Regulatory Quality: Percentile Rank, Upper Bound of 90% Confidence Interval
49 Rule of Law: Number of Sources

A.4 Prompt Examples

- **Direct style, Structured format**
  ```
  1  sys_prompt = You will be asked to make
         your best guess about the value a
         country had for a particular
         feature in 2022. Respond in the
         following json format: {feature:
         value}. Where feature is the
         characteristic about the country,
         and value is your numeric guess.
         If you don't know, make your best
         guess
  2  user_prompt = Country: Namibia,
         Population ages 45-49, male (% of
         male population):
  ```

- **Direct style, Descriptive format**
  ```
  1  sys_prompt = You will be asked to make
         your best guess about the value a
         country had for a particular
         feature in 2022. Respond in the
         following json format: {feature:
         value}. Where feature is the
         characteristic about the country,
         and value is your numeric guess.
         If you don't know, make your best
         guess
  2  user_prompt = What was the value of
         the 'Population ages 45-49, male
         (% of male population)' in Namibia
          for 2022?
  ```

- **Report style, Structured format**
  ```
  1  sys_prompt = You are an expert
         historian. Please complete the
         following document.
  2  user_prompt = This document contains
         demographics and other variables
         of countries in 2022. It is
         documented by an expert historian
         and socio-political expert. \n
         Country: Namibia, Population ages
         45-49, male (% of male population)
         :
  ```

- **Report style, Descriptive format**
  ```
  1  sys_prompt = You are an expert
         historian. Please complete the
         following document.
  2  user_prompt = This document contains
         demographics and other variables
         of countries in 2022. It is
         documented by an expert historian
         and socio-political expert. \n  I
         conclude that the Population ages
         45-49, male (% of male population)
          in Namibia for 2022 was
  ```
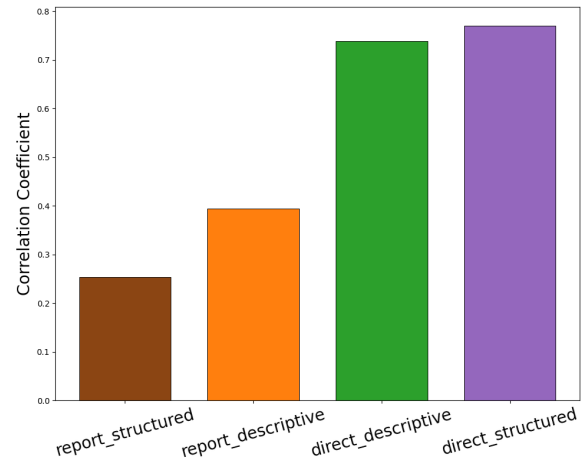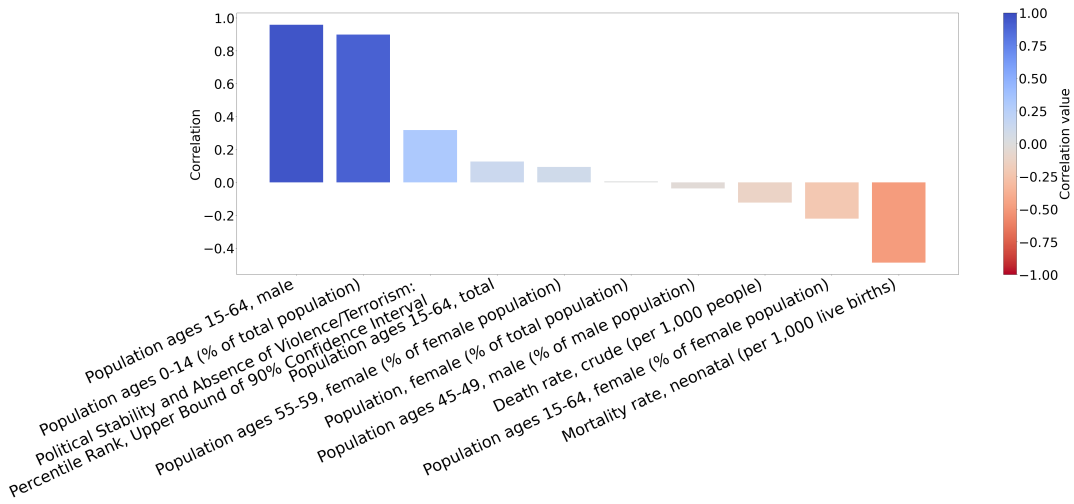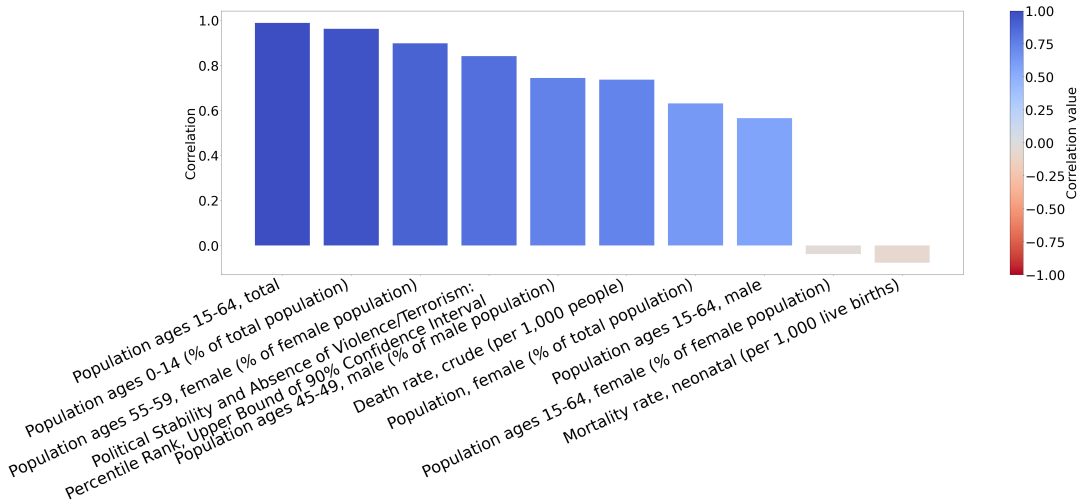
A.5 Extra Figures

Figure A.1: Correlation coefficients by prompting strategy for Countries Domain. "Report" is considerably worse of a prompting style than "Direct". "Direct-structured" was found to be the best performing prompting strategy.

(a) LLaMA-3-8B correlations between and real and simulated features.

(b) LLaMA-3-70B correlations between and real and simulated features.
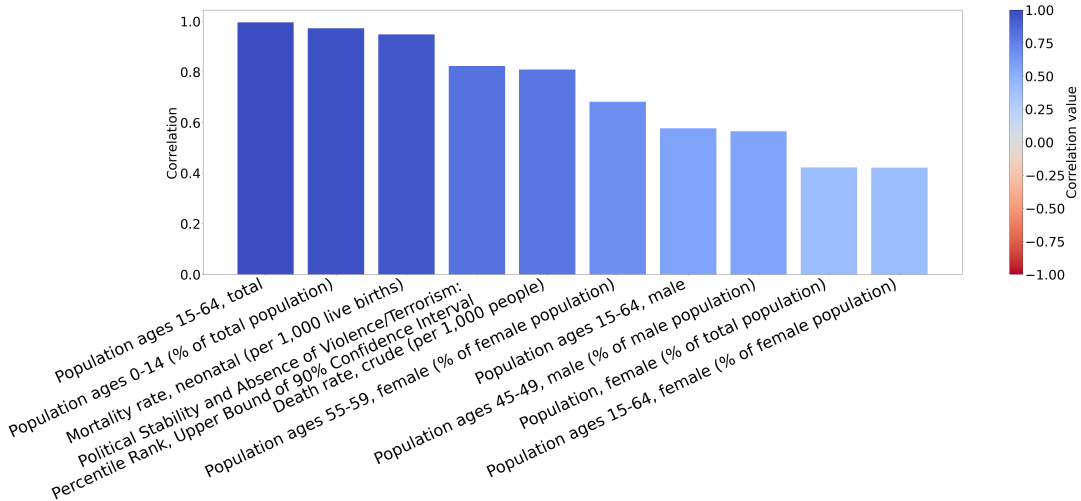
(c) GPT-4o-mini correlations between and real and simulated features.

Figure A.2: Comparison of correlations between real and simulated features for the Countries Domain in LLaMA-3-8B, LLaMA-3-70B, and GPT-4o-mini. Larger models exhibit stronger correlations, indicating improved simulation quality.
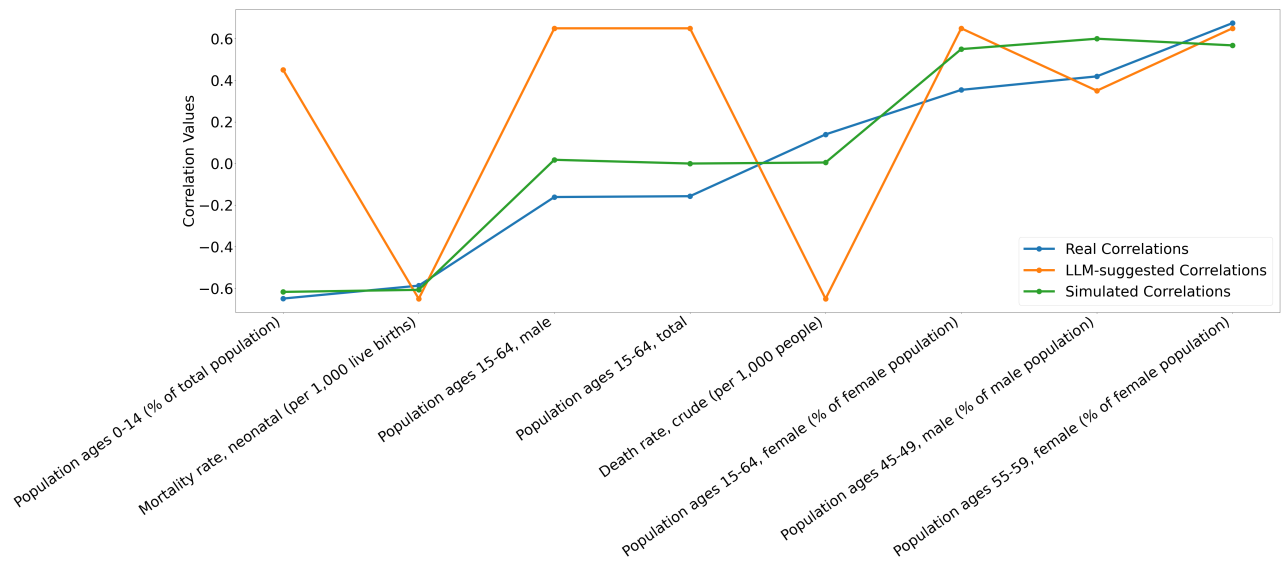
Figure A.3: Comparison of real, LLM-suggested and simulated correlations for Countries domain. The LLM-suggested method consistently underperforms compared to our simulation approach in accurately capturing complex relationships between demographic variables.
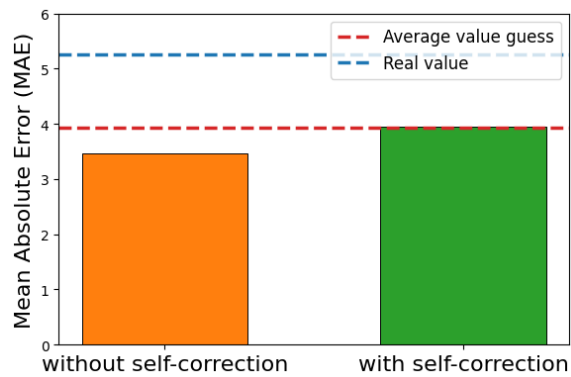
B Athletes Domain

Figure B.1: Comparison of simulation mean absolute error (MAE) with and without self-correction. The dashed blue line indicates the MAE achieved using real data, while the dashed red line shows the baseline error from a dummy model predicting the mean value.
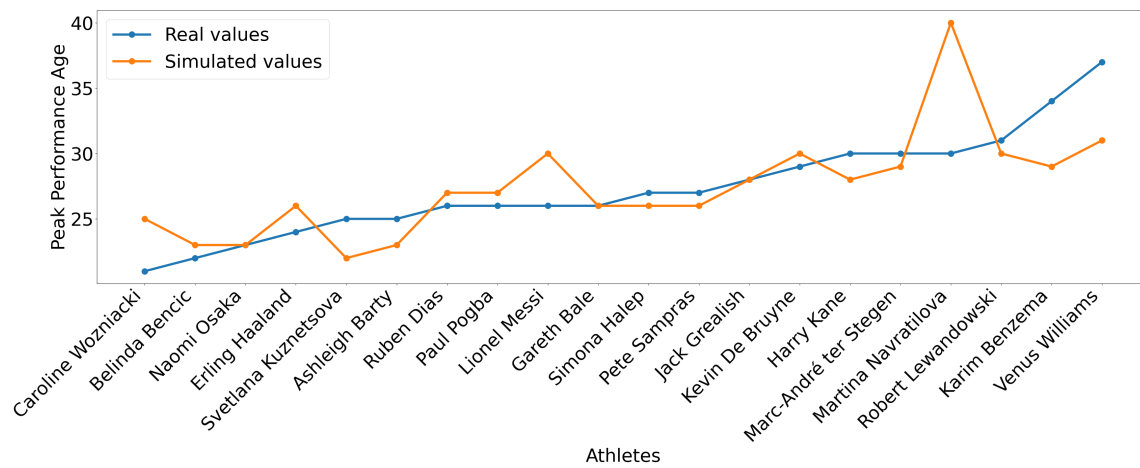
Figure B.2: Line plot showing the real and simulated values for peak performance age.
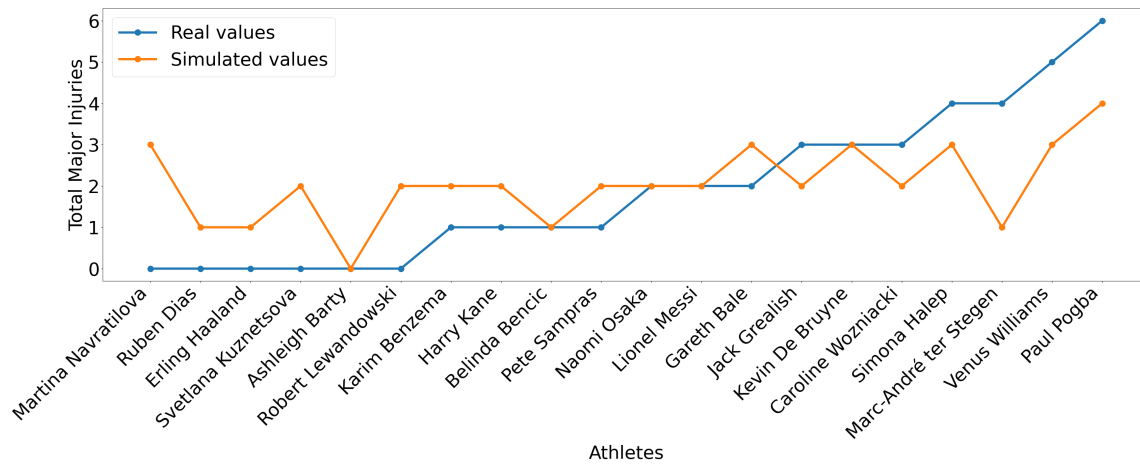

Figure B.3: Line plot showing the real and simulated values for total major injuries.