# MULTI-AGENT CAUSAL DISCOVERY USING LARGE LANGUAGE MODELS

Anonymous authors

004

010

011

012

013

014

015

016

017

018

019

021

023

024

Paper under double-blind review

## ABSTRACT

Large Language Models (LLMs) have demonstrated significant potential in causal discovery tasks by utilizing their vast expert knowledge from extensive text corpora. However, the multi-agent capabilities of LLMs in causal discovery remain underexplored. This paper introduces a general framework to investigate this potential. The first is the Meta Agents Model, which relies exclusively on reasoning and discussions among LLM agents to conduct causal discovery. The second is the Coding Agents Model, which leverages the agents' ability to plan, write, and execute code, utilizing advanced statistical libraries for causal discovery. The third is the Hybrid Model, which integrates both the Meta Agents Model and Coding Agents Model approaches, combining the statistical analysis and reasoning skills of multiple agents. Our proposed framework shows promising results by effectively utilizing LLMs' expert knowledge, reasoning capabilities, multi-agent potential of LLMs, we aim to establish a foundation for further research in utilizing LLMs multi-agent for solving causal-related problems.

025 026 1 INTRODUCTION

Understanding causal relationships is crucial across scientific fields. While statistical causal infer ence is widely used, it heavily relies on assumed causal graphs. To address this limitation, data driven methods have evolved, leading to statistical causal discovery (SCD) approaches and the cre ation of datasets for evaluation. Despite advancements in SCD algorithms, data-driven causal graphs
 without domain knowledge can be inaccurate. This inaccuracy is often due to a mismatch between
 SCD algorithm assumptions and real-world phenomena (Reisach et al. (2021)). Incorporating expert
 knowledge can mitigate this issue, but it is costly.

The advent of Large Language Models (LLMs), trained on vast amounts of data, has enabled them to acquire extensive knowledge, from common sense to specific domains such as math and science. Recent studies suggest that complex behaviors, such as writing code, generating long stories, and even reasoning capabilities, can emerge from large-scale training (Wei et al. (2023); Rozière et al. (2024); Zhao et al. (2023b); Yao et al. (2023a)). LLMs present a promising alternative for obtaining expert knowledge more accessible and affordable. Recent research (Kıcıman et al. (2023);Choi et al. (2022); Long et al. (2024); Chen et al. (2024b)) has attempted to leverage these capabilities for causal discovery based on metadata and knowledge-based reasoning, akin to human domain experts.

042 However, most methods have not yet fully utilized the full capacity of LLMs, i.e., LLMs' multi-043 agent approaches. An LLM agent can be seen as an entity with memory, reasoning, and the ability 044 to access external tools or APIs such as a calculator, web search, and code compiler. Agent-based systems have demonstrated significant problem-solving abilities. However, a single-agent-based system sometimes still suffers from hallucinations despite having self-reflection capabilities (Li et al. 046 (2023); Shinn et al. (2023), Madaan et al. (2023)). Inspired by the Society of Mind concept (Minsky 047 (1988)), LLM multi-agent system discussion frameworks like MAD Liang et al. (2023), ReConcile 048 (Chen et al. (2024a), and CMD Wang et al. (2024) have been proposed to address these issues. These LLM multi-agent systems not only achieve impressive results but also enable less capable models to perform on par with superior ones. 051

Despite the potential benefits, few studies have explored leveraging LLM multi-agent system capabilities in causal discovery. To address this gap, we propose a novel framework called Multi-Agent for Causality (MAC), which comprises three different models.

054

056

058

067 068 069

070

М	ethod / Approach	LLMs for metadata	Statistical Approach for structured data	Agentic Ability	Multi-Agent Workflow	Introduced By
Pai	iirwise Causal Discov- y	$\checkmark$	×	×	×	Kıcıman et al. (2023); Zečević et al. (2023)
Va eng	arious of prompt- gineering strategies	$\checkmark$	×	×	×	Chen et al. (2024b)
Efi tio Dia	ficiently asking ques- on for Causal Graph iscovery Using LLMs	$\checkmark$	$\checkmark$	×	×	Jiralerspong et al. (2024)
Co tra od:	ombining LLMs with aditional causal meth- ls.	$\checkmark$	$\checkmark$	×	×	Vashishtha et al. (2023); Takayama et al. (2024)
Mu Di	ulti-Agent Causal iscover (MAC)	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	Our approach

Table 1: Comparison of Approaches for Using LLMs in Causal Discovery

The first is called **Meta Agents Model**, which includes a *Meta-Debate Module* with two debater agents and one judge agent. These agents are designed for causal discovery problems and utilize the nature of debating to find causal relationships among variables through multiple rounds of discussion. The second is called **Coding Agents Model**, which includes a *Debate-Coding Module* with four agents operating in two phases. This group leverages both the debating abilities of the agents and statistical causal algorithms. The third one is **Hybrid Model**, which hybridizes the statistical causal discovery algorithms with the reasoning skills of multiple agents to construct the causal graph.

In this research, we experiment with various LLM models across small, medium, and large scales, paired with different MAC models. We conduct an in-depth analysis of performance for each MAC model and LLM, token consumption, and identify their usage patterns and limitations. Additionally, we propose alternative solutions to address the computational costs associated with LLMs.

Our proposed framework shows promising results by effectively utilizing LLMs' expert knowledge, reasoning capabilities, multi-agent cooperation, and statistical causal methods. As far as we know, this paper is first work to explore LLMs' multi-agent abilities in a causal context. We hope that our work will lay the foundation for further research in utilizing LLM multi-agent systems for solving causal-related problems.

- <sup>8</sup> 2 Related Works
- 090 2.1 LLMS' AGENTIC WORKFLOW

A general LLM agent framework consists of core components: user request, agent/brain, planning, 092 memory, and tools. The agent/brain acts as the main coordinator, activated by a prompt template. It can be profiled with specific details to define its role, using handcrafted, LLM-generated, or data-094 driven strategies. Planning employs techniques like Chain of Thought and Tree of Thoughts, and for complex tasks, feedback mechanisms like ReAct Yao et al. (2023b) and Reflexion Shinn et al. (2023) 096 refine plans based on past actions and observations. Memory stores the agent's logs, with short-097 term memory for the current context and long-term memory for past behaviors. Hybrid memory 098 combines both to enhance reasoning and experience accumulation. Tools enable interaction with 099 external environments, such as APIs and code interpreters. Frameworks like MRKL Karpas et al. (2022), Toolformer Schick et al. (2023), Function Calling OpenAI (2024), and HuggingGPT Shen 100 et al. (2023) integrate tools to solve tasks effectively. 101

However, for more complex problems where a single LLM agent may struggle, LLM-MA (multi-agent) systems excel. Current LLM-MA systems primarily employ three communication paradigms:
Cooperative, Competitive, and Debating. In the Cooperative paradigm, agents collaborate towards a shared goal, typically exchanging information to enhance a collective solution Qian et al. (2023);
Chen et al. (2024c). In the Competitive paradigm, agents work towards their own goals, which might conflict with those of other agents Zhao et al. (2023a). The Debating paradigm involves agents engaging in argumentative interactions, where they present and defend their viewpoints or



Figure 1: Meta-Debate Module

solutions while critiquing those of others. This approach is ideal for reaching a consensus or a more refined solution (Li et al. (2023); Liang et al. (2023); Xiong et al. (2023)). In this work, the debating paradigm will be implemented, as the nature of causal discovery problems requires diverse and potentially conflicting opinions to approach the truth.

2.2 STATISTICAL AND LLM-BASED CAUSAL METHODS

127 Traditional methods of statistical causal inference often depend heavily on assumed causal graphs to identify and measure causal impacts. To overcome this limitation, data-driven algorithmic ap-128 proaches have been developed into statistical causal discovery (SCD) methods, encompassing both 129 non-parametric (e.g., Spirtes et al. (2000); ; ; Yuan & Malone (2013); Huang et al. (2018); Xie et al. 130 (2020)) and semi-parametric (e.g., Shimizu et al. (2006); Hoyer et al. (2009); Shimizu et al. (2011)) 131 techniques. Many SCD algorithms can be systematically augmented with background knowledge 132 and have accessible software packages. For example, the non-parametric and constraint-based Peter-133 Clerk (PC) algorithm (Spirtes et al. (2000)) in "causal-learn" integrates background knowledge of 134 mandatory or forbidden directed edges. "Causal-learn" also includes the Exact Search algorithm 135 (Shimizu et al. (2011); Yuan & Malone (2013) ), a non-parametric and score-based SCD method 136 that can incorporate background knowledge in the form of a super structure matrix of forbidden 137 directed edges. Furthermore, the semi-parametric DirectLiNGAM (Shimizu et al. (2011)) algorithm can use prior knowledge of causal order (Inazumi et al. (2010)) in the "LiNGAM" project (Ikeuchi 138 et al. (2023)). 139

140 In the context of knowledge-driven approaches using large language models (LLMs), applying 141 LLMs for causal inference is relatively new. There have been a few significant efforts to use LLMs 142 for causal inference among variables by merely prompting with the variable names, without going 143 through the traditional SCD process with benchmark datasets (Kiciman et al. (2023); Zečević et al. (2023)). Jiralerspong et al. (2024) even uses a breadth-first search (BFS) approach which allows it to 144 use only a linear number of queries to have a higher efficiency. Additionally, Chen et al. (2024b) pro-145 poses 9 prompting techniques comprise ICL, 0-shot CoT (e.g. "let's think step by step"), adversarial 146 prompt, manual CoT, and explicit function (e.g using encouraging language in prompts). Moreover, 147 the works of Ban et al. (2023) and Vashishtha et al. (2023) introduce interesting approach by inte-148 grate LLMs into traditional data-driven approaches. However, most of above-mentioned works have 149 not investigate the LLM-agentic work-flows for causal graph discovery ??, which indeed requires 150 heavy investigation on various models and graphs' scales. 151

151 152

119 120

121

122

123

124 125

126

## 3 MAC: MULTI-AGENT CAUSALITY FRAMEWORK

- 153 154 3.1 Multi-Agent Causality Modules
- 155 3.1.1 META-DEBATE MODULE

The Meta-Debate Module is an advanced system comprising three intelligent agents: two causal debaters (an affirmative one and a negative one) and one causal judge. This structure emulates the dynamic and rigorous nature of human debate, specifically within the realm of causal discovery. The design of this debating module is meticulously crafted to ensure a thorough examination of critical elements in causal discovery, such as understanding the temporal order necessary to establish cause-and-effect relationships and identifying potential confounding variables that could distort perceived relationships between primary variables. Each side engages in active disagreement by



Figure 2: Prompting design of each agent in MAC, details in the Appendix A.3

presenting different opinions and viewpoints, fostering a comprehensive and robust debate. Ad ditionally, each agent within Meta-Debate Module utilizes the ReAct prompting technique. This
 technique integrates the ability to dynamically formulate, modify, and refine action plans based on
 new information or insights gained during the debate. This integration allows the agents to engage
 in more sophisticated and adaptive reasoning processes, closely mimicking human-like debate and
 decision-making (Figure 2).

The debating process begins with a meta-question. Initially, the causal affirmative side presents its answer along with the supporting rationale. Subsequently, the causal negative side offers diverse or conflicting perspectives by providing alternative viewpoints on the same question. The causal judge then evaluates the responses from both sides, determining either a winner or identifying the need for additional clarification. If further information is required, the causal judge poses specific follow-up questions, prompting the debaters to clarify and elaborate on their initial propositions. The causal judge reaches a final verdict once all relevant information has been thoroughly examined.

203 For meta-questions, questions might address aspects such as the appropriate algorithm for a partic-204 ular problem, the causal relationship between two variables, or a step-by-step approach for solving 205 a causal problem, as illustrated in Figure 1. The affirmative side initiates the debating process based 206 on the constructed input. The final output is derived from the causal judge's decision, reflecting 207 a comprehensive evaluation of the arguments presented by both sides. This rigorous process ensures that the outcome is well-informed and considers multiple perspectives, thereby enhancing the 208 robustness of causal discovery. A detailed visualization of this module can be found in Appendix 209 A.2.1. 210

211

189

### 212 3.1.2 DEBATE-CODING MODULE

213

The Debate-Coding Module leverages statistical algorithms to achieve precise causal discovery
 through a structured two-phase process. This group consists of four agents, divided into two phases:
 debating the algorithm and executing the algorithm.

216

224

225

231

233

242

243 244

245

246 247

248

269

217 Plan Debate Battle Input for Causal Coding Agent Causal Coding Agent Execute 218 Prompting Output: plan Negative give a more compre Input 2: observational data Output: a causal grap 219 mpare answer of Affirmative side. So I atrix representatio decide to propose negative side. Here is Step-by-Step Plan for Using Gradient 220 ng Machines (GBM) for Reg Step 1: Understand the Problem Step 2: Data Preparation Step 3: Data Preparation Step 4: Model Selection 221 obs 2 4 2 222 ning d tep 5: Model Training 223

Figure 3: Debate-Coding Module

226 In the initial phase, three agents engage in a debate format, similar to the workflow of Meta-Debate 227 module at Section 3.1.1. However, the difference is that the affirmative and negative agent are 228 pre-prompts the information with 3-5 statistical causal algorithms. Additionally, the meta-question posed to these agents is specifically curated to determine which algorithm should be used given the 229 metadata, which includes the description and structure of the data. After the debate, the output is 230 the most suitable algorithm for the particular dataset and question. Compared to the Meta-Debate Module, an additional step in this process is that the agent participating in the debate (whether 232 affirmative or negative) will provide a step-by-step plan for implementing the selected algorithm.

In the second phase, the causal coding executor receives the plan from the previous phase along 234 with the observational data. The causal coding executor is responsible for writing, executing, and 235 debugging the code. It pre-prompts the functions and provides parameters within a specific Python 236 library<sup>1</sup> based on the algorithm selected by the debaters in the initial stage. This pre-prompting 237 is crucial because LLMs can call functions from their training dataset, which may be outdated or 238 incorrect, leading to errors and excessive debugging (details of prompting design can be found in the 239 Appendix A.3). After executing the code, the causal coding executor outputs a matrix representation 240 of the causal graph. A detailed visualization of this module can be found in the Appendix A.2.2. 241

3.2IMPLEMENTATIONS OF MAC

In this section, we will elaborate on the detailed implementation of the three models regarding their input and basic workflow.

3.2.1 META AGENTS MODEL

#### 249 Algorithm 1 Algorithm for Meta Agents Model 250 1: **Input:** Data $X = [x_1, ..., x_n]$ 251 2: function META\_DEBATE\_MODULE(query) **Result:** Response to the query about causal relationships 3: 253 4: end function 254 5: **Output:** Graph $G_{ij}$ where i and j are indices in N 255 6: for $i \leftarrow 1$ to |X| do 256 7: for $j \leftarrow 1$ to i - 1 do $G_{ij} \leftarrow \text{Meta_Debate_Module}(\text{``Are there any direct causal from } X[i] \text{ to } X[j]\text{''})$ 8: 257 9: end for 258 10: for $j \leftarrow i + 1$ to |X| do 259 $G_{ii} \leftarrow$ Meta\_Debate\_Module("Are there any direct causal relationship from X[i] to 11: 260 X[j]") 261 12: end for 262 13: end for 14: return Graph 264 265

The algorithm for the Meta Agents Model aims to identify direct causal relationships between vari-266 ables in a dataset. It starts with the input data  $X = [x_1, ..., x_n]$ . The output of the algorithm is a 267 graph  $\mathcal{G}$ , with its edges  $G_{ij}$ , where i and j are indices in the set of variables n 268

<sup>&</sup>lt;sup>1</sup>More functions can be found at https://causal-learn.readthedocs.io/en/latest/ index.html

270 The algorithm proceeds by iterating through each variable i from 1 to the number of variables in X. 271 For each i, it checks all j values less than i (i.e., j ranges from 1 to i-1). It queries the Meta-Debate 272 Module to check if there is a direct causal relationship from X[i] to X[j] and stores the result in 273  $G_{ij}$ . Then, it checks all j values greater than i (i.e., j ranges from i + 1 to the number of labels in 274 X). Those queries are meta-questions that use the Meta-Debate Module function to check if there is a direct causal relationship from X[i] to X[j] and store the result in  $G_{ij}$ . The algorithm concludes 275 by returning the constructed graph  $G_{ij}$ . A detail of the function Meta-Debate Module has been 276 described in section 3.1.1 278 3.2.2 CODING AGENTS MODEL 279 281 Algorithm 2 Algorithm for Coding Agents Model 282 1: Input 1: Meta-question  $Q_X$ 283 2: Input 2: Observational data  $O_X$ 284 3: function META\_DEBATE\_MODULE(query) 286 **Result:** Response to the query and give a step-by-step causal plan 4: 287 5: end function 288 6: **function** CAUSAL\_CODE\_EXECUTOR(plan, data) 289 **Result:** Return the result according to the plan via code execution 7: 8: end function 291 292 9: **Output:** Graph  $G_{ij}$  where *i* and *j* are indices in N 293 10: Plan  $\leftarrow$  Meta\_Debate\_Module( $Q_X$ ) 11: Graph  $\leftarrow$  CAUSAL\_CODE\_EXECUTOR(Plan,  $O_X$ ) 294 12: return Graph 295 296 297 The inputs to the algorithm are a meta-question  $Q_X$  and observational data  $O_X$ , and the output is the 298 construction of a causal graph  $G_{ij}$ . The algorithm for the Coding Agents Model aims to determine 299 causal relationships by first generating a causal analysis plan using a debate format. After yielding 300 the plan, it will then be executed with the observational data using a causal code executor. The detailed implementation of the function causal code executor can also refer to the section 4.4.1 301 302 303 3.2.3 HYBRID MODEL 304 Algorithm 3 Algorithm for Hybrid Group 306 1: Input 1: meta\_question  $Q_X$ 307 2: Input 2: observational data  $O_X$ 308 3: Input 3: Data  $X = [x_1, ..., x_n]$ 4: **Output:** Graph  $G_{ij}$  where *i* and *j* are indices in N 310 311 5: if Coding-Debating Hybrid then 312 ▷ Algorithm 2 initial\_graph  $\leftarrow$  Debate\_Coding\_Module $(Q_X, O_X)$ 6: 313 Graph ← Meta\_Debate\_Module(initial\_graph, X) 7: ▷ Algorithm 1 314 8: **end if** 315 9: if Debating-Coding Hybrid then 316

10:prior\_knowledge  $\leftarrow$  Meta\_Debate\_Module(X) $\triangleright$  Algorithm 111:Graph  $\leftarrow$  Debate\_Coding\_Module(prior\_knowledge,  $Q_X, O_X$ ) $\triangleright$  Algorithm 212:end if

- 319 12. enu n 13: return Graph
- 320 321

317

318

There are two combinations of Hybrid Group: **Coding-Debating Hybrid** and **Debating-Coding Hybrid**. They are fundamentally identical to the Meta Agents Model and Coding Agents Model in terms of their internal architectures, algorithms, and outputs. The difference lies in their inputs. For **Coding-Debating Hybrid**, the initial result will be obtained from the Coding Agents Model of Algorithm 2 given the input of a meta-question and observational data. The graph and the proposed algorithm for achieving the final graph will also be extracted and fed into the Meta Agents Model 1. For example, the proposed algorithm in the Coding Agents Model is PC, and the initial graph  $\hat{G}$ , when both of them were input to the Meta Agents Model, the query would change slightly at line 8 and 11 in the algorithm 1 which is illustrated by the box below. The final return output is a matrix representation of a causal graph.

 $G_{ij} \leftarrow \text{Debating\_Group}("$  From the PC algorithm and analysis, there is  $\{ \text{"no" if } \hat{G}[i][j] == 0 \text{ else ""} \}$  direct causal relationship from X[i] to X[j]. But from your expert and the suggested result above, are there any direct causal relationships from X[i] to X[j]?")

For the **Debating-Coding Hybrid**, the initial graph comes from the Meta Agents Model of algorithm 1. This output is then considered as prior knowledge or background knowledge. It is aggregated with the meta-question, and input to the Coding Agent Model of algorithm 2. It will select a suitable statistical causal discovery algorithm and plan. This plan is then given to the code executor to implement, resulting in a matrix representation of a causal graph.

- 343 4 EXPERIMENT AND RESULTS
- 344 345 4.1 EXPERIMENTAL SETUP AND METRICS

346 We use OpenAI API including GPT-3.5, GPT-40, and GPT-40 mini for our experiment, Grog API 347 for Llama-8.1-70b, Llama-8.1-8b, and Gemini-9B for most of our experiments with the temperature 348 set at 0. We experiment on three different datasets that adopted from the work of Takayama et al. 349 (2024) including Auto MPG data (Quinlan (1993)), DWD climate data (Mooij et al. (2016)), and Sachs protein data (Sachs et al. (2005)). For the evaluation metrics, we assess the adjacency matrix 350 obtained from LLMs or a code executor using structural hamming distance (SHD) and Normal-351 ized Hamming Distance (NHD) as described by Takayama et al. (2024) and Kıcıman et al. (2023) 352 respectively. 353

354 355

331

332

333

334

335

336 337

338

339

340

341

342

4.2 PERFORMANCE OF MAC

The results of our experiments indicate that the performance is highly dependent on the complexity of the dataset. We rank and summarize the performance of each model based on the empirical results: (1) Coding Agent, this model exhibits strong performance, particularly when larger models. (2) Coding-Debating Hybrid / Debating-Coding Hybrid, these models provide balanced performance but do not excel in handling highly complex datasets. They are particularly effective in moderate complexity settings. (3) Causal Agent Debate, this model performs well for simpler datasets but faces challenges with more complex ones.

363 Auto MPG Dataset: In the Auto MPG dataset, which has a 5x5 matrix graph's structure, the Causal 364 Agent Debate method performed strongly, particularly with GPT-40 mini, achieving the lowest 365 SHD of 4 and a NHD of 0.16. The Coding Agent method also showed good results, with GPT-366 3.5 achieving a similarly low SHD of 4 but a higher NHD of 0.48. The Coding-Debating Hybrid 367 method demonstrated balanced performance, with both GPT-40 mini and Llama 3.1 8b achieving 368 an SHD of 5, while Llama 3.1 8b had the lowest NHD of 0.2. The traditional methods and other advanced methods show higher SHD values, indicating lower structural accuracy. For example, PC, 369 DirectLiNGAM, and Single-agent zero-shot prompting (GPT-40) all have SHD values of 8, demon-370 strating less accurate structural learning compared to the Coding agents 371

372 DWD Climate Dataset: In the DWD climate dataset, which comprises a 6x6 matrix graph, the
373 Causal Agent Debate method again showed strong performance, with GPT-3.5 achieving the lowast SHD of 5 and NHD of 0.194. The Coding Agent method performed well with GPT-40 mini,
achieving an SHD of 6 and NHD of 0.194. The Coding-Debating Hybrid method stood out, with
Llama-3.1-8b achieving the lowest SHD (5) and NHD (0.138). Similarly, the Hybrid-DebatingCoding Hybrid method also performed well, with GPT-40 achieving an SHD of 5 and NHD of
0.138. The traditional methods such as PC and DirectLiNGAM have higher SHD values (9 and 10,

378	Model	Auto MPG		Climate		Sachs	
379	Model	SHD	NHD	SHD	NHD	SHD	NHD
380		Other m	ethods				
381	PC	8	0.48	9	0.305	24	0.206
382	Exact Search	7	0.44	6	0.194	31	0.33
302	LINGAM	8	0.48	10	0.388	29	0.289
303	PC LLM-KBCI	7	0.44	7	0.222	30	0.314
384	ES LLM-KBCI	7	0.44	7	0.222	31	0.33
385	DirectLiGam LLM-KBCI	7	0.4	9	0.305	29	0.289
386	Single-agent (GPT-40)	8	0.36	11	0.388	18	0.214
387	Single-agent (GPT-3.5)	7	0.28	10	0.361	31	0.363
388	Cau	isal Age	nt Debat	e			
389	GPT-40	5	0.2	9	0.333	35	0.371
200	GPT-40 mini	4	0.16	11	0.416	35	0.338
390	GPT-3.5	5	0.2	5	0.194	21	0.231
391	Llama 3.1 70b	10	0.44	8	0.222	35	0.380
392	Llama 3.1 8b	7	0.28	9	0.277	35	0.338
393	Gemini 2 9b	5	0.2	7	0.222	25	0.223
394		Coding A	Agents				
395	GPT-40	8	0.48	9	0.388	30	0.322
396	GPT-40 mini	6	0.6	6	0.194	30	0.322
390	GPT-3.5	4	0.48	9	0.305	29	0.28
397	Llama 3.1 70b	6	0.6	6	0.194	18	0.157
398	Llama 3.1 8b	6	0.6	9	0.388	36	0.396
399	Gemini 2 9b (GPT-4o-mini's plan)	7	0.36	-	-	-	-
400	Hybr	id-Codir	ig-Debat	ing			
401	GPT-40	6	0.28	10	0.305	23	0.247
402	GPT-40 mini	5	0.24	9	0.25	21	0.190
102	GPT-3.5	6	0.32	7	0.25	23	0.198
403	Llama 3.1 70b	6	0.36	6	0.166	33	0.314
404	Llama 3.1 8b	5	0.2	5	0.138	33	0.305
405	Gemini 2 9b	-	-	-	-	-	-
406	Hybr	id-Debat	ing-Cod	ing			
407	GPT-40	5	0.24	5	0.138	29	0.297
408	GPT-40 mini	8	0.48	7	0.25	22	0.190
409	GPT-3.5	8	0.48	7	0.277	28	0.272
410	Llama 3.1 70b	5	0.24	6	0.194	29	0.289
410	Llama 3.1 8b	6	0.4	9	0.388	-	-
411	Gemini 2 9b	-	-	-	-	-	-

412 413

Table 2: Combined results across Auto MPG, Climate, and Sachs Datasets

414 415

respectively) and lower overall performance in other metrics. The Single-agent zero-shot prompting
methods, including GPT-40, show higher SHD values (10 and 11, respectively) and less competitive
performance across the other metrics.

419 Sachs' Protein Dataset : In the Sachs' protein dataset, the Causal Agent Debate method displayed
420 mixed results, with GPT-40 achieving an SHD of 21 and an NHD of 0.231. The Coding Agent
421 method outperformed others, with Llama-3.1-70b achieving the lowest SHD of 18 and an NHD
422 of 0.157. The Coding-Debating Hybrid method demonstrated consistent performance, with GPT-40
423 mini reaching an SHD of 21 and GPT-3.5 achieving the lowest NHD of 0.198. The Debating-Coding
424 Hybrid method also performed well, with GPT-40 mini achieving an SHD of 22 and NHD of 0.190.

425 4.3 PERFORMANCE AMONG LLMS

427 Small size LLMs: Llama 3.1 8b is a strong small model, consistently achieving solid results in hybrid models and competing with larger models in simpler datasets like Auto MPG and Climate.
429 It excels in Hybrid Coding-Debating settings. Gemini 2 9b, while capable, tends to lag behind Llama 3.1 8b, especially since the MAC models require coding, particularly when the prompting token count exceeds 8,000 to 9,000 tokens, which leads to random code generation and prolonged debugging loops. Therefore, we do not include the result of this model in our research.



Figure 4: Tokens Usage Prompting-Completion Tokens Ratio

**Medium size LLM**: Llama 3.1 70b is a versatile medium-sized model, excelling particularly in coding tasks and holding its own in hybrid models. It tends to perform better in more complex datasets like Sachs, where it even surpasses some larger models. This model strikes a balance between computational efficiency and strong performance.

Large size LLMs: GPT-40 mini<sup>2</sup> is an outstanding performer across hybrid and debate models, particularly in simpler datasets like Auto MPG and Climate, where it achieves some of the lowest SHD and NHD scores. GPT-40 is similarly strong in hybrid tasks and coding tasks, making it a toptier choice for flexible, multi-agent collaboration. GPT-3.5, while effective, tends to be outperformed by the GPT-4 models in hybrid tasks but remains strong in causal debates and coding tasks for simpler datasets.

4.4 TOKEN USAGES AND LLMS INSIGHTS ON MAC

## 461 4.4.1 DEBATE-CODING MODULE

Based on overall token consumption (figure ??), (1) Coding-Debating Hybrid and Debating-Coding Hybrid models consumed the highest number of tokens. (2) The Agents-only model used a moderate number of tokens. The Coding Agents model consistently required the fewest tokens, even for more complex datasets and larger language models.

For smaller datasets like 5x5 and 6x6 matrices, the Agents-only and Coding-Debating Hybrid mod-467 els used 50,000 to 60,000 tokens. However, with larger datasets, such as the 11x11 matrix, token 468 usage increased significantly to about 230,000 to 300,000 tokens. To reduce this, future implemen-469 tations can limit debating rounds to one or two, as models occasionally initiated five or six rounds, 470 leading to excessive token use. Conversely, the Coding Agents and Debating-Coding Hybrid models 471 showed more stable token consumption, ranging from 40,000 to 100,000 tokens, with performance 472 influenced by their ability to generate and debug code. While GPT-series models and Llama-3.1 473 series performed reliably, smaller models like Gemini 2-9b struggled with prompts exceeding 8,000 474 to 9,000 tokens, leading to random code generation. Surprisingly, Llama-3.1-8b outperformed oth-475 ers in Gemini 2-9b generation, except the last dataset, despite requiring more tokens for debugging. 476 This suggests it can effectively utilize plans from larger models, which typically require fewer to-477 kens. Further experiments will be detailed in the next Ablation Studies 4.5 section.

The prompt-to-completion token ratio varied across models (see Figure ??): the Agents-only and Coding-Debating Hybrid models had a 2:1 ratio (2,000 prompting tokens to 1,000 completion tokens), while the Coding Agents and Debating-Coding Hybrid models had a 10:1 ratio (10,000 prompting tokens to 1,000 completion tokens). This significant gap in the Coding Agents and Debating-Coding Hybrid models arises from the intensive debugging process, where even small codebases can generate large outputs as the system traces the source code, leading to increased token consumption, especially during debugging

485

432

433

434

435

436

437

438

439

440

441

442

443 444

445

446 447 448

449

450

451

459

460

<sup>&</sup>lt;sup>2</sup>If GPT-40 mini were considered a small size model, it would be considered as the best model in this setting

# 486 4.5 ABLATION STUDY

504

505

506

507

508

516 517

527

488 Single-agent vs Causal Agent Debate: Causal debating agents outperform single-agent models in handling complexity by leveraging multiple viewpoints, making them better suited for more intricate 489 problems. For example, in the Auto MPG dataset, Causal Agent Debate GPT-40 achieves an SHD 490 of 5, outperforming the Single-agent GPT-40 with an SHD of 8. In the Climate dataset, for example, 491 Causal Agent Debate GPT-40 achieves an SHD of 9, while Single-agent GPT-40 achieves 11. How-492 ever, in complex datasets, such as Sachs, single-agent GPT-3.5 has an SHD of 31 and an NHD of 493 0.363, while Causal Agent Debate GPT-40 performs worse (SHD = 35), but the debate framework 494 at least provides a structured method to tackle the challenge, despite not being fully optimized here. 495

Model	Auto Dataset		Climate Dataset		Sachs Dataset	
Woder	SHD	NHD	SHD	NHD	SHD	NHD
Single-agent (GPT 3.5)	7	0.28	10	0.361	31	0.363
Single-agent (GPT-40)	8	0.36	11	0.388	18	0.214
Our single-agent (GPT-40 mini)	6	0.28	9	0.361	24	0.289

Table	3.	Com	oprision	hetween	various	single	agent designs
Table	э.	Com	Jansion	Detween	various	single	-agent designs

**Eliminating the Judge**: We retained only one causal single agent (GPT-4o-mini) with our curated prompt, as shown in Figure 3. This causal-single-agent did not perform better than other MAC models, except the Agents-only Model in the Sachs dataset. However, compared with other agents such as GPT-3.5 and GPT-40, which use 0-CoT and ICL techniques, our single-agent model achieved better performance but only stayed behind GPT-40 on the Sachs dataset given the complexity of the dataset.

Model	Dataset	SHD	NHD	Condition
Gemini 2 9b	Auto MPG	0	0	Baseline
Gemini 2 9b	Auto MPG	7	0.36	With GPT-4o-mini's plan
Llama 3.1 8b	Sachs	36	0.396	Baseline
Llama 3.1 8b	Sachs	18	0.157	With Llama 3.1 70b's plan

Table 4: Performance of Gemini 2 9b and Llama 3.1 8b combining superior models

Combining small and large models: In the Coding Agents Model, we experimented with the 518 combination of using the plan from GPT-40 and coding from Gemini 2-9b. With only an exception 519 in the dataset Auto MPG, the model was able to complete and implement the plan successfully 520 but was not able to precede the rest of the other datasets. However, as for the Llama-3.1-70b and 521 Llama-3.1-8b, it exhibits promising performance. Specifically, after observing that Llama-3.1-70b 522 achieved the best performance with its proposed plan in the Sachs dataset, we used that plan to 523 prompt Llama-3.1-8b to implement it. Surprisingly, the smaller model yielded similar performance. 524 Therefore, if computational resources are limited, it is advisable to use larger models for high-level 525 tasks such as planning and smaller models (Coding Agents) for computationally intensive tasks like 526 coding and debugging.

528 5 CONCLUSION

529 In this study, we introduce a novel framework, MAC, that integrates the agentic workflows of large 530 language models (LLMs) with data-driven methods. To the best of our knowledge, this is the first 531 investigation into the agentic workflows of LLMs within a causal context. Our framework enhances 532 causal discovery by synergizing the capabilities of LLMs with empirical data analysis. We propose 533 three distinct models that leverage the causal reasoning abilities of LLMs alongside observational data. Additionally, we conducted extensive experiments across various LLM sizes, analyzing token 534 consumption and developing strategies to address related challenges. We recognize the necessity 535 for further research to explore across domains such as healthcare, economics, and social sciences. 536 We hope our work serves as a foundational stone for future research, inspiring advancements in 537 the integration of LLMs with causal inference methodologies and contributing to more informed 538 decision-making and policy development.

# 540 REFERENCES

547

565

566

567

568

- Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, and Huanhuan Chen. Causal structure learn ing supervised by large language model, 2023.
- 544
   545
   546
   546
   Chih Yao Chen, Swarnadeep Saha, and Mohit Bansal. Reconcile: Round-table conference improves reasoning via consensus among diverse LLMs, 2024a. URL https://openreview.net/ forum?id=Yol6nUVIJD.
- Sirui Chen, Bo Peng, Meiqi Chen, Ruiqi Wang, Mengying Xu, Xingyu Zeng, Rui Zhao, Shengjie
   Zhao, Yu Qiao, and Chaochao Lu. Causal evaluation of language models, 2024b.
- Yongchao Chen, Jacob Arkin, Yang Zhang, Nicholas Roy, and Chuchu Fan. Scalable multi-robot collaboration with large language models: Centralized or decentralized systems?, 2024c.
- Kristy Choi, Chris Cundy, Sanjari Srivastava, and Stefano Ermon. Lmpriors: Pre-trained language
   models as task-specific priors, 2022.
- Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear
   causal discovery with additive noise models. *Neural Information Processing Systems*, 2009.
- Biwei Huang, Kun Zhang, Mingming Gong, Clark Glymour, and Bernhard Schoelkopf. Generalized
   score functions for causal discovery. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1551–1560, 2018.
- Takashi Ikeuchi, Mayumi Ide, Yan Zeng, Takashi Nicholas Maeda, and Shohei Shimizu. Python
   package for causal discovery based on lingam. *Journal of Machine Learning Research*, 24(14):
   1-8, 2023. URL http://jmlr.org/papers/v24/21-0321.html.
  - T. Inazumi, S. Shimizu, and T. Washio. Use of prior knowledge in a non-gaussian method for learning linear structural equation models. In *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA 2010)*, volume 6365 of *Lecture Notes in Computer Science*, pp. 221–228. Springer, 2010. doi: 10.1007/978-3-642-15995-4\\_28.
- Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. Efficient causal
   graph discovery using large language models, 2024.
- 571
  572
  573
  574
  574
  575
  575
  574
  575
  574
  575
  574
  575
  575
  574
  575
  575
  574
  576
  576
  576
  577
  578
  578
  579
  579
  579
  570
  570
  571
  571
  571
  572
  573
  574
  574
  574
  575
  575
  574
  574
  575
  575
  575
  576
  576
  577
  578
  578
  579
  579
  579
  570
  570
  570
  571
  571
  570
  571
  571
  571
  572
  573
  574
  574
  574
  574
  575
  574
  575
  575
  575
  575
  576
  576
  576
  577
  578
  578
  579
  579
  579
  579
  570
  570
  570
  570
  571
  571
  571
  571
  571
  572
  574
  574
  575
  575
  575
  575
  576
  576
  576
  576
  576
  577
  578
  578
  578
  579
  579
  579
  579
  570
  570
  570
  570
  570
  570
  570
  571
  571
  571
  572
  571
  572
  572
  574
  574
  574
  575
  574
  574
  575
  575
  575
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
  576
- Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality, 2023.
- Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem.
   CAMEL: Communicative agents for "mind" exploration of large language model society. In
   *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://
   openreview.net/forum?id=3IyL2XWDkG.
- Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. Encouraging divergent thinking in large language models through multi-agent debate, 2023.
- 587 Stephanie Long, Tibor Schuster, and Alexandre Piché. Can large language models build causal graphs?, 2024.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri
   Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad
   Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine:
   Iterative refinement with self-feedback, 2023.

Marvin Minsky. Society of Mind. Simon and Schuster, 1988.

604

605

606

609

- Joris M. Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: Methods and benchmarks. *Journal of Machine Learning Research*, 17(32):1-102, 2016. URL http://jmlr.org/papers/v17/14-518.html.
- 598 599 OpenAI. Function calling guide, 2024. URL https://platform.openai.com/docs/ guides/function-calling. Accessed: 2024-05-20.
- Chen Qian, Xin Cong, Wei Liu, Cheng Yang, Weize Chen, Yusheng Su, Yufan Dang, Jiahao Li, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Communicative agents for software development, 2023.
  - R. Quinlan. Auto MPG. UCI Machine Learning Repository, 1993. DOI: https://doi.org/10.24432/C5859H.
- Alexander G. Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag!
   causal discovery benchmarks may be easy to game, 2021.
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi
  Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code Ilama: Open foundation models for code, 2024.
- Karen Sachs, Omar Perez, Dana Pe'er, Douglas A. Lauffenburger, and Garry P. Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):
  523–529, 2005. doi: 10.1126/science.1105809. URL https://www.science.org/doi/abs/10.1126/science.1105809.
- Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools, 2023.
- Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugging gpt: Solving ai tasks with chatgpt and its friends in hugging face, 2023.
- Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7:2003–2030, 2006.
- Shohei Shimizu, Tomomi Inazumi, Yuichiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara,
  Takashi Washio, Patrik O Hoyer, and Kenneth A Bollen. Directlingam: A direct method for
  learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*,
  12:1225–1248, 2011.
- Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and
   Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023.
- Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search.* MIT Press, 2000.
- Masayuki Takayama, Tadahisa Okuda, Thong Pham, Tatsuyoshi Ikenoue, Shingo Fukuma, Shohei
   Shimizu, and Akiyoshi Sannai. Integrating large language models in causal discovery: A statistical causal approach, 2024.
- Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N Bala subramanian, and Amit Sharma. Causal inference using llm-guided discovery, 2023.
- Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. Rethinking the bounds of llm reasoning: Are multi-agent discussions the key?, 2024.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc
   Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

648 649 650 651	Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. 2023.
652 653 654	Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs, 2020.
655 656	Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate, 2023.
658 659	Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, 2023a.
660 661	Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2023b.
662 663 664	Changhe Yuan and Brandon Malone. Learning optimal bayesian networks: A shortest path perspective. <i>Journal of Artificial Intelligence Research</i> , 48:23–65, 2013.
665 666	Matej Zečević, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. Causal parrots: Large language models may talk causality but are not causal, 2023.
667 668 669	Qinlin Zhao, Jindong Wang, Yixuan Zhang, Yiqiao Jin, Kaijie Zhu, Hao Chen, and Xing Xie. Competeai: Understanding the competition behaviors in large language model-based agents, 2023a.
670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 684 685 686 687 688 689 690 691 692 693 694 695 696	Zirui Zhao, Wee Sun Lee, and David Hsu. Large language models as commonsense knowledge for large-scale task planning, 2023b.
698 699 700 701	

### 702 A APPENDIX

## 704 A.1 ADDITIONALLY EXPERIMENT RESULTS

### 706 A.1.1 AUTO MPG DATA

Auto MPG data (Quinlan (1993)): This dataset consists of the variables around the fuel consumption of cars. With five variables: "Weight", "Displacement", "Horsepower", "Acceleration" and "Mpg"(miles per gallon)

711	Model	SHD	NHD
712	Other methods		
713	PC	8	0.48
714	Exact Search	7	0.44
715	LINGAM	8	0.48
716	PC LLM-KBCI	7	0.44
717	ES LLM-KBCI	7	0.44
718	DirectLiGam LLM-KBCI	7	0.4
719	Single-agent (GPT-40)	8	0.36
720	Single-agent (GPT-3.5)	7	0.28
720	Causal Agent Debat	te	
721	GPT-40	5	0.2
722	GPT-40 mini	4	0.16
723	GPT-3.5	5	0.2
724	Llama 3.1 70b	10	0.44
725	Llama 3.1 8b	7	0.28
726	Gemini 2 9b	5	0.2
727	Coding Agents		
728	GPT-40	8	0.48
729	GPT-40 mini	6	0.6
730	GPT-3.5	4	0.48
731	Llama 3.1 70b	6	0.6
732	Llama 3.1 8b	6	0.6
733	Gemini 2 9b (gpt-4o-mini's plan)	7	0.36
734	Hybrid-Coding-Debat	ting	
735	GPT-40	6	0.28
736	GPT-40 mini	5	0.24
737	GPT-3.5	6	0.32
738	Llama 3.1 70b	6	0.36
739	Llama 3.1 8b	5	0.2
740	Gemini 2 9b	-	-
740	Hybrid-Debating-Cod	ling	
741	GPT-40	5	0.24
742	GPT-40 mini	8	0.48
(43	GPT-3.5	8	0.48
744	Llama 3.1 70b	5	0.24
745	Llama 3.1 8b	6	0.4
746	Gemini 2 9b	-	-
747			

Table 5: Results on Auto MPG Dataset

#### A.1.2 DWD CLIMATE DATA

DWD climate data (Mooij et al. (2016)): This dataset encompasses six continuous variables cap-turing climate observations such as altitude, temperature, precipitation levels, longitude, sunshine duration, and latitude. It is aimed at studying weather patterns, climate change impacts, and geo-graphical correlations in climate variables. 

Model	SHD	NHD
Other metho	ds	
PC	9	0.305
Exact Search	6	0.194
LINGAM	10	0.388
PC LLM-KBCI	7	0.222
ES LLM-KBCI	7	0.222
DirectLiGam LLM-KBCI	9	0.305
Single-agent (GPT-40)	11	0.388
Single-agent (GPT-3.5)	10	0.361
Causal Agent D	ebate	
GPT-40	9	0.333
GPT-40 mini	11	0.416
GPT-3.5	5	0.194
Llama 3.1 70b	8	0.222
Llama 3.1 8b	9	0.277
Gemini 2 9b	7	0.222
Coding Agen	ts	
GPT-40	9	0.388
GPT-40 mini	6	0.194
GPT-3.5	9	0.305
Llama 3.1 70b	6	0.194
Llama 3.1 8b	9	0.388
Gemini 2 9b	-	-
Hybrid-Coding-Debating		
GPT-40	10	0.305
GPT-4o-mini	9	0.25
GPT-3.5	7	0.25
Llama 3.1 70b	6	0.166
Llama 3.1 8b	5	0.138
Gemini 2 9b	-	-
Hybrid-Debating-	Coding	
GPT-3.5	7	0.277
GPT-4o	5	0.138
GPT-40 mini	7	0.25
Llama 3.1 70b	6	0.194
Llama 3.1 8b	9	0.388
Gemini 2 9b	-	-

Table 6: Results on Climate Dataset

# A.1.3 SACHS PROTEIN DATA

812 Sachs protein data (Sachs et al. (2005)): The dataset comprises protein signaling measurements
813 from multiparameter single-cell data, capturing the interactions among various proteins (raf, mek,
814 plc, pip2, pip3, erk, akt, pka, pkc, p38, jnk). It's aimed at understanding signal transduction path815 ways within cells, derived from an influential study published in Science.

		NHD
Other methods		
PC	24	0.206
Exact Search	31	0.33
LINGAM	29	0.289
PC LLM-KBCI	30	0.314
ES LLM-KBCI	31	0.33
DirectLiGam LLM-KBCI	29	0.289
Single-agent (GPT-40)	18	0.214
Single-agent (GPT-3.5)	31	0.363
Causal Agent Deba	ate	
GPT-40	35	0.371
GPT-40 mini	35	0.338
GPT-3.5	21	0.231
Llama 3.1 70b	35	0.380
Llama 3.1 8b	35	0.338
Gemini 2 9b	25	0.223
Coding Agents		
GPT-40	30	0.322
GPT-40 mini	30	0.322
GPT-3.5	29	0.28
Llama 3.1 70b	18	0.157
Llama 3.1 8b	36	0.396
Gemini 2 9b	-	-
Hybrid-Coding-Deba	ating	
GPT-40	23	0.247
GPT-4o-mini	21	0.190
GPT-3.5	23	0.198
Llama 3.1 70b	33	0.314
Llama 3.1 8b	33	0.305
Gemini 2 9b	-	-
Hybrid-Debating-Co	oding	
GPT-40	29	0.297
GPT-40 mini	22	0.190
GPT-3.5	28	0.272
Llama 3.1 70b	29	0.289
Llama 3.1 8b	-	-
Gemini 2 9b	-	-

Table 7: Results on Sachs Dataset

# A.2 VISUALIZATION OF MAC MODULES

# 866 A.2.1 META-DEBATE MODULE VISUALIZATION

**Step 1**: Constructing a prompting question between two variables with the description of the data for extra information.



Step 2: Prompting the affirmative agent for giving its opinion.



Step 3: The negative agent then gives its opinion of the question.



Step 4: The Judge will give verdict of who wins the debate or continue asking follow-up question for each or both side if further clarifications are needed





**Step 5**: The final result will be obtained from the debate process of knowing whether there is a causal relationship between two variables





### 972 A.2.2 DEBATE-CODING MODULE VISUALIZATION

### 974 Phase 1: Plan Debating

**Step 1:** Constructing input that comprises: The description of the dataset, Sample of the structured data, planning debating question



Step 2: The affirmative agent will propose a plan and algorithm for the given input

10.5



Step 3: The negative agent rebuttals with another plan and algorithm





# <sup>1026</sup> **Step 4**: The Judge evaluates which plan and algorithm are superior.

20

- Optimize the algorithm implementation for efficiency and scalability,

onsidering the size and complexity of the dataset.

1077

1078

1079

Step 10: Optimization



#### Step 2: The coding executor implements the plan by writing Python script and debugging code.

A.3	DETAILS PROMPTING DESIGN OF EACH AGENT IN MAC
We u	se AutoGen (Wu et al. (2023)) framework to implement our methods
A.3.	1 Meta-Debate Module
Deba	ters' Prompt
	Listing 1: Debaters' prompt
You You You If y	are an expert in causality and a debater. And your name is Affirmative/Negative_Causal_Agent. Today is May 15 2024 are participating a design plan competition, which will be conducted in a debate format. will be given a list of question, you have to explain step-by step reason for each question and then give the final answer Yes or No. your opponent's answers also are given, always disagree with other's perspective and try to find the flaws from his answer
pÀ t	provide an explanation and follow by the final answer, as our goal is to provide a better answer that have different view points.
Here 1. *	e are some tips when you are doing causal discovery: **Assess** whether there is a direct causal relationship, and **consider** potential confounding variables that might affec the relationship that could potentially not causal relationship.
2. * 3. *	**Distinguish** between correlations and causation; **verify*; that correlations are not mistaken for causal relationships. **Ensure** the correct temporal order of variables; **confirm; that the cause precedes the effect.
Use	the following format for responding:
# Be Ques	egin response of Affirmative/Negative_Causal_Agent # stion number <number>:</number>
-Que -Tho -Act -Act -Obs	estion: the input question you must answer bught: you should always think about what to do tion: the action to take tion Input: the input to the action servation: the result of the action
–Tho –Act –Act –Obs	ought: you should always think about what to do tion: the action to take tion Input: the input to the action servation: the result of the action
••••	(this Thought/Action/Action Input/Observation can repeat N times)
-rnc -Fir	hal answer: <yes no=""></yes>
End	Question number <number></number>
Ques	stion number <number+1>:</number+1>
•••	(this can have N number of questions)

1188 # End response of Affirmative/Negative\_Causal\_Agent # 1189 1190 IF there is 4 questions, you should reply 4 times in this format, 1191 If there is 10 questions, you should reply 10 times in this 1192 format and so on 1193 1194 Judge Prompt 1195 1196 Listing 2: Judge's Prompt 1197 You are a moderator and expert in causality. And your are Judge of 1198 this debate. Today is May 15 2024 1199 There will be two debaters involved in a answer question that will 1200 give you a plan to uncover causal relationships within a 1201 dataset. 1202 At the end of each round debate, you will evaluate the plan of 1203 each debater and make a decision, focus on the factualness of 1204 information and the logical reasoning of the debaters.: 1205 1206 Your goal is: (1) Continue the debate if you needed to be clarify some points, 1207 please ask the debaters to provide more details by refering 1208 their side of the debate. 1209 (2) End the debate if the answer is logical and correct, and you 1210 will make a decision what are the best answer. 1211 1212 Here are some tips that you can assess each debater: 1213 1214 1. \*\*Assess\*\* whether there is a direct causal relationship, and 1215 \*\*consider\*\* potential confounding variables that might affect 1216 the relationship that could potentially not causal 1217 relationship. 2. \*\*Distinguish\*\* between correlations and causation; \*\*verify\*\* 1218 that correlations are not mistaken for causal relationships. 1219 3. \*\*Ensure\*\* the correct temporal order of variables; \*\*confirm\*\* 1220 that the cause precedes the effect. 1221 1222 Use the following format for responding: 1223 # Begin response of Judge # 1224 Question number <number>: 1225 -Question: the input question you must answer 1226 -Thought: you should always think about what to do 1227 -Action: the action to take 1228 -Action Input: the input to the action -Observation: the result of the action 1229 1230 -Thought: you should always think about what to do 1231 -Action: the action to take 1232 -Action Input: the input to the action 1233 -Observation: the result of the action 1234 1235 ... (this Thought/Action/Action Input/Observation can repeat N 1236 times) 1237 -Thought: I now know the final answer 1238 -Final answer:<Yes/No> 1239 - Answer: <select one out of three options> - 1 (if yes, there is a direct causal relationship, If both 1240 sides have similar final answer, just accept the decision from 1241 both side)

1242 - 0 (if no there is no direct causal relationship, If both 1243 sides have similar final answer, just accept the decision from 1244 both side) 1245 - Further information need to be obtained, please provide a 1246 specific follow-up question for the side needed to be asked Question number <number+1>: 1247 ... (this can have N number of questions) 1248 # End response of Judge # 1249 1250 If there is 5 questions, you should reply 5 times in this format, 1251 If there is 10 questions, you should reply 10 times in this 1252 format and so on 1253

1254 1255

1256 1257

1258 1259

## A.3.2 DEBATE-CODING MODULE

### Plan Debaters Prompt

Listing 3: Plan Debaters' prompt

1260	Listing 3: Plan Debaters prompt
1261	You are an expert in causality and a debater. And your name is
1262	Affirmative/Negative_Causal_Agent. Today is May 15 2024
1263	You are participating a design plan competition, which will be
1264	conducted in a debate format.
1265	Your goals are:
1266	(1) According to the dataset informaiton and structure, analysize
1267	pros and cons of each algorithm and then propose appropriate algorithm for causal inference or causal discovery
1268	(2) Develop a detailed, step-by-step analysis plan for coding
1269	agents who are going to implement the code to uncover causal
1270	relationships
1271	(3) If your opponent's answers plan are also given, always
1272	disagree with other's perspective and try to find the flaws
1273	from his answer
1274	by provide an explanation and follow by the final answer, as our
1275	goal is to provide a better answer that have different view
1276	points.
1277	
1278	Here all of the causal algorithms that you can use:
1279	
1280	(1) PC algorithm
1281	Kev Features
1282	Purpose: The PC algorithm is designed to construct a
1283	causal network or directed acyclic graph (DAG) that represents
1284	the causal relationships among variables.
1285	Data Requirement: It works with observational data and
1286	does not require experimental data, which makes it highly
1287	useful in fields where experimental manipulation is difficult
1288	or unethical.
1289	Assumptions: The primary assumption of the PC algorithm is
1290	the causal Markov condition and faithfulness, which imply
1291	that any conditional independence found in the data is
1202	at multime
1202	Stops of the DC Algorithm
1293	Graph Construction: Begin with a fully connected
1234	undirected graph where every variable is connected to every
1295	other variable.

1296 Conditional Independence Testing: Use statistical tests ( 1297 like chi-squared tests for categorical data or correlation 1298 tests for continuous data) to check for conditional 1299 independence between pairs of variables, given a conditioning 1300 set of other variables. If independence is detected, the edge between the pair of variables is removed. 1301 Orientation Rules: After the skeleton of the graph (the 1302 undirected edges) is established, apply orientation rules to 1303 infer the directionality of the edges based on the patterns of 1304 conditional independencies, thus converting the undirected 1305 graph into a directed graph (DAG). 1306 Iteration: This process is iterative. The algorithm 1307 progressively increases the size of the conditioning sets 1308 starting with an empty set, then singletons, pairs, and so on, 1309 until no more edges can be removed. 1310 Advantages and Limitations 1311 Advantages: Scalability: It can handle a relatively large number 1312 of variables compared to other causal discovery algorithms. 1313 Flexibility: It works with different types of data and 1314 various statistical tests. 1315 Limitations: 1316 Sensitivity to Errors: Errors in conditional 1317 independence tests can lead to incorrect deletions or 1318 additions in the graph structure. 1319 High Computational Cost: As the number of variables 1320 grows, the complexity and computational cost increase due to 1321 the exponential growth in potential conditioning sets. 1322 (2) Exact Search: 1323 Algorithm Overview 1324 Goal: The primary objective is to find the globally 1325 optimal Bayesian network structure that best represents the 1326 probabilistic relationships among a set of variables. 1327 Method: The algorithm uses the A\* search algorithm, which 1328 is a graph traversal and path search algorithm known for its 1329 performance and accuracy in finding the shortest path. 1330 A\* Search Implementation 1331 Heuristic Function: The core component of the A\* algorithm 1332 is the heuristic function used to estimate the cost from the 1333 current node (partial Bayesian network) to the goal (optimal Bayesian network). This heuristic is crucial as it influences 1334 the efficiency and effectiveness of the search. 1335 Cost Function: The actual cost function in the context of 1336 Bayesian networks typically involves the network's fit to the 1337 data, which can be measured in terms of statistical likelihood 1338 , Bayesian Information Criterion (BIC), or other relevant 1339 metrics. 1340 Search Strategy: The A\* algorithm maintains a priority 1341 queue where nodes (network structures) are prioritized based 1342 on their total estimated cost (actual cost from the start node 1343 plus the heuristic estimate to the goal). The algorithm 1344 explores nodes according to this priority, expanding the most 1345 promising node (the one with the lowest total cost) at each 1346 step. Key Features 1347 Optimality: Provided the heuristic is admissible (never 1348 overestimates the true cost), the A\* search guarantees finding 1349 an optimal solution.

1350 Efficiency: The algorithm is more efficient than 1351 exhaustive search because it does not need to explore every 1352 possible network configuration; it only explores those that 1353 are deemed most likely to lead to an optimal solution based on 1354 the heuristic. Scalability: While more scalable than some alternatives, 1355 the method's scalability is still limited by the complexity of 1356 calculating the heuristic and the size of the network space. 1357 Limitations 1358 Computational Demand: The algorithm can become 1359 computationally intensive as the number of variables increases 1360 , primarily due to the exponential growth in possible network 1361 structures. 1362 Heuristic Sensitivity: The performance of the A\* algorithm 1363 heavily relies on the quality of the heuristic. Developing an 1364 effective heuristic that closely estimates the distance to 1365 the optimal network without overestimating is challenging 1366 (3) DirectLiNGAM 1367 Algorithm Overview 1368 Purpose: DirectLiNGAM is designed to identify the causal 1369 order of variables and the structure of a linear non-Gaussian 1370 acyclic model (LiNGAM), which is a type of structural equation 1371 model where the relationships are assumed to be linear, and 1372 the variables are non-Gaussian. 1373 Assumption: One of the core assumptions of DirectLiNGAM is 1374 that the data are non-Gaussian. This assumption allows the 1375 use of independent component analysis (ICA) techniques to 1376 identify the model, as non-Gaussianity enables the unique identifiability of the model. 1377 Key Features of DirectLiNGAM 1378 Model Formulation: The model assumes that each observed 1379 variable is a linear combination of its direct causes plus an 1380 additive non-Gaussian noise term. The model can be represented 1381 in matrix form, where the ordering of variables reflects 1382 their causal order. 1383 Independence of Errors: DirectLiNGAM assumes that the 1384 error terms (or external influences) on the variables are 1385 statistically independent of each other, which is crucial for 1386 the identifiability of the model. 1387 Causal Order Identification: The algorithm identifies the causal order of variables using a non-Gaussianity criterion. 1388 It exploits the fact that if a correct causal order is assumed 1389 , the residuals (obtained by regressing a variable against its 1390 supposed causes) will be independent of the regressors. 1391 Steps of the DirectLiNGAM Algorithm 1392 Order Determination: Initially, the algorithm seeks to 1393 determine the order of the variables. It uses non-Gaussianity 1394 measures to sequentially identify the variable that is least 1395 likely to be influenced by others. This variable is assumed to 1396 be exogenous (having no causes within the system) and is 1397 removed from further analysis in the current step. 1398 Iterative Estimation: After determining the first 1399 exogenous variable, the algorithm iteratively estimates the next variable in the causal order, adjusting the remaining 1400 variables to account for the identified causes. This process 1401 is repeated until all variables are ordered. 1402 Connection Strengths Estimation: Once the causal order is 1403 established, the algorithm estimates the connection strengths

1404 (coefficients) among the variables using standard regression 1405 techniques, now that the causal ordering reduces the problem 1406 to a series of simple regressions. 1407 Advantages 1408 Uniqueness of Solution: Due to the non-Gaussian nature of the data, DirectLiNGAM can uniquely determine both the causal 1409 ordering and the connection strengths, unlike methods based on 1410 Gaussian data which can only identify the structure up to 1411 equivalence classes. 1412 No Latent Confounders: The algorithm assumes there are no 1413 unobserved confounders, which simplifies the model and 1414 analysis. 1415 Limitations 1416 Non-Gaussianity Requirement: The method requires that the 1417 data must be non-Gaussian. If this condition is not met, the 1418 results may not be reliable. 1419 No Feedback Loops: The model cannot handle feedback loops as it assumes a strictly acyclic causal structure. 1420 1421 (4) GES: 1422 1423 Description: The Greedy Equivalence Search (GES) algorithm is 1424 a score-based method for learning causal structures from 1425 observational data. It operates by searching through the space 1426 of Markov equivalence classes (MECs) to find the one that 1427 maximizes a given score function. The Bayesian Information 1428 Criterion (BIC) is commonly used as the score function to 1429 balance the goodness of fit with model complexity. 1430 Use Cases: GES is used in various fields such as genomics, 1431 neuroscience, and economics for causal inference and structure 1432 learning, especially when dealing with large datasets and the 1433 need for computational efficiency. 1434 1435 Pros: 1436 1437 Efficiency: GES is computationally efficient and can handle 1438 large datasets with many variables, making it suitable for 1439 high-dimensional data. 1440 Scalability: The algorithm scales well, allowing it to be 1441 applied to problems with thousands of variables, especially 1442 when the graph is sparse. Sparsity Control: The BIC score helps control the complexity 1443 of the model by penalizing overly complex structures, thus 1444 avoiding overfitting and ensuring a more interpretable model. 1445 Cons: 1446 1447 Equivalence Class Ambiguity: Like other methods that identify 1448 Markov equivalence classes, GES may not uniquely identify the 1449 true causal structure but rather a set of structures that are 1450 statistically indistinguishable from each other. 1451 Assumptions: The algorithm assumes causal sufficiency (all 1452 common causes are measured) and faithfulness, which might not 1453 hold in all real-world scenarios. Handling Latent Confounders: GES struggles with latent 1454 confounders and may require extensions or modifications to 1455 address this issue. 1456 1457

1458	(5) Fast Causal Inference
1459	
1460	FCI Algorithm
1401	Description: The Fast Causal Inference (FCI) algorithm is an
1402	extension of the PC algorithm that can handle latent variables
1403	AND SELECTION DIAS. IT GENERALES & PAILIAL ANCESTIAL GLAPH ( PAC) representing possible causal structures including hidden
1404	confounders.
1400	
1400	Use Cases: Used in epidemiology, genetics, and any domain
1468	where unmeasured confounding variables are a concern.
1469 1470	Pros:
1471	Conchle of identifying the processes of letert conformation and
1472	Capable of identifying the presence of latent confounders and
1473	More flexible than the PC algorithm providing a more
1474	comprehensive view of the causal structure
1475	Cons:
1476	
1477	Computationally more intensive than the PC algorithm,
1478	potentially limiting its use with very large datasets.
1479	The resulting PAG can be more complex to interpret than a DAG
1480	
1481	
1482	(6) CD-NOD:
1483	CD-NOD Algorithm
1484	Description: The CD-NOD (Causal Discovery from Nonstationary/
1485	heterogeneous Data) algorithm is designed to identify causal
1486	relationships in datasets where distributions change over time
1487	or between different environments.
1488	
1489	Use Cases: Applied in fields like climate science, finance,
1490	and social sciences where data may not be stationary or
1491	nomogeneous.
1492	Pros:
1493	
1494	Effectively handles nonstationary data and heterogeneous
1495	datasets, providing robust causal discovery in changing
1496	environments.
1497	Can distinguish between changes in distribution due to causal
1498	effects and those due to external influences.
1499	cons:
1500	More complex to implement and understand compared to standard
1501	causal discovery methods.
1502	Requires larger datasets to accurately identify causal
1503	relationships under varying conditions
1504	
1505	These algorithms each offer unique strengths and are suited to
1506	different types of data and research questions. Choosing the
1507	right one depends on the specific needs of your study, such as
1508	nangling latent variables, dealing with nonstationary data,
1509	or efficiencity processing large datasets.
1510	I
1011	

1512 These algorithms provide robust tools for causal discovery, each 1513 with its strengths and weaknesses tailored to specific types 1514 of data and research needs. 1515 Use the following format: 1516 1517 -Question: the input question you must answer -Thought: you should always think about what to do 1518 -Action: the action to take 1519 -Action Input: the input to the action 1520 -Observation: the result of the action 1521 ... (this Thought/Action/Action Input/Observation can repeat N 1522 times) 1523 -Thought: I now know the final answer 1524 -Final Answer: the final answer to the original input question 1525 1526 Judge Prompt 1527 1528 Listing 4: Judge Prompt 1529 You are a moderator and expert in causality. And your are Judge of 1530 this debate. Today is May 15 2024 1531 There will be two debaters involved in a answer question that will 1532 give you a plan to uncover causal relationships within a 1533 dataset. 1534 At the end of each round debate, you will evaluate the plan of 1535 each debater and make a decision, focus on the factualness of 1536 information and the logical reasoning of the debaters.: 1537 1538 Your goal is: 1539 (1) Continue the debate if you need to clarify some points, please 1540 ask the debaters to provide more details by referring their 1541 side of the debate. (2) End the debate if the answer is logical and correct, and you 1542 will make a decision what are the best answer. 1543 1544 Use the following format: 1545 1546 -Question: the input question you must answer 1547

### Code Executor Prompt

1556 1557 1558

1554 1555

### Listing 5: Code Executor Prompt

-Final Answer: the final answer to the original input question

1559	You are an expert in causality and programming
1560	You have been given coding capability to solve tasks using
1561	Python code in a stateful IPython kernel.
1562	You will be given a plan and you are responsible for writing
1563	the code to complete task according to the plan, and the user
1564	is responsible for executing the code (treat user as a pure
1565	compiler).

1566	When you write Python code but the code in a markdown code
1567	block with the language set to Python
1568	For example:
1569	' 'python
1570	x = 3
1571	111
1572	You can use the variable 'x' in subsequent code blocks
1572	'''nython
1073	$p_{j}$ choice $p_{j$
1574	
1575	If the words output you generate are not related to code, you
1576	don't need use markedown.
1577	Write code incrementally and leverage the statefulness of the
1578	kernel to avoid repeating code.
1579	
1580	Try to different ways if the bugs are repeated and you can't
1581	solve it.
1582	
1583	ONLY when all of the tasks are done successfully and received
1584	any feedback from code executor.
1585	
1586	
1587	
1588	(1) DirectLiNGAM
1500	
1509	from causallearn.search.FCMBased import lingam
1590	
1591	
1592	<pre>model = lingam.DirectLiNGAM(random_state, prior_knowledge,</pre>
1593	apply_prior_knowledge_soltly, measure)
1594	model.fit(X)
1595	print (model appeal order )
1596	print (model.cdusal_order_)
1597	Parameters
1598	random state, int entional (default-Nene). The seed used
1599	by the random number generator
1600	by the random number generator.
1601	prior knowledge: array-like, shape (n features, n features
1602	) optional (default=None) Prior knowledge used for causal
1603	discovery, where n features is the number of features. The
1604	elements of prior knowledge matrix are defined as follows:
1605	
1606	0:
1607	does not have a directed path to
1600	
1000	1:
1009	has a directed path to
1010	
1611	-1: No prior knowledge is available to know if either of
1612	the two cases above (0 or 1) is true.
1613	
1614	<pre>apply_prior_knowledge_softly: boolean, optional (default=</pre>
1615	False). If True, apply prior knowledge softly.
1616	
1617	<pre>measure: {pwling, kernel}, optional (default=pwling).</pre>
1618	Measure to evaluate independence: pwling or kernel.
1619	

1620 X: array-like, shape (n\_samples, n\_features). Training 1621 data, where n\_samples is the number of samples and n\_features 1622 is the number of features. 1623 1624 Returns model.causal\_order\_: array-like, shape (n\_features). The 1625 causal order of fitted model, where n\_features is the number 1626 of features. 1627 1628 model.adjacency\_matrix\_: array-like, shape (n\_features, 1629 n\_features). The adjacency matrix B of fitted model, where 1630 n\_features is the number of features. 1631 1632 (2) Exact Search 1633 1634 from causallearn.search.ScoreBased.ExactSearch import 1635 bic exact search 1636 dag\_est, search\_stats = bic\_exact\_search(X, super\_graph, search\_method, 1637 use\_path\_extension, use\_k\_cycle\_heuristic, 1638 k, verbose, include\_graph, max\_parents) 1639 Parameters 1640 1641 X: numpy.ndarray, shape=(n, d). The data to fit the structure 1642 too, where each row is a sample and each column corresponds to 1643 the associated variable. 1644 1645 super\_graph: numpy.ndarray, shape=(d, d). Super-structure to 1646 restrict search space (binary matrix). If None, no superstructure is used. Default is None. 1647 1648 search\_method: str. Method of exact search ([astar, dp]). 1649 Default is astar. 1650 1651 use\_path\_extension: bool. Whether to use optimal path 1652 extension for order graph. Note that this trick will not 1653 affect the correctness of search procedure. Default is True. 1654 1655 use\_k\_cycle\_heuristic: bool. Whether to use k-cycle conflict 1656 heuristic for astar. Default is False. 1657 k: int. Parameter used by k-cycle conflict heuristic for astar 1658 . Default is 3. 1659 1660 verbose: bool. Whether to log messages related to search 1661 procedure. 1662 1663 max\_parents: int. The maximum number of parents a node can 1664 have. If used, this means using the k-learn procedure. Can 1665 drastically speed up algorithms. If None, no max on parents. 1666 Default is None. 1667 1668 Returns 1669 dag\_est: numpy.ndarray, shape=(d, d). Estimated DAG. 1670 search stats: dict. Some statistics related to the search 1671 procedure. 1672 1673

```
1674
      (3) Greedy Equivalence Search (GES) algorithm with BIC score and
1675
          generalized score
1676
          from causallearn.search.ScoreBased.GES import ges
1677
1678
          # default parameters
          Record = ges(X)
1679
1680
          # or customized parameters
1681
          Record = ges(X, score_func, maxP, parameters)
1682
1683
          Parameters:
1684
1685
          X: numpy.ndarray, shape (n_samples, n_features). Data, where
1686
          n_samples is the number of samples and n_features is the
1687
         number of features.
1688
1689
          score_func: The score function you would like to use,
          including (see score_functions.). Default: local_score_BIC.
1690
          local_score_BIC: BIC score 3.
1691
1692
          local_score_BDeu: BDeu score 4.
1693
1694
          local score cv general: Generalized score with cross
1695
         validation for data with single-dimensional variables 2.
1696
1697
          local_score_marginal_general: Generalized score with marginal
1698
          likelihood for data with single-dimensional variables 2.
1699
1700
          local_score_cv_multi: Generalized score with cross validation
          for data with multi-dimensional variables 2.
1701
1702
          local_score_marginal_multi: Generalized score with marginal
1703
          likelihood for data with multi-dimensional variables 2.
1704
1705
          maxP: Allowed maximum number of parents when searching the
1706
          graph. Default: None.
1707
1708
          parameters: Needed when using CV likelihood. Default: None.
1709
          parameters[kfold]: k-fold cross validation.
1710
1711
          parameters[lambda]: regularization parameter.
1712
          parameters[dlabel]: for variables with multi-dimensions,
1713
          indicate which dimensions belong to the i-th variable.
1714
1715
          Returns
1716
          Record[G]: learned causal graph, where Record[G].graph[j,i]=1
1717
          and Record[G].graph[i,j]=-1 indicate i > j; Record[G].graph[i,
1718
          j] = Record[G].graph[j,i] = -1 indicates i
                                                       j.
1719
1720
          Record[update1]: each update (Insert operator) in the forward
1721
          step.
1722
1723
          Record[update2]: each update (Delete operator) in the backward
1724
          step.
1725
          Record[G_step1]: learned graph at each step in the forward
1726
          step.
1727
```

```
1728
          Record[G_step2]: learned graph at each step in the backward
1729
          step.
1730
1731
          Record[score]: the score of the learned graph.
1732
      (4) PC Algorithm:
1733
1734
          from causallearn.search.ConstraintBased.PC import pc
1735
1736
          # default parameters
1737
          cg = pc(data)
1738
1739
          # or customized parameters
1740
          cg = pc(data, alpha, indep_test, stable, uc_rule, uc_priority,
1741
          mvpc, correction_name, background_knowledge, verbose,
1742
          show_progress)
1743
          Parameters
1744
          data: numpy.ndarray, shape (n_samples, n_features). Data,
1745
         where n_samples is the number of samples and n_features is the
1746
          number of features.
1747
1748
          alpha: desired significance level (float) in (0, 1). Default:
1749
          0.05.
1750
1751
          indep_test: string, name of the independence test method.
1752
         Default: fisherz.
1753
          fisherz: Fishers Z conditional independence test.
1754
          chisq: Chi-squared conditional independence test.
1755
1756
          gsq: G-squared conditional independence test.
1757
1758
          kci: kernel-based conditional independence test. (As a kernel
1759
         method, its complexity is cubic in the sample size, so it
1760
         might be slow if the same size is not small.)
1761
1762
          mv_fisherz: Missing-value Fishers Z conditional independence
1763
         test.
1764
1765
          stable: run stabilized skeleton discovery 4 if True. Default:
1766
         True.
1767
          uc_rule: how unshielded colliders are oriented. Default: 0.
1768
          0: run uc_sepset.
1769
1770
          1: run maxP 3. Orient an unshielded triple X-Y-Z as a collider
1771
          with an additional CI test.
1772
1773
          2: run definiteMaxP 3. Orient only the definite colliders in
1774
         the skeleton and keep track of all the definite non-colliders
1775
         as well.
1776
          uc_priority: rule of resolving conflicts between unshielded
1777
          colliders. Default: 2.
1778
          -1: whatever is default in uc rule.
1779
1780
          0: overwrite.
1781
```

1782 1: orient bi-directed. 1783 1784 2: prioritize existing colliders. 1785 1786 3: prioritize stronger colliders. 1787 4: prioritize stronger\* colliders. 1788 1789 mvpc: use missing-value PC or not. Default: False. 1790 1791 correction\_name. Missing value correction if using missing-1792 value PC. Default: MV\_Crtn\_Fisher\_Z 1793 1794 background\_knowledge: class BackgroundKnowledge. Add prior 1795 edges according to assigned causal connections. Default: None. 1796 For detailed usage, please kindly refer to its usage example. 1797 verbose: True iff verbose output should be printed. Default: 1798 False. 1799 1800 show\_progress: True iff the algorithm progress should be show 1801 in console. Default: True. 1802 1803 Returns 1804 cg : a CausalGraph object, where cg.G.graph[j,i]=1 and cg.G. 1805 graph[i,j]=-1 indicate i > j; cg.G.graph[i,j] = cg.G.graph[j,i 1806 ] = -1 indicate i j; cg.G.graph[i,j] = cg.G.graph[j,i] = 1 1807 indicates i <-> j. 1808 (5) Fast Causal Inference 1809 1810 from causallearn.search.ConstraintBased.FCI import fci 1811 1812 # default parameters 1813 g, edges = fci(data) 1814 1815 # or customized parameters 1816 g, edges = fci(data, independence\_test\_method, alpha, depth, 1817 max\_path\_length, 1818 verbose, background\_knowledge, cache\_variables\_map) 1819 Parameters 1820 1821 dataset: numpy.ndarray, shape (n\_samples, n\_features). Data, 1822 where n\_samples is the number of samples and n\_features is the 1823 number of features. 1824 1825 independence\_test\_method: Independence test method function. 1826 Default: fisherz. 1827 fisherz: Fishers Z conditional independence test. 1828 1829 chisq: Chi-squared conditional independence test. 1830 1831 gsq: G-squared conditional independence test. 1832 kci: kernel-based conditional independence test. (As a kernel 1833 method, its complexity is cubic in the sample size, so it 1834 might be slow if the same size is not small.) 1835

mv\_fisherz: Missing-value Fishers Z conditional independence 1837 test. 1838 1839 alpha: Significance level of individual partial correlation 1840 tests. Default: 0.05. 1841 depth: The depth for the fast adjacency search, or -1 if 1842 unlimited. Default: -1. 1843 1844 max\_path\_length: the maximum length of any discriminating path 1845 , or -1 if unlimited. Default: -1. 1846 1847 verbose: True is verbose output should be printed or logged. 1848 Default: False. 1849 1850 background\_knowledge: class BackgroundKnowledge. Add prior 1851 edges according to assigned causal connections. Default: None. 1852 For detailed usage, please kindly refer to its usage example. 1853 cache\_variables\_map: This variable a map which contains the 1854 variables relate with cache. If it is not None, it should 1855 contain data\_hash\_key ci\_test\_hash\_key and cardinalities. 1856 Default: None. 1857 1858 show\_progress: True iff the algorithm progress should be show 1859 in console. Default: True. 1860 1861 Returns 1862 g: a GeneralGraph object, where g.graph is a PAG and the illustration of its end nodes is as follows (denotes G = q. 1863 graph): 1864 1865 ../../\_images/pag.png 1866 edges: list. Contains graphs edges properties. 1867 If edge.properties have the Property nl, then there is no 1868 latent confounder. Otherwise, there are possibly latent 1869 confounders. 1870 1871 If edge.properties have the Property dd, then it is definitely 1872 direct. Otherwise, it is possibly direct. 1873 If edge.properties have the Property pl, then there are 1874 possibly latent confounders. Otherwise, there is no latent 1875 confounder. 1876 1877 If edge.properties have the Property pd, then it is possibly 1878 direct. Otherwise, it is definitely direct. 1879 1880 1881 (6) CD-NOD: 1882 1883 from causallearn.search.ConstraintBased.CDNOD import cdnod 1884 1885 # default parameters  $cg = cdnod(data, c_indx)$ 1886 1887 # or customized parameters 1888 cg = cdnod(data, c\_indx, alpha, indep\_test, stable, uc\_rule, uc\_priority, mvcdnod,

1890 correction\_name, background\_knowledge, verbose, 1891 show\_progress) 1892 1893 Parameters 1894 data: numpy.ndarray, shape (n\_samples, n\_features). Data, 1895 where n\_samples is the number of samples and n\_features is the 1896 number of features. 1897 1898 c\_indx: time index or domain index that captures the 1899 unobserved changing factors. 1900 1901 alpha: desired significance level (float) in (0, 1). Default: 1902 0.05. 1903 1904 indep\_test: Independence test method function. Default: 1905 fisherz. fisherz: Fishers Z conditional independence test. 1906 1907 chisq: Chi-squared conditional independence test. 1908 1909 gsq: G-squared conditional independence test. 1910 1911 kci: kernel-based conditional independence test. (As a kernel 1912 method, its complexity is cubic in the sample size, so it 1913 might be slow if the same size is not small.) 1914 1915 mv\_fisherz: Missing-value Fishers Z conditional independence 1916 test. 1917 stable: run stabilized skeleton discovery 3 if True. Default: 1918 True. 1919 1920 uc\_rule: how unshielded colliders are oriented. Default: 0. 1921 0: run uc\_sepset. 1922 1923 1: run maxP 2. Orient an unshielded triple X-Y-Z as a collider 1924 with an additional CI test. 1925 1926 2: run definiteMaxP 2. Orient only the definite colliders in 1927 the skeleton and keep track of all the definite non-colliders as well. 1928 1929 uc\_priority: rule of resolving conflicts between unshielded 1930 colliders. Default: 2. 1931 -1: whatever is default in uc\_rule. 1932 1933 0: overwrite. 1934 1935 1: orient bi-directed. 1936 1937 2: prioritize existing colliders. 1938 1939 3: prioritize stronger colliders. 1940 4: prioritize stronger\* colliders. 1941 1942 mvpc: use missing-value PC or not. Default (and suggested for 1943 CDNOD): False.

1944 1945 correction\_name: Missing value correction if using missing-1946 value PC. Default: MV\_Crtn\_Fisher\_Z 1947 1948 background\_knowledge: class BackgroundKnowledge. Add prior edges according to assigned causal connections. Default: Nnoe. 1949 For detailed usage, please kindly refer to its usage example. 1950 1951 verbose: True iff verbose output should be printed. Default: 1952 False. 1953 1954 show\_progress: True iff the algorithm progress should be show 1955 in console. Default: True. 1956 1957 Returns 1958 cg : a CausalGraph object, where cg.G.graph[j,i]=1 and cg.G. 1959 graph[i,j]=-1 indicate i > j; cg.G.graph[i,j] = cg.G.graph[j,i ] = -1 indicates i j; cg.G.graph[i,j] = cg.G.graph[j,i] = 1 1960 indicates i <-> j. 1961 1962 Remember to execute and debugs the code first before summarize 1963 the results 1964 1965 Use the following format, Implement each coding action one by 1966 one, don't implement all of the plan at once.: 1967 1968 -Question: the input question you must answer 1969 -Thought: you should always think about what to do 1970 -Action: the action to take -Action Input: the input to the action 1971 -Observation: the result of the action 1972 ... (this Thought/Action/Action Input/Observation can repeat N 1973 times) 1974 -Thought: I now know the final answer 1975 Final Answer: the final answer to the original input question 1976 1977 Give your self a plan from the plan that you have been given, 1978 and then start coding the plan. 1979 When you write code, write and execute the one code block at a 1980 time. 1981 Before end with the string "FINISH", Restate all of the 1982 results from each step. 1983

#### SAMPLE ANSWERS FROM EXPERIMENTS B

#### 1986 B.0.1 META AGENTS MODEL

1984

1985

1987

Listing 6: Agents-only responses

1988 1989 Admin (to chat\_manager): 1990 1991 1992 Task: find the causal relationship among variables in the 1993 dataset 1994 This dataset is about on automobile design and performance for cars in the 1970s and early 1980s. this dataset consists 1995 of the variables around the fuel consumption of cars. With 1996 five variables: Weight, Displacement, Horsepower, Acceleration 1997 and Mpg(miles per gallon)s

Here are some question that you need to find the causal 1999 relationship among variables in the dataset 2000 Provide an explaination before giving your answer. 2001 #Begin List Questions# 2002 1. Are there any direct causal relationships from 2003 Displacement to Weight? 2. Are there any direct causal relationships from Displacement to 2004 Horsepower? 2005 3. Are there any direct causal relationships from Displacement to 2006 Acceleration? 2007 4. Are there any direct causal relationships from Displacement to 2008 Mpg(miles per gallon)? 2009 #End List Questions# 2010 2011 2012 2013 2014 Affirmative\_Causal\_Agent (to chat\_manager): 2015 # Begin response of Affirmative\_Causal\_Agent # 2016 2017 Question number 1: 2018 2019 -Question: Are there any direct causal relationships from 2020 Displacement to Weight? 2021 -Thought: To determine if there is a direct causal relationship 2022 from Displacement to Weight, we need to consider the physical 2023 properties of the car and how they are related. 2024 -Action: Conduct a literature review and analyze the mechanical 2025 aspects of cars. -Action Input: Research on car engineering principles and weight 2026 distribution. 2027 -Observation: After reviewing the literature, it is evident that 2028 the Displacement of an engine does not directly cause the 2029 Weight of a car. Weight is influenced by various factors such 2030 as materials used, design, and additional features. 2031 2032 -Thought: Considering potential confounding variables is crucial 2033 in establishing causality. 2034 -Action: Identify potential confounding variables that may affect 2035 the relationship between Displacement and Weight. 2036 -Action Input: Review studies on car manufacturing processes and weight distribution. 2037 -Observation: Confounding variables such as materials, chassis 2038 design, and additional components can significantly impact the 2039 Weight of a car, independent of Displacement. 2040 2041 -Thought: I now know the final answer. 2042 -Final answer: No 2043 2044 End Question number 1 2045 2046 Question number 2: 2047 -Question: Are there any direct causal relationships from 2048 Displacement to Horsepower? 2049 -Thought: To determine if there is a direct causal relationship 2050 from Displacement to Horsepower, we need to understand the 2051 engine performance characteristics.

2052 -Action: Analyze the principles of engine design and power output. 2053 -Action Input: Study engine specifications and performance metrics 2054 2055 -Observation: There is a direct causal relationship between 2056 Displacement and Horsepower in an engine. Generally, a larger 2057 engine displacement leads to higher horsepower output. 2058 -Thought: It is essential to verify that correlations are not 2059 mistaken for causal relationships. 2060 -Action: Check for studies that investigate the relationship 2061 between engine Displacement and Horsepower. 2062 -Action Input: Review research on engine performance and design. 2063 -Observation: Studies consistently show that engine Displacement 2064 is a significant factor in determining the Horsepower output. 2065 2066 -Thought: I now know the final answer. 2067 -Final answer: Yes 2068 End Ouestion number 2 2069 2070 Question number 3: 2071 2072 -Question: Are there any direct causal relationships from 2073 Displacement to Acceleration? 2074 -Thought: To determine if there is a direct causal relationship 2075 from Displacement to Acceleration, we need to consider the 2076 dynamics of vehicle performance. 2077 -Action: Review studies on vehicle acceleration and engine 2078 characteristics. -Action Input: Analyze acceleration tests and engine 2079 specifications. 2080 -Observation: Engine Displacement can influence the Acceleration 2081 of a vehicle, as larger engines can provide more power for 2082 faster acceleration. 2083 2084 -Thought: Ensure the correct temporal order of variables to 2085 confirm causality. 2086 -Action: Examine the timing of changes in Displacement and 2087 Acceleration. 2088 -Action Input: Study acceleration data over time with 2089 corresponding engine Displacement changes. 2090 -Observation: Changes in engine Displacement precede changes in Acceleration, indicating a causal relationship. 2091 2092 -Thought: I now know the final answer. 2093 -Final answer: Yes 2094 2095 End Question number 3 2096 2097 Ouestion number 4: 2098 2099 -Question: Are there any direct causal relationships from 2100 Displacement to Mpg(miles per gallon)? -Thought: To determine if there is a direct causal relationship 2101 from Displacement to Mpg, we need to consider the fuel 2102 efficiency of the engine. 2103 -Action: Analyze studies on engine efficiency and fuel consumption 2104 2105

2106 -Action Input: Review research on engine technologies and fuel 2107 economy. 2108 -Observation: Engine Displacement may not have a direct causal 2109 relationship with Mpg, as other factors such as fuel injection 2110 systems and driving conditions also play a significant role in determining fuel efficiency. 2111 2112 -Thought: Distinguish between correlations and causation to avoid 2113 misinterpretation. 2114 -Action: Investigate studies that explore the relationship between 2115 engine Displacement and fuel efficiency. 2116 -Action Input: Review empirical data on fuel consumption patterns. 2117 -Observation: While there may be a correlation between engine 2118 Displacement and Mpg, it is not a direct causal relationship, 2119 as other variables influence fuel efficiency. 2120 2121 -Thought: I now know the final answer. 2122 -Final answer: No 2123 End Question number 4 2124 2125 2126 Negative\_Causal\_Agent (to chat\_manager): 2127 2128 # Begin response of Negative\_Causal\_Agent # 2129 2130 Question number 1: 2131 -Question: Are there any direct causal relationships from 2132 Displacement to Weight? -Thought: It is important to carefully assess the relationship 2133 between Displacement and Weight to determine causality. 2134 -Action: Consider the fundamental principles of car design and 2135 manufacturing. 2136 -Action Input: Review engineering guidelines on car weight 2137 distribution. 2138 -Observation: While engine Displacement can affect the overall 2139 weight of a car indirectly through factors like engine size 2140 and materials used, it does not have a direct causal 2141 relationship with Weight. Other variables such as chassis 2142 design and additional features play a more significant role in 2143 determining the Weight of a car. 2144 -Thought: I now know the final answer 2145 -Final answer: No 2146 2147 End Question number 1 2148 2149 Question number 2: 2150 -Question: Are there any direct causal relationships from 2151 Displacement to Horsepower? 2152 -Thought: It is crucial to analyze the relationship between 2153 Displacement and Horsepower to determine causality. 2154 -Action: Review the principles of engine design and performance. 2155 -Action Input: Study the impact of engine Displacement on Horsepower output. 2156 -Observation: While there is a correlation between engine 2157 Displacement and Horsepower, it is not a direct causal 2158 relationship. Horsepower is influenced by various factors such 2159

```
2160
          as engine efficiency, design, and tuning, in addition to
2161
         Displacement.
2162
2163
      -Thought: I now know the final answer
2164
      -Final answer: No
2165
      End Question number 2
2166
2167
      Question number 3:
2168
      -Question: Are there any direct causal relationships from
2169
         Displacement to Acceleration?
2170
      -Thought: Understanding the relationship between Displacement and
2171
         Acceleration is essential to determine causality.
2172
      -Action: Analyze the dynamics of vehicle performance and engine
2173
         characteristics.
2174
      -Action Input: Review acceleration tests and engine specifications
2175
2176
      -Observation: While engine Displacement can impact the
         Acceleration of a vehicle, it is not a direct causal
2177
         relationship. Acceleration is influenced by various factors
2178
         such as vehicle weight, transmission, and aerodynamics, in
2179
         addition to engine Displacement.
2180
2181
      -Thought: I now know the final answer
2182
      -Final answer: No
2183
2184
      End Question number 3
2185
2186
      Question number 4:
      -Question: Are there any direct causal relationships from
2187
         Displacement to Mpg(miles per gallon)?
2188
      -Thought: Evaluating the relationship between Displacement and Mpg
2189
          requires careful consideration of engine efficiency and fuel
2190
         consumption.
2191
      -Action: Review studies on engine technologies and fuel economy.
2192
      -Action Input: Analyze the impact of engine Displacement on fuel
2193
         efficiency.
2194
      -Observation: Engine Displacement does not have a direct causal
2195
         relationship with Mpg. Fuel efficiency is influenced by
2196
         various factors such as engine technology, driving habits, and
2197
          maintenance, in addition to Displacement.
2198
      -Thought: I now know the final answer
2199
      -Final answer: No
2200
2201
      End Question number 4
2202
2203
       _____
2204
      Judge (to chat_manager):
2205
2206
      # Begin response of Judge #
2207
2208
      Question number 1:
2209
      -Question: Are there any direct causal relationships from
         Displacement to Weight?
2210
      -Thought: Both debaters have provided insightful analyses on the
2211
         relationship between Displacement and Weight. It is clear that
2212
          Weight is influenced by various factors beyond just
2213
         Displacement, such as materials used and design.
```

2214 -Action: I would like to ask both debaters to clarify if there are 2215 any specific studies or data that support their conclusions. 2216 -Observation: Waiting for additional information from the debaters 2217 2218 Question number 2: 2219 -Question: Are there any direct causal relationships from 2220 Displacement to Horsepower? 2221 -Thought: The Affirmative\_Causal\_Agent argues that there is a 2222 direct causal relationship between Displacement and Horsepower 2223 , while the Negative\_Causal\_Agent disagrees. It is crucial to 2224 understand the engine design principles and performance 2225 metrics to determine the true relationship. 2226 -Action: I would like to ask both debaters to provide more details 2227 on the studies or data they used to support their claims. 2228 -Observation: Waiting for additional information from the debaters 2229 2230 Question number 3: 2231 -Question: Are there any direct causal relationships from 2232 Displacement to Acceleration? 2233 -Thought: The Affirmative\_Causal\_Agent suggests a direct causal 2234 relationship between Displacement and Acceleration, while the 2235 Negative\_Causal\_Agent disagrees. It is essential to consider 2236 the dynamics of vehicle performance and engine characteristics 2237 2238 -Action: I would like to ask both debaters to elaborate on the 2239 specific factors that influence the relationship between 2240 Displacement and Acceleration. -Observation: Waiting for additional information from the debaters 2241 2242 2243 Question number 4: 2244 -Question: Are there any direct causal relationships from 2245 Displacement to Mpg(miles per gallon)? 2246 -Thought: The Affirmative Causal Agent argues that there is no 2247 direct causal relationship between Displacement and Mpg, while 2248 the Negative\_Causal\_Agent agrees. It is important to consider 2249 the various factors that affect fuel efficiency. 2250 -Action: I would like to ask both debaters to provide examples of 2251 these factors and how they interact with Displacement. 2252 -Observation: Waiting for additional information from the debaters 2253 . 2254 # End response of Judge # 2255 2256 \_\_\_\_\_ 2257 Provide feedback to chat\_manager. Press enter to skip and use auto 2258 -reply, or type 'exit' to end the conversation: Affirmative, 2259 please answer the questions and concerns from the Judge of the 2260 question from 1 to 4 in detail, and step-by-step 2261 Admin (to chat\_manager): 2262 2263 Affirmative, please answer the questions and concerns from the Judge of the question from 1 to 4 in detail, and step-by-step 2264 2265 \_\_\_\_\_ 2266 Affirmative\_Causal\_Agent (to chat\_manager): 2267

```
2268
      # Begin response of Affirmative_Causal_Agent #
2269
2270
      Question number 1:
2271
2272
      -Question: Are there any direct causal relationships from
2273
         Displacement to Weight?
      -Thought: The Judge raised a valid point about the need for
2274
         specific studies or data to support our conclusions. It is
2275
         essential to provide more detailed information to clarify the
2276
         relationship between Displacement and Weight.
2277
      -Action: Present empirical evidence from car engineering studies.
2278
      -Action Input: Share research findings on engine Displacement and
2279
         car Weight.
2280
      -Observation: Studies have shown that while engine Displacement
2281
         can indirectly influence the Weight of a car through factors
2282
         like engine size and materials, there is no direct causal
2283
         relationship between Displacement and Weight. Additional
2284
         variables such as chassis design and components play a more
         significant role in determining the overall Weight of a
2285
         vehicle.
2286
2287
      -Thought: Addressing the Judge's concern is crucial to strengthen
2288
         our argument.
2289
      -Action: Provide examples of specific car models or case studies.
2290
      -Action Input: Discuss real-world examples where Displacement and
2291
         Weight do not have a direct causal relationship.
2292
      -Observation: By examining specific car models and case studies,
2293
         it becomes evident that factors other than Displacement, such
2294
         as materials and design choices, have a more substantial
         impact on the Weight of a vehicle.
2295
2296
      -Thought: I now know the final answer.
2297
      -Final answer: No
2298
2299
      End Question number 1
2300
2301
      Question number 2:
2302
2303
      -Question: Are there any direct causal relationships from
2304
         Displacement to Horsepower?
2305
      -Thought: The Judge requested more details on the studies or data
2306
         supporting our claims regarding the relationship between
         Displacement and Horsepower. It is essential to provide
2307
         concrete evidence to validate our argument.
2308
      -Action: Present empirical data on engine performance.
2309
      -Action Input: Share research findings on the impact of engine
2310
         Displacement on Horsepower.
2311
      -Observation: Engine Displacement has a direct causal relationship
2312
          with Horsepower, as evidenced by numerous studies and
2313
         empirical data. Larger engine Displacement generally leads to
2314
         higher Horsepower output.
2315
2316
      -Thought: Providing specific examples can further support our
2317
         argument.
      -Action: Discuss specific engine specifications and performance
2318
         metrics.
2319
      -Action Input: Present case studies where engine Displacement
2320
         directly influences Horsepower.
2321
```

2322 -Observation: By examining specific engine designs and performance 2323 metrics, it is clear that engine Displacement plays a 2324 significant role in determining the Horsepower output. 2325 2326 -Thought: I now know the final answer. -Final answer: Yes 2327 2328 End Question number 2 2329 2330 Question number 3: 2331 2332 -Question: Are there any direct causal relationships from 2333 Displacement to Acceleration? 2334 -Thought: The Judge highlighted the importance of elaborating on 2335 the factors influencing the relationship between Displacement 2336 and Acceleration. It is crucial to provide a detailed 2337 explanation to address this concern. 2338 -Action: Analyze the specific mechanisms through which engine Displacement affects Acceleration. 2339 -Action Input: Discuss the impact of engine size on power delivery 2340 and Acceleration. 2341 -Observation: Engine Displacement directly influences the 2342 Acceleration of a vehicle by providing more power for faster 2343 acceleration. Larger engines typically result in quicker 2344 Acceleration due to increased torque and horsepower. 2345 2346 -Thought: Providing real-world examples can enhance our argument. 2347 -Action: Present case studies or acceleration tests. 2348 -Action Input: Share examples where changes in engine Displacement lead to observable differences in vehicle Acceleration. 2349 -Observation: By examining specific acceleration tests and case 2350 studies, it becomes evident that changes in engine 2351 Displacement directly impact the Acceleration of a vehicle. 2352 2353 -Thought: I now know the final answer. 2354 -Final answer: Yes 2355 2356 End Question number 3 2357 2358 Question number 4: 2359 2360 -Question: Are there any direct causal relationships from Displacement to Mpg(miles per gallon)? 2361 -Thought: The Judge requested examples of factors that affect fuel 2362 efficiency and how they interact with Displacement. It is 2363 essential to provide a detailed explanation to address this 2364 concern. 2365 -Action: Discuss the various factors influencing fuel efficiency. 2366 -Action Input: Explain how engine Displacement interacts with 2367 factors like fuel injection systems and driving conditions. 2368 -Observation: While engine Displacement can impact fuel efficiency 2369 , it is not a direct causal relationship. Other variables such 2370 as engine technology, driving habits, and maintenance play 2371 significant roles in determining Mpg. 2372 -Thought: Providing specific examples can clarify the relationship 2373 between Displacement and Mpg. 2374 -Action: Present case studies or fuel consumption data. 2375

2376 -Action Input: Share examples where changes in engine Displacement 2377 do not lead to proportional changes in Mpg. 2378 -Observation: By examining specific fuel consumption data and case 2379 studies, it becomes evident that Displacement alone is not 2380 the sole determinant of Mpg, as other factors also contribute to fuel efficiency. 2381 2382 -Thought: I now know the final answer. 2383 -Final answer: No 2384 2385 End Question number 4 2386 2387 \_\_\_\_\_ 2388 Provide feedback to chat\_manager. Press enter to skip and use auto 2389 -reply, or type 'exit' to end the conversation: Negative, 2390 please answer the questions and concerns from the Judge of 2391 question from 1 to 4 in detail, and step-by-step 2392 Admin (to chat\_manager): 2393 Negative, please answer the questions and concerns from the Judge 2394 of question from 1 to 4 in detail, and step-by-step 2395 2396 2397 Negative\_Causal\_Agent (to chat\_manager): 2398 2399 # Begin response of Negative\_Causal\_Agent # 2400 2401 Question number 1: 2402 -Question: Are there any direct causal relationships from 2403 Displacement to Weight? -Thought: The Judge rightly pointed out the importance of 2404 providing specific studies or data to support the conclusions 2405 regarding the relationship between Displacement and Weight. It 2406 is crucial to address this concern by delving deeper into the 2407 available evidence. 2408 -Action: Present empirical studies and data on car engineering. 2409 -Action Input: Share research findings that explore the impact of 2410 engine Displacement on car Weight. 2411 -Observation: While engine Displacement can indirectly influence 2412 the Weight of a car through factors like engine size and 2413 materials, there is no direct causal relationship between 2414 Displacement and Weight. Studies show that Weight is determined by various factors beyond just Displacement, such 2415 as chassis design and additional components. 2416 2417 -Thought: Addressing the need for specific evidence is essential 2418 to strengthen the argument. 2419 -Action: Provide examples of real-world scenarios. 2420 -Action Input: Discuss instances where Displacement and Weight do 2421 not have a direct causal relationship in practical car design. 2422 -Observation: By examining specific car models and case studies, 2423 it becomes evident that factors other than Displacement, such 2424 as materials and design choices, have a more significant 2425 impact on the Weight of a vehicle. 2426 -Thought: I now know the final answer. 2427 -Final answer: No 2428 2429 End Question number 1

```
2430
2431
      Question number 2:
2432
      -Question: Are there any direct causal relationships from
2433
         Displacement to Horsepower?
2434
      -Thought: The Judge requested more details on the studies or data
2435
         supporting the claims about the relationship between
         Displacement and Horsepower. It is crucial to provide concrete
2436
          evidence to validate the argument.
2437
      -Action: Present empirical data on engine performance.
2438
      -Action Input: Share research findings on the impact of engine
2439
         Displacement on Horsepower.
2440
      -Observation: While there is a correlation between engine
2441
         Displacement and Horsepower, it is not a direct causal
2442
         relationship. Horsepower is influenced by various factors such
2443
          as engine efficiency, design, and tuning, in addition to
2444
         Displacement.
2445
2446
      -Thought: Providing specific examples can further support the
         argument.
2447
      -Action: Discuss specific engine specifications and performance
2448
         metrics.
2449
      -Action Input: Present case studies where engine Displacement does
2450
          not directly influence Horsepower.
2451
      -Observation: By examining specific engine designs and performance
2452
          metrics, it is clear that engine Displacement alone is not
2453
         the sole determinant of Horsepower output.
2454
2455
      -Thought: I now know the final answer.
2456
      -Final answer: No
2457
      End Question number 2
2458
2459
      Question number 3:
2460
      -Question: Are there any direct causal relationships from
2461
         Displacement to Acceleration?
2462
      -Thought: The Judge emphasized the importance of elaborating on
2463
         the factors influencing the relationship between Displacement
2464
         and Acceleration. It is crucial to provide a detailed
2465
         explanation to address this concern.
2466
      -Action: Analyze the specific mechanisms through which engine
2467
         Displacement affects Acceleration.
2468
      -Action Input: Discuss the impact of engine size on power delivery
          and Acceleration.
2469
      -Observation: While engine Displacement can impact the
2470
         Acceleration of a vehicle, it is not a direct causal
2471
         relationship. Acceleration is influenced by various factors
2472
         such as vehicle weight, transmission, and aerodynamics, in
2473
         addition to engine Displacement.
2474
2475
      -Thought: Providing real-world examples can enhance the argument.
2476
      -Action: Present case studies or acceleration tests.
2477
      -Action Input: Share examples where changes in engine Displacement
2478
          do not lead to proportional changes in vehicle Acceleration.
2479
      -Observation: By examining specific acceleration tests and case
         studies, it becomes evident that changes in engine
2480
         Displacement alone do not always directly impact the
2481
         Acceleration of a vehicle.
2482
2483
      -Thought: I now know the final answer.
```

2484 -Final answer: No 2485 2486 End Question number 3 2487 2488 Question number 4: -Question: Are there any direct causal relationships from 2489 Displacement to Mpg(miles per gallon)? 2490 -Thought: The Judge requested examples of factors that affect fuel 2491 efficiency and how they interact with Displacement. It is 2492 essential to provide a detailed explanation to address this 2493 concern. 2494 -Action: Discuss the various factors influencing fuel efficiency. 2495 -Action Input: Explain how engine Displacement interacts with 2496 factors like fuel injection systems and driving conditions. 2497 -Observation: While engine Displacement can impact fuel efficiency 2498 , it is not a direct causal relationship. Other variables such 2499 as engine technology, driving habits, and maintenance play 2500 significant roles in determining Mpg. 2501 -Thought: Providing specific examples can clarify the relationship 2502 between Displacement and Mpg. 2503 -Action: Present case studies or fuel consumption data. 2504 -Action Input: Share examples where changes in engine Displacement 2505 do not lead to proportional changes in Mpg. 2506 -Observation: By examining specific fuel consumption data and case 2507 studies, it becomes evident that Displacement alone is not 2508 the sole determinant of Mpg, as other factors also contribute 2509 to fuel efficiency. 2510 -Thought: I now know the final answer. 2511 -Final answer: No 2512 2513 End Question number 4 2514 2515 2516 Provide feedback to chat manager. Press enter to skip and use auto 2517 -reply, or type 'exit' to end the conversation: Judge, after 2518 receiving the responses from both sides, please make a final 2519 decision for each question Admin (to chat\_manager): 2521 2522 Judge, after receiving the responses from both sides, please make a final decision for each question 2523 2524 2525 Judge (to chat\_manager): 2526 2527 # Begin response of Judge # 2528 2529 Question number 1: 2530 -Question: Are there any direct causal relationships from 2531 Displacement to Weight? 2532 -Thought: Both the Affirmative\_Causal\_Agent and the 2533 Negative\_Causal\_Agent provided detailed explanations regarding the relationship between Displacement and Weight. They both 2534 highlighted that while engine Displacement can indirectly 2535 influence the Weight of a car, there is no direct causal 2536 relationship. The Affirmative\_Causal\_Agent supported their 2537

2538 argument with empirical evidence and real-world examples, 2539 strengthening their position. 2540 -Final Answer: No direct causal relationship from Displacement to 2541 Weight. 2542 Question number 2: 2543 -Question: Are there any direct causal relationships from 2544 Displacement to Horsepower? 2545 -Thought: The Affirmative\_Causal\_Agent argued that there is a 2546 direct causal relationship between Displacement and Horsepower 2547 , supported by empirical data and specific engine 2548 specifications. In contrast, the Negative\_Causal\_Agent 2549 disagreed, stating that while there is a correlation, it is 2550 not a direct causal relationship. However, the 2551 Affirmative\_Causal\_Agent's evidence and examples provided a 2552 more convincing argument. 2553 -Final Answer: Yes, there is a direct causal relationship from 2554 Displacement to Horsepower. 2555 Question number 3: 2556 -Question: Are there any direct causal relationships from 2557 Displacement to Acceleration? 2558 -Thought: The Affirmative\_Causal\_Agent claimed a direct causal 2559 relationship between Displacement and Acceleration, supported 2560 by the impact of engine size on power delivery. The 2561 Negative\_Causal\_Agent disagreed, stating that while 2562 Displacement can impact Acceleration, it is not a direct 2563 causal relationship. The Affirmative\_Causal\_Agent's detailed 2564 explanation and real-world examples provided a stronger 2565 argument. -Final Answer: Yes, there is a direct causal relationship from 2566 Displacement to Acceleration. 2567 2568 Question number 4: 2569 -Question: Are there any direct causal relationships from 2570 Displacement to Mpg(miles per gallon)? 2571 -Thought: Both sides agreed that there is no direct causal 2572 relationship between Displacement and Mpg. The 2573 Affirmative\_Causal\_Agent highlighted the influence of other 2574 factors such as engine technology and driving habits on fuel 2575 efficiency, while the Negative\_Causal\_Agent also supported 2576 this conclusion. Both sides provided examples and explanations that supported the lack of direct causality. 2577 -Final Answer: No direct causal relationship from Displacement to 2578 Mpg(miles per gallon). 2579 2580 # End response of Judge # 2581 2582 2583

B.0.2 CODING AGENTS MODEL

2587 Phase 1: Plan Debating

2584 2585

2586

2588

2589

2590 2591 Listing 7: Plan Debating responses

Admin (to chat\_manager):

2592 Task: find the causal relationship among variables in the 2593 dataset 2594 Here are some information about the dataset: on automobile 2595 design and performance for cars in the 1970s and early 1980s, 2596 this dataset consists of the variables around the fuel 2597 consumption of cars. With five variables: Weight, Displacement, Horsepower, Acceleration and Mpg(miles per 2598 gallon)s 2599 There is some sample of the data : Displacement Mpg 2600 Horsepower Weight Acceleration 2601 0 307.0 18.0 130 3504 12.0 2602 1 350.0 15.0 165 3693 11.5 2603 318.0 18.0 2 150 3436 11.0 2604 304.0 16.0 3 150 3433 12.0 2605 302.0 17.0 4 140 3449 10.5 2606 Don't need to understand the data, just give a plan of how to 2607 find the causal relationship among variables 2608 The required output is a human-readable causal graph should be in a matrix for example with 3 variables: 2609 [[0, 0, 1], [0, 0, 1], [1,0, 0]] and link format (Hot whether 2610 -> More ice scream sale, More people-> increase ice scream 2611 sale, increase ice scream sale ->Hot whether). If the size is 2612 5 use 5x5 matrix and so on 2613 2614 2615 2616 Affirmative\_Causal\_Agent (to chat\_manager): 2617 2618 -Question: How can we find the causal relationship among variables in the dataset? 2619 -Thought: We need to choose an appropriate causal inference 2620 algorithm that can handle observational data and construct a 2621 causal graph representing the relationships among the 2622 variables. 2623 -Action: Choose the appropriate causal inference algorithm for the 2624 dataset 2625 -Action Input: Evaluate the pros and cons of each algorithm based 2626 on the dataset information 2627 -Observation: Each algorithm has unique features that make it 2628 suitable for different types of data and research questions. 2629 -Thought: Considering the dataset consists of observational data 2630 on automobile design and performance variables, we need an algorithm that can handle observational data and construct a 2631 causal graph efficiently. 2632 -Action: Select the algorithm that best fits the requirements of 2633 the dataset 2634 -Action Input: Analyze the pros and cons of each algorithm in the 2635 context of the dataset 2636 -Observation: Each algorithm has advantages and limitations that 2637 need to be considered for causal inference in this specific 2638 dataset. 2639 -Thought: Based on the dataset characteristics and the need to 2640 construct a causal graph among the variables, we should choose 2641 an algorithm that can handle observational data efficiently and provide a clear causal structure. 2642 -Action: Propose the appropriate algorithm for causal inference in 2643 this dataset 2644 2645

2646 -Action Input: Consider the strengths and limitations of each 2647 algorithm and select the one that aligns best with the dataset 2648 requirements 2649 -Observation: The selected algorithm should be able to handle the 2650 variables related to automobile design and performance 2651 effectively to uncover the causal relationships. -Thought: I now know the final answer 2652 -Final Answer: The PC algorithm is the most appropriate choice for 2653 causal inference in this dataset as it can handle 2654 observational data efficiently and construct a causal network 2655 representing the relationships among the variables. 2656 2657 2658 Negative\_Causal\_Agent (to chat\_manager): 2659 2660 -Question: How can we find the causal relationship among variables 2661 in the dataset? 2662 -Thought: We need to critically evaluate the proposed algorithm choice and provide a different perspective to ensure a 2663 comprehensive analysis. 2664 -Action: Disagree with the previous answer and provide an 2665 alternative viewpoint. 2666 -Action Input: Highlight the limitations of the PC algorithm and 2667 propose a different algorithm for causal inference in the 2668 dataset. 2669 -Observation: By presenting a different algorithm choice, we can 2670 enrich the discussion and explore alternative approaches to 2671 causal inference. 2672 -Thought: It's essential to consider the dataset's characteristics 2673 and the specific requirements for causal inference to provide a well-rounded recommendation. 2674 -Action: Analyze the pros and cons of another algorithm for causal 2675 inference. 2676 -Action Input: Evaluate the strengths and limitations of the GES 2677 algorithm in the context of the dataset information. 2678 -Observation: By considering a different algorithm, we can offer a 2679 diverse perspective on the most suitable approach for 2680 uncovering causal relationships in the dataset. 2681 -Thought: After a thorough analysis of the dataset and the 2682 available causal inference algorithms, it's crucial to present 2683 a well-justified recommendation. 2684 -Action: Propose the alternative algorithm for causal inference in this dataset. 2685 -Action Input: Highlight how the GES algorithm can address the 2686 specific characteristics of the dataset and provide valuable 2687 insights into causal relationships. 2688 -Observation: By presenting a different algorithm choice, we can 2689 broaden the understanding of causal inference methods and 2690 their applicability to the given dataset. 2691 -Thought: I now know the final answer 2692 -Final Answer: The GES algorithm is a suitable choice for causal 2693 inference in this dataset as it offers efficient handling of 2694 large datasets and provides scalability, which is crucial for 2695 analyzing variables related to automobile design and 2696 performance. 2697 \_\_\_\_\_ 2698 Judge (to chat\_manager): 2699

2700 -Question: What is the best approach to finding the causal 2701 relationship among variables in the dataset? 2702 -Thought: Both debaters have presented different perspectives on 2703 the choice of causal inference algorithm. It's essential to 2704 evaluate the arguments and determine the most suitable approach for uncovering causal relationships in the given 2705 dataset. 2706 -Action: Review the arguments presented by both debaters and 2707 assess the strengths and limitations of each proposed 2708 algorithm. 2709 -Action Input: Examine the characteristics of the dataset and the 2710 requirements for causal inference. 2711 -Observation: By comparing the pros and cons of the PC algorithm 2712 and the GES algorithm, we can determine which approach aligns 2713 best with the dataset's needs. 2714 -Thought: It's crucial to prioritize accuracy, efficiency, and 2715 scalability in selecting the causal inference algorithm for 2716 this dataset. -Action: Evaluate the applicability of each algorithm in handling 2717 observational data related to automobile design and 2718 performance variables. 2719 -Action Input: Consider how each algorithm constructs causal 2720 graphs and represents relationships among the variables. 2721 -Observation: By analyzing the capabilities of the PC algorithm 2722 and the GES algorithm, we can determine which one is better 2723 suited for uncovering causal relationships in the dataset. 2724 -Thought: After a comprehensive review of the arguments and 2725 algorithm characteristics, it's time to make a decision based 2726 on the dataset's requirements. -Final Answer: The GES algorithm is the most suitable choice for 2727 causal inference in this dataset as it offers scalability and 2728 efficiency in handling variables related to automobile design 2729 and performance, providing valuable insights into causal 2730 relationships. 2731 2732 2733 Provide feedback to chat\_manager. Press enter to skip and use auto 2734 -reply, or type 'exit' to end the conversation: Negative, 2735 please give step-by-step plan for the coder to implement, 2736 remember the output requirements, output in the format: step 2737 1: ..., step 2:..., step N:... 2738 Admin (to chat\_manager): 2739 Negative, please give step-by-step plan for the coder to implement 2740 , remember the output requirements, output in the format: step 2741 1: ..., step 2:..., step N:... 2742 2743 2744 Negative Causal Agent (to chat manager): 2745 2746 -Question: What is the step-by-step plan for implementing the GES 2747 algorithm for causal inference in the dataset? 2748 -Thought: To provide a detailed plan for the coder, we need to 2749 outline the specific steps involved in implementing the GES algorithm effectively. 2750 -Action: Develop a comprehensive step-by-step plan for coding the 2751 GES algorithm. 2752 -Action Input: Consider the requirements of the dataset and the 2753 algorithm's characteristics.

2754	-Step 1: Preprocessing
2755	- Perform data preprocessing to handle missing values,
2756	normalize the data if necessary, and ensure data quality.
2757	-Step 2: Implement GES Algorithm Initialization
2758	- Set up the initial conditions for the GES algorithm,
2759	including defining the search space and initializing the score
2760	function.
2761	-Step 3: Score Calculation
2762	- Calculate the scores for potential causal relationships using
2763	the BIC score function to evaluate the goodness of fit and
2764	model complexity.
2765	-Step 4: Greedy Equivalence Search
2766	- Implement the Greedy Equivalence Search algorithm to explore
2767	causal structure
2768	-Step 5. Model Evaluation
2769	- Evaluate the identified causal structure and assess its
2770	interpretability and relevance to the dataset.
2771	-Step 6: Output Generation
2772	- Generate the causal graph representing the relationships
2773	among the variables in a human-readable format as specified,
2774	linking variables based on the causal connections.
2775	-Step 7: Interpretation
2776	- Interpret the causal graph to understand the causal
2777	relationships among the variables and their implications for
2778	automobile design and performance.
2779	-Step 8: Documentation
2780	approximation and any insights approximately and any insights
2781	-Step 9. Testing and Validation
2701	- Test the implemented GES algorithm on the dataset, validate
2782	the results, and ensure the accuracy of the causal graph
2784	generated.
2704	-Step 10: Optimization
2705	- Optimize the algorithm implementation for efficiency and
2700	scalability, considering the size and complexity of the
2707	dataset.
2700	-Thought: By providing a detailed step-by-step plan, the coder can
2709	effectively implement the GES algorithm for causal inference
2790	in the dataset.
2791	-Final Answer: The step-by-step plan for coding the GES algorithm
2192	algorithm initialization score calculation. Creedy
2793	Fauivalence Search model evaluation output generation
2794	interpretation, documentation, testing and validation and
2795	optimization steps to ensure accurate and insightful causal
2796	relationships among the variables related to automobile design
2797	and performance.
2798	
2799	
2800	

2800 2801 2802

Phase 2: Code Executing

2803

Listing 8: Code executing responses

ask: find the causal relationship among variables in the dataset Here is the dataset's directory: benchmark datasets/DWD climate data.csv

2808 2809 2810 2811 2812 2813	Here are some information about the dataset: on global climate variables, This dataset encompasses six continuous variables capturing climate observations such as altitude, temperature, precipitation levels, longitude, sunshine duration, and latitude. It is aimed at studying weather patterns, climate change impacts, and geographical correlations in climate
2814 2815 2816 2817	variables. Here is step-by-step plan for implement the code to find the causal relationship among variables:
2818 2819 2820 2821 2822	<ol> <li>**Data Preparation**:         <ul> <li>Load the dataset containing variables: Weight, Displacement, Horsepower, Acceleration, Mpg.</li> <li>Handle missing values and ensure data cleanliness.</li> </ul> </li> </ol>
2823 2824 2825 2826	<ul> <li>2. **Graph Initialization**:</li> <li>- Create an initial fully connected, undirected graph with all variables connected to each other.</li> </ul>
2827 2828 2829 2830 2831	<ul> <li>3. **Conditional Independence Testing**:</li> <li>Implement statistical tests (e.g., chi-squared tests, correlation tests) to check for conditional independence between variable pairs given a set of other variables.</li> <li>Remove edges where conditional independence is observed.</li> </ul>
2832 2833 2834 2835 2836 2837	<ul> <li>4. **Orientation Rules**:</li> <li>Apply orientation rules to determine the directionality of edges based on conditional independence patterns.</li> <li>Convert the undirected graph into a directed acyclic graph (DAG).</li> </ul>
2838 2839 2840 2841 2842	<ul> <li>5. **Iteration**:</li> <li>Iterate through increasing sizes of conditioning sets (empty set, singletons, pairs, etc.) to refine the graph structure.</li> <li>Remove edges based on conditional independence tests until no more edges can be deleted.</li> </ul>
2843 2844 2845 2846 2847 2848	<ul> <li>6. **Implementation Steps**:</li> <li>Develop functions or modules for graph construction, conditional independence testing, orientation rules, and iterative edge removal.</li> <li>Utilize appropriate statistical libraries for conducting tests and graph manipulation.</li> </ul>
2849 2850 2851 2852 2853 2854 2855	<ul> <li>7. **Execution and Evaluation**:</li> <li>Run the implemented PC algorithm on the preprocessed dataset.</li> <li>Evaluate the resulting causal graph for meaningful relationships and causal directions.</li> <li>Validate the graph structure and assess the strength of causal connections.</li> </ul>
2856 2857 2858 2859 2860 2861	<ul> <li>8. **Documentation and Reporting**:</li> <li>Document the code implementation, including functions, algorithms used, and any modifications made.</li> <li>Generate a human-readable causal graph matrix based on the algorithm's output.</li> <li>Prepare a detailed report summarizing the causal relationships discovered and their implications.</li> </ul>

2862 2863 2864 The required output is a human-readable causal graph should be 2865 in a matrix for example with 3 variables: [[0, 0, 1], [0, 0, 1], [1,0, 0]] and link format (Hot whether 2867 -> More ice scream sale, More people-> increase ice scream sale, increase ice scream sale ->Hot whether). If the size is 2868 5 use 5x5 matrix and so on 2869 2870 2871 2872 Code\_Writer (to chat\_manager): 2873 2874 -Question: What is the initial step to find the causal 2875 relationship among variables in the dataset? 2876 2877 -Thought: The initial step is to load the dataset and handle 2878 missing values to ensure data cleanliness. 2879 -Action: Load the dataset and handle missing values 2880 2881 -Action Input: 2882 '''python 2883 import pandas as pd 2884 # Load the dataset 2886 data = pd.read\_csv("benchmark datasets/DWD climate data.csv") 2887 2888 # Check for missing values missing\_values = data.isnull().sum() 2889 missing\_values 2890 ... 2891 2892 -Observation: The dataset has been loaded and missing values have 2893 been checked. 2894 2895 -Question: What is the next step after loading the dataset and 2896 handling missing values? 2897 -Thought: The next step is to create an initial fully connected, 2899 undirected graph with all variables connected to each other. 2900 -Action: Create an initial fully connected, undirected graph. 2901 2902 -Action Input: 2903 '''python 2904 # Create an initial fully connected, undirected graph 2905 import numpy as np 2906 2907 num\_variables = len(data.columns) 2908 initial\_graph = np.ones((num\_variables, num\_variables)) 2909 2910 initial\_graph ... 2911 2912 -Observation: The initial fully connected, undirected graph has 2913 been created. 2914 2915

2916 -Question: What is the next step after creating the initial fully 2917 connected, undirected graph? 2918 2919 -Thought: The next step is to implement statistical tests to check 2920 for conditional independence between variable pairs given a set of other variables. 2921 2922 -Action: Implement statistical tests for conditional independence. 2923 2924 -Action Input: Implementing statistical tests for conditional 2925 independence. 2926 2927 -Question: What is the next step after implementing statistical 2928 tests for conditional independence? 2929 2930 -Thought: The next step is to apply orientation rules to determine 2931 the directionality of edges based on conditional independence 2932 patterns and convert the graph into a directed acyclic graph (DAG). 2933 2934 -Action: Apply orientation rules and convert the graph into a DAG. 2935 2936 -Action Input: Applying orientation rules based on conditional 2937 independence patterns. 2938 2939 We will continue with the rest of the steps after applying 2940 orientation rules and converting the graph into a DAG. 2941 2942 2943 >>>>>> EXECUTING 2 CODE BLOCKS (inferred languages are [python, 2944 python])... 2945 Code\_Executor (to chat\_manager): 2946 2947 exitcode: 0 (execution succeeded) 2948 Code output: Altitude 0 2949 Temperature 0 2950 Precipitation 0 2951 0 Longitude 2952 0 Sunshine 2953 Latitude 0 2954 dtype: int64 array([[1., 1., 1., 1., 1.], 2955 [1., 1., 1., 1., 1., 1.], 2956 [1., 1., 1., 1., 1., 1.], 2957 [1., 1., 1., 1., 1., 1.], 2958 [1., 1., 1., 1., 1., 1.],2959 [1., 1., 1., 1., 1., 1.]]) 2960 2961 2962 Code\_Writer (to chat\_manager): 2963 2964 -Question: What is the next step after implementing statistical 2965 tests for conditional independence and handling missing values ? 2966 2967 -Thought: The next step is to apply orientation rules to determine 2968 the directionality of edges based on conditional independence 2969

2970 patterns and convert the graph into a directed acyclic graph 2971 (DAG). 2972 2973 -Action: Apply orientation rules and convert the graph into a DAG. 2974 2975 -Action Input: Applying orientation rules based on conditional independence patterns. 2976 '''python 2977 # We will continue with this step after applying orientation rules 2978 and converting the graph into a DAG 2979 # The next step is to implement the PC Algorithm for causal 2980 relationship discovery. 2981 • • • 2982 2984 2985 >>>>>> EXECUTING CODE BLOCK (inferred language is python)... 2986 Code\_Executor (to chat\_manager): 2987 exitcode: 0 (execution succeeded) 2988 Code output: 2989 2990 \_\_\_\_\_ 2991 Code\_Writer (to chat\_manager): 2992 2993 -Question: What is the next step after applying orientation rules 2994 and converting the graph into a DAG? 2995 2996 -Thought: The next step is to implement the PC Algorithm for causal relationship discovery. 2997 2998 -Action: Implement the PC Algorithm for causal relationship 2999 discovery. 3000 3001 -Action Input: Implementing the PC Algorithm with the custom 3002 parameters if needed. 3003 3004 3005 Code\_Executor (to chat\_manager): 3006 3007 3008 3009 Code\_Writer (to chat\_manager): 3010 3011 -Question: What is the next step after implementing the PC 3012 Algorithm for causal relationship discovery? 3013 3014 -Thought: The next step is to create a human-readable causal graph 3015 matrix based on the algorithm's output. 3016 3017 -Action: Create a human-readable causal graph matrix. 3018 3019 -Action Input: '''python 3020 # Using the provided create\_adjacency\_matrix function to get the 3021 causal graph matrix 3022 def create\_adjacency\_matrix(cg): 3023 num\_nodes = len(cg.G.nodes)

```
3024
         adj_matrix = np.zeros((num_nodes, num_nodes), dtype=int)
3025
3026
         for i in range(num_nodes):
3027
             for j in range(num_nodes):
3028
                 # i <- j
3029
                 if cg.G.graph[i][j] == 1 and cg.G.graph[j][i] == -1:
                     adj_matrix[i, j] = 1
3030
                 # i -- j
3031
                 elif cq.G.graph[i][j] == -1 and cq.G.graph[j][i] == -1:
3032
                     adj_matrix[i, j] = -1
3033
                 # i <->
3034
                 elif cg.G.graph[i][j] == 1 and cg.G.graph[j][i] == 1:
3035
                     adj_matrix[i, j] = 2
3036
         return adj_matrix
3037
3038
      # Get adjacency matrix from the PC Algorithm output
3039
      total_adj_matrix_pc = create_adjacency_matrix(cg)
3040
      total_adj_matrix_pc
3041
3042
3043
      This will provide a human-readable causal graph matrix based on
3044
         the output of the PC Algorithm.
3045
3046
3047
3048
      >>>>>> EXECUTING CODE BLOCK (inferred language is python)...
3049
      Code_Executor (to chat_manager):
3050
      exitcode: 1 (execution failed)
3051
      Code output: ERROR: ERROR: NameError: name 'cg' is not defined
3052
      ['\x1b[0;31m------
3053
         create_adjacency_matrix(\x1b[43mcg\x1b[49m)\n\x1b[1;32m
3054
         22\x1b[0m total_adj_matrix_pc\n', "\x1b[0;31mNameError\x1b[0m:
3055
          name 'cg' is not defined"]
3056
3057
       _____
3058
      Code_Writer (to chat_manager):
3059
3060
      -Question: What can be done to resolve the NameError and
3061
         successfully create a human-readable causal graph matrix?
3062
      -Thought: We need to ensure that the variable 'cg', which holds
3063
         the output of the PC Algorithm, is properly defined before
3064
         calling the function to create the adjacency matrix.
3065
3066
      -Action: Execute the PC Algorithm to obtain the graph object 'cg'
3067
         before creating the adjacency matrix.
3068
3069
      -Action Input:
3070
      '''python
3071
      from causallearn.search.ConstraintBased.PC import pc
3072
      # Assuming 'data' is prepared and available
3073
      # Execute the PC Algorithm to obtain the causal graph 'cg'
3074
      cg = pc(data)
3075
3076
      # Function to create the adjacency matrix
3077
     def create_adjacency_matrix(cg):
```

```
3078
         num_nodes = len(cg.G.nodes)
3079
         adj_matrix = np.zeros((num_nodes, num_nodes), dtype=int)
3080
3081
         for i in range(num_nodes):
3082
             for j in range(num_nodes):
                  # i <- j
3083
                 if cg.G.graph[i][j] == 1 and cg.G.graph[j][i] == -1:
3084
                     adj_matrix[i, j] = 1
3085
                 # i -- j
3086
                 elif cg.G.graph[i][j] == -1 and cg.G.graph[j][i] == -1:
3087
                     adj_matrix[i, j] = -1
3088
                  # i <->
3089
                 elif cg.G.graph[i][j] == 1 and cg.G.graph[j][i] == 1:
3090
                     adj_matrix[i, j] = 2
3091
         return adj_matrix
3092
3093
      # Obtain the adjacency matrix from the PC Algorithm output
      total_adj_matrix_pc = create_adjacency_matrix(cg)
3094
3095
      total_adj_matrix_pc
3096
      • • •
3097
3098
            _____
3099
3100
      >>>>>> EXECUTING CODE BLOCK (inferred language is python)...
3101
      Code_Executor (to chat_manager):
3102
3103
      exitcode: 1 (execution failed)
3104
      Code output: ERROR: ERROR: AssertionError: Input data must be a
3105
         numpy array.
                               -----24mutf-8\x1b[39m\x1b]
      ['\x1b[0;31m-----
3106
          [38;5;124m\'\x1b[39m))\x1b[38;5;241m.\x1b[39mhexdigest()\n',
3107
          /\x1b[0;31mAssertionError\x1b[0m: Input data must be a numpy
3108
         array.']
3109
3110
3111
      Code_Writer (to chat_manager):
3112
3113
      -Question: How can we resolve the Assertion Error regarding the
3114
          input data format in order to successfully create the
3115
         adjacency matrix for the causal graph?
3116
      -Thought: The error indicates that the input data must be in the
3117
          format of a NumPy array. We need to convert the data into a
3118
         NumPy array before using it in the PC Algorithm.
3119
3120
      -Action: Convert the dataset to a NumPy array before executing the
3121
          PC Algorithm.
3122
3123
      -Action Input:
3124
      '''python
3125
      import numpy as np
3126
3127
      # Convert the Pandas DataFrame to a NumPy array
      data_np = data.to_numpy()
3128
3129
      # Execute the PC Algorithm with the converted NumPy array data
3130
      cg = pc(data_np)
3131
```

```
3132
      # Function to create the adjacency matrix
3133
      def create_adjacency_matrix(cg):
3134
         num_nodes = len(cg.G.nodes)
3135
         adj_matrix = np.zeros((num_nodes, num_nodes), dtype=int)
3136
3137
         for i in range(num_nodes):
             for j in range(num_nodes):
3138
                  # i <- j
3139
                  if cq.G.graph[i][j] == 1 and cq.G.graph[j][i] == -1:
3140
                      adj_matrix[i, j] = 1
3141
                  # i -- j
3142
                 elif cg.G.graph[i][j] == -1 and cg.G.graph[j][i] == -1:
3143
                      adj_matrix[i, j] = -1
3144
                  # i <->
3145
                  elif cg.G.graph[i][j] == 1 and cg.G.graph[j][i] == 1:
3146
                      adj_matrix[i, j] = 2
3147
         return adj_matrix
3148
      # Obtain the adjacency matrix from the PC Algorithm output
3149
      total_adj_matrix_pc = create_adjacency_matrix(cg)
3150
3151
      total_adj_matrix_pc
3152
      ...
3153
3154
      By converting the dataset to a NumPy array before executing the PC
3155
          Algorithm, the Assertion Error regarding the input data
3156
          format can be resolved.
3157
3158
3159
      >>>>>> EXECUTING CODE BLOCK (inferred language is python)...
3160
      Code_Executor (to chat_manager):
3161
3162
      exitcode: 0 (execution succeeded)
3163
      Code output: 0%|
                                   | 0/6 [00:00<?, ?it/s]
3164
      {"version major": 2, "version minor": 0, "model id": "
3165
         a1056c8cce5046d49d2ba46d506c5bc8"}
3166
      array([[ 0, 1, 0, 0, 0, 1],
3167
                      Ο,
                           Ο,
                              Ο,
            [0, 0,
                                    0],
3168
            [ 1,
                               Ο,
                   Ο,
                      Ο,
                           Ο,
                                    0],
3169
                              1,
            [ 0,
                  1,
                       Ο,
                           Ο,
                                   01,
                              0, -1],
3170
            [ 0,
                  Ο,
                      Ο,
                           Ο,
            [ 0,
                      Ο,
                          0, -1,
                  Ο,
                                   011)
3171
3172
3173
      Code_Writer (to chat_manager):
3174
3175
      -Question: What is the adjacency matrix for the human-readable
3176
         causal graph based on the output of the PC Algorithm?
3177
3178
      -Observation: The adjacency matrix for the human-readable causal
3179
         graph based on the output of the PC Algorithm is as follows:
      • • •
3180
      [[ 0, 1, 0,
                             1],
3181
                    Ο,
                        Ο,
      [ 0,
            Ο,
                Ο,
                    Ο,
                         Ο,
                             0],
3182
            Ο,
                Ο,
                     Ο,
                         Ο,
      [ 1,
                             01,
3183
                    Ο,
      [ 0,
           1,
                Ο,
                        1,
                             0],
3184
                    Ο,
           Ο,
                        0, -1],
      [ 0,
                Ο,
3185
            Ο,
                Ο,
      [ 0,
                    0, -1,
                             0]]
```

3186 .... 3187 3188 This adjacency matrix represents the causal relationships among 3189 the variables: Altitude, Temperature, Precipitation, Longitude 3190 , Sunshine, and Latitude. The numbers in the matrix indicate 3191 the directionality of the edges between the variables. 3192 \_\_\_\_\_ 3193 3194 >>>>>> EXECUTING CODE BLOCK (inferred language is python)... 3195 Code\_Executor (to chat\_manager): 3196 3197 exitcode: 0 (execution succeeded) 3198 Code output: [[0, 1, 0, 0, 0, 1], 3199 [0, 0, 0, 0, 0, 0],3200 [1, 0, 0, 0, 0, 0], 3201 [0, 1, 0, 0, 1, 0], 3202 [0, 0, 0, 0, 0, -1],[0, 0, 0, 0, -1, 0]]3203 3204 3205 Code\_Writer (to chat\_manager): 3206 3207 -Question: What is the adjacency matrix for the human-readable 3208 causal graph based on the output of the PC Algorithm? 3209 3210 -Final Answer: The adjacency matrix for the human-readable causal 3211 graph based on the output of the PC Algorithm is: • • • 3212 3213 [[0, 1, 0, 0, 0, 1],[0, 0, 0, 0, 0, 0],3214 [1, 0, 0, 0, 0, 0], 3215 [0, 1, 0, 0, 1, 0], 3216 [0, 0, 0, 0, 0, -1],3217 [0, 0, 0, 0, -1, 0]]3218 ... 3219 3220 This matrix represents the causal relationships among the 3221 variables: Altitude, Temperature, Precipitation, Longitude, 3222 Sunshine, and Latitude. 3223 3224 \_\_\_\_\_ 3225 >>>>>> EXECUTING CODE BLOCK (inferred language is python)... 3226 Code\_Executor (to chat\_manager): 3227 3228 exitcode: 0 (execution succeeded) 3229 Code output: [[0, 1, 0, 0, 0, 1], 3230 [0, 0, 0, 0, 0, 0],3231 [1, 0, 0, 0, 0, 0], 3232 [0, 1, 0, 0, 1, 0], 3233 [0, 0, 0, 0, 0, -1],3234 [0, 0, 0, 0, -1, 0]]3235 3236 \_\_\_\_\_ Code\_Writer (to chat\_manager): 3237 3238 3239

3240 Final Answer: Based on the output of the PC Algorithm on the 3241 provided climate dataset, the human-readable causal graph 3242 matrix is as follows: • • • 3243 3244 [[0, 1, 0, 0, 0, 1],[0, 0, 0, 0, 0, 0],3245 [1, 0, 0, 0, 0, 0],3246 [0, 1, 0, 0, 1, 0],3247 [0, 0, 0, 0, 0, -1],3248 [0, 0, 0, 0, -1, 0]]3249 1 1 1 3250 3251 This matrix indicates the causal relationships among the variables 3252 in the dataset: Altitude, Temperature, Precipitation, 3253 Longitude, Sunshine, and Latitude. 3254 3255 3256 >>>>>> EXECUTING CODE BLOCK (inferred language is python)... 3257 Code\_Executor (to chat\_manager): 3258 3259 exitcode: 0 (execution succeeded) 3260 Code output: [[0, 1, 0, 0, 0, 1], 3261 [0, 0, 0, 0, 0, 0], 3262 [1, 0, 0, 0, 0, 0], 3263 [0, 1, 0, 0, 1, 0], 3264 [0, 0, 0, 0, 0, -1],3265 [0, 0, 0, 0, -1, 0]]3266 3267 \_\_\_\_\_ Code\_Writer (to chat\_manager): 3268 3269 Final Results: 3270 - The adjacency matrix for the human-readable causal graph based 3271 on the output of the PC Algorithm is: 3272 • • • 3273 [[0, 1, 0, 0, 0, 1],3274 [0, 0, 0, 0, 0, 0], 3275 [1, 0, 0, 0, 0, 0], 3276 [0, 1, 0, 0, 1, 0],3277 [0, 0, 0, 0, 0, -1],3278 [0, 0, 0, 0, -1, 0]]• • • 3279 3280 This matrix represents the causal relationships among the 3281 variables Altitude, Temperature, Precipitation, Longitude, 3282 Sunshine, and Latitude in the climate dataset. The numbers 3283 indicate the directionality of the causal edges in the graph. 3284 3285 FINISH 3286 3287 \_\_\_\_\_ 3288 Admin (to chat\_manager): 3289 export the matrix into csv file call "Auto\_Coding\_MAD\_React.csv" 3290 3291 \_\_\_\_\_ 3292 Code\_Writer (to chat\_manager): 3293

3294 '''python 3295 import pandas as pd 3296 3297 # Create the adjacency matrix as a Pandas DataFrame 3298 adj\_matrix\_data = [[0, 1, 0, 0, 0, 1], [0, 0, 0, 0, 0, 0], 3299 [1, 0, 0, 0, 0, 0],3300 [0, 1, 0, 0, 1, 0],3301 [0, 0, 0, 0, 0, -1],3302 [0, 0, 0, 0, -1, 0]]3303 3304 columns = ['Altitude', 'Temperature', 'Precipitation', 'Longitude 3305 ', 'Sunshine', 'Latitude'] 3306 3307 adj\_matrix\_df = pd.DataFrame(adj\_matrix\_data, columns=columns, 3308 index=columns) 3309 3310 # Export the adjacency matrix to a CSV file adj\_matrix\_df.to\_csv("Auto\_Coding\_MAD\_React.csv") 3311 3312 3313 \_\_\_\_\_ 3314 3315 >>>>>> EXECUTING CODE BLOCK (inferred language is python)... 3316 Code\_Executor (to chat\_manager): 3317 3318 exitcode: 0 (execution succeeded) 3319 Code output: 3320 3321 \_\_\_\_\_ Code\_Writer (to chat\_manager): 3322 3323 The adjacency matrix representing the causal relationships among 3324 the variables Altitude, Temperature, Precipitation, Longitude, 3325 Sunshine, and Latitude has been successfully exported to a 3326 CSV file named "Auto\_Coding\_MAD\_React.csv". 3327 3328 \_\_\_\_\_ 3329 Code\_Executor (to chat\_manager): 3330 3331 3332 3333 Code\_Writer (to chat\_manager): 3334 3335 I have successfully completed all the steps to find the causal 3336 relationship among variables in the climate dataset: 3337 1. Data Preparation: Loaded the dataset and handled missing values 3338 3339 2. Graph Initialization: Created an initial fully connected, 3340 undirected graph. 3341 3. Conditional Independence Testing: Implemented statistical tests 3342 to check for conditional independence between variable pairs. 3343 4. Orientation Rules: Applied orientation rules to determine the directionality of edges based on conditional independence 3344 patterns and converted the graph into a directed acyclic graph 3345 (DAG). 3346 3347

5. Implemented the PC Algorithm for causal relationship discovery and obtained the adjacency matrix for the human-readable causal graph. The adjacency matrix has been exported to a CSV file named " Auto\_Coding\_MAD\_React.csv". I have provided a comprehensive solution to find the causal relationships in the climate dataset. If you have any more tasks or questions, feel free to let me know! \_\_\_\_\_ Code\_Executor (to chat\_manager): \_\_\_\_\_ Code\_Writer (to chat\_manager): FINISH \_\_\_\_\_