DEVELOPING NICHE PERSONA FOR PERSONALIZED DECISION-MAKING THROUGH SYMBOLIC LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Ecological niches illustrate the close relationship between the information organisms perceive and the behavioral strategies they employ. A similar interdependence arises in modeling human decisions across heterogeneous populations, which requires determining both which information is relevant and how that information should be used. The usefulness of a feature set depends on the decision model, while the model's effectiveness depends on the information it receives. This paper captures this interdependence through the concept of *decision niches*. We propose **Niche Persona from Observation** (NPO), a theory-guided LLM-agent framework that jointly conducts feature selection and human decision modeling. NPO defines a decision niche as the alignment between three elements: the population being modeled, the information exposed to the model, and the behavioral theory used to structure reasoning. For each group, theory-guided decision agents learn choices under candidate behavioral assumptions, while selection agents refine the visible feature space using validation feedback and error patterns. We evaluate NPO on three real-world datasets covering job choice, marijuana behavior, and alcohol behavior. Across all LLM backbones, NPO achieves average relative F1 gains of 9.38%–10.24% over the strongest baseline for each dataset, while using up to 43.75% fewer features than the strongest LLM-based baseline. Beyond prediction, NPO uncovers task- and group-specific decision niches, such as the stronger role of structured trade-offs in job choice and distinct theory–feature alignments across health-related behaviors. With NPO, personalized decision agents achieve stronger prediction with less information, select group-specific behavioral theories, and lay the foundation for future decision companions. The source code will be fully released upon acceptance.

1 INTRODUCTION

Perception and decision-making are interdependent: the information an organism can perceive shapes its actions, while the demands of those actions influence which perceptual capabilities are advantageous over evolutionary timescales (Endler, 1992). As illustrated in Figure 1, ecological niches provide a useful perspective on this relationship. For human decision modeling, this interdependence motivates a central premise: the information supplied to a model should be considered together with the behavioral assumptions used to interpret it. A feature's relevance may depend on the decision strategy, while a strategy's effectiveness may depend on the available information. Both may also vary across population groups. We therefore adopt a *decision-niche* perspective, in which a niche is a group-specific configuration of selected features and behavioral theory.

Classical behavioral models already reflect aspects of this relationship. They specify how attributes enter utility or comparison functions and represent decision patterns such as alternative substitution, preference heterogeneity, and regret-based comparison (Von Neumann & Morgenstern, 2007; Train, 2000; McFadden, 1972; Chorus, 2010). Machine learning models offer greater predictive flexibility by learning complex relationships from data, but typically operate within a predefined input space (Breiman, 2001; Chen & Guestrin, 2016; Arik & Pfister, 2021). Feature selection methods refine this space by evaluating feature subsets against predictive performance or introducing regularization to reduce the number of included variables (Kohavi & John, 1997; Tibshirani, 1996). These approaches provide foundations for modeling decisions and identifying relevant information. However, predictive relevance alone does not establish how selected information should be inter-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

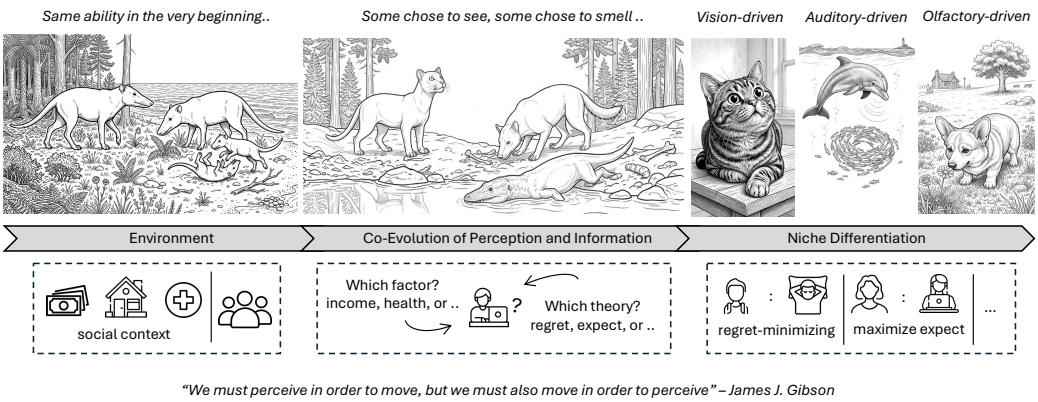

Figure 1: **Overview of Niche Differentiation.** From the perspective of ecology, niche differentiation results from the co-evolution of the decision and perception during the evolutionary process. In NPO, the decision niches also result from the alignment between information and decision strategy.

preted behaviorally or how that interpretation should vary across groups. Addressing this challenge requires a process in which behavioral assumptions guide feature selection and evidence from the selected information informs subsequent decision reasoning.

Large language models (LLMs) offer an opportunity to implement this reciprocal process. Their ability to interpret heterogeneous contextual descriptions and reorganize information makes them promising tools for contextual feature selection (Wu et al., 2025; Zhao et al., 2025a). They can also integrate textual information into decision reasoning across scenarios (Zhao et al., 2025b; Yang et al., 2026). Textual feedback methods, including prompt revision and textual gradient-based optimization, further allow these processes to be refined iteratively (Yuksekgonul et al., 2025). Yet this flexibility does not ensure behavioral validity: LLMs may select spurious or redundant cues, and self-correction without proper feedback may reinforce model-specific biases (Shi et al., 2023; Huang et al., 2024). The challenge is therefore to ground this process in behavioral theory, coordinate its updates, and accommodate differences across population groups. We organize this challenge into three research questions: *1) Behavioral grounding:* How can behavioral theories guide LLM-based feature selection and decision reasoning? *2) Joint refinement:* How can feature selection and decision reasoning improve one another through iterative feedback? *3) Group heterogeneity:* How can this joint refinement capture behavioral differences across population groups?

Inspired by ecological niche differentiation (Zuppinger-Dingley et al., 2014; MacKinnon & MacKinnon, 1980) and bounded rationality (Simon, 1990), we propose Niche Persona from Observation (NPO), a theory-guided LLM-based framework for joint feature selection and personalized decision modeling. The central idea is that a personalized decision niche should not be defined only by a group label, a fixed feature set, or a single decision theory. Instead, it should emerge from the alignment among three elements: the person being modeled, the information selected for reasoning, and the behavioral theory that structures the decision process. To achieve this, NPO jointly evolves theory-guided reasoning and active feature selection. In the reasoning stage, candidate decision theories instantiate symbolic utility functions and domain knowledge to guide the interpretable decision reasoning. In the selection stage, the system collects signals from utility structures, validation performance, and error cases to propose additions to the feature subset for each demographic group. Through this iterative process, perception-like feature selection and decision-theoretic reasoning are no longer treated as separate modules; instead, they jointly iterate toward behaviorally meaningful decision niches. Based on the comprehensive evaluation on three real-world datasets from job choice to marijuana and alcohol misuse, we found that:

1. Across three real-world decision datasets, NPO consistently outperforms the strongest baseline, with average gains ranging from 9.38% to 10.24% in F1 and from 8.61% to 12.66% in accuracy.
2. Beyond performance gains, NPO uses up to 43.75% fewer features across datasets, showing that better group-specific decisions do not require more information, but better theory-information alignment.

3. The ablation study and theory level comparisons further highlight the value of adaptive theory selection: full NPO outperforms the best fixed-theory variant on each dataset, while the niche discovery consistently improves performance.
4. NPO identifies distinct decision niches across tasks and demographic groups, making the selected information and behavioral assumptions explicit and improving the transparency of LLM-based decision modeling.

2 Related Works

2.1 Discrete Choice Models

Utility-based discrete choice models are foundational for modeling decision-making, assuming that individuals choose the alternative that yields the highest latent utility under specific statistical assumptions (Train, 2009). Classical models such as Multinomial Logit and Nested Logit emphasize tractable utility-based choice probabilities under increasingly flexible correlation structures, while later extensions including Mixed Logit and Latent Class Logit were introduced to capture unobserved taste heterogeneity across individuals (Mcfadden, 1977; Train, 2000; Mcfadden, 1974). These methods ensure interpretability through utility calculations and are widely used in numerous fields (Berbeglia et al., 2022; Habib, 2023). While these decision theories provide transparent formulation and maintain cost-efficiency, the static structure of utility functions limits their adaptation to tasks with larger datasets.

2.2 LLM Reasoning for Decision-making

Machine learning methods, such as Random Forests, XGBoost, and neural network models, provide a different perspective by directly learning from the entire set of diverse features (Breiman, 2001; Chen & Guestrin, 2016). Most recently, LLM-based frameworks such as chain-of-thought reasoning, in-context learning, and prompt optimization have demonstrated the reasoning and multi-step problem-solving ability of LLMs beyond text generation and question answering (Wei et al., 2022; Lampinen et al., 2022; Ramnath et al., 2025). Preliminary works regarding LLM-driven decision making suggest the capability of LLMs in human decision scenarios with structured prompt design and persona information (Liu et al., 2024; Joshi et al., 2025). Following works such as ATHENA and DualMind also demonstrate that the performance of LLMs in human decision making tasks can be improved by introducing numerical information and a self-correction mechanism (Zhao et al., 2025b;a; Yuksekgonul et al., 2025; Yang et al., 2026).

2.3 Feature Selection for Decision-making

Feature selection and dimensionality reduction are of great importance when processing high-dimensional data, especially in real-world tasks where measuring or acquiring certain features may incur economic or time costs (Remeseiro & Bolon-Canedo, 2019; Guyon & Elisseeff, 2003). Early regularization-based methods incorporate feature selection directly into model estimation by penalizing model parameters, thereby encouraging sparse solutions through coefficient shrinkage (Tibshirani, 1996; Xu et al., 2012). In contrast, wrapper-based methods such as recursive feature elimination (RFE) iteratively retain or discard features according to their contribution to predictive performance (Guyon et al., 2002; Chen & Jeong, 2007). Prior studies show that LLMs can perform selection either by relying on textual understanding and internal reasoning rules (Balek et al., 2025) or combining semantic assessments with task-specific, data-driven learning (Jeong et al., 2024). By converting structured attributes into natural language descriptions, these methods allow feature relevance to be assessed at the semantic level (Jia et al., 2025). These studies motivate a complementary question: how can LLM-guided feature selection be integrated with explicit behavioral decision theories for human choice prediction?

3 Method

Overview. NPO identifies group-specific feature-theory conditions for heterogeneous decision modeling. We define a *decision niche state* as $\eta = (S, m)$, where S is the selected feature subset

and m is the behavioral theory used to organize these features. Under each niche state, a theory-guided **decision agent** is instantiated to model the group's decision behavior. A **selection agent** then refines the niche state by searching for feature-theory conditions that better support group-specific decision-making. Instead of treating feature selection as a standalone preprocessing step, NPO couples feature selection with theory-guided decision modeling: the niche state determines how the decision agent reasons, and the agent's observed decision performance guides the evolution of the decision niche state.

3.1 PRELIMINARIES

Given a discrete choice problem with a static choice set $\mathcal{C} = \{C_1, C_2, \ldots, C_K\}$, the behavioral data of group g are denoted as $\mathcal{D}_g = \{(x_i, y_i)\}_{i=1}^{N_g}$, where x_i represents the observed attributes of sample i, and $y_i \in \mathcal{C}$ is the observed choice. Following random utility theory (Manski, 1977), the utility of alternative C_k for sample i is written as $u_{i,k} = f(x_i) + \epsilon_{i,k}$, and the predicted choice is

$$\hat{y}_i = \arg\max_{C_k \in \mathcal{C}} u_{i,k}. \tag{1}$$

Although utility-based discrete choice models provide transparent behavioral structures, their fixed utility assumptions often make them difficult to adapt across heterogeneous groups. Large language models provide a complementary form of adaptability by reasoning over flexible textual contexts. Let $\mathcal{F}(\cdot)$ denote the LLM-based reasoning engine, and let $\mathcal{T}(\cdot)$ denote the textual mapping function that converts structured attributes into textual prompts. Given a selected feature set S and a decision theory m, DualMind (Yang et al., 2026) learns a decision agent $D = (U, K)$, where U is the theory-guided utility function and K is the contextual knowledge set. The resulting prediction can be written as

$$\hat{y}_i = \mathcal{F}\left(\mathcal{T}(x_i), D \mid S, m\right). \tag{2}$$

3.2 DECISION NICHE STATE

Real-world decision behavior is rarely explained by a single globally valid feature set or behavioral theory. Different groups may rely on different contextual cues, and the same theory may be predictive for one group but restrictive for another. This motivates our notion of a *decision niche*: a group-specific condition under which decision modeling becomes most predictive. In NPO, this condition is represented as a joint feature-theory state rather than a feature subset alone.

For clarity, we first describe the process for a fixed group and omit the group index g. The same procedure is later applied independently to each group. At iteration t, the niche state is defined as

$$\eta_t = (S_t, m_t), \tag{3}$$

where $S_t \subseteq \mathcal{X}$ is the **selected feature subset**, $m_t \in \mathcal{M}$ is the **selected decision theory** in the iteration t, and $\mathcal{X}$ denotes the full feature space. The state η_t specifies the external condition under which decision agents will be instantiated and optimized. We consider a theory pool $\mathcal{M}$ that contains multiple utility-based behavioral assumptions. Building upon multinomial logit (MNL), the pool may include Nested Logit (NL) (Mcfadden, 1977; 1974), Mixed Logit (MXL) (Train, 2000), Latent Class Logit (LCL) (Greene & Hensher, 2003), and Random Regret Minimization (RRM) (Chorus, 2010). These theories encode different behavioral structures, such as substitution patterns, continuous heterogeneity, discrete heterogeneity, and regret-based comparison. Rather than selecting one theory globally, NPO allows the preferred theory to emerge jointly with the feature subset for each group.

3.3 CO-EVOLUTIONARY FEATURE-THEORY ALIGNMENT

The best niche state is unknown before decision modeling. A feature subset can only be judged after it shapes a decision model, while a decision model can only be optimized after a feature-theory state is specified. Therefore, NPO treats feature-theory selection and decision-model learning as a co-evolutionary process: the selection agent proposes a niche state, the decision agent learns under that state, and the validation behavior of the learned decision agent guides the next proposal.

Decision Agent Optimization under Exploratory Feature-Theory Combinations. Given the proposed feature subset S_t, NPO instantiates one candidate decision agent for each theory $m \in$

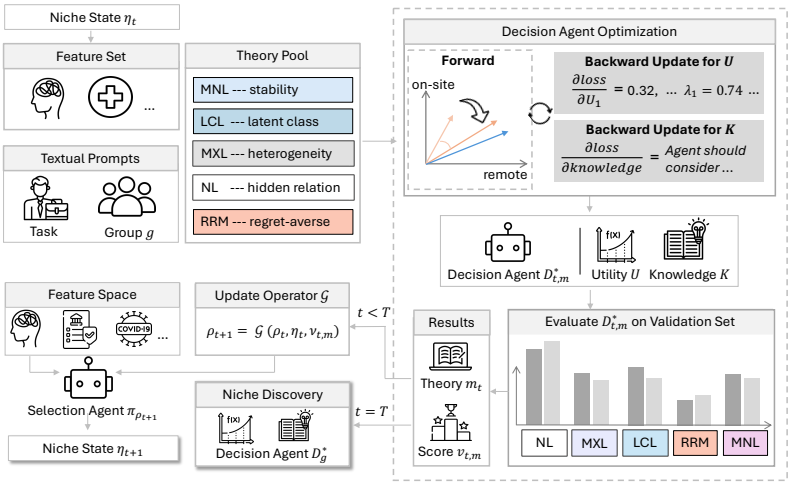

Figure 2: **Overview of the NPO Framework.** Within each iteration for group g, the selection agent proposes a candidate feature subset. Given this feature subset, NPO instantiates one decision agent for each behavioral theory in the theory pool and evaluates all candidate agents on the validation set. The best-performing theory determines the recorded niche state, and the corresponding validation feedback is used to update the adaptive prompt. After all iterations, the final prediction agent is selected from the accumulated decision agent pool through a two-stage validation-based procedure.

$\mathcal{M}$. Each candidate agent is optimized on the training data under the corresponding feature-theory combination:

$$D^*_{t,m} = \arg\min_{D} \mathcal{L}_{\mathrm{DM}}\left(D; (S_t, m), \mathcal{D}^{\mathrm{train}}\right). \tag{4}$$

Here, $D = (U, K)$ consists of the initial utility function U and the knowledge set K. The feature subset $S_t \in \eta_t$ determines what behavioral information is exposed to the decision agent, while the theory m shapes the utility function structure and agent decision principles. Therefore, each decision agent is not learned independently of feature selection; it is the decision agent induced by the corresponding feature-theory condition. This makes the search model-aware: the proposed niche state is assessed through the validation performance of the exploratory agents it induces. After validation, each candidate agent receives a **validation score** $v_{t,m}$, measured by F1 in our implementation, where larger values indicate better performance.

Selection Agent Prompt Update. After the candidate agents are evaluated on the validation set, the resulting feedback $\{v_{t,m}\}_{m\in\mathcal{M}}$ measures how well the current niche state supports decision modeling under different behavioral theories. The feedback is then used to update the prompt parameter:

$$\rho_{t+1} = \mathcal{G}\left(\rho_t, \eta_t, \{v_{t,m}\}_{m\in\mathcal{M}}\right), \tag{5}$$

where $\mathcal{G}(\cdot)$ denotes the prompt update operator (Yuksekgonul et al., 2025). Through this feedback loop, feature-theory selection and the decision agent learning co-evolve: the selection agent proposes states that shape the decision agents, and the agents' validation performance reshapes the subsequent search.

New Niche State Proposal. The search over niche states is performed by a bounded-rational selection agent parameterized by an adaptive prompt ρ_t. Given the current decision niche state η_t and the validation feedback from the candidate agents trained under this state, the selection agent proposes the next niche state:

$$\eta_{t+1} \sim \pi_{\rho_t}\left(\cdot \mid \eta_t, \{v_{t,m}\}_{m\in\mathcal{M}}\right). \tag{6}$$

The action of the selection agent is therefore the proposal of a new decision niche state, $a_t = \eta_{t+1}$. This transition is not intended to exhaustively enumerate all feature subsets and theories. Instead, it represents a bounded-rational search step, where the prompt-guided agent uses previous validation feedback to propose a plausible next feature-theory state.

Decision Agent Pool Construction. At each iteration, the agents $\{D^*_{t,m} : m \in \mathcal{M}\}$ form an iteration-level candidate pool. Across all iterations, these candidates are accumulated into the full

decision agent pool:

$$\mathcal{P} = \{D^*_{t,m} : t = 1, \ldots, T,\ m \in \mathcal{M}\}. \tag{7}$$

This pool stores the theory-conditioned agents explored at each iteration, along with their corresponding validation scores $\{v_{g,t,m}\}$.

3.4 Group-Specific Niche Discovery

The above co-evolutionary process is applied independently to each group $g \in \mathcal{G}$. For each group, the search produces a set of theory-guided decision agents, each associated with a proposed niche state and a validation score. The remaining question is how to convert this explored agent pool into a single prediction agent for the group. We use validation performance to select the final agent, but avoid selecting directly over all agents without regard to theory identity. For each theory m, we define a theory-level validation score by aggregating its validation records across all NPO iterations: $E_{g,m} = \sum_{t \in \{1,\ldots,T\}} v_{g,t,m}$. The selected theory is then the one with the highest aggregated validation score:

$$m^*_g = \arg\max_{m \in \mathcal{M}} E_{g,m}. \tag{8}$$

Given the selected theory m^*_g, the final prediction agent is the best-performing agent trained under this theory:

$$t^*_g = \arg\max_{t \in \{1,\ldots,T\}} v_{g,t,m^*_g}, \qquad D^*_g = D^*_{g,t^*_g,m^*_g}, \qquad \eta^*_g = (S_{g,t^*_g}, m^*_g). \tag{9}$$

This gives a final group-specific decision niche η^*_g and its induced decision agent D^*_g. This procedure keeps theory selection at the group level while allowing the final agent to be recovered from the most predictive iteration under the selected theory. The final prediction for sample i from group g is generated by

$$\hat{y}_i = \mathcal{F}\left(\mathcal{T}(x_i), D^*_g \mid \eta^*_g\right), \qquad i \in \mathcal{D}^{\text{test}}_g. \tag{10}$$

4 Experiment

4.1 Experiment Setup

Datasets and Splits. To evaluate the performance and adaptivity of NPO across different domains, we choose three real-world datasets spanning everyday behavioral tasks and health-related decisions. (1) **Job Behavior (JOB)**: This dataset originates from a national survey that measures the behavioral shifts of respondents after the outbreak of COVID-19 (Wozniak et al., 2020). It predicts respondents' employment status with three labels: working for salary, self-employed, and not working. (2) **Alcohol (ALC) and Marijuana (MAJ) Misuse**: These two datasets are derived from the National Survey on Drug Use and Health in the United States (U.S. Department of Health and Human Services et al., 2024). They predict the time since respondents' last marijuana misuse and alcohol use, respectively, with three labels: within the past 30 days, within the past 12 months, and more than 12 months ago. Each dataset is partitioned into nine age-income groups, with separate training, validation, and test splits within each group. Feature selection is conducted using only the validation set and more details are provided in Appendix Table 4.

Baselines and Selection Methods. The choice of baseline models can be divided into: (1) Machine Learning Methods including logistic regression (LR), random forest (RF), and XGBoost (XGB). (2) Symbolic Regression Methods including PySR and gplearn (Cranmer, 2023; Stephens, 2015). (3) LLM-based Methods. We choose vanilla LLMs, TextGrad (Yuksekgonul et al., 2025), ATHENA (Zhao et al., 2025b), and DualMind (Yang et al., 2026). For feature selection methods, we select three methods that could identify definite subsets of features as upstream frameworks: (1) Forward Selection with LR as the prediction model. (2) Lasso with LR as the prediction model. (3) Recursive Feature Elimination (RFE) with LR, RF and XGB as prediction models. For baselines with selection methods, hyperparameters are tuned separately for each group using the validation split.

Models and Configurations. To comprehensively evaluate the performance and generalizability of NPO, we choose `GPT-4o-mini` (Hurst et al., 2024), `GPT-5-mini` (OpenAI, 2025),

DeepSeek-V3.2 (DeepSeek-AI et al., 2025) and Qwen3-32B (Yang et al., 2025) as the reasoning model and selection model. Since LLM-based baselines require a static subset of features, the feature budgets for NPO on the three datasets match those used by ATHENA and DualMind, as shown in Table 4. The number of reasoning steps in NPO is set to 4, and the number of selection iterations is set to 3.

4.2 RESULTS

Table 1: **Performance Comparison on ALC, MAJ, and JOB.** Bold numbers indicate the best result in each dataset-metric column. F1 and Acc refer to weighted F1 and accuracy. Feat. denotes the number of effective features. 4o-mini, 5-mini, dp-v3.2, and qwen3 refer to GPT-4o-mini, GPT-5-mini, DeepSeek-v3.2 and Qwen3-32B as the decision agent and selection agent respectively. *F* and *S* refer to the full feature set and the feature set selected by NPO(4o-mini).

Methods	ALC			MAJ			JOB		
	F1 ↑	Acc ↑	Feat. ↓	F1 ↑	Acc ↑	Feat. ↓	F1 ↑	Acc ↑	Feat. ↓
LR	0.467	0.470	310.0	0.487	0.457	290.0	0.452	0.441	26.0
RF	0.409	0.454	310.0	0.555	0.540	290.0	0.513	0.514	26.0
XGB	0.420	0.429	310.0	0.516	0.493	290.0	0.474	0.470	26.0
gplearn	0.373	0.400	25.2	0.535	0.504	17.1	0.470	0.443	12.2
Forward (LR)	0.423	0.458	17.3	0.487	0.449	15.1	0.538	0.525	17.1
RFE (RF)	0.486	0.508	11.6	0.499	0.461	13.3	0.538	0.533	16.4
RFE (XGB)	0.438	0.475	8.4	0.550	0.520	16.9	0.565	0.553	11.1
RFE (LR)	0.460	0.475	8.0	0.499	0.465	14.2	0.525	0.525	10.7
Lasso (LR)	0.405	0.429	11.3	0.535	0.539	8.7	0.527	0.525	11.0
PySR	0.437	0.429	11.0	0.513	0.473	9.2	0.492	0.480	8.9
Vanilla LLM (F)	0.284	0.338	310.0	0.088	0.191	290.0	0.224	0.230	26.0
TextGrad (F)	0.274	0.312	310.0	0.106	0.168	290.0	0.389	0.426	26.0
Vanilla LLM (S)	0.315	0.375	14.8	0.137	0.211	12.7	0.308	0.287	12.6
TextGrad (S)	0.318	0.425	14.8	0.156	0.211	12.7	0.374	0.357	12.6
Athena	0.224	0.278	16.0	0.522	0.535	14.0	0.399	0.460	15.0
DualMind	0.465	0.459	16.0	0.611	0.582	14.0	0.550	0.532	15.0
NPO(4o-mini)	0.496	0.513	14.8	**0.673**	**0.711**	12.7	0.655	0.635	12.6
NPO(dp-v3.2)	0.520	0.517	10.8	0.611	0.645	12.6	**0.699**	**0.684**	10.3
NPO(5-mini)	**0.538**	**0.525**	9.0	0.644	0.656	13.3	0.637	0.607	9.0
NPO (qwen3)	0.520	**0.525**	9.9	0.635	0.668	11.3	0.670	0.656	8.7

Performance and Selection Efficiency. As shown in Table 1, across three datasets and four LLM configurations, NPO improves F1 by 6.21%–36.26% and accuracy by 9.15%–33.07% relative to the strongest full-feature baseline for each dataset and metric. Compared with the strongest conventional feature-selection baseline (Forward, RFE, or Lasso), NPO also achieves consistent corresponding relative gains for all backbone LLMs. These improvements are achieved with compact feature subsets, averaging 9.0–14.8 features on ALC, 11.3–13.3 on MAJ, and 8.7–12.6 on JOB, compared with 310, 290, and 26 full features respectively. Together, these results support the effectiveness of NPO in constructing predictive decision niches across different decision scenarios and backbone LLMs.

4.3 ABLATION STUDY

Table 2: **Ablation Study on ALC, MAJ, and JOB Datasets.** DeepSeek-v3.2 serves as the selection and prediction model.

Setting	ALC		MAJ		JOB	
	F1	Acc	F1	Acc	F1	Acc
Initial Only	0.398	0.400	0.046	0.145	0.478	0.471
+ DualMind Optimization	0.429	0.421	0.502	0.457	0.476	0.471
+ Feature Selection	0.438	0.442	0.558	0.519	0.554	0.582
+ Theory Competition	0.480	0.458	0.582	0.594	0.598	0.594
+ Niche Discovery (NPO)	0.520	0.517	0.611	0.645	0.699	0.684

Contribution of Numerical Information and Selection. Table 2 shows that incorporating numerical information with DualMind-style MNL-guided reasoning substantially improves performance

over using only the initial information for ALC and MAJ tasks. Further adding feature selection leads to additional gains, indicating that the selection process helps the LLM focus on the most relevant information rather than reasoning over the full high-dimensional feature space.

Contribution of Theory Competition and Niche Discovery. In Table 2, adding theory competition yields average relative improvements of 7.28% in F1 and 6.71% in accuracy over feature selection. Niche discovery replaces the final iteration's validation winner with a theory selected by cumulative performance and its best validation checkpoint, further improving F1 and accuracy by 10.07% and 12.21%, respectively. Table 3 provides complementary evidence using `GPT-4o-mini`: Full NPO exceeds the best Fixed-NPO by 1.02%, 6.32%, and 8.99% in F1 on ALC, MAJ, and JOB. These results suggest that individual behavioral theories provide a foundation for decision reasoning, and theory competition together with niche discovery yields better performance than fixing a single theory.

Table 3: F1 Performance Comparison between Theory Candidates and Full NPO.

Method	ALC	MAJ	JOB
Fixed-NPO-MNL	0.486	0.633	0.521
Fixed-NPO-NL	0.485	0.602	0.571
Fixed-NPO-MXL	0.491	0.629	0.601
Fixed-NPO-LCL	0.455	0.622	0.504
Fixed-NPO-RRM	0.429	0.594	0.417
Full NPO	**0.496**	**0.673**	**0.655**

4.4 Feature-Theory Alignment Analysis

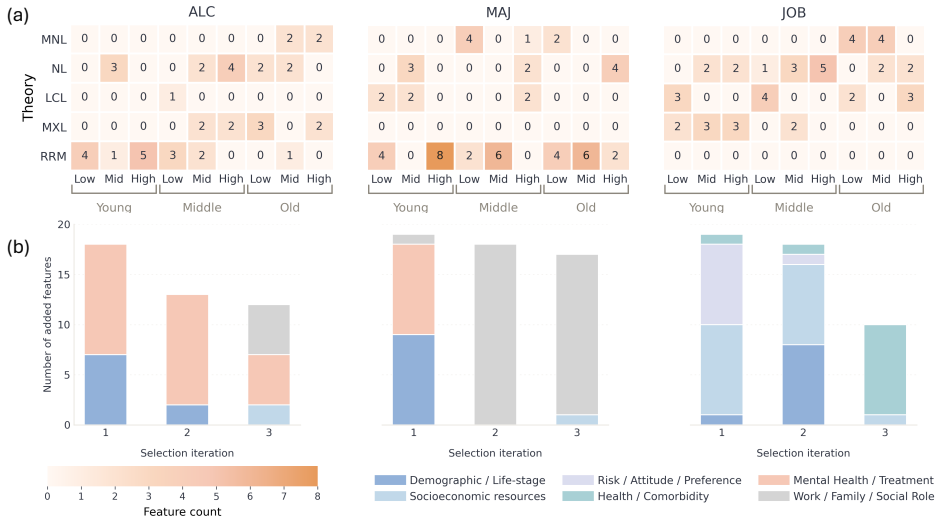

Figure 3: **Analysis of the Selected Features.** (a) Theory–group heatmap showing how the final selected features are distributed across decision theories for each group. (b) Evolution of selected feature types across selection iterations.

NPO reveals behavior modeling patterns specific to tasks and groups through the selection iterations. Figure 3a shows the distribution of final selected features. At the task level, MAJ is strongly dominated by RRM, which accounts for 32 out of 54 selected features. A similar pattern appears for the younger group in the task of ALC. By contrast, the modeling of JOB relies mainly on NL, LCL, and MXL, indicating a stronger role of structured trade-offs, latent group differences, and preference heterogeneity. The changes in selected feature types exhibit task specific patterns as well. Mental factors dominate the selected feature set for ALC, while MAJ quickly shifts toward work, family, and social-role features. Given the COVID background of the JOB dataset, selected features concentrated on socioeconomic resources and health-related features.

4.5 Feature Selection Iteration Details

Dynamic theory–feature adaptation improves group-specific decision modeling. As presented in Figure 4, NPO improves rapidly during the early selection iterations, but the strongest decision niche state does not always appear in the final iteration. This occurs because the relative performance of theory-conditioned agents changes as new features are added; a theory that is effective

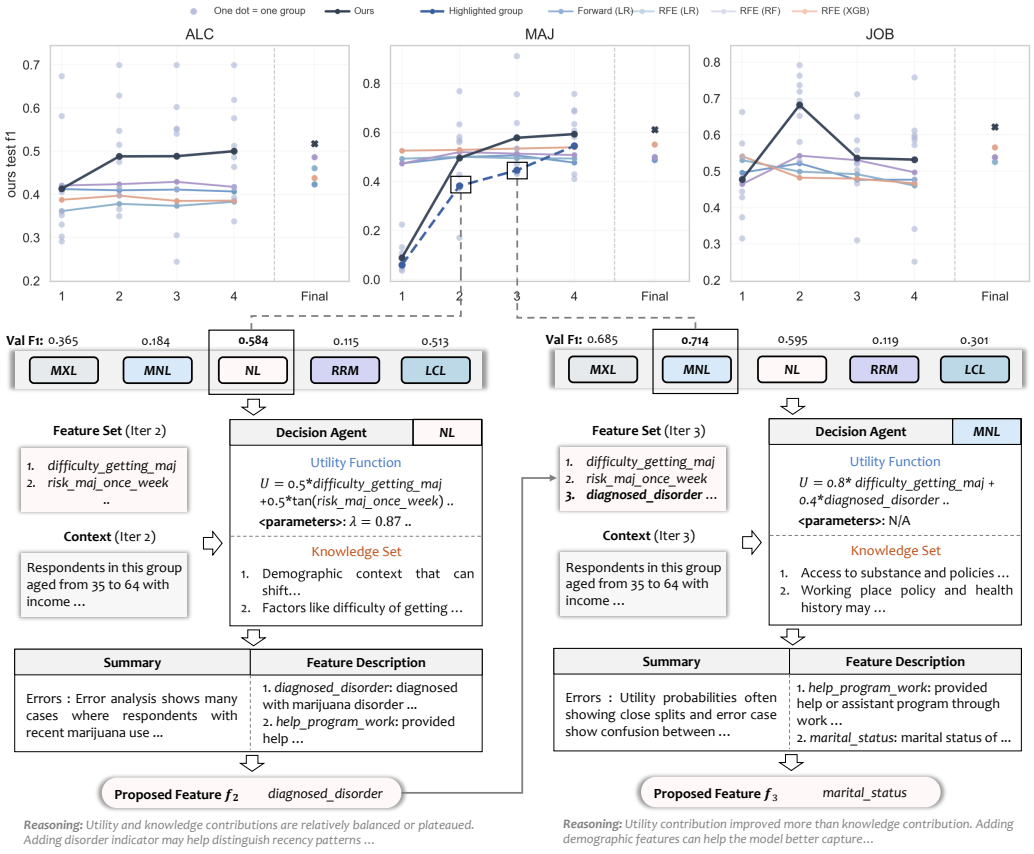

Figure 4: **An Example of the Detailed NPO Iteration Process. The top panel** shows post-hoc test F1 trajectories for validation-selected checkpoints. Each light purple dot represents one group, while the solid lines show the average F1 score across groups. **The bottom panel** details the update process for a representative group (shown in blue dots and dash lines), individuals aged 35–64 with income between \$50k and \$75k, from iteration 2 to iteration 3 on the MAJ dataset.

under one feature subset may be replaced under another. Therefore, NPO traces the full optimization trajectory and selects the best-supported decision niche, rather than directly committing to the terminal selection state. The highlighted MAJ group illustrates this process. From iteration 2 to iteration 3, the selected agent shifts from NL to MNL, and the validation F1 increases from 0.584 to 0.714 after adding a new feature. Keeping NL under the updated feature subset would lead to a clear performance drop relative to the adapted MNL agent. This example shows how NPO dynamically adapts the feature subset and theory to identify a stronger group-specific decision niche.

5 Conclusion

We propose NPO, a theory-guided LLM-agent framework that jointly refines human decision modeling and feature selection. To guide feature selection and reduce spurious or redundant cues, NPO grounds LLM agents with explicit behavioral theories. To couple information selection with decision modeling, NPO introduces a reciprocal optimization process in which decision agents co-evolve with the selection agents iteratively. Experiments on three real-world datasets show that NPO achieves average relative F1 gains of 9.38%–10.24% over the strongest baseline, while using up to 43.75% fewer features than the strongest LLM-based baseline. Beyond predictive gains, NPO reveals task- and group-specific behavioral modeling patterns, showing how different theories and feature subsets align with heterogeneous decision contexts. NPO provides a step toward theory-grounded LLM agents that can adapt both what information they use and how they reason about human decisions.

AI USE STATEMENT

Generative AI tools were used to assist in debugging, result visualization, checking grammar, and improving readability. Separately, LLMs are components of the proposed NPO framework, supporting feature selection, decision prediction, and updates to textual knowledge and utility representations; their roles and configurations are described in the paper. The authors reviewed AI-assisted revisions. All scientific claims and conclusions were evaluated by the authors, who take responsibility for the final content of this work.

REFERENCES

Sercan Ö Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 6679–6687, 2021.

Vojtěch Balek, Lukáš Sỳkora, Vilém Sklenák, and Tomáš Kliegr. Llm-based feature generation from text for interpretable machine learning. *Machine Learning*, 114(11):1–30, 2025.

Gerardo Berbeglia, Agustín Garassino, and Gustavo Vulcano. A comparative empirical study of discrete choice models in retail operations. *Management Science*, 68(6):4005–4023, 2022.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Xue-wen Chen and Jong Cheol Jeong. Enhanced recursive feature elimination. In *Sixth international conference on machine learning and applications (ICMLA 2007)*, pp. 429–435. IEEE, 2007.

Caspar G Chorus. A new model of random regret minimization. *European Journal of Transport and Infrastructure Research*, 10(2), 2010.

Miles Cranmer. Interpretable machine learning for science with pysr and symbolicregression. jl. *arXiv preprint arXiv:2305.01582*, 2023.

Aixin Liu DeepSeek-AI, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

John A Endler. Signals, signal conditions, and the direction of evolution. *The American Naturalist*, 139:S125–S153, 1992.

William H Greene and David A Hensher. A latent class model for discrete choice analysis: contrasts with mixed logit. *Transportation Research Part B: Methodological*, 37(8):681–698, 2003.

Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.

Khandker Nurul Habib. Rational inattention in discrete choice models: Estimable specifications of ri-multinomial logit (ri-mnl) and ri-nested logit (ri-nl) models. *Transportation Research Part B: Methodological*, 172:53–70, 2023.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International conference on learning representations*, volume 2024, pp. 32808–32824, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Daniel P Jeong, Zachary C Lipton, and Pradeep Ravikumar. Llm-select: Feature selection with large language models. *arXiv preprint arXiv:2407.02694*, 2024.

Pengyue Jia, Zhaocheng Du, Yichao Wang, Xiangyu Zhao, Xiaopeng Li, Yuhao Wang, Qidong Liu, Huifeng Guo, and Ruiming Tang. Self: Surrogate-light feature selection with large language models in deep recommender systems. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 1145–1155, 2025.

Brihi Joshi, Xiang Ren, Swabha Swayamdipta, Rik Koncel-Kedziorski, and Tim Paek. Improving language model personas via rationalization with psychological scaffolds. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Suzhou, China. Association for Computational Linguistics*, 2025.

Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97 (1-2):273–324, 1997.

Andrew Lampinen, Ishita Dasgupta, Stephanie Chan, Kory Mathewson, Mh Tessler, Antonia Creswell, James McClelland, Jane Wang, and Felix Hill. Can language models learn from explanations in context? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 537–563, 2022.

Tianming Liu, Manzi Li, and Yafeng Yin. Can large language models capture human travel behavior? evidence and insights on mode choice. *Evidence and Insights on Mode Choice (August 26, 2024)*, 2024.

John R MacKinnon and Kathleen S MacKinnon. Niche differentiation in a primate community. In *Malayan forest primates: Ten years' study in tropical rain forest*, pp. 167–190. Springer, 1980.

Charles F Manski. The structure of random utility models. *Theory and decision*, 8(3):229, 1977.

Daniel McFadden. Conditional logit analysis of qualitative choice behavior. 1972.

Daniel Mcfadden. Modelling the choice of residential location. *Spatial Interaction Theory Residential Location*, 673(477):72–77, 1977.

Daniel L. Mcfadden. *Conditional Logit Analysis of Qualitative Choice Behavior*. Frontiers in Econometrics, 1974.

OpenAI. Gpt-5 system card. Technical report, OpenAI, August 2025. URL `https://cdn.openai.com/gpt-5-system-card.pdf`.

Kiran Ramnath, Kang Zhou, Sheng Guan, Soumya Smruti Mishra, Xuan Qi, Zhengyuan Shen, Shuai Wang, Sangmin Woo, Sullam Jeoung, Yawei Wang, et al. A systematic survey of automatic prompt optimization techniques. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33066–33098, 2025.

Beatriz Remeseiro and Veronica Bolon-Canedo. A review of feature selection methods in medical applications. *Computers in biology and medicine*, 112:103375, 2019.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *International conference on machine learning*, pp. 31210–31227. PMLR, 2023.

Herbert A Simon. Bounded rationality. In *Utility and probability*, pp. 15–18. Springer, 1990.

Trevor Stephens. gplearn: Genetic programming in python, with a scikit-learn inspired api, 2015. URL `https://github.com/trevorstephens/gplearn`.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

Mc Faddenkenneth Train. Mixed mnl models for discrete response. *Journal of Applied Econometrics*, 15(5):447–470, 2000.

U.S. Department of Health and Human Services, Substance Abuse and Mental Health Services Administration, and Center for Behavioral Health Statistics and Quality. National survey on drug use and health, 2023 Public Use File. Public Use Dataset, 2024. URL https://www.samhsa.gov/data/data-we-collect/nsduh/datafiles.

John Von Neumann and Oskar Morgenstern. Theory of games and economic behavior: 60th anniversary commemorative edition. In *Theory of games and economic behavior*. Princeton university press, 2007.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Abigail Wozniak, Joe Willey, Jennifer Benz, and Nick Hart. COVID Impact Survey: Version # [dataset]. National Opinion Research Center, 2020. URL https://www.covid-impact.org/. Conducted by NORC at the University of Chicago for the Data Foundation.

Guanlin Wu, Boyan Su, Yang Zhao, Pu Wang, Yichen Lin, and Hao Frank Yang. Towards physics-informed spatial intelligence with human priors: An autonomous driving pilot study. *arXiv preprint arXiv:2510.21160*, 2025.

Zhixiang Xu, Kilian Weinberger, and Olivier Chapelle. The greedy miser: Learning under test-time budgets. *arXiv preprint arXiv:1206.6451*, 2012.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Hao Frank Yang, Yang Zhao, Yichen Lin, and Takeru Igusa. Thinking, fast and slow in ai agents for personalized decision-making. 2026.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative ai by backpropagating language model feedback. *Nature*, 639(8055):609–616, 2025.

Yang Zhao, Pu Wang, and Hao Frank Yang. How to auto-optimize prompts for domain tasks? adaptive prompting and reasoning through evolutionary domain knowledge adaptation. *arXiv preprint arXiv:2510.21148*, 2025a.

Yibo Zhao, Yang Zhao, Hongru Du, and Hao Frank Yang. Personalized decision modeling: Utility optimization or textualized-symbolic reasoning. *arXiv preprint arXiv:2511.02194*, 2025b.

Debra Zuppinger-Dingley, Bernhard Schmid, Jana S Petermann, Varuna Yadav, Gerlinde B De Deyn, and Dan FB Flynn. Selection for niche differentiation in plant communities increases biodiversity effects. *Nature*, 515(7525):108–111, 2014.

A Appendix

A.1 Background on Discrete Choice Models

This section provides a brief introduction to the discrete choice models used in our pipeline, including Multinomial Logit (MNL), Mixed Logit (MXL), Nested Logit (NL), Random Regret Minimization (RRM), and Latent Class Logit (LCL). These models provide structured behavioral priors for modeling how individuals make choices among a finite set of alternatives.

We consider an individual n choosing one alternative from a choice set $\mathcal{J}_n$. Each alternative $j \in \mathcal{J}_n$ is associated with an attribute vector x_{nj}. The goal of a discrete choice model is to characterize the probability that alternative j is chosen.

A.2 Random Utility Framework

Most models in our pipeline are based on the random utility framework. The utility that individual n obtains from alternative j is written as

$$U_{nj} = V_{nj} + \varepsilon_{nj}, \tag{11}$$

where V_{nj} is the systematic utility component explained by observed variables, and ε_{nj} is an unobserved stochastic term. Individual n chooses alternative j if and only if

$$U_{nj} \geq U_{ni}, \qquad \forall i \neq j. \tag{12}$$

A common specification for the systematic utility is linear-in-parameters:

$$V_{nj} = \beta^\top x_{nj}, \tag{13}$$

where β denotes the preference coefficients. Different discrete choice models mainly differ in how they model the stochastic term, heterogeneity, or dependence structure across alternatives.

A.3 Multinomial Logit (MNL)

The Multinomial Logit model is the standard benchmark in discrete choice analysis. It assumes that the error terms ε_{nj} are independent and identically distributed with a Type-I Extreme Value distribution. Under this assumption, the choice probability has the closed form

$$P_{nj} = \frac{\exp(V_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(V_{ni})}. \tag{14}$$

With the linear utility specification, this becomes

$$P_{nj} = \frac{\exp(\beta^\top x_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(\beta^\top x_{ni})}. \tag{15}$$

MNL is attractive because of its simplicity, interpretability, and efficient estimation. However, it relies on the independence of irrelevant alternatives (IIA) property, meaning that the odds ratio between any two alternatives is unaffected by the presence or characteristics of other alternatives. This assumption may be restrictive when alternatives are correlated or when preference heterogeneity is substantial.

In our pipeline, MNL serves as the most basic theory-driven choice model and provides an interpretable behavioral prior for the LLM.

A.4 Mixed Logit (MXL)

Mixed Logit extends MNL by allowing preference coefficients to vary across individuals. Instead of assuming a fixed coefficient vector β, MXL assumes that

$$\beta_n \sim f(\beta \mid \theta), \tag{16}$$

where $f(\cdot)$ is a parametric mixing distribution governed by parameters θ, such as a normal or log-normal distribution.

Conditional on β_n, the choice probability remains logit:

$$P_{nj}(\beta_n) = \frac{\exp(\beta_n^\top x_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(\beta_n^\top x_{ni})}. \tag{17}$$

The unconditional choice probability integrates over the distribution of random coefficients:

$$P_{nj} = \int \frac{\exp(\beta^\top x_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(\beta^\top x_{ni})} f(\beta \mid \theta)\, d\beta. \tag{18}$$

This integral generally has no closed form and is usually approximated by simulation:

$$P_{nj} \approx \frac{1}{R} \sum_{r=1}^{R} \frac{\exp(\beta^{(r)\top} x_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(\beta^{(r)\top} x_{ni})}, \tag{19}$$

where $\beta^{(r)}$ is the r-th draw from the mixing distribution.

Compared with MNL, MXL can capture rich taste heterogeneity and relax the IIA assumption. In our pipeline, MXL provides a more flexible behavioral theory when individuals may respond differently to the same attributes.

A.5 Nested Logit (NL)

Nested Logit relaxes the IIA assumption by grouping similar alternatives into nests. Let the set of alternatives be partitioned into nests $m \in \mathcal{M}$, and let B_m denote the set of alternatives in nest m. The utility of alternative $j \in B_m$ remains $U_{nj} = V_{nj} + \varepsilon_{nj}$, but the error structure now allows correlation within the same nest.

The probability of choosing alternative j in nest m can be decomposed as

$$P_{nj} = P_n(j \mid m)\, P_n(m), \tag{20}$$

where the conditional probability within nest m is

$$P_n(j \mid m) = \frac{\exp\left(V_{nj}/\lambda_m\right)}{\sum_{i \in B_m} \exp\left(V_{ni}/\lambda_m\right)}, \tag{21}$$

and the marginal probability of choosing nest m is

$$P_n(m) = \frac{\exp\left(\lambda_m I_{nm}\right)}{\sum_{k \in \mathcal{M}} \exp\left(\lambda_k I_{nk}\right)}. \tag{22}$$

Here,

$$I_{nm} = \log\left(\sum_{i \in B_m} \exp\left(V_{ni}/\lambda_m\right)\right) \tag{23}$$

is the inclusive value for nest m, and $\lambda_m \in (0, 1]$ is the dissimilarity parameter. A smaller λ_m implies stronger correlation among alternatives in the same nest.

NL is useful when alternatives naturally form structured groups, such as brands, travel modes, or product categories. In our pipeline, it represents behavioral theories in which choices are made hierarchically rather than from a completely flat alternative set.

A.6 Random Regret Minimization (RRM)

Random Regret Minimization departs from the utility-maximization paradigm and assumes that decision makers seek to minimize anticipated regret. Instead of evaluating each alternative in terms of absolute utility, RRM models pairwise regret from choosing alternative j over competing alternative i.

For attribute k, the regret contribution of choosing j when compared with i is commonly defined as

$$r_{njik} = \log\left(1 + \exp\left(\beta_k(x_{nik} - x_{njk})\right)\right). \tag{24}$$

Summing over all competing alternatives and attributes yields the total regret of alternative j:

$$R_{nj} = \sum_{i \in \mathcal{J}_n,\, i \neq j} \sum_k \log\left(1 + \exp\left(\beta_k(x_{nik} - x_{njk})\right)\right). \tag{25}$$

An alternative with lower regret is more likely to be chosen. The corresponding choice probability is

$$P_{nj} = \frac{\exp(-R_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(-R_{ni})}. \tag{26}$$

RRM is especially appropriate when decision makers compare alternatives comparatively and are sensitive to potential disadvantage relative to forgone options. In our pipeline, RRM enriches the theory pool by introducing a non-utility-based behavioral mechanism that may better fit comparative or regret-sensitive decisions.

A.7 Latent Class Logit (LCL)

Latent Class Logit models unobserved heterogeneity by assuming that the population consists of a finite number of latent classes. Each class $c \in \{1, \ldots, C\}$ has its own coefficient vector β_c, and each individual belongs probabilistically to one of these classes.

Conditional on class c, the choice probability follows the MNL form:

$$P_{nj|c} = \frac{\exp(\beta_c^\top x_{nj})}{\sum_{i \in \mathcal{J}_n} \exp(\beta_c^\top x_{ni})}. \tag{27}$$

Let π_{nc} denote the probability that individual n belongs to class c, with

$$\sum_{c=1}^{C} \pi_{nc} = 1. \tag{28}$$

These class membership probabilities are often modeled by

$$\pi_{nc} = \frac{\exp(\gamma_c^\top z_n)}{\sum_{s=1}^{C} \exp(\gamma_s^\top z_n)}, \tag{29}$$

where z_n are observed individual characteristics and γ_c are class assignment parameters.

The unconditional choice probability is then

$$P_{nj} = \sum_{c=1}^{C} \pi_{nc}\, P_{nj|c}. \tag{30}$$

LCL captures heterogeneity through a discrete segmentation of the population rather than a continuous coefficient distribution as in MXL. In our pipeline, LCL provides a theory of behavior in which different subpopulations may follow qualitatively different decision rules.

A.8 Comparison of the Modeling Assumptions

The five models differ in the behavioral assumptions they impose. MNL assumes homogeneous preferences and IIA. MXL introduces continuous preference heterogeneity. NL captures correlation among similar alternatives through a hierarchical structure. RRM replaces utility maximization with regret minimization. LCL models discrete latent segments with class-specific preferences.

A.9 Role in Our Pipeline

The above models are used in our framework as structured behavioral theory candidates. Each theory provides a distinct interpretation of how observed attributes may influence decisions, such as homogeneous utility maximization (MNL), heterogeneous preferences (MXL), hierarchical substitution patterns (NL), comparative regret minimization (RRM), or latent population segmentation (LCL). By incorporating these theories into the theory pool, the LLM is encouraged to reason over decision patterns grounded in established behavioral modeling principles rather than relying solely on free-form textual inference.

B Experiment Extended Results

B.1 Information of Datasets

The job choice dataset (JOB) records the work arrangement of respondents under the influence of the COVID-19 pandemic, such as transportation constraints, government policy, and personal attitudes. The marijuana misuse (MAJ) and alcohol use (ALC) datasets originate from the same survey, which provides detailed demographic information (e.g., age, income, education) and health-related topics (e.g., attitude, mental health, daily habit) about the respondents.

Detailed information about the three datasets is shown in Table 4. Each dataset comprises nine groups defined by the cross-product of three age groups (< 35, 35–64, and ≥ 65 years) and three household income groups. The income bins are chosen based on the differences in the question design across different questionnaires.

Table 4: Dataset Statistics and Group Definitions.

	Features		Household income ($K)			Sample count		
Dataset	Feat.B	Feat.S	Low	Middle	High	Train	Val	Test
JOB	15	26	< 40	$[40, 100)$	≥ 100	405	227	244
MAJ	14	290	< 50	$[50, 75)$	≥ 75	450	274	256
ALC	16	310	< 50	$[50, 75)$	≥ 75	349	243	240

B.2 Direct Behavioral Theory Baselines

To assess the predictive performance of standalone behavioral theories, we evaluate Direct-T for each theory $T \in \{\mathrm{MNL}, \mathrm{NL}, \mathrm{MXL}, \mathrm{LCL}, \mathrm{RRM}\}$. Each Direct-T fits numerical parameters using the initial utility specification and feature set, without LLM calls, feature selection, knowledge optimization, or theory competition. All methods use the same train/validation/test splits as NPO.

Table 5: Standalone behavioral theory performance. Bold values indicate the best Direct result for each dataset and metric.

	ALC		MAJ		JOB	
Theory	F1	Acc	F1	Acc	F1	Acc
MNL	0.4469	**0.4708**	0.5765	0.5469	0.4185	0.4877
NL	**0.4538**	0.4667	**0.5855**	**0.5586**	0.4136	0.4713
MXL	0.4454	0.4583	0.5629	0.5312	**0.5247**	**0.5902**
LCL	0.4354	0.4500	0.5620	0.5273	0.4259	0.5000
RRM	0.4335	0.4583	0.5780	0.5469	0.4011	0.4795

Table 5 shows that no single theory performs best across all three datasets: NL achieves the highest F1 on ALC and MAJ, whereas MXL performs best on JOB. Together with the Fixed-NPO results in Table 3, these baselines distinguish standalone theory performance from the gains obtained through the agent-based optimization procedure.

B.3 Theory Dynamics and Performance

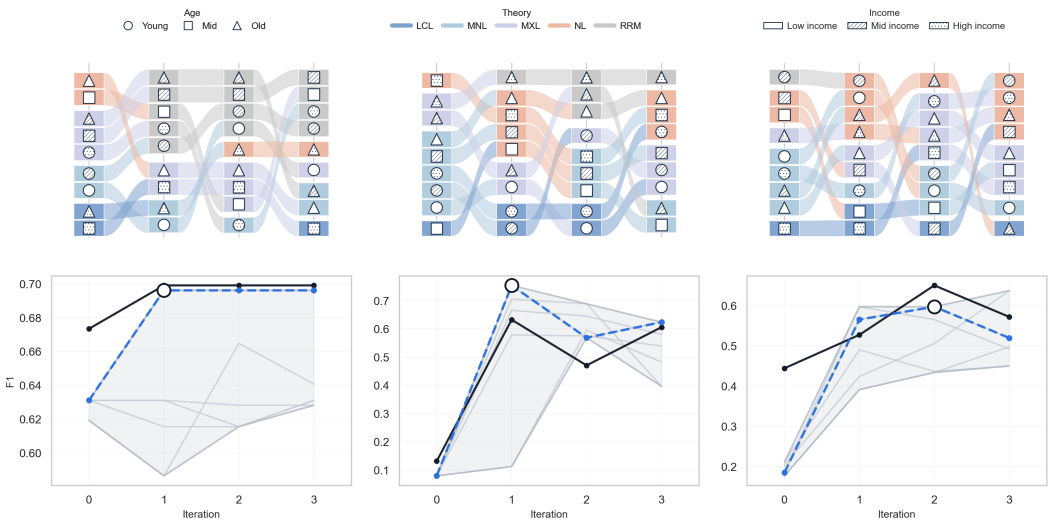

Figure 5: Group-level theory dynamics and final theory selection. Top: group-level microflow trajectories across four iterations for Alcohol Use (ALC), Marijuana Use (MAJ), and Job Choice (JOB). Each small marker denotes one group, where marker shape encodes age and marker pattern encodes income level; ribbon colors indicate the theory assigned at each iteration. Bottom: validation-based theory comparison for one representative group from each dataset. Light gray lines show the validation trajectories of all candidate theories, the gray band marks the range between the upper and lower validation bounds across theories, the blue dashed line highlights the finally selected theory, and the black line shows the corresponding test performance across iterations. By selecting theories that perform best overall on the validation set, we ensured the reliability of the chosen theories and avoided single mutations and variance. Meanwhile, the test set results demonstrate that the overall performance of the selected theories remains reliable.

B.4 Computing Cost and Resources

Table 6: GPT-4o-mini token usage and cost across the three datasets.

Dataset	Calls	Prompt Tokens (M)	Completion Tokens (M)	Cost (USD)
ALC	40,895	45.977	1.650	7.89
MAJ	47,586	53.397	1.681	9.02
JOB	40,266	32.923	1.391	5.77
Total	128,747	132.296	4.722	22.68

The experiments are conducted through API-based LLM inference and optimization. We run all scripts on a Macbook, which is responsible for data preprocessing, prompt generation, API scheduling, output parsing, and metric calculation. As a result, local hardware mainly supports workflow management, while the dominant computational expense comes from repeated LLM calls during the optimization process.

B.5 Extended Results of Selection-based Methods

Table 7: Validation-tuned feature selection baselines on ALC, MAJ, and JOB. For each method, hyperparameters are selected separately for each group using the validation split, and *Feat.* reports the mean number of selected features across groups.

Dataset	Method	Val F1	Val Acc	Test F1	Test Acc	Feat.
ALC	Forward (LR)	0.514	0.547	0.423	0.458	17.3
ALC	Lasso (LR)	0.501	0.523	0.405	0.429	11.3
ALC	RFE (LR)	0.546	0.568	0.460	0.475	8.0
ALC	RFE (RF)	0.533	0.547	0.486	0.508	11.6
ALC	RFE (XGB)	0.531	0.551	0.438	0.475	8.4
MAJ	Forward (LR)	0.570	0.547	0.487	0.449	15.1
MAJ	Lasso (LR)	0.559	0.558	0.535	0.539	8.7
MAJ	RFE (LR)	0.538	0.489	0.499	0.465	14.2
MAJ	RFE (RF)	0.591	0.558	0.499	0.461	13.3
MAJ	RFE (XGB)	0.516	0.493	0.550	0.520	16.9
JOB	Forward (LR)	0.631	0.617	0.538	0.525	17.1
JOB	Lasso (LR)	0.565	0.564	0.527	0.525	11.0
JOB	RFE (LR)	0.612	0.595	0.525	0.525	10.7
JOB	RFE (RF)	0.595	0.581	0.538	0.533	16.4
JOB	RFE (XGB)	0.620	0.604	0.565	0.553	11.1

Table 7 presents detailed results for prediction models with different selection methods. The feature-selection baselines reported in Table 1 are obtained by tuning hyperparameters separately for each behavioral group using validation F1. Tables 8–10 provide the detailed group-wise selected configurations and their corresponding validation/test performance.

Table 10: Group-wise validation-tuned feature selection baselines on JOB. For each behavioral group and method, hyperparameters are selected independently using validation F1. For forward selection and RFE, *Best HP* reports the selected target number of features k; for lasso, it reports the selected regularization parameter C. *Feat.* denotes the number of selected features under the chosen configuration.

Group	Method	Best HP	Feat.	Val F1	Val Acc	Test F1	Test Acc
mid_high	Forward (LR)	k=26	26	0.529	0.500	0.448	0.424
	Lasso (LR)	C=0.01	1	0.505	0.500	0.581	0.576
	RFE (LR)	k=16	16	0.564	0.542	0.482	0.485
	RFE (RF)	k=26	26	0.621	0.625	0.593	0.636
	RFE (XGB)	k=4	4	0.713	0.708	0.628	0.606
mid_low	Forward (LR)	k=16	16	0.694	0.667	0.676	0.650
	Lasso (LR)	C=1	19	0.657	0.619	0.322	0.300
	RFE (LR)	k=8	8	0.732	0.714	0.676	0.650
	RFE (RF)	k=20	20	0.618	0.571	0.471	0.450
	RFE (XGB)	k=4	4	0.688	0.667	0.635	0.600
mid_mid	Forward (LR)	k=26	26	0.596	0.565	0.422	0.375
	Lasso (LR)	C=0.3	12	0.617	0.609	0.591	0.542
	RFE (LR)	k=4	4	0.617	0.609	0.524	0.458
	RFE (RF)	k=26	26	0.552	0.522	0.391	0.333
	RFE (XGB)	k=16	16	0.602	0.609	0.498	0.417
old_high	Forward (LR)	k=20	20	0.711	0.710	0.554	0.484
	Lasso (LR)	C=0.3	13	0.684	0.677	0.640	0.645
	RFE (LR)	k=4	4	0.769	0.774	0.508	0.548
	RFE (RF)	k=8	8	0.722	0.710	0.629	0.613
	RFE (XGB)	k=4	4	0.870	0.871	0.727	0.710
old_low	Forward (LR)	k=20	20	0.513	0.458	0.371	0.346
	Lasso (LR)	C=0.3	13	0.512	0.458	0.385	0.346
	RFE (LR)	k=20	20	0.537	0.500	0.473	0.462
	RFE (RF)	k=12	12	0.649	0.625	0.457	0.385
	RFE (XGB)	k=8	8	0.510	0.458	0.462	0.423
old_mid	Forward (LR)	k=4	4	0.589	0.613	0.546	0.607
	Lasso (LR)	C=0.3	10	0.570	0.581	0.630	0.643
	RFE (LR)	k=12	12	0.520	0.516	0.636	0.643
	RFE (RF)	k=20	20	0.518	0.516	0.682	0.679
	RFE (XGB)	k=4	4	0.518	0.516	0.596	0.571
young_high	Forward (LR)	k=26	26	0.592	0.621	0.566	0.600
	Lasso (LR)	C=3	21	0.592	0.621	0.566	0.600
	RFE (LR)	k=16	16	0.630	0.655	0.597	0.629
	RFE (RF)	k=16	16	0.470	0.517	0.552	0.600
	RFE (XGB)	k=16	16	0.595	0.621	0.528	0.600
young_low	Forward (LR)	k=4	4	0.845	0.850	0.748	0.750
	Lasso (LR)	C=0.3	9	0.450	0.450	0.496	0.500
	RFE (LR)	k=8	8	0.506	0.500	0.362	0.400
	RFE (RF)	k=16	16	0.555	0.550	0.263	0.350
	RFE (XGB)	k=20	20	0.460	0.450	0.263	0.350
young_mid	Forward (LR)	k=12	12	0.651	0.583	0.593	0.556
	Lasso (LR)	C=0.1	1	0.450	0.500	0.427	0.444
	RFE (LR)	k=8	8	0.607	0.500	0.450	0.407
	RFE (RF)	k=4	4	0.655	0.583	0.664	0.593
	RFE (XGB)	k=24	24	0.565	0.458	0.649	0.593

Table 8: Group-wise validation-tuned feature selection baselines on ALC. For each behavioral group and method, hyperparameters are selected independently using validation F1. For forward selection and RFE, *Best HP* reports the selected target number of features k; for lasso, it reports the selected regularization parameter C. *Feat.* denotes the number of selected features under the chosen configuration.

Group	Method	Best HP	Feat.	Val F1	Val Acc	Test F1	Test Acc
mid_high	Forward (LR)	k=4	4	0.602	0.600	0.454	0.448
	Lasso (LR)	C=0.3	11	0.569	0.567	0.451	0.448
	RFE (LR)	k=8	8	0.598	0.600	0.356	0.345
	RFE (RF)	k=20	20	0.556	0.567	0.479	0.483
	RFE (XGB)	k=8	8	0.602	0.600	0.351	0.345
mid_low	Forward (LR)	k=4	4	0.251	0.308	0.232	0.296
	Lasso (LR)	C=0.3	10	0.264	0.346	0.234	0.296
	RFE (LR)	k=4	4	0.320	0.346	0.209	0.259
	RFE (RF)	k=8	8	0.377	0.346	0.340	0.333
	RFE (XGB)	k=8	8	0.408	0.346	0.418	0.370
mid_mid	Forward (LR)	k=8	8	0.404	0.429	0.391	0.391
	Lasso (LR)	C=3	21	0.444	0.429	0.523	0.522
	RFE (LR)	k=24	24	0.487	0.464	0.570	0.565
	RFE (RF)	k=20	20	0.542	0.536	0.655	0.652
	RFE (XGB)	k=8	8	0.467	0.464	0.395	0.391
old_high	Forward (LR)	k=20	20	0.557	0.667	0.502	0.621
	Lasso (LR)	C=1	11	0.557	0.667	0.502	0.621
	RFE (LR)	k=12	12	0.557	0.667	0.502	0.621
	RFE (RF)	k=12	12	0.534	0.633	0.502	0.621
	RFE (XGB)	k=4	4	0.491	0.633	0.433	0.586
old_low	Forward (LR)	k=20	20	0.520	0.593	0.515	0.552
	Lasso (LR)	C=1	11	0.447	0.519	0.515	0.517
	RFE (LR)	k=4	4	0.492	0.556	0.515	0.552
	RFE (RF)	k=4	4	0.492	0.556	0.631	0.690
	RFE (XGB)	k=4	4	0.441	0.593	0.608	0.724
old_mid	Forward (LR)	k=4	4	0.700	0.792	0.481	0.625
	Lasso (LR)	C=0.3	3	0.679	0.750	0.391	0.417
	RFE (LR)	k=4	4	0.679	0.750	0.435	0.417
	RFE (RF)	k=4	4	0.700	0.792	0.590	0.667
	RFE (XGB)	k=4	4	0.700	0.792	0.481	0.625
young_high	Forward (LR)	k=32	32	0.569	0.538	0.365	0.357
	Lasso (LR)	C=0.3	10	0.637	0.538	0.254	0.286
	RFE (LR)	k=8	8	0.593	0.538	0.466	0.429
	RFE (RF)	k=16	16	0.551	0.462	0.227	0.214
	RFE (XGB)	k=8	8	0.597	0.500	0.343	0.357
young_low	Forward (LR)	k=32	32	0.451	0.423	0.412	0.400
	Lasso (LR)	C=0.1	3	0.370	0.346	0.434	0.440
	RFE (LR)	k=4	4	0.544	0.538	0.444	0.440
	RFE (RF)	k=4	4	0.483	0.462	0.442	0.440
	RFE (XGB)	k=12	12	0.457	0.423	0.364	0.360
young_mid	Forward (LR)	k=32	32	0.574	0.577	0.445	0.423
	Lasso (LR)	C=1	22	0.544	0.538	0.340	0.308
	RFE (LR)	k=4	4	0.654	0.654	0.668	0.654
	RFE (RF)	k=16	16	0.577	0.577	0.540	0.500
	RFE (XGB)	k=20	20	0.628	0.615	0.540	0.500

Table 9: Group-wise validation-tuned feature selection baselines on MAJ. For each behavioral group and method, hyperparameters are selected independently using validation F1. For forward selection and RFE, *Best HP* reports the selected target number of features k; for lasso, it reports the selected regularization parameter C. *Feat.* denotes the number of selected features under the chosen configuration.

Group	Method	Best HP	Feat.	Val F1	Val Acc	Test F1	Test Acc
mid_high	Forward (LR)	k=4	4	0.548	0.517	0.372	0.281
	Lasso (LR)	C=0.3	10	0.420	0.345	0.398	0.344
	RFE (LR)	k=16	16	0.441	0.379	0.277	0.250
	RFE (RF)	k=4	4	0.595	0.552	0.459	0.375
	RFE (XGB)	k=4	4	0.466	0.448	0.336	0.250
mid_low	Forward (LR)	k=24	24	0.560	0.531	0.440	0.429
	Lasso (LR)	C=0.01	1	0.553	0.594	0.498	0.571
	RFE (LR)	k=12	12	0.507	0.469	0.428	0.429
	RFE (RF)	k=20	20	0.647	0.656	0.382	0.393
	RFE (XGB)	k=16	16	0.554	0.531	0.448	0.464
mid_mid	Forward (LR)	k=24	24	0.570	0.500	0.394	0.375
	Lasso (LR)	C=0.3	18	0.501	0.469	0.404	0.375
	RFE (LR)	k=12	12	0.481	0.438	0.391	0.375
	RFE (RF)	k=4	4	0.506	0.438	0.453	0.438
	RFE (XGB)	k=12	12	0.538	0.469	0.538	0.500
old_high	Forward (LR)	k=4	4	0.797	0.750	0.737	0.688
	Lasso (LR)	C=0.1	2	0.741	0.688	0.727	0.688
	RFE (LR)	k=12	12	0.639	0.562	0.742	0.656
	RFE (RF)	k=4	4	0.701	0.625	0.611	0.531
	RFE (XGB)	k=24	24	0.607	0.531	0.652	0.562
old_low	Forward (LR)	k=4	4	0.781	0.781	0.766	0.719
	Lasso (LR)	C=0.1	2	0.770	0.781	0.828	0.812
	RFE (LR)	k=28	28	0.655	0.594	0.700	0.625
	RFE (RF)	k=4	4	0.752	0.750	0.714	0.656
	RFE (XGB)	k=4	4	0.749	0.781	0.817	0.812
old_mid	Forward (LR)	k=20	20	0.357	0.423	0.622	0.650
	Lasso (LR)	C=1	18	0.492	0.500	0.578	0.600
	RFE (LR)	k=12	12	0.583	0.577	0.578	0.600
	RFE (RF)	k=16	16	0.474	0.423	0.624	0.600
	RFE (XGB)	k=24	24	0.357	0.423	0.622	0.650
young_high	Forward (LR)	k=28	28	0.509	0.469	0.177	0.148
	Lasso (LR)	C=0.01	1	0.557	0.562	0.432	0.407
	RFE (LR)	k=8	8	0.505	0.469	0.501	0.444
	RFE (RF)	k=24	24	0.441	0.406	0.409	0.333
	RFE (XGB)	k=32	32	0.410	0.375	0.564	0.481
young_low	Forward (LR)	k=4	4	0.625	0.594	0.482	0.448
	Lasso (LR)	C=0.01	1	0.601	0.719	0.563	0.690
	RFE (LR)	k=4	4	0.623	0.562	0.563	0.552
	RFE (RF)	k=32	32	0.701	0.688	0.474	0.483
	RFE (XGB)	k=8	8	0.585	0.531	0.508	0.517
young_mid	Forward (LR)	k=24	24	0.305	0.296	0.353	0.292
	Lasso (LR)	C=3	25	0.336	0.296	0.333	0.333
	RFE (LR)	k=24	24	0.381	0.333	0.282	0.250
	RFE (RF)	k=12	12	0.458	0.444	0.337	0.333
	RFE (XGB)	k=28	28	0.313	0.296	0.453	0.458

C LIMITATIONS

We summarize three major limitations of NPO. First, the co-evolution of feature selection and theory-guided decision agents introduces additional computational overhead. Each iteration involves LLM calls and the evaluation of candidate theories, so the cost grows with the theory pool, the number of population groups, and the search budget. Second, NPO uses a finite-budget, LLM-guided search over feature subsets and agent configurations. It does not guarantee a globally optimal configuration, and the selected niche may vary with different initialization settings. Third, NPO aims to improve the effectiveness and transparency of LLM-assisted human decision modeling by combining the structured constraints of traditional behavioral models with the contextual adaptability of LLMs, while jointly refining feature selection. The resulting decision niches are useful modeling configurations supported by the available data, rather than uniquely identified ground-truth accounts of human decision-making.

D BROADER IMPACT

This work may have positive societal impacts by supporting more adaptive and decision-oriented data collection. By co-evolving feature selection with decision modeling, NPO can help identify which information is most useful for predicting and explaining decisions, thereby informing the design of surveys that are better aligned with the target decision task. It can also help researchers understand task- and group-specific decision patterns, revealing the needs, constraints, and behavioral tendencies of different population groups. At the same time, this work may introduce negative societal risks. Because NPO relies on LLM-based agents, biases embedded in the underlying LLMs may influence feature selection, theory alignment, and final decision modeling. If such biases are not carefully audited, the framework may amplify misleading or unfair patterns, especially when applied to sensitive demographic groups or high-stakes decision contexts. Therefore, practical use of NPO should include further bias evaluation and domain-specific validation.

E PROMPT EXAMPLES

Prompt E.1: ***Selection Agent in Current Pipeline***

[SYS]
You are an iterative feature-selection agent for a human behavior prediction pipeline.
Your goal is to propose new features that can improve the model based on observed metrics, error patterns, and theory signals.
You will receive the following information:
`<Static Context></Static Context>`
`<Previous Rounds Memory></Previous Rounds Memory>`
`<Current Round State></Current Round State>`
`<Current Sampled Error Cases></Current Sampled Error Cases>`

INSTRUCTIONS:

1. Only select features from the provided candidate features.
2. Prefer small, targeted updates over large unstable changes.
3. Ground your recommendation in observed error patterns, metrics, and signals.
4. This agent is add-only. Do not suggest removing features.
5. Do not repeat features already in the current feature set.
6. Return valid JSON only.

Your JSON output must contain exactly these keys:

- `decision`
- `features_to_add`
- `reasoning_summary`
- `hypothesis`
- `expected_effect`
- `confidence`

[USR]
Iteration: {`iteration_id`}/{`max_iterations`}
`<Current Round State>`JSON format`</Current Round State>`
`<Current Sampled Error Cases>` JSON format `</Current Sampled Error Cases>`

Task:

1. Focus on the current round state, especially current feature set, metrics, error summary, and signal.
2. Choose concrete new feature names to add for the next run.
3. Only output candidate feature names that appear in the candidate features list from the system prompt.
4. Do not repeat features already in the current feature set.
5. This agent is add-only. Do not suggest removing features.
6. Return valid JSON only.