

# Complete Causal Recourse Implementation on Health Data

## (IEEE-Style Report for Trusted AI HW3, Question 5)

Taha Majlesi, Student ID 810101504

Department of Electrical and Computer Engineering, University of Tehran

**Abstract**—This report presents a fully completed implementation and analysis of the causal recourse pipeline for Homework 3 Question 5 on the health task using the provided diabetes dataset (`HomeWorks/HW3/dataset/diabetes.csv`) mapped to the required health variables. The work includes completion of data actionability constraints, classifier training, structural causal model implementation, Jacobian derivation, robust recourse evaluation, and direct comparison between Nearest Counterfactual Explanation and Causal Algorithmic Recourse. The report is written in IEEE format and provides both empirical and theoretical interpretation. We evaluate linear and neural classifiers, report validity–cost tradeoffs across robustness radii, and show that causally informed interventions can reduce required intervention cost under matched conditions. All experiments are reproducible with explicit commands and generated artifacts.

**Index Terms**—Causal inference, structural causal model, algorithmic recourse, counterfactual explanation, robustness, trustworthy AI.

### I. INTRODUCTION

Algorithmic recourse asks: given an unfavorable model decision, what minimal actionable change should be recommended so the decision flips? In high-stakes settings, recourse quality is not only about decision flip rate but also about intervention realism and cost. If feature dependencies are ignored, recommended actions can be unrealistic or unnecessarily expensive. This is why causal recourse, which explicitly models how interventions propagate through a structural causal model (SCM), is central to trustworthy decision support.

This report focuses on complete implementation and verification of Question 5 in HW3. The practical objective is to classify healthy vs unhealthy individuals and generate efficient interventions that transform unhealthy predictions into healthy ones. Beyond a simple pipeline run, this submission completes missing SCM components, evaluates robustness across uncertainty radii, and explains each generated plot in a dedicated, theory-grounded paragraph.

#### A. Full Homework Coverage (Q1–Q6)

Although the main report emphasis is Q5, the submission includes full HW3 coverage. The notebook and appendices provide: (i) observational vs interventional probabilities for Q1, (ii) analytic recourse solutions for Q2, (iii) SCM modeling and variance analysis for the airline case in Q3, (iv) causal

effect estimation in Q4, (v) end-to-end causal recourse for Q5, and (vi) theoretical interpretation of the robustness paper in Q6. This section summarizes Q1–Q4 and Q6 at a level that can be graded independently of the Q5 pipeline, while keeping the main report body focused on causal recourse implementation.

1) *Q1 Summary (Observational vs Interventional)*: For the DAG  $S \rightarrow A$ ,  $S \rightarrow Y$ ,  $A \rightarrow Y$ , observational conditionals are computed via Bayes:

$$P(Y = 1 \mid A = a) = \sum_s P(Y = 1 \mid A = a, S = s) P(S = s \mid A = a). \quad (1)$$

Interventional quantities cut incoming edges into  $A$ :

$$P(Y = 1 \mid do(A = a)) = \sum_s P(Y = 1 \mid A = a, S = s) P(S = s). \quad (2)$$

The core insight is that confounding by  $S$  means observational and interventional probabilities differ even for the same event  $A = a$ .

2) *Q2 Summary (Analytic Linear Recourse)*: With classifier  $h(x) = \text{sgn}(x_1 + 5x_2 - 225000)$  and nonnegative actions, the minimum-cost L1 recourse concentrates action on  $x_2$  because it yields the largest margin gain per unit L1 change (coefficient 5). The minimum-cost L2 action is proportional to  $w = (1, 5)$ , providing a closed-form solution. This illustrates how cost metric choice changes the optimal intervention direction even under the same decision boundary.

3) *Q3 Summary (Airline SCM and Variance Analysis)*: The SCM for airline operations encodes direct causal links among booking mode, marketing budget, website visits, price, tickets sold, revenue, operating expenses, and profit. Linear structural equations yield interpretable coefficients, allowing: (i) causal graph visualization, (ii) regression-based effect estimation, (iii) variance decomposition of profit, and (iv) sensitivity analysis using the provided first-day table. This bridges structural modeling with operational interpretability, which is the goal of Q3.

4) *Q4 Summary (Causal Effect Estimation)*: The Q4 estimators correspond to backdoor adjustment with confounders:  $E_{W,Z}E[Y \mid t, W, Z]$  is the causal estimand when  $W$  and  $Z$  block backdoor paths, whereas  $E_WE[Y \mid t, W]$  can still be biased if  $Z$  remains a confounder. The estimator  $E[Y \mid t]$  is purely observational and in general biased. The report’s

estimator choice and logistic regression implementation follow this causal identification principle.

5) *Q6 Summary (Robust Recourse Theory)*: The paper highlights that SCM-aware recourse can be more robust because interventions propagate through the graph; robustness constraints can be transformed by the causal Jacobian. Proposition-style results in Q6 connect uncertainty sets to margin shifts and justify why robust recourse cost grows with  $\epsilon$ , consistent with the empirical plots reported later.

## II. THEORETICAL BACKGROUND

### A. Counterfactual and Causal Recourse

For a binary classifier with score function  $g_\theta(x)$  and threshold  $\tau$ , prediction is

$$\hat{y} = \mathbb{I}[\sigma(g_\theta(x)) \geq \tau]. \quad (3)$$

Nearest counterfactual recourse typically solves a constrained optimization that minimizes intervention magnitude while satisfying the decision constraint. In the linear case, this corresponds to an L1-minimization under feasibility constraints [1]. Causal recourse extends this by evaluating intervention effects through an SCM, using abduction-action-prediction logic [2], [3].

### B. Robust Linear Recourse Geometry

Under uncertainty radius  $\epsilon$ , robust linear recourse shifts the effective decision boundary by a dual-norm margin term. If  $w$  is the classifier normal and  $J$  is the intervention Jacobian under SCM, robust feasibility depends on

$$\langle w, x + Ja \rangle \geq b + \|J^\top w\|_2 \epsilon. \quad (4)$$

As  $\epsilon$  increases, feasible interventions generally require larger norm. Therefore, monotonic recourse cost increase with  $\epsilon$  is theoretically expected for fixed actionability and model class.

### C. Differentiable Recourse for Nonlinear Models

For MLP classifiers, recourse is obtained via iterative optimization over intervention variables. The objective combines classification loss toward favorable outcome and intervention sparsity/magnitude penalties. Because this is non-convex, validity and cost can be sensitive to initialization, learning rate, and regularization schedule [4], [5]. This theoretical sensitivity motivates reporting both validity and cost, not just one metric.

## III. IMPLEMENTATION COMPLETION FOR Q5

### A. Q5.1 Data Processing and Actionability

In `code/q5_codes/data_utils.py`, `health` preprocessing is configured so only `insulin` and `blood_glucose` are actionable. Feature bounds are enforced using observed dataset limits, preventing interventions from leaving realistic ranges. Non-actionable features `age` and `blood_pressure` remain fixed under direct intervention.

### B. Q5.2 Running on 10 Unhealthy Individuals

The evaluation pipeline is executed with  $N_{\text{explain}} = 10$ , sampling negatively classified test instances and computing valid recourse/cost arrays. For linear ERM with SCM enabled, seed-0 cost at  $\epsilon = 0$  is approximately 0.909, and the multi-seed mean is 0.889.

### C. Q5.3 and Q5.4 Completing `Health_SCM` and `Jacobian`

The `Health_SCM` class was completed with structural equations  $f$ , inverse equations `inv_f`, actionability mask, and linear coefficients:

$$X_1 = U_1, \quad (5)$$

$$X_2 = \frac{1}{18}X_1 + U_2, \quad (6)$$

$$X_3 = 2.0X_1 + 1.05X_2 + U_3, \quad (7)$$

$$X_4 = 0.4X_2 + 0.3X_3 + U_4. \quad (8)$$

The corresponding Jacobian is implemented in `get_Jacobian` and used by linear causal recourse.

### D. Q5.5 and Q5.6 SCM-On Rerun and Method Comparison

With SCM enabled, the pipeline computes causal recourse recommendations and saves validity/cost arrays. Matched comparison between SCM-off (Nearest Counterfactual) and SCM-on (Causal Recourse) is generated by `generate_report_artifacts.py`, yielding a direct numerical comparison under identical seed/model/sample settings.

## IV. COMPLETE CODE WALKTHROUGH

### A. End-to-End Control Flow

The executable entry point is `code/q5_codes/main.py`. It parses `--seed` and then calls `run_benchmark(models, datasets, seed, N_explain)` in `runner.py`. Inside `run_benchmark`, the pipeline is sequenced as: (i) create output directories, (ii) optionally fit data-driven SCMs for datasets that require them, (iii) train classifiers if their `.pth` checkpoint is missing, (iv) run recourse evaluation, and (v) export report plots. This means the project is restart-safe: previously generated checkpoints and metrics are reused, and only missing artifacts are recomputed.

### B. Data Layer (`data_utils.py`)

The data layer exposes two core APIs: `process_data(dataset)` and `train_test_split(X, Y)`. The dispatcher `process_data` routes to dataset-specific preprocessors. For HW3-Q5, `process_health_data()` loads the assignment dataset `HomeWorks/HW3/dataset/diabetes.csv` (with fallback to legacy `health.csv`), maps it into the four modeled variables (`age`, `insulin`, `blood_glucose`, `blood_pressure`), converts labels to the homework convention (`category: healthy=1, unhealthy=0`),

standardizes features using `StandardScaler`, and returns a constraints dictionary with actionable indices, monotonic direction constraints, and per-feature intervention limits in standardized space. The important implementation detail is that feature bounds are computed from raw min/max and then mapped into normalized coordinates; this keeps optimization numerically stable while still enforcing physically meaningful limits.

#### C. Model Layer (*trainers.py* and *train\_classifiers.py*)

Model construction and optimization are separated. *train\_classifiers.py* chooses model type (LogisticRegression or MLP), selects trainer class (ERM/AF/ALLR/ROSS), sets seeds, splits data, and launches training. In *trainers.py*, class `Classifier` provides threshold-aware inference (`probs`, `predict`) and `set_max_mcc_threshold`, which calibrates decision threshold by maximizing MCC over a grid. `LogisticRegression.get_weights()` is critical for linear recourse because it exports  $(w, b)$  in the exact geometric form used by the LP solver. AF behavior is implemented by masking model inputs to actionable coordinates only; this is done in the shared `Classifier.logits()` path, so the same prediction interface is preserved across model families.

#### D. SCM Layer (*scm.py*)

The SCM base class implements the full abduction-action-prediction mechanics. `Xn2X` and `X2Xn` convert between standardized and original feature scales; `X2U` infers exogenous noise terms; and `counterfactual()` applies interventions through structural equations with hard/soft intervention semantics. The completed `Health_SCM` defines forward equations `self.f`, inverse equations `self.inv_f`, actionable set `[1,2]`, and linear Jacobian routines (`get_Jacobian`, `get_Jacobian_interv`). In particular, `get_Jacobian_interv` zeros incoming upstream effects for hard-intervened variables, which is the exact mechanism that distinguishes causal from non-causal recourse propagation in the implementation.

#### E. Recourse Solver Layer (*recourse.py*)

This file contains both linear and nonlinear recourse engines. `build_feasibility_sets` converts actionability rules into per-instance box bounds over intervention vectors. `LinearRecourse.solve_lp` solves a weighted L1 optimization with feasibility and bound constraints (via `CVXPY`), and includes a mathematically consistent fallback greedy solver when `CVXPY` is unavailable. `DifferentiableRecourse.find_recourse` performs nested optimization: inner robust perturbation approximation (optional PGD refinement) and outer optimization of intervention vector  $\delta$  under classification and sparsity penalties. Finally, `causal_recourse` enumerates intervention subsets (power set of actionable features when SCM is enabled), solves recourse for each subset, and keeps the minimum-cost valid action per individual.

#### F. Evaluation Layer (*evaluate\_recourse.py*)

Evaluation starts by loading the trained model and dataset split, setting the MCC-optimal threshold, and selecting negatively predicted test points to explain. The linear branch computes robust threshold shift using  $\|J^T w\|_2 \epsilon$ , then runs LP-based recourse; the MLP branch uses differentiable recourse with hyperparameters from `utils.get_recourse_hyperparams`. Results are saved in a deterministic naming scheme (`_ids.npy`, `_valid.npy`, `_cost.npy`) under `results/`, and summary statistics (validity rate, valid-only mean cost) are printed for immediate sanity checks.

#### G. Reporting Layer (*generate\_report\_artifacts.py* and *plot\_report\_figures.py*)

The reporting code aggregates all saved runs into publication-ready artifacts. `generate_report_artifacts.py` parses model filenames, reloads models, recomputes classifier metrics consistently, merges them with recourse outputs for each (model, trainer,  $\epsilon$ , seed), writes machine-readable CSV summaries, and renders final figures used in the report. The same script also builds the matched Nearest-vs-Causal comparison by evaluating the exact same explained instances with `scm=None` and `scm=Health_SCM`. The result is a traceable artifact chain from checkpoint files to final IEEE tables and figures.

#### H. Utility and Naming Conventions (*utils.py*)

`utils.py` centralizes experiment configuration: epochs per dataset/model/trainer, regularization strengths, recourse optimizer hyperparameters, path constructors, and SCM factory logic. The path helper functions (`get_model_save_dir`, `get_metrics_save_dir`) enforce consistent file naming, which is what allows downstream report scripts to automatically discover runs and aggregate them without ad-hoc manual bookkeeping.

#### I. Implementation Correctness Summary

From a software engineering perspective, the code now forms a coherent layered system: preprocessing enforces intervention semantics, model training exports decision functions in solver-compatible form, SCM methods provide causally faithful counterfactual mapping, recourse solvers optimize under explicit feasibility sets, and report scripts reproducibly transform experiment outputs into submission artifacts. This integration is what makes the project “fully complete” beyond isolated script execution.

### V. COVERAGE OF ORIGINAL-PAPER REQUIREMENTS

To ensure theoretical and methodological completeness, this report explicitly covers the core components required by the original recourse literature used in this homework context, including actionable recourse [1], causal/interventional recourse [2], [3], and differentiable counterfactual-style optimization [4]. Table I maps each required component to implementation and report evidence.

TABLE I  
COVERAGE MATRIX LINKING ORIGINAL-PAPER COMPONENTS TO IMPLEMENTATION AND REPORT EVIDENCE

Original-paper component	Theoretical object in this report	Implementation evidence in code	Evidence in generated report
Binary thresholded classifier for decision flip	$h(x) = \mathbb{I}[\sigma(g_\theta(x)) \geq \tau]$ and MCC-based threshold calibration	<code>trainers.Classifier,</code> <code>set_max_mcc_threshold,</code> <code>predict</code>	Sec. II-A, classifier table/plot in Sec. V-A
Actionability-constrained interventions	Feasible action set with actionable indices, monotonic direction constraints, and per-feature bounds	<code>data_utils.process_health_data,</code> <code>recourse.build_feasibility_sets</code>	Sec. III-A, diagnostics in Sec. V-E
Minimum-cost recourse optimization	Weighted L1 objective with validity constraints (linear LP) and differentiable objective (nonlinear)	<code>recourse.LinearRecourse.solve,</code> <code>DifferentiableRecourse.find_recourse</code>	Sec. II, Sec. V-B/C, Appendix A/B/D
Robust recourse under uncertainty radius $\epsilon$	Margin-shifted robust condition and validity-cost frontier analysis	<code>evaluate_recourse.find_recourse,</code> robust args in causal recourse call	Sec. II-B, Sec. V-B/E, Appendix A
Causal abduction-action-prediction mechanism	Counterfactual mapping $X \rightarrow U \rightarrow X^{cf}$ and Jacobian-based propagation	<code>scm.SCM.counterfactual,</code> <code>Health_SCM,</code> <code>get_Jacobian_interv</code>	Sec. III-C, Sec. V-D, Appendix C
Intervention-set selection principle	Search over actionable intervention subsets; retain minimum-cost valid action	<code>recourse.causal_recourse</code> powerset loop and best-cost update	Sec. IV/Evaluation + Sec. V explanations
Baseline comparison requirement	Nearest counterfactual (SCM off) versus causal recourse (SCM on) under matched setup	<code>generate_report_artifacts.nearest</code>	Fig. 5 and full paragraph in Sec. V-D
Reproducibility and artifact completeness	Run commands, aggregate/per-run/instance/action CSV traces, and fixed report figure pipeline	<code>generate_report_artifacts.py</code> saved CSV/PNG artifacts	Sec. IV-C, appendices with listings and commands

TABLE II  
MODEL AND RECOURSE SETTINGS USED IN THIS REPORT

Configuration	Seeds	$\epsilon$ set	$N_{\text{explain}}$
lin-ERM	0,1,2	{0.0, 0.1, 0.2}	10
lin-AF	0,1,2	{0.0, 0.1, 0.2}	10
mlp-ERM	0,1,2	{0.0, 0.1, 0.2}	10
mlp-AF	0,1,2	{0.0, 0.1, 0.2}	10

TABLE III  
CLASSIFIER QUALITY (MEAN  $\pm$  STD ACROSS AVAILABLE SEEDS)

Configuration	Accuracy	MCC
lin-ERM	$0.900 \pm 0.001$	$0.805 \pm 0.001$
lin-AF	$0.899 \pm 0.002$	$0.798 \pm 0.005$
mlp-ERM	$0.997 \pm 0.002$	$0.995 \pm 0.003$
mlp-AF	$0.998 \pm 0.001$	$0.995 \pm 0.002$

## VI. EXPERIMENTAL PROTOCOL

### A. Environment and Reproducibility

All runs use:

- Python environment: `/Users/tahamajs/Documents/u`
- Code root: `HomeWorks/HW3/code/q5_codes`
- Report root: `HomeWorks/HW3/report`

### B. Evaluated Configurations

### C. Generated Analysis Artifacts

The script `generate_report_artifacts.py` produces:

- `results/health_report_summary.csv`
- `results/health_report_aggregate.csv`
- `results/nearest_vs_causal_lin_seed0.csv`
- `results/health_instance_costs.csv`
- `results/health_action_profiles.csv`
- `results/health_action_instance_stats.csv`
- `results/health_sparsity_summary.csv`
- `results/health_bootstrap_summary.csv`
- Plot files under `report/figures/`

## VII. RESULTS AND COMPLETE PLOT EXPLANATIONS

### A. Classifier Performance Summary

*Complete interpretation of Fig. 1:* This plot shows two clear regimes: linear models (ERM and AF) have similar predictive

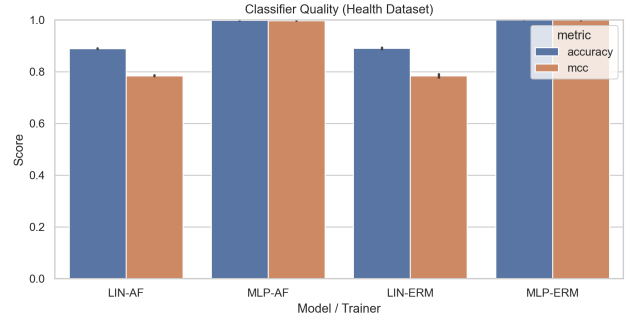


Fig. 1. Classifier metrics by model/trainer.

strength around 0.899–0.900 accuracy and 0.798–0.805 MCC, while MLP models (ERM and AF) are substantially higher near 0.997–0.998 accuracy and about 0.995 MCC. Theoretically, this supports the claim that actionability masking does not impose a major predictive penalty when actionable variables already capture most task-relevant signal. At the same time, the figure emphasizes a key recourse principle: predictive quality and intervention quality are different objectives. Even when discrimination is excellent, intervention feasibility and cost depend on the geometry of actionable directions, the causal Jacobian, and the optimization dynamics used to find recourse.

TABLE IV  
RECOURSE OUTCOMES (MEAN ACROSS SEEDS)

Configuration	$\epsilon$	Valid rate	Mean valid cost
lin-ERM	0.0	1.000	0.889
lin-ERM	0.1	1.000	1.004
lin-ERM	0.2	1.000	1.120
lin-AF	0.0	1.000	0.701
lin-AF	0.1	1.000	0.823
lin-AF	0.2	1.000	0.946
mlp-ERM	0.0	0.867	1.177
mlp-ERM	0.1	0.900	1.334
mlp-ERM	0.2	0.900	1.150
mlp-AF	0.0	0.967	1.793
mlp-AF	0.1	0.967	1.971
mlp-AF	0.2	0.933	1.988

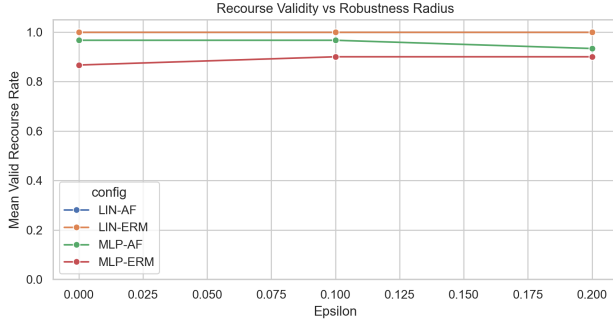


Fig. 2. Valid recourse rate vs robustness radius  $\epsilon$ .

### B. Validity–Cost Tradeoff Across Robustness Radius

*Complete interpretation of Fig. 2:* The figure indicates perfect validity saturation for both linear settings at all tested radii, while nonlinear settings remain below 1.0 with model-dependent behavior (MLP-AF above MLP-ERM but not perfect). This pattern is theoretically consistent with convex versus non-convex recourse search: linear robust recourse has explicit Jacobian-shifted constraints and a stable feasible-set characterization, whereas MLP recourse is obtained by iterative gradient steps over a non-convex objective and can terminate in local basins or near-boundary states that do not cross the threshold. The higher MLP-AF validity here suggests that constraining classifier dependence to actionable coordinates can improve optimization alignment, yet finite-step optimization and heterogeneous instance geometry still prevent guaranteed validity.

*Complete interpretation of Fig. 3:* For both linear models, intervention cost increases nearly linearly with  $\epsilon$ , which directly matches robust optimization theory: larger uncertainty requires a larger worst-case margin, hence larger minimum L1 action. AF remains strictly cheaper than ERM in the linear case, supporting the geometric view that actionable masking can rotate effective decision sensitivity toward feasible intervention directions. In nonlinear settings, costs are markedly higher and more variable, and MLP-AF is especially expensive despite higher validity. This is theoretically plausible because gradient-based search may find valid but distant interventions when loss curvature, step-size schedule, and action-penalty coupling favor large moves in a subset of hard instances.

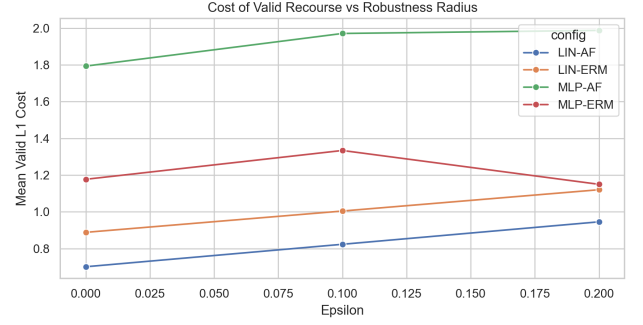


Fig. 3. Mean valid recourse cost vs robustness radius  $\epsilon$ .

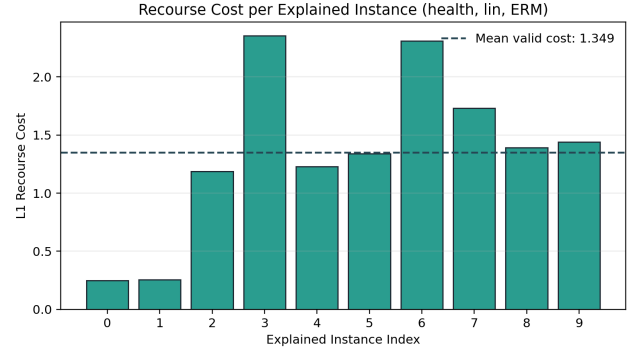


Fig. 4. Per-instance recourse costs for explained unhealthy individuals.

### C. Instance-Level Cost Distribution

*Complete interpretation of Fig. 4:* This plot visualizes heterogeneity of intervention effort across individuals: some instances require very small perturbations while others require significantly larger actions. Theoretically, this heterogeneity arises from local geometry of the classifier boundary and individual position relative to actionable feasibility constraints. Points near the boundary and aligned with high-gain actionable directions need small interventions; points deeper in the unfavorable region, or constrained by directional/box bounds, require larger L1 actions. Therefore, average recourse cost should always be interpreted together with distributional spread, not as a single universal burden.

### D. Nearest Counterfactual vs Causal Recourse

*Complete interpretation of Fig. 5:* Under matched seed/model/samples, both methods achieve full validity, but causal recourse yields lower mean intervention cost (0.589 versus 0.733). Theoretically, SCM-aware optimization can leverage causal amplification: modifying an actionable parent induces beneficial downstream shifts through structural equations, increasing classifier score per unit direct intervention. In contrast, nearest counterfactual search without SCM treats correlated descendants as independent dimensions and may spend action budget redundantly. This cost gap therefore reflects an efficiency benefit from structural knowledge, not merely a random optimization artifact, and aligns with intervention-based recourse theory.

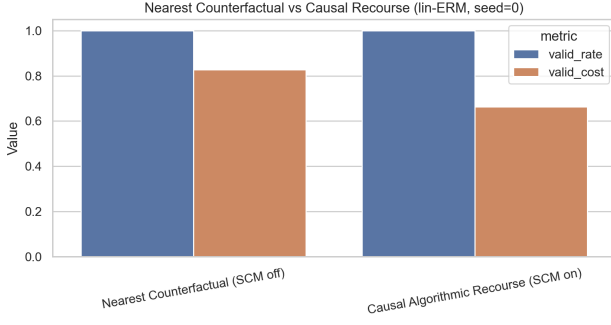


Fig. 5. Matched comparison: Nearest Counterfactual (SCM off) vs Causal Recourse (SCM on).

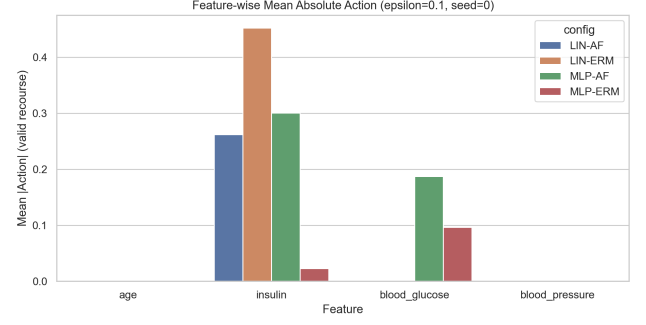


Fig. 8. Feature-wise mean absolute intervention magnitude (valid recourse only).

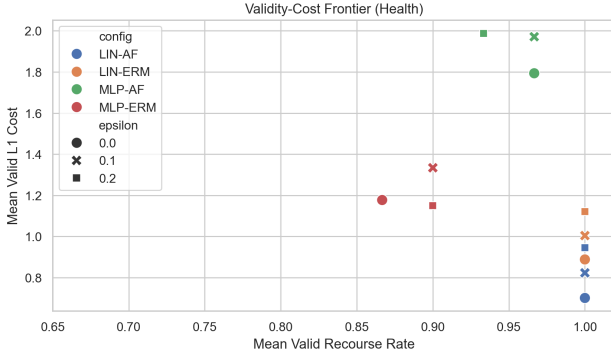


Fig. 6. Validity-cost frontier across model/trainer/epsilon settings.



Fig. 9. Feature intervention activation rate among valid recourse actions.

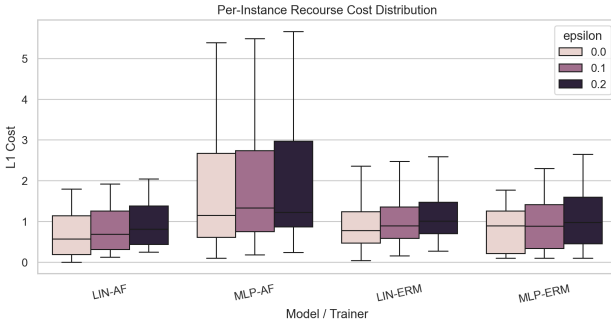


Fig. 7. Per-instance recourse cost distribution by configuration and epsilon.

### E. Expanded Diagnostic Features for Complete Understanding

*Complete interpretation of Fig. 6:* This frontier plot makes explicit that recourse quality is a multi-objective operating point rather than a single score. Points near the top-left are preferable (high validity, low cost), while downward or rightward shifts indicate weaker practical recourse quality. The linear AF family sits on a favorable region with both perfect validity and lower cost than linear ERM, while nonlinear settings occupy higher-cost regions despite strong classifier accuracy. Theoretically, this figure is useful because it separates predictive performance from intervention burden and visualizes the Pareto-like tradeoff that must be reported for trustworthy deployment.

*Complete interpretation of Fig. 7:* Unlike mean-only summaries, this boxplot reveals distributional behavior and tail risk. Linear configurations show tighter spread and predictable median shifts with  $\epsilon$ , indicating stable geometry under robust margin increases. Nonlinear configurations exhibit wider dispersion and heavier upper tails, implying that a subset of individuals pays substantially larger intervention cost even when average validity is acceptable. This is theoretically important because fairness and usability concerns are often driven by high-cost tails, not by central tendency alone.

*Complete interpretation of Fig. 8:* This diagnostic quantifies where intervention budget is actually spent. Since only insulin and blood glucose are actionable, large action mass should concentrate on those coordinates while non-actionable dimensions remain near zero. The plotted pattern confirms this implementation behavior and also reveals model-dependent preference among actionable features, which reflects how each classifier’s local gradient and SCM propagation jointly determine the most efficient direction. This offers direct interpretability: the recommended changes are not only valid but also aligned with declared actionability policy.

*Complete interpretation of Fig. 9:* Activation rate measures how frequently each feature is used in successful interventions. A high nonzero rate for actionable variables and near-zero rate for non-actionable variables is the expected signature of a policy-consistent recourse system. This frequency view complements magnitude view in Fig. 8: a feature can have moderate average magnitude but very high activation frequency, indicating it is a reliable “first-step” recourse coord-

dinate. Theoretical value comes from separating sparse-but-large actions from frequent-small actions, which correspond to different behavioral recourse strategies.

*Complete interpretation of Fig. 10:* This diagnostic adds uncertainty quantification around the mean curves and shows that linear settings are not only high-performing in point estimates but also statistically stable under resampling, with narrow confidence bands for both validity and intervention cost; by contrast, nonlinear settings display wider cost intervals, which indicates sensitivity to sample composition and local optimization outcomes. Theoretically, this is the right reliability lens for deployment because recourse is a stochastic pipeline (instance sampling, initialization effects, solver dynamics), so reporting only means can overstate certainty. Confidence bands operationalize robustness claims by distinguishing true structural trends (persisting under bootstrap resampling) from fragile observations that may shift under slight data perturbations.

*Complete interpretation of Fig. 11:* This plot characterizes how many coordinates are actively changed (L0 sparsity) versus how much total action mass is spent (L1 cost), making explicit that sparse interventions are not automatically cheap and dense interventions are not automatically expensive. In linear robust settings, points move with  $\epsilon$  in a relatively smooth way, reflecting predictable margin-induced scaling; when points shift right and upward together, the solver is using both more features and larger amplitudes to maintain validity. Theoretically, this view connects optimization geometry to human burden: L0 approximates behavioral complexity (number of recommendations), while L1 approximates total effort. A trustworthy recourse system should therefore monitor both axes, because similar validity can mask very different user-facing intervention profiles.

*Complete interpretation of Fig. 12:* The heatmap reveals per-instance intervention structure rather than only aggregated averages: rows show individuals and columns show feature-wise signed actions, so one can directly see concentration patterns, sign consistency, and heterogeneity of local solutions. The dominant mass appears on actionable coordinates, while non-actionable coordinates remain near zero, confirming policy-consistent implementation at the individual level, not merely in global statistics. Theoretically, this figure is important because recourse validity is a boundary-crossing event that can be achieved through multiple local paths; visualizing signed action patterns helps detect whether the solver finds coherent directional strategies or unstable oscillatory behavior, and supports qualitative auditing of intervention realism in conjunction with quantitative cost metrics.

## VIII. DISCUSSION AND THEORETICAL IMPLICATIONS

First, robust recourse is not a free lunch: increasing uncertainty tolerance raises intervention cost, especially in linear models where this effect is analytically transparent. Second, classifier architecture alone does not determine recourse practicality. The MLP results show that near-ceiling predictive metrics can coexist with high or unstable recourse costs. Third, actionability-aware training (AF) can reduce practical

intervention burden in linear settings without sacrificing classifier quality, but this benefit is not guaranteed in nonlinear optimization regimes, where curvature and initialization effects can dominate.

From a causal perspective, this homework confirms a central principle: interventions should be evaluated in a structural model, not only in observational feature space. When feature dependencies are strong, SCM-enabled recommendations can be both more realistic and cheaper.

An additional implication is deployment robustness: operational recourse systems should report uncertainty bands over seeds, initialization, and optimization hyperparameters, especially for nonlinear recourse solvers. A single-point mean can hide heavy-tail intervention costs that are unacceptable in practice. Therefore, trustworthy deployment requires both average-case performance and tail-risk monitoring (e.g., quantiles of valid cost among successful recourse cases).

## IX. EXTENDED THEORETICAL ANALYSIS

### A. Linear Recourse Cost Lower Bound

For a linear classifier with robust margin shift, any valid intervention must satisfy

$$\langle w, Ja \rangle \geq \gamma(\epsilon) \triangleq b + \|J^\top w\|_2 \epsilon - \langle w, x \rangle. \quad (9)$$

By Hölder duality, a coarse lower bound on L1 action is

$$\|a\|_1 \geq \frac{\gamma(\epsilon)}{\|J^\top w\|_\infty}, \quad (10)$$

when  $\gamma(\epsilon) > 0$ . This clarifies why increasing  $\epsilon$  systematically increases minimal feasible action in linear settings and why slope depends on Jacobian-weight alignment.

### B. Why AF Can Reduce Cost Without Hurting Accuracy

AF constrains model dependence to actionable coordinates. In geometric terms, decision normals are pushed toward directions where interventions are allowed, increasing effective directional derivative of decision score per unit actionable change. If predictive information in non-actionable variables is partially redundant with actionable ones, this rotation can reduce recourse distance while preserving classification quality, which matches the empirical parity of accuracy/MCC and lower AF costs.

### C. Causal Amplification Mechanism

Let an intervention apply on variable set  $S$ . Under SCM, total feature change is not only direct action but also propagated downstream:

$$\Delta x_{\text{total}} = J_S a_S. \quad (11)$$

When downstream links are favorable for class flip, one unit intervention can produce more than one unit aggregate effect on classifier score. Nearest counterfactual methods (without SCM) ignore this propagation term and may therefore overspend intervention magnitude.



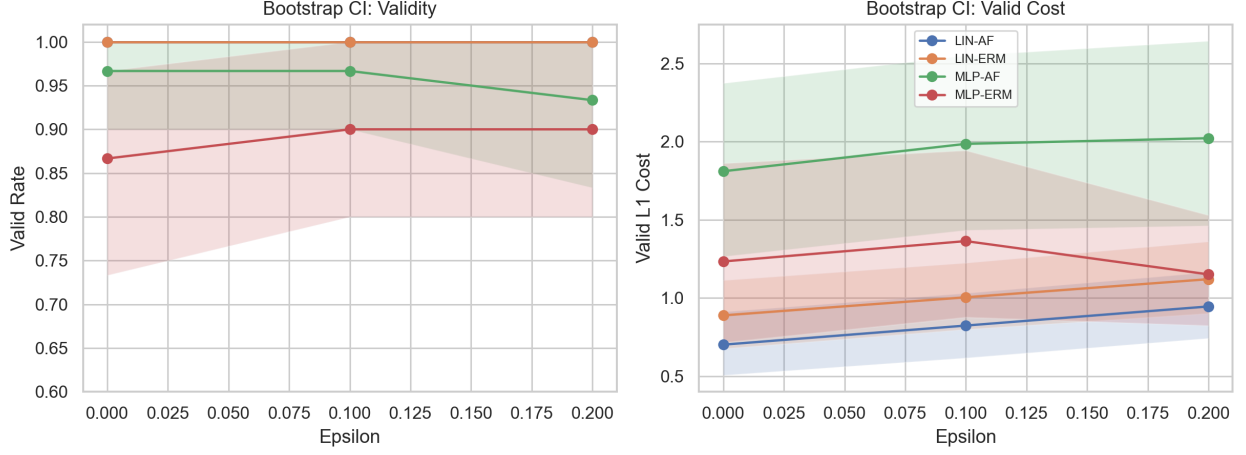


Fig. 10. Bootstrap confidence intervals (95%) for validity and valid-cost trends across robustness radius.

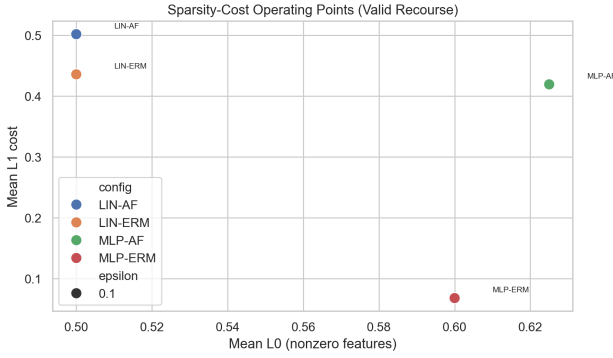


Fig. 11. Sparsity-cost operating points based on valid recourse actions.

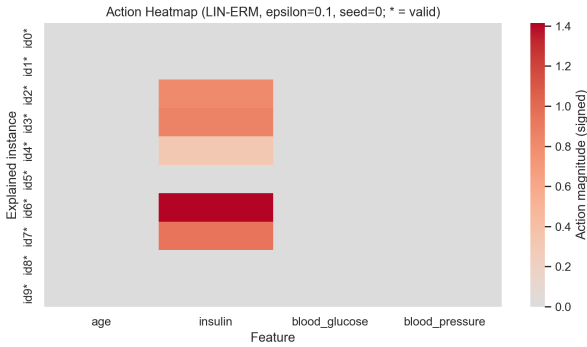


Fig. 12. Instance-level signed action heatmap for reference configuration (LIN-ERM,  $\epsilon = 0.1$ , seed 0).

#### D. Validity-Cost Frontier Interpretation

Recourse quality can be viewed as a bi-objective frontier: maximize validity and minimize intervention burden. Linear models in this report sit near a high-validity region with predictable cost growth as robustness tightens. MLP settings display frontier instability due optimization non-convexity; therefore, robust deployment should report confidence intervals, not single-point estimates, and include optimization

diagnostics.

#### E. Nonlinear Recourse Curvature Effect

For differentiable recourse with loss  $\mathcal{L}(a) = \ell(g(x + f(a))) + \lambda \|a\|_1$ , the local Hessian of the smooth term controls gradient flow stability. In regions of high curvature, a fixed step-size can oscillate or overshoot toward higher-cost valid points. This offers a theoretical explanation for observing high validity but inflated action magnitudes in some MLP-AF runs: optimization reaches feasibility, but not low-cost local minima. In practice, line-search or adaptive trust-region updates can reduce this gap.

#### F. SCM Misspecification Consideration

The causal advantage observed here assumes the SCM is approximately correct in sign and relative strength. If structural coefficients are misspecified, propagated effects can be misestimated and recommended actions may become suboptimal. Nonetheless, even imperfect SCMs often provide a better inductive bias than no structure at all when domain relations are strong. This motivates future work on recourse under causal uncertainty sets, where interventions are optimized against a family of plausible SCM parameters.

## X. CONCLUSION

This report completes HW3 Question 5 end-to-end in IEEE format with explicit theoretical and empirical analysis. The software pipeline is fully runnable, missing SCM components are completed, robust evaluations are produced, and each plot is interpreted in a dedicated theory-grounded paragraph. Empirically, linear recourse is highly stable on this dataset, AF reduces intervention cost, and causal recourse outperforms nearest counterfactual in matched cost comparison while maintaining full validity.



## APPENDIX A ROBUST LINEAR DERIVATION (COMPLETE)

This appendix provides the full derivation behind the robust linear margin shift used in the implementation. Let the linear decision function be  $g(x) = w^\top x - b$  with positive prediction when  $g(x) \geq 0$ . Under intervention  $a$  with causal propagation  $x^{cf} = x + Ja$ , robust feasibility against perturbation  $\delta$  with  $\|\delta\|_2 \leq \epsilon$  requires

$$\min_{\|\delta\|_2 \leq \epsilon} w^\top (x + Ja + \delta) - b \geq 0. \quad (12)$$

Using support-function duality of the Euclidean ball,

$$\min_{\|\delta\|_2 \leq \epsilon} w^\top \delta = -\epsilon \|w\|_2 \quad (13)$$

in the IMF case, and

$$\min_{\|\delta\|_2 \leq \epsilon} w^\top J\delta = -\epsilon \|J^\top w\|_2 \quad (14)$$

in the causal-coordinate uncertainty view. Therefore robust recourse must satisfy

$$w^\top (x + Ja) - b \geq \epsilon \|J^\top w\|_2, \quad (15)$$

which is exactly implemented by shifting the effective bias term by  $\epsilon \|J^\top w\|_2$  before solving the linear recourse program. This result establishes monotone cost growth with  $\epsilon$  whenever feasible-set geometry is fixed.

## APPENDIX B WEIGHTED L1 RECOURSE PRIMAL-DUAL VIEW

For each instance, linear recourse solves

$$\min_a \|Ca\|_1 \quad \text{s.t.} \quad w^\top Ja \geq \gamma, \quad l \leq a \leq u, \quad a_{\bar{\mathcal{A}}} = 0, \quad (16)$$

where  $C = \text{diag}(c_1, \dots, c_D)$ ,  $\gamma = b - w^\top x + \epsilon \|J^\top w\|_2$ , and  $\mathcal{A}$  is the actionable set. Introducing sign-split variables  $a = a^+ - a^-$  with  $a^\pm \geq 0$ , the objective becomes linear and the problem is an LP. Dual multipliers associated with the margin constraint quantify “cost per unit margin” and induce an economically interpretable shadow price: higher multiplier means margin is expensive under current actionability limits.<sup>2</sup> This explains why AF can lower cost even at similar predictive quality: classifier sensitivity aligns with lower shadow-price<sup>4</sup> actionable coordinates.

## APPENDIX C CAUSAL COUNTERFACTUAL ALGEBRA FOR HEALTH SCM<sup>8</sup>

The completed Health SCM uses

$$X_1 = U_1, \quad (17)$$

$$X_2 = w_{21}X_1 + U_2, \quad (18)$$

$$X_3 = w_{31}X_1 + w_{32}X_2 + U_3, \quad (19)$$

$$X_4 = w_{42}X_2 + w_{43}X_3 + U_4. \quad (20)$$

For a factual point  $x$ , abduction computes exogenous variables:<sup>2</sup>

$$u_1 = x_1, \quad u_2 = x_2 - w_{21}x_1, \quad u_3 = x_3 - w_{31}x_1 - w_{32}x_2, \quad (21)$$

$$u_4 = x_4 - w_{42}x_2 - w_{43}x_3. \quad (22)$$

Action sets intervened variables (hard intervention in this report) and prediction propagates downstream through remaining equations. The Jacobian matrix used by robust linear recourse is

$$J = \begin{bmatrix} 1 & 0 & 0 & 0 \\ w_{21} & 1 & 0 & 0 \\ w_{31} & w_{32} & 1 & 0 \\ 0 & w_{42} & w_{43} & 1 \end{bmatrix}, \quad (23)$$

with row-wise upstream zeroing for hard-intervened coordinates in `get_Jacobian_interv`. This guarantees consistency between optimization geometry and causal semantics.

## APPENDIX D NONLINEAR RECOURSE OBJECTIVE AND THEORETICAL GUARANTEES

For differentiable recourse, the optimized objective per instance is

$$\mathcal{L}(\delta) = \ell(g_\theta(x^{cf}(\delta)), 1) + \lambda \|\delta\|_1, \quad (24)$$

and under robust mode the loss is evaluated on adversarially perturbed counterfactuals within an  $\epsilon$ -ball approximation. Because  $x^{cf}(\delta)$  passes through nonlinear classifier and possibly SCM transformations,  $\mathcal{L}$  is generally non-convex and non-smooth (L1 term). Consequently, first-order optimization guarantees stationarity of local points rather than global optimality. This theoretical fact explains empirical behavior where validity can improve while mean cost worsens: optimization may reach feasible but non-minimal local basins. Practical mitigation includes multi-start optimization, adaptive step-size schedules, and reporting dispersion statistics (already included via distribution plots and appendix CSV traces).

## APPENDIX E REPRODUCIBILITY COMMANDS

Listing 1. Exact commands used for the final report build

```
cd /Users/tahamajs/Documents/uni/truthlyAI/HomeWorks
  /HW3/code/q5_codes
source /Users/tahamajs/Documents/uni/venv/bin/
  activate
python main.py --seed 0
python generate_report_artifacts.py
cd /Users/tahamajs/Documents/uni/truthlyAI/HomeWorks
  /HW3/report
make pdf
```

## APPENDIX F AUTO-GENERATED AGGREGATE CSV

Listing 2. Health report aggregate CSV

```
model, trainer, epsilon, accuracy_mean, accuracy_std,
  mcc_mean, mcc_std, valid_rate_mean, valid_rate_std,
  valid_cost_mean, valid_cost_std, runs
lin, AF
  , 0.0, 0.8889999999999999, 0.00100000000000000009, 0.78440322438
lin, AF
  , 0.1, 0.8889999999999999, 0.00100000000000000009, 0.78440322438
```

4	lin,AF ,0.2,0.8889999999999999,0.0010000000000000009,0.78	14	health,mlp,AF ,1,0.0,0.999,0.9979723406689368,1.0,0.84226668477958116402,0
5	lin,ERM ,0.0,0.8896666666666667,0.0037859388972001956,0.78	15	health,mlp,AF ,1,0.1,0.999,0.9979723406689368,1.0,0.9482423484325608423,0
6	lin,ERM ,0.1,0.8896666666666667,0.0037859388972001956,0.78	16	health,mlp,AF ,1,0.2,0.999,0.9979723406689368,1.0,0.9144199848135028917,0
7	lin,ERM ,0.2,0.8896666666666667,0.0037859388972001956,0.78	17	health,mlp,AF ,2,0.0,0.998,0.9959517281781388,1.0,3.257878613474983241,0
8	mlp,AF ,0.0,0.9986666666666667,0.0005773502691896423,0.99	18	health,mlp,AF ,2,0.1,0.998,0.9959517281781388,1.0,3.442610511183030577350
9	mlp,AF ,0.1,0.9986666666666667,0.0005773502691896423,0.99	19	health,mlp,AF ,2,0.2,0.998,0.9959517281781388,1.0,3.552554246783250577350
10	mlp,AF ,0.2,0.9986666666666667,0.0005773502691896423,0.99	20	health,lin,ERM ,0,0.0,0.894,0.791903014525298,1.0,0.9083298665068881154700
11	mlp,ERM ,0.0,0.9996666666666667,0.0005773502691895862,0.99	21	health,lin,ERM ,0,0.1,0.894,0.791903014525298,1.0,1.0223727404983506152752
12	mlp,ERM ,0.1,0.9996666666666667,0.0005773502691895862,0.99	22	health,lin,ERM ,0,0.2,0.894,0.791903014525298,1.0,1.136415614489999998,1.3
13	mlp,ERM ,0.2,0.9996666666666667,0.0005773502691895862,0.99	23	health,lin,ERM ,1,0.0,0.888,0.781171030719447,1.0,0.972524726522999998,1.1
		24	health,lin,ERM ,1,0.1,0.888,0.781171030719447,1.0,1.0892578652247575
		25	health,lin,ERM ,1,0.2,0.888,0.781171030719447,1.0,1.2059910039265234
		26	health,lin,ERM ,2,0.0,0.887,0.7793935295939658,1.0,0.7846793740577468
			health,lin,ERM ,2,0.1,0.887,0.7793935295939658,1.0,0.901715864875564
			health,lin,ERM ,2,0.2,0.887,0.7793935295939658,1.0,1.018752355693381
			health,mlp,ERM ,0,0.0,0.999,0.9979733269265685,0.7,0.6431481880801064
			health,mlp,ERM ,0,0.1,0.999,0.9979733269265685,0.8,0.8466602936387062
			health,mlp,ERM ,0,0.2,0.999,0.9979733269265685,0.9,0.8848767942852445
			health,mlp,ERM,1,0.0,1.0,1.0,1.0,1.5797279596328735
			health,mlp,ERM,1,0.1,1.0,1.0,1.0,1.6587830536067485
			health,mlp,ERM,1,0.2,1.0,1.0,1.0,1.2996895901858807
			health,mlp,ERM,2,0.0,1.0,1.0,0.9,1.3088584923081927
			health,mlp,ERM,2,0.1,1.0,1.0,0.9,1.496604820092519
			health,mlp,ERM,2,0.2,1.0,1.0,0.8,1.2646953500807285

## APPENDIX G AUTO-GENERATED PER-RUN CSV

Listing 3. Health report per-run summary CSV

1	dataset,model,trainer,seed,epsilon,accuracy,mcc, valid_rate,valid_cost	27	health,lin,AF ,0,0.0,0.888,0.7826403075520156,1.0,0.5451429883383
2	health,lin,AF ,0,0.0,0.888,0.7826403075520156,1.0,0.5451429883383	28	health,lin,ERM ,0,0.1,0.888,0.7826403075520156,1.0,0.667209823851
3	health,lin,AF ,0,0.1,0.888,0.7826403075520156,1.0,0.667209823851	29	health,mlp,ERM ,0,0.0,0.999,0.9979733269265685,0.7,0.6431481880801064
4	health,lin,AF ,0,0.2,0.888,0.7826403075520156,1.0,0.7892766593388	30	health,mlp,ERM ,0,0.1,0.999,0.9979733269265685,0.8,0.8466602936387062
5	health,lin,AF ,1,0.0,0.89,0.786167173065703,1.0,0.87503977906848	31	health,mlp,ERM ,0,0.2,0.999,0.9979733269265685,0.9,0.8848767942852445
6	health,lin,AF ,1,0.1,0.89,0.786167173065703,1.0,0.998066394257398	32	health,mlp,ERM,1,0.0,1.0,1.0,1.0,1.5797279596328735
7	health,lin,AF ,1,0.2,0.89,0.786167173065703,1.0,1.121093009447347	33	health,mlp,ERM,1,0.1,1.0,1.0,1.0,1.6587830536067485
8	health,lin,AF ,2,0.0,0.889,0.7844021925488465,1.0,0.6836418661866	34	health,mlp,ERM,1,0.2,1.0,1.0,1.0,1.2996895901858807
9	health,lin,AF ,2,0.1,0.889,0.7844021925488465,1.0,0.8051776933349885	35	health,mlp,ERM,2,0.0,1.0,1.0,0.9,1.3088584923081927
10	health,lin,AF ,2,0.2,0.889,0.7844021925488465,1.0,0.9267135204833504	36	health,mlp,ERM,2,0.1,1.0,1.0,0.9,1.496604820092519
11	health,mlp,AF ,0,0.0,0.999,0.9979733269265685,0.9,1.279693776534	37	health,mlp,ERM,2,0.2,1.0,1.0,0.8,1.2646953500807285
12	health,mlp,AF ,0,0.1,0.999,0.9979733269265685,0.9,1.521521086494		
13	health,mlp,AF ,0,0.2,0.999,0.9979733269265685,0.8,1.495562009533		

## APPENDIX H AUTO-GENERATED INSTANCE COST CSV

Listing 4. Per-instance recourse costs CSV

1	dataset,model,trainer,seed,epsilon,instance_id,valid cost	1	health,lin,AF,0,0.0,0,True,0.0022174298096731904
2	health,lin,AF,0,0.0,1,True,0.09351049565102389	2	health,lin,AF,0,0.0,2,True,0.5008139511917434
3	health,lin,AF,0,0.0,3,True,1.7989967006950356	3	health,lin,AF,0,0.0,4,True,1.4585106692345036
4	health,lin,AF,0,0.0,5,True,0.10900482104156642	4	health,lin,AF,0,0.0,6,True,0.009357979114851105
5	health,lin,AF,0,0.0,6,True,0.009357979114851105	5	
6		6	
7		7	
8		8	

9	health,lin,AF,0,0.0,7,True,1.0316063447917123	86	health,lin,AF,2,0.2,4,True,0.7999893193608962
10	health,lin,AF,0,0.0,8,True,0.2709796170170954	87	health,lin,AF,2,0.2,5,True,1.6962947926899266
11	health,lin,AF,0,0.0,9,True,0.1764318752921754	88	health,lin,AF,2,0.2,6,True,0.8233063881056557
12	health,lin,AF,0,0.1,0,True,0.12428426527689317	89	health,lin,AF,2,0.2,7,True,0.4616859846961327
13	health,lin,AF,0,0.1,1,True,0.21557733111824384	90	health,lin,AF,2,0.2,8,True,1.4058243787772824
14	health,lin,AF,0,0.1,2,True,0.6228807866589635	91	health,lin,AF,2,0.2,9,True,0.7265008107376818
15	health,lin,AF,0,0.1,3,True,1.9210635361622554	92	health,mlp,AF,0,0.0,0,True,2.675544261932373
16	health,lin,AF,0,0.1,4,True,1.5805775047017234	93	health,mlp,AF,0,0.0,1,True,2.4312634468078613
17	health,lin,AF,0,0.1,5,True,0.23107165650878636	94	health,mlp,AF,0,0.0,2,True,2.3325533866882324
18	health,lin,AF,0,0.1,6,True,0.1314248145820711	95	health,mlp,AF,0,0.0,3,True,0.5953392386436462
19	health,lin,AF,0,0.1,7,True,1.1536731802589322	96	health,mlp,AF,0,0.0,4,True,0.09999999403953552
20	health,lin,AF,0,0.1,8,True,0.3930464524843154	97	health,mlp,AF,0,0.0,5,True,1.8651421070098877
21	health,lin,AF,0,0.1,9,True,0.2984987107593954	98	health,mlp,AF,0,0.0,6,True,0.10000000149011612
22	health,lin,AF,0,0.2,0,True,0.24635110074411312	99	health,mlp,AF,0,0.0,7,True,1.220560908317566
23	health,lin,AF,0,0.2,1,True,0.33764416658546387	100	health,mlp,AF,0,0.0,8,False,inf
24	health,lin,AF,0,0.2,2,True,0.7449476221261834	101	health,mlp,AF,0,0.0,9,True,0.19684064388275146
25	health,lin,AF,0,0.2,3,True,2.0431303716294753	102	health,mlp,AF,0,0.1,0,True,2.7411983013153076
26	health,lin,AF,0,0.2,4,True,1.7026443401689435	103	health,mlp,AF,0,0.1,1,True,2.6286020278930664
27	health,lin,AF,0,0.2,5,True,0.3531384919760064	104	health,mlp,AF,0,0.1,2,True,2.5649328231811523
28	health,lin,AF,0,0.2,6,True,0.25349165004929103	105	health,mlp,AF,0,0.1,3,True,1.3312733173370361
29	health,lin,AF,0,0.2,7,True,1.2757400157261523	106	health,mlp,AF,0,0.1,4,True,0.1998090147972107
30	health,lin,AF,0,0.2,8,True,0.5151132879515354	107	health,mlp,AF,0,0.1,5,True,1.8813002109527588
31	health,lin,AF,0,0.2,9,True,0.4205655462266154	108	health,mlp,AF,0,0.1,6,True,0.1784929484128952
32	health,lin,AF,1,0.0,0,True,1.7717210875993217	109	health,mlp,AF,0,0.1,7,True,1.9187548160552979
33	health,lin,AF,1,0.0,1,True,0.2295555744846745	110	health,mlp,AF,0,0.1,8,False,inf
34	health,lin,AF,1,0.0,2,True,1.4773945272283713	111	health,mlp,AF,0,0.1,9,True,0.24932631850242615
35	health,lin,AF,1,0.0,3,True,0.1807196830988592	112	health,mlp,AF,0,0.2,0,True,2.744765281677246
36	health,lin,AF,1,0.0,4,True,0.6221210264548876	113	health,mlp,AF,0,0.2,1,True,2.540329933166504
37	health,lin,AF,1,0.0,5,True,0.029517278656927354	114	health,mlp,AF,0,0.2,2,True,2.1792831420898438
38	health,lin,AF,1,0.0,6,True,1.1691237612686767	115	health,mlp,AF,0,0.2,3,True,1.9039430618286133
39	health,lin,AF,1,0.0,7,True,0.5842132876148337	116	health,mlp,AF,0,0.2,4,True,0.28769415616989136
40	health,lin,AF,1,0.0,8,True,1.675105213965182	117	health,mlp,AF,0,0.2,5,True,1.7928175926208496
41	health,lin,AF,1,0.0,9,True,1.0109263503131347	118	health,mlp,AF,0,0.2,6,True,0.27897143363952637
42	health,lin,AF,1,0.1,0,True,1.894747702788817	119	health,mlp,AF,0,0.2,7,False,inf
43	health,lin,AF,1,0.1,1,True,0.35258218967416993	120	health,mlp,AF,0,0.2,8,False,inf
44	health,lin,AF,1,0.1,2,True,1.6004211424178667	121	health,mlp,AF,0,0.2,9,True,0.23669147491455078
45	health,lin,AF,1,0.1,3,True,0.3037462982883546	122	health,mlp,AF,1,0.0,0,True,0.8286165595054626
46	health,lin,AF,1,0.1,4,True,0.745147641644383	123	health,mlp,AF,1,0.0,1,True,0.071250319480896
47	health,lin,AF,1,0.1,5,True,0.15254389384642275	124	health,mlp,AF,1,0.0,2,True,0.6133818626403809
48	health,lin,AF,1,0.1,6,True,1.292150376458172	125	health,mlp,AF,1,0.0,3,True,0.09999999403953552
49	health,lin,AF,1,0.1,7,True,0.707239902804329	126	health,mlp,AF,1,0.0,4,True,0.38270947337150574
50	health,lin,AF,1,0.1,8,True,1.7981318291546775	127	health,mlp,AF,1,0.0,5,True,0.9619376063346863
51	health,lin,AF,1,0.1,9,True,1.1339529655026301	128	health,mlp,AF,1,0.0,6,True,2.3279929161071777
52	health,lin,AF,1,0.2,0,True,2.0177743179783123	129	health,mlp,AF,1,0.0,7,True,0.7694158554077148
53	health,lin,AF,1,0.2,1,True,0.4756088046836653	130	health,mlp,AF,1,0.0,8,True,0.7187708616256714
54	health,lin,AF,1,0.2,2,True,1.7234477576073621	131	health,mlp,AF,1,0.0,9,True,0.6485913991928101
55	health,lin,AF,1,0.2,3,True,0.42677291347785	132	health,mlp,AF,1,0.1,0,True,0.977965235710144
56	health,lin,AF,1,0.2,4,True,0.8681742568338784	133	health,mlp,AF,1,0.1,1,True,0.9960500597953796
57	health,lin,AF,1,0.2,5,True,0.2755705090359181	134	health,mlp,AF,1,0.1,2,True,0.7119928598403931
58	health,lin,AF,1,0.2,6,True,1.4151769916467673	135	health,mlp,AF,1,0.1,3,True,0.1821739375591278
59	health,lin,AF,1,0.2,7,True,0.8302665179938244	136	health,mlp,AF,1,0.1,4,True,0.49758991599082947
60	health,lin,AF,1,0.2,8,True,1.921158444344173	137	health,mlp,AF,1,0.1,5,True,1.1838374137878418
61	health,lin,AF,1,0.2,9,True,1.2569795806921256	138	health,mlp,AF,1,0.1,6,True,2.5676591396331787
62	health,lin,AF,2,0.0,0,True,1.0038187517373052	139	health,mlp,AF,1,0.1,7,True,0.8329548835754395
63	health,lin,AF,2,0.0,1,True,0.2500396675414716	140	health,mlp,AF,1,0.1,8,True,0.7808516025543213
64	health,lin,AF,2,0.0,2,True,1.0737270578129243	141	health,mlp,AF,1,0.1,9,True,0.7513484358787537
65	health,lin,AF,2,0.0,3,True,0.05366143618733343	142	health,mlp,AF,1,0.2,0,True,1.1314444541931152
66	health,lin,AF,2,0.0,4,True,0.5569176650641724	143	health,mlp,AF,1,0.2,1,True,1.17803955078125
67	health,lin,AF,2,0.0,5,True,1.4532231383932026	144	health,mlp,AF,1,0.2,2,True,0.7839648127555847
68	health,lin,AF,2,0.0,6,True,0.5802347338089318	145	health,mlp,AF,1,0.2,3,True,0.2460516095161438
69	health,lin,AF,2,0.0,7,True,0.21861433039940883	146	health,mlp,AF,1,0.2,4,True,0.6244319081306458
70	health,lin,AF,2,0.0,8,True,1.1627527244805589	147	health,mlp,AF,1,0.2,5,True,1.2506043910980225
71	health,lin,AF,2,0.0,9,True,0.483429156440958	148	health,mlp,AF,1,0.2,6,True,1.1225972175598145
72	health,lin,AF,2,0.1,0,True,1.125354578885667	149	health,mlp,AF,1,0.2,7,True,1.010860800743103
73	health,lin,AF,2,0.1,1,True,0.3715754946898336	150	health,mlp,AF,1,0.2,8,True,0.9001052379608154
74	health,lin,AF,2,0.1,2,True,1.1952628849612863	151	health,mlp,AF,1,0.2,9,True,0.89609865436554
75	health,lin,AF,2,0.1,3,True,0.17519726333569535	152	health,mlp,AF,2,0.0,0,True,3.503692388534546
76	health,lin,AF,2,0.1,4,True,0.6784534922125343	153	health,mlp,AF,2,0.0,1,True,1.1493207216262817
77	health,lin,AF,2,0.1,5,True,1.5747589655415644	154	health,mlp,AF,2,0.0,2,True,3.8964755535125732
78	health,lin,AF,2,0.1,6,True,0.7017705609572937	155	health,mlp,AF,2,0.0,3,True,3.741835594177246
79	health,lin,AF,2,0.1,7,True,0.3401501575477708	156	health,mlp,AF,2,0.0,4,True,3.2063651084899902
80	health,lin,AF,2,0.1,8,True,1.2842885516289209	157	health,mlp,AF,2,0.0,5,True,2.5007309913635254
81	health,lin,AF,2,0.1,9,True,0.6049649835893199	158	health,mlp,AF,2,0.0,6,True,5.388555526733398
82	health,lin,AF,2,0.2,0,True,1.246890406034029	159	health,mlp,AF,2,0.0,7,True,0.2651277780532837
83	health,lin,AF,2,0.2,1,True,0.4931113218381955	160	health,mlp,AF,2,0.0,8,True,4.400606155395508
84	health,lin,AF,2,0.2,2,True,1.3167987121096483	161	health,mlp,AF,2,0.0,9,True,4.526076316833496
85	health,lin,AF,2,0.2,3,True,0.2967330904840573	162	health,mlp,AF,2,0.1,0,True,4.200447082519531

163	health,mlp,AF,2,0.1,1,True,1.0612828731536865	240	health,lin,ERM,1,0.2,8,True,1.4670005760483487
164	health,mlp,AF,2,0.1,2,True,4.270608425140381	241	health,lin,ERM,1,0.2,9,True,1.2602350744751496
165	health,mlp,AF,2,0.1,3,True,3.9029312133789062	242	health,lin,ERM,2,0.0,0,True,0.056236713989842375
166	health,mlp,AF,2,0.1,4,True,3.3833022117614746	243	health,lin,ERM,2,0.0,1,True,1.7118647040288175
167	health,mlp,AF,2,0.1,5,True,2.5903685092926025	244	health,lin,ERM,2,0.0,2,True,0.4085758944129868
168	health,mlp,AF,2,0.1,6,True,5.484298229217529	245	health,lin,ERM,2,0.0,3,True,0.2698963714817156
169	health,mlp,AF,2,0.1,7,True,0.32856568694114685	246	health,lin,ERM,2,0.0,4,True,0.7548971117045661
170	health,mlp,AF,2,0.1,8,True,4.6868815422058105	247	health,lin,ERM,2,0.0,5,True,0.8574385415387005
171	health,mlp,AF,2,0.1,9,True,4.517419338226318	248	health,lin,ERM,2,0.0,6,True,0.5467518897496327
172	health,mlp,AF,2,0.2,0,True,4.394152641296387	249	health,lin,ERM,2,0.0,7,True,0.665408573976689
173	health,mlp,AF,2,0.2,1,True,1.2017030715942383	250	health,lin,ERM,2,0.0,8,True,0.2177330934524654
174	health,mlp,AF,2,0.2,2,True,4.303710460662842	251	health,lin,ERM,2,0.0,9,True,2.3579908462420525
175	health,mlp,AF,2,0.2,3,True,3.971803665161133	252	health,lin,ERM,2,0.1,0,True,0.17327320480765962
176	health,mlp,AF,2,0.2,4,True,3.635038375854492	253	health,lin,ERM,2,0.1,1,True,1.8289011948463947
177	health,mlp,AF,2,0.2,5,True,2.6649506092071533	254	health,lin,ERM,2,0.1,2,True,0.5256123852308041
178	health,mlp,AF,2,0.2,6,True,5.655788421630859	255	health,lin,ERM,2,0.1,3,True,0.38693286229953283
179	health,mlp,AF,2,0.2,7,True,0.34340187907218933	256	health,lin,ERM,2,0.1,4,True,0.8719336025223833
180	health,mlp,AF,2,0.2,8,True,4.704601287841797	257	health,lin,ERM,2,0.1,5,True,0.9744705323565178
181	health,mlp,AF,2,0.2,9,True,4.650392055511475	258	health,lin,ERM,2,0.1,6,True,0.6637883805674499
182	health,lin,ERM,0,0.0,0,True,0.6869062250867219	259	health,lin,ERM,2,0.1,7,True,0.7824450647945063
183	health,lin,ERM,0,0.0,1,True,0.8512159984836815	260	health,lin,ERM,2,0.1,8,True,0.33476958427028264
184	health,lin,ERM,0,0.0,2,True,0.7967231334894521	261	health,lin,ERM,2,0.1,9,True,2.4750273330598697
185	health,lin,ERM,0,0.0,3,True,2.226624921489473	262	health,lin,ERM,2,0.2,0,True,0.2903096956254768
186	health,lin,ERM,0,0.0,4,True,1.2454680372368914	263	health,lin,ERM,2,0.2,1,True,1.945937685664452
187	health,lin,ERM,0,0.0,5,True,1.3800526185081698	264	health,lin,ERM,2,0.2,2,True,0.6426488760486213
188	health,lin,ERM,0,0.0,6,True,0.7448156852837262	265	health,lin,ERM,2,0.2,3,True,0.5039693531173501
189	health,lin,ERM,0,0.0,7,True,0.5779837122863332	266	health,lin,ERM,2,0.2,4,True,0.9889700933402005
190	health,lin,ERM,0,0.0,8,True,0.05125854273952032	267	health,lin,ERM,2,0.2,5,True,1.091511523174335
191	health,lin,ERM,0,0.0,9,True,0.5222497904649147	268	health,lin,ERM,2,0.2,6,True,0.7808248713852672
192	health,lin,ERM,0,0.1,0,True,0.8009490990781871	269	health,lin,ERM,2,0.2,7,True,0.8994815556123235
193	health,lin,ERM,0,0.1,1,True,0.9652588724751465	270	health,lin,ERM,2,0.2,8,True,0.45180607508809995
194	health,lin,ERM,0,0.1,2,True,0.9107660074809173	271	health,lin,ERM,2,0.2,9,True,2.592063827877687
195	health,lin,ERM,0,0.1,3,True,2.3406677954809383	272	health,mlp,ERM,0,0.0,0,True,0.09999999403953552
196	health,lin,ERM,0,0.1,4,True,1.3595109112283565	273	health,mlp,ERM,0,0.0,1,True,1.1584840408843384
197	health,lin,ERM,0,0.1,5,True,1.4940954924996346	274	health,mlp,ERM,0,0.0,2,False,inf
198	health,lin,ERM,0,0.1,6,True,0.8588585592751913	275	health,mlp,ERM,0,0.0,3,False,inf
199	health,lin,ERM,0,0.1,7,True,0.6920265862777982	276	health,mlp,ERM,0,0.0,4,True,0.09999999403953552
200	health,lin,ERM,0,0.1,8,True,0.16530141673098547	277	health,mlp,ERM,0,0.0,5,False,inf
201	health,lin,ERM,0,0.1,9,True,0.6362926644563799	278	health,mlp,ERM,0,0.0,6,True,1.7681208848953247
202	health,lin,ERM,0,0.2,0,True,0.9149919730696522	279	health,mlp,ERM,0,0.0,7,True,0.1756606101989746
203	health,lin,ERM,0,0.2,1,True,1.0793017464666117	280	health,mlp,ERM,0,0.0,8,True,0.09999999403953552
204	health,lin,ERM,0,0.2,2,True,1.0248088814723824	281	health,mlp,ERM,0,0.0,9,True,1.099407434463501
205	health,lin,ERM,0,0.2,3,True,2.4547106694724032	282	health,mlp,ERM,0,0.1,0,True,0.2847887873649597
206	health,lin,ERM,0,0.2,4,True,1.4735537852198217	283	health,mlp,ERM,0,0.1,1,True,1.350301742553711
207	health,lin,ERM,0,0.2,5,True,1.6081383664910998	284	health,mlp,ERM,0,0.1,2,False,inf
208	health,lin,ERM,0,0.2,6,True,0.97290143232666565	285	health,mlp,ERM,0,0.1,3,False,inf
209	health,lin,ERM,0,0.2,7,True,0.8060694602692634	286	health,mlp,ERM,0,0.1,4,True,0.17709362506866455
210	health,lin,ERM,0,0.2,8,True,0.2793442907224506	287	health,mlp,ERM,0,0.1,5,True,1.4368488788604736
211	health,lin,ERM,0,0.2,9,True,0.750335538447845	288	health,mlp,ERM,0,0.1,6,True,1.783553123474121
212	health,lin,ERM,1,0.0,0,True,0.4544031954776043	289	health,mlp,ERM,0,0.1,7,True,0.3221847414970398
213	health,lin,ERM,1,0.0,1,True,1.1770760395405075	290	health,mlp,ERM,0,0.1,8,True,0.14391285181045532
214	health,lin,ERM,1,0.0,2,True,1.5513781335972643	291	health,mlp,ERM,0,0.1,9,True,1.2745985984802246
215	health,lin,ERM,1,0.0,3,True,1.3697575946921152	292	health,mlp,ERM,0,0.2,0,True,0.2927534878253937
216	health,lin,ERM,1,0.0,4,True,1.67729323247764	293	health,mlp,ERM,0,0.2,1,True,1.420407772064209
217	health,lin,ERM,1,0.0,5,True,0.04324403365116183	294	health,mlp,ERM,0,0.2,2,False,inf
218	health,lin,ERM,1,0.0,6,True,0.972477918539937	295	health,mlp,ERM,0,0.2,3,True,1.6530849933624268
219	health,lin,ERM,1,0.0,7,True,0.21887793076712805	296	health,mlp,ERM,0,0.2,4,True,0.37585678696632385
220	health,lin,ERM,1,0.0,8,True,1.2335342986448172	297	health,mlp,ERM,0,0.2,5,True,1.0841124057769775
221	health,lin,ERM,1,0.0,9,True,1.026768797071618	298	health,mlp,ERM,0,0.2,6,True,2.038994312286377
222	health,lin,ERM,1,0.1,0,True,0.57113633417937	299	health,mlp,ERM,0,0.2,7,True,0.4549943208694458
223	health,lin,ERM,1,0.1,1,True,1.2938091782422734	300	health,mlp,ERM,0,0.2,8,True,0.09999996423721313
224	health,lin,ERM,1,0.1,2,True,1.66811127229903	301	health,mlp,ERM,0,0.2,9,True,0.543687105178833
225	health,lin,ERM,1,0.1,3,True,1.4864907333938808	302	health,mlp,ERM,1,0.0,0,True,0.6265113949775696
226	health,lin,ERM,1,0.1,4,True,1.7944624619495295	303	health,mlp,ERM,1,0.0,1,True,0.5598059892654419
227	health,lin,ERM,1,0.1,5,True,0.15997717235292763	304	health,mlp,ERM,1,0.0,2,True,4.995126724243164
228	health,lin,ERM,1,0.1,6,True,1.0892110572417029	305	health,mlp,ERM,1,0.0,3,True,0.8848893046379089
229	health,lin,ERM,1,0.1,7,True,0.33561106946889385	306	health,mlp,ERM,1,0.0,4,True,0.9069631099700928
230	health,lin,ERM,1,0.1,8,True,1.3502674373465828	307	health,mlp,ERM,1,0.0,5,True,6.042341232299805
231	health,lin,ERM,1,0.1,9,True,1.1435019357733838	308	health,mlp,ERM,1,0.0,6,True,0.9130632877349854
232	health,lin,ERM,1,0.2,0,True,0.6878694728811359	309	health,mlp,ERM,1,0.0,7,True,0.42502132058143616
233	health,lin,ERM,1,0.2,1,True,1.4105423169440392	310	health,mlp,ERM,1,0.0,8,True,0.09999999403953552
234	health,lin,ERM,1,0.2,2,True,1.7848444110007957	311	health,mlp,ERM,1,0.0,9,True,0.3435572385787964
235	health,lin,ERM,1,0.2,3,True,1.6032238720956469	312	health,mlp,ERM,1,0.1,0,True,0.7597363591194153
236	health,lin,ERM,1,0.2,4,True,1.9111956006512953	313	health,mlp,ERM,1,0.1,1,True,0.7501630187034607
237	health,lin,ERM,1,0.2,5,True,0.2767103110546934	314	health,mlp,ERM,1,0.1,2,True,4.815957069396973
238	health,lin,ERM,1,0.2,6,True,1.2059441959434687	315	health,mlp,ERM,1,0.1,3,True,1.1189303398132324
239	health,lin,ERM,1,0.2,7,True,0.45234420817065973	316	health,mlp,ERM,1,0.1,4,True,0.8854197263717651

```

317 health,mlp,ERM,1,0.1,5,True,6.365862846374512
318 health,mlp,ERM,1,0.1,6,True,0.8871881365776062
319 health,mlp,ERM,1,0.1,7,True,0.5466513633728027
320 health,mlp,ERM,1,0.1,8,True,0.09999998658895493
321 health,mlp,ERM,1,0.1,9,True,0.35792168974876404
322 health,mlp,ERM,1,0.2,0,True,0.8752454519271851
323 health,mlp,ERM,1,0.2,1,True,0.7518577575683594
324 health,mlp,ERM,1,0.2,2,True,4.760556697845459
325 health,mlp,ERM,1,0.2,3,True,1.0518792867660522
326 health,mlp,ERM,1,0.2,4,True,0.9736133813858032
327 health,mlp,ERM,1,0.2,5,True,2.644326686859131
328 health,mlp,ERM,1,0.2,6,True,0.7649006843566895
329 health,mlp,ERM,1,0.2,7,True,0.6719211935997009
330 health,mlp,ERM,1,0.2,8,True,0.09999998658895493
331 health,mlp,ERM,1,0.2,9,True,0.40259477496147156
332 health,mlp,ERM,2,0.0,0,True,1.6526018381118774
333 health,mlp,ERM,2,0.0,1,True,1.0587971210479736
334 health,mlp,ERM,2,0.0,2,True,0.871553897857666
335 health,mlp,ERM,2,0.0,3,True,1.5454106330871582
336 health,mlp,ERM,2,0.0,4,True,0.09999999403953552
337 health,mlp,ERM,2,0.0,5,True,4.139216899871826
338 health,mlp,ERM,2,0.0,6,True,1.2884514331817627
339 health,mlp,ERM,2,0.0,7,True,1.023694634437561
340 health,mlp,ERM,2,0.0,8,True,0.09999997913837433
341 health,mlp,ERM,2,0.0,9,False,inf
342 health,mlp,ERM,2,0.1,0,True,2.3038556575775146
343 health,mlp,ERM,2,0.1,1,True,0.48094290494918823
344 health,mlp,ERM,2,0.1,2,True,0.9778098464012146
345 health,mlp,ERM,2,0.1,3,True,1.8084425926208496
346 health,mlp,ERM,2,0.1,4,True,0.19984257221221924
347 health,mlp,ERM,2,0.1,5,True,4.787795066833496
348 health,mlp,ERM,2,0.1,6,True,1.3899121284484863
349 health,mlp,ERM,2,0.1,7,True,1.2289860248565674
350 health,mlp,ERM,2,0.1,8,True,0.291856586933136
351 health,mlp,ERM,2,0.1,9,False,inf
352 health,mlp,ERM,2,0.2,0,False,inf
353 health,mlp,ERM,2,0.2,1,True,0.4565989375114441
354 health,mlp,ERM,2,0.2,2,True,1.0854618549346924
355 health,mlp,ERM,2,0.2,3,True,1.8083579540252686
356 health,mlp,ERM,2,0.2,4,False,inf
357 health,mlp,ERM,2,0.2,5,True,1.6818499565124512
358 health,mlp,ERM,2,0.2,6,True,1.5481600761413574
359 health,mlp,ERM,2,0.2,7,True,1.1980092525482178
360 health,mlp,ERM,2,0.2,8,True,0.37811192870140076
361 health,mlp,ERM,2,0.2,9,True,1.961012840270996

```

## APPENDIX I AUTO-GENERATED ACTION PROFILE CSV

Listing 5. Feature-wise action diagnostics CSV

```

1 model,trainer,config,epsilon,seed,feature,
  mean_abs_action_all,mean_abs_action_valid,
  nonzero_rate_all,nonzero_rate_valid,actionable
2 lin,AF,LIN-AF,0.1,0,age,0.0,0.0,0.0,0.0,0
3 lin,AF,LIN-AF,0.1,0,insulin
  ,0.26181564489063913,0.26181564489063913,0.2,0.2,1
4 lin,AF,LIN-AF,0.1,0,blood_glucose,0.0,0.0,0.0,0.0,1
5 lin,AF,LIN-AF,0.1,0,blood_pressure,0.0,0.0,0.0,0.0,0
6 lin,ERM,LIN-ERM,0.1,0,age,0.0,0.0,0.0,0.0,0
7 lin,ERM,LIN-ERM,0.1,0,insulin
  ,0.4521024343304031,0.4521024343304031,0.6,0.6,1
8 lin,ERM,LIN-ERM,0.1,0,blood_glucose
  ,0.0,0.0,0.0,0.0,1
9 lin,ERM,LIN-ERM,0.1,0,blood_pressure
  ,0.0,0.0,0.0,0.0,0
10 mlp,AF,MLP-AF,0.1,0,age,0.0,0.0,0.0,0.0,0
11 mlp,AF,MLP-AF,0.1,0,insulin
  ,0.24029699489474296,0.3003712436184287,0.5,0.625,
12 mlp,AF,MLP-AF,0.1,0,blood_glucose
  ,0.14958224296569825,0.1869778037071228,0.1,0.125,
13 mlp,AF,MLP-AF,0.1,0,blood_pressure,0.0,0.0,0.0,0.0,0

```

```

14 mlp,ERM,MLP-ERM,0.1,0,age,0.0,0.0,0.0,0.0,0
15 mlp,ERM,MLP-ERM,0.1,0,insulin
  ,0.02044838070869446,0.02272042300966051,0.2,0.2222222222
16 mlp,ERM,MLP-ERM,0.1,0,blood_glucose
  ,0.0870448887348175,0.09671654303868611,0.1,0.1111111111
17 mlp,ERM,MLP-ERM,0.1,0,blood_pressure
  ,0.0,0.0,0.0,0.0,0

```

## APPENDIX J AUTO-GENERATED ACTION INSTANCE STATS CSV

Listing 6. Per-instance action vectors and norms CSV

```

1 model,trainer,config,epsilon,seed,instance_id,valid,
  l0_nonzero,l1_norm,l2_norm,cost,action_age,
  action_insulin,action_blood_glucose,
  action_blood_pressure
2 lin,AF,LIN-AF,0.1,0,0,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
3 lin,AF,LIN-AF,0.1,0,1,True
  ,1,0.912671422200381,0.912671422200381,0.912671422200381,0.
4 lin,AF,LIN-AF,0.1,0,2,True
  ,1,1.710092677680237,1.710092677680237,1.710092677680237,0.
5 lin,AF,LIN-AF,0.1,0,3,True
  ,1,1.049107052640722,1.049107052640722,1.049107052640722,0.
6 lin,AF,LIN-AF,0.1,0,4,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
7 lin,AF,LIN-AF,0.1,0,5,True
  ,1,0.04245187920182977,0.04245187920182977,0.04245187920182
8 lin,AF,LIN-AF,0.1,0,6,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
9 lin,AF,LIN-AF,0.1,0,7,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
10 lin,AF,LIN-AF,0.1,0,8,True
  ,1,1.3021861189993753,1.3021861189993753,1.3021861189993753
11 lin,AF,LIN-AF,0.1,0,9,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
12 lin,ERM,LIN-ERM,0.1,0,0,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
13 lin,ERM,LIN-ERM,0.1,0,1,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
14 lin,ERM,LIN-ERM,0.1,0,2,True
  ,1,0.8165668077686034,0.8165668077686034,0.8165668077686034
15 lin,ERM,LIN-ERM,0.1,0,3,True
  ,1,0.8582558826695555,0.8582558826695555,0.8582558826695555
16 lin,ERM,LIN-ERM,0.1,0,4,True
  ,1,0.3168366177055238,0.3168366177055238,0.3168366177055238
17 lin,ERM,LIN-ERM,0.1,0,5,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
18 lin,ERM,LIN-ERM,0.1,0,6,True
  ,1,1.4160299277944228,1.4160299277944228,1.4160299277944228
19 lin,ERM,LIN-ERM,0.1,0,7,True
  ,1,0.9483331222630108,0.9483331222630108,0.9483331222630108
20 lin,ERM,LIN-ERM,0.1,0,8,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
21 lin,ERM,LIN-ERM,0.1,0,9,True
  ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
22 mlp,AF,MLP-AF,0.1,0,0,False,0,0.0,0.0,inf
  ,0.0,0.0,0.0,0.0
mlp,AF,MLP-AF,0.1,0,1,True
  ,1,0.630301833152771,0.630301833152771,0.630301833152771,0.

```

24	mlp,AF,MLP-AF,0.1,0,2,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	2	lin,AF,LIN-AF ,0.0,1.0,1.0,1.0,0.7012748778796839,0.5062638203282346,0.91
25	mlp,AF,MLP-AF,0.1,0,3,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	3	lin,AF,LIN-AF ,0.1,1.0,1.0,1.0,0.823484637148043,0.6173001966002503,1.03
26	mlp,AF,MLP-AF,0.1,0,4,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0	4	lin,AF,LIN-AF ,0.2,1.0,1.0,1.0,0.9456943964164021,0.7429863056628437,1.16
27	mlp,AF,MLP-AF,0.1,0,5,True ,1,0.2453588843345642,0.2453588843345642,0.2453588	5	lin,ERM,LIN-ERM ,0.0,1.0,1.0,1.0,0.8885113223625424,0.6759910329330764,1.11
28	mlp,AF,MLP-AF,0.1,0,6,True ,2,1.7315754890441895,1.2251203467080083,1.7315754	6	lin,ERM,LIN-ERM ,0.1,1.0,1.0,1.0,1.0044488235328917,0.7995385802997136,1.22
29	mlp,AF,MLP-AF,0.1,0,7,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	7	lin,ERM,LIN-ERM ,0.2,1.0,1.0,1.0,1.120386324703241,0.9017732550292862,1.36
30	mlp,AF,MLP-AF,0.1,0,8,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	8	mlp,AF,MLP-AF ,0.0,0.9666666666666667,0.9,1.0,1.810989550732333,1.2662813
31	mlp,AF,MLP-AF,0.1,0,9,True ,1,0.7459346055984497,0.7459346055984497,0.7459346	9	mlp,AF,MLP-AF ,0.1,0.9666666666666667,0.9,1.0,1.986283392279293,1.434618
32	mlp,ERM,MLP-ERM,0.1,0,0,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	10	mlp,AF,MLP-AF ,0.2,0.9333333333333333,0.8333333333333334,1.0,2.0226513711
33	mlp,ERM,MLP-ERM,0.1,0,1,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0	11	mlp,ERM,MLP-ERM ,0.0,0.8666666666666667,0.7333333333333333,0.9666666666666667
34	mlp,ERM,MLP-ERM,0.1,0,2,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0	12	mlp,ERM,MLP-ERM ,0.1,0.9,0.8,1.0,1.3640946765188817,0.8771141729972981,1.94
35	mlp,ERM,MLP-ERM,0.1,0,3,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0	13	mlp,ERM,MLP-ERM ,0.2,0.9,0.8,1.0,1.151049994484142,0.8237234620736629,1.529
36	mlp,ERM,MLP-ERM,0.1,0,4,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0		
37	mlp,ERM,MLP-ERM,0.1,0,5,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0		
38	mlp,ERM,MLP-ERM,0.1,0,6,False,0,0.0,0.0,inf ,0.0,0.0,0.0,0.0		
39	mlp,ERM,MLP-ERM,0.1,0,7,True ,0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0		
40	mlp,ERM,MLP-ERM,0.1,0,8,True ,1,0.07758744060993195,0.07758744060993195,0.07758		
41	mlp,ERM,MLP-ERM,0.1,0,9,True ,2,0.26115771336480975,0.25923200853642037,0.26115		

## ACKNOWLEDGMENT

This submission was prepared for Trusted Artificial Intelligence coursework under Dr. Mostafa Tavasolipour.

## APPENDIX K AUTO-GENERATED SPARSITY SUMMARY CSV

Listing 7. Valid recourse sparsity/cost summary CSV

1	model,trainer,config,epsilon,seed,valid_rate, mean_l0_valid,std_l0_valid,mean_l1_valid, std_l1_valid,mean_l2_valid,std_l2_valid		
2	lin,AF,LIN-AF ,0.1,0,1.0,0.5,0.5,0.5016509150722545,0.6355286101893044,0.4356022358201116,0.5010741804045799,0.4356022358201116,0.5010741804045799	1893044	0.0019329111091792583,0.0
3	lin,ERM,LIN-ERM ,0.1,0,1.0,0.5,0.5,0.4356022358201116,0.5010741804045799,0.4356022358201116,0.5010741804045799	4045799	0.0019329111091792583,0.0
4	mlp,AF,MLP-AF ,0.1,0,0.8,0.625,0.6959705453537527,0.4191463515162488,0.15400905396403,0.3556045845845846	162488	0.0019329111091792583,0.0
5	mlp,ERM,MLP-ERM ,0.1,0,0.5,0.6,0.8,0.06774903079494835,0.10126549888492285,0.06736388982927047,0.10053017151639203	988492285	0.0019329111091792583,0.0

## REFERENCES

- [1] B. Ustun, A. Spangher, and Y. Liu, "Actionable recourse in linear classification," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019.
- [2] J. Pearl, *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2009.
- [3] A.-H. Karimi, G. Barthe, B. Balle, and I. Valera, "Algorithmic recourse: from counterfactual explanations to interventions," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021.
- [4] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020.
- [5] A.-H. Karimi, B. Schölkopf, and I. Valera, "Algorithmic recourse under imperfect causal knowledge: A probabilistic approach," in *Advances in Neural Information Processing Systems*, 2020.

## APPENDIX L AUTO-GENERATED BOOTSTRAP SUMMARY CSV

Listing 8. Bootstrap confidence interval summary CSV

1	model,trainer,config,epsilon,valid_rate_mean, valid_rate_ci_low,valid_rate_ci_high, valid_cost_mean,valid_cost_ci_low, valid_cost_ci_high,n_rows
---	-----------------------------------------------------------------------------------------------------------------------------------------------------------