On the Reproducibility of: Improvement-Focused Causal Recourse

Anonymous authors
Paper under double-blind review

Abstract

This work aims to reproduce the main findings of "Improvement-Focused Causal Recourse (ICR)"(König et al., 2023) within the field of algorithmic recourse recommendations. The authors demonstrate that acceptance-focused recourse recommendation methods, like counterfactual explanations (CE), may suggest actions that revert the model's verdict by gaming the predictor whenever possible. To tackle this, the authors introduce ICR, which focuses on improvement by optimizing for a new target variable in their causal model. It is also demonstrated that improvement guarantees consequently translate into acceptance guarantees. We can confirm the findings of the original paper. The contribution of the current study is a more extensive assessment of the robustness and generalizability of ICR. Various techniques were employed to test the algorithm's performance under different architectural choices, such as different classifiers or optimization methods, data and model shifts, and a new dataset. Our findings suggest that ICR is more robust than CE and causal recourse (CR).

1 Introduction

As the deployment of predictive systems becomes more and more prevalent in critical areas of decision-making such as employee hiring (Raghavan et al., 2020), organ transplant priority determination (Obermeyer & Mullainathan, 2019), or judiciary decisions (Zeng et al., 2017), more emphasis should be placed in algorithmic explainability methods that offer the explainee an intuitive understanding of the system, and the possibility to apply recourse, i.e. actions that revert an unfavorable decision. In the domain of algorithmic decision-making, recourse methods play a crucial role in informing stakeholders on actions to reverse unfavorable model predictions. Counterfactual explanations (CE) are concerned with changing the inputs to the model such that the model prediction changes in the desired way (Wachter et al., 2017). Causal Recourse (CR) shifts away from the counterfactual explanations paradigm and proposes recourse through minimal interventions, emphasizing the actions that need to be taken to achieve a favorable decision (Karimi et al., 2021; 2022). It utilizes causal knowledge to translate into recommendable recourse actions.
Traditional approaches, such as Causal Recourse (CR), primarily focus on achieving acceptance (reverting the model's decision) without necessarily ensuring improvement in the underlying real-world state. This emphasis on acceptance can lead to actions that may deceive the predictor without effectively addressing the actual improvement needed. To address this limitation, König et al. (2023) introduce Improvement-Focused Causal Recourse (ICR). In ICR, recommendations are explicitly geared towards achieving improvement and are not tailored for acceptance by a specific predictor. Causal knowledge is leveraged through structural causal models (SCMs) (Pearl et al., 2000) or causal graphs to design decision systems that accurately predict both pre- and post-recourse. This is done by defining the improvement confidence γ, which can then be optimized to yield ICR, ensuring that improvement guarantees consistently translate into acceptance guarantees. In this work, we aim to reproduce the authors' findings, verify their claims, and perform additional experiments to assess the robustness and generalizability of the proposed method, providing further evidence to strengthen their claims. The code of our project is available here.

2 Scope of reproducibility

Improvement-Focused Causal Recourse belongs in the family of local post-hoc explainability methods. They are one of the more "human-friendly" approaches towards the goal of algorithmic transparency since they are contrastive to the current instance of a specific individual and selective since they focus on a small number of feature changes. ICR improves upon this by considering the causal dependencies between covariates (i.e. additional variables that may relate to the outcome of interest). The innovation of ICR is that, while CE and CR aim to revert the prediction, ICR seeks to revert the target, i.e., the underlying ground truth. The latter makes it a more holistic method for understanding the dynamics between input features and the outcomes in the context of algorithmic decision-making.

In the current reproducibility study, our main goal is to verify the following claims of the original paper:

- **Claim 1 - Attaining Improvement:** ICR reliably guides individuals towards actions that lead to improvement in scenarios where gaming is possible and lucrative.
- **Claim 2 - Attaining Acceptance:** CE, CR, and ICR all lead to acceptance, but CE and CR show higher observed acceptance rates than ICR.
- **Claim 3 - Attaining Acceptance Robustly:** ICR actions are more likely to be accepted by other model fits with similar performance on the same data.
- **Claim 4 - Recommendation Cost:** ICR actions are more costly than CR but lead to improvement, acceptance, and greater robustness to model refits.

In addition to reproducing the results presented in the paper, we perform additional experiments that test the robustness and, to some extent, the generalizability of the approach.

3 Methodology

The original code is publicly available in a GitHub repository by König et al. (2022). To test the original study's reproducibility, we use the provided code and the Python packages listed in the requirements file in the repository. We conduct the same set of experiments using the same hyperparameters defined in the repository by the authors to assess the reproducibility of the project results and claims. There is a minor inconsistency between the hyperparameters' description in the authors' paper and the codebase. Specifically, the hyperparameter referring to the number of observations is omitted from the paper and is only present as part of the repository. The complete list of hyperparameters used for our reproducibility investigation can be found in Appendix B.

3.1 Algorithm description

The ICR mechanism proposed by König et al. (2023) is one of the first recourse methods proposed that ensures reversion of the underlying real-world state (improvement) while also leading to acceptance (reverting an unfavorable decision). Influenced by Karimi et al. (2020), ICR utilizes the causal knowledge of either an SCM or a causal graph to steer individuals who need recourse towards improvement.

3.1.1 Individualized Improvement Confidence

An SCM comprises structural equations that specify how an endogenous variable is determined by its endogenous causes and the corresponding exogenous variable. SCMs allow us to predict the effect of actions and imagine the results of alternative actions in light of factual observations.
The ICR method builds upon SCM recourse-based techniques. It introduces a new target variable Y in the SCM that captures improvement. As a result, the modified SCM captures the probability of individualized improvement as well as the probability of individualized acceptance.

3.1.2 Subpopulation Improvement Confidence

In the case where knowledge of the SCM is unknown, no observed variables influence both the dependent and the independent variables. We have to fall back to the effect of interventions (rung 2 in Pearl's ladder of causation (Pearl, 2009)). Since the interventional distribution captures broader characteristics of the whole population, it cannot be used to capture the action effects on specific individuals accurately. In this scenario, a subpopulation-based improvement confidence expresses the probability of improvement Y being a desired outcome in a subgroup of individuals with similar characteristics.

3.1.3 Optimization Problem

Equation 1 indicates the action cost that is optimized to generate ICR suggestions. The objective is to discover actions that inflict a minimal cost while constrained by a user-specified improvement target confidence $\bar{\gamma}$. Confidence can be intuitively interpreted as the probability of improvement, given that the individual follows the recommended recourse actions. The cost function $\text{cost}\,(a, x^{\text{pre}}\,)$ reflects the effort needed by an individual to complete an action α. The optimization objective for ICR can be interpreted as two smaller intervention objectives (Karimi et al., 2020). First, optimization is applied to the intervention targets I_α, followed by optimizing intervention values θ_α. Considering our objective is to achieve improvement, we limit I_α to all parents of Y. The authors motivate their decision to use the genetic algorithm NSGA-II(Deb et al., 2002) for optimizing the constrained objective below, following previous work (Dandl et al., 2020).

$$\text{argmin}_{a=do(X_I=\theta)} \quad \text{cost}\,(a, x^{\text{pre}}\,) \quad \text{s.t.} \quad \gamma(a) \geq \bar{\gamma} \tag{1}$$

3.2 Datasets

The authors have experimented with semi-synthetic and synthetic datasets in their study. The datasets comprise the SCMs and the corresponding directed acyclic graphs G. In addition to the four original datasets, we create an additional 3var-causal-nonlinear synthetic dataset. It is similar to the 3var-causal dataset used in the original experiments, but we introduce non-linearity through defining one of the features as a binomial distribution, and another one as a quadratic relation. The purpose of this new dataset is to compare the performance of CE, CR and ICR on a small, non-linear dataset that is lower in complexity compared to the semi-synthetic 5var-skill and 7var-covid datasets that the authors use. Table 1 showcases essential information for the datasets, while Appendix A provides more detailed information, as well as a visual depiction of the causal graphs and their structural equations.

Name	Non-Linear	Features Affecting Y	Potential Gaming Variables (Features Affected by Y)	Source
3var-causal	No	3	0	Synthetic
3var-noncausal	No	2	1	Synthetic
5var-skill	Yes	2	3	Semi-Synthetic
7var-covid	Yes	4	3	Semi-Synthetic
3var-causal-nonlinear	Yes	3	0	Synthetic

Table 1: Information about the datasets

3.3 Hyperparameters

The set of hyperparameters used to reproduce the original paper's results are sourced from the description in the main text and the Appendices of the original work, and a detailed specification can be found in Appendix B. For the 7var-covid dataset, we observe smaller values for the number of generations and population size in the codebase compared to the paper. For our reproduction, we use the full-scale hyperparameter values as specified in the paper. Furthermore, we used 10 iterations for the whole procedure.

Some hyperparameters were kept constant throughout the experiments: the model's decision threshold t and γ_{LCB}, which determines how many standard deviations the expected prediction shall be away from the model's decision threshold t.

In order to conduct the additional robustness experiments in the available time frame and stay within the allocated resources, we down-scaled the set of hyperparameters to reduce the running time by a factor of two for the 3var datasets. For the same reason, the experiments we conducted use only the datasets 3var-noncausal and 3var-causal and omitted the confidence values of 0.85 and 0.90. Table 2 summarizes the hyperparameters used in those experiments.

Data set	**Number of observations**	**Number of individuals having recourse calculation**	**Confidence**	**Number of Generations**	**POP SIZE**	**n digits**	**iterations**
3var-noncausal	1000	100	0.75, 0.95	300	150	1	3
3var-causal	1000	100	0.75, 0.95	300	150	1	3
3var-causal-nonlinear	1000	100	0.75, 0.95	300	150	1	3

Table 2: Hyperparameters for the robustness experiments beyond the original paper.

3.4 Experimental setup and code

The original implementation of the ICR framework is publicly available on GitHub. Most parts of the code were running, only the requirements of the packages were not complete and the newest version of these packages do not work. The package dependencies were resolved by reverting some of the package versions back to the existing versions when the original paper was published. Additionally, there is some inconsistency between the original paper and the code regarding the runs performed per dataset.
We follow the specifications in the paper, and for every dataset, we evaluate CE, individualized and subpopulation-based CR, and ICR, for 10 iterationd, with each iteration consisting of 5 model refits and 4 confidence levels for 200 individuals. All experiments shown in this paper can easily be reproduced using our provided code.
König et al. (2023) used the outputs from all the experiments in order to answer four questions, relying on a different metric for each question; the observed improvement rate γ_{obs} (Claim 1), the observed acceptance rates η^{obs} (Claim 2), observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$ (Claim 3), and the average recourse cost for individuals who were rejected and were consequently provided with a recourse recommendation (Claim 4). Additionally, an invalidity metric is used for the robustness experiments, which expresses the percentage of post-recourse classifications that become invalid after the data has shifted (Rawal et al., 2020). More details on the metrics can be found in Appendix C.
In order to make the experiment running process more cost-effective with respect to computation resources, the *multiprocess* python package is used to enable running the experiments for CE, CR, ICR, and the individualized/subpopulation settings simultaneously. Details on the speedup can be found in the Appendix H.
An additional seed is introduced to ensure future reproducibility of numerical results. In the original paper, the values for each SCM of the datasets are sampled randomly from corresponding distributions each time. This makes reproducing the exact numerical results impossible. We add a seed, which makes the distribution generation process deterministic. For the reproduction, we use seed "1".

3.5 Robustness Assessment beyond original paper

While the original paper compares the robustness of CE, CR, and ICR on refits of the same data, we extend this robustness comparison to model and data shifts. This has been previously done on CE and CR (Upadhyay et al., 2021; Rawal et al., 2020) but, to the best of our knowledge, not on ICR. For this robustness comparison, we test different classifiers, shift the data, and use a different genetic algorithm.

3.5.1 Classifiers

The authors use random forest for classification, except in the *3var* datasets where logistic regression models are used. The former is utilized for non-linear datasets and the latter for linear ones. In this study, we compare the capabilities of ICR with different classifiers. The alternative classifiers tested are the AdaBoost Classifier (Schapire, 2013), a Support Vector Machine (SVM) for classification, and a simple Multi-Layer Perceptron (MLP). AdaBoost and SVM are implemented using the scikit-learn packages with the default hyperparameters (Pedregosa et al., 2011). The simple MLP consists of three hidden layers of 10, 10, and 5 nodes respectively. Adam is used for optimization (Kingma & Ba, 2014). ReLU is the activation function applied to all the layers.

3.5.2 Data shift

To create the data shift, we use the synthetic datasets 3var-causal and 3var-noncausal, where the features follow a standard normal distribution. We apply the same methodology as Upadhyay et al. (2021) and shift each dataset one feature at a time. Three settings can be distinguished: shifting the mean, the variance, and both simultaneously. The metric used is invalidity (Rawal et al., 2020). It measures the number of recourse recommendations that are not valid anymore for a model retrained on the shifted data. Details on this procedure can be found in Appendix D.

3.5.3 Genetic Algorithm

The authors employed a modified NSGA-II instance. This is done by altering the crowding distance computations, which are tailored for multi-objective counterfactual explanations as introduced in (Dandl et al., 2020) to minimize the cost of the optimization objective. In our study, we assess the capabilities of ICR by utilizing the newest version of Non-Dominated Sorting Genetic Algorithm (NSGA-III) (Deb & Jain, 2013). Building upon its predecessor NSGA-II, NSGA-III allows for improvements in diversity preservation and efficiency, promoting diversity among solutions. The optimization objective (described in Section 3.1.3) is a two-step problem modeled as a single-objective problem in the ICR original codebase. The capabilities of the two genetic algorithm variants in single-objective scenarios have not been widely studied as they are mainly used for multi-objective optimization. On top of that, the authors' implementation for NSGA-II is based on DEAP (Fortin et al., 2012)(evolutionary computation framework for rapid prototyping), which also supports NSGA-III natively. Thus, we firmly believe that comparing the two genetic algorithms is valuable for evaluating ICR for the given constrained optimization problem.

3.6 Computational requirements

The experiments were executed on a server using an AMD Rome CPU with 128 threads with a computational cost of 5-24 hours per experiment setting while using parallelization and 10-55 hours without parallelization. We use our personal computers for the down-scaled experiments on an AMD Ryzen 5 5500U. In Appendix H, a visualization of the parallelization speedup captures the overall computational hours and a table documenting the CPU hours per dataset needed. The reproduction of the original results took an estimated 34 CPU hours after parallelization. It should be noted that the runtime would take up to around 100 CPU hours if it is run without the parallelization setting. The additional robustness and generalization experiments took an estimated 160 CPU hours in the parallelized setting while using the scaled-down version of the hyperparameters.

4 Results

In the upcoming subsections, we first compare our results with the authors' and validate which claims hold. We then further assess the robustness of ICR and compare its performance with CE and CR.

4.1 Results reproducing original paper

The results of our reproduction can be seen in Fig. 1. Fig. 1a shows the observed improvement rates γ^{obs} for the different confidence intervals of CE, CR, and ICR. CE does not use confidence levels. Therefore, only one number is reported. Fig. 1b shows the observed acceptance rate η^{obs}. The robustness of refits from the same distribution can be seen in Fig. 1c. This graph shows the average acceptance rate of 5 refits. The average recourse cost can be seen in Fig. 1d. Appendix F is dedicated to a more detailed side-by-side comparison of the authors' outputs and ours.

Claim 1: It can be seen that only ICR has high improvement rates in Fig. 1a. CE and CR have very low improvement rates. The latter confirms the claim made by the authors. While the general trend still holds, the numbers retrieved during our experiments are still very close to the ones provided by authors but not identical. Furthermore, it can be confirmed that CE and CR games by only applying recourse to the number of GitHub commits. In contrast, ICR suggests modifying values like years of experience and getting a degree, which are non-gaming variables. As a side effect of the causal model suggesting recourse actions on the years of experience and education, the number of commits also increases. Claim 1, which supports that ICR leads to improvement in situations where gaming is beneficial, can be confirmed.

Claim 2: CE, CR, and ICR all lead to high acceptance rates in Fig. 1b. Furthermore, it can be observed that ICR has lower acceptance levels than CE and CR. As such, we can confirm Claim 2. While we could not reproduce the exact numbers the authors provide, the general trends are the same. The subpopulation method performs worse than the individualized method.

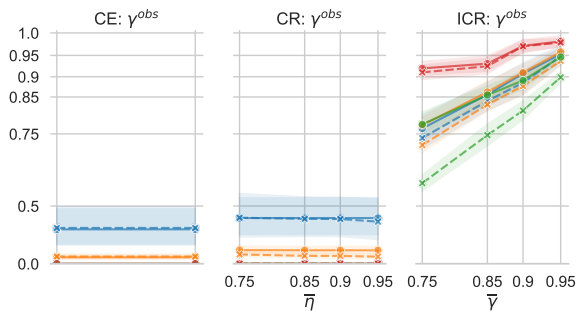
Claim 3: The performance of the refitted models, which were created by sampling a new dataset from the SCM, varies per method as seen in Fig. 1c. CE and CR perform much worse on the refitted models, except on the 7var-covid dataset. This makes CE and CR not applicable to situations where the model will be refitted since the previous recourse recommendations could be invalidated, and the individual has to implement even more actions to achieve recourse. The acceptance rates of ICR are barely affected by refitting the models. This leads to the conclusion that ICR is able to give recourse recommendations that will not change if a model is refitted with other data from the same distribution. This confirms Claim 3.

Claim 4: The recommendation costs, provided in Fig. 1d, are on average more expensive for ICR. This is due to the fact that ICR does not game by only applying the cheapest action, like the other methods do, often repeatedly. In the 5var-skill dataset ICR suggests getting a degree and gaining years of experience instead of creating a lot of commits, which makes it more costly than CE and CR. However, there are exceptions where ICR is cheaper than CE or CR, but on average, Claim 4 holds.

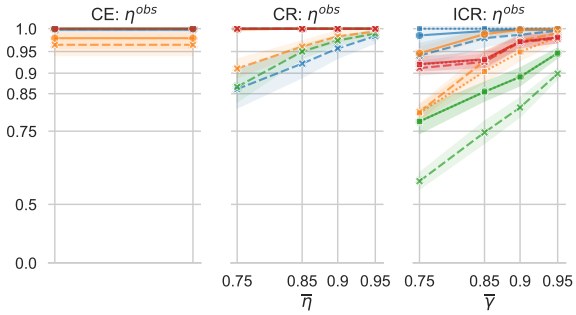
4.2 Results beyond original paper

Classifiers: We evaluate how robust a recourse recommendation is with respect to different classifiers. For the improvement rate, Table 3 indicates that the classifier does have an impact on the performance of CE and CR. The improvement rate of ICR is not dependent on the classifier, and therefore, we can show that ICR is indeed more robust towards different decision algorithms. Notice that the reported γ^{obs} values in Table 5 refer to the average improvement rate calculated across two synthetic datasets/SCMs (3var-causal & 3var-noncausal as in Table 1), using the reduced case hyper-parameters, as in Table 2. The acceptance performance is very similar regardless of the classifier being used. Using the refits for the acceptance, we conclude that the classifier does not have a big impact on performance for CE, CR, and ICR; however, ICR has a slightly lower difference between the different classifiers. The detailed results analysis can confirm the latter observations on acceptance and acceptance under refits carried out in Appendix E.

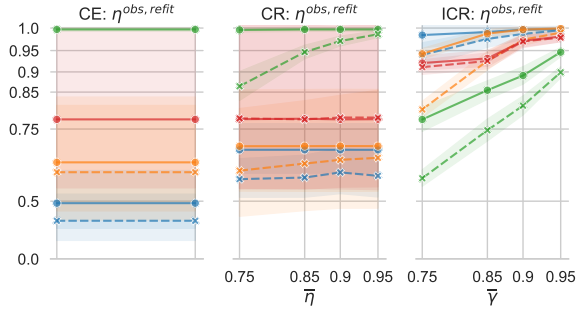
Data shift: To further assess the robustness of our different recourse methods, the data is shifted and it is compared whether a refit of the model leads to invalidation of previous recommendations. Since CE and CR already struggle with refits from the same distribution, it is of no surprise that CE and CR perform even worse when the distribution slightly changes. Variance shifts seem to be slightly worse for the model invalidity than mean shifts. ICR, on the other hand, does not show any of these issues. As can be seen from Table 4, the highest amount of invalidity appears in the subpopulation approach with 6%. This means 6% of the previous recourse recommendations are not valid anymore after the model was refitted on the shifted



(a) Observed improvement rates γ^{obs}



(b) Observed acceptance rates η^{obs}



(c) Observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$

method	**cost**
CE	2.23 ± 0.18
ind. CR	1.33 ± 0.13
sub. CR	2.12 ± 0.17
ind. ICR	3.80 ± 0.15
sub. ICR	3.79 ± 0.16

(d) The recourse cost from reproducing the paper

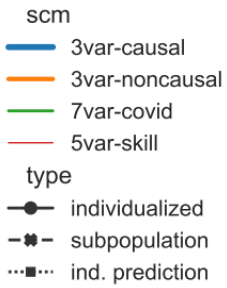


Figure 1: Experimental results for CE, CR and ICR for the reproducibility

recourse	**MLP**	**SVM**	**adaboost**	**logreg**	**random forest**
CE	0.29 ± 0.11	0.26 ± 0.1	0.19 ± 0.04	0.28 ± 0.15	0.26 ± 0.02
ind. CR	0.33 ± 0.08	0.32 ± 0.10	0.20 ± 0.01	0.31 ± 0.16	0.27 ± 0.03
ind. ICR	0.95 ± 0.00	0.95 ± 0.00	0.95 ± 0.01	0.96 ± 0.00	0.95 ± 0.01
sub. CR	0.31 ± 0.13	0.31 ± 0.13	0.18 ± 0.02	0.31 ± 0.16	0.24 ± 0.04
sub. ICR	0.95 ± 0.00	0.95 ± 0.00	0.95 ± 0.01	0.95 ± 0.01	0.96 ± 0.00

Table 3: γ^{obs} for different classifiers with specified confidence of 0.95, on the reduced case scenario

data. This leads to the conclusion that ICR is robust not only to refits from the same distribution but even to refits from distribution shifts. A more detailed comparison of distributional changes and models would be optimal; however, due to the computational resources necessary to calculate recourse, only a small sample of data shifts and models are compared here. All values in Table 4 show the average invalidity calculated across 3var-causal and 3var-noncausal datasets/SCMs.

recourse	**classifier**	**both shift**	**variance shift**	**mean shift**
CE	MLP	0.34 ± 0.34	0.86 ± 0.23	0.82 ± 0.34
	SVM	0.35 ± 0.36	0.89 ± 0.22	0.85 ± 0.32
	adaboost	0.77 ± 0.11	0.90 ± 0.06	0.89 ± 0.06
	logreg	0.46 ± 0.34	0.92 ± 0.16	0.84 ± 0.33
	rf	0.78 ± 0.08	0.90 ± 0.07	0.91 ± 0.05
ind. CR	MLP	0.30 ± 0.30	0.84 ± 0.21	0.80 ± 0.31
	SVM	0.32 ± 0.31	0.88 ± 0.20	0.84 ± 0.28
	adaboost	0.73 ± 0.11	0.88 ± 0.08	0.87 ± 0.06
	logreg	0.40 ± 0.27	0.90 ± 0.15	0.83 ± 0.30
	rf	0.75 ± 0.07	0.88 ± 0.08	0.88 ± 0.07
ind. ICR	MLP	0.05 ± 0.13	0.02 ± 0.03	0.01 ± 0.02
	SVM	0.04 ± 0.10	0.00 ± 0.01	0.00 ± 0.01
	adaboost	0.05 ± 0.09	0.04 ± 0.04	0.04 ± 0.04
	logreg	0.04 ± 0.10	0.01 ± 0.02	0.00 ± 0.01
	rf	0.04 ± 0.07	0.05 ± 0.09	0.05 ± 0.08
sub. CR	MLP	0.33 ± 0.32	0.84 ± 0.22	0.81 ± 0.34
	SVM	0.34 ± 0.32	0.87 ± 0.22	0.84 ± 0.31
	adaboost	0.77 ± 0.11	0.90 ± 0.05	0.90 ± 0.07
	logreg	0.41 ± 0.31	0.90 ± 0.15	0.82 ± 0.32
	rf	0.79 ± 0.07	0.92 ± 0.06	0.90 ± 0.06
sub. ICR	MLP	0.02 ± 0.06	0.03 ± 0.04	0.02 ± 0.02
	SVM	0.02 ± 0.06	0.01 ± 0.02	0.01 ± 0.02
	adaboost	0.05 ± 0.05	0.05 ± 0.06	0.06 ± 0.05
	logreg	0.02 ± 0.06	0.02 ± 0.03	0.01 ± 0.01
	rf	0.03 ± 0.05	0.05 ± 0.08	0.06 ± 0.09

Table 4: Invalidity for the shifted features, averaged across 3var-causal and 3var-noncausal. The demonstrated results are the average and standard deviation for shifts overall features and three iterations, with a user-specified confidence level set to 0.95.

Genetic algorithms: Table 5 compares the yielded improvement rates γ^{obs} between the modified NSGA-II variant, utilized in (König et al., 2023) and NSGA-III (Deb & Jain, 2013) for minimizing the optimization objective in our disposal, as defined in Subsection 3.1.3. The figures present in Table 5 refer to the average improvement rate calculated across two synthetic datasets/SCMs (3var-causal & 3var-noncausal as in Table 1), using the reduced case hyper-parameters. Both genetic algorithms acquire similar performance when targeting improvement. Additional experiments for acceptance rate η^{obs} and $\eta^{obs,refit}$ further compare the two different algorithms and are provided in Appendix G, Tables 9 and 10 respectively. Interestingly, when considering the acceptance under refits, NSGA-III in most cases yield similar, if not higher rates.

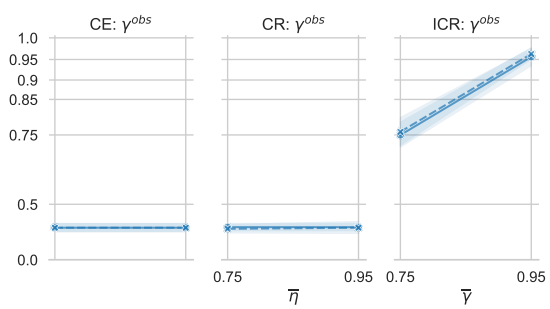
New Dataset: To further assess the generalizability of ICR, we test its performance on a synthetic dataset we created and refer to it as the 3var-causal-nonlinear[1]. The results for the observed improvement γ^{obs}, acceptance rates η^{obs} are shown in Fig. 2a and Fig. 2b, respectively. Additionally, Fig.2c depicts the robustness of refits from the same distribution.

CR attains a perfect rate for acceptance, while individualized ICR and CE follow closely behind. As expected, CE and CR obtain low improvement rate values, whereas ICR consistently leads to improvement for

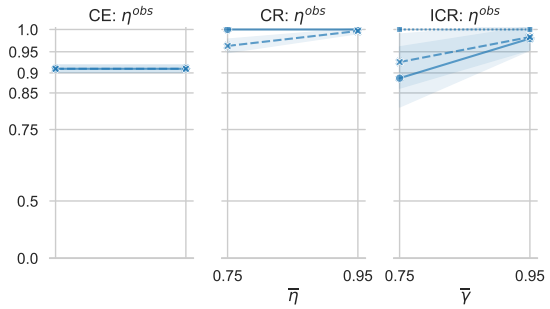
[1]Avid readers can find the SCM model along with the structural equations in the Appendix.A

recourse	NSGA-II	NSGA-III
CE	0.28 ± 0.13	0.31 ± 0.13
ind. CR	0.32 ± 0.14	0.33 ± 0.12
ind. ICR	0.96 ± 0.01	0.98 ± 0.02
sub. CR	0.31 ± 0.15	0.32 ± 0.14
sub. ICR	0.95 ± 0.03	0.95 ± 0.02

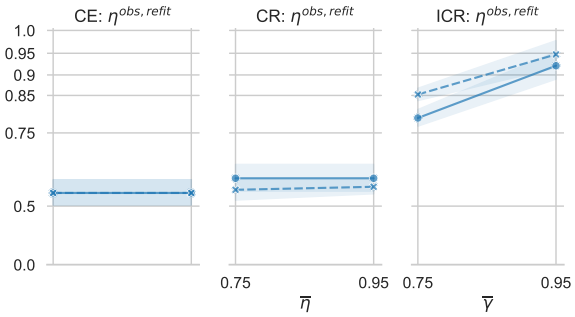
Table 5: γ^{obs} (observed rate $\pm$ standard deviation) of each genetic algorithm achieved with user-specified confidence of 0.95 on the reduced hyper-parameter case scenario. All rates in the table have been rounded to the third decimal place.



(a) Observed improvement rates γ^{obs}



(b) Observed acceptance rates η^{obs}



(c) Observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$

Figure 2: Experimental results for CE, CR and ICR on the 3var-causal-nonlinear dataset

both user-specified confidence levels. Ultimately, ICR is the prevailing method when testing for robustness regarding refits on the same data, confirming the empirical claims made by the authors.

5 Discussion

König et al. (2023) distinguish two purposes for contrastive explanations: contestability of algorithmic decisions and actionable recourse recommendations. ICR targets improvement, which is a necessity of actual recourse. Thus, recourse is achieved by improving the underlying condition rather than just the features that can game the predictor model.

Our contributions were twofold: firstly, we reproduced the experiments by König et al. (2023) and provided evidence of their claims' validity. While it was impossible to replicate the exact numbers of the authors due to how the seeds were set, we can replicate the trends of all claims. Secondly, we assessed the robustness of the ICR claim against different model fits and data fits, as well as the generalizability across a new dataset.

Our additional experiments tested different classifiers, an alternative genetic algorithm for minimizing the optimization objective, and the dataset's mean and variance distributional shifts.

To test the influence of genetic algorithms in the minimization objective, we adapted the NSGA-III algorithm to also make use of the same crowding distance and principles as in the NSGA-II variant used by the authors, inspired from (Dandl et al., 2020), we discovered that it performs equally well and even attains better outcome at some dataset/SCMs runs. Since these very similar genetic algorithms acquire similar performance, it would be interesting to try out other evolutionary and/or genetic algorithms that specifically target single-objective functions, aiming to derive better recourse recommendations and reduce the computation time spent during the optimization phase. One research direction that has yet to be explored is effectively using the multi-objective capabilities of the two NSGA variants. As for future research, we aim to optimize for improvement and acceptance rate jointly, which was suggested during our correspondence with the author. Specifically, one could jointly target improvement rate and cost and let the user choose from the Pareto front.

Concerning the robustness assessment, we can verify to a greater degree that ICR is more robust than CE and CR, specifically towards mean and variance shifts in the data. However, given more computational resources, we would like to conduct a more extensive assessment by testing different magnitudes in shifts. As for the generalizability experiment with the additional dataset 3var-causal-nonlinear, the evidence partially points towards the generalizability strength of ICR since the trends are similar to the performance for the larger non-linear datasets. Nevertheless, they also closely follow the performance trends of the other 3var datasets.

A possible limitation of our experiments on robustness is that we ran them on a down-scaled set of hyper-parameters. Even though running on the complete set of hyperparameters would make a difference in the reliability of our conclusions, we must recognize the significant computational resources that the original experiments require. The environmental impact of computationally expensive methods is a solid motivation for further research into making ICR more efficient and effective.

What was easy The public repository containing the original code was well-structured and documented. The provided scripts to produce and visualize the results were very helpful, and thus, analyzing and comparing our results was reasonably straightforward. Furthermore, the original paper provided detailed information on implementation details and theoretical background in the appendix.

What was difficult The initial reproduction of the complete experiments proved quite computationally and time-intensive. Some inconsistencies existed between the experiment details specified in the paper and the repository instructions. Moreover, the authors provided a seed in their code run, but it could not help us achieve the exact outcome with them. The documentation of the code base was lacking, making it difficult to understand how the data was generated and the implementation of each method.

Communication with original authors We have contacted the paper's first author to ask for clarification of the theoretical aspects and some technical parts of the code. Moreover, we asked for some feedback on the proposed extensions. Although a bit late, the author responded to all of our questions, while he found our extensions interesting and provided constructive feedback on them. A new direction for research was also proposed, not implemented in the author's paper but was only hinted at.

References

Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. Multi-objective counterfactual explanations. In *International Conference on Parallel Problem Solving from Nature*, pp. 448–469. Springer, 2020.

Kalyanmoy Deb and Himanshu Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints. *IEEE transactions on evolutionary computation*, 18(4):577–601, 2013.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

Lara Jehi, Xinge Ji, Alex Milinovich, Serpil Erzurum, Brian P Rubin, Steve Gordon, James B Young, and Michael W Kattan. Individualizing risk prediction for positive coronavirus disease 2019 testing: results from 11,672 patients. *Chest*, 158(4):1364–1375, 2020.

Amir-Hossein Karimi, Julius Von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. *Advances in neural information processing systems*, 33:265–277, 2020.

Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 353–362, 2021.

Amir-Hossein Karimi, Julius von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Towards causal algorithmic recourse. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pp. 139–166. Springer, 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Gunnar König, Timo Freiesleben, and Moritz Grosse-Wentrup. Improvement-focused causal recourse (icr) github, 2022. URL `https://github.com/gcskoenig/icr`.

Gunnar König, Timo Freiesleben, and Moritz Grosse-Wentrup. Improvement-focused causal recourse (icr). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11847–11855, 2023.

João Eduardo Montandon, Marco Tulio Valente, and Luciana L. Silva. Mining the technical roles of github users. *Information and Software Technology*, 131:106485, 2021. ISSN 0950-5849. doi: https://doi.org/10.1016/j.infsof.2020.106485. URL `https://www.sciencedirect.com/science/article/pii/S0950584920302275`.

Ziad Obermeyer and Sendhil Mullainathan. Dissecting racial bias in an algorithm that guides health decisions for 70 million people. In *Proceedings of the conference on fairness, accountability, and transparency*, pp. 89–89, 2019.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Judea Pearl et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2):3, 2000.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Manish Raghavan, Solon Barocas, Jon Kleinberg, and Karen Levy. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 469–481, 2020.

Kaivalya Rawal, Ece Kamar, and Himabindu Lakkaraju. Algorithmic recourse in the wild: Understanding the impact of data and model shifts. *arXiv preprint arXiv:2012.11788*, 2020.

Robert E Schapire. Explaining adaboost. In *Empirical inference*, pp. 37–52. Springer, 2013.

Sohini Upadhyay, Shalmali Joshi, and Himabindu Lakkaraju. Towards robust and reliable algorithmic recourse. *Advances in Neural Information Processing Systems*, 34:16926–16937, 2021.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Jiaming Zeng, Berk Ustun, and Cynthia Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 180(3):689–722, 2017.

A Dataset Information

This section will provide more information about the dataset we used. Moreover, we have added each dataset's causal graph and structural equation in the following figures 3 - 7.

3var-causal: A linear Gaussian SCM with a binary target Y, having all other features influencing it.

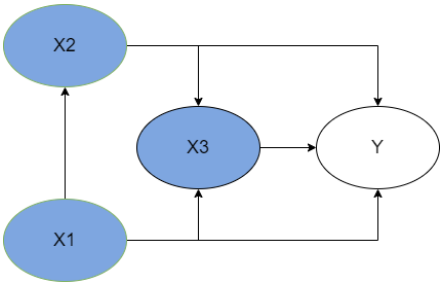
3var-noncausal: Similar to the 3var-causal, but one feature is affected by Y.

5var-skill: A categorical semi-synthetic SCM where the target is the programming skill level based on causes like university degree and non-causal factors obtained from GitHub such as commit count. This dataset was inspired by Montandon et al. (2021)

7var-covid: A semi-synthetic dataset replicated by a real-world COVID screening model provided by (Jehi et al., 2020). The model has causes like COVID-19 vaccination and population density, including symptoms like fever and fatigue. The dataset illustrates a mix of categorical and continuous data with various noise distributions. Their relationships include nonlinear structural equations.

3var-causal-nonlinear A fully synthetic non-linear SCM with a binary target Y, having all other features influencing it.

In the following cost equations, we define δ as the vector of absolute changes to the intervened-upon variables.

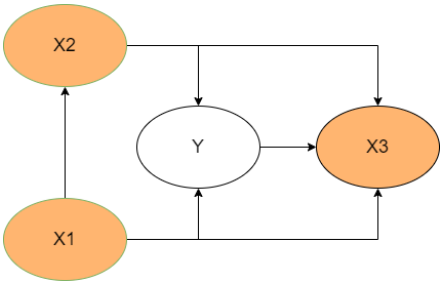


Causal graph

$$
\begin{aligned}
X_1 &:= U_1, & U_1 &\sim \mathcal{N}(0,1) \\
X_2 &:= X_1 + U_2, & U_2 &\sim \mathcal{N}(0,1) \\
X_3 &:= X_1 + X_2 + U_3, & U_3 &\sim \mathcal{N}(0,1) \\
Y &:= \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + U_Y, & U_Y &\sim \text{Uniform}(0,1).
\end{aligned}
$$

Structural Equations

Figure 3: SCM for 3var-causal with cost(a) $= \delta_1 + \delta_2 + \delta_3$

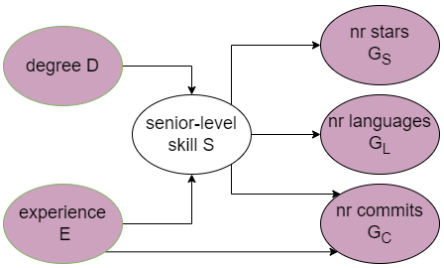


Causal graph

$$
\begin{aligned}
X1 &:= U1, & U1 &\sim \mathcal{N}(0,1) \\
X2 &:= X1 + U1, & U1 &\sim \mathcal{N}(0,1) \\
Y &:= [\sigma(X1 + X2) \leq UY], & UY &\sim Unif(0,1) \\
X3 &:= X1 + X2 + Y + U3, & U3 &\sim \mathcal{N}(0,1)
\end{aligned}
$$

Structural Equations

Figure 4: SCM for 3var-noncausal with cost(a) $= \delta_1 + \delta_2 + \delta_3$

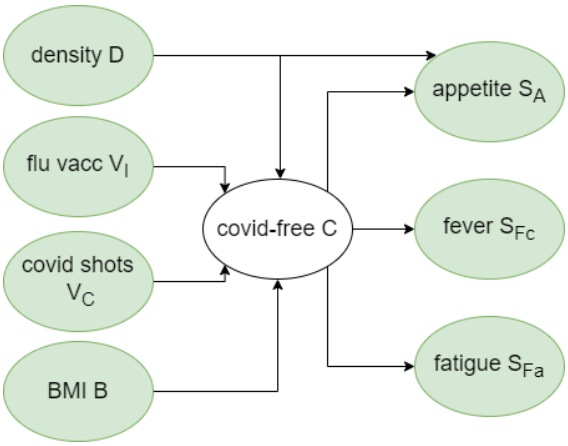


Causal graph

$$
\begin{aligned}
E &:= UE, & UE &\sim \text{GaP}(8, 8/3)\\
D &:= UD, & UD &\sim \text{Cat}(0.4, 0.2, 0.3, 0.1)\\
S &:= [\sigma(-10 + 3E + 4D) \leq US], & US &\sim \text{Unif}(0, 1)\\
GC &:= 10E(11 + 100D) + UGC, & UGC &\sim \text{GaP}(40, 40/4)\\
GL &:= \sigma(10S) + UGL, & UGL &\sim \text{GaP}(2, 2/4)\\
GS &:= 10S + UGS, & UGS &\sim \text{GaP}(5, 5/4)
\end{aligned}
$$

Structural Equations

Figure 5: SCM for 5var-skill with cost(a) $= = 5\delta_E + 5\delta_D + 0.0001\delta_{G_C} + 0.01\delta_{G_L} + 0.1\delta_{G_S}$

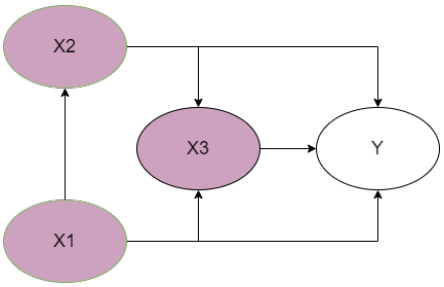


Causal graph

$$
\begin{aligned}
D &:= UD, UD \sim \Gamma(4, 4/3)\\
VI &:= UVI, UVI \sim \text{Bern}(0.39)\\
VC &:= UVC, UVC \sim \text{Cat}(0.24, 0.02, 0.15, 0.59)\\
B &:= UB, UB \sim \mathcal{N}(0, 1)\\
C &:= [\sigma\left(-\left(D - 3 - VI - 2.5VC + 0.2B^2\right)\right) \leq UC];\\
UC &\sim \text{Unif}(0, 1)\\
SA &:= [\sigma(-2C) \leq USA], USA \sim \text{Unif}(0, 1)\\
SF_e &:= [\sigma(5 - 9C) \leq USF_e], USF_e \sim \text{Unif}(0, 1)\\
SF_a &:= \sigma\left(-1 + B^2 - 2C\right) \leq USF_a, USF_a \sim \text{Unif}(0, 1)
\end{aligned}
$$

Structural Equations

Figure 6: SCM for 7var-covid with cost(a) $= \delta_D + \delta_{VI} + \delta_{VC} + \delta_B + \delta_{SA} + \delta_{SF_e} + \delta_{SF_a}$.



Causal graph

$$
\begin{aligned}
X_1 &:= U_1, & U_1 &\sim \mathcal{N}(0, 1)\\
X_2 &:= X_1 + U_2, & U_2 &\sim \mathcal{B}|\nabla\backslash\wr\sqcap\updownarrow\updownarrow\rangle(0.60)\\
X_3 &:= -0.05 * X_1 + 0.25X_2^2 + U_3, & U_3 &\sim \mathcal{N}(0, 0.1)\\
Y &:= \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + U_Y, & U_Y &\sim \text{Uniform}(0, 1).
\end{aligned}
$$

Structural Equations

Figure 7: SCM for 3var-nonlinear with cost(a) $= \delta_1 + \delta_2 + \delta_3$

B Hyperparameters for reproducibility study

In table 6, we present the hyperparameters used to reproduce the author's results.

Data set	Number of observations	Number of individuals having recourse calculation	Confidence	Number of Generations	POP SIZE	n digits	nr refits
3var-noncausal	4000	200	0.75, 0.85, 0.9, 0.95	600	300	1	5
3var-causal	4000	200	0.75, 0.85, 0.9, 0.95	600	300	1	5
5var-skill	4000	200	0.75, 0.85, 0.9, 0.95	1000	500	1	5
7var-covid	20000	200	0.75, 0.85, 0.9, 0.95	1000	500	1	5

Table 6: Hyperparameters based on the original paper.

C Experiment metrics

Experiment 1: Do CE, CR, and ICR lead to improvement? The observed improvement rates γ_{obs} was the metric to assess the data. In the setting where the structural equations are assumed, it is possible to acquire individualized improvement confidence. The subpopulation-based improvement confidence is derived in a setting where only the causal graph is assumed.

Experiment 2: Do CE, CR, and ICR lead to acceptance (by pre- and post-recourse predictor)? Recourse recommendations should lead to improvement and change the classifier's original decision. Whether acceptance naturally ensues from the improvement rate depends on the ability of the predictor to recognize improvements. Thus, the metric calculated here is the observed acceptance rates η^{obs} w.r.t. the optimal pre-recourse observational predictor h^*; and in the case of individualized ICR additionally w.r.t. the individualized post-recourse predictor h^*_{ind}, in order to account for an imbalance between ICR and the predictor.

Experiment 3: Do CE, CR, and ICR lead to acceptance by other predictors with comparable test error? The metric deployed for the experiment is the observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$

Experiment 4: How costly are CE, CR, and ICR recommendations? For the last experiment, the authors used the average recourse cost for rejected individuals and were consequently provided with a recourse recommendation. The cost is defined differently for each dataset and can be found in Appendix 3 of the original paper.

D Robustness on shifted data

The 3var datasets consist of Standard Normal Distributions (mean 0 and variance 1) that are causally related. For each feature, we create a new dataset by shifting once the mean (from 0 to 0.5), once the variance (from 1.0 to 0.5), and once both (mean from 0 to 0.5 and variance from 1.0 to 0.5). Due to the causal relationships, a shift for x1 also affects all children of x1. A similar procedure for shifting the data was used by Upadhyay et al. (2021).
We have our unshifted data D_1 and model M_1, which is trained on D_1. Now, we shift one feature by a specific mean or variance or both and then create a new dataset D_2. On this data, we create the model M_2. We used 50% for model training and 50% for the validation, like König et al. (2023) did. Recourse is applied to data D_1 to revert the decision of M_1. Invalidity calculates how many individuals' recourse recommendations are invalid after the data shift. To implement this, the recourse recommendation is used as an input for M_2, and a recourse is marked as invalid if M_2 predicts the decision as 0. Therefore, the recourse recommendation did not change the algorithm's decision. The process of calculating the invalidity was implemented by Rawal et al. (2020).

E Robustness of the classifier

Tables 7 and 8 depict our experimental results for the robustness of ICR when considering different classifiers.

recourse	MLP	SVM	adaboost	logreg	random forest
CE	0.98 ± 0.00	0.98 ± 0.00	0.93 ± 0.03	0.96 ± 0.05	0.88 ± 0.04
ind. CR	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
ind. ICR	1.00 ± 0.01	1.00 ± 0.00	0.98 ± 0.01	1.00 ± 0.00	0.99 ± 0.00
sub. CR	0.99 ± 0.00	0.98 ± 0.02	1.00 ± 0.00	0.99 ± 0.00	1.00 ± 0.01
sub. ICR	1.00 ± 0.01	1.00 ± 0.00	0.98 ± 0.02	1.00 ± 0.00	0.99 ± 0.01

Table 7: η^{obs} of different classifiers with confidence 0.95, reduced datasets for the classifiers

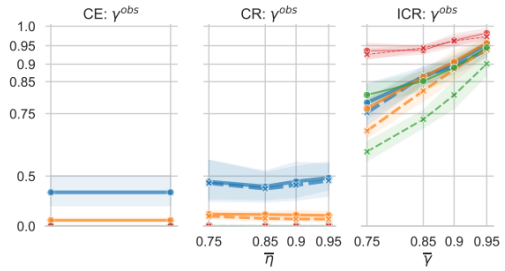
recourse	MLP	SVM	adaboost	logreg	random forest
CE	0.42 ± 0.15	0.41 ± 0.12	0.39 ± 0.01	0.54 ± 0.1	0.42 ± 0.01
ind. CR	0.48 ± 0.13	0.46 ± 0.11	0.41 ± 0.01	0.59 ± 0.08	0.45 ± 0.01
ind. ICR	1.00 ± 0.01	1.00 ± 0.00	0.98 ± 0.02	1.00 ± 0.0	0.98 ± 0.02
sub. CR	0.44 ± 0.17	0.43 ± 0.13	0.41 ± 0.0	0.57 ± 0.09	0.43 ± 0.01
sub. ICR	0.99 ± 0.01	1.00 ± 0.00	0.98 ± 0.02	1.00 ± 0.00	0.97 ± 0.02

Table 8: $\eta^{obs,refit}$ of different classifiers with confidence 0.95, reduced datasets for the classifiers

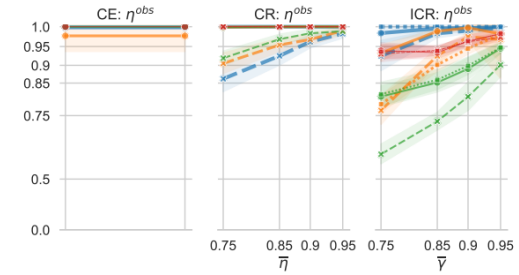
F Trend comparison

In this section, we provide all the results of König et al. (2023) next to our findings for the reproducibility part.

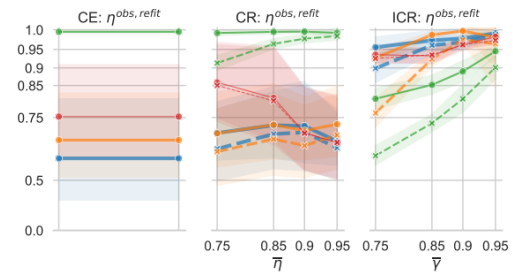
From authors



Observed improvement rates γ^{obs} (authors, Q1)



Observed acceptance rates η^{obs} w.r.t. h^*; for ind. ICR additionally w.r.t. $h^{*,ind}$ (authors, Q3)

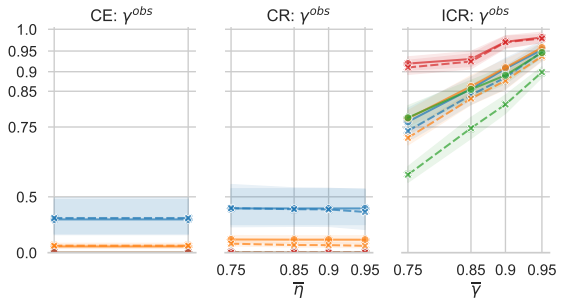


Observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$ (authors, Q5)

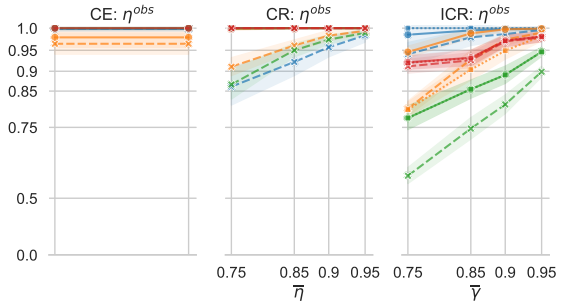
method	cost
CE	1.82 ± 1.09
ind. CR	1.34 ± 1.14
sub. CR	1.65 ± 1.02
ind. ICR	4.26 ± 3.34
sub. ICR	4.20 ± 3.33

cost table (authors, Q7)

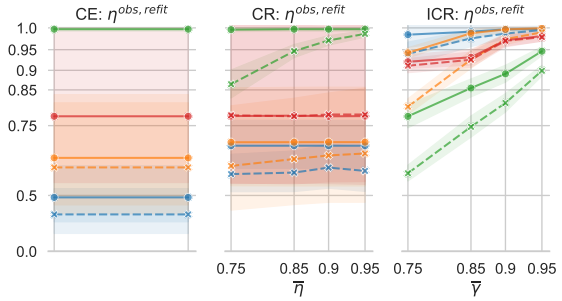
Ours



Observed improvement rates γ^{obs} (ours, Q2)



Observed acceptance rates η^{obs} w.r.t. h^*; for ind. ICR additionally w.r.t. $h^{*,ind}$ (ours, Q4)



Observed acceptance rates for other fits with comparable test set performance $\eta^{obs,refit}$ (ours, Q6)

method	cost
CE	2.23 ± 0.18
ind. CR	1.33 ± 0.13
sub. CR	2.12 ± 0.17
ind. ICR	3.80 ± 0.15
sub. ICR	3.79 ± 0.16

cost table (ours, Q8)

Figure 8: Trend comparison

G Robustness of the genetic algorithm

The following Tables 9 and 10 depict the experimental results for acceptance rate η^{obs} and $\eta^{obs,refit}$, respectively, to compare the robustness of ICR when using two different algorithms for minimizing the specified optimization problem.

recourse	NSGA-II	NSGA-III
CE	0.96 ± 0.07	0.96 ± 0.08
ind. CR	1.00 ± 0.00	1.00 ± 0.00
ind. ICR	1.00 ± 0.00	0.99 ± 0.00
sub. CR	0.99 ± 0.01	0.99 ± 0.01
sub. ICR	0.99 ± 0.00	0.99 ± 0.00

Table 9: η^{obs} (observed rate $\pm$ standard deviation) of each genetic algorithm achieved for user-specified confidence of 0.95, using the reduced case scenario hyper-parameters, as in Table 2. All rates present in the table have been rounded to the third decimal place.

recourse	NSGA-II	NSGA-III
CE	0.54 ± 0.16	0.54 ± 0.12
ind. CR	0.59 ± 0.13	0.59 ± 0.09
ind. ICR	1.00 ± 0.00	0.99 ± 0.00
sub. CR	0.57 ± 0.15	0.57 ± 0.12
sub. ICR	0.99 ± 0.00	0.99 ± 0.00

Table 10: $\eta^{obs,refit}$ (observed rate $\pm$ standard deviation) of each genetic algorithm achieved for user-specified confidence of 0.95, using the reduced case scenario hyper-parameters, as in Table 2. All rates present in the table have been rounded to the third decimal place.

H Computational Resources

In the following Figure 9, the effect of multiprocessing on the reproduction speed of different methods is illustrated.

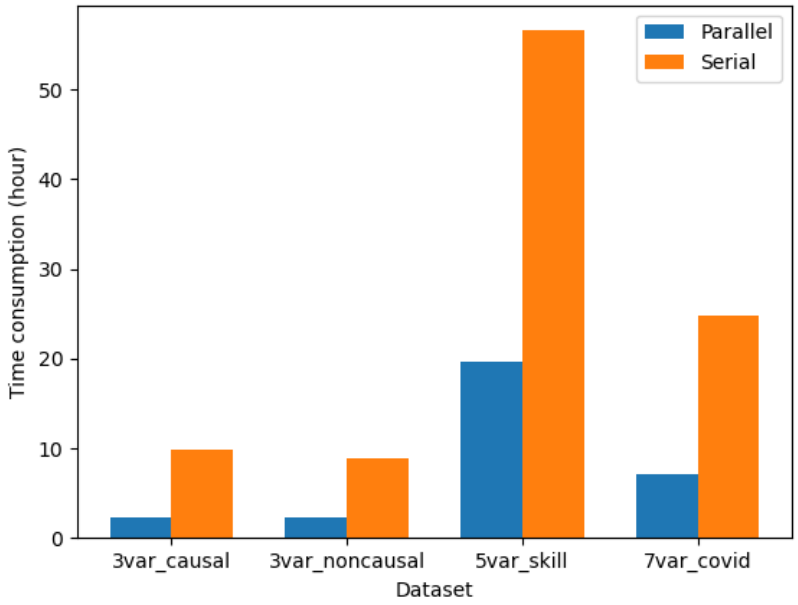


Figure 9: Time usage comparison for the original experiments with original (linear) method and the improved one (parallel)