The Game Of Recourse: Simulating Algorithmic Recourse over Time to Improve Its Reliability and Fairness

Andrew Bell*
New York University
New York, NY, USA
alb9742@nyu.edu

Joao Fonseca*
NOVA IMS
Lisbon, Portugal
jpfonseca@novaims.unl.pt

Julia Stoyanovich
New York University
New York, NY, USA
stoyanovich@nyu.edu

ABSTRACT

Algorithmic recourse, or providing recommendations to individuals who receive an unfavorable outcome from an algorithmic system on *how they can take action and change that outcome*, is an important tool for giving individuals agency against algorithmic decision systems. Unfortunately, research on algorithmic recourse faces a fundamental challenge: there are no publicly available datasets on algorithmic recourse. In this work, we begin to explore a solution to this challenge by creating an agent-based simulation called The Game Of Recourse (an homage to Conway's Game of Life) to synthesize realistic algorithmic recourse data. We designed The Game Of Recourse with a focus on *reliability* and *fairness*, two areas of critical importance in socio-technical systems. You can access the application at https://game-of-recourse.streamlit.app.

CCS CONCEPTS

• **Information systems → Data management systems**; • **Social and professional topics → Socio-technical systems**; • **Human-centered computing**;

KEYWORDS

algorithmic recourse, fairness, reliability, ranking, data generation, temporal data, simulation

ACM Reference Format:
Andrew Bell, Joao Fonseca, and Julia Stoyanovich. 2024. The Game Of Recourse: Simulating Algorithmic Recourse over Time to Improve Its Reliability and Fairness. In *Companion of the 2024 International Conference on Management of Data (SIGMOD-Companion '24), June 9–15, 2024, Santiago, AA, Chile.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3626246.3654742

1 INTRODUCTION

Algorithmic recourse refers to the provision of recommendations or actions that individuals can take after receiving an unfavorable outcome from an algorithmic system. This is aimed at empowering individuals to challenge or change the decision made by the system, thereby providing them with a sense of agency in the face of automated decision-making processes.

In principle, recourse can be provided for any type of an algorithmic system, including *predictive algorithms* (*i.e.,* machine learning classifiers) and *rankers*. For example, if a predictive algorithm is being used to evaluate whether or not an individual is a good candidate for a credit loan, algorithmic recourse would imply providing recommendations to denied applicants on what actions they can take to be approved if they apply again at a later date. These *recourse recommendations* may take the form of "you need to raise your credit by Y points," or "you need to reduce your debt by $\$X$." In settings where rankers are used, low-ranked items may be provided recourse recommendations on how they can improve their rank, or become a top-k ranked item.

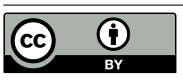
Unfortunately, research on algorithmic recourse faces a fundamental data challenge: the entirety of well-cited work on recourse relies on popular open and archival datasets that were *not* created with algorithmic recourse as their primary purpose (*e.g.,* Adult, German Credit, and COMPAS). This limitation has been noted by us and others [2, 4, 5]. To our knowledge, there are no public real-world algorithmic recourse datasets. In fact, even for the near-universally accepted example of "recourse when applying for a bank loan," there is no publicly available data. Yet, the lack of real-world data does not mean that research on algorithmic recourse should be abandoned. In fact, studying recourse and its related challenges (*i.e.,* robustness, fairness) is arguably more important now than ever: the EU AI Act mandates recourse for many algorithmic systems.[1]

Rather than continuing to adapt existing datasets to algorithmic recourse problems, we explore an alternative solution to the data challenge: creating an agent-based simulation to synthesize realistic recourse data. We call this simulator The Game Of Recourse, and discuss its implementation and use-cases in this work.

2 SYSTEM OVERVIEW

The Game Of Recourse was built using Python, with Streamlit for the front-end interface, and implements an agent-based modeling algorithm created by the authors in previous work [5]. In Section 2.1, we describe multi-agent recourse model. Then, in Sections 2.2 and 2.3, we motivate reliability and fairness in recourse—both quantified over time—and briefly describe how these desiderata may be achieved, also based on our prior work [3, 5]. Note that a unique challenge for synthesizing recourse data is the *lack of benchmarks*. For example, we know very little about how people behave in response to recourse recommendations. We designed The Game Of Recourse to be as flexible as possible, and allow for many parameter settings to produce a range of different datasets.

*Both authors contributed equally to this research.



SIGMOD-Companion '24, June 9–15, 2024, Santiago, AA, Chile

ACM ISBN 979-8-4007-0422-2/24/06
https://doi.org/10.1145/3626246.3654742

[1] Annex IV, Section 2; https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52021PC0206

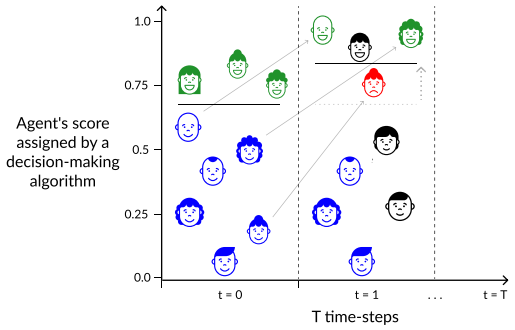


Figure 1: Reproduced from [5]. The x-axis shows time-steps t, and the y-axis shows agent scores $f(x_t)$. There are $k = 3$ positive outcomes available at each time-step. At $t = 0$, green agents receive a positive outcome ($f(x_0) \geq s_0$, where s_0 is represented by the horizontal line), and blue agents receive a negative outcome along with a recourse recommendation x'. At time $t = 1$, new agents N_1, shown in black, enter the environment. Grey arrows show action. The agent shown in red acted on the recourse recommendation, but (disappointingly) its effort turned out to be insufficient because competition from other agents "raised the bar."

2.1 Multi-Agent Recourse

Consider a population of agents P. Each $x \in P$ is described by a set of features $x \in \mathcal{X}$ and is evaluated for a desired outcome (*e.g.,* applying for a loan) by a classifier or ranker $f : \mathcal{X} \rightarrow [0, 1]$. Agents are competing for a limited number of positive outcomes k (*e.g.,* the number of available loans) over a series of timesteps $t = \{0, 1, ..., T\}$. An agent may change its features over time, and we use x_t to refer to the state of x at time t.

Time is intrinsic in algorithmic recourse, as it often involves a first unsuccessful attempt at a favorable outcome at time t, followed by one or several subsequent attempts at times $t + \delta_1, t + \delta_2, ...t + \delta_n$. At each timestep, a score $f(x_t)$ is calculated for each agent. This score is used to assign outcomes: positive when $f(x_t) \geq s_t$, where s_t is the score cut-off at time t, and negative when $f(x_t) < s_t$. Agents who receive a negative outcome also get a *recourse recommendation* $x' \in \mathcal{X}$ that satisfies two conditions: *(i)* $f(x') \geq s_t$ and *(ii)* x' is associated with the lowest cost $c(x, x')$ of making the change:

$$x' = \min_{x'} \quad c(x_t, x') \text{ s.t. } \quad f(x') \geq s_t, x' \in \mathcal{X}$$

Agents who receive the positive outcome exit the simulation. A new set of agents N_t enters the simulation at each time step. Figure 1 illustrates the dynamics of our simulation.

To model the likelihood that an agent x_t will act on a recourse recommendation x', and the amount by which they change their features, an *action* function $a : \mathcal{X} \times \mathcal{X} \rightarrow \mathcal{X}$ is used. As part of this function, we consider three important considerations about agents' reasonable behavior with respect to recourse:

Effort. This consideration refers to *the likelihood of an agent to take any action.* It is determined by several factors, like their implicit willingness to take on challenges or the amount of effort the action requires. For example, if an agent is told to increase their credit score by 20 points to qualify for a loan, they may be more likely to make the effort as opposed to being told to increase it by 200 points. When agents have a fixed willingness to act on recourse we call it *constant* effort, and when agents base their decision to take action on the magnitude of required change, we call it *flexible* effort.

Adaptation. This consideration refers to *how faithfully an agent follows the recourse recommendation.* Agents may follow the recourse recommendation *exactly*, or they may outperform (or underperform) the recommendation. Returning to the loan example, if an individual is told to increase their credit score by 50 points, they may do so exactly, or they may actually increase their score by 40 point, or by 60 points. We call the behavior where agents exactly match recourse recommendations *binary* adaptation, and otherwise, we call it *continuous* adaptation.

Global difficulty of recourse. The third consideration refers to the *difficulty of acting on a recourse recommendation.* For example, it may be easier to act on a recommendation when it is related to appealing a social media ban versus improving one's credit score. The parameter $g \in [0, 1]$ is set *a priori* for the simulation, and values of g closer to 1.0 indicate a setting recourse is easier. Note that we don't model an individual level of persistence—*i.e.,* agents choosing to stop trying for recourse in the middle of the simulation—because agents that "drop out" do not affect the threshold. However, if desired, our framework can be adapted to accommodate this.

2.2 Reliability in Recourse

Continuously changing contexts can weaken the reliability of recourse recommendations over time due to data and model drift [1, 4, 5]. For example, in the lending setting, an individual may be told that their loan application was denied because their credit score is 50 points lower than necessary. One could imagine that it takes the individual 6 months to a year to improve their credit score — which is enough time for the criteria for approving the loan to change. There are numerous reasons why selection criteria can change over time, including data drift, model drift, and competition between agents. An illustration of the competitive effects and their impact on recourse can be seen in Figure 1.

Metrics. In prior work [5], we defined a recourse reliability measure that quantifies the proportion of agents who acted on recourse and *received* a positive outcome, out of all those agents who acted on recourse and *expected* a positive outcome. Intuitively, recourse reliability measures how well recourse expectations of the agents are met. We quantify recourse reliability in the The Game Of Recourse simulation, allowing system designers to explore how real-world settings impact recourse reliability. They can then control the levers at their disposal to improve recourse reliability in practice (*e.g.,* by limiting the number of applications they accept) or to add confidence intervals to recourse recommendations.

2.3 Fairness in Recourse

There is an emerging body of work demonstrating the importance of fairness in algorithmic recourse, especially because it can allow marginalized and vulnerable individuals to counteract adverse algorithmic decisions [6, 7]. Unfortunately, there is a well-documented *disconnect between fairness in classification and fair recourse*—even

under fair decision making, no guarantees can be made about the fairness of recourse for that classifier. In The Game Of Recourse, we include two important considerations for fairness in recourse: *(i)* whether initial qualification data is biased and *(ii)* what method will be used to mitigate unfairness.

The initial data distribution. Fairness in recourse is highly dependent on the initial feature distributions of individuals. If a disadvantaged group has substantially lower feature values as compared to an advantaged group, the acting on recourse recommendations and achieving the positive outcome will require substantially more effort. Figure 2 illustrates this, showing this for a classifier that is "fair" under demographic parity.

In The Game Of Recourse, support three initial data distribution settings. The first is when features for all individuals are drawn from the same distribution (*i.e.,* no bias). The second is when the data is strongly biased, and there is no parity at all between individuals in the groups; imagine Figure 2, but if all those individuals to the right of the vertical dashed line were not present. Third, we allow users to generate data according to the distribution shown in Figure 2. For the latter two options, the amount of bias is controlled by the *disparity in qualifications* parameter (he number of standard deviations between the means of the feature distributions of the advantaged and disadvantaged groups, μ_a and μ_d), where higher values imply stronger bias.

Metrics. When disparities in qualifications are present between groups, unfairness in recourse can arise. We quantify this unfairness using two metrics, proposed in our prior work [3]: *(i)* effort-to-recourse disparity ratio, where values over 1.0 mean that the disadvantaged group is exerting more effort per successful recourse event than the advantaged group; and *(ii)* time-to-recourse difference that captures the expected number of additional timesteps it takes for a member of the disadvantaged group to achieve recourse.

Bias mitigation strategies. There are three known strategies for mitigating unfairness in recourse, which are all implemented in The Game Of Recourse. The first, proposed by us in recent work, is *Circumstance-Normalized Selection (CNS)* [3] —a post-processing intervention based on *rank-aware proportional representation* by [8]. It involves assigning positive outcomes to the highest-scoring individuals from each sub-population, proportionally by population size. The second is a pre-processing intervention known as *Counterfactual Data Augmentation (CDA)* [4]. It works by augmenting the initial data with counterfactuals for individuals who received the negative outcome, and then re-training the classifier (or ranker) on this new data. The third mitigation strategy is *Group Regularized Recourse (GRR)* [6], which involves re-positioning the decision boundary of a classifier during training to be equidistant from negatively-classified individuals from different groups. In The Game Of Recourse, we also implement a fourth method, which is a combination of CNS and CDA, proposed by us in recent work [3].

2.4 User Interface

Configuration panel. One advantage of The Game Of Recourse is its versatility: there are up to 10 parameters that can be tuned before running a simulation. (We list the parameters in the caption of Figure 3 for convenience.) The Game Of Recourse features a sidebar on the left side of the interface. First, the user defines a "population," which includes determining aspects like the number of sensitive groups and how features are distributed. Second, the user defines the "enviornment" (*i.e.,* how the agents behave) and specifies whether a fairness intervention should be applied. An optional random state parameter allows users to replicate past simulations or create new simulations with the same parameters. Lastly, there is a *button to download to the simulated data.*

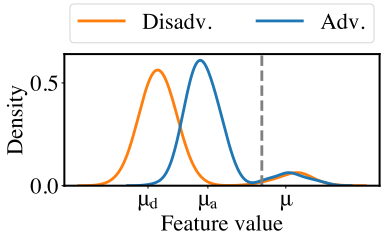


Figure 2: Let individuals to the right of the vertical dashed line receive a positive outcome; then decision-making is fair with respect to *Demographic Parity* between the advantaged and disadvantaged groups, but recourse is unfair.

Main panel. The main panel contains four widgets, arranged vertically. The *Introduction* widget provides general information about The Game Of Recourse. The *Visualize initial population* widget provides an overview of the data defined in the "population" configuration. The initial population data can be visualized in a scatter plot, or, alternatively, manually inspected.

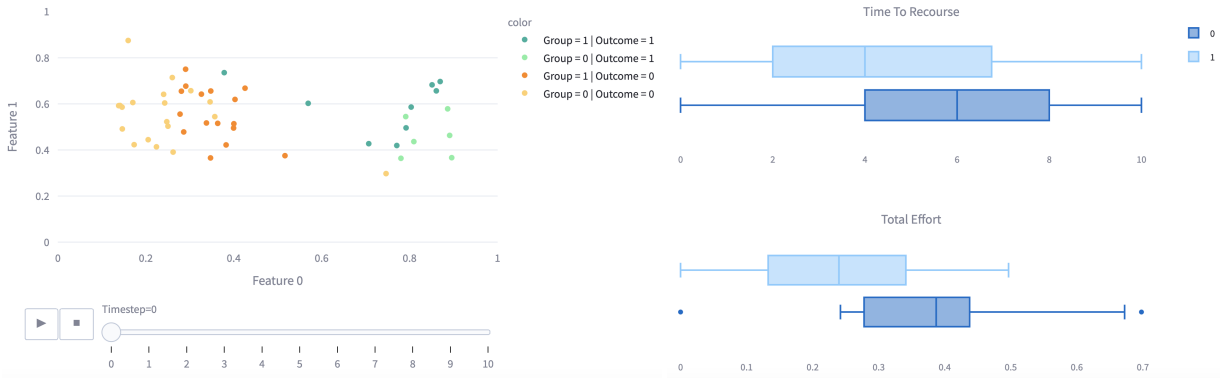
The *Explore environment* widget contains the plot seen in Figure 3(a). The simulation can be visualized in two different interactive scatter plots: either as a function of ranker score and timestep, or within the feature space (*i.e.,* see how agent features are changing). Both visualizations contain a play and stop button to see the progression of the agents through the simulation.

The fourth widget, *Inspect simulation*, is split into three tabs. The "Agent Info" tab contains a table with metadata about every agent that entered the simulation at some point in time. The "Simulation Metrics" tab, shown in Figure 3(b), presents two box plots about the number of timesteps required for agents to achieve the outcome (*i.e.,* time-to-recourse), the score variation incurred to achieve the outcome (*i.e.,* total effort), both from [3], and a group-wise recourse reliability metric from [5]. In biased population settings, the effort-to-recourse ratio and time-to-recourse difference are displayed, with the disadvantaged population as the reference group.

3 DEMONSTRATION PLAN

To demonstrate The Game Of Recourse, we will take the audience through three different scenarios: exploring recourse reliability, fairness, and bias mitigation strategies. We will also discuss how the generated data can be used by *(i)* researchers to study open questions in algorithmic recourse, and *(ii)* system owners to design better recourse in practice. We also hope to spark an interest in our audience about using agent-based modeling to simulate data.

Scenario 1: The standard algorithmic recourse setting. Our first scenario illustrates the standard use of The Game Of Recourse. We



(a) Animation of recourse over time **(b) Simulation metrics panel**

Figure 3: Screenshots of the fairness metrics and recourse over time widgets. The application has the following parameters; (1) Initial agents: number of agents at timestep; (2) New agents: the number of new agents entering the environment at each timestep; (3) Distribution type: the distribution of agents' features; (4) Qualification (bias factor): the distance between feature distributions of agents from different groups; (5) Favorable outcomes: the number of positive outcomes available at each timestep; (6) Global difficulty: how easy (or difficult) it is for agents to act on recourse; (7) Adaptation type: binary or continuous; (8) Effort type: constant or flexible; (9) Timesteps: the number of timesteps the simulation will run; (10) Bias mitigation strategy: see Section 2.3; (11) Random state (12) Configure ranker: define β_0, β_1 for a linear scoring function $f(x_t) = x_t^0\beta_0 + x_t^1\beta_1$

will describe a real-world scenario, where individuals are applying for a bank loan. We will begin by demonstrating a *highly competitive* scenario—*e.g.*, one with 100 initial agents, 10 favorable outcomes, and 20 new agents entering the environment at each step—and show how damaging this amount of competition is to recourse reliability over time. We will then illustrate how increasing the number of favorable outcomes, or lowering the global difficulty of recourse, results in more reliable recourse. Next, we will gather input from the audience on *how they think agents would behave*, and show how those different choices result in different simulation outcomes and output datasets.

Scenario 2: Exploring fairness in recourse. To demonstrate how The Game Of Recourse can be used to explore fairness in recourse, we will consider two bias settings: *(i)* bias without parity and *(ii)* bias with some parity. We will also demonstrate two qualification settings (*i.e.*, the amount of bias), and will mention how biased decision-making with some parity *is often found in the real world.* Importantly, we will demonstrate that even when the decision-making system itself is "fair," it can still result in unfair recourse. Again, we will take audience input as to how parameters should be modified. After running several simulations, we will compare how metrics like the effort-to-recourse ratio and the time-to-recourse difference change. We will show the audience how this data can be downloaded and used for their own projects.

Scenario 3: Mitigating unfairness in recourse. Naturally, after exploring unfairness in recourse, we will demonstrate how different bias mitigation methods (see Section 2.3) can be used to mitigate unfairness on simulated data. We will demonstrate the effect of three state-of-the-art bias mitigation methods, namely. CNS [3], CDA [4] and GRR [6], and will highlight the strong points and the weak points of these methods. For example, CNS is highly effective at mitigating bias on time-to-recourse, while CDA's strength is reducing effort-to-recourse disparities. We will demonstrate how, under some settings, GRR can cause agents to behave irrationally by attempting to *decrease* their feature values.

ACKNOWLEDGMENTS

This work was supported in part by NSF awards 1916505, 1922658, 2312930, 2326193, and DGE-2234660.

REFERENCES

[1] Patrick Altmeyer, Giovan Angela, Aleksander Buszydlik, Karol Dobiczek, Arie van Deursen, and Cynthia C. S. Liem. 2023. Endogenous Macrodynamics in Algorithmic Recourse. *CoRR* abs/2308.08187 (2023). https://doi.org/10.48550/ARXIV.2308.08187 arXiv:2308.08187

[2] Kasun Amarasinghe, Kit T Rodolfa, Hemank Lamba, and Rayid Ghani. 2023. Explainable machine learning for public policy: Use cases, gaps, and research directions. *Data & Policy* 5 (2023), e5.

[3] Andrew Bell, Joao Fonseca, Carlo Abrate, Francesco Bonchi, and Julia Stoyanovich. 2024. Fairness in Algorithmic Recourse Through the Lens of Substantive Equality of Opportunity. (2024). arXiv:2401.16088 [cs.LG]

[4] Andrea Ferrario and Michele Loi. 2022. The Robustness of Counterfactual Explanations Over Time. *IEEE Access* 10 (2022), 82736–82750. https://doi.org/10.1109/ACCESS.2022.3196917

[5] João Fonseca, Andrew Bell, Carlo Abrate, Francesco Bonchi, and Julia Stoyanovich. 2023. Setting the Right Expectations: Algorithmic Recourse Over Time. In *Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization, EAAMO 2023.* ACM. https://doi.org/10.1145/3617694.3623251

[6] Vivek Gupta, Pegah Nokhiz, Chitradeep Dutta Roy, and Suresh Venkatasubramanian. 2019. Equalizing Recourse across Groups. *CoRR* abs/1909.03166 (2019). arXiv:1909.03166 http://arxiv.org/abs/1909.03166

[7] Julius von Kügelgen, Amir-Hossein Karimi, Umang Bhatt, Isabel Valera, Adrian Weller, and Bernhard Schölkopf. 2022. On the Fairness of Causal Algorithmic Recourse. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022.* AAAI Press, 9584–9594. https://doi.org/10.1609/AAAI.V36I9.21192

[8] Ke Yang and Julia Stoyanovich. 2017. Measuring Fairness in Ranked Outputs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management, Chicago, IL, USA, June 27-29, 2017.* ACM, 22:1–22:6. https://doi.org/10.1145/3085504.3085526