Closing the Loop: Feedback Loops and Biases in Automated Decision-Making

Anonymous Authors

Abstract

Prediction-based decision-making systems are increasingly used in various domains, but they are vulnerable to feedback loops that exacerbate existing biases. The final decision of machine learning (ML)-based systems often feeds back into the system, and the existence of these feedback loops renders the deployment of short-term bias mitigation techniques insufficient to overcome their detrimental effects in the long run. A more rigorous examination of feedback loops and the biases they affect is necessary to design efficient bias mitigation techniques. We use dynamical systems theory to analyze the ML-based decision-making pipeline, classify feedback loops, and show which specific types of ML biases are affected by each type of feedback loop. We encourage readers to consult the more complete manuscript [1].

Keywords

feedback loops, bias, machine learning, dynamical systems theory, sequential decision-making

Motivation Automated decision-making processes that use machine learning algorithms have become widespread, but researchers have found that these systems often perpetuate or even introduce biases. Efforts have been made to mitigate these biases using fairness criteria. However, these solutions are designed for stationary systems [2, 3]. They are often not effective in the long term [4, 5] due to the feedback loops created by the decision-making process [2, 6–24]. To design effective long-term bias mitigation techniques, an interdisciplinary approach is needed to understand the role of feedback loops in perpetuating and amplifying biases.

Contributions We rigorously analyze the ML-based decision-making pipeline and establish a classification of distinct types of feedback loops. We represent the typical ML-based decision-making pipeline as a block diagram (as is usual in dynamical systems theory), which is composed of different sub-systems: the individuals' sampling process s, the individual i's unobservable characteristics representing the decision-relevant construct θ, the observed features x and outcomes y, the ML model f (producing a prediction $\hat{y}$ for i), and the final decision d. The final decision can feed back into any of the other sub-systems, thus forming different types of feedback loops (see Fig. 1): A **sampling feedback loop** comprises the effects of the decision on the probability certain types of individuals enter the decision-making pipeline (e.g., apply for a loan). An **individual feedback loop** is present if the decision acts directly on the individual's characteristics. In contrast to the individual feedback loop, in a **feature feedback loop** the decision affects the *observable* characteristics of the individual (e.g., the credit score) rather than the actual ones (likelihood of repaying a loan). In an **ML model feedback loop**, the decision affects the ML model by modifying the training data set that will be used for future predictions (the outcome is realized and added to the training data set only for positive decisions). Finally, in

EWAF'23: European Workshop on Algorithmic Fairness, June 07–09, 2023, Winterthur, Switzerland

CEUR Workshop Proceedings (CEUR-WS.org)

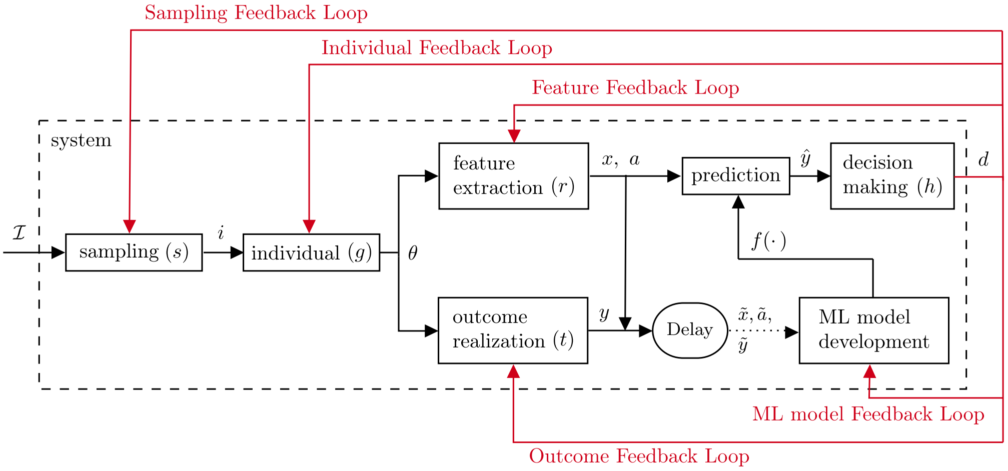


Figure 1: The ML-based decision-making pipeline as a closed-loop system in which different feedback loops can emerge. The pipeline is described in more detail in Appendix A along with the notation used.

an **outcome feedback loop**, the decision affects the outcome before it is realized and ultimately observed (e.g., a loan given at a higher interest rate increases the probability of defaulting). To validate this terminology, we reviewed and classified 24 recent relevant papers (see Table 1) – where some feedback loops can be classified as **adversarial** whenever the decision feeds back into the system involving some strategic action of the affected individual(s).

Furthermore, we associate the different types of feedback loops with the biases they affect (see Table 1). Sampling and ML model feedback loops can change the representation of the training or evaluation sample dataset compared to the target population, thus leading to representation bias. An individual feedback loop can cause life bias by changing an individual's decision-relevant (though, often unobservable) attributes. In contrast, feature and outcome feedback loops act on the extraction and realization of those attributes, which can affect the measurement bias of the observable attributes. In general, we find that the existence of feedback loops in the ML-based decision-making pipeline can perpetuate, reinforce, or even reduce ML biases.

Table 1
Feedback loops in the algorithmic fairness literature and their relation to biases.

Feedback loop	**Literature**		**ML biases**
	non-adversarial	*adversarial*	
Sampling Feedback Loop	[15, 16, 25]	–	Representation bias
Individual Feedback Loop	[26, 27]	[17, 21, 22, 28–30]	Life bias
Feature Feedback Loop	[4, 5, 11, 16, 17, 31, 32]	[9, 17, 18, 21, 28, 29, 33, 34]	Measurement bias
ML Model Feedback Loop	[6, 19, 20, 32, 35]	–	Representation bias
Outcome Feedback Loop	[18]	–	Measurement bias

Potential impact By rigorously analyzing the ML pipeline, we believe that our framework is a necessary first step toward understanding the exact role of the feedback loops in it. Providing a rigorous classification of feedback loops will enable a deeper understanding of the existing works in the ML literature and it will allow putting their results into the perspective of their assumptions (e.g., which types of feedback loops are considered and which are not). We believe that our framework will be helpful for the purposeful design of feedback loops [7, 13, 36], and for the development of long-term bias and unfairness mitigation techniques [37–39].

References

[1] Anonymous, A classification of feedback loops and their relation to biases in automated decision-making systems (2023). Submitted at the end of this PDF.

[2] A. Chouldechova, A. Roth, The Frontiers of Fairness in Machine Learning (2018) 1–13. URL: http://arxiv.org/abs/1810.08810.

[3] S. Mitchell, E. Potash, S. Barocas, A. D'Amour, K. Lum, Algorithmic Fairness: Choices, Assumptions, and Definitions, Annual Review of Statistics and Its Application 8 (2021) 141–163. URL: https://www.annualreviews.org/doi/10.1146/annurev-statistics-042720-125902. doi:10.1146/annurev-statistics-042720-125902.

[4] L. T. Liu, S. Dean, E. Rolf, M. Simchowitz, M. Hardt, Delayed Impact of Fair Machine Learning, in: J. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning, volume 80 of *Proceedings of Machine Learning Research*, PMLR, 2018, pp. 3150–3158. URL: https://proceedings.mlr.press/v80/liu18c.html.

[5] Y. Sun, A. Cuesta-Infante, K. Veeramachaneni, The Backfire Effects of Fairness Constraints, ICML 2022 Workshop on Responsible Decision Making in Dynamic Environments (2022). URL: https://responsibledecisionmaking.github.io/assets/pdf/papers/44.pdf.

[6] D. Ensign, S. A. Friedler, S. Neville, C. Scheidegger, S. Venkatasubramanian, C. Wilson, Runaway Feedback Loops in Predictive Policing, in: Proceedings of Machine Learning Research, volume 81, 2018, pp. 1–12. URL: https://github.com/algofairness/.

[7] S. Barocas, M. Hardt, A. Narayanan, Fairness and Machine Learning, fairmlbook.org, 2019. URL: http://www.fairmlbook.org.

[8] Y. Hu, L. Zhang, Achieving Long-Term Fairness in Sequential Decision Making, Proceedings of the AAAI Conference on Artificial Intelligence 36 (2022) 9549–9557. URL: https://ojs.aaai.org/index.php/AAAI/article/view/21188. doi:10.1609/aaai.v36i9.21188.

[9] L. Hu, N. Immorlica, J. W. Vaughan, The Disparate Effects of Strategic Manipulation, in: Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 259–268. URL: https://doi.org/10.1145/3287560.3287597. doi:10.1145/3287560.3287597.

[10] A. Chouldechova, A. Roth, A Snapshot of the Frontiers of Fairness in Machine Learning, Commun. ACM 63 (2020) 82–89. URL: https://doi.org/10.1145/3376898. doi:10.1145/3376898.

[11] Y. Sun, Algorithmic Fairness in Sequential Decision Making, Ph.D. thesis, 2022.

[12] S. Dean, J. Morgenstern, Preference Dynamics Under Personalized Recommendations (2022) 1–25. URL: http://arxiv.org/abs/2205.13026.

[13] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, A. Galstyan, A Survey on Bias and Fairness in Machine Learning, ACM Comput. Surv. 54 (2021). URL: https://doi.org/10.1145/3457607. doi:10.1145/3457607.

[14] X. Zhang, M. Liu, Fairness in Learning-Based Sequential Decision Algorithms: A Survey, Studies in Systems, Decision and Control 325 (2021) 525–555. doi:10.1007/978-3-030-60990-0{_}18.

[15] X. Zhang, M. M. Khalili, C. Tekin, M. Liu, Group retention when using machine learning in sequential decision making: The interplay between user dynamics and fairness, Advances in Neural Information Processing Systems 32 (2019).

[16] X. Zhang, M. M. Khalili, M. Liu, Long-Term Impacts of Fair Machine Learning, Ergonomics in Design 28 (2020) 7–11. doi:10.1177/1064804619884160.

[17] A. D'Amour, H. Srinivasan, J. Atwood, P. Baljekar, D. Sculley, Y. Halpern, Fairness is not static: Deeper understanding of long term fairness via simulation studies, FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (2020) 525–534. doi:10.1145/3351095.3372878.

[18] J. C. Perdomo, T. Zrnic, C. Mendler-Dunner, M. Hardt, Performative prediction, 37th International Conference on Machine Learning, ICML 2020 PartF16814 (2020) 7555–7565.

[19] D. Ensign, S. A. Friedler, S. Neville, C. Scheidegger, S. Venkatasubramanian, M. Mohri, K. Sridharan, Decision making with limited feedback: Error bounds for predictive policing and recidivism prediction, Proceedings of Machine Learning Research 83 (2018) 1–9.

[20] Y. Bechavod, K. Ligett, A. Roth, B. Waggoner, Z. S. Wu, Equal opportunity in online classification with partial feedback, Advances in Neural Information Processing Systems 32 (2019).

[21] L. T. Liu, A. T. Kalai, A. Wilson, C. Borgs, N. Haghtalab, J. Chayes, The disparate equilibria of algorithmic decision making when individuals invest rationally, FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (2020) 381–391. doi:10.1145/3351095.3372861.

[22] L. Hu, Y. Chen, A short-term intervention for long-term fairness in the labor market, The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018 2 (2018) 1389–1398. doi:10.1145/3178876.3186044.

[23] L. Reader, P. Nokhiz, C. Power, N. Patwari, S. Venkatasubramanian, S. Friedler, Models for Understanding and Quantifying Feedback in Societal Systems, in: 2022 ACM Conference on Fairness, Accountability, and Transparency, FAccT '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 1765–1775. URL: https://doi.org/10.1145/3531146.3533230. doi:10.1145/3531146.3533230.

[24] C. O'neil, Weapons of math destruction: How big data increases inequality and threatens democracy, Crown, 2017.

[25] T. Hashimoto, M. Srivastava, H. Namkoong, P. Liang, Fairness Without Demographics in Repeated Loss Minimization, in: J. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning, volume 80 of *Proceedings of Machine Learning Research*, PMLR, 2018, pp. 1929–1938. URL: https://proceedings.mlr.press/v80/hashimoto18a.html.

[26] W. S. Rossi, J. W. Polderman, P. Frasca, The closed loop between opinion formation and personalised recommendations, IEEE Transactions on Control of Network Systems (2021) 1. doi:10.1109/TCNS.2021.3105616.

[27] N. Perra, L. E. C. Rocha, Modelling opinion dynamics in the age of algorithmic personalisation, Scientific reports 9 (2019) 1–11.

[28] H. Heidari, V. Nanda, K. P. Gummadi, On the Long-term Impact of Algorithmic Decision Policies: Effort unfairness and feature segregation through social learning, 36th International Conference on Machine Learning, ICML 2019 2019-June (2019) 4787–4796.

[29] J. Kleinberg, M. Raghavan, How Do Classifiers Induce Agents to Invest Effort Strategically?, ACM Transactions on Economics and Computation 8 (2020). doi:10.1145/3417742.

[30] X. Zhang, R. Tu, Y. Liu, M. Liu, H. Kjellström, K. Zhang, C. Zhang, How do fair decisions

fare in long-term qualification?, Advances in Neural Information Processing Systems 2020-Decem (2020) 1–13.

[31] A. J. B. Chaney, B. M. Stewart, B. E. Engelhardt, How algorithmic confounding in recommendation systems increases homogeneity and decreases utility (2018) 224–232. doi:10.1145/3240323.3240370.

[32] A. Sinha, D. F. Gleich, K. Ramani, Deconvolving Feedback Loops in Recommender Systems, in: D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 29, Curran Associates, Inc., 2016. URL: https://proceedings.neurips.cc/paper/2016/file/962e56a8a0b0420d87272a682bfd1e53-Paper.pdf.

[33] S. Tsirtsis, B. Tabibian, M. Khajehnejad, A. Singla, B. Schölkopf, M. Gomez-Rodriguez, Optimal Decision Making Under Strategic Behavior (2019). URL: http://arxiv.org/abs/1905.09239.

[34] S. Milli, J. Miller, A. D. Dragan, M. Hardt, The social cost of strategic classification, FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency (2019) 230–239. doi:10.1145/3287560.3287576.

[35] H. Elzayn, M. Kearns, S. Jabbari, S. Neel, Z. Schutzman, C. Jung, A. Roth, Fair algorithms for learning in allocation problems, FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency (2019) 170–179. doi:10.1145/3287560.3287571.

[36] M. Kearns, A. Roth, The Ethical Algorithm: The Science of Socially Aware Algorithm Design, Oxford University Press, Inc., USA, 2019.

[37] M. Hardt, E. Price, N. Srebro, Equality of opportunity in supervised learning, in: Advances in Neural Information Processing Systems, NIPS'16, Curran Associates Inc., Red Hook, NY, USA, 2016, pp. 3323–3331.

[38] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, A. Huq, Algorithmic Decision Making and the Cost of Fairness, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 797–806. URL: https://doi.org/10.1145/3097983.3098095. doi:10.1145/3097983.3098095.

[39] J. Baumann, A. Hannák, C. Heitz, Enforcing Group Fairness in Algorithmic Decision Making: Utility Maximization Under Sufficiency, in: Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, FAccT '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 2315–2326. URL: https://doi.org/10.1145/3531146.3534645. doi:https://doi.org/10.1145/3531146.3534645.

[40] C. Hertweck, C. Heitz, M. Loi, On the Moral Justification of Statistical Parity, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 747–757. URL: https://doi.org/10.1145/3442188.3445936. doi:10.1145/3442188.3445936.

[41] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, On the (im)possibility of fairness, 2016. URL: https://arxiv.org/abs/1609.07236.

[42] D. Pessach, E. Shmueli, A Review on Fairness in Machine Learning, ACM Comput. Surv. 55 (2022). URL: https://doi.org/10.1145/3494672. doi:10.1145/3494672.

[43] S. Caton, C. Haas, Fairness in Machine Learning: A Survey (2020). URL: http://arxiv.org/abs/2010.04053.

A. Notation for the ML-based decision-making pipeline in Fig. 1

At the beginning of the pipeline, an individual i is sampled from the world (i.e., the environment) $\mathcal{I}$, which represents a signal entering in the sampling function block $s : \mathcal{I} \to i$. Let i be the individual's identity – i.e., its index in the population, which [40] call *potential space* (PS) – and let $g : i \to \theta$ be a function that returns the individual's attributes. More precisely, θ denotes the construct that is relevant for the prediction – what [41] call *construct space* (CS). The features x, extracted through the function $r : \theta \to x, a$, and the outcome y (also called label or target), realized through the function $t : \theta \to y$, are imperfect proxies that can be measured. For instance, y can represent whether or not an individual repays a granted loan and x is a set of features (for example, the credit score, as widely used in the US) that are used by the decision-maker to predict the repayment probability $\hat{y}$ in order to decide whether to grant the loan or not. For each sampled individual, the final decision d is informed by the prediction $\hat{y}$, which is produced based on the observed features x to approximate y using a learned function $f : x \to \hat{y}$. Once the outcome is observed, i.e., after one time-unit of delay, the past time's feature label pair $(\tilde{x}, \tilde{y})$ can end up as a sample in the dataset (X, Y) that is used to (re)train and (re)evaluate an ML model. In fully-automated decision-making systems, the decision rule h is solely based on the prediction ($h : \hat{y} \to d$), usually taking the form of a simple threshold rule, e.g., $d = 1$ if and only if $\hat{y} \geq \bar{y}$. The symbol a indicates the sensitive attribute of the individual (e.g., race or gender) and can possibly also be incorporated in the features x. More precisely, the training, evaluation, prediction, or decision-making can use the information on the individual group memberships.Notice that d does not always directly follow from $\hat{y}$. Efforts to ensure group fairness usually take the group membership a into account, e.g., to avoid disparate impact [42, 43]. Similarly, in non-automated decision-making systems, human decision-makers might consider any external, environmental information z, resulting in a more complex decision rule $h : f, x, a, \hat{y}, z \to d$.

B. Full paper and code

Next, we provide the full paper in an anonymized form for reference [1]. We will link to the non-anonymized online reference of the full paper once this short paper has been accepted.

The code used to run simulation experiments will be made publicly available on GitHub after the acceptance of this manuscript.

A Classification of Feedback Loops and Their Relation to Biases in Automated Decision-Making Systems

ANONYMOUS AUTHOR(S)*

Prediction-based decision-making systems are becoming increasingly prevalent in various domains. Previous studies have demonstrated that such systems are vulnerable to runaway feedback loops, e.g., when police are repeatedly sent back to the same neighborhoods regardless of the actual rate of criminal activity, which exacerbate existing biases. In practice, the outcome of ML-based decision-making systems (i.e., the final decision) feeds back into the system, and the existence of these feedback loops often renders the deployment of short-term bias mitigation techniques insufficient to overcome their detrimental effects in the long run. Thus, it is necessary to first undertake a more rigorous examination of feedback loops and the biases they affect, and only then it will be possible to design efficient bias mitigation techniques. In this paper, we use the language of dynamical systems theory, a branch of applied mathematics that deals with the analysis of dynamical engineering systems, to rigorously analyze the ML-based decision-making pipeline and to establish a vocabulary for the different types of feedback loops. We classify feedback loops into distinct types based on which component of the ML pipeline they affect and whether they are a consequence of some strategic action of the affected individual(s). By reviewing existing scholarly work, we show that this classification covers many examples discussed in the algorithmic fairness community, thereby providing a unifying and principled framework to study feedback loops. By qualitative analysis, and through a simulation example of recommender systems, we show which specific types of ML biases are affected by each type of feedback loop. We find that the existence of feedback loops in the ML-based decision-making pipeline can perpetuate, reinforce, or even reduce ML biases.

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

Additional Key Words and Phrases: feedback loops, bias, machine learning, dynamical systems theory, sequential decision-making

ACM Reference Format:
Anonymous Author(s). 2023. A Classification of Feedback Loops and Their Relation to Biases in Automated Decision-Making Systems. In *2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT '23), June 12–15, 2023, Chicago, US.* ACM, New York, NY, USA, 19 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Many of today's automated processes rely on machine learning (ML) algorithms to inform decisions that have a profound impact on people's lives. For instance, they are employed to evaluate whether an individual should be admitted to a certain college [37], be granted a loan [19], or treated as high risk of recidivism [5]. The advantage of these ML-based decision-making systems is their scalability, i.e., the capability to handle a vast number of decisions in an efficient manner. However, researchers have found evidence that these algorithms often exacerbate existing biases that underlie human decisions [13, 21, 36] and even introduce new ones [1, 6, 11, 56].

To solve this problem, a recent line of research in algorithmic fairness started investigating solutions that can mitigate these biases at different stages of the ML pipeline by enforcing some metrics of individual or group fairness [7, 43].

Manuscript submitted to ACM

Although these attempts prove to be successful in the short term, they often do not perform equally well in the long term, i.e., after multiple rounds of the decision-making process [40, 59].[1] The underlying reason seems to lie in the fact that the mitigating solutions are designed for stationary systems [9, 45], while the system itself dynamically evolves over time. More specifically, the system changes over time because the output (the decision) feeds back as input to the system itself, thus creating what researchers refer to as a "feedback loop" [3, 4, 9, 10, 12, 14, 16, 17, 28–30, 41, 43, 49, 50, 54, 58, 67–69]. The result is that biases are perpetuated (or even reinforced) due to the existence of the feedback loop, despite enforcing the mitigation techniques. Thus, it is crucial to first understand the role of the feedback loops, and how they relate to the amplification of different types of bias. This comprehension will lay the necessary foundation for analyzing the dynamics of automated decision-making systems and pave the way for the design of long-term bias mitigation techniques in the future.

This paper is the first attempt to fill this gap by providing a formal definition and a rigorous classification of feedback loops in the ML-based decision-making pipeline, and by linking them to the biases they affect. To do so, we first clarify the difference between open-loop and closed-loop (or feedback-loop) systems by borrowing the language and the tools from dynamical systems theory, the discipline that focuses on the analysis of dynamical systems in engineered processes. Then, we apply this system-theoretic framework to the decision-making pipeline, which is composed of different sub-systems: the individuals' sampling process, the individuals' characteristics representing the decision-relevant construct, the observed features and outcomes, the ML model, and the final decision. The final decision can feed back into any of the other sub-systems, thus forming the different types of feedback loops. This, in turn, means that the effect on the whole pipeline and the amplification of biases depends on the type of feedback loop.

The first contribution of this paper (see Sec. 2) is to cast the ML-based decision-making pipeline into a system-theoretic framework. Our second contribution (see Sec. 3) consists in providing a classification of the different types of feedback loops, which we call *sampling*, *individual*, *feature*, *outcome*, and *ML model feedback loop* depending on which sub-system is affected. Additionally, we introduce the notion of "adversarial feedback loops," which represent special cases of feedback loops in which the final decision feeds back into the system as a consequence of some strategic action of the affected individual(s). We identified relevant ML literature that discuss feedback loops in order to make sure that our framework is exhaustive with respect to the recent advances in the field. As a third contribution (see Sec. 4), we provide an overview of the different types of bias that can be reduced, perpetuated, or amplified by each of the five feedback loops we introduce. As a fourth and final contribution (see Sec. 5), we illustrate the impact of the different types of feedback loops on ML biases by means of a unifying example of recommendation systems.

2 THE ML-BASED DECISION-MAKING PIPELINE THROUGH THE LENS OF DYNAMICAL SYSTEMS THEORY

Dynamical systems theory provides helpful language and tools, which we will borrow in this paper, accounting for the fact that ML-based decision-making systems are usually not static but evolve over time. A *dynamical system* is a process that relates a set of *input signals* to a set of *output signals*. A *signal* is a variable or quantity of interest that may vary over time. Thus, an algorithm is an example of a dynamical system that receives observable features as input signals and produces predictions or decisions as output signals. Dynamical systems theory is concerned with the mathematical modeling of dynamical systems with the objective of understanding and/or manipulating fundamental properties, such as whether the system reaches a predictable operating point or exhibits oscillatory behaviors.

[1] More notably, enforcing fairness constraints often leads to counter-intuitive and undesired results, increasing the gap between advantaged and disadvantaged groups, thus again exacerbating existing initial biases [12].

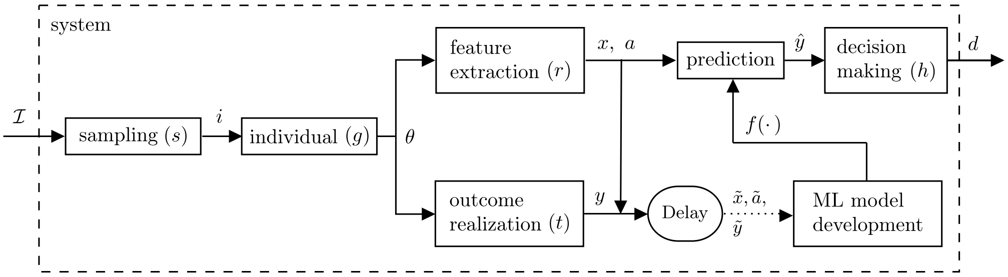


Fig. 1. The ML-based decision-making pipeline as an open-loop system.

It is common to represent dynamical systems in block diagrams, where blocks denote systems and arrows denote signals, as a way to provide a high-level graphical representation of a real-world system. Block diagrams are particularly useful to understand and study the *interconnection* of different (sub-)systems, which are composed to form larger systems. A *series interconnection* occurs when the output of a system (or algorithm) is the input for another one. A *parallel interconnection* occurs when the same input enters two systems whose outputs are then combined. In a *feedback interconnection*, the output of a system is injected back as an input to one (or more) of its components, creating a *feedback loop*. Series and parallel interconnections lead to *open-loop systems*, whereas feedback interconnections lead to *closed-loop systems* – see Fig. 4 in Appendix B for a visual representation.

The prototypical ML-based decision-making pipeline can also be represented as a block diagram. We start by describing its open-loop components, shown in Fig. 1, before characterizing possible feedback interconnections in Section 3. At the beginning of the pipeline, an individual i is sampled from the world (i.e., the environment) $\mathcal{I}$, which represents a signal entering in the sampling function block $s : \mathcal{I} \longrightarrow i$. Let i be the individual's identity – i.e., its index in the population, which [27] call *potential space* (PS) – and let $g : i \longrightarrow \theta$ be a function that returns the individual's attributes. More precisely, θ denotes the construct that is relevant for the prediction – what [18] call *construct space* (CS). The features x, extracted through the function $r : \theta \longrightarrow x, a$, and the outcome y (also called label or target), realized through the function $t : \theta \longrightarrow y$, are imperfect proxies that can be measured – what [18] call *observed space* (OS). For instance, y can represent whether or not an individual repays a granted loan and x is a set of features (for example, the credit score, as widely used in the US) that are used by the decision-maker to predict the repayment probability $\hat{y}$ in order to decide whether to grant the loan or not. For each sampled individual, the final decision d is informed by the prediction $\hat{y}$, which is produced based on the observed features x to approximate y using a learned function $f : x \longrightarrow \hat{y}$. Once the outcome is observed, i.e., after one time-unit of delay, the past time's feature label pair $(\tilde{x}, \tilde{y})$ can end up as a sample in the dataset (X, Y) that is used to (re)train and (re)evaluate an ML model (more details on the ML model development process are discussed in Appendix C). In fully-automated decision-making systems, the decision rule h is solely based on the prediction ($h : \hat{y} \longrightarrow d$), usually taking the form of a simple threshold rule, e.g., $d = 1$ if and only if $\hat{y} \geq \bar{y}$. The symbol a indicates the sensitive attribute of the individual (e.g., race or gender) and can possibly also be incorporated in the features x. More precisely, the training, evaluation, prediction, or decision-making can use the information on the individual group memberships.[2]

[2] Notice that d does not always directly follow from $\hat{y}$. Efforts to ensure group fairness usually take the group membership a into account, e.g., to avoid disparate impact [7, 52]. Similarly, in non-automated decision-making systems, human decision-makers might consider any external, environmental information z, resulting in a more complex decision rule $h : f, x, a, \hat{y}, z \longrightarrow d$.

3 FEEDBACK LOOPS IN THE ML-BASED DECISION-MAKING PIPELINE

In contrast to ML, in the field of dynamical systems theory, feedback loops are not always seen as an undesirable property of a system. Lots of the emphasis of dynamical systems theory is on relating properties of the open-loop system, i.e., the system without a feedback loop, to those of the closed-loop system, i.e., the system with a feedback loop. In this paper, we leverage the idea of closed-loop system properties to define feedback mechanisms in ML-based decision-making systems. Interestingly, closed-loop systems may exhibit desirable properties compared to their open-loop counterparts.

In this section, we complete the specification of the ML pipeline as a dynamical system by considering the feedback interconnections that could be present. We first define various types of feedback loops depending on the component of the ML pipeline affected by the outcome of the system (i.e., the final decision of the decision-maker). Next, we introduce the concept of *adversarial feedback loops*. Then, we describe how different types of feedback loops can coexist. Finally, we clarify some terminology with respect to positive and negative feedback loops.

3.1 Feedback Loops

In many real-life settings, the decision taken at the end of the ML pipeline may feed back into some of its blocks. Every block in the ML pipeline (except the prediction block, as this usually simply consists of applying f to a new input example x) can be affected by the decision, each forming a different type of feedback loop, as depicted in Fig. 2. In what follows, we classify these feedback loops to provide a vocabulary and some examples. To validate this terminology, we reviewed a total of 24 recent relevant papers that discuss issues of feedback loops in the context of ML-based decision-making systems – many of which particularly focus on fairness aspects. These papers are listed in Table 1 (we describe the literature search process in more detail in Appendix A). We emphasize that the classification of the five feedback loops represented in Fig. 2 is complete with respect to the examples and use cases we identified in the current state of the literature on fair dynamic decision-making systems. Despite covering existing literature, this feedback loop classification can easily be extended to capture more nuanced kinds of feedback.[3]

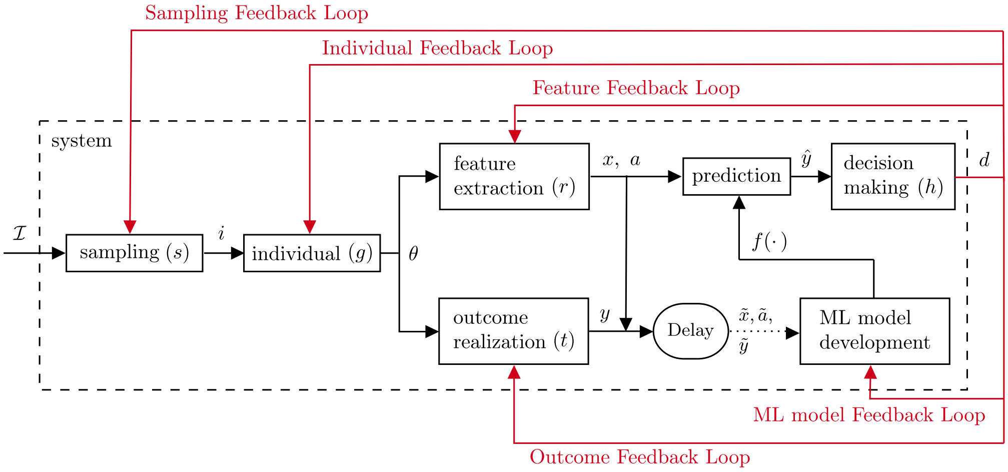


Fig. 2. The ML-based decision making pipeline as a closed-loop system in which different feedback loops can emerge.

[3] Notice that certain blocks of the ML pipeline depicted in Fig. 2 aggregate several processes that could be split up into several blocks, potentially resulting in a more nuanced classification of feedback loops. For example, the block denominated "feature extraction" could be split up into measurement followed

Table 1. Overview of feedback loops in the algorithmic fairness literature

Feedback loop	***non-adversarial***	***adversarial***
Sampling Feedback Loop	[25, 67, 68]	–
Individual Feedback Loop	[51, 55]	[12, 26, 28, 38, 41, 70]
Feature Feedback Loop	[8, 12, 40, 57–59, 67]	[12, 26, 29, 38, 41, 44, 50, 61]
ML Model Feedback Loop	[4, 15–17, 57]	–
Outcome Feedback Loop	[50]	–

3.1.1 Sampling Feedback Loop. The first type of feedback loop we introduce is the one that comprises the effects of the decision on the sampling of the individual from the population. This influences the retention rate of different groups and modifies their representation. Consider the following example of a college admission scenario discussed in [46]. First, let the total population be partitioned into two groups A and B. The population undergoes a selection process in which an institution, the decision-maker, designs a policy that maps each individual to a probability of being selected, possibly depending on the group identity a and on observable attributes x that bear information about qualification, e.g., GPA, SAT, or recommendation letters. According to the authors of [46], the selection process at time t might change the qualification profiles of either group at time $t + 1$ through a self-selection process acting in the form of filtering the pool of individuals available at the next iteration. In other words, with the existence of a *sampling* feedback loop, individuals belonging to a group that had received lower admission rates at the previous iteration might be discouraged from applying as candidates at the next iteration, thus affecting the application rates from the two groups (and ultimately the selection rates). Note that, this feedback loop might lead to one of the two groups disappearing from the candidate pool. To understand this, consider a similar example related to speech recognition products such as Amazon's Alexa and Google Home, which have been shown to have accent bias against non-native speakers [24], with native speakers experiencing much higher quality than non-native speakers. This difference can lead to a sampling feedback loop, where non-native speakers cease to use such products. This may be hard to detect because the speech recognition model, from that point on, only receives input and training data from native speakers, potentially resulting in a model that is even more skewed towards the remaining users, i.e., the native speakers. Without intervention, the model becomes even less accurate for non-native speakers, which reinforces the initial user experience [25]. Additional examples of the sampling feedback loop can also be found in [67, 68].

3.1.2 Individual Feedback Loop. Another possible effect of the decision acts directly on the individual's characteristics θ, i.e., through the function g. An example of this type of feedback loop can be found in the users' reactions to personalized recommendations. As discussed in [51, 55], a user's opinion on, e.g., a certain political issue, is influenced by the news articles received. Therefore, the decision of the recommender system to promote a certain type of content has the effect of shifting the opinion of the individuals that receive such a recommendation. Additional examples of the individual feedback loop are discussed in the context of adversarial feedback loops (see Sec. 3.2).

3.1.3 Feature Feedback Loop. The third type of feedback loop is relatively close to the previous one. However, in contrast to the individual feedback loop, the decision has an effect on the *observable* characteristics of the individual rather than on the actual ones, i.e., on x rather than θ. One of the most common examples of this feature feedback loop can be found in credit lending scenarios in which a lender decides whether or not to approve a loan application based

by feature engineering, creating two subcategories of the feature feedback loop. However, existing works modeling feature feedback loops (e.g., [40, 44]) consider the effect of the decision on the distribution of x, a, which forms the input for the prediction model, without differentiating between the effects of the decision on the measurement and the engineered features. Similarly, we consider the entire ML model development process as one block with (one or more) new feature label pairs as input and a learned prediction function as output (see Appendix C for more details).

on the applicant's credit score, which is interpreted as a measurable and observable proxy for the individual's capability of paying back a granted loan [40]. For any positive decision, we observe a feature feedback loop: if the loan is repaid, the credit score increases; otherwise, if the applicant defaults, the credit score decreases. Note that, in this example, the feedback loop takes place only if the decision is positive, and it also requires information on the actual outcome y. However, none of these conditions is strictly necessary for a feature feedback loop to occur.[4]

Another example is constituted by content recommender systems where the time a user looks at some content is part of the observation captured in the feature x [8, 57]. However, the time explicitly depends on what the recommender system has previously suggested, thus closing a feature feedback loop. This happens irrespective of whether this recommendation affects the individual's interests, i.e., even in the absence of an individual feedback loop.

Additional examples of the feature feedback loop can also be found in [12, 58, 59, 67]. Furthermore, similarly to the individual feedback loop, also for the feature feedback loop, there exists an adversarial counterpart (see Sec. 3.2).

3.1.4 ML Model Feedback Loop. While the previous types of feedback loops could apply to any (human or automated) decision-making system, in the ML model feedback loop, the final decision d affects the ML model by modifying the training or the validation data sets (X, Y) that will be used for future predictions. Typical examples in this category are known as ML-based decision-making with *limited* [16] or *partial feedback* [4] and the reason is that ML models are retrained using newly available data. ML model feedback loops describe the case when the data that becomes newly available over time depends on the decision taken. For example, hiring algorithms only receive feedback on people who were hired, credit lending algorithms only receive feedback on people who received the loan, and predictive policing algorithms only register crime in patrolled neighborhoods. In all these scenarios, the decision will create a gate to the pair (x, y), which will be added to the existing data set (X, Y) only when the decision is positive ($d = 1$). Notice that, when the retraining of the model does not depend on the decision (i.e., if the feature label pair x, y is added to the existing data set independently of d), there is no ML model feedback loop. Using the language of dynamical systems theory, this case is simply viewed as an open-loop system with memory where the state variable (X, Y) evolves according to the inner dynamics, but independently of the output variable (the decision d). Additional examples of the ML model feedback loop can also be found in [15, 17, 57].

3.1.5 Outcome Feedback Loop. Finally, in the outcome feedback loop, the decision (d) affects the outcome (y) before it is realized and ultimately observed. Notice that this observed outcome then needs to be reused in some form in order to close the loop. Namely, it only forms a loop if the outcome is used, e.g., as part of the training or validation data when retraining the model[5]. To see how an outcome feedback loop can arise, consider again the credit lending scenario: if a person is predicted at high risk of default, the loan might be granted, but at a higher interest rate. However, the decision to enforce a higher interest rate further increases the chances that the customer defaults [50]. In contrast to the example provided in Section 3.1.3, here we assume that the lender's decision d has an effect on the realization of the outcome y, i.e., whether the loan is paid back or not.

3.2 Adversarial Feedback Loops

Some of the previously described feedback loops can take the form of what we call *adversarial feedback loops*. This describes any feedback loop that depends on the decision d intertwined with an adversarial reaction to it. In practice,

[4] Suppose granting a loan is already enough to increase an individual's credit score. In this case, the outcome of the lender's decision is fed back into x, creating a feature feedback loop irrespective of the realized outcome y.

[5] Notice that this can, but does not need to, happen through an ML model feedback loop.

these are scenarios in which the individuals subjected to the decision-making process take strategic actions that increase their chances of receiving favorable decisions. For instance, consider the attention allocation problem discussed in [12]. Here, the decision-maker has limited (insufficient) resources to exhaustively inspect N different locations, and therefore they have to decide where to (dynamically) allocate the attention. As the authors argue, the incident rate of each of the N sites responds dynamically (and adversarially) to the previous allocation, i.e., it increases where there was absolutely no control, and vice-versa it decreases proportionally to the amount of inspection. In essence, this example describes the case of an *adversarial individual feedback loop*, because the decision ultimately affects the incident rate, i.e., θ.

To give another example, consider a college that publishes the decision rule for its admission policy. Prospective students can strategically invest in their own qualifications in order to meet the requirements. If this action truly changes the preparation level of the student [41], then it is again an *adversarial individual feedback loop*. However, it is also possible that only the observable features of the individual are changed [26], e.g., if the students invest in SAT exam preparation without changing their actual qualification for the college. Then, we are facing an *adversarial feature feedback loop*. Similarly, if an individual is applying for a loan, it might be beneficial to open multiple credit lines to improve their observable features [50]. This action is not truly modifying the individual's capability of paying back the loan, but it is only a way to game the decision-making policy, thus we have an *adversarial feature feedback loop*.

Additional examples of adversarial individual and feature feedback loops can be found in [26, 28, 38, 70] and [12, 29, 38, 44, 61], respectively. However, we emphasize that it is not always easy to distinguish between the individual and the feature adversarial feedback loops, because many of these works assume that the decision affects the qualification θ of the individuals, but oftentimes they intend that it only affects its observable features x.

3.3 Coexistence of Feedback Loops

As seen in the previous sections, different feedback loops can coexist within the same application domain. For instance, the recommender systems for an online platform can affect the opinion of the users θ (individual feedback loop) or just their representation in the feature space x (feature feedback loop). College admission policies can induce students to improve their qualification (adversarial individual feedback loop) or just their representation x (adversarial feature feedback loop). Alternatively, they can also lead to different retention rates across groups (sampling feedback loop). Lending decisions can affect an individual's creditworthiness θ (individual feedback loop), credit score x (feature feedback loop), realized outcome y (i.e., whether or not the granted loan is paid back, representing an outcome feedback loop), or even the data used for the ML model development (X, Y) (resulting in an ML model feedback loop) or the sample of individuals applying for a loan in the first place (causing a sampling feedback loop). All five classified feedback loops represent some causal effect of the final decision on another component of the ML-based decision-making pipeline. Thus, which type(s) of feedback loop(s) (co)exists solely depends on the context-specific assumptions regarding the underlying causal effects of the decision. The possibility of the coexistence of different combinations of feedback loops gives rise to coupled behavior and even more complex dynamics.

3.4 Positive/Negative Feedback Loops and Relation to Stability

In many disciplines, including the ML community, a considerable emphasis is placed on classifying feedback loops as either *positive* or *negative* [32, 47, 53]. This is often accompanied by some ambiguity in the definition of these notions. In systems theory, a positive feedback loop (also known as *reinforcing*) amplifies the effect of inputs on the outputs, while a negative feedback loop (also known as *balancing*) attenuates it. In other domains, the notion of a positive/negative feedback loop is sometimes associated with desirable/undesirable outcomes, regardless of how it acts to amplify/attenuate inputs. For example, the feedback loop that increases recidivism due to incarcerated individuals' reduced access to finance is referred to as a negative feedback loop in [69, p. 2]. This ambiguity is problematic, especially considering that in systems theory the desired goal is often to make the output a predictable function of the input and independent from other exogenous but inevitable inputs (considered as *disturbances*). For this reason, properly designed negative feedback loops are deemed preferable, while positive feedback loops are often considered problematic.

However, systems theory often places more emphasis on the *stability* of the closed-loop system rather than classifying feedback loops as positive or negative. A stable system converges to a predictable equilibrium point, while an unstable system either oscillates or grows beyond bounds. It is intuitive to associate positive feedback with instability and negative feedback with stability, however, this intuition is not universal [2, 66]. On the one hand, positive feedback is guaranteed to lead to instability only in the special class of linear systems. The presence of non-linearity (e.g., saturation or hysteresis) can stabilize a positive feedback loop, which is intentionally introduced in many cases (e.g., the design of signal amplifiers). On the other hand, negative feedback does not guarantee stability (even in linear systems). Moreover, the same system could be in either positive or negative feedback depending on the operating regime (e.g., the frequency of the input signal). Thus, in this paper, we shift the focus from classifying feedback loops as positive/negative to asking whether the closed-loop system converges (or not) to a (desirable) state. As we will see in the examples in Section 5, feedback loops often drive the ML-based decision system to stable equilibrium points in the long run.

4 FEEDBACK LOOPS AND ALGORITHMIC BIASES

Being able to reason about what caused certain types of bias is of incredible practical importance in order to avoid or counteract them in the long term. Otherwise, ML-based decision-making systems can result in socially undesirable outcomes over time. Many works claim that those biases can be perpetuated or even reinforced due to feedback loops [3, 9, 10, 35, 42, 43, 48, 49, 62]. However, a clear understanding of the causal effects of feedback loops on algorithmic biases is currently missing. We fill this gap by connecting the classification of feedback loops (which we introduced in Section 3.1) to algorithmic biases and explain in more detail *which* types of bias they affect. Table 2 provides a general overview.

Table 2. Feedback loops and the ML biases they affect

Feedback loop	**ML bias**
Sampling, ML model	Representation bias
Individual	Life bias
Feature, Outcome	Measurement bias

Representation Bias. According to [60], there are different nuances of representation bias[6]: Representation bias can arise (i) if the defined target population does not reflect the use population, (ii) if the target population contains underrepresented groups, and (iii) if the sampled group of individuals is not representative of the target population. All three versions represent some difference between the used dataset (X, Y) and the population $\mathcal{I}$.

Sampling feedback loops can affect representation bias. Sampling feedback loops affect the sampling function s that outputs a set of individuals on which an ML-based decision-making system acts. A sampling feedback loop changes the sample of individuals for whom a prediction and, ultimately, a decision is made (i.e., those who get a chance to be selected). Thus, it can result in representation bias, which describes the situation in which s undersamples some part of

[6] Representation bias is sometimes called sampling bias, population bias, sample selection bias, (self) selection bias, or negative legacy [33, 43, 48, 60, 62].

the population. As a result, the available data is not representative of $\mathcal{I}$ and, for this reason, the ML model likely does not generalize well for the disadvantaged group [60].

ML model feedback loops can also affect representation bias. The ML model feedback loop changes the sample of individuals whose realized outcome becomes observable, i.e., those that are selected and can thus be added as a new feature-label pair (x, y) to the sample (X, Y)[7]. Therefore, it can affect representation bias, which stems from a shift in the training data distributions[8].

Life Bias. Individual feedback loops can affect life bias[9]. Individual feedback loops act on the *construct space* (CS) of an individual, i.e., the inherent properties of an individual θ change and not only the observed proxies x, y, which are measured in the *observed space* (OS). This can affect life bias, which describes injustices that manifest in inequality between groups in the CS [27]. As decisions can change individuals' properties θ, which can manifest in altered future features x, it becomes more difficult to treat individuals fairly since the decision actually changed them. This is sometimes also called historical bias, where, at a certain point in time, the world is accurately represented by the data (i.e., the measurement functions r and t are acceptable), but the state of the world (i.e., an individual's inherent decision-relevant attributes θ) is the result of unfair treatments in previous decision rounds [60]. For example, not considering counterfactual decisions for individuals (i.e., assuming that individuals would have evolved identically over time, even if they had been assigned different decisions) can drive the decision system to a state in which individuals are disadvantaged solely because of an unlucky event in the past, even if their attributes are perfectly measurable.

Measurement Bias. Outcome feedback loops and *feature feedback loops* can affect measurement bias [18, 43, 48, 60, 62]. These two feedback loops act on the measurement functions r and t and thus affect an individual's observable properties x, a, y.[10] Thus, both types of feedback loops can affect measurement bias: the features x and labels y are usually just proxies as they try to measure an inherent property of an individual, which might represent a construct that is not directly measurable or even observable (θ) [60]. Measurement bias describes the transition between CS and OS [18]. Thus, it describes a situation in which those proxies less closely approximate the intended attribute for certain individuals or groups, which means that r or t (or both) are not appropriate to capture the relevant construct.[11] For example, using arrests as a proxy for the risk of committing a crime (as is the case in the recidivism risk prediction tool COMPAS [1]) is problematic if there are groups that are much more likely to be arrested for certain crimes.

5 CASE STUDY: FEEDBACK LOOPS IN RECOMMENDER SYSTEMS

We demonstrate the connection between feedback loops and biases with a unifying case study on recommender systems (RS). We consider the case of an online platform where the RS is used to provide content the users are interested in. For simplicity, we consider just one relevant item (e.g., a specific video) and denote a user's interest in this item with

[7] This process is visualized in Fig. 5 in Appendix C.

[8] Notice that, additionally, ML model feedback loops can affect evaluation bias. More generally, evaluation bias exists if the sample used to evaluate an ML-based decision system does not represent the population it is used for, i.e., it stems from a shift in the evaluation data distributions [43, 48, 60, 62]. Hence, ML model feedback loops can affect evaluation bias if the sample used for evaluation (X_m, Y_m) is influenced by past decisions.

[9] The idea of life bias is sometimes referred to as historical bias, individual bias, social bias, societal bias, or pre-existing bias [33, 43, 48, 60, 62].

[10] The outcome feedback loop changes the realization of the outcome (y). In contrast, the feature feedback loop changes the observable attributes that are fed into the prediction model (x and, potentially, a), i.e., the features for future decisions.

[11] Notice that the list of the biases in Table 2 is non-exhaustive since some biases are connected: feedback loops could also indirectly affect learning bias or aggregation bias. Learning bias represents a limitation of the learned function f that occurs when erroneously assuming that $p(y|x)$ is homogeneous across groups [60]. Aggregation bias arises if the ML-based system fails to draw the correct conclusions for certain individuals and, therefore, results in disproportionately worse decisions for some group [60]. For example, feature or outcome feedback loops directly affect measurement bias and indirectly affect learning bias at the same time if the shift of the distribution (X, Y) – which is connected to measurement bias – also results in heterogeneous conditional probabilities, $p(y|x)$, of getting a certain output for a given input across groups. Similarly, they could indirectly result in learning bias if the way the learned function f is optimized is less suited for the new distribution (X, Y).

Table 3. Initial conditions for the different experiments. The acronyms stand for group 1 (G1), group 2 (G2), population size (n), training sample size (n_{train}), distribution mean (μ) and standard deviation (σ), distribution of the feature realization (r), distribution of the outcome realization during the simulation (t) and for the initial training set (t_{train}). For all experiments, we set the following parameters: $n = 1000$, $n_{\text{train},G1} = n_{\text{train},G2} = 500$, $\sigma_{\theta,G2} = 0.15$, $\sigma_{t,G1} = 0.1$, $\mu_{r,G1} = 0$, $\mu_{t,G1} = 0$, $\sigma_{\theta,G1} = 0.15$, $\mu_{t,G2} = 0$, $\sigma_{t,G2} = 0.1$, $\mu_{t_{\text{train}}} = 0$. In the table, we describe the parameters that vary from one experiment to another.

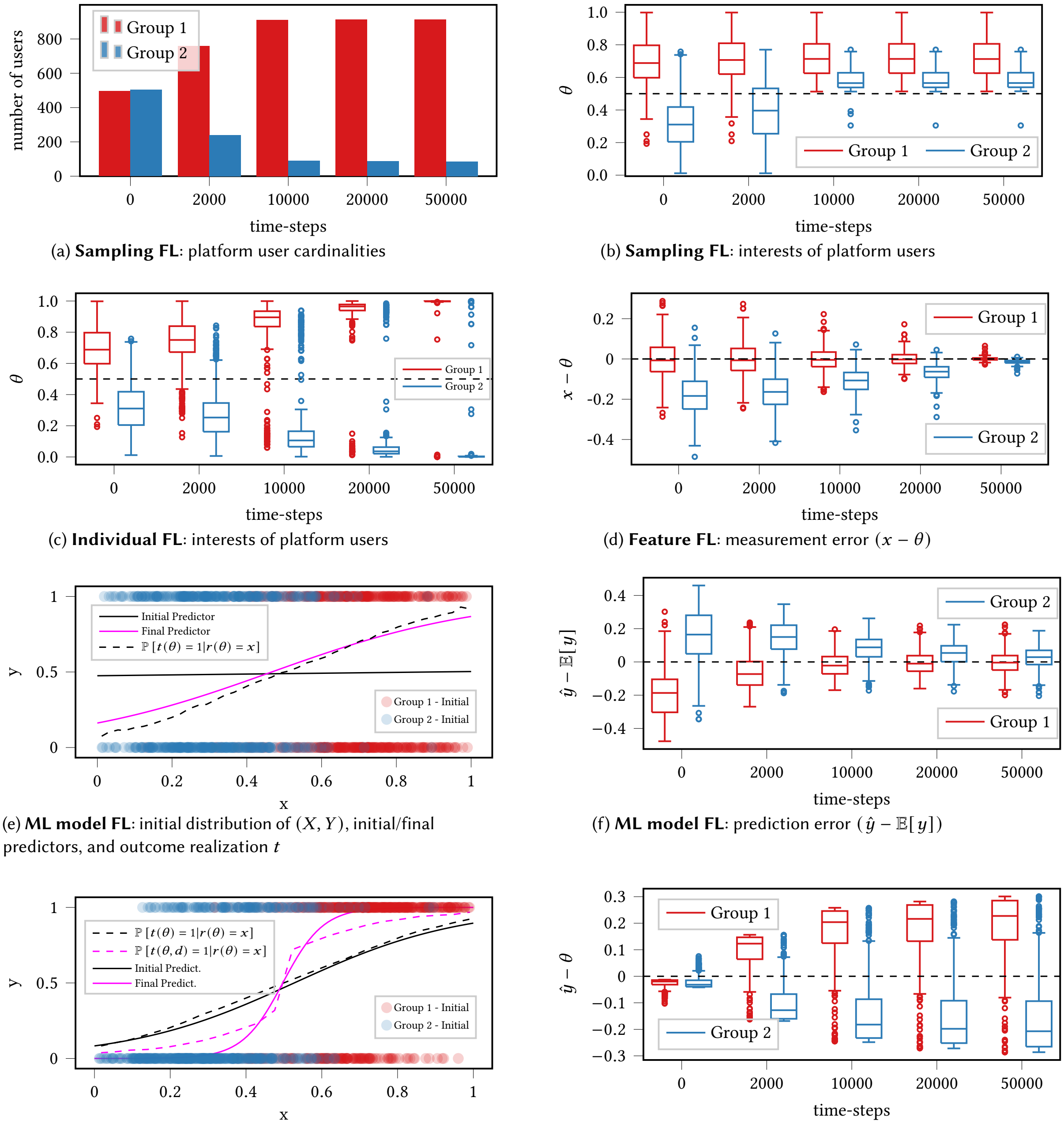
Feedback loop	$\mu_{\theta,G1}$	$\mu_{\theta,G2}$	$\sigma_{r,G1}$	$\mu_{r,G2}$	$\sigma_{r,G2}$	$\sigma_{t_{\text{train}}}$
Sampling, Individual, Outcome	0.7	0.3	0.0	0.0	0.0	0
ML model						1
Feature	0.5	0.5	0.1	-0.2	0.1	0

$\theta \in [0, 1]$, where a larger θ corresponds to a higher interest. The realized outcome y denotes whether a user shows interest (e.g., clicks on the relevant item in question), $y = 1$, or not, $y = 0$. The platform uses an RS to predict a user's interest $\hat{y} = f(x)$, where the feature $x \in [0, 1]$ represents the user's past clicking behavior on the platform. For this simple example, x is the percentage of recommended relevant items that the user has clicked on in the past and thus serves as a proxy of the user's interest in the relevant item. The function $f : [0, 1] \longrightarrow [0, 1]$ is learned through a logistic regression (LR) algorithm (which is fitted to a sigmoid function) trained on data (X, Y), which consists of a collection of feature label pairs (x, y). To decide whether the relevant item should be shown as one of the top recommendations ($d = 1$) or not ($d = 0$), the following threshold rule is used: $d = 1$ if $\hat{y} > 0.5$, and $d = 0$ otherwise. After every recommendation round, y is observed and (x, y) is added to the existing dataset (X, Y) and the RS is retrained. We consider two groups of users $a \in \{\text{G1}, \text{G2}\}$, however, for simplicity, a is not used as an input for the RS.

We now provide one example for each type of feedback loop described in Section 3.1 to illustrate how they are associated with different biases. The initial conditions specific to each of these simulation examples are described in Table 3 and the initial θ distribution is shown in Fig. 6 in Appendix D. Notice that the mean is higher for group G1 (i.e., $\mu_{\theta,G1} = 0.7$, $\mu_{\theta,G2} = 0.3$), which means that individuals of group G1 are more interested in the item, on average.

Sampling Feedback Loop. First, we look at a special case in which $d = 0$ corresponds to not receiving any recommendation, leading to users leaving the platform. Instead, users who receive the recommendation ($d = 1$) stay on the platform. Initially, about 50% of the active users on the platform are from group 1 and 50% from group 2: $n_{G1} = 496$ and $n_{G2} = 504$. Every time someone leaves the platform, a new user replaces them. To mimic users' homophily, the new user is drawn from group 1 with probability $p = \frac{n_{G1}}{n}$ (else, from group 2), i.e., the higher the percentage of users from group 1 in the platform, the higher the probability the new user belongs to group 1. As can be seen in Fig. 3a, this phenomenon leads to the reduction of n_{G2} from 504 to 89 individuals after 10,000 time-steps. This distribution persists in future time-steps, suggesting that it is a (locally) stable equilibrium point of the dynamical system. Group 2 is underrepresented on the platform in the long term with just 8.9% of the platform users. This corresponds to nuance (ii) of the representation bias as described in Section 4. However, at the same time, nuance (iii) of the representation bias is present for both groups: since only those given $d = 1$ stay on the platform, the sample of active users becomes less representative over time, i.e., only interested users (those with high values for θ) stay on the platform (see Fig. 3b). Notice that it is difficult to classify the sampling feedback loop as positive or negative in this case, as there is no initial representation bias against Group 2 that gets amplified by the loop. The resulting biased equilibrium point is simply a property of the closed-loop dynamics.

Individual Feedback Loop. An example of an individual feedback loop is when the recommended content influences the user's opinion θ, which we model by letting the new opinion be a convex combination of the previous one and



(a) **Sampling FL**: platform user cardinalities

(b) **Sampling FL**: interests of platform users

(c) **Individual FL**: interests of platform users

(d) **Feature FL**: measurement error $(x - \theta)$

(e) **ML model FL**: initial distribution of (X, Y), initial/final predictors, and outcome realization t

(f) **ML model FL**: prediction error $(\hat{y} - \mathbb{E}[y])$

(g) **Outcome FL**: initial distribution of (X, Y), initial/final predictors, and initial/final outcome realization t

(h) **Outcome FL**: prediction error with respect to the true, unobserved individual characteristics $(\hat{y} - \theta)$

Fig. 3. Dynamic effects of different types of feedback loops (FL) acting on an RS pipeline for an online platform. Circles in the box plots denote outliers.

the recommended content. Fig. 3c shows that this results in a polarization of interests on the platform. Namely, users with high initial interest (i.e., $\theta > 0.5$) are more likely to be recommended the item and, as a result of this, their θ further increases over time, and vice versa for users with low initial interest.[12] Due to the difference in the two groups' initial distributions of θ (see Table 3), the polarization increases life bias. Namely, it results in even bigger group-level disparities with a very high θ for group 1 users and a very low θ for group 2 users, on average. The steady-state value reached by the trajectories in Fig. 3c represents a biased stable equilibrium point of the closed-loop ML system in which the opinions are polarized to the extremes.

Feature Feedback Loop. Fig. 3d shows the result of an example in which the content recommendation feeds back into the feature extraction block x (rather than acting on the actual opinion θ, as was the case in the previous example), thus forming a feature feedback loop. Compared to all the other examples, there is no difference in the mean of the initial θ distribution across groups G1 and G2 $\left(\mu_{\theta,G1} = \mu_{\theta,G2} = 0.5\right)$. However, initially, x is a noisy observation of θ for both groups (i.e., there is measurement error $\sigma_{r,G1} = \sigma_{r,G2} = 0.1$) and there is measurement bias on the feature observation of group 2 $\left(\mu_{r,G1} = 0, \mu_{r,G2} = -0.2\right)$. This means that the interest of individuals of group 2 is systematically underestimated. Over time, the true interest of users is learned, since the feature feedback loop updates x using new accurate information on the users' interests. This reduces the measurement error, which can be seen by the reduced variance in Fig. 3d. Fig. 3d further shows that the feature feedback loop reduces the measurement bias (which is measured as $x - \theta$) of group 2 over time, i.e., $x - \theta \sim 0$ after $50,000$ time-steps. Hence, the system converges to an unbiased stable equilibrium point, and the feature feedback loop leads to eliminating the measurement bias in this case.

ML Model Feedback Loop. Next, we consider the case in which the content recommendation feeds back into the ML model. For the initial training of the model, we specify $\sigma_{t_{train}} = 1$, which is why the trained model f does not contain any information to map an input feature x to an outcome y, resulting in $\hat{y} = 0.5$ for all individuals (see solid black line in Fig. 3e). Over time, the model observes new feature label pairs and becomes more accurate whenever it is retrained, i.e., the final predictor approximates the true outcome realization $t(\theta)$.

In this example, the training dataset (X, Y) is enriched with feature label pairs (x, y) only if the relevant item was recommended to the user (i.e., if $d = 1$, namely, partial feedback), assuming that the platform cannot measure the user's interest in an item that was not recommended. This forms an ML model feedback loop. The result is that after an initial period of exploration, the ML model quickly learns how to predict y for individuals with large values of x, as those become the ones more likely to receive positive decisions. Here, we measure the prediction error as $\hat{y} - \mathbb{E}[y]$, however, in the absence of any measurement error in the outcome realization $(t(\theta) = y)$, θ is approximately equivalent to $\mathbb{E}[y]$ except for some noise, which is negligible for the average over a group of individuals. As can be seen in Fig. 3f, the prediction error quickly approaches 0 for G1, but the LR algorithm continues to perform poorly for G2 in the short to medium term. In the long term, thanks to the noise in the observation of x[13] it eventually approaches 0 also for G2.

Retraining the ML model over time reduces the representation bias, nuance (ii) of the representation bias as described in Section 4. However, it is due to the ML model feedback loop that the sample (X, Y) becomes more representative of group 1 after just very few time-steps while taking much longer to reduce representation bias for group 2.

[12] Notice that the underlying assumption is that any decision reinforces a user's opinion. This also means that users lose interest (i.e., θ decreases) if the relevant item ($d = 0$) is not recommended.

[13] Namely, in few cases, it can happen that an individual from G2 has $x > 0.5$ and therefore receives $d = 1$. Thereby, the RS slowly explores the true distribution of group 2.

Outcome Feedback Loop. Finally, we consider an example using the same initial conditions as in the sampling and individual feedback loops, but this time the RS's decision affects the outcome realization t. Namely, the probability of the realized outcome to be $y = 1$ increases/decreases by 20% for positive/negative decisions, respectively. This means that the realized outcomes $t(\theta, d)$ are more extreme than they would be if there were no outcome feedback loop (see dashed lines in Fig. 3g). Despite starting with an unbiased ML model, over time, the retrained ML model approximates $t(\theta, d)$, i.e., the initial predictor is a much flatter sigmoid function compared to the final predictor. Namely, the outcome feedback loop introduces a measurement bias on the realized outcome y for both groups G1 and G2. Thus, as is visible in Fig. 3h, the prediction error $\hat{y} - \theta$ diverges from 0 (as $\hat{y}$ predicts the realized outcome y and not θ) until it reaches a stable equilibrium point after approximately 10,000 time-steps (at approximately 0.2 and -0.2 for G1 and G2). From the perspective of platform users, an outcome feedback loop can result in a situation in which one keeps receiving recommendations due to having clicked on similar content in the past, despite not being interested in it.

6 RELATED WORK AND DISCUSSION

As we show in the case study in Section 5, feedback loops do not necessarily amplify ML biases over time. Feedback loops steer the system by shifting distributions, which can harm or benefit disadvantaged individuals. These insights extend existing work that have investigated long-term effects of algorithmic fairness and ML-based decision making, which we briefly discuss in the following.

Feedback Loops and Long-Term Fairness. Many researchers started investigating feedback loops and long-term effects on the fairness of ML-based decisions through simulations [12, 14, 17, 28–30, 41, 50, 58, 67–69]. However, these papers lack a common terminology (and, sometimes, an understanding of the specific feedback loops introduced in the dynamical model) that would allow them to compare the results with those of other studies. For example, some ML-based simulation studies include multiple interacting feedback loops without discussing them in isolation, making it more challenging to interpret the driving effects. In this paper, we do not attempt to provide a solution to the existence of feedback loops. Still, we provide a classification of their different types to facilitate a thorough formalization of the assumptions specific to each of these simulation examples and possibly to clear the way for the design of (long-term) bias mitigation techniques [3, 35, 43]. This is very different from what Reader et al. [54] provide, as they study broader societal systems without classifying different types of feedback loops on the level of an ML-based decision-making pipeline.

Distribution Shifts. Many works have investigated ML under different types of distribution shifts over time. The problem of *concept drift* is broadly defined as a shift of the target distribution over time [20, 64]. This is a rather broad definition, which includes distribution shifts due to exogenous effects, e.g., a pandemic or a financial crisis. However, such shifts can be arbitrary and do not assume that feedback loops are present in general. Recently, endogenous distribution shifts, i.e., target distribution shifts caused by the deployed prediction model, have been investigated more thoroughly. The concept of *performative predictions* acknowledges the fact that ML-based decision-making systems can affect the outcome they try to predict [50]. The notion of performative stability, which is defined as a predictor that is not only calibrated against historical data but also against future outcomes that are produced by acting based on the prediction, is a possible solution that achieves a stable point for retraining [50]. This stable point means that a model remains exactly the same if it is retrained on future outcomes. Performative prediction is an umbrella term for a situation where ML-based decisions cause a shift in the outcome distribution. However, this distribution shift can occur through any type of feedback loop we introduced in Section 3.1.[14] As we showed in Section 5, these feedback loops have different properties and implications. For example, changing a platform user's opinion (through an individual feedback loop) is very different from changing the individual's realized outcome (through an outcome feedback loop). In all cases, the individual's consumption changes as a result of the recommendation. In the former case, this is caused by shaped preferences. In contrast, the decision-relevant individual attributes remain unaltered in the latter case. More research is needed to investigate the effects of the specific feedback loops we classify on the concept of performative power [22], which only considers shifting outcome distributions in its more general understanding as in the literature on performative prediction.

Adversarial Machine Learning. Adversarial ML studies attacks on ML algorithms and how they can be defended [31, 63]. The idea is that adversarial attacks are executed by an attacker who intends to influence some part of the ML pipeline, whereas the developer of the ML algorithm thwarts the attacker's objective. In contrast, feedback loops do not occur due to malicious external manipulation but are a direct consequence of the dynamics in sequential decision-making systems. Yet, the outcomes of certain adversarial attacks are closely related to the feedback loops we classify in this paper. For example, data poisoning attacks are associated with ML model feedback loops in that they modify the data used for training. Applying measures designed to counter adversarial attacks to deal with feedback loops in sequential decision-making systems represents an interesting avenue for future research – for example, robust learning through data sub-sampling [34] or trimmed optimization [39] to counter ML model feedback loops. First results have shown that this becomes more complicated if the fairness of the decision-making systems is a concern [65].

7 CONCLUSION

The output of ML-based decision-making systems, i.e., the decision, often affects various parts of the system itself, creating a so-called feedback loop. Yet, ML evaluation techniques usually omit potentially important temporal dynamics [9, 40, 45] and taking feedback loops into account is crucial to avoid unintended consequences [12, 40, 59, 68]. In this work, we build on dynamical systems theory to provide a theoretical framework that sheds light on the different types of feedback loops that can occur throughout the ML pipeline. We identify five distinct types of feedback loops, some of which can be classified as "adversarial" whenever the decision feeds back into the system as a consequence of some strategic action of the affected individual(s). Furthermore, we associate the different types of feedback loops with the corresponding biases they affect, and we demonstrate these dynamics using a recommender system example.

By rigorously analyzing the ML pipeline, we believe that our framework is a necessary first step toward understanding the exact role of the feedback loops in it. Providing a rigorous classification of feedback loops will enable a deeper understanding of the existing works in the ML literature and it will allow putting their results into the perspective of their assumptions (e.g., which types of feedback loops are considered and which are not). However, more research is needed to be able to overcome the challenges posed by distributions shifting over time and to achieve long-term fairness. We believe that our framework will be helpful in purposefully designing feedback loops and developing bias and unfairness mitigation techniques for ML-based decision-making systems.

[14] For certain feedback loops (e.g., sampling feedback loops), the distribution shift is delayed, and for others (such as the outcome feedback loop), the feedback effect occurs immediately, i.e., at the same time-step. Furthermore, a distribution shift caused by an adversarial feature or adversarial individual feedback loop is a special case of performative prediction, which has been referred to as strategic classification [23, 44].

REFERENCES

[1] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias. *ProPublica, May* 23, 2016 (2016), 139–159. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[2] Karl Johan Åström and Richard M Murray. 2021. *Feedback systems: an introduction for scientists and engineers.* Princeton university press.

[3] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning.* fairmlbook.org. http://www.fairmlbook.org

[4] Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Zhiwei Steven Wu. 2019. Equal opportunity in online classification with partial feedback. *Advances in Neural Information Processing Systems* 32, NeurIPS (2019).

[5] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2021. Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods & Research* 50, 1 (2021), 3–44. https://doi.org/10.1177/0049124118782533

[6] Joy Buolamwini and Timnit Gebru. 2018. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (Proceedings of Machine Learning Research, Vol. 81)*, Sorelle A Friedler and Christo Wilson (Eds.). PMLR, 77–91. https://proceedings.mlr.press/v81/buolamwini18a.html

[7] Simon Caton and Christian Haas. 2020. Fairness in Machine Learning: A Survey. (10 2020). http://arxiv.org/abs/2010.04053

[8] Allison J. B. Chaney, Brandon M. Stewart, and Barbara E. Engelhardt. 2018. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. (2018), 224–232. https://doi.org/10.1145/3240323.3240370

[9] Alexandra Chouldechova and Aaron Roth. 2018. The Frontiers of Fairness in Machine Learning. (2018), 1–13. http://arxiv.org/abs/1810.08810

[10] Alexandra Chouldechova and Aaron Roth. 2020. A Snapshot of the Frontiers of Fairness in Machine Learning. *Commun. ACM* 63, 5 (4 2020), 82–89. https://doi.org/10.1145/3376898

[11] Kate Crawford. 2016. Artificial intelligence's white guy problem. *The New York Times* 25, 06 (2016).

[12] Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D. Sculley, and Yoni Halpern. 2020. Fairness is not static: Deeper understanding of long term fairness via simulation studies. *FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (2020), 525–534. https://doi.org/10.1145/3351095.3372878

[13] Robyn M Dawes, David Faust, and Paul E Meehl. 1989. Clinical Versus Actuarial Judgment. *Science* 243, 4899 (1989), 1668–1674. https://doi.org/10.1126/science.2648573

[14] Sarah Dean and Jamie Morgenstern. 2022. Preference Dynamics Under Personalized Recommendations. (2022), 1–25. http://arxiv.org/abs/2205.13026

[15] Hadi Elzayn, Michael Kearns, Shahin Jabbari, Seth Neel, Zachary Schutzman, Christopher Jung, and Aaron Roth. 2019. Fair algorithms for learning in allocation problems. *FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency* (2019), 170–179. https://doi.org/10.1145/3287560.3287571

[16] Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, Suresh Venkatasubramanian, Mehryar Mohri, and Karthik Sridharan. 2018. Decision making with limited feedback: Error bounds for predictive policing and recidivism prediction. *Proceedings of Machine Learning Research* 83 (2018), 1–9.

[17] Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, Suresh Venkatasubramanian, and Christo Wilson. 2018. Runaway Feedback Loops in Predictive Policing. In *Proceedings of Machine Learning Research*, Vol. 81. 1–12. https://github.com/algofairness/

[18] Sorelle A Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. 2016. On the (im)possibility of fairness. https://arxiv.org/abs/1609.07236

[19] Andreas Fuster, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther. 2022. Predictably Unequal? The Effects of Machine Learning on Credit Markets. *Journal of Finance* 77, 1 (2022), 5–47. https://doi.org/10.1111/jofi.13090

[20] João Gama, Indrundefined Žliobaitundefined, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A Survey on Concept Drift Adaptation. *ACM Comput. Surv.* 46, 4 (3 2014). https://doi.org/10.1145/2523813

[21] W M Grove, D H Zald, B S Lebow, B E Snitz, and C Nelson. 2000. Clinical versus mechanical prediction: a meta-analysis. *Psychological assessment* 12, 1 (3 2000), 19–30.

[22] Moritz Hardt, Meena Jagadeesan, and Celestine Mendler-Dünner. 2022. Performative Power. In *Advances in Neural Information Processing Systems*, Alice H Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.). https://doi.org/10.48550/ARXIV.2203.17232

[23] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. 2016. Strategic Classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science (ITCS '16)*. Association for Computing Machinery, New York, NY, USA, 111–122. https://doi.org/10.1145/2840728.2840730

[24] Drew Harwell. 2018. Amazon's Alexa and Google Home show accent bias, with Chinese and Spanish hardest to understand. https://www.scmp.com/magazines/post-magazine/long-reads/article/2156455/amazons-alexa-and-google-home-show-accent-bias

[25] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness Without Demographics in Repeated Loss Minimization. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 1929–1938. https://proceedings.mlr.press/v80/hashimoto18a.html

[26] Hoda Heidari, Vedant Nanda, and Krishna P. Gummadi. 2019. On the Long-term Impact of Algorithmic Decision Policies: Effort unfairness and feature segregation through social learning. *36th International Conference on Machine Learning, ICML 2019* 2019-June (2019), 4787–4796.

[27] Corinna Hertweck, Christoph Heitz, and Michele Loi. 2021. On the Moral Justification of Statistical Parity. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21)*. Association for Computing Machinery, New York, NY, USA, 747–757. https://doi.org/10.1145/3442188.3445936

[28] Lily Hu and Yiling Chen. 2018. A short-term intervention for long-term fairness in the labor market. *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018* 2 (2018), 1389–1398. https://doi.org/10.1145/3178876.3186044

[29] Lily Hu, Nicole Immorlica, and Jennifer Wortman Vaughan. 2019. The Disparate Effects of Strategic Manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 259–268. https://doi.org/10.1145/3287560.3287597

[30] Yaowei Hu and Lu Zhang. 2022. Achieving Long-Term Fairness in Sequential Decision Making. *Proceedings of the AAAI Conference on Artificial Intelligence* 36, 9 (2022), 9549–9557. https://doi.org/10.1609/aaai.v36i9.21188

[31] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin I P Rubinstein, and J D Tygar. 2011. Adversarial Machine Learning. In *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence (AISec '11)*. Association for Computing Machinery, New York, NY, USA, 43–58. https://doi.org/10.1145/2046684.2046692

[32] Sterman John D. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. McGraw-Hill Education.

[33] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-Aware Classifier with Prejudice Remover Regularizer. In *Machine Learning and Knowledge Discovery in Databases*, Peter A Flach, Tijl De Bie, and Nello Cristianini (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 35–50.

[34] Michael Kearns and Ming Li. 1993. Learning in the Presence of Malicious Errors. *SIAM J. Comput.* 22, 4 (1993), 807–837. https://doi.org/10.1137/0222052

[35] Michael Kearns and Aaron Roth. 2019. *The Ethical Algorithm: The Science of Socially Aware Algorithm Design*. Oxford University Press, Inc., USA.

[36] Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. 2017. *Human Decisions and Machine Predictions*. Technical Report 23180. National Bureau of Economic Research. https://doi.org/10.3386/w23180

[37] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Ashesh Rambachan. 2018. Algorithmic Fairness. *AEA Papers and Proceedings* 108 (5 2018), 22–27. https://doi.org/10.1257/pandp.20181018

[38] Jon Kleinberg and Manish Raghavan. 2020. How Do Classifiers Induce Agents to Invest Effort Strategically? *ACM Transactions on Economics and Computation* 8, 4 (2020). https://doi.org/10.1145/3417742

[39] Chang Liu, Bo Li, Yevgeniy Vorobeychik, and Alina Oprea. 2017. Robust Linear Regression Against Training Data Poisoning. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (AISec '17)*. Association for Computing Machinery, New York, NY, USA, 91–102. https://doi.org/10.1145/3128572.3140447

[40] Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. 2018. Delayed Impact of Fair Machine Learning. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 3150–3158. https://proceedings.mlr.press/v80/liu18c.html

[41] Lydia T. Liu, Adam Tauman Kalai, Ashia Wilson, Christian Borgs, Nika Haghtalab, and Jennifer Chayes. 2020. The disparate equilibria of algorithmic decision making when individuals invest rationally. *FAT* 2020 - Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (2020), 381–391. https://doi.org/10.1145/3351095.3372861

[42] Kristian Lum and William Isaac. 2016. To predict and serve? *Significance* 13, 5 (2016), 14–19. https://doi.org/10.1111/j.1740-9713.2016.00960.x

[43] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* 54, 6 (7 2021). https://doi.org/10.1145/3457607

[44] Smitha Milli, John Miller, Anca D. Dragan, and Moritz Hardt. 2019. The social cost of strategic classification. *FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency* (2019), 230–239. https://doi.org/10.1145/3287560.3287576

[45] Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. 2021. Algorithmic Fairness: Choices, Assumptions, and Definitions. *Annual Review of Statistics and Its Application* 8, 1 (3 2021), 141–163. https://doi.org/10.1146/annurev-statistics-042720-125902

[46] Hussein Mouzannar, Mesrob I. Ohannessian, and Nathan Srebro. 2019. From fair decision making to social equality. *FAT* 2019 - Proceedings of the 2019 Conference on Fairness, Accountability, and Transparency* (2019), 359–368. https://doi.org/10.1145/3287560.3287599

[47] Nataša Obermajer, Ravikumar Muthuswamy, Jamie Lesnock, Robert P Edwards, and Pawel Kalinski. 1983. Positive feedback between PGE$_2$ and COX2 redirects the differentiation of human dendritic cells toward stable myeloid-derived suppressor cells. *Immunobiology* 119, 20 (1983). https://doi.org/10.1182/blood-2011-07-365825

[48] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. 2019. Social Data: Biases, Methodological Pitfalls, and Ethical Boundaries. *Frontiers in Big Data* 2 (7 2019). https://doi.org/10.3389/fdata.2019.00013

[49] Cathy O'neil. 2017. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown.

[50] Juan C. Perdomo, Tijana Zrnic, Celestine Mendler-Dunner, and Moritz Hardt. 2020. Performative prediction. *37th International Conference on Machine Learning, ICML 2020* PartF16814 (2020), 7555–7565.

[51] Nicola Perra and Luis E C Rocha. 2019. Modelling opinion dynamics in the age of algorithmic personalisation. *Scientific reports* 9, 1 (2019), 1–11.

[52] Dana Pessach and Erez Shmueli. 2022. A Review on Fairness in Machine Learning. *ACM Comput. Surv.* 55, 3 (2 2022). https://doi.org/10.1145/3494672

[53] Arkalgud Ramaprasad. 1983. On the definition of feedback. *Journal of the Society for General Systems Research* 28, 1 (1983). https://doi.org/10.1002/bs.3830280103

[54] Lydia Reader, Pegah Nokhiz, Cathleen Power, Neal Patwari, Suresh Venkatasubramanian, and Sorelle Friedler. 2022. Models for Understanding and Quantifying Feedback in Societal Systems. In *2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT '22)*. Association for Computing Machinery, New York, NY, USA, 1765–1775. https://doi.org/10.1145/3531146.3533230

[55] Wilbert Samuel Rossi, Jan Willem Polderman, and Paolo Frasca. 2021. The closed loop between opinion formation and personalised recommendations. *IEEE Transactions on Control of Network Systems* (2021), 1. https://doi.org/10.1109/TCNS.2021.3105616

[56] Tom Simonite. 2015. Probing the dark side of google's ad-targeting system. *MIT Technology Review* (2015).

[57] Ayan Sinha, David F Gleich, and Karthik Ramani. 2016. Deconvolving Feedback Loops in Recommender Systems. In *Advances in Neural Information Processing Systems*, D Lee, M Sugiyama, U Luxburg, I Guyon, and R Garnett (Eds.), Vol. 29. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2016/file/962e56a8a0b0420d87272a682bfd1e53-Paper.pdf

[58] Yi Sun. 2022. *Algorithmic Fairness in Sequential Decision Making.* Ph. D. Dissertation.

[59] Yi Sun, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2022. The Backfire Effects of Fairness Constraints. *ICML 2022 Workshop on Responsible Decision Making in Dynamic Environments* (2022). https://responsibledecisionmaking.github.io/assets/pdf/papers/44.pdf

[60] Harini Suresh and John Guttag. 2021. A Framework for Understanding Sources of Harm throughout the Machine Learning Life Cycle. In *Equity and Access in Algorithms, Mechanisms, and Optimization (EAAMO '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3465416.3483305

[61] Stratis Tsirtsis, Behzad Tabibian, Moein Khajehnejad, Adish Singla, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. 2019. Optimal Decision Making Under Strategic Behavior. (2019). http://arxiv.org/abs/1905.09239

[62] Benjamin van Giffen, Dennis Herhausen, and Tobias Fahse. 2022. Overcoming the pitfalls and perils of algorithms: A classification of machine learning biases and mitigation methods. *Journal of Business Research* 144 (2022), 93–106. https://doi.org/10.1016/j.jbusres.2022.01.076

[63] Yevgeniy Vorobeychik and Murat Kantarcioglu. 2018. *Adversarial Machine Learning.* Springer International Publishing, Cham. https://doi.org/10.1007/978-3-031-01580-9

[64] Geoffrey I Webb, Roy Hyde, Hong Cao, Hai Long Nguyen, and Francois Petitjean. 2016. Characterizing concept drift. *Data Mining and Knowledge Discovery* 30, 4 (2016), 964–994. https://doi.org/10.1007/s10618-015-0448-4

[65] Han Xu, Xiaorui Liu, Yaxin Li, Anil Jain, and Jiliang Tang. 2021. To be Robust or to be Fair: Towards Fairness in Adversarial Training. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 11492–11501. https://proceedings.mlr.press/v139/xu21b.html

[66] Bernard P Zeigler, Tag Gon Kim, and Herbert Praehofer. 2000. *Theory of modeling and simulation.* Academic press.

[67] Xueru Zhang, Mohammad Mahdi Khalili, and Mingyan Liu. 2020. Long-Term Impacts of Fair Machine Learning. *Ergonomics in Design* 28, 3 (2020), 7–11. https://doi.org/10.1177/1064804619884160

[68] Xueru Zhang, Mohammad Mahdi Khalili, Cem Tekin, and Mingyan Liu. 2019. Group retention when using machine learning in sequential decision making: The interplay between user dynamics and fairness. *Advances in Neural Information Processing Systems* 32, NeurIPS (2019).

[69] Xueru Zhang and Mingyan Liu. 2021. Fairness in Learning-Based Sequential Decision Algorithms: A Survey. *Studies in Systems, Decision and Control* 325 (2021), 525–555. https://doi.org/10.1007/978-3-030-60990-0{_}18

[70] Xueru Zhang, Ruibo Tu, Yang Liu, Mingyan Liu, Hedvig Kjellström, Kun Zhang, and Cheng Zhang. 2020. How do fair decisions fare in long-term qualification? *Advances in Neural Information Processing Systems* 2020-Decem, NeurIPS (2020), 1–13.

A LITERATURE SEARCH PROCESS

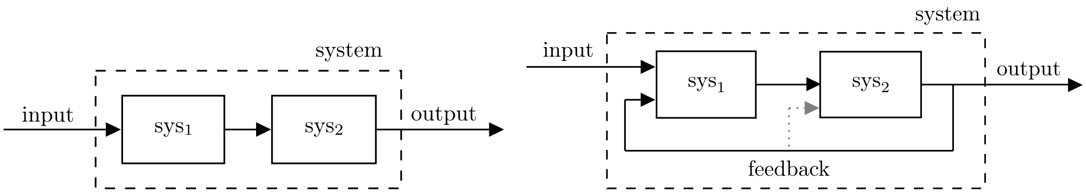
We conducted a literature search to identify relevant articles for our analysis. We started by performing a forward and backward search using an initially small set of important papers [12, 17, 40, 69]. Then, we performed a keyword search on the ACM Digital Library and google scholar using various combinations of the following keyword: "feedback loop," "algorithmic bias," "algorithmic fairness," "machine learning," and "feedback." We performed the classification of existing works in three stages: First, we skimmed each paper and added it to a list of potentially relevant papers if it contained some mention of feedback effects and ML – this list consisted of 75 papers. Next, we looked at each paper in more detail and only included it in the final base of literature to be considered if it investigates feedback loops as part of ML-based decision-making systems. Finally, looked at each description of feedback loops in those papers and mapped it to our conceptualization of feedback loops. This resulted in the 24 papers listed in Table 1 and, through an iterative process, ultimately also in Figure 2 presented in Section 3.1.

B OPEN- AND CLOSED-LOOP DYNAMICAL SYSTEMS

In Fig. 4, we provide a simple visualization of the two types of *series* and *feedback interconnections.* A *series interconnection* (Fig. 4a) composes two systems into an *open-loop system.* A *feedback interconnection* (Fig. 4b) composes them into a *closed-loop system*: the output is injected back as an input to one (or more) of the components, creating a *feedback loop.*

Unlike their open-loop counterpart, closed-loop systems are not straightforward to predict from their components and require specialized techniques to analyze.

The interconnections of systems are systems themselves, and a model for the interconnected system can be derived from the models of the individual subsystems that compose the interconnection. This derivation is straightforward in the case of series interconnection, but it becomes significantly more involved in the case of feedback interconnection. Only for special classes of systems, for example, systems where the input-output relation of each subsystem is linear, a model for the resulting interconnected system can be derived analytically. For general dynamical systems, a tractable direct derivation is typically very difficult, and numerical methods (including simulations) come to help.



(a) Open-loop

(b) Closed-loop

Fig. 4. Open- and closed-loop dynamical systems

C ML-BASED DECISION-MAKING PIPELINE

Fig. 5 visualizes a more detailed ML pipeline, also zooming into the ML model development process. It shows that once observed, an individual's feature label pair x, y can end up in a sample (X, Y) that is used to (re)train and evaluate a predictor. This sample is split into training data (X_n, Y_n) and testing data (X_m, Y_m). The training data is used to learn a function $f : x \rightarrow \hat{y}$. This learned function is evaluated using the test data, which outputs some evaluation metrics E that are computed using a function $k : f, X_m, Y_m \rightarrow E$. Finally, f is used to predict the outcome of previously unseen feature values in the next iteration.

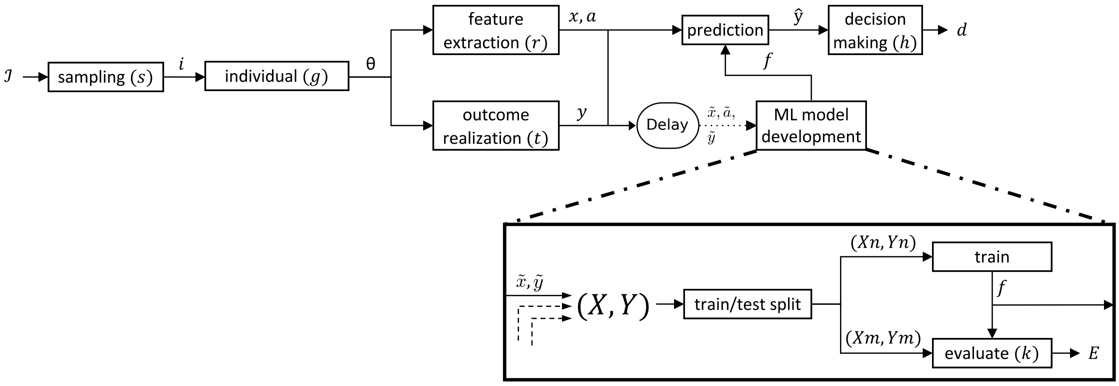


Fig. 5. The detailed ML-based decision-making pipeline

Notice that this is an example of a potential extension of the feedback loop classification presented in Section 3.1: the ML model feedback loop could also be split up into an ML model training feedback and an ML model evaluation feedback loop. However, once a feature label pair is added to (X, Y), it is usually assigned randomly to either the training or the evaluation data (or even to both in the process of cross-validation), which is why we chose to use the umbrella term ML model feedback loop.

D ADDENDUM TO CASE STUDY ON RECOMMENDER SYSTEMS

Fig. 6 shows the initial empirical θ distribution used in all the examples in Section 5 except the one on feature feedback loop. For the feature feedback loop, the two initial θ distributions for groups G1 and G2 have the same mean value of $(\mu_{\theta,G1} = \mu_{\theta,G2} = 0.5)$, i.e., in this case, there is no group-level difference between the platform users' interests.

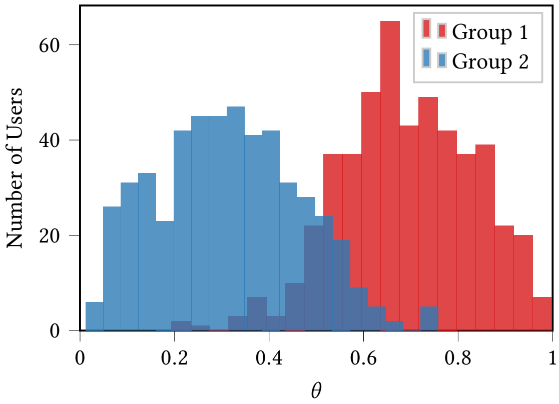


Fig. 6. Initial empirical θ distribution used in all the examples except the one on feature feedback loop.