Exposing Algorithmic Bias through Inverse Design

Carmen Mazijn [1 2] **Carina Prunkl** [3] **Andres Algaba** [1] **Jan Danckaert** [2] **Vincent Ginis** [1 2 4]

Abstract

Traditional group fairness notions assess a model's equality of outcome by computing statistical metrics on the outputs. We argue that these output metrics encounter fundamental obstacles and present a novel approach that aligns with equality of treatment. Through gradient-based inverse design, we generate a canonical set that shows the desired inputs for a model given a preferred output. The canonical set reveals the internal logic of the model and thereby exposes potential unethical biases. For the UCI Adult data set, we find that the biases detected by a canonical set interestingly differ from those of output metrics.

1. Introduction

Artificial intelligence (AI) systems are used in decision-making processes throughout all aspects of human life, ranging from detecting child abuse, determining access to education or healthcare, and granting loans (Amrit et al., 2017; Ledford, 2019; Makhlouf et al., 2021). However, it is by now a well-established fact that algorithms can be biased and lead to discrimination against already disadvantaged population groups (Barocas & Selbst, 2016; Chouldechova & Roth, 2018; Whittaker et al., 2018; Buolamwini & Gebru, 2018). The sources of such biases are multiple and include problem specification, historical bias, unrepresentative data, biased measurement methods, or choice of objective function (Fazelpour & Danks, 2021; Barocas et al., 2019; Lee & Singh, 2021b; Suresh & Guttag, 2021).

Recent efforts to identify algorithmic discrimination often focus on the statistical properties of a model's *output*. The standard approach is to translate philosophical or political notions of group fairness into a statistical metric (Makhlouf et al., 2021). The model's output can then be analysed with respect to the chosen notion of group fairness and the model is judged to be "fair" or "unfair". There are several widely recognised issues with output-based fairness evaluations of this kind. For one, there often is substantial philosophical disagreement as to what ought to be considered a "fair" outcome distribution (Binns, 2018; Gallie, 2019). The now infamous controversy about the alleged racism of the COMPAS recidivism risk algorithm boiled down to such a disagreement. In this case, the two fairness measures under debate were accuracy equality and equalised odds with respect to race. Second, different notions of group fairness are incompatible with each other, except for highly special cases (Kleinberg et al., 2016). Third, the computation of these metrics depends on a benchmark dataset. Usually this is done on the set that was used to evaluate the model on other metrics such as accuracy. However, by using a portion of the training data, the metric is only calculating how well a model learned to optimize in its task with respect to this dataset, not to the whole population after deployment (Northcutt et al., 2021). Fourth, work on group fairness usually relies on the evaluation of a limited number of prescribed protected attributes, running risk of missing discrimination either against people who are at the intersection of different groups or against groups that do not share a protected characteristic (Crenshaw, 1990; Binns, 2020; Wachter & Mittelstadt, 2019).

Finally, focussing exclusively on output distributions to determine fairness is only part of the story. In everyday life and when stakes are high, we are also interested in how the decision came about, e.g. *why* I wasn't granted the loan I applied for or *why* I didn't receive the job I interviewed for (AIHLEG, 2019; Wachter et al., 2017). Understanding the reasoning behind a decision is not just relevant from a moral point of view, it is equally important within a legal context. Disparate treatment and direct discrimination both aim at addressing cases in which similarly situated individuals are not treated alike on grounds of a protected characteristic. In these cases, it becomes relevant both *that* such individuals were treated differently and *why* they were treated differently. Output-based fairness evaluations cannot address these issues as they do not take into account the internal logic of the model in question.

While output-based fairness evaluations remain important in

[1]Data Analytics Lab, Vrije Universiteit Brussel, Brussels, Belgium [2]Applied Physics, Vrije Universiteit Brussel, Brussels, Belgium [3]AI Lab, Oxford University, Oxford, United Kingdom [4]School of Engineering and Applied Sciences, Harvard University, Cambridge, United States of America. Correspondence to: Carmen Mazijn <carmen.mazijn@vub.be>.

ICML 2022 Workshop on Responsible Decision Making in Dynamic Environments, Baltimore, Maryland, USA, 2022.

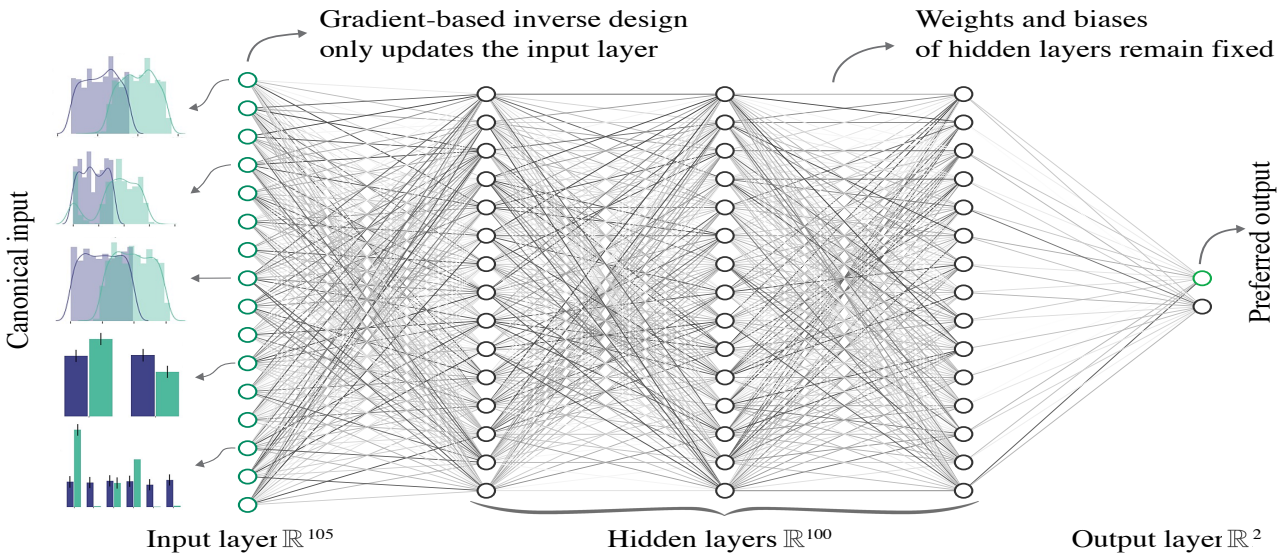

Figure 1. Through gradient-based inverse design on the input layer, we generate a canonical set for a preferred output of a trained decision-making algorithm. The weights and biases of the hidden layers remain fixed; the greyscale of each connection encodes the fixed value. The canonical set reveals the model's internal logic and is visualized via the histograms. We assess the model's fairness by analyzing whether the distributions of the protected features within the canonical set are balanced.

the fight against discrimination, we here present a complementary method that takes into account a model's internal logic. In particular, we introduce the notion of a "canonical set" that allows us to evaluate the fairness of a model's decision-making processes. Through gradient-based inverse design, we generate a canonical input, which can be thought of as the desired input given a preferred output for a trained model (see Fig. 1). The canonical set then emerges from repeatedly interrogating the model's decision-making process by generating canonical inputs. By revealing information about the model's mechanisms, the canonical set provides us with information as to how the model reaches certain decisions, e.g. what features play a role in the model's decision-making process. To expose potential unethical biases in the model's logic, we inspect the distribution of a protected demographic feature within the canonical set. This approach aligns with a focus on equality of treatment rather than a focus on equality of outcome. In contrast to output metrics, there is no need for a specific fairness metric, a ground truth, or a benchmark data set.

We show that canonical inputs can be obtained for any differentiable model but that the resulting canonical set in its current form is only meaningful for tabular data. To illustrate our approach, we evaluate a binary fully-connected neural network classifier on the UCI Adult data set (Dua & Graff, 2017). We find that analyzing the canonical set exposes several unethical biases, which interestingly differ from those found by traditional group fairness metrics.

2. Background and Related Work

The canonical sets lie at the intersection of fairness and interpretability in algorithmic decision-making. There is a strong interaction between these fields, and their connections (Meng et al., 2022) and trade-offs (Kleinberg & Mullainathan, 2019) are part of ongoing research. The biggest group of methods to gauge fair decision-making translate philosophical or political notions of group fairness into mathematical statements on the model's output (Makhlouf et al., 2021). The number of this kind of fairness metrics has grown over the past years, accounting for at least 19 definitions (Barocas et al., 2019; Hardt et al., 2016; Zafar et al., 2017). Furthermore, most prominent open-source fairness toolkits rely on these statistical output metrics (Lee & Singh, 2021a). While the model's output can help foster understanding, it remains a black box when its internal machinery is opaque. An essential element of a model is thus the logic of how it takes decisions. For decision-making algorithms, this is especially important in order to increase the transparency of and trust in high-impact decisions (AIHLEG, 2019).

Over the past few years, much work has been done on post hoc interpretability methods, especially in the computer vision literature (Das & Rad, 2020). The most prominent example is feature importance estimation methods that help understand which features have a high impact on the output of a model by giving a score to each input. While the feature

Algorithm 1 Canonical sets, our proposed algorithm. The default values in this paper are: Number of canonical inputs in the set $N = 1000$, Number of epochs $E = 200$, Learning rate $\alpha = 0.1$, a binary classifier s, and a cross-entropy loss function f.

Require: $s(X)$: Trained model with an input vector X.
Require: $f(\hat{y}|y)$: Objective function with a prediction $\hat{y}$ and preferred output y.
$M \leftarrow length(X)$
for $i = 0, ..., N$ **do**
  $\{X_i^{(m)}\}_{m=1}^{M} \sim \mathcal{U}(0, 1)$
  for $j = 0, ..., E$ **do**
    $\hat{y}_j \leftarrow s(X_i)$
    $X_i \leftarrow X_i - \alpha \nabla_{X_i} f(\hat{y}_j|y)$
  end for
end for

importance estimation methods differ in various ways, they can be broadly categorized into perturbation- and gradient-based explanations (Agarwal et al., 2021). LIME (Ribeiro et al., 2016) and SHAP (Lundberg & Lee, 2017) are examples of the former as these methods construct local explanations of decision-making algorithms by using perturbations of individual samples. However, they have their drawbacks as the resulting explanations are found to be unreliable (Slack et al., 2020), they only relate to one prediction class, and they do not account for feature dependence (Gohel et al., 2021). Nevertheless, perturbation-based methods are often used in combination with statistical fairness metrics (Datta et al., 2016). Thereby, this interpretable fairness analysis inherits the obstacles from the output metrics, namely philosophical disagreements (Binns, 2018; Gallie, 2019), statistical incompatibilities (Kleinberg et al., 2016), the absence of universal ground truth, and the selection of a benchmark data set (Northcutt et al., 2021).

The Integrated Gradients (Sundararajan et al., 2017) method is an example of gradient-based explanations as it uses the gradients of the outputs of individual samples with respect to their inputs to construct local explanations. Importantly, the gradients can also be used to generate global explanations by creating an input with the highest activation and certainty for a specific output starting from random noise (Simonyan et al., 2014). The technique to determine hidden parameters of a complex system through inverse design has been used in several other research fields, including physics, computer science, engineering, and biotechnology (Ferruz & Höcker, 2022; Forte et al., 2022; Lenaerts et al., 2021; Ren et al., 2020). We show in the following section that the canonical sets build upon these methods as they are the result of repeatedly applying inverse design to generate canonical inputs, thereby revealing the internal logic of a trained model.

3. Retrieving the Canonical Set using Inverse Design

Conventional neural networks use gradient descent to improve their workings by taking advantage of their mathematical structure, which can be differentiated straightforwardly (Nielsen, 2015). All the layers in a model can be optimized through gradient descent, including the input values. Indeed, the input vector can be seen as a special layer of the model. In our technique, we use this property to create a canonical input for a preferred output. In other words, starting from a random input vector one can construct the ideal input of a trained model through gradient descent on the input layer. To make this work, one needs to keep the weights and biases of the hidden layers of the model fixed.

This gradient-based inverse design has been extensively used in the computer vision literature (Mordvintsev et al., 2015; Simonyan et al., 2014; Sundararajan et al., 2017), but there is a key difference in our application to tabular data. For images, canonical inputs are interpretable individually and difficult to aggregate, whereas for tabular data we have the opposite scenario. In addition, due to the stochastic nature of randomly generated vectors, there is little information in the canonical version of a single input vector. Therefore, we generate a canonical set which results from repeatedly interrogating the model's decision-making process by generating canonical inputs, revealing its internal workings. To expose potential unethical biases in the model's logic, we inspect the distribution of a protected demographic feature within the canonical set, and compare it to the initial random distribution. This approach aligns with the increasing focus on equality of treatment beyond equality of outcome, as this requires interpretability, which builds and supports trust, and contributes to procedural fairness. In contrast to output metrics, there is no need for a specific fairness metric, a ground truth, or a benchmark data set.

In Algorithm 1, we show an example of how a canonical set can be generated for a trained binary classifier by updating a random input vector via gradient descent on the input layer. First, we generate an extensive set of randomly initialized input vectors where the features are sampled from a uniform distribution. Then, these randomly initialized input vectors are transformed into canonical inputs through inverse design. The transformations are the result of minimizing the (cross-entropy) loss between the model predictions and the preferred output (e.g., a loan is granted) . We refer to the Appendix for all the details on the design considerations, such as initialization and the impact of hyperparameters. Afterwards, the canonical set is analyzed to learn about the internal workings of the model and evaluated if the model is insensitive to protected attributes.

Table 1. Positivity Rates and True Positive Rates of subpopulations for the Protected Features

	MALE	FEMALE	WHITE	ASIAN PAC. ISLANDER	AMER. INDIAN ESKIMO	BLACK	OTHER
PR	24.7	8.0	47.7	0.6	38.6	0.6	4.2
TPR	60.0	54.6	75.0	29.5	62.4	13.3	34.0
	MARRIED	DIVORCED	NEVER MARRIED	SEPARATED	WIDOWED	SPOUSE ABSENT	MILITARY SPOUSE
PR	38.8	4.3	1.6	3.4	5.0	5.7	42.9
TPR	63.4	36.5	28.5	33.3	46.5	41.7	100.0
	WIFE	OWN CHILD	HUSBAND	NOT IN FAMILY	OTHER RELATIVES	UNMARRIED	
PR	47.7	0.6	38.6	4.2	0.6	2.4	
TPR	75.0	29.5	62.4	34.0	13.3	34.1	

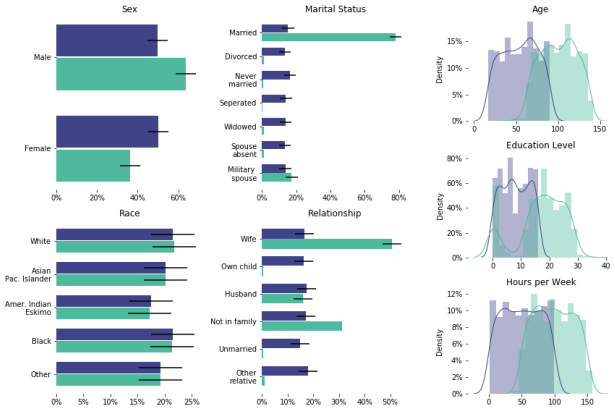

Figure 2. To analyze if the binary classifier trained on the UCI Adult data set is fair, we assess if the protected features have a uniform distribution in the canonical set. This canonical set was created with a learning rate of 0.1 looping over 200 epochs. To assess if it is balanced w.r.t. the four protected features "sex," "race," "marital status," "relationship," their distribution before and after inverse design is analyzed, respectively represented by the dark purple and light green histograms. The error bars indicate the variance of a uniform distribution with the respective number of categories. We learn that the "race" feature keeps its uniform distribution. "Sex," "marital status," and "relationship" do not keep their uniform distribution after inverse design. This indicates a preference of the model for certain values of these features. Additionally, three numerical features are analyzed: "age," "education level," and "hours per week." All three distributions shift to higher values to achieve a positive output.

4. Interpretation of the Canonical Set of a Binary Classifier

We evaluate the canonical set for a binary classifier. The model consists of three fully-connected hidden layers and a softmax output layer, and achieved an accuracy of 85.1%. The model is trained on the UCI Adult data set (Dua & Graff, 2017). We focus on the legally protected featured encoded in the UCI Adult data set as "sex," "race," "native country," "marital status," and "relationship," respectively.

To analyze if the binary classifier trained on the UCI Adult data set is fair, we assess if the protected features have a uniform distribution in the canonical set. To assess if it is balanced w.r.t. the four protected features "sex," "race," "marital status," "relationship," their distribution before and after inverse design is analyzed, respectively represented by the dark purple and light green histograms in Fig. 2. We learn that the "race" feature keeps its uniform distribution. "Sex," "marital status," and "relationship" do not keep their uniform distribution after inverse design. This indicates a preference of the model for certain values of these features. Additionally, three numerical features are analyzed: "age," "education level," and "hours per week." All three distributions shift to higher values to achieve a positive output.

Two well-known traditional output-based notions of assessing fairness based on group membership are Statistical Parity and Equal Opportunity (Makhlouf et al., 2021). Statistical Parity holds when all subpopulations have an equal Positivity Rate (PR). This means that the same proportion of each subpopulation receives a positive output. Equal Opportunity holds when all subpopulations have an equal True Positive Rate (TPR). This implies that for each subpopulation the same rate of people who should receive a favorable output also receive this output. We compare the canonical set with the results from these traditional fairness metrics calculated on the test data set. See Table 1 for the Positivity Rates and the True Positive Rates of the respective subpopulations of the four protected features. For all protected features there is unfairness between certain subpopulations, including the 'race' feature. For the other features there are subtle differences between the subpopulations that receive the highest TP and TPR, and the preferred subpopulations in the canonical set. Note that the statistical metrics can only be evaluated using a ground truth benchmark and they do not consider the internal dynamics of the model.

References

Agarwal, S., Jabbari, S., Agarwal, C., Upadhyay, S., Wu, S., and Lakkaraju, H. Towards the unification and robustness of perturbation and gradient based explanations. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 110–119. PMLR, 18–24 Jul 2021.

AIHLEG. Ethics guidelines for trustworthy ai, 2019.

Amrit, C., Paauw, T., Aly, R., and Lavric, M. Identifying child abuse through text mining and machine learning. *Expert systems with applications*, 88:402–418, 2017.

Barocas, S. and Selbst, A. D. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

Barocas, S., Hardt, M., and Narayanan, A. *Fairness and Machine Learning*. fairmlbook.org, 2019. http://www.fairmlbook.org.

Binns, R. Fairness in machine learning: Lessons from political philosophy. In *Conference on Fairness, Accountability and Transparency*, pp. 149–159. PMLR, 2018.

Binns, R. On the apparent conflict between individual and group fairness. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 514–524, 2020.

Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pp. 77–91. PMLR, 2018.

Chouldechova, A. and Roth, A. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*, 2018.

Crenshaw, K. Mapping the margins: Intersectionality, identity politics, and violence against women of color. *Stan. L. Rev.*, 43:1241, 1990.

Das, A. and Rad, P. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*, 2020.

Datta, A., Sen, S., and Zick, Y. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 598–617, 2016. doi: 10.1109/SP.2016.42.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Fazelpour, S. and Danks, D. Algorithmic bias: Senses, sources, solutions. *Philosophy Compass*, 16(8):e12760, 2021.

Ferruz, N. and Höcker, B. Dreaming ideal protein structures. *Nature Biotechnology*, pp. 1–2, 2022.

Forte, A. E., Hanakata, P. Z., Jin, L., Zari, E., Zareei, A., Fernandes, M. C., Sumner, L., Alvarez, J., and Bertoldi, K. Inverse design of inflatable soft membranes through machine learning. *Advanced Functional Materials*, pp. 2111610, 2022.

Gallie, W. B. *Essentially contested concepts*. Cornell University Press, 2019.

Gohel, P., Singh, P., and Mohanty, M. Explainable ai: current status and future directions. *arXiv preprint arXiv:2107.07045*, 2021.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

Kleinberg, J. and Mullainathan, S. Simplicity Creates Inequity: Implications for Fairness, Stereotypes, and Interpretability. New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367929.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Ledford, H. Millions of black people affected by racial bias in health-care algorithms. *Nature*, 574(7780):608–610, 2019.

Lee, M. S. A. and Singh, J. The landscape and gaps in open source fairness toolkits. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, pp. 1–13, 2021a.

Lee, M. S. A. and Singh, J. Risk identification questionnaire for detecting unintended bias in the machine learning development lifecycle. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 704–714, 2021b.

Lenaerts, J., Pinson, H., and Ginis, V. Artificial neural networks for inverse design of resonant nanophotonic components with oscillatory loss landscapes. *Nanophotonics*, 10(1):385–392, 2021.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017.

Makhlouf, K., Zhioua, S., and Palamidessi, C. On the applicability of machine learning fairness notions. *ACM SIGKDD Explorations Newsletter*, 23(1):14–23, 2021.

Meng, C., Trinh, L., Xu, N., Enouen, J., and Liu, Y. Interpretability and fairness evaluation of deep learning models on MIMIC-IV dataset. *Scientific Reports*, 12 (7166), 2022.

Mordvintsev, A., Olah, C., and Tyka, S. M. Inceptionism: Going deeper into neural networks - google ai blog, 2015. URL https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html.

Nielsen, M. A. *Neural networks and deep learning*, volume 25. Determination press San Francisco, CA, 2015.

Northcutt, C. G., Athalye, A., and Mueller, J. Pervasive label errors in test sets destabilize machine learning benchmarks. *arXiv preprint arXiv:2103.14749*, 2021.

Ren, S., Padilla, W., and Malof, J. Benchmarking deep inverse models over time, and the neural-adjoint method. *Advances in Neural Information Processing Systems*, 33: 38–48, 2020.

Ribeiro, M. T., Singh, S., and Guestrin, C. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *In Workshop at International Conference on Learning Representations*. Citeseer, 2014.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 180–186, 2020.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3319–3328. PMLR, 06–11 Aug 2017.

Suresh, H. and Guttag, J. A framework for understanding sources of harm throughout the machine learning life cycle. In *Equity and Access in Algorithms, Mechanisms, and Optimization*, pp. 1–9. 2021.

Wachter, S. and Mittelstadt, B. A right to reasonable inferences: re-thinking data protection law in the age of big data and ai. *Colum. Bus. L. Rev.*, pp. 494, 2019.

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

Whittaker, M., Crawford, K., Dobbe, R., Fried, G., Kaziunas, E., Mathur, V., West, S. M., Richardson, R., Schultz, J., and Schwartz, O. *AI now report 2018*. AI Now Institute at New York University New York, 2018.

Wijaya, C. Y. 4 categorical encoding concepts to know for data scientists, 2021. URL https://towardsdatascience.com/.

Zafar, M. B., Valera, I., Rogriguez, M. G., and Gummadi, K. P. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pp. 962–970. PMLR, 2017.

Appendix: Design Considerations

We discuss the details of how to implement the inverse design technique to construct the canonical set, including the encoding of features into vectors, the relation between the learning rate and the number of epochs, the initialization of the vectors, and the difference between binary classifiers and risk predictors. Although these technical choices have an influence on the canonical set, the technique itself is agnostic to these choices.

Numerical vs. Categorical Features

Data referring to humans usually contains categorical features such as gender, occupation, and nationality, and numerical features such as age, weight, and income. However, models only process numerical vectors. To feed data to a neural network, the categorical features are encoded. These techniques include 'one-hot encoding' if the number of categories is known when designing the model, 'hash encoding' if the number of categories is not known upfront, 'label encoding' to transform a categorical feature into numerical values (Wijaya, 2021).

Initialization of the Input Vectors

The input layer needs to be initialized with a set of randomly generated vectors. These vectors can be created in multiple ways. Indeed, the features in the training data each satisfy a particular distribution. These distributions might be the result of defective data collection practices, might not represent the distributions of the populations, or reveal the impact of discriminatory practices. To create the initial input vectors, the values of the numerical and categorical features could follow these distributions or they could be uniformly distributed. The latter option ensures an entirely random initialization and also works when the training data set is unknown. See Fig. 3 for the distribution of the initialized vectors (in dark purple) when their numerical and categorical features are generated according to a uniform distribution.

The Preferred Output

A canonical input can be created for each possible output. In the case of a binary decision, both output carry information. The positive output results in a benefit or advantage for the individual. This canonical set of this output tells us which features positively impact the decision-making process. The negative output results in a disadvantage or punishment. This corresponding canonical set tells us which features have a negative impact on the decision. We focus on the positive output set for the rest of the paper.

Evolution of Numerical Features

In practice, numerical characteristics have lower and upper limits. For example, the age range currently goes from 0 to 120. However, a smaller range could be considered. The inverse design process is agnostic to the meaning of these values and might update them outside of the real-world range. It is possible to enforce chosen boundaries in each epoch, each couple of epochs, or simply at the end. Enforcing boundaries shift the focus to update other features. This also means that certain information is lost. Therefore, we do not limit the numerical features as the shift of these values contains information.

Recoding the Input Vectors

When a category is one-hot-encoded, the vector includes zeros for the number of possible values. One of these entrees then receives the value one, signaling the value of the feature. However, during the inverse design process, all values in the numerical input vector are updated. This process, therefore, can also update the values on the positions which were initially zero. However, a vector with real values on all positions does not correspond to an actual individual, with only one specific value for each feature. Therefore we need to recode those vectors to correspond to people. For each categorical feature, the highest value related to said feature indicates the value of the category and is indicated as one. All the other positions reset to zero. Note that a part of the information in the vector is now lost. The impact on the predictions of the numerical vectors and the formatted vectors is shown between the first and second row in Fig. 3. Here as well, it is possible to recode the input vector during each epoch, each couple of epochs, or only at the end during the inverse design process. We have chosen to only recode the vectors at the end for this paper.

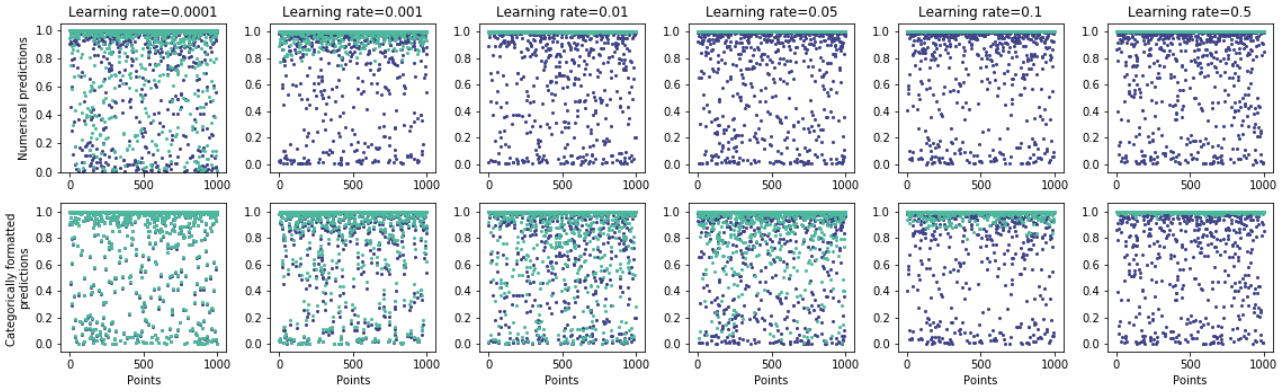

Figure 3. The learning rate parameter of inverse design influences the prediction of the canonical inputs. For six different learning rates between 10^{-4} and 0.5, a thousand vectors are randomly initialized following a uniform distribution. On the first row, the predictions are plotted before and after inverse design, respectively, in dark purple and light green. On the second row, the predictions of the same initial thousand vectors and the canonical inputs after categorical formatting are plotted, again respectively, in dark purple and light green. The canonical inputs receive a higher prediction when the learning rate is higher. After categorical formatting, the predictions are less high due to a loss of information. Each inverse design has the same amount of epochs, here 200.

Learning Rate and Number of Epochs

The traditional gradient descent method requires the specification of the "learning rate." This parameter determines how much the weights and biases are adjusted in each epoch. During the learning phase, this parameter values typically between 10^{-4} and 10^{-1} (Nielsen, 2015). In creating the canonical set, this parameter indicates how fast the randomly generated input vectors are adjusted. Here, the parameter can typically take higher values.

The aim of creating the canonical set is that all elements strongly activate the preferred output. The learning rate and the number of epochs both have an influence on how fast this is achieved. See Fig. 3 for the evolution of the predictions when the learning rate increases. A sufficient number of epochs is needed to achieve adequate vectors in the canonical set. When the learning rate is lower, the number of epochs needs to be higher to achieve a canonical set with a similar mean loss. However, the categorical features do not change when updating the vector for very small learning rates.

Classifiers and Predictors

Several types of decision-making algorithms learn through gradient descent, including binary classifiers and score predictors. The main difference between these two models is how the output is interpreted. A binary classifier has two output nodes, where an optional soft-max layer already interprets the output values in terms of chances. The threshold for the decision is then at 0.5. For risk prediction with one output node, the decision boundary can be at the discretion of the human interpreting the score. There is not necessarily a fixed threshold. For both types of algorithms, this method works as the threshold, either fixed in the case of a binary classifier or flexible for a score predictor, can be taken into account when defining the ideal input set.