A Various Types of Perturbations

As discussed in Section 2, current literature about robust recourse has inconsistent formulations of adversarial perturbations, especially of perturbations on individual data instances. In this section, we summarise different representations of noise used in the most recent work. We provide the pictorial representations of robust recourse in Figure 3.

Pawelczyk et al. [25] proposed that recourse should be robust to noisy human implementation. To model the noise in human implementation, they perturbed the recourse for a negative data instance by a random variable ϵ which is drawn from a Gaussian probability distribution (i.e., $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$), as shown in Figure 3a. However, they assumed the same magnitude of uncertainty in recourse regardless of how difficult it is to achieve the recourse. Plausibility was not considered either in their work.

Dominguez-Olmedo et al. [7] argued that robust recourse should guide *all* instances in the uncertainty set around $\mathring{x}$ to the same desirable prediction outcome. The uncertainty set (i.e., perturbations) includes all of the plausible data points similar to $\mathring{x}$, as shown in Figure 3b. To model the plausibility of noise, they explicitly took into account the *linear* causal relationships between features when creating the perturbation set, provided that the underlying causal model is known or can be approximated well. As such, the distribution of the noise is re-shaped to accommodate the linear causal model. Unlike Gaussian noise in Figure 3a which follows a probability distribution, the uncertainty set in Figure 3b assumes that the occurrences of instances within this set are equally possible.

Similarly, Virgolin and Fracaros [42] accounted for unforeseen circumstances when generating recourse. They manually defined the perturbation ranges for each feature in their experiment dataset based on their domain knowledge. They further broke down the perturbations into two types: perturbations to features that need to be acted upon when implementing the recourse, and perturbations to features that should remain the same when pursuing the recourse. On an abstract level, the perturbation set can be modelled as a hyper-box (Figure 3c) which contains all possible combinations of feature values within their perturbation ranges. Similar to the causal noise, they also assumed that all instances from this perturbation set are equally possible to happen.

In this work, we model the sub-optimal recourse implementation and simultaneously consider the plausibility of noisy implementation and accumulated uncertainty. To do so, we associate the magnitude of noise with the number of actions and the scale thereof in a recourse. As shown in Figure 3d, we assume that the recourse takes two steps/actions to complete, and each action might be executed imperfectly. Thus, after taking the first action, the intermediate state could land in a position different from the intended place. When determining the likely positions of any intermediate states, we consider the plausibility of their distribution, i.e., they follow plausible noise distribution. We argue that such plausible noise accumulates as the recourse gets longer, and recourse should be robust to such increasing uncertainty. We also implement the idea shown in Figure 3d on a synthetic dataset and compute the invalidation rate for plausible noise added to the recourse of varying length. As shown in Figure 4, each unit of action is perturbed with the same magnitude of plausible noise. Recourse in Figure 4a is shorter so plausible noise is only added once. As a result, its invalidation rate is 0.03. On the other hand, the length of recourse in Figure 4b is double that of in Figure 4a, so plausible noise is added twice – one to each step. As a result, the invalidation rate is 0.12. Note that the actual path in Figure 4b leans towards the bottom-right patch of data points, whereas the actual path in Figure 4a is less affected by this cluster of data.

B Additional Experiments

B.1 Different Hyper-Parameter Values in ROSE

In this section, we report the performance of ROSE under different hyper-parameter values in its recourse generation. Specifically, when generating robust recourse, ROSE uses σ^2 and τ to control the magnitude of noise that recourse should be robust to and the degree of robustness adherence, respectively. Once recourse is generated, we evaluate them using the same magnitude of noise.

B.1.1 One-off Plausible Noise

Figure 5 shows the results of Average Invalidation Rate (AIR) and $\ell 1$ distance when different robustness threshold τ and different variance of plausible noise distribution σ^2 are used in ROSE-one to generate robust recourse. We can empirically observe that a lower AIR is achieved at the cost of $\ell 1$ distance. In other words, the longer the recourse, the more robust the recourse is to one-off plausible noise. Additionally, higher τ and higher σ^2 lead to lower AIR and greater robustness. In these experiments, the efficacy remains 100% on the Adult Income dataset. We also observe a similar trade-off from results on the German Credit and COMPAS datasets. Note that we only experiment with τ up to 0.9, as setting τ greater than 0.9 would further sacrifice efficacy. Our results also support the argument by Pawelczyk et al. [25] that there is a trade-off between cost ($\ell 1$ distance) and robustness to one-off noise.

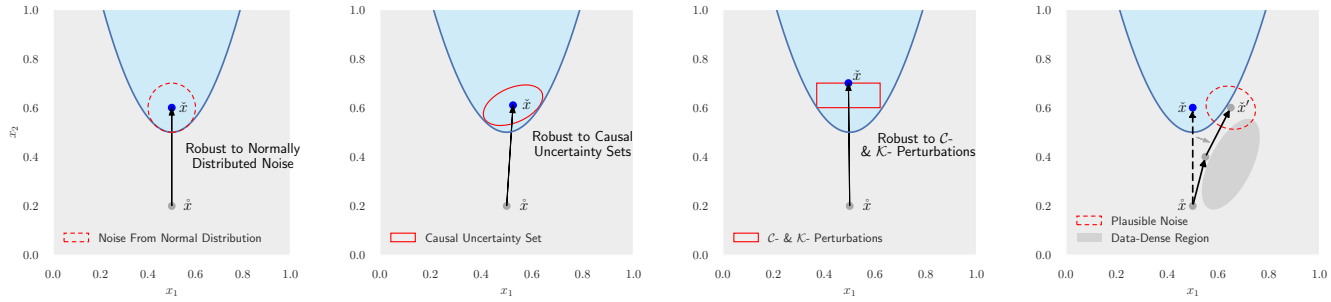
B.1.2 Accumulated Plausible Noise

When ROSE-mul generates recourse that is robust to the accumulated plausible noise, the hyper-parameters that would influence its robustness are σ^2, u and τ. In ROSE-mul, σ^2 is the percentage of the size of one-off noise, which is added to each action unit u. Table 2 shows the performance of ROSE-mul under different σ^2 and u. A bigger σ^2 indicates that, when generating recourse, each action unit is robustified against bigger noise. A bigger u means that fewer actions are involved in the recourse, thus noise is added less frequently. All of the recourse generated under different hyper-parameters are evaluated with the same scale of noise perturbation. In general, as Acc AIR decreases, $\ell 1$ distance of recourse increases. In other words, a higher level of robustness comes at a cost of longer recourse. More robust recourse also requires more computation time. Further, in the German Credit dataset, achieving higher robustness also sacrifices the efficacy of our method. τ has the same effect on ROSE-one and ROSE-mul.

B.2 Different Magnitude of Noise for Evaluation

B.2.1 One-off Plausible Noise

In this section, we report the invalidation rates for different sizes of one-off plausible noise when evaluating all methods, and show the results in Table 3. For all methods across all datasets, as σ^2 gets larger, the invalidation rate increases. ROSE-one or ROSE-mul maintains the lowest invalidation rate across all experiments.



(a) Noise sampled from normal distribution [25].

(b) Noise that follows a causal model [7].

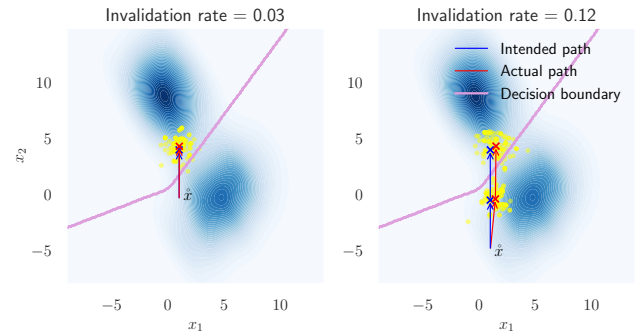
(c) Noise sampled from a manually defined range [42].

(d) Plausible noise adapted to local data geometry (our work).

Figure 3: Various ways to represent human's noisy implementation of algorithmic recourse. In (a) and (d), noise around $\check{x}$ (shown in dotted lines) follows a probability distribution. In (b) and (c), the shape of the noise distribution (shown in solid lines) has to be manually defined. In addition, it is assumed that the occurrence of noise inside the solid region is equally possible regardless of their distance to $\check{x}$.

Table 2: Different σ^2 and u used in ROSE-mul to generate robust recourse. When varying σ^2, we keep $u = 0.025$; when varying u, we keep $\sigma^2 = 5\%$. Results of ROSE-mul in this table are complementary to results in Table 1. When evaluating IR for the accumulated noise, we perturb recourse using the same size of noise set-up – $\sigma^2 = 0.0005, u = 0.025$.

		Metrics				
		Efficacy ↑	Sparsity ↓	$\ell 1$ ↓	Time (s) ↓	Acc AIR ↓
German Credit	$\sigma^2 = 3\%$	0.75	1.82 ± 0.88	0.49 ± 0.32	0.171	0.26 ± 0.20
	$\sigma^2 = 7\%$	0.72	1.84 ± 0.92	0.48 ± 0.34	0.173	0.21 ± 0.15
	$\sigma^2 = 10\%$	0.70	1.83 ± 0.90	0.49 ± 0.25	0.185	0.19 ± 0.18
	$u = 0.05$	0.78	1.71 ± 0.85	0.47 ± 0.33	0.160	0.30 ± 0.19
	$u = 0.10$	0.81	1.71 ± 0.82	0.44 ± 0.32	0.147	0.33 ± 0.28
Adult Income	$\sigma^2 = 3\%$	1.00	1.07 ± 0.26	0.22 ± 0.09	0.179	0.08 ± 0.07
	$\sigma^2 = 7\%$	1.00	1.04 ± 0.22	0.28 ± 0.10	0.196	0.05 ± 0.08
	$\sigma^2 = 10\%$	1.00	1.03 ± 0.17	0.31 ± 0.18	0.239	0.02 ± 0.05
	$u = 0.05$	1.00	1.02 ± 0.14	0.16 ± 0.06	0.120	0.11 ± 0.09
	$u = 0.10$	1.00	1.01 ± 0.10	0.16 ± 0.06	0.113	0.11 ± 0.10
COMPAS	$\sigma^2 = 3\%$	1.00	1.00 ± 0.00	0.04 ± 0.02	0.417	0.03 ± 0.08
	$\sigma^2 = 7\%$	1.00	1.00 ± 0.00	0.04 ± 0.02	0.437	0.01 ± 0.03
	$\sigma^2 = 10\%$	1.00	1.00 ± 0.00	0.04 ± 0.04	0.452	0.01 ± 0.01
	$u = 0.05$	1.00	1.00 ± 0.00	0.04 ± 0.02	0.424	0.04 ± 0.01
	$u = 0.10$	1.00	1.00 ± 0.00	0.04 ± 0.02	0.410	0.05 ± 0.02



(a) Recourse has one step.

(b) Recourse has two steps.

Figure 4: Two recourse with varying length for a factual instance $\mathring{x}$ targeting the top-left desirable class patch. The length of recourse in (b) is double the length of recourse in (a). In (b), the noisy state after the first step is determined by sampling from a plausible noise distribution and use the mean as the new starting state for the subsequent action.

B.2.2 Accumulated Plausible Noise

In this section we report the performance of different methods in terms of different magnitude of accumulated plausible noise. Both the unit size of an action – u – and the variance of noise added to each action unit – σ^2 – influence the magnitude of accumulated noise used for evaluation. Table 4 and Table 5 report the performance by varying either σ^2 or u and fixing the other. Across all datasets, Acc AIR of all methods increases as the value of σ^2 increases (Table 4); Acc AIR decreases as the value of u increases (Table 5). ROSE-mul consistently outperforms other methods in terms of Acc AIR.

B.3 Performance of Additional Baselines

In this section, we report the performance of two other baselines – ROAR and ARAR. ROAR is designed to be robust against model shifts [37]. We explore whether ROAR is also robust to noisy human implementation of recourse. The other baseline, ARAR, is robust to noisy user inputs [7]. However, ARAR is only compatible with logistic regression classifiers; its efficacy is as low as 0.02 when the underlying classifier is a neural network [25]. Therefore, in Table 6, we

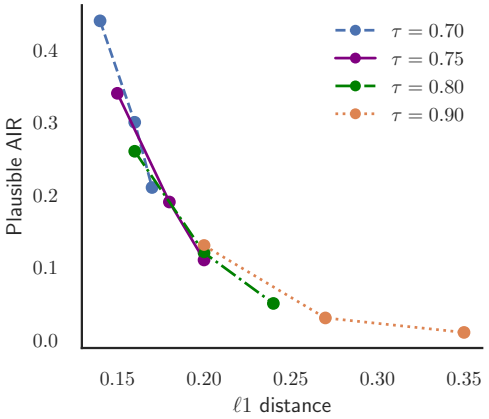


Figure 5: The effects of using different τ and σ^2 when generating robust sequential recourse with ROSE-one for the Adult Income dataset. Plausible AIR is calculated for $\sigma^2 = 0.01$. We experiment with robustness threshold $\tau \in \{0.70, 0.75, 0.80, 0.90\}$; for each τ, we generate robust recourse by setting $\sigma^2 \in \{0.005, 0.01, 0.015\}$.

Table 3: Different σ^2 in one-off plausible noise during evaluation. The configurations of all methods remain the same as the set-ups described in Section 5.1 and Table 1. In this table, we further report Plausible AIR of all methods under $\sigma^2 = \{0.015, 0.01, 0.005\}$.

		Plausible AIR		
	Approach	$\sigma^2 = 0.005$	$\sigma^2 = 0.01$	$\sigma^2 = 0.015$
German Credit	Wachter	0.56 ± 0.11	0.79 ± 0.16	0.80 ± 0.11
	GrSp	0.52 ± 0.11	0.77 ± 0.15	0.78 ± 0.11
	DiCE	0.21 ± 0.21	0.26 ± 0.22	0.32 ± 0.23
	FastAR	0.52 ± 0.10	0.54 ± 0.11	0.55 ± 0.11
	CoGS	0.38 ± 0.09	0.44 ± 0.11	0.47 ± 0.11
	PROBE	0.47 ± 0.04	0.66 ± 0.18	0.68 ± 0.10
	CROCO	0.15 ± 0.14	0.16 ± 0.13	0.25 ± 0.11
	ROSE-one	0.13 ± 0.19	0.22 ± 0.09	0.28 ± 0.08
	ROSE-mul	$\mathbf{0.09 \pm 0.09}$	$\mathbf{0.10 \pm 0.04}$	$\mathbf{0.23 \pm 0.08}$
Adult Income	Wachter	0.73 ± 0.14	0.75 ± 0.15	0.76 ± 0.15
	GrSp	0.65 ± 0.15	0.68 ± 0.16	0.69 ± 0.16
	DiCE	0.19 ± 0.26	0.36 ± 0.30	0.37 ± 0.21
	FastAR	0.68 ± 0.15	0.75 ± 0.14	0.77 ± 0.14
	CoGS	0.70 ± 0.18	0.76 ± 0.17	0.77 ± 0.16
	PROBE	0.30 ± 0.37	0.37 ± 0.40	0.36 ± 0.40
	CROCO	0.38 ± 0.16	0.54 ± 0.28	0.55 ± 0.18
	ROSE-one	$\mathbf{0.05 \pm 0.03}$	$\mathbf{0.19 \pm 0.07}$	$\mathbf{0.30 \pm 0.09}$
	ROSE-mul	0.06 ± 0.05	0.26 ± 0.10	0.31 ± 0.12
COMPAS	Wachter	0.48 ± 0.18	0.79 ± 0.10	0.80 ± 0.18
	GrSp	0.48 ± 0.19	0.75 ± 0.10	0.76 ± 0.06
	DiCE	0.10 ± 0.13	0.21 ± 0.16	0.24 ± 0.18
	FastAR	0.09 ± 0.12	0.15 ± 0.14	0.18 ± 0.13
	CoGS	0.23 ± 0.12	0.29 ± 0.15	0.30 ± 0.14
	PROBE	0.51 ± 0.16	0.73 ± 0.11	0.73 ± 0.17
	CROCO	0.06 ± 0.05	0.12 ± 0.12	0.17 ± 0.10
	ROSE-one	$\mathbf{0.04 \pm 0.05}$	$\mathbf{0.11 \pm 0.08}$	$\mathbf{0.15 \pm 0.09}$
	ROSE-mul	0.05 ± 0.05	$\mathbf{0.11 \pm 0.08}$	0.16 ± 0.09

report the performance of all baselines that are designed to produce recourse that is robust to varying adversarial events. For a fair comparison, and to showcase that ROSE is agnostic to different classifiers (i.e., model-agnostic), we use logistic regression for classification.

B.4 Detailed Comparison with ROAR and ARAR

In this section, we further discuss the performance of ROAR and ARAR reported in Table 6. When the underlying classifier is a linear model, both ROAR and ARAR have good performance in finding valid recourse (i.e., efficacy is 100% or near 100%) in a short time. For ROAR, the invalidation rates of one-off noise and accumulated noise are zero or close to zero across all datasets. ARAR can maintain a low invalidation rate on the COMPAS and German Credit datasets. However, achieving high robustness comes at a cost of high $\ell 1$ distance and high sparsity. Specifically, recourse produced by ARAR requests all users to change every single feature across all datasets; ROAR provides recourse that changes at least half of the feature set. In reality, high sparsity indicates high complexity and more efforts, thus is less preferable.

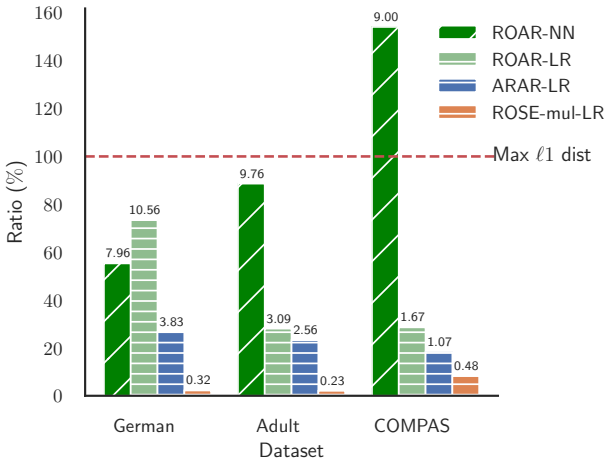


Figure 6: Comparisons of recourse by different robust methods in terms of $\ell 1$ distance. We report the results of ROAR under both a neural network (NN) and a logistic regression (LR) classifier. ARAR does not support NN, so we omit its corresponding result. The distances of recourse by ROSE-one and ROSE-mul under LR and NN are similar, thus we only report one result in this figure. The red dotted line indicates the distance between two further points in each dataset, and the y-axis indicates the ratio of the average recourse distance to the maximum data-points distance. The actual $\ell 1$ distance of recourse is annotated on the top of each bar.

In addition, ARAR provides recourse with up to 10 times higher $\ell 1$ distance than our method ROSE. Distances of recourse by ROAR can be up to 30 times higher than that of ROSE. To visually understand the magnitude of such difference, we compare the average distance of recourse provided by ROAR, ARAR and ROSE against the maximum $\ell 1$ distance between the two furthest data points in a dataset. As shown in Figure 6, under a logistic regression classifier, $\ell 1$ distances in ROAR and ARAR are notably higher than ROSE-mul. We further find out that, if a neural network is used as the classifier, the average distance of recourse by ROAR can be even longer than the distance between two furthest data points in a dataset. This implies that, if users follow recourse by ROAR, they have to move across all data points. In summary, even though ROAR and ARAR are robust to noisy human implementation, their recourse is impractical for users to follow.

It is also worth noting that ARAR is only compatible with linear classifiers. Under non-linear classifiers, its efficacy is as low as 1-2% in each dataset. ROAR supports non-linear classifiers, although under such cases the distance of recourse increases. On the other hand, our method ROSE is model-agnostic, and the performance does not vary significantly across classifiers.

Table 4: Different σ^2 in accumulated plausible noise with fixed action size unit $u = 0.025$. The configurations of all methods remain the same as the set-ups described in Section 5.1 and Table 1, therefore only Acc AIR differs under different σ^2 in accumulated plausible noise. In Table 1, 5% of the size of the one-off noise ($\sigma^2 = 0.0005$) was added to each action unit. In this table, we further report Acc AIR of all methods under $\sigma^2 = \{3\%(0.0003), 5\%(0.0005), 7\%(0.0007), 10\%(0.001)\}$ of the size of one-off noise.

		Acc AIR			
	Approach	$\sigma^2 = 3\%$	$\sigma^2 = 5\%$	$\sigma^2 = 7\%$	$\sigma^2 = 10\%$
German Credit	Wachter	0.55 ± 0.10	0.75 ± 0.17	0.77 ± 0.12	0.78 ± 0.11
	GrSp	0.56 ± 0.13	0.77 ± 0.18	0.78 ± 0.15	0.79 ± 0.13
	DiCE	0.27 ± 0.21	0.32 ± 0.20	0.34 ± 0.21	0.36 ± 0.19
	FASTAR	0.49 ± 0.14	0.51 ± 0.14	0.52 ± 0.14	0.52 ± 0.13
	CoGS	0.37 ± 0.12	0.41 ± 0.14	0.41 ± 0.11	0.44 ± 0.12
	PROBE	0.41 ± 0.16	0.70 ± 0.27	0.73 ± 0.18	0.74 ± 0.16
	CROCO	0.23 ± 0.12	0.25 ± 0.13	0.35 ± 0.07	0.38 ± 0.05
	ROSE-one	0.21 ± 0.14	0.28 ± 0.05	0.31 ± 0.15	0.34 ± 0.14
	ROSE-mul	$\mathbf{0.20 \pm 0.04}$	$\mathbf{0.23 \pm 0.02}$	$\mathbf{0.29 \pm 0.19}$	$\mathbf{0.33 \pm 0.18}$
Adult Income	Wachter	0.65 ± 0.17	0.68 ± 0.18	0.71 ± 0.18	0.72 ± 0.19
	GrSp	0.54 ± 0.17	0.60 ± 0.17	0.62 ± 0.17	0.65 ± 0.16
	DiCE	0.30 ± 0.22	0.45 ± 0.24	0.48 ± 0.25	0.48 ± 0.32
	FASTAR	0.42 ± 0.16	0.51 ± 0.18	0.55 ± 0.17	0.59 ± 0.17
	CoGS	0.46 ± 0.27	0.56 ± 0.27	0.61 ± 0.25	0.65 ± 0.24
	PROBE	0.45 ± 0.17	0.47 ± 0.18	0.49 ± 0.16	0.51 ± 0.16
	CROCO	0.21 ± 0.12	0.26 ± 0.25	0.29 ± 0.07	0.30 ± 0.18
	ROSE-one	0.04 ± 0.07	0.11 ± 0.11	0.19 ± 0.15	0.28 ± 0.17
	ROSE-mul	$\mathbf{0.01 \pm 0.04}$	$\mathbf{0.04 \pm 0.06}$	$\mathbf{0.09 \pm 0.10}$	$\mathbf{0.14 \pm 0.10}$
COMPAS	Wachter	0.47 ± 0.16	0.67 ± 0.16	0.67 ± 0.23	0.69 ± 0.12
	GrSp	0.45 ± 0.13	0.50 ± 0.16	0.53 ± 0.17	0.54 ± 0.22
	DiCE	0.19 ± 0.12	0.31 ± 0.13	0.34 ± 0.14	0.35 ± 0.13
	FASTAR	0.03 ± 0.01	0.05 ± 0.16	0.05 ± 0.11	0.05 ± 0.11
	CoGS	0.15 ± 0.14	0.22 ± 0.15	0.24 ± 0.14	0.28 ± 0.15
	PROBE	0.29 ± 0.13	0.65 ± 0.16	0.68 ± 0.17	0.69 ± 0.15
	CROCO	0.04 ± 0.05	0.05 ± 0.07	0.06 ± 0.04	0.08 ± 0.03
	ROSE-one	$\mathbf{0.01 \pm 0.02}$	$\mathbf{0.01 \pm 0.03}$	$\mathbf{0.02 \pm 0.04}$	$\mathbf{0.02 \pm 0.05}$
	ROSE-mul	$\mathbf{0.01 \pm 0.04}$	$\mathbf{0.01 \pm 0.03}$	$\mathbf{0.02 \pm 0.04}$	$\mathbf{0.02 \pm 0.04}$

Table 5: Different action unit size u in accumulated plausible noise; σ^2 is fixed to $\sigma^2 = 0.0005$.

		$u = 0.025$		$u = 0.05$		$u = 0.10$	
	Approach	# of steps	Acc AIR	# of steps	Acc AIR	# of steps	Acc AIR
German Credit	Wachter	18.55 ± 12.16	0.75 ± 0.17	9.27 ± 6.08	0.54 ± 0.12	4.64 ± 3.04	0.51 ± 0.11
	GrSp	43.92 ± 31.19	0.77 ± 0.18	21.96 ± 15.59	0.55 ± 0.13	10.98 ± 7.80	0.50 ± 0.12
	DiCE	44.83 ± 16.78	0.32 ± 0.20	22.41 ± 8.39	0.26 ± 0.20	22.21 ± 4.19	0.20 ± 0.18
	FASTAR	17.22 ± 13.61	0.51 ± 0.14	8.61 ± 6.81	0.49 ± 0.14	4.31 ± 3.43	0.45 ± 0.14
	CoGS	19.95 ± 13.44	0.41 ± 0.14	9.98 ± 6.72	0.34 ± 0.13	4.98 ± 3.36	0.26 ± 0.14
	PROBE	55.09 ± 31.91	0.70 ± 0.27	27.55 ± 15.90	0.40 ± 0.18	13.78 ± 7.95	0.26 ± 0.13
	CROCO	49.21 ± 17.37	0.25 ± 0.13	24.61 ± 8.69	0.23 ± 0.11	12.31 ± 4.35	0.19 ± 0.16
	ROSE-one	21.60 ± 13.21	0.28 ± 0.05	10.84 ± 6.61	0.20 ± 0.13	5.42 ± 3.31	0.12 ± 0.11
	ROSE-mul	21.60 ± 12.80	$\mathbf{0.23 \pm 0.02}$	10.82 ± 6.40	$\mathbf{0.19 \pm 0.13}$	5.41 ± 3.21	$\mathbf{0.10 \pm 0.11}$
Adult Income	Wachter	11.48 ± 6.92	0.68 ± 0.18	5.74 ± 3.46	0.63 ± 0.17	2.87 ± 1.73	0.57 ± 0.16
	GrSp	28.15 ± 30.40	0.60 ± 0.17	14.10 ± 15.20	0.53 ± 0.18	7.04 ± 7.60	0.45 ± 0.20
	DiCE	43.82 ± 20.40	0.45 ± 0.24	21.91 ± 10.20	0.39 ± 0.21	10.96 ± 5.10	0.25 ± 0.20
	FASTAR	3.60 ± 2.01	0.51 ± 0.18	1.82 ± 1.03	0.40 ± 0.18	0.93 ± 0.54	0.28 ± 0.17
	CoGS	4.61 ± 2.33	0.56 ± 0.27	2.31 ± 1.16	0.43 ± 0.27	1.15 ± 0.58	0.28 ± 0.24
	PROBE	489.66 ± 265.49	0.47 ± 0.18	244.83 ± 132.74	0.44 ± 0.19	122.41 ± 66.37	0.40 ± 0.20
	CROCO	9.59 ± 4.87	0.26 ± 0.25	4.80 ± 0.44	0.18 ± 0.17	2.41 ± 0.22	0.27 ± 0.13
	ROSE-one	7.20 ± 2.40	0.11 ± 0.11	3.60 ± 1.21	0.03 ± 0.06	1.83 ± 0.63	$\mathbf{0.00 \pm 0.01}$
	ROSE-mul	8.80 ± 4.80	$\mathbf{0.04 \pm 0.06}$	4.42 ± 2.42	$\mathbf{0.01 \pm 0.04}$	2.21 ± 1.21	$\mathbf{0.00 \pm 0.00}$
COMPAS	Wachter	7.41 ± 5.89	0.67 ± 0.16	3.70 ± 2.95	0.47 ± 0.18	1.85 ± 1.47	0.46 ± 0.19
	GrSp	10.08 ± 11.93	0.50 ± 0.16	5.04 ± 5.97	0.51 ± 0.16	2.52 ± 2.98	0.51 ± 0.16
	DiCE	33.76 ± 20.30	0.31 ± 0.13	16.88 ± 10.15	0.26 ± 0.10	8.44 ± 5.08	0.13 ± 0.10
	FASTAR	1.20 ± 0.40	0.05 ± 0.11	0.60 ± 0.20	0.03 ± 0.09	0.30 ± 0.10	0.02 ± 0.08
	CoGS	9.26 ± 4.84	0.22 ± 0.15	4.63 ± 2.42	0.13 ± 0.13	2.31 ± 1.21	0.05 ± 0.08
	PROBE	21.17 ± 12.49	0.65 ± 0.16	10.58 ± 6.24	0.45 ± 0.16	5.29 ± 3.12	0.23 ± 0.15
	CROCO	19.74 ± 8.63	0.05 ± 0.07	9.87 ± 4.32	0.03 ± 0.05	4.94 ± 2.16	0.03 ± 0.08
	ROSE-one	1.60 ± 0.80	$\mathbf{0.01 \pm 0.03}$	0.80 ± 0.40	$\mathbf{0.01 \pm 0.02}$	0.40 ± 0.20	$\mathbf{0.00 \pm 0.00}$
	ROSE-mul	1.60 ± 0.80	$\mathbf{0.01 \pm 0.03}$	0.80 ± 0.40	$\mathbf{0.01 \pm 0.02}$	0.40 ± 0.20	0.01 ± 0.05

Table 6: Comparing ROSE with two additional baselines, ROAR and ARAR, and three previously discussed baselines, CoGS, PROBE and CROCO. The underlying predictor is a logistic regression model. For ROAR, we set learning rate $= 0.1$; For ARAR, we set $\epsilon = 0.01$. The set-ups for the remaining methods are the same as the set-ups described in Table 1.

		Metrics						
	Approach	Efficacy ↑	Sparsity ↓	$\ell 1$ ↓	Time (s) ↓	Gaussian AIR ↓	Plausible AIR ↓	Acc AIR ↓
German	ROAR	**1.00**	20.00 ± 0.00	10.56 ± 1.95	**0.015**	**0.00 ± 0.00**	**0.00 ± 0.00**	**0.00 ± 0.00**
	ARAR	0.99	20.00 ± 0.00	3.83 ± 0.50	0.086	**0.00 ± 0.00**	**0.00 ± 0.00**	0.08 ± 0.08
	CoGS	**1.00**	6.94 ± 1.20	0.36 ± 0.23	1.622	0.39 ± 0.04	0.43 ± 0.07	0.37 ± 0.11
	PROBE	**1.00**	20.00 ± 0.00	1.16 ± 0.65	1.271	0.22 ± 0.03	0.33 ± 0.06	0.38 ± 0.13
	CROCO	0.55	20.00 ± 0.00	1.14 ± 0.40	0.198	0.11 ± 0.05	0.11 ±0.12	0.17 ± 0.08
	ROSE-one	0.65	**1.53 ± 0.79**	0.33 ± 0.35	0.086	0.10 ± 0.06	0.11 ± 0.07	0.14 ± 0.14
	ROSE-mul	0.65	**1.53 ± 0.79**	**0.32 ± 0.33**	0.152	0.09 ± 0.04	0.08 ± 0.06	0.19 ± 0.25
Adult	ROAR	**1.00**	6.00 ± 0.00	3.09 ± 0.72	0.041	**0.00 ± 0.00**	**0.00 ± 0.00**	0.06 ± 0.08
	ARAR	**1.00**	13.00 ± 0.00	2.56 ± 0.78	0.194	0.03 ± 0.01	0.30 ± 0.28	0.53 ± 0.28
	CoGS	**1.00**	4.90 ± 0.33	0.35 ± 0.19	3.864	0.38 ± 0.12	0.79 ± 0.24	0.67 ± 0.24
	PROBE	**1.00**	13.00 ± 0.00	1.64 ± 1.18	2.731	0.31 ± 0.04	0.76 ± 0.19	0.75 ± 0.21
	CROCO	0.98	6.00 ± 0.00	0.43 ± 0.07	0.634	0.08 ± 0.05	0.23 ± 0.08	0.24 ± 0.13
	ROSE-one	**1.00**	**1.01 ± 0.10**	**0.17 ± 0.04**	**0.030**	0.05 ± 0.02	0.17 ± 0.06	0.09 ± 0.08
	ROSE-mul	**1.00**	1.04 ± 0.20	0.23 ± 0.10	0.186	0.05 ± 0.02	0.19 ± 0.08	**0.02 ± 0.03**
COMPAS	ROAR	**1.00**	4.00 ± 4.00	1.67 ± 0.23	**0.015**	**0.00 ± 0.00**	**0.00 ± 0.00**	**0.00 ± 0.01**
	ARAR	**1.00**	7.00 ± 0.00	1.07 ± 0.28	0.072	0.03 ± 0.01	0.07 ± 0.08	0.08 ± 0.09
	CoGS	**1.00**	2.88 ± 0.32	**0.25 ± 0.14**	1.279	0.26 ± 0.02	0.27 ± 0.12	0.19 ± 0.13
	PROBE	**1.00**	7.00 ± 0.00	0.61 ± 0.36	1.231	0.34 ± 0.02	0.32 ± 0.16	0.27 ± 0.15
	CROCO	**1.00**	5.00 ± 0.00	0.54 ± 0.17	0.118	0.13 ± 0.02	0.25 ± 0.17	0.26 ± 0.11
	ROSE-one	**1.00**	**1.00 ± 0.00**	0.30 ± 0.05	0.156	0.11 ± 0.07	0.21 ± 0.05	0.24 ± 0.09
	ROSE-mul	**1.00**	**1.00 ± 0.00**	0.48 ± 0.13	3.113	0.09 ± 0.04	0.21 ± 0.05	0.05 ± 0.10

C Implementation Details

C.1 Datasets

Across all datasets, we use all features for classifier training as well as for recourse generation. We acknowledge that some features might be immutable – e.g., race – and some might be mutable but not actionable – e.g., age; some actionable features could have also actionability constraints. However, the main focus of this work is on robustness. Following the practice in [25], we treat all features as actionable and their feature values as continuous. Future work could be extended to address robustness and actionability simultaneously.

C.2 Details about Classifiers

In Section 5.2, we use the neural network as the underlying classifier to determine the negative instances across all three datasets. Specifically, the neural network is fully connected, with RuLU activation function and a hidden layer of 50 neurons. Further, we use logistic regression as the underlying classifier across all datasets, for experiments presented in Appendix B.3. The training details are shown in Table 7. The performance of classifiers and the number of negative instances for which the recourse is generated are reported in Table 8.

Table 7: Training details for two classifiers on three datasets.

		German	Adult	COMPAS
Batch size	LR	50	512	50
	NN	512	512	512
Epochs	LR	50	50	50
	NN	50	50	50
Learning rate	LR	0.001	0.001	0.001
	NN	0.002	0.002	0.002

Table 8: The first two rows show the accuracy of underlying classifiers. The last two rows show the number of negative instances (# of points) for which the recourse is generated.

		German	Adult	COMPAS
Accuracy	LR	0.78	0.84	0.86
	NN	0.83	0.85	0.86
# of points	LR	157	200	97
	NN	285	200	104

C.3 Implementation Details of Policy Gradient Method in ROSE

In this section, we provide more details about the policy-gradient method discussed in Section 4, which we use to solve our MDP problem. We use the off-the-shelf implementation of the PPO with GAE algorithm [17]. To leverage this implementation of the PPO algorithm, we need to create the environment using the open-source toolkit OpenAI Gym library [4]. The environment is created separately to model each dataset. In the PPO algorithm, both an actor and critic are approximated by neural networks – in our method, we use a fully connected neural network with two two hidden layers and each layer having 64 neurons. For other hyper-parameters in the PPO algorithm, we follow the hyper-parameter set-up used in [41]. For training the PPO algorithm, we also follow their practice by using random data instances from the training set as the starting point – it is argued that this leads to better learning by the RL agent [41].

With the MDP set-up described in Section 4, the total rewards gained while training the PPO algorithm to learn the approximated policy converge fast. For ROSE-one, the one-time training cost for learning policies was 30 minutes for each of the three datasets. For ROSE-mul, the training time was about one hour for each dataset. We use the same CPU machine for training. Since the most time-consuming computation is calculating IR of the accumulated plausible noise, using GPU does not accelerate the computation.