

## 1 A Dataset Descriptions

2 This appendix provides detailed descriptions of the datasets used in our experimental evaluation.  
3 Table 1 summarizes the key characteristics of each dataset, including the number of samples used for  
4 our experiments, number of numerical and categorical features, and number of classes.

Table 1: Dataset characteristics and statistics.

Dataset	Samples	# Feat	# Num	# Cat	Target Class Dist.
Lending Club	30,000	12	8	4	21.8% default
Give Me Some Credit	30,000	9	6	3	6.7% yes
Bank Marketing	30,000	16	7	9	11.3% yes
Credit Default	27,000	23	14	9	22.1% yes
Adult Census	32,000	12	4	8	24.5% $\geq 50K$

### 5 A.1 Lending Club

6 The Lending Club dataset [4] contains detailed information about loans issued through the Lending  
7 Club peer-to-peer lending platform. It includes borrower characteristics (such as credit score, annual  
8 income, employment length), loan specifics (loan amount, interest rate, purpose), and performance  
9 indicators (payment status, delinquency). The binary classification task is to predict whether a loan  
10 will be fully paid or charged off (default). This dataset is particularly relevant for financial counter-  
11 factual explanations as it represents real-world credit risk assessment scenarios where understanding  
12 model decisions is crucial for both borrowers and lenders.

### 13 A.2 Give Me Some Credit

14 The Give Me Some Credit dataset [12] contains anonymized records of credit users with features  
15 such as debt-to-income ratio, number of times delinquent, monthly income, age, and number of open  
16 credit lines. The target variable indicates whether a user experienced a serious delinquency (more  
17 than 90 days overdue) within the previous two years. This dataset provides insight into credit risk  
18 prediction in consumer finance, where counterfactual explanations can offer actionable guidance to  
19 consumers looking to improve their creditworthiness.

### 20 A.3 Bank Marketing

21 The Bank Marketing dataset [6] contains information from a direct marketing campaign conducted by  
22 a Portuguese banking institution. The features include client data (age, job, marital status, education),  
23 campaign contact information (communication type, day, month), economic indicators, and previous  
24 campaign outcomes. The prediction task is to determine whether a client will subscribe to a term  
25 deposit. This dataset represents a real-world marketing scenario where understanding model decisions  
26 can improve campaign efficiency and provide insights for personalized marketing strategies.

### 27 A.4 Credit Default

28 The Credit Default dataset [15] contains information on credit card clients in Taiwan, including  
29 demographic factors, credit data, payment history, and bill statements. The target variable indicates  
30 whether the client defaulted on their payment in the following month. With 23 features (14 numerical  
31 and 9 categorical), this dataset presents complex feature interdependencies common in financial data.  
32 The dataset is valuable for counterfactual explanation research because it represents real-world credit  
33 risk assessment with diverse feature types and nonlinear relationships.

### 34 A.5 Adult

35 The Adult Census dataset [1] contains demographic information extracted from the 1994 U.S. Census  
36 database. Features include age, education, occupation, work hours per week, and capital gain/loss.  
37 The binary classification task is to predict whether an individual’s income exceeds \$50,000 per year.

This dataset is widely used in fairness and explainability research, as it contains sensitive attributes like race, gender, and age, making it valuable for studying how counterfactual explanations handle demographic factors.

## B Evaluation Metrics

We evaluate counterfactual quality using metrics that correspond to key desiderata. For a test set  $\mathcal{X}_{\text{test}}^0 = \{\mathbf{x}_n^0 | h(\mathbf{x}_n^0) = 0\}_{n=1}^N$ , we generate set of  $N$  counterfactuals  $\mathcal{X}'^1 = \{\mathbf{x}_n'^1\}_{n=1}^N$  for class 1 with the evaluated model.

**Validity** measures the success rate of changing model predictions:

$$\text{Validity } (\uparrow) = \frac{N_{\text{val}}}{N} = \frac{1}{N} \sum_{n=1}^N \mathbb{I}(h(\mathbf{x}_n') = 1) \quad (1)$$

**Sparsity** quantifies feature modifications, separately for categorical and numerical features:

$$\text{Sparsity Cat/Num } (\downarrow) = \frac{1}{N_{\text{val}}} \sum_{n=1}^{N_{\text{val}}} \frac{\|\mathbf{x}_{n,\text{cat/num}}^0 - \mathbf{x}_{n,\text{cat/num}}'^1\|_0}{D_{\text{cat/num}}}, \quad (2)$$

where  $\mathbf{x}_{n,\text{cat/num}}^0$  represents  $n$ -th generated counterfactual reduced to categorical numerical attributes.

For continuous features, we use  $\epsilon$ -**sparsity**, counting features modified beyond  $\epsilon \cdot V$  ( $V$  is feature range,  $\epsilon = 0.05$ ):

$$\epsilon\text{-Sparsity cont. } (\downarrow) = \frac{1}{N_{\text{val}}} \sum_{n=1}^{N_{\text{val}}} \frac{1}{D} \sum_{d=1}^D \mathbb{I}(|x_{n,d}^0 - x_{n,d}'^1| > \epsilon \cdot V_g) \quad (3)$$

**Proximity** measures distance to original instances using Gower distance for mixed data and Euclidean for continuous features:

$$\text{Proximity Num } (\downarrow) = \frac{1}{N_{\text{val}}} \sum_{n=1}^{N_{\text{val}}} \|\mathbf{x}_{n,\text{num}}^0 - \mathbf{x}_{n,\text{num}}'^1\|_1 \quad (4)$$

**Predictive Performance** is assessed via **Validity** and **Classif. prob.** (model confidence in target prediction).

**Diversity** is measured by **Hypervol. log scale** [8], which quantifies spread and coverage in objective space.

**Plausibility** is evaluated with **LOF log scale** (isolation from training distribution) and **Log Density** (probability under training data distribution).

Lower values are better for sparsity, proximity, and plausibility metrics ( $\downarrow$ ), while higher values are better for validity, model confidence, and diversity ( $\uparrow$ ).

## C Generative Model Selection

We conducted an ablation study to select the most suitable normalizing flow architecture for our framework, comparing different models based on negative log-likelihood (NLL) on the Adult dataset.

Table 2: Negative log-likelihood comparison of generative models on the Adult dataset. Lower values indicate better performance.

	MAF	NICE	RealNVP	KDE
NLL	<b>-43.2998</b>	26.6644	26.5827	30.5120

We compared three normalizing flow architectures—MAF [10], NICE [2], and Real NVP [3], as well as KDE as a non-parametric baseline. MAF significantly outperformed all alternatives, while NICE and RealNVP showed comparable performance, and KDE exhibited the poorest results.

MAF’s superior performance stems from its autoregressive structure, which enables more expressive transformations by conditioning each dimension on previously transformed dimensions. This property is crucial for accurately modeling the conditional distribution of counterfactual explanations. Based on these results, we selected MAF as the architecture for DiCoFlex, contributing significantly to its ability to generate high-quality, diverse counterfactual explanations.

## D Details of training algorithm

---

### Algorithm 1 Training procedure

---

**Require:** number of steps  $T$ , training examples  $\mathcal{X}$ , classification model  $h(\cdot)$ , prior class distribution  $\pi$ , set of sparsity levels  $\mathcal{P}$ , set of considered masking  $\mathcal{M}$ , number of nearest neighbors  $K$   
Initialize  $\theta_0$   
**for**  $t = 1$  to  $T$  **do**  
    Sample  $\mathbf{x} \sim \mathcal{X}$  and class  $y' \sim \pi \setminus \{h(\mathbf{x})\}$   
    Sample sparsity level  $p \sim \mathcal{P}$  and mask  $m \sim \mathcal{M}$ ;  
    Sample  $K$  counterfactual  $\mathbf{x}' \sim \hat{q}(\mathbf{x}' | \mathbf{x}, y', d_{p,m})$  given by (7) in main paper;  
    Update parameters  $\theta_t$  by optimizing  $\mathcal{Q}$  given by (4) in main paper;  
**end for**  
**return**  $x_0$

---

The training procedure is outlined in Algorithm 1. In each training iteration, a data example  $\mathbf{x}$  is drawn from the dataset  $\mathcal{X}$ , and a target class  $y'$  is sampled from the class distribution excluding the current class of  $\mathbf{x}$ , i.e.,  $\pi \setminus h(\mathbf{x})$ . Subsequently, a sparsity level  $p$  is selected from a predefined set  $\mathcal{P}$ , along with a corresponding masking vector  $\mathbf{m}$  from the set  $\mathcal{M}$ .

Next, the set of  $K$  nearest neighbors, denoted by  $N(\mathbf{x}, y', d_{p,m}, K)$ , is computed using the distance measure  $d_{p,m}$  defined in (9) from the main paper. A counterfactual example  $\mathbf{x}'$  is then sampled from the distribution  $\hat{q}(\mathbf{x}' | \mathbf{x}, y', d_{p,m})$  as defined in (7) from the main paper. Finally, the parameters  $\theta_t$  are updated via a gradient-based optimization procedure aimed at maximizing the objective in (4) from the main paper, with conditioning on both  $p$  and  $\mathbf{m}$ .

## E Limitations of our method

DiCoFlex is deliberately designed for tabular data with mixed feature types, the most common domain for counterfactual explanations in high-stakes decision-making contexts. Although this focus enables strong performance where interpretability is most needed, extending to other data modalities would require architectural adaptations. Our approach exhibits natural trade-offs between competing explanation criteria, favoring diversity and plausibility while maintaining competitive performance in sparsity metrics. The computational efficiency during inference requires an initial training investment, though this one-time cost enables subsequent real-time applications that outperform existing methods by orders of magnitude.

## F Computational Resources

Our experimental framework utilized Python [13] as the primary programming language. Additionally, the open-source machine learning library PyTorch [11] is used to implement DiCoFlex. All experiments were conducted on a GPU cluster equipped with a GeForce RTX 4090 graphics card (24 GB VRAM) and an AMD Ryzen Threadripper PRO 5975WX 32-core processor, with 256 GB of available RAM. These resources provide sufficient computational power and processing speed to meet the requirements of our algorithm.

Table 3: Influence of imposing actionability constraints on the metrics used to evaluate counterfactual explanations.

Model	Classif. prob. $\uparrow$	Proximity cont. $\downarrow$	Sparsity cat. $\downarrow$	$\epsilon$ -sparsity cont. $\downarrow$	LOF log scale $\downarrow$	Hypervol. log scale $\uparrow$
mask 1	0.987	<b>0.496</b>	0.557	<b>0.483</b>	<b>1.881</b>	0.881
mask 2	0.980	0.553	0.555	0.517	2.364	1.703
mask 3	<b>0.993</b>	0.627	0.568	0.513	2.310	2.411
mask 4	0.992	0.517	0.590	0.520	2.327	<b>2.640</b>
unconstrained	0.998	0.581	<b>0.515</b>	0.498	2.156	2.036

## G Additional experimental results

### G.1 Runtime Comparison Across Datasets

Figure 1 displays runtime comparisons between DiCoFlex and baseline methods across all datasets, with time shown on a logarithmic scale (this extends the results from Section 4.5 in the main paper). Both variants of DiCoFlex consistently achieve substantially faster execution times for counterfactual generation, while competing approaches demand processing times that are orders of magnitude longer. This substantial performance advantage stems from fundamental architectural differences. DiCE [7] performs separate optimization procedures for each explanation. CCHVAE [12] requires expensive latent space searching. Similarly, Wachter [14], ReViSE [5] and TABCF [9] rely on iterative or gradient-based optimization procedures that scale poorly with the number of counterfactuals generated. In contrast, DiCoFlex leverages conditional normalizing flows trained solely on labeled data to generate multiple diverse counterfactuals in a single forward pass. By eliminating iterative optimization procedures and model access at inference time, DiCoFlex enables real-time counterfactual generation.

### G.2 Impact of Actionability Constraints

We present the extended results from Section 4.6 in the main paper (see for details). Table 3 presents the evaluation of counterfactual explanations generated by DiCoFlex with different actionability constraints imposed through feature masks. The analysis reveals complex relationships between masking constraints and evaluation metrics. In particular, the application of different masks results in varying impacts across metrics without following a consistent directional pattern.

Mask 1, which prevents modifications to Capital Gain and Capital Loss, achieves the lowest continuous proximity score but exhibits reduced diversity, as indicated by its hypervolume score. In contrast, mask 4 (restricting Sex and Native Country modifications) yields the highest hypervolume while maintaining moderate performance across other metrics. Mask 3 (constraining Age) demonstrates high classification probability but the highest proximity score, indicating a greater deviation from the original instances.

The observed results indicate that actionability constraints introduce complex trade-offs that do not follow simple patterns. The absence of consistent correlations between metrics under different masking configurations suggests that performance characteristics are highly dependent on the specific constraints applied rather than adhering to predictable trade-off relationships.

These findings further validate DiCoFlex flexibility in selecting various user-defined constraints without retraining, while demonstrating that the selection of appropriate constraints should be guided by domain-specific requirements rather than general optimization principles. The mechanism enables the practical customization of counterfactual explanations according to specific application needs, where different feature restrictions may be necessary due to legal, ethical, or practical considerations.

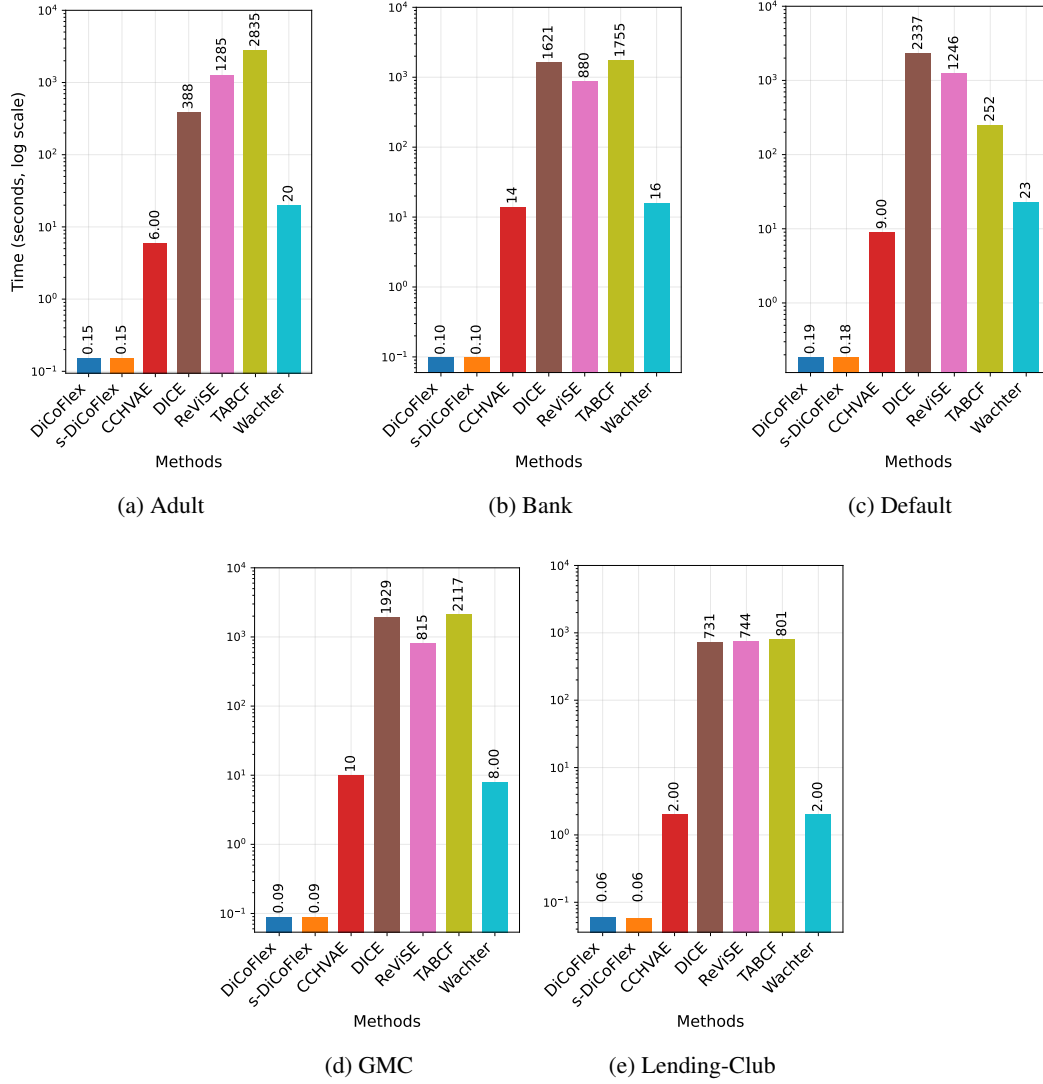


Figure 1: Visualization of runtime of DiCoFlex method and other baseline methods.

## References

- [1] Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: <https://doi.org/10.24432/C5XW20>.
- [2] Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: non-linear independent components estimation. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015.
- [3] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.
- [4] Julapa Jagtiani and Catharine Lemieux. The roles of alternative data and machine learning in fintech lending: evidence from the lendingclub consumer platform. *Financial Management*, 48(4):1009–1029, 2019.
- [5] Shalmali Joshi, Oluwasanmi Koyejo, Warut Vijitbenjaronk, Been Kim, and Joydeep Ghosh. Towards realistic individual recourse and actionable explanations in black-box decision making systems. *arXiv preprint arXiv:1907.09615*, 2019.
- [6] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.
- [7] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 607–617, 2020.
- [8] Krzysztof Nowak, Marcus Mörtens, and Dario Izzo. Empirical performance of the approximation of the least hypervolume contributor. In *Parallel Problem Solving from Nature—PPSN XIII: 13th International Conference, Ljubljana, Slovenia, September 13-17, 2014. Proceedings 13*, pages 662–671. Springer, 2014.
- [9] Emmanouil Panagiotou, Manuel Heurich, Tim Landgraf, and Eirini Ntouts. Tabcf: Counterfactual explanations for tabular data using a transformer-based vae. In *Proceedings of the 5th ACM International Conference on AI in Finance*, pages 274–282, 2024.
- [10] George Papamakarios, Iain Murray, and Theo Pavlakou. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2338–2347, 2017.
- [11] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
- [12] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of the web conference 2020*, pages 3126–3132, 2020.
- [13] Guido Van Rossum and Fred L Drake Jr. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.
- [14] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.
- [15] I-Cheng Yeh. Default of Credit Card Clients. UCI Machine Learning Repository, 2009. DOI: <https://doi.org/10.24432/C55S3H>.