A ridiculously simple approach to counterfactual explanations

Martin Jullum (jullum@nr.no)

NordicXAI, Trondheim, March 29th 2023

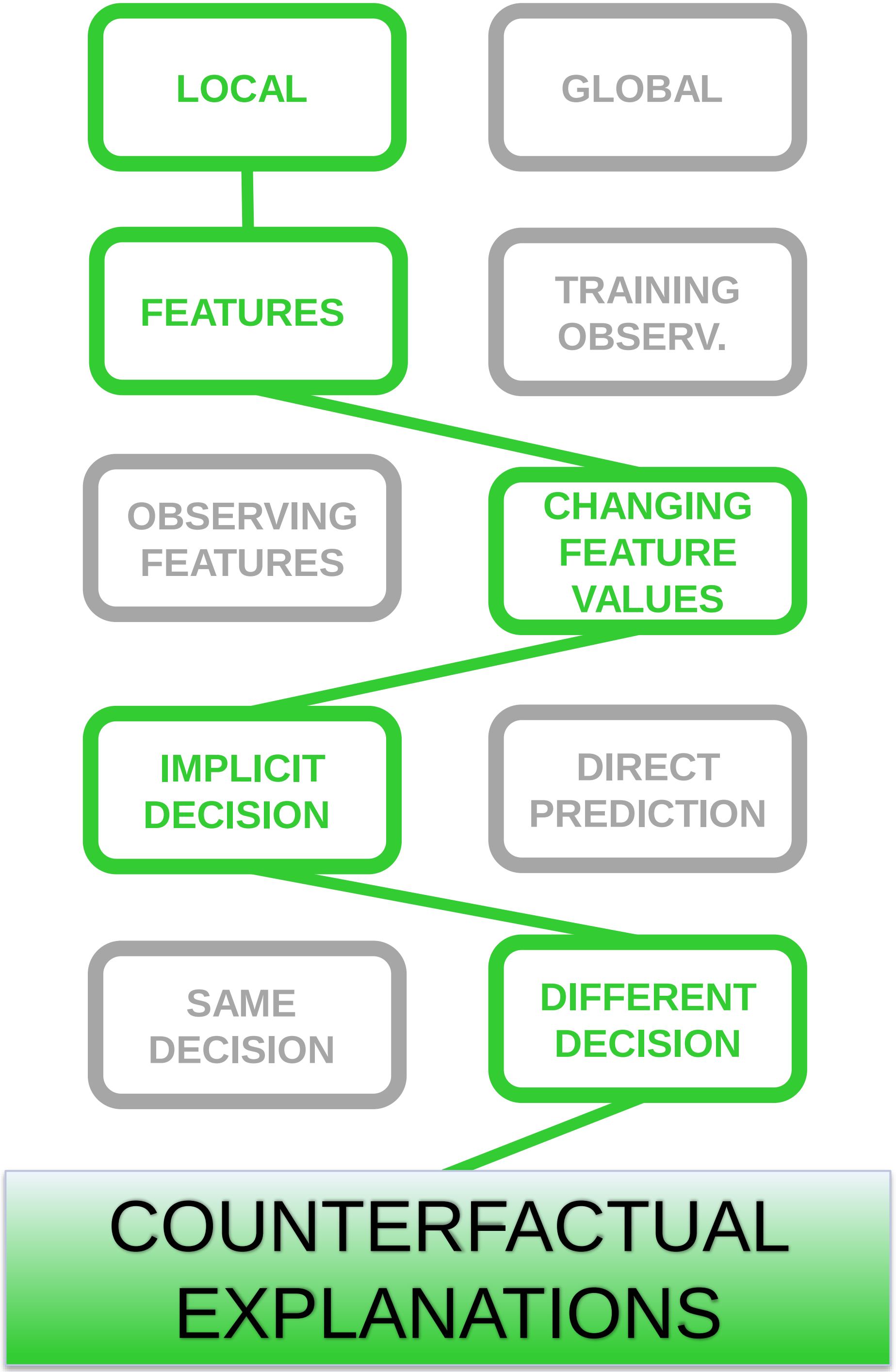
Explanation case

Automatic processing of loaning applications based on default prediction model

- Response y: Loan defaulted or not
- Features $\boldsymbol{x} = (x_1, \ldots, x_p)$: Info about the applicant, salary, previous defaults, transactions history, etc
- Predictive model f: Model trained to predict probability of default: $f(\boldsymbol{x}) \approx \Pr(y = \text{default} | \boldsymbol{x})$
- *Loan approved if* $f(\boldsymbol{x}) < c = 0.1$

CASE: Peter has features $\boldsymbol{x}^*$, and got his loan application rejected as $f(\boldsymbol{x}^*) = 0.2 > c$

Question: What can Peter do to receive a loan?



Explanation case

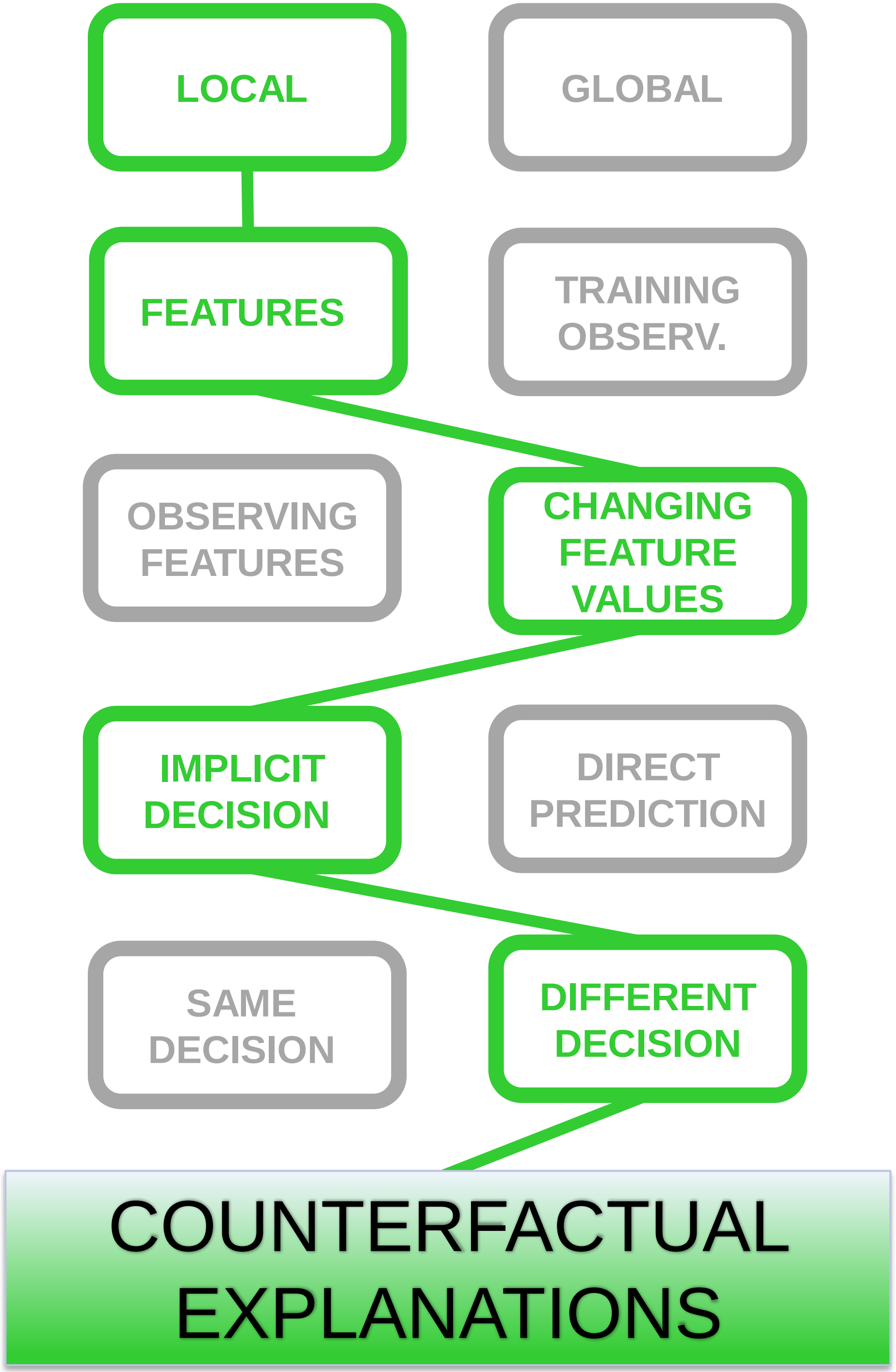
Automatic processing of loaning applications based on default prediction model

Tool for choosing XAI-method (WIP)

xai-tree.nr.no

CASE: Peter has features $\boldsymbol{x}^*$, and got his loan application rejected as $f(\boldsymbol{x}^*) = 0.2 > c$

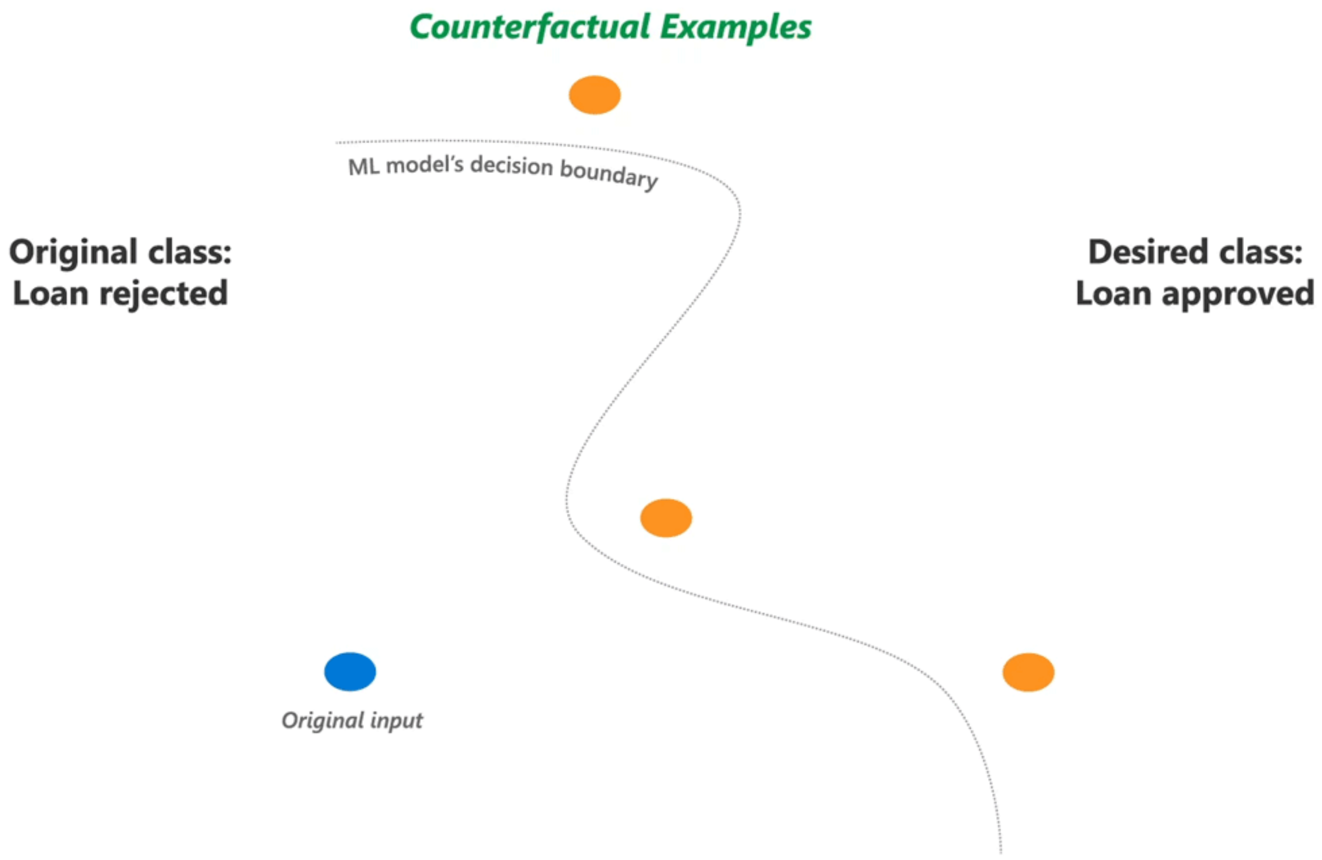
Question: What can Peter do to receive a loan?



Explanation case

CE solution: Examples of (minimal) changes in features which approves the application

Automatic processing of loaning applications based on default prediction model

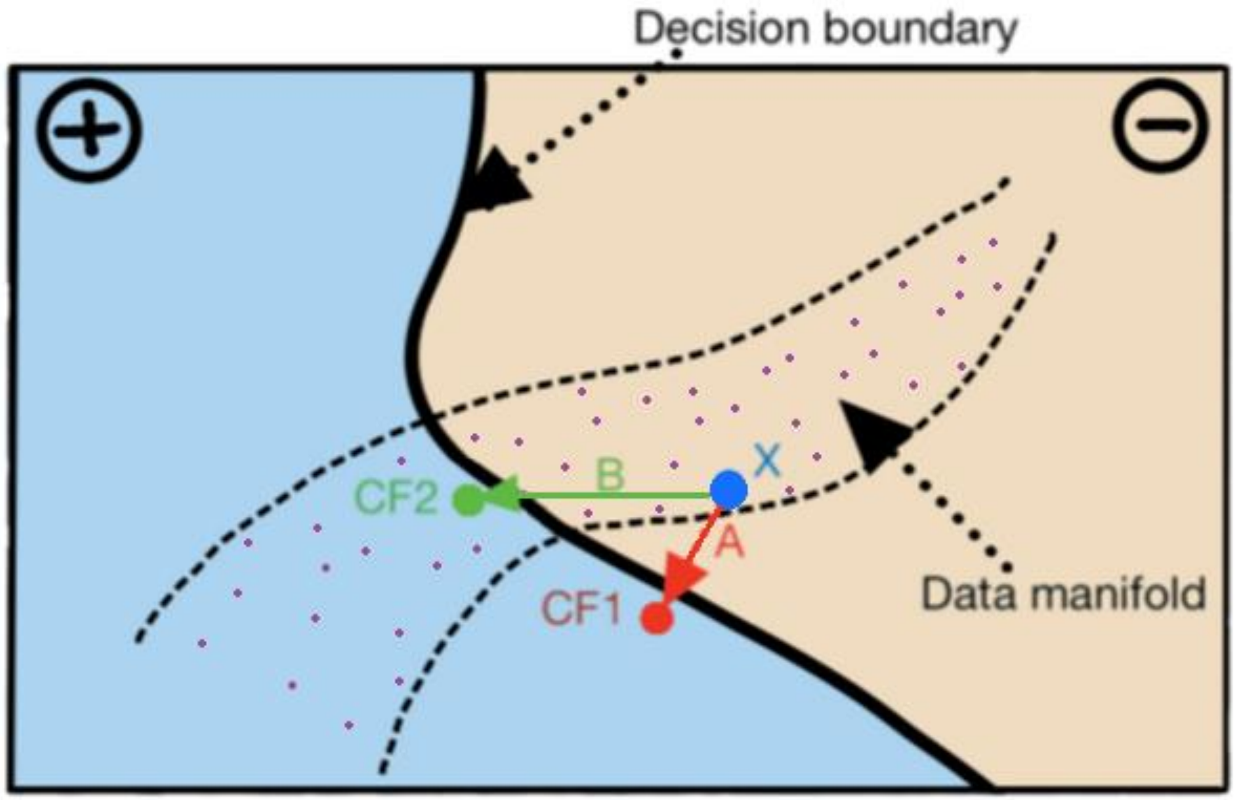


Counterfactual explanations – criteria

e is a CE of $f(x^*)$
Define an acceptable decision interval c_{int}
Divide features into mutable x^m and fixed x^f features

Criteria: e must be

1. On-manifold, i.e. $p\left(X^m = e^m \middle| X^f = e^f\right) > \varepsilon$, for some $\varepsilon > 0$
2. Actionable, i.e. not change fixed features x^f
3. Valid, i.e. $f(e) \in c_{int}$
4. of low cost, i.e. $dist(x^*, e)$ is small



Types of CE methods

Optimization based methods

- Minimize loss functions (wrt ***e***) of type
  - Often require differentiable *f*
  - Not necessarily on-manifold
  - Categorical features more troublesome

$$L_{\boldsymbol{x}^*}(\boldsymbol{e}) = \mathrm{dist}_1(f(\boldsymbol{e}), c) + \lambda \cdot \mathrm{dist}_2(\boldsymbol{x}^*, \boldsymbol{e})$$

Heuristic search-based methods

- Optimization with heuristic search strategies

Instance-based methods

- Finds counterfactuals by searching for instances in a reference distribution/dataset

Our simple CE method: MCCE

MCCE: Monte Carlo sampling of valid and realistic counterfactual explanations

3-step procedure to produce CE $\boldsymbol{e}$ of $f(\boldsymbol{x}^*)$

1. **Model**: Model the distribution of mutable features, given the fixed features and *the decision*
2. **Generate**: Generate a large number K of samples from the modelled distribution with the specified fixed features $\boldsymbol{x}^{*f}$ and desired decision
3. **Post-process**: Discard the invalid samples, and choose the one "nearest" to $\boldsymbol{x}^*$

Walk-through example: Automatic loan

Training data

Features					
Fixed		Mutable			
Age	Sex	Salary	Def. last year	f(x)	Decision
30	M	\$ 3500	yes	0.24	0
28	F	\$ 8000	no	0.12	0
42	M	\$ 7500	no	0.04	1
26	F	\$ 6000	no	0.02	1
27	F	\$ 9500	yes	0.21	0
39	M	\$ 5000	no	0.09	1
28	F	\$ 4000	no	0.08	1
32	F	\$ 7300	no	0.12	0
⋮	⋮	⋮	⋮	⋮	⋮
23	M	\$ 4300	yes	0.31	0

Predictions to explain

Features					
Fixed		Mutable			
Age	Sex	Salary	Def. last year	f(x)	Decision
30	F	\$ 6000	yes	0.18	0
25	M	\$ 4500	no	0.30	0

Step 1: Model

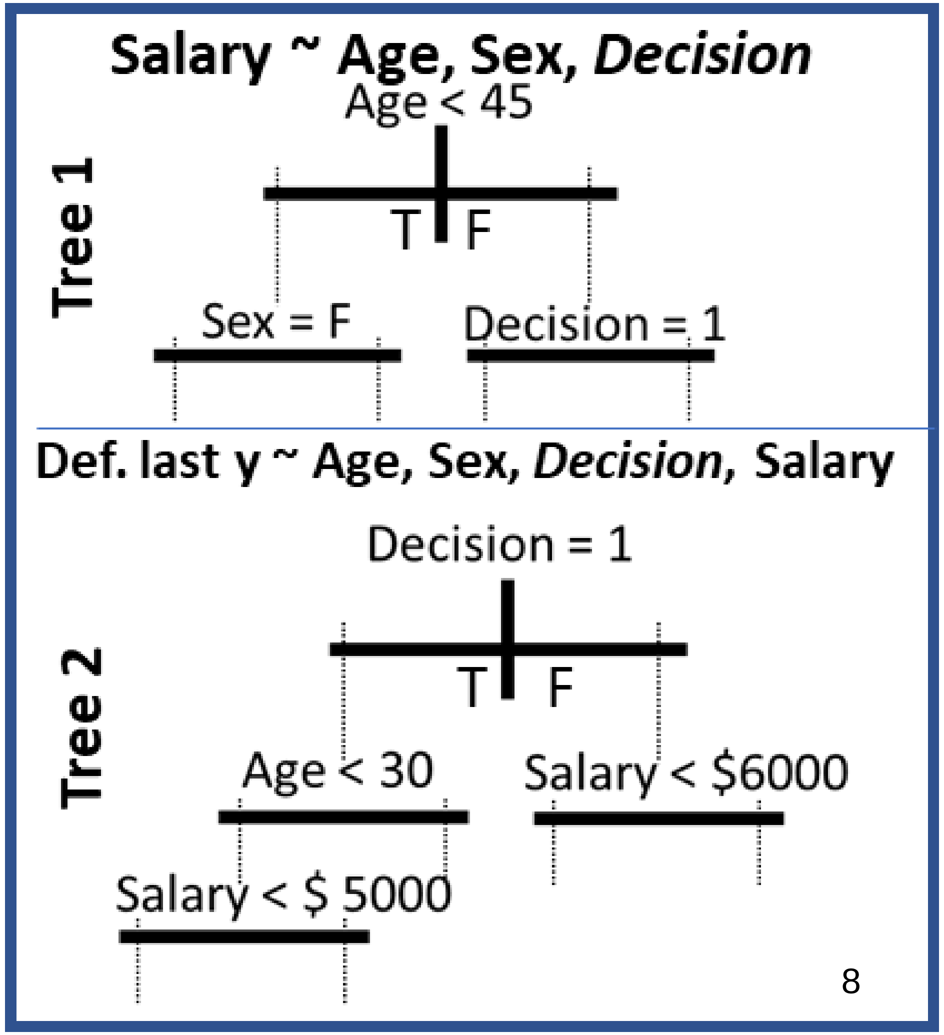
- Denote the decision by $y' = \mathbf{1}\{f(\boldsymbol{x}) \in c_{int}\}$
- We utilize the general property

$$p(\boldsymbol{X}^m \mid \boldsymbol{X}^f, Y') = p(X_1^m \mid \boldsymbol{X}^f, Y') \prod_{i=2}^{q} p(X_i^m \mid \boldsymbol{X}^f, Y', X_1^m, \ldots, X_{i-1}^m)$$

- Use tree models (CART or conditional inference trees) to fit the q distributions $\boldsymbol{X}_i^m \sim (\boldsymbol{X}^f, Y', \boldsymbol{X}_1^m, \ldots, \boldsymbol{X}_{i-1}^m)$, and keep the observations in the end nodes

Training data

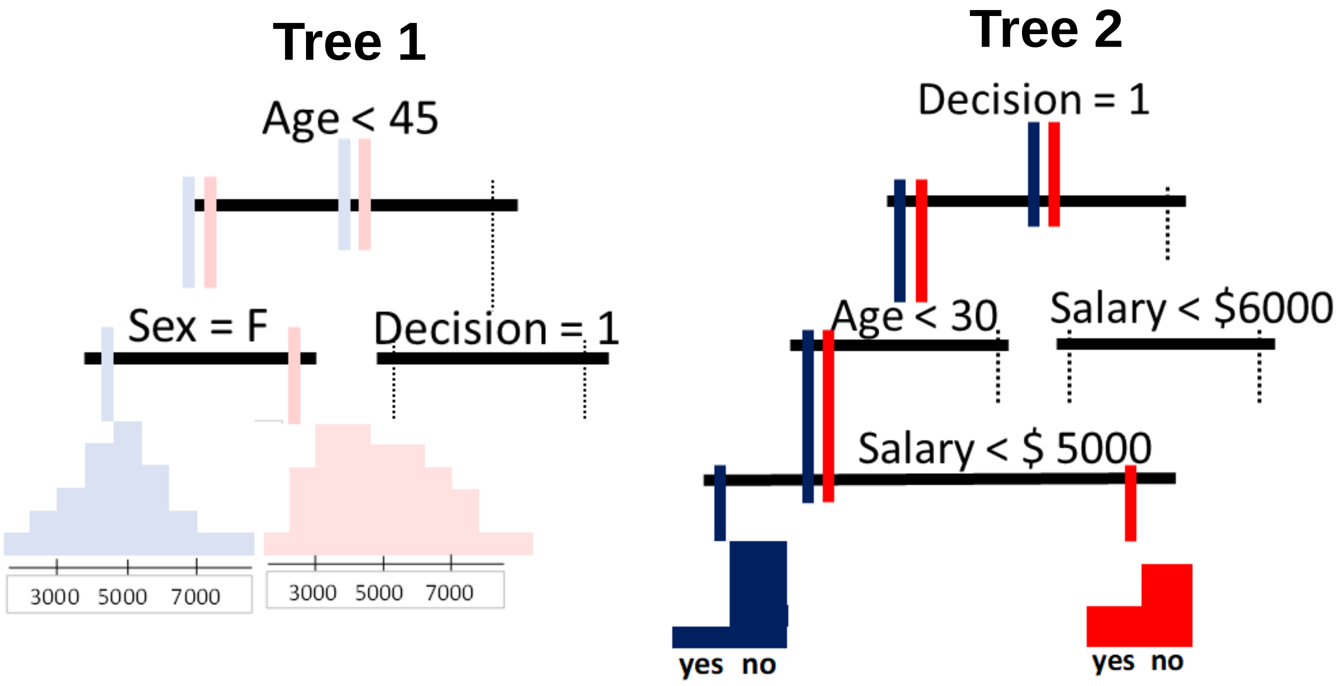
Features					
Fixed		Mutable			
Age	Sex	Salary	Def. last year	f(x)	Decision
30	M	$ 3500	yes	0.24	0
28	F	$ 8000	no	0.12	0
42	M	$ 7500	no	0.04	1
26	F	$ 6000	no	0.02	1
27	F	$ 9500	yes	0.21	0
39	M	$ 5000	no	0.09	1
28	F	$ 4000	no	0.08	1
32	F	$ 7300	no	0.12	0
⋮	⋮	⋮	⋮	⋮	⋮
23	M	$ 4300	yes	0.31	0



Step 2: Generation

For each prediction $f(x^*)$ we want to explain:

- Start with table D with K copies of the fixed features and $y' = 1$
- For each tree: $i = 1, \ldots, q$:
  - For each unique row of D, follow the tree to the end nodes and sample therein
  - Append the samples to the table D as a new column



D

	Age	Sex	Decision	Salary	Def. last year
K	30	F	1	-	-
	30	F	1	-	-
	30	F	1	-	-
	30	F	1	-	-
	25	M	1	-	-
	25	M	1	-	-
	25	M	1	-	-
	25	M	1	-	-

Updating D

Age	Sex	Decision	Salary	Def. last year
30	F	1	$ 4500	-
30	F	1	$ 6000	-
30	F	1	$ 7500	-
30	F	1	$ 3800	-
25	M	1	$ 6000	-
25	M	1	$ 4800	-
25	M	1	$ 5300	-
25	M	1	$ 4600	-

Updating D

Age	Sex	Decision	Salary	Def. last year
30	F	1	$ 4500	no
30	F	1	$ 6000	no
30	F	1	$ 7500	yes
30	F	1	$ 3800	no
25	M	1	$ 6000	yes
25	M	1	$ 4800	no
25	M	1	$ 5300	no
25	M	1	$ 4600	no

Step3: Post-process

Recall

Criteria: e must be

1. On-manifold, i.e. $p\left(X^m = e^m \middle| X^f = e^f\right) > \varepsilon$, for some $\varepsilon > 0$
2. Actionable, i.e. not change fixed features x^f
3. Valid, i.e. $f(e) \in c_{int}$
4. of low cost, i.e. $dist(x^*, e)$ is small

Filter the data set $\boldsymbol{D}$ to obey our four criteria

- ▸ 1 & 2 already satisfied
- ▸ Most samples satisfies 3, remove the others
- ▸ Choose the sample closest to $\boldsymbol{x}^*$ as follows:
  - Per explainee, restrict to smallest number of features being changed (L0)
  - Amongst the remaining, chose the one minimizing the Gower distance

$$\text{Gower distance} = \frac{1}{p}\sum_{j=1}^{p} \delta_G(d_j, x_j) \in [0,1],$$

$$\delta_G(d_j, x_j) = \begin{cases} \frac{1}{R_j} \mid d_j - x_j \mid & \text{if } x_j \text{ is numerical,} \\ \mathbb{1}_{d_j \neq x_j} & \text{if } x_j \text{ is categorical,} \end{cases}$$

Age	Sex	Decision	Salary	Def. last year	f(x)	valid	L0	Gower
~~30~~	~~F~~	~~1~~	~~$ 4500~~	~~no~~	~~0.08~~	~~1~~	~~2~~	0.6
30	F	1	$ 6000	no	0.07	1	1	0.5
~~30~~	~~F~~	~~1~~	~~$ 7500~~	~~yes~~	~~0.09~~	~~1~~	~~1~~	~~0.8~~
~~30~~	~~F~~	~~1~~	~~$ 3800~~	~~no~~	~~0.07~~	~~1~~	~~2~~	0.7
~~25~~	~~M~~	~~1~~	~~$ 6000~~	~~yes~~	~~0.05~~	~~1~~	~~2~~	0.8
25	M	1	$ 4800	no	0.08	1	1	0.3
~~25~~	~~M~~	~~1~~	~~$ 5300~~	~~no~~	~~0.05~~	~~1~~	~~1~~	~~0.5~~
~~25~~	~~M~~	~~1~~	~~$ 4600~~	~~no~~	~~0.12~~	~~0~~	1	0.1

Benchmarks – setup

- Real data sets
- Generate CE to explain predictions from a test set
  - Use MCCE + 6 other on-manifold methods
- Compare the methods in terms of performance measures
  - L0, Gower, feasibility (on-manifoldness), actionability, validity, computation time

Benchmarks – Give me some credit

- Binary classification of financial distress or not
- 10 cont features
- 150 000 obs
- Use 3-layer ANN for modelling

Data set: Give Me Some Credit, $n_{\text{test}} = 1000$, $K = 1000$

Method	L_0 ↓	Gower↓	feasibility↓	actionability↓	validity↑	t(s) all↓
C-CHVAE	8.98 (0.13)	0.95 (0.28)	**0.26** (0.04)	**0.00** (0.00)	**1.00**	151.81
CEM-VAE	8.62 (1.08)	1.61 (0.57)	0.27 (0.07)	0.96 (0.19)	0.93	813.99
CLUE	10.00 (0.04)	1.41 (0.32)	0.37 (0.06)	1.00 (0.03)	**1.00**	3600.35
CRUDS	9.00 (0.00)	1.68 (0.36)	0.42 (0.02)	**0.00** (0.00)	**1.00**	11823.25
FACE	8.59 (1.08)	1.66 (0.53)	0.32 (0.09)	0.98 (0.16)	**1.00**	32308.78
REViSE	8.36 (1.06)	0.70 (0.27)	0.32 (0.05)	**0.00** (0.00)	**1.00**	8286.04
MCCE	**4.52** (0.97)	**0.61** (0.32)	0.27 (0.07)	**0.00** (0.00)	**1.00**	**32.18**

Benchmarks – Adult

- Binary classification of income >= $50 000
- 4 cont + 8 cat features
- 49 000 obs
- Use 3-layer ANN for modelling

Data set: Adult, $n_{\text{test}} = 1000$, $K = 1000$						
Method	$L_0 \downarrow$	*Gower* ↓	feasibility↓	*actionability*↓	*validity* ↑	t(s) all↓
C-CHVAE	7.76 (1.02)	3.13 (1.10)	0.27 (0.17)	**0.00** (0.00)	**1.00**	140.33
CEM-VAE	6.92 (2.06)	3.18 (1.65)	0.21 (0.15)	1.38 (0.59)	0.49	768.76
CLUE	13.00 (0.00)	7.83 (0.31)	0.93 (0.12)	1.36 (0.48)	**1.00**	3578.00
CRUDS	7.87 (1.08)	4.55 (1.09)	1.10 (0.16)	**0.00** (0.00)	**1.00**	15013.56
FACE	6.98 (1.56)	3.3 (1.50)	**0.24** (0.20)	1.42 (0.51)	**1.00**	10280.69
REViSE	5.91 (1.23)	1.62 (1.23)	0.46 (0.33)	**0.00** (0.00)	**1.00**	11806.86
MCCE	**2.70** (0.73)	**0.56** (0.45)	0.32 (0.25)	**0.00** (0.00)	**1.00**	**24.97**

Conclusion

MCCE

- Models both features and the decision to ensure on-manifold and valid CE
- Conditional sampling guarantees to not violate fixed features
- Relies on trees, which handle continuous/discrete/categorical features
- Breaks up tasks into 3 steps – each step can easily be altered to specific needs
- Scalable
- Easy to implement
- Outperforms competing methods in terms of both accuracy and speed

Preprint on arXiv: arxiv.org/abs/2111.09790
R-package, with Python wrapper at github.com/NorskRegnesentral/mcceR