Explainable Cautious Classifiers

Haifei ZHANG

UMR-CNRS 7253 Heudiasyc
Université de technologie de Compiègne

Seminar - LIFAT, March 29, 2024

About me

Educational experience

- November 2020-November 2023, Doctoral degree,
  Université de technologie de Compiègne, France
  - Title: Cautious Explainable Classifiers
  - Supervisors: Benjamin Quost and Marie-Hélène Masson
- September 2019 to September 2020, Master's degree (optimization of complex systems),
  Université de technologie de Compiègne, France
- September 2017 to September 2020, Engineer's degree (AI and data science),
  Université de technologie de Compiègne, France
- September 2014 to July 2017, Bachelor's degree (information engineering),
  Shanghai University, Chine

About me

Actual position: Temporary lecturer and research assistant (ATER) at UTC

Teaching

- Data science (TD)
- Statistical methods for engineering (TP)
- Basic linear algebra (TD)

Research fields

- Machine Learning
- Explainable artificial intelligence
- Uncertainty modeling
- Information fusion

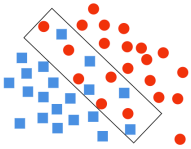
Outline

Uncertainty

Classifier $\boldsymbol{h}$ is trained with pictures of dog and cat

$p(\text{dog} \mid \boldsymbol{x}, \boldsymbol{h}) = 0.51$

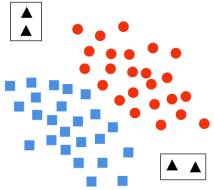
$p(\text{cat} \mid \boldsymbol{x}, \boldsymbol{h}) = 0.49$



Aleatory uncertainty

$p(\text{dog} \mid \boldsymbol{x}, \boldsymbol{h}) = 0.85$

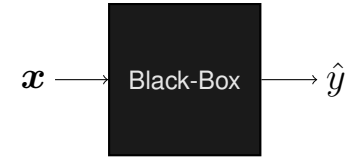
$p(\text{cat} \mid \boldsymbol{x}, \boldsymbol{h}) = 0.15$



Epistemic uncertainty

Determinate predictions may be unreliable

Explainability



We know the prediction $\hat{y}$ is made for $\boldsymbol{x}$

But we don't know

- why the prediction $\hat{y}$ is made
- how to get another desired prediction different from $\hat{y}$

eXplainable AI: we need explanations [1]

[1] A. B. Arrieta, N. Díaz-Rodríguez, D. Ser, *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020.

Objectives

1. Propose a new cautious classifier: cautious random forests

- make indeterminate (set-valued) predictions when the uncertainty is too high
- reduce the risk of making wrong decisions

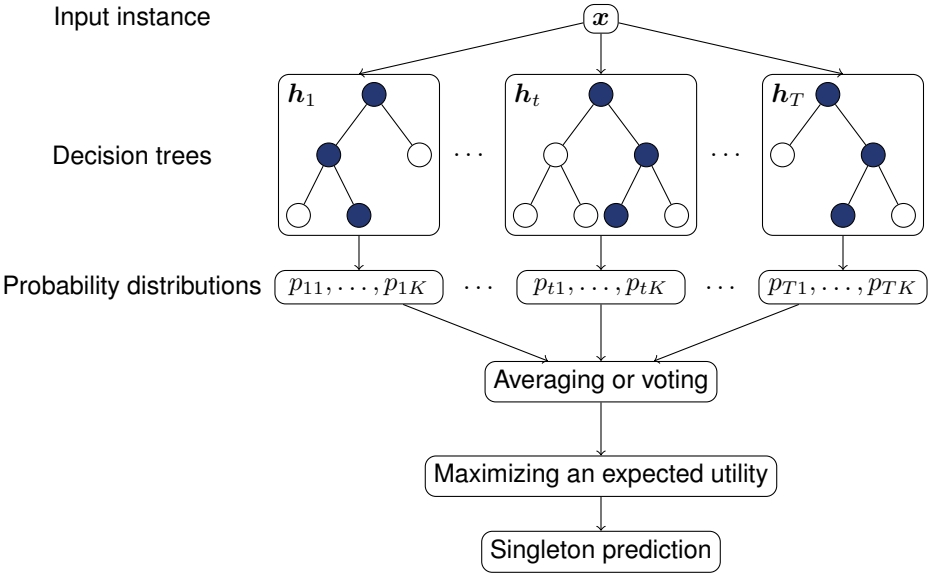
2. Provide explanations for indeterminate predictions: counterfactual examples

- answer the question why indeterminate prediction is made for a given instance
- indicate how to resolve the indeterminacy

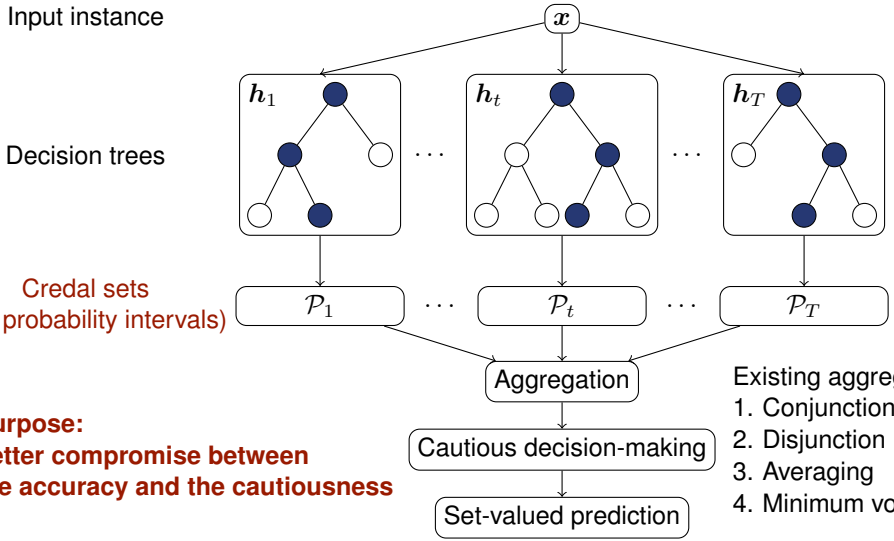
Outline

- Introduction
- **Cautious random forests**
- Resolving indeterminacy via counterfactuals
- Conclusion and research project

Random forest



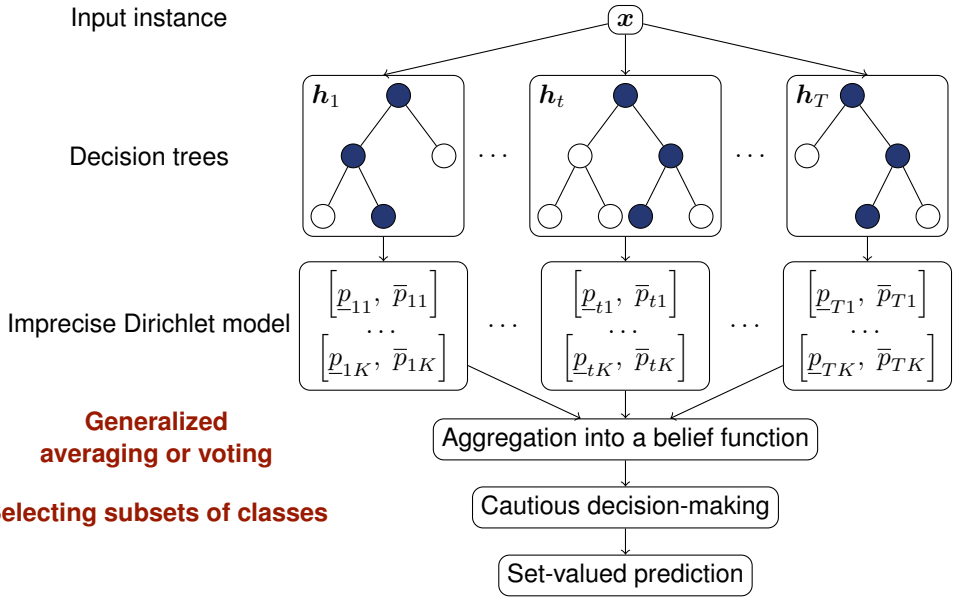
Extend random forest to a cautious one



Purpose:
better compromise between the accuracy and the cautiousness

Existing aggregation methods:
1. Conjunction
2. Disjunction
3. Averaging
4. Minimum vote against

Cautious random forest



Imprecise decision trees

Setting:

- leaf with class counts $(n_1, \ldots, n_k, \ldots, n_K)$
- $N = \sum n_k$: the total number of samples

Imprecise Dirichlet model (IDM) [2]:

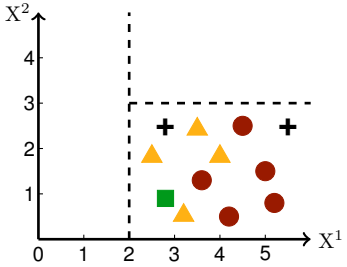
$$\mathcal{I}_k = \left[\underline{p}_k, \overline{p}_k\right] = \left[\frac{n_k}{N+s}, \frac{n_k+s}{N+s}\right], \; k = 1, \ldots, K \quad (1)$$

where s is interpreted as the number of virtual samples

Interval dominance:

retain the set of non-dominated classes

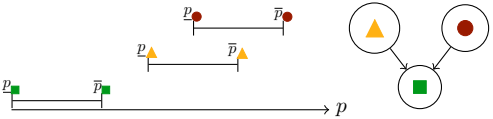
$$\hat{Y} = \{c_k : \nexists\, c_j \in \Omega \text{ s.t. } \underline{p}_j \geq \overline{p}_k\}$$



$$n(\blacksquare) = 1, \; n(\blacktriangle) = 4, \; n(\bullet) = 5$$

$$N = 10, \; s = 2$$

$$\mathcal{I}_{\blacksquare} = \left[\tfrac{1}{12}, \tfrac{1}{4}\right], \mathcal{I}_{\blacktriangle} = \left[\tfrac{1}{3}, \tfrac{1}{2}\right], \mathcal{I}_{\bullet} = \left[\tfrac{5}{12}, \tfrac{7}{12}\right]$$



[2] P. Walley, "Inferences from Multinomial Data: Learning About a Bag of Marbles," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, pp. 3–34, 1996.

Belief functions [3]

- Frame of discernment: $\Omega = \{c_1, \ldots, c_K\}$
- Mass function $m : 2^\Omega \to [0, 1]$, $m(\emptyset) = 0$ and $\sum_{A \subseteq \Omega} m(A) = 1$
- Focal element: $A \subseteq \Omega$ such that $m(A) > 0$
- Belief degree: total support

$$Bel(A) = \sum_{B \subseteq A} m(B), \ \forall A \subseteq \Omega \tag{2}$$

- Plausibility degree: potential support

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B), \ \forall A \subseteq \Omega \tag{3}$$

[3] A. P. Dempster, "Upper and Lower Probabilities Induced by a Multivalued Mapping," *The Annals of Mathematical Statistics*, vol. 38, pp. 325–339, 1967.

Cautious decision-making with belief functions [4]

The lower and upper expected utilities of taking $A \subseteq \Omega$ as prediction:

$$\underline{EU}(m, A, \mathbf{U}) = \sum_{B \subseteq \Omega} m(B) \min_{c_k \in B} u_{Ak} \quad \text{and} \quad \overline{EU}(m, A, \mathbf{U}) = \sum_{B \subseteq \Omega} m(B) \max_{c_k \in B} u_{Ak} \tag{4}$$

where u_{Ak} is the utility when A is taken as prediction and c_k is the actual class.

If 0/1 loss is considered, $\underline{EU}(m, A, \mathbf{U}) = Bel(A)$ and $\overline{EU}(m, A, \mathbf{U}) = Pl(A)$.

[4] T. Denoeux, "Decision-making with belief functions: a review," *International Journal of Approximate Reasoning*, vol. 109, pp. 87–110, 2019.

Cautious decision-making with belief functions [4]

The lower and upper expected utilities of taking $A \subseteq \Omega$ as prediction:

$$\underline{EU}(m, A, \mathbf{U}) = \sum_{B \subseteq \Omega} m(B) \min_{c_k \in B} u_{Ak} \quad \text{and} \quad \overline{EU}(m, A, \mathbf{U}) = \sum_{B \subseteq \Omega} m(B) \max_{c_k \in B} u_{Ak} \tag{4}$$

where u_{Ak} is the utility when A is taken as prediction and c_k is the actual class.

If 0/1 loss is considered, $\underline{EU}(m, A, \mathbf{U}) = Bel(A)$ and $\overline{EU}(m, A, \mathbf{U}) = Pl(A)$.

1. Partial preorder over precise assignments ($c \in \Omega$) via interval dominance:

$$\hat{Y} = \{c_k \colon \nexists\, c_j \in \Omega \text{ s.t. } \underline{EU}(\{c_j\}) \geq \overline{EU}(\{c_k\})\} \tag{5}$$

2. Complete preorder over partial assignments ($A \subseteq \Omega$):

$$\hat{Y} = \arg\max_{A \subseteq \Omega} \underline{EU}(m, A, \mathbf{U}) \tag{6}$$

[4] T. Denoeux, "Decision-making with belief functions: a review," *International Journal of Approximate Reasoning*, vol. 109, pp. 87–110, 2019.

Cautious random forests

Setting

- $\Omega = \{c_1, \ldots, c_K\},\ K > 2$
- Probability intervals: $\{\mathcal{I}_{tk} = \left[\underline{p}_{tk}, \overline{p}_{tk}\right],\ k = 1, \ldots, K,\ t = 1, \ldots, T\}$

Interpretation

Probability intervals provided by each tree are turned into a quasi-Bayesian mass function

$$m_t(\{c_k\}) = \underline{p}_{tk},\ k = 1, \ldots, K, \quad m_t(\Omega) = 1 - \sum_{k=1}^{K} m_t(\{c_k\}) \tag{7}$$

Problems

1. **How to aggregate them into a single mass function?**
2. **How to make cautious predictions based on it?**

Aggregation via generalized averaging and voting

1. Generalized averaging [5]

From averaging probability distributions to averaging mass functions:

$$m(\{c_j\}) = \frac{\sum_{t=1}^{T} m_t(\{c_j\})}{T},\ j = 1, \ldots, K \quad m(\Omega) = \frac{\sum_{t=1}^{T} m_t(\Omega)}{T} \tag{8}$$

The mass function m is still quasi-Bayesian

2. Generalized voting

From voting for a single class to voting for a subset of classes:

$$m(A) = \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}(\mathrm{ID}(m_t) = A), \tag{9}$$

where $\mathrm{ID}(\cdot)$ is the interval dominance that returns the set of non-dominated classes for each tree

The mass function m is no longer quasi-Bayesian

[5] C. K. Murphy, "Combining belief functions when evidence conflicts," *Decision support systems*, vol. 29, no. 1, pp. 1–9, 2000.

Cautious decision-making: maximizing the lower expected utility

Objective:

Consider mass function m and the discounted utility $u_{Aj} = d_\alpha(|A|)\mathbb{1}(c_j \in A)$

where $d_\alpha(|A|) = \frac{1.6}{|A|} - \frac{0.6}{|A|^2}$

Cautious predictions can be made according to $\hat{Y} = \underset{A \subseteq \Omega}{\arg\max}\ \underline{EU}(m, A, \mathbf{U})$

Strategy:

1. **We showed that $\underline{EU}(m, A, \mathbf{U}) = d_\alpha(|A|)Bel(A)$**
2. Consider $A \subseteq \Omega$ with $|A| = i$, $d_\alpha(|A|)$ is a constant
3. Then, $\underset{|A|=i, A \subseteq \Omega}{\arg\max}\ \underline{EU}(m, A, \mathbf{U}) = \underset{|A|=i, A \subseteq \Omega}{\arg\max}\ Bel(A), \quad i = 1, \ldots, |\Omega|$

Search for the maximizer of the $\underline{EU}$ by scanning $A \subseteq \Omega$ with increasing cardinality

Cautious decision-making based on averaging

The averaged mass function is still quasi-Bayesian

Decision-making process:

1. sort classes by decreasing mass: $m(\{c_{(k)}\}) \geq m(\{c_{(k+1)}\})$, for $k = 1, \ldots, K-1$
2. add classes one-by-one to the prediction and calculate the $\underline{EU}$
3. stop when larger predictions can not further improve the $\underline{EU}$

Example:

$$m(\{c_2\}) = 0.48$$
$$m(\{c_1\}) = 0.32$$
$$m(\{c_4\}) = 0.06$$
$$m(\{c_3\}) = 0.04$$
$$m(\Omega) = 0.1$$

$\lvert A\rvert$	A	$d_\alpha(\lvert A\rvert)$	$Bel(A)$	$\underline{EU}(A)$	
1	$\{c_2\}$	1	0.48	0.48	
2	$\boldsymbol{\{c_2, c_1\}}$	0.65	0.8	0.52	Stop
3	$\{c_2, c_1, c_4\}$	0.467	0.86	0.401	

Cautious decision-making based on voting

The mass function obtained by the generalized voting is no longer quasi-Bayesian

Decision-making process:

- maximizing the $\underline{EU}$ requires in principle checking all subsets of Ω
- it is intractable for data with a large number of classes

To reduce the complexity, we introduce three tricks:

1. restrict the prediction size to $M \leq K$
2. stop the process when larger predictions are known not further to improve the $\underline{EU}$
3. for cardinality i, only classes appearing in focal elements B such that $|B| \leq i$ need to be considered

Cautious decision-making based on voting

Let $\Omega = \{c_1, c_2, c_3, c_4\}$, and let the mass function m obtained via generalized voting be defined by

$$m(\{c_1\}) = 0.15, \quad m(\{c_2\}) = 0.25, \quad m(\{c_1, c_2\}) = 0.35,$$
$$m(\{c_1, c_3\}) = 0.05, \quad m(\{c_2, c_3\}) = 0.1, \quad m(\{c_2, c_3, c_4\}) = 0.05, \quad m(\Omega) = 0.05$$

The decision-making process can be done by

Iteration $i = 1$: subset of considered classes $\Omega_1 = \{c_1, c_2\}$

A	$d_{65}(\lvert A\rvert)$	$Bel(A)$	$\underline{\mathbb{E}}_m(A, U)$	$> d_{65}(\lvert A\rvert + 1)$?	Status
$\{c_1\}$	1	0.15	0.15	No (< 0.65)	
$\{c_2\}$	1	0.25	0.25	No (< 0.65)	Continue

Iteration $i = 2$: subset of considered classes $\Omega_2 = \{c_1, c_2, c_3\}$

A	$d_{65}(\lvert A\rvert)$	$Bel(A)$	$\underline{\mathbb{E}}_m(A, U)$	$> d_{65}(\lvert A\rvert + 1)$?	Status
$\{c_1, c_3\}$	0.65	0.20	0.13	No (< 0.4667)	
$\{c_2, c_3\}$	0.65	0.35	0.2275	No (< 0.4667)	
$\{\boldsymbol{c_1, c_2}\}$	0.65	0.75	0.4875	Yes (> 0.4667)	Stop

Example of cautious random forest

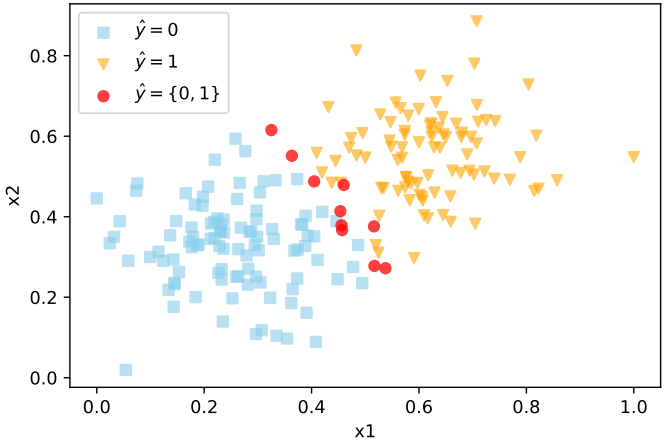


Figure: Cautious predictions made by the cautious random forest

Experiment 1: efficiency of cautious decision-making based on voting

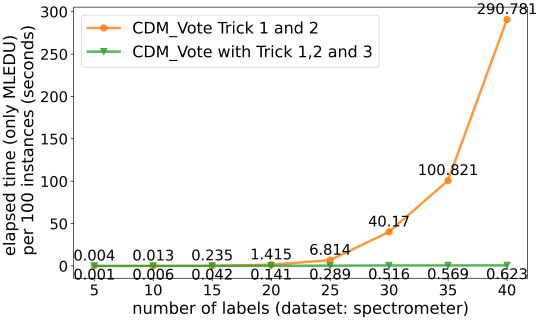


Figure: Decision-making time complexity of cautious decision-making based on voting (CDM_Vote)

Tricks:

1. restrict the prediction size
2. stop when larger predictions can not further to improve the $\underline{EU}$
3. for cardinality i, consider only classes appearing in focal elements B such that $|B| \leq i$

Experiment 2: comparison of different imprecise tree aggregation strategies

Table: Average evaluation results across 12 datasets

Criteria	MVA	AVE	CDM_Vote	CDM_AVE
determinacy	**0.9943**	0.7983	0.8382	0.8351
set size	**2.0272**	4.9464	2.2013	2.2180
single-set accuracy	0.8810	**0.9458**	0.9357	0.9376
set accuracy	0.9328	**0.9610**	0.9185	0.9251
u_{65} score	0.8792	0.8490	0.8794	**0.8798**
u_{80} score	0.8798	0.8712	0.8999	**0.9008**

Compared models:

1. MVA: minimum vote against
2. AVE: averaging with interval dominance

$$u_{65} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1.6}{|\boldsymbol{h}(\boldsymbol{x}_i)|} - \frac{0.6}{|\boldsymbol{h}(\boldsymbol{x}_i)|^2}\right)\mathbb{1}(y_i \in \boldsymbol{h}(\boldsymbol{x}_i))$$

$$u_{65} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{2.2}{|\boldsymbol{h}(\boldsymbol{x}_i)|} - \frac{1,2}{|\boldsymbol{h}(\boldsymbol{x}_i)|^2}\right)\mathbb{1}(y_i \in \boldsymbol{h}(\boldsymbol{x}_i))$$

Open issues:

- the indeterminacy in predictions has a cost
- it needs human intervention to resolve
- assist users in resolving the indeterminacy by providing explanations

Outline

- Introduction
- Cautious random forests
- **Resolving indeterminacy via counterfactuals**
- Conclusion and research project

Counterfactual explanations in XAI [6]

A counterfactual example of the prediction $\hat{y} = \boldsymbol{h}(\boldsymbol{x})$:

the closest instance to $\boldsymbol{x}$ that produces the predefined output $y' \neq \hat{y}$,

$$\boldsymbol{x}' = \underset{\boldsymbol{z} \in \mathcal{X}}{\arg\min}\, d(\boldsymbol{x}, \boldsymbol{z}) \text{ s.t. } \boldsymbol{h}(\boldsymbol{z}) = y' \tag{10}$$

Desiderata of counterfactual examples

- Validity: reach the desired prediction
- Proximity: the smallest possible changes
- Sparsity: only a few changed features
- Actionability: feasible changes for users
- Plausibility: similar to instances in data
- Efficiency: quick generation process

[6] S. Verma, V. Boonsanong, M. Hoang, *et al.*, "Counterfactual explanations and algorithmic recourses for machine learning: A review," *arXiv preprint arXiv:2010.10596*, 2020.

Counterfactual explanations for indeterminate predictions

For instance $\boldsymbol{x} \in \mathcal{X}$ such that $\boldsymbol{h}(\boldsymbol{x}) = \{c_1, c_2\}$, its counterfactual instances $\boldsymbol{x}^{c_1}$ and $\boldsymbol{x}^{c_2}$ are defined as

$$\boldsymbol{x}^{c_1} = \underset{\boldsymbol{z} \in \mathcal{X}}{\arg\min}\, d(\boldsymbol{x}, \boldsymbol{z}) \text{ s.t. } \boldsymbol{h}(\boldsymbol{z}) = \{c_1\} \tag{11a}$$

$$\boldsymbol{x}^{c_2} = \underset{\boldsymbol{z} \in \mathcal{X}}{\arg\min}\, d(\boldsymbol{x}, \boldsymbol{z}) \text{ s.t. } \boldsymbol{h}(\boldsymbol{z}) = \{c_2\} \tag{11b}$$

Counterfactual explanations for indeterminate predictions

For instance $\boldsymbol{x} \in \mathcal{X}$ such that $\boldsymbol{h}(\boldsymbol{x}) = \{c_1, c_2\}$, its counterfactual instances $\boldsymbol{x}^{c_1}$ and $\boldsymbol{x}^{c_2}$ are defined as

$$\boldsymbol{x}^{c_1} = \underset{\boldsymbol{z} \in \mathcal{X}}{\arg\min}\, d(\boldsymbol{x}, \boldsymbol{z}) \text{ s.t. } \boldsymbol{h}(\boldsymbol{z}) = \{c_1\} \tag{11a}$$

$$\boldsymbol{x}^{c_2} = \underset{\boldsymbol{z} \in \mathcal{X}}{\arg\min}\, d(\boldsymbol{x}, \boldsymbol{z}) \text{ s.t. } \boldsymbol{h}(\boldsymbol{z}) = \{c_2\} \tag{11b}$$

Utilities of counterfactual examples:

1. indicate the smallest necessary modifications to obtain a desired prediction
2. identify the closest class to an indeterminate instance
3. reveal the differences between two classes

How to generate counterfactual examples with determinate predictions?

Some existing methods [7]

1. Solving the optimization problem $\underset{\boldsymbol{z}\in\mathcal{X}}{\arg\min}\ \lambda\mathcal{L}(\boldsymbol{h}(\boldsymbol{z}), y') + d(\boldsymbol{x}, \boldsymbol{z})$: only for differentiable models
2. Searching in dataset or leaves: low proximity
3. Surrogate model: low validity
4. Integer linear programming: low efficiency

[7] R. Guidotti, "Counterfactual explanations and how to find them: literature review and benchmarking," *Data Mining and Knowledge Discovery*, pp. 1–55, 2022.

Efficient counterfactual example generation for CRF

Branch-and-bound search algorithm for counterfactual generation [8]:

1. decompose the feature space into decision regions of the classifier
2. start the search from the region containing $\boldsymbol{x}$ and expand it to further regions

[8] P. Blanchart, "An exact counterfactual-example-based approach to tree-ensemble models interpretability," *arXiv preprint arXiv:2105.14820*, 2021.

Efficient counterfactual example generation for CRF

Branch-and-bound search algorithm for counterfactual generation [8]:

1. decompose the feature space into decision regions of the classifier
2. start the search from the region containing $\boldsymbol{x}$ and expand it to further regions

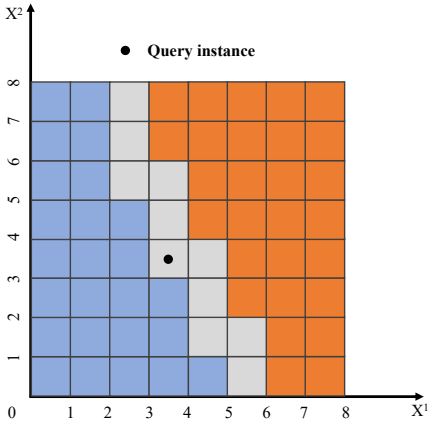
Our contributions:

1. simplification of the feature space decomposition
2. closer initial counterfactual examples to narrow the search regions
3. consideration of actionability and plausibility
4. feature importance to accelerate the counterfactual example generation process

[8] P. Blanchart, "An exact counterfactual-example-based approach to tree-ensemble models interpretability," *arXiv preprint arXiv:2105.14820*, 2021.

Preprocessing: narrow the search region and ensure the actionability

1. **Decomposition of the feature space into elementary decision regions**



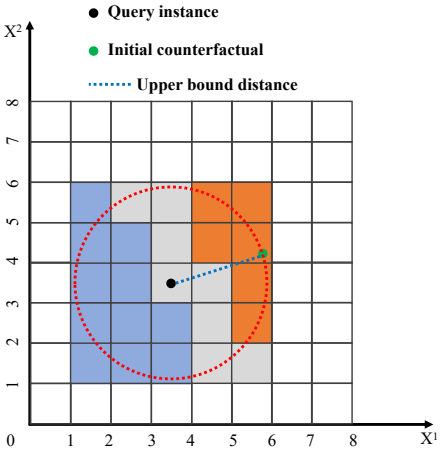
Preprocessing: narrow the search region and ensure the actionability

1. **Decomposition of the feature space into elementary decision regions**

2. **Upper-bound distance**

   $\boldsymbol{x}'$ is the initial counterfactual example for $\boldsymbol{x}$,

$$d_{sup} = d(\boldsymbol{x}, \boldsymbol{x}')$$



Preprocessing: narrow the search region and ensure the actionability

1. Decomposition of the feature space into elementary decision regions

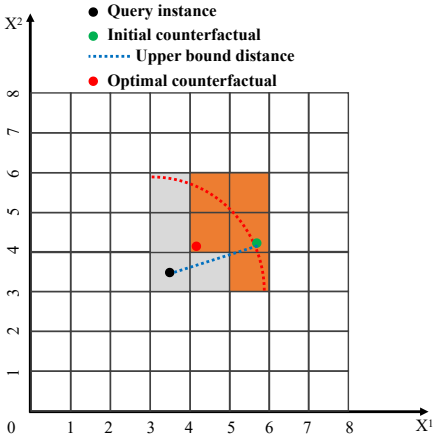
2. Upper-bound distance

$\boldsymbol{x}'$ is the initial counterfactual example for $\boldsymbol{x}$,

$$d_{sup} = d(\boldsymbol{x}, \boldsymbol{x}')$$

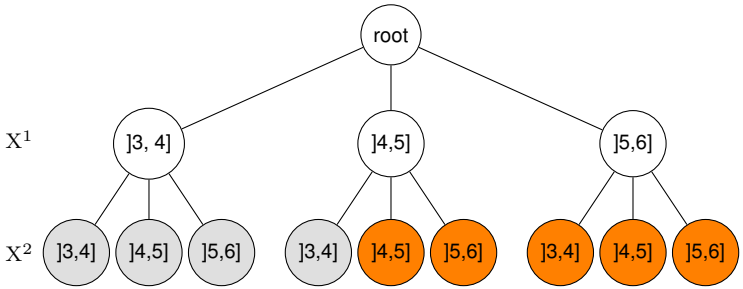
3. Feature constraints

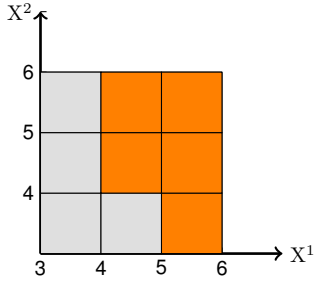
- immutable features
- features can be only increased
- features can be only decreased



Search tree

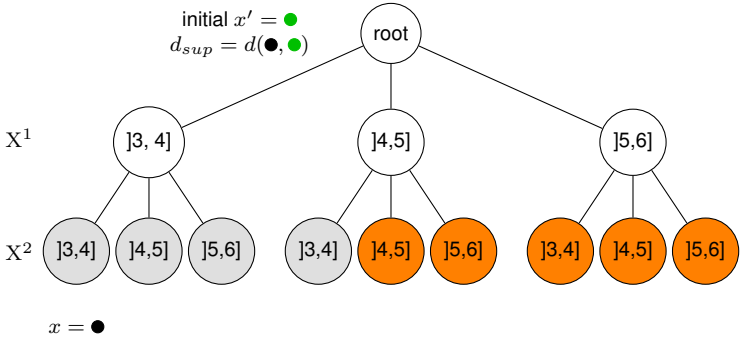
- Each level of the tree: an input feature
- Each node of the tree: a split interval
- Split intervals for each feature are sorted in ascending order according to their distance to x^j
- Each path from the root to a leaf: a decision region

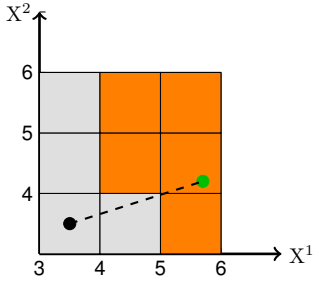




Branch-and-bound search for counterfactual examples

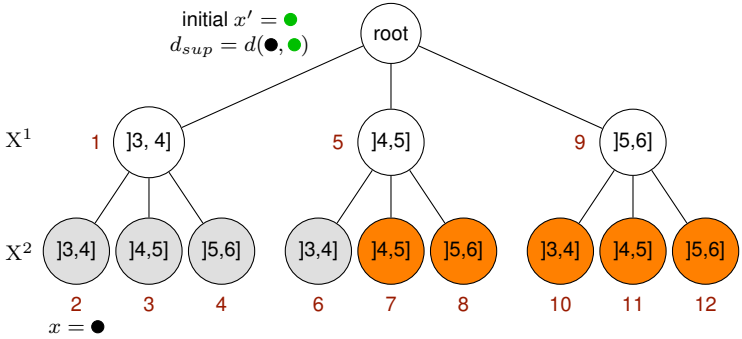
- Conduct a pre-order tree traversal
- In eligible leaves: generate counterfactual examples and update d_{sup}
- Use d_{sup} to skip far regions

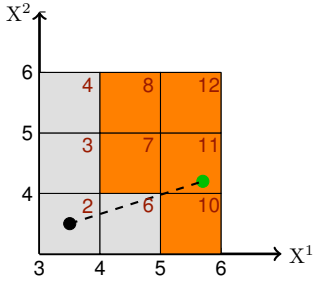




Branch-and-bound search for counterfactual examples

- Conduct a pre-order tree traversal
- In eligible leaves: generate counterfactual examples and update d_{sup}
- Use d_{sup} to skip far regions
- Visit nodes in the order 1, 2, 3...

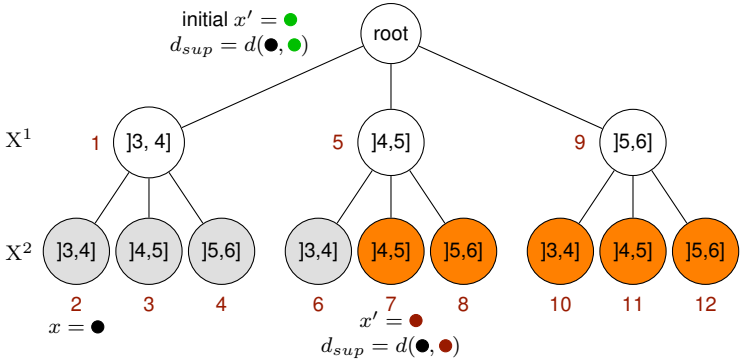


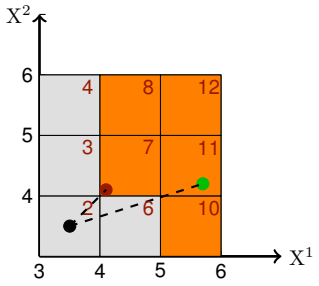


Branch-and-bound search for counterfactual examples

- Conduct a pre-order tree traversal
- In eligible leaves: generate counterfactual examples and update d_{sup}
- Use d_{sup} to skip far regions

- Visit nodes in the order 1, 2, 3...
- Node 7: eligible leaf

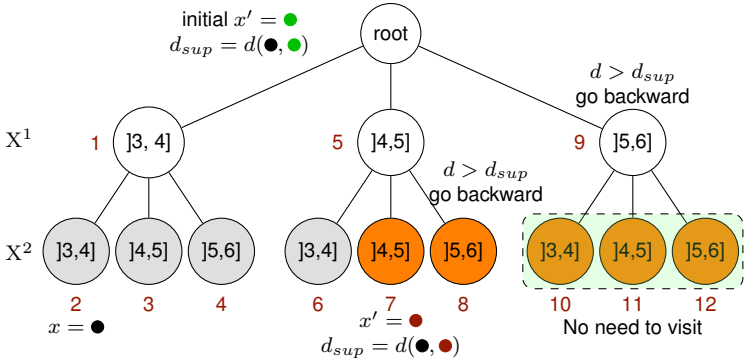


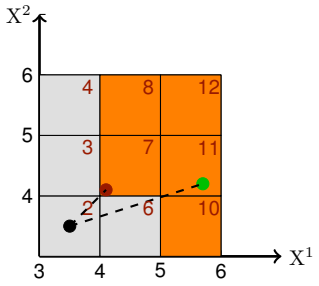


Branch-and-bound search for counterfactual examples

- Conduct a pre-order tree traversal
- In eligible leaves: generate counterfactual examples and update d_{sup}
- Use d_{sup} to skip far regions

- Visit nodes in the order 1, 2, 3...
- Node 7: eligible leaf
- Node 8 and 9: distance larger than d_{sup}





Experiment 5: comparison of different counterfactual generation methods

Compared methods

1. **MO**: Minimum Observable [9]
2. **DisCERN**: Discover Counterfactual Explanations using Relevance Features from Neighbours [10]
3. **OFCC**: One-Feature-Changed Counterfactual [11].

Evaluation metrics

1. **proximity** (L_2 and L_1): average distance between query instances and their counterfactuals
2. **sparsity**: average number of modified features
3. **plausibility**: proportion of non-outlier counterfactual examples detected by the LOF [12]
4. **efficiency**: average elapsed time to generate a counterfactual example

[9] R. R. Fernández, I. M. De Diego, V. Aceña, A. Fernández-Isabel, and J. M. Moguerza, "Random forest explainability using counterfactual sets," *Information Fusion*, vol. 63, pp. 196–207, 2020.

[10] N. Wiratunga, A. Wijekoon, Nkisi-Orji, *et al.*, "Discern: discovering counterfactual explanations using relevance features from neighbourhoods," in *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 1466–1473, IEEE, 2021.

[11] H. Zhang, B. Quost, and M.-H. Masson, "Explaining cautious random forests via counterfactuals," in *Building Bridges between Soft and Statistical Methodologies for Data Science*, pp. 390–397, Springer, 2022.

[12] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, pp. 93–104, 2000.

Experiment 3: comparison of different counterfactual generation methods

Evaluation results

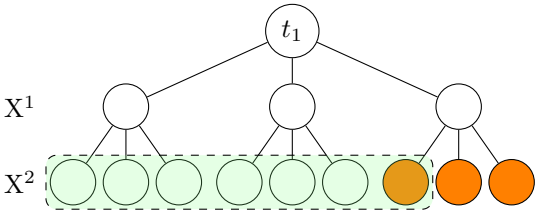
Table: Average evaluation results of generated counterfactuals across 10 datasets

Metrics	MO	DisCERN	OFCC	Ours
$L_2 \downarrow$	8.10	4.05	1.43	**1.00**
$L_1 \downarrow$	18.28	6.10	1.15	**1.00**
Sparsity (opt L_2) $\downarrow$	8.77	3.12	**1.18**	1.90
Sparsity (opt L_1) $\downarrow$	8.19	3.00	**1.17**	1.40
Plausibility $\uparrow$	**0.975**	0.961	0.931	0.934
Elapsed time $\downarrow$	**0.047**	0.087	1.517	3.632

Accelerate the counterfactual generation

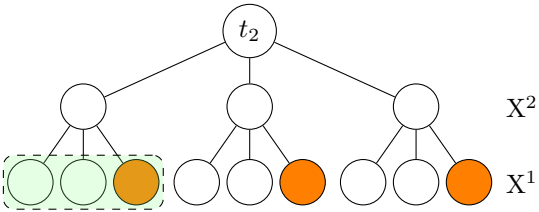
Proposition

- Determine decisive features that must be modified to get the desired prediction
- Put decisive features close to the bottom of the search tree



Orange leaves can generate counterfactuals

X^1 is the decisive feature



Help reducing the d_{sup} more quickly

Increase the chance to skip more regions

Determine decisive features by feature importance

Feature importance: asses the influence of input features on the output of a classifier

Output of cautious random forests:

- prediction uncertainty [13]: $\mathrm{Imp}(\boldsymbol{h}, \boldsymbol{x}) = \min(\overline{p}_1, \overline{p}_2)$
- determinacy

Assessment methods [14]:

Methods	Scope	Used to explain
LIME: Local Interpretable Model-agnostic Explanations	Local	$\mathrm{Imp}(\boldsymbol{h}, \boldsymbol{x})$
SHAP: SHapley Additive exPlanations	Local	$\mathrm{Imp}(\boldsymbol{h}, \boldsymbol{x})$
SHAP-FI: SHAP Feature Importance	Global	$\mathrm{Imp}(\boldsymbol{h}, \boldsymbol{x})$
PFI: Permutation Feature Importance	Global	Determinacy
MDI: Mean Decrease in Impurity	Global	Tree structure

[13] E. Hüllermeier, S. Destercke, and M. H. Shaker, "Quantification of credal uncertainty in machine learning: A critical analysis and empirical comparison," in *Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence*, pp. 548–557, PMLR, 2022.

[14] A. Adadi and M. Berrada, "Peeking inside the black-box: a survey on explainable artificial intelligence (xai)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

Experiment 4: acceleration of counterfactual generation

Table: Impact of feature importance for the acceleration of counterfactual generation, reported in the percentage of the improvement compared with the original feature order

Data	MDI (Global)	PFI (Global)	SHAP-FI (Global)	SHAP (Local)	LIME (Local)
ADLT	**33.02**	28.39	15.37	9.14	-1.89
BIOD	**99.58**	98.53	97.39	29.45	46.27
COMP	**24.22**	23.60	23.60	17.39	-265.22
GERM	**98.75**	98.75	98.75	89.84	86.12
HELO	**49.63**	46.98	45.83	29.25	13.54
LIVR	6.59	7.72	**13.28**	1.91	-5.10
MAMO	**86.96**	82.61	80.43	78.26	23.91
PIMA	**6.83**	6.23	6.13	-13.41	-5.66
SPAM	10.41	3.64	13.76	**22.10**	13.42
WINE	2.80	**4.30**	3.18	-1.69	0.77

Outline

Summary of the research

Cautious random forests

- Generalized averaging and voting
- Efficient maximization of the lower expected utility to make cautious predictions

Resolving indeterminacy

1. Counterfactual examples as explanations for indeterminate predictions
2. Improved branch-and-bound search method for counterfactual generation
3. Feature importance to accelerate counterfactual example generation

Research project: reliable and trustworthy AI

1. Investigate how to improve the **robustness** of different models (machine learning, deep learning) on different types of data (tabular, image, textual, or time series) and develop fusion methods to **handle uncertainty from multiple data sources**.
2. Study how to integrate **domain knowledge** into XAI and investigate how to integrate **domain-relevant causality** when generating post-hoc explanations, with the aim of making these explanations more rational and credible.
3. Explore the applications of reliable artificial intelligence in various fields such as **healthcare, finance and manufacturing**.

Ideas for joining the BdTIn team

1. Strengthen interactive data analysis via explicability and build a **"human in the loop"** paradigm.
2. Combining the team's three axes through explicability: data analysis can collaborate with knowledge representation to generate **structured explanations**, which can then be converted into **textual explanations** understandable by users through natural language processing systems.
3. **Collaborate with other teams in LIFAT** such as RFAI on image analysis and machine learning.

Thanks for your attention!