Operational Research for Fairness, Privacy and Interpretability in Machine Learning

Leveraging ILP to Learn Optimal Fair Rule Lists

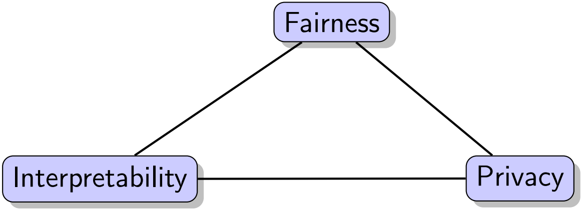
Julien Ferry[1], Ulrich Aïvodji[2], Sébastien Gambs[3], Marie-José Huguet[1] and Mohamed Siala[1]

[1]LAAS-CNRS, Université de Toulouse, CNRS, INSA, Toulouse, France
[2]École de Technologie Supérieure, Montréal, Canada
[3]Université du Québec à Montréal, Montréal, Canada

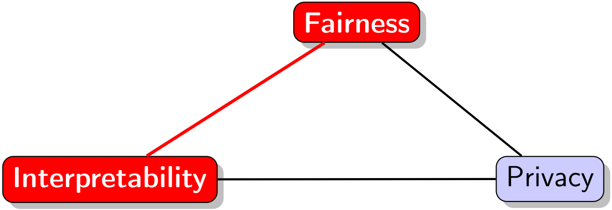
jferry@laas.fr

Introduction: PhD Topics and Research Interests

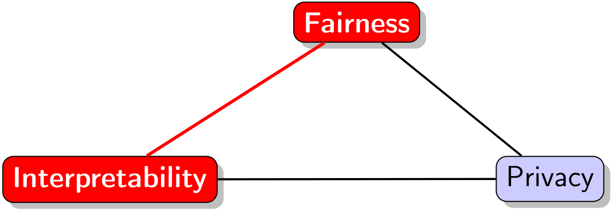
- Academic background: *Ingénieur en Informatique et Réseaux, INSA Toulouse, France*
- PhD student (since 2020) at LAAS-CNRS (Toulouse, France)
- PhD supervisors:
  - Marie-José Huguet (LAAS-CNRS)
  - Sébastien Gambs (UQAM)
  - Mohamed Siala (LAAS-CNRS)
  - Ulrich Aïvodji (ETS Montréal)
- PhD topic: Addressing interpretability, fairness and privacy in machine learning through combinatorial optimization methods



Introduction: PhD Topics and Research Interests

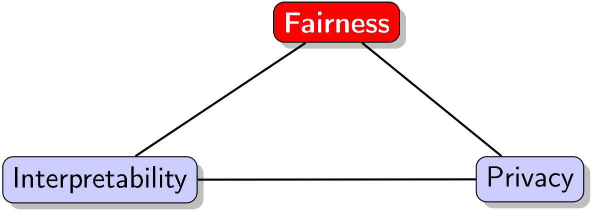


- Incorporating statistical fairness constraints within a supervised learning algorithm producing inherently interpretable models (rule lists)
- Python library: `https://github.com/ferryjul/fairCORELS`
- Preprint: "Learning fair rule lists." @ArXiV [1]
- Conference (Demo) paper: "FairCORELS, an Open-Source Library for Learning Fair Rule Lists." @CIKM '21 (30th ACM International Conference on Information & Knowledge Management) [2]



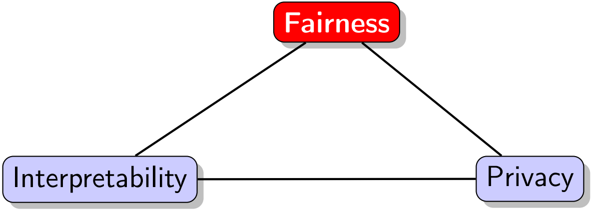
- Leveraging Integer Linear Programming to enhance the exploration of `FairCORELS`' search space by considering jointly accuracy and fairness
- Python library: `https://github.com/ferryjul/fairCORELSV2`
- Conference paper: "Leveraging Integer Linear Programming to Learn Optimal Fair Rule Lists." @CPAIOR '22 (19th International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research) [11]

Introduction: PhD Topics and Research Interests



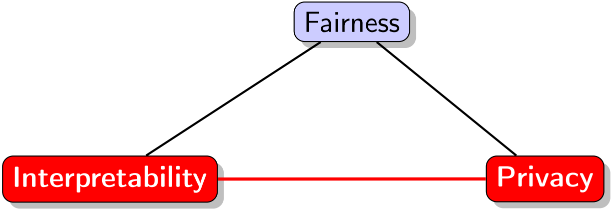
- Improving statistical fairness generalization through a sample-robust optimization method
  - New framework for quantifying fairness robustness from a sampling perspective, inspired by Distributionally Robust Optimization (considering subsets of the training set within a given Jaccard distance)
  - Use of this framework to learn sample-robust fair models
  - Design and use of an heuristic method to efficiently learn sample-robust fair models

Introduction: PhD Topics and Research Interests



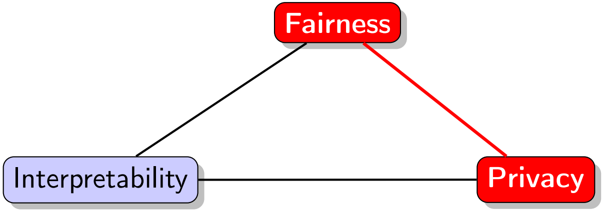
- National conference paper: "Améliorer la généralisation de l'équité en apprentissage grâce à l'Optimisation Distributionnellement Robuste" @RJCIA '21 (Rencontres des Jeunes Chercheurs en Intelligence Artificielle) [9]
- Journal paper: "Improving Fairness Generalization Through a Sample-Robust Optimization Method" @Machine Learning (S.I. on Safe & Fair ML) [10]

Introduction: PhD Topics and Research Interests



- Partially reconstruct a probabilistic dataset, given only access to an interpretable model
- Goal: Quantify (theoretically and empirically) the reconstruction quality, for different hypothesis classes (decision tree, rule list, . . .)

Introduction: PhD Topics and Research Interests



- Leverage black-box access to a fair model to improve training set sensitive attributes reconstruction
- Intuition: Even if they do not use sensitive attributes for inference, fair models are built to respect some fairness constraints over these attributes, hence they inherently learn some information about them (which may be used by an adversary)

Conciliating Fairness and Accuracy in Interpretable ML

Leveraging ILP to Learn Optimal Fair Rule Lists

Julien Ferry[1], Ulrich Aïvodji[2], Sébastien Gambs[3], Marie-José Huguet[1] and Mohamed Siala[1]

[1]LAAS-CNRS, Université de Toulouse, CNRS, INSA, Toulouse, France
[2]École de Technologie Supérieure, Montréal, Canada
[3]Université du Québec à Montréal, Montréal, Canada

jferry@laas.fr

Table of Contents

Problem Notations

- We consider the binary classification task of predicting a binary label $y \in \{0, 1\}$ from a set of attributes $\mathcal{F}$
- Let $\mathcal{E}$ be a dataset and c be a classifier: $c : \mathcal{F} \to \{0, 1\}$
- Dataset $\mathcal{E}$ is partitioned into a set of positively labeled examples $\mathcal{E}^+$ and a set of negatively labeled examples $\mathcal{E}^-$
- Based on the values of some sensitive attributes (*e.g.*, gender, age, race ...), $\mathcal{E}$ is partitioned into a protected group $\mathcal{E}^p$ and an unprotected group $\mathcal{E}^u$ of examples
- We let $TP^c_{\mathcal{E},h}$ ($h \in \{p, u\}$) be the number of true positive examples within $\mathcal{E}^h$, given classifier c's predictions (*e.g.*, the number of examples within $\mathcal{E}^h \cap \mathcal{E}^+$ that are positively classified by c). We similarly define $FP^c_{\mathcal{E},h}$, $TN^c_{\mathcal{E},h}$ and $FN^c_{\mathcal{E},h}$

Group/Statistical Fairness

- Principle: ensure that some measure *differs by no more than* ϵ between several *protected* subgroups
- Many metrics proposed, depending on the measure to be equalized

Table: Summary of four statistical fairness metrics widely used in the literature.

Metric	Statistical Measure	$\text{unf}(d, \mathcal{E})$
Equal Opportunity (EOpp) [12]	**True Positive Rate**	$\left\| \frac{TP^c_{\mathcal{E},p}}{\|\mathcal{E}^p \cap \mathcal{E}^+\|} - \frac{TP^c_{\mathcal{E},u}}{\|\mathcal{E}^u \cap \mathcal{E}^+\|} \right\| \leq \epsilon$
Statistical Parity (SP) [8]	Probability of Positive Prediction	$\left\| \frac{TP^c_{\mathcal{E},p} + FP^c_{\mathcal{E},p}}{\|\mathcal{E}^p\|} - \frac{TP^c_{\mathcal{E},u} + FP^c_{\mathcal{E},u}}{\|\mathcal{E}^u\|} \right\| \leq \epsilon$
Predictive Equality (PE) [6]	False Positive Rate	$\left\| \frac{FP^c_{\mathcal{E},p}}{\|\mathcal{E}^p \cap \mathcal{E}^-\|} - \frac{FP^c_{\mathcal{E},u}}{\|\mathcal{E}^u \cap \mathcal{E}^-\|} \right\| \leq \epsilon$
Equalized Odds (EO) [12]	PE and EOpp	Conjunction of PE and EOpp

Rule Lists: Definition

Rule lists [13] are classifiers formed by an ordered list of *if-then* rules with antecedents in the *if* clauses and predictions in the *then* clauses.
More precisely, a *rule list* is a tuple $d = (\delta_d, q_0)$ in which
$\delta_d = (r_1, r_2, \ldots, r_k)$ is d's *prefix*, and $q_0 \in \{0, 1\}$ is a *default prediction*.
A prefix is an ordered list of k distinct association rules $r_i = a_i \to q_i$.

Example rule list

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
else [high]
```

CORELS and FairCORELS

- CORELS [3, 4] is a branch-and-bound algorithm proposed to learn Certifiably Optimal sparse RulE ListS, minimizing the following objective function:

$$\mathrm{obj}(d, \mathcal{E}) = \mathrm{misc}(d, \mathcal{E}) + \lambda \cdot K_d$$

  where K_d is the length of rule list d, $\mathrm{misc}(d, \mathcal{E})$ is the misclassification error of d on $\mathcal{E}$, and λ an hyperparameter to balance the sparsity/accuracy tradeoff

- FairCORELS [1, 2] is a bi-objective extension of CORELS, addressing the following problem (where $\mathcal{R}$ is the space of rule lists):

$$\begin{aligned} \arg\min_{d \in \mathcal{R}} \quad & \mathrm{obj}(d, \mathcal{E}) \\ \text{s.t.} \quad & \mathrm{unf}(d, \mathcal{E}) \leq \epsilon \end{aligned}$$

CORELS/FairCORELS search space

- FairCORELS represents the search space of rule lists as a prefix tree (trie)
- FairCORELS leverages several bounds and proposes a collection of exploration strategies (BFS, DFS, Best-First searches...) to efficiently explore this search space
- The different exploration strategies differ by the *priority queue ordering*

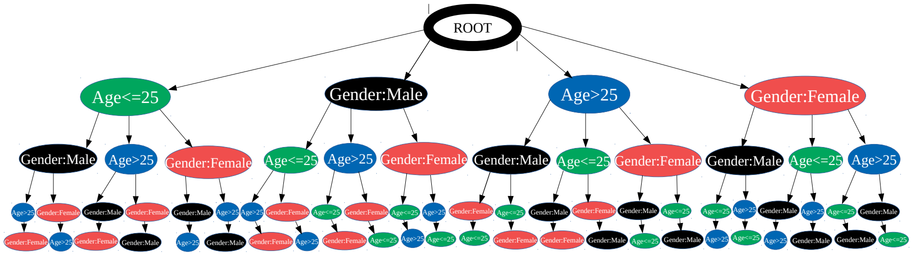


Figure: Example prefix tree with 4 attributes

Example rule list

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
else [high]
```

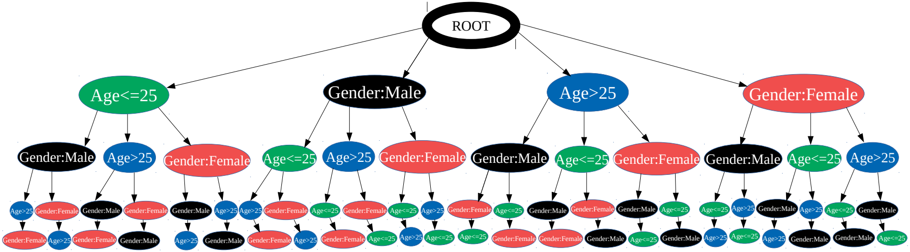


Figure: Example prefix tree with 4 attributes

Limits of existing FairCORELS implementation [1, 2]

- FairCORELS is mostly an incremental extension of CORELS, updating the current best solution only if it satisfies a fairness constraint
- However, the fairness constraints modify the set of acceptable solutions, and make CORELS' original bounds and exploration heuristics weaker
- Indeed, learning optimal interpretable models under constraints (*e.g.*, fairness constraints) has been identified as of the main technical challenges towards interpretable machine learning [14]

Example prefix δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
```

Example rule list d_1, extension of δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
else if [Education:Master] then [high]
else if [Capital_Gain>0] then [high]
else [low]
```

	Gender	Age	Education	...	true label
e_1	**Female**	30	Masters	...	high
e_2	Male	30	School	...	low
...	...	...	...	...	...

Table: Example dataset $\mathcal{E}$

Intuition

- Each example can either be determined by δ_1 (if a rule in δ_1 captures it) or not
- Any example determined by δ_1 will have the same classification for any extension of δ_1 (*e.g.*, d_1)

Example prefix δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
```

Example rule list d_1, extension of δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
else if [Education:Master] then [high]
else if [Capital_Gain>0] then [high]
else [low]
```

	Gender	Age	Education	...	true label
e_1	**Female**	30	Masters	...	high
e_2	Male	30	School	...	low
...	...	...	...	...	...

Table: Example dataset $\mathcal{E}$

Intuition

- Here, any extension of δ_1 will have at least one True Positive (e_1)
- Similarly, it can have at most $(|\mathcal{E}| - 1)$ False Negatives

Example prefix δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
```

Example rule list d_1, extension of δ_1

```
if [Gender:Female] then [high]
else if [Age<=25] then [low]
else if [Education:Master] then [high]
else if [Capital_Gain>0] then [high]
else [low]
```

	Gender	Age	Education	...	true label
e_1	**Female**	30	Masters	...	high
e_2	Male	30	School	...	low
...	...	...	...	...	...

Table: Example dataset $\mathcal{E}$

Intuition

- At each node of `FairCORELS`'s prefix tree, we check whether it is possible that an extension of the associated prefix simultaneously improves the current best objective function and meets the fairness requirement, given the prefix's predictions.
- If it is not possible, we prune the entire subtree

The ILP model for Equal Opportunity: $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$

- Inputs: Prefix δ, dataset $\mathcal{E}$, accuracy lower and upper bounds L and U, unfairness tolerance ϵ
- Variables:

δ's predictions define the variables' domains

$$x^{TP_{\mathcal{E},p}} \in [TP^{\delta}_{\mathcal{E},p}, |\mathcal{E}^p \cap \mathcal{E}^+| - FN^{\delta}_{\mathcal{E},p}],\ x^{TP_{\mathcal{E},u}} \in [TP^{\delta}_{\mathcal{E},u}, |\mathcal{E}^u \cap \mathcal{E}^+| - FN^{\delta}_{\mathcal{E},u}],$$
$$x^{FP_{\mathcal{E},p}} \in [FP^{\delta}_{\mathcal{E},p}, |\mathcal{E}^p \cap \mathcal{E}^-| - TN^{\delta}_{\mathcal{E},p}],\ x^{FP_{\mathcal{E},u}} \in [FP^{\delta}_{\mathcal{E},u}, |\mathcal{E}^u \cap \mathcal{E}^-| - TN^{\delta}_{\mathcal{E},u}].$$

- Constraints:

#well classified examples

$$L \leq x^{TP_{\mathcal{E},p}} + x^{TP_{\mathcal{E},u}} + |\mathcal{E}^p \cap \mathcal{E}^-| - x^{FP_{\mathcal{E},p}} + |\mathcal{E}^u \cap \mathcal{E}^-| - x^{FP_{\mathcal{E},u}} \leq U \qquad (1)$$

$$-C_3 \leq |\mathcal{E}^p \cap \mathcal{E}^+| \times x^{TP_{\mathcal{E},u}} - |\mathcal{E}^u \cap \mathcal{E}^+| \times x^{TP_{\mathcal{E},p}} \leq C_3 \qquad (2)$$

with $C_3 = \epsilon \times |\mathcal{E}^p \cap \mathcal{E}^+| \times |\mathcal{E}^u \cap \mathcal{E}^+|$

fairness constraint

Theorem: Sufficient Condition to Reject Prefixes

- We define $\sigma(\delta)$ to be the set of all rule lists whose prefixes start with δ: $\sigma(\delta) = \{(\delta_d, q_0) \mid \delta_d \text{ starts with } \delta\}$, and $W^d_{\mathcal{E}}$ the number of examples in $\mathcal{E}$ well classified by d.
- Given a prefix δ, an unfairness tolerance $\epsilon \in [0,1]$, and $0 \leq L \leq U \leq |\mathcal{E}|$, if $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$ is unsatisfiable then we have:

$$\nexists d \in \sigma(\delta) \mid L \leq W^d_{\mathcal{E}} \leq U \text{ and } \text{unf}_{EOpp}(d, \mathcal{E}) \leq \epsilon$$

Setting L and U

- L (lower bound on #well classified examples) is set to a tight value, corresponding to the minimum #examples that must be correctly classified to improve the current best objective function, given δ's length (the higher λ, the higher L)
- U (upper bound on #well classified examples) is set to a tight value, corresponding to the maximum number of examples that a classification function can classify correctly, given δ's errors and the inconsistencies in $\mathcal{E}$

Using the ILP to Enhance Exploration

Pruning Version

- We solve $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$ at each node of the prefix tree
- If UNSAT, then we (safely) prune the entire subtree

Guiding Version

- We add an objective to $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$:
  - Objective: maximize $x^{TP_{\mathcal{E},p}} - x^{FP_{\mathcal{E},p}} + x^{TP_{\mathcal{E},u}} - x^{FP_{\mathcal{E},u}}$ ← $\propto$ #well classified examples
- If UNSAT, then we (safely) we prune the entire subtree
- If SAT, we get an upper bound on the accuracy that any classification function consistent with δ's predictions can reach. This gives us a **lower bound on `FairCORELS`'s objective function, which can be used to order the priority queue and guide exploration**

Table of Contents

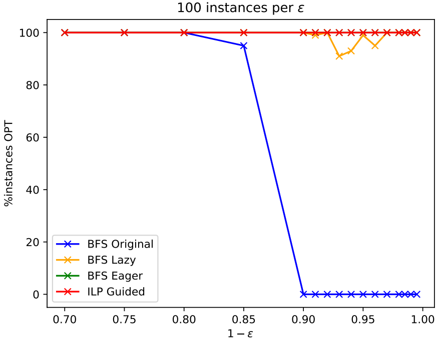
Integrating our ILP within FairCORELS

- We implement and solve the ILP models in C++ using the ILOG CPLEX 20.10 solver
- We consider different integrations
  - BFS Original: original FairCORELS with existing Breadth-First Search (BFS) exploration heuristic
  - BFS Eager: using a BFS policy, performs the ILP-based pruning **before** inserting a node into the priority queue
  - BFS Lazy: using a BFS policy, performs the ILP-based pruning **after** extracting a node from the priority queue
  - ILP Guided: best-first search (priority queue ordered by the ILP objectives) with an Eager pruning

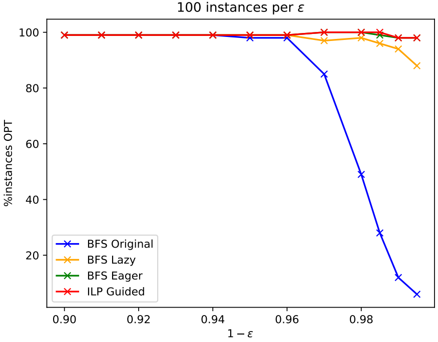
Experimental Setup

We compare the four approaches:

- On two datasets:
  - COMPAS [5]
    - Number of examples: 6150
    - Binary classification task: Recidivism within two years
    - Sensitive attribute: Race (African-American/Caucasian)
    - Number of binary rules: 18
  - German Credit [7]
    - Number of examples: 1000
    - Binary classification task: Good or bad credit score
    - Sensitive attribute: Age (Low/High)
    - Number of binary rules: 49
- On the four fairness metrics of Table 1 (we report results for the Equal Opportunity metric hereafter)
- Maximum memory use: 4 Gb
- Maximum CPU time: 20 minutes (COMPAS), 40 minutes (German Credit)
- For each dataset: 100 random different train/test splits

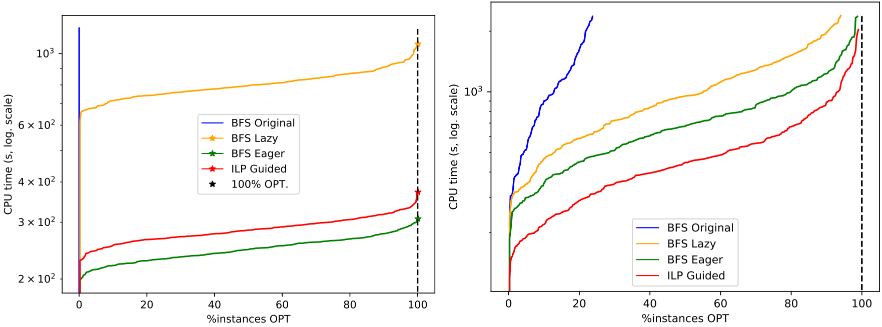


(a) COMPAS dataset



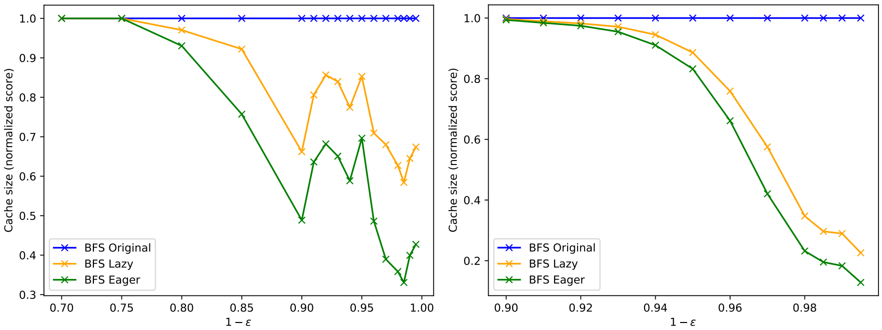
(b) German Credit dataset

Figure: Proportion of instances solved to optimality as a function of $1-\epsilon$.



(a) COMPAS dataset (b) German Credit dataset

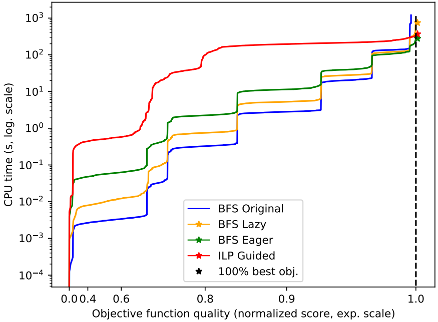
Figure: CPU time as a function of the proportion of instances solved to optimality, for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).



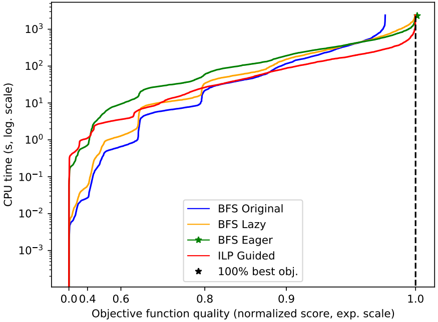
(a) COMPAS dataset

(b) German Credit dataset

Figure: Relative cache size (#nodes) as a function of $1 - \epsilon$ (experiments for the Equal Opportunity fairness metric).



(a) COMPAS dataset



(b) German Credit dataset

Figure: Solving time as a function of the objective function quality normalized score, for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).

CORELS' Prefix Permutation Map

- CORELS' prefix tree contains many symmetries
- A *prefix permutation map* ensures that only the best permutation of each set of rules is kept
- This symmetry-aware data structure considerably reduces the running time and the memory consumption [3, 4]
- It cannot be used within FairCORELS without sacrificing optimality

FairCORELS' fairness-compatible Prefix Permutation Map

- We design a weaker prefix permutation map, which can by used while maintaining the guarantee of optimality
- Our new symmetry-breaking mechanism:
  - Considers that two prefixes are equivalent if and only if they define exactly the same confusion matrix and their rules contain the same antecedents
  - Pushes a new prefix into the priority queue if and only if it contains no equivalent prefix

Compared Approaches

- We implement and solve the ILP models in C++ using the `ILOG CPLEX 20.10` solver
- We consider different ILP-based pruning approaches, with (`PMAP`) or without (`No PMAP`) the new Prefix Permutation Map
  - `BFS Original`: original `FairCORELS` with existing Breadth-First Search (BFS) exploration heuristic
  - `BFS Eager`: using a BFS policy, performs the ILP-based pruning **before** inserting a node into the priority queue
  - `BFS Lazy`: using a BFS policy, performs the ILP-based pruning **after** extracting a node from the priority queue
- Results for the `ILP-Guided` approach are not reported because they are always the worst

Experimental Setup

We compare the three pruning approaches, with or without the new PMAP:

- On the Adult Income dataset [7]
  - Number of examples: 48, 842
  - Binary classification task: Income greater than $50, 000 per year
  - Sensitive attribute: Gender (Female/Male)
  - Number of binary rules: 47
- On the Statistical Parity fairness metric
- Maximum memory use: 8 Gb
- Maximum CPU time: 120 minutes
- 100 random different train/test splits

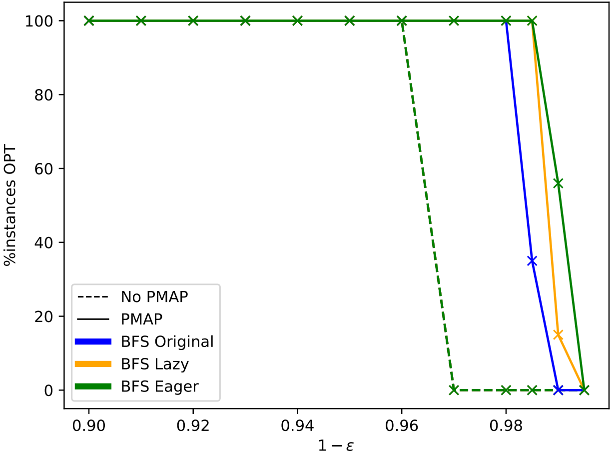


Figure: Proportion of instances solved to optimality as a function of $1 - \epsilon$.

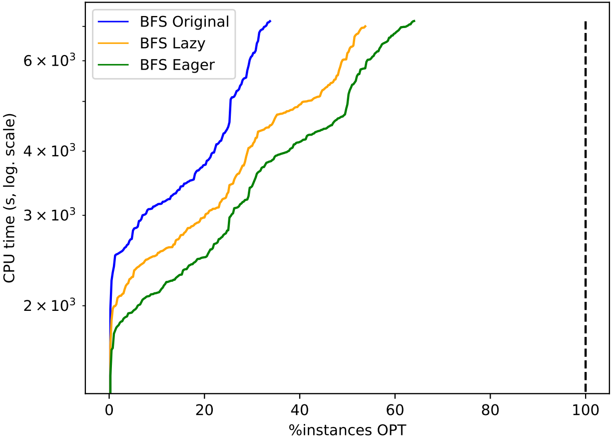


Figure: CPU time as a function of the proportion of instances solved to optimality (using the new PMAP), for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).

Reducing Priority Queue (Cache) Size

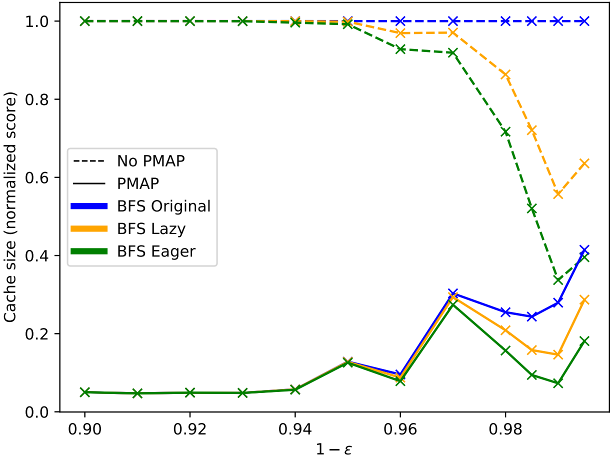


Figure: Relative cache size (#nodes) as a function of $1-\epsilon$ (experiments for the Equal Opportunity fairness metric).

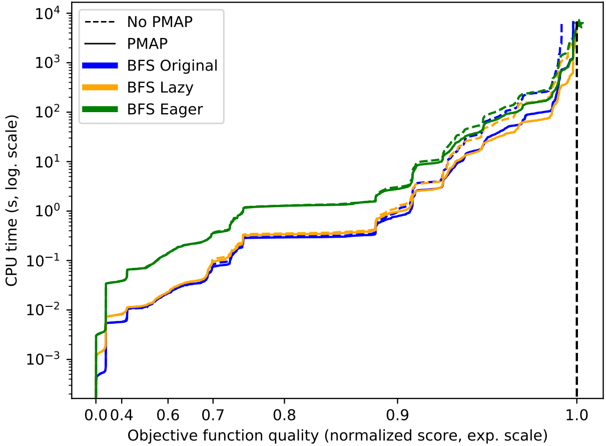


Figure: Solving time as a function of the objective function quality normalized score, for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).

ϵ	PMAP	BFS Original Train Acc	BFS Original Test Acc	BFS Original Test Unf viol.	BFS Lazy Train Acc	BFS Lazy Test Acc	BFS Lazy Test Unf viol.	BFS Eager Train Acc	BFS Eager Test Acc	BFS Eager Test Unf viol.
All	No	.938	.942	**-.004**	.963	.966	**-.004**	.964	.967	**-.004**
	Yes	.966	.97	**-.004**	.998	.987	**-.004**	**1**	**.989**	**-.004**
< 0.02	No	.815	.835	**.0**	.89	.907	.001	.892	.91	.001
	Yes	.897	.91	.001	.993	.96	.001	**1**	**.968**	.001

Table: Learning quality evaluation (Adult Income dataset, $\epsilon \in [0.005, 0.1]$): Proportion of instances for which each method led to the best train (resp. test) accuracy, and average violation of the fairness constraint at test time.

Our ILP-based pruning approach

- Leverages jointly accuracy and fairness to prune the search space of `FairCORELS`
- Leads to significant improvements on all evaluated criteria: reaching better objective function values and certifying optimality using a reduced amount of memory and time
- Is flexible thanks to its declarative nature and can handle multiple fairness criteria and/or sensitive groups

Future Works

- Considering other learning algorithms and machine learning models
- Guiding the exploration (as attempted with the `ILP-Guided` approach)

Useful Links

- Full paper accepted at CPAIOR 2022 (preprint available on my homepage: `https://homepages.laas.fr/jferry`)
- Source code available online: `https://github.com/ferryjul/fairCORELSV2`

- Thank you for your attention
- Any questions ?
- In Montreal until the 23rd of July, in UQAM until the 3rd of June $\Longrightarrow$ feel free to reach out!

Contact

- Homepage: `https://homepages.laas.fr/jferry`
- Mail: `jferry@laas`

References I

[1] Aïvodji, U., Ferry, J., Gambs, S., Huguet, M.-J., and Siala, M. (2019). Learning fair rule lists. *arXiv preprint arXiv:1909.03977*.

[2] Aïvodji, U., Ferry, J., Gambs, S., Huguet, M.-J., and Siala, M. (2021). Faircorels, an open-source library for learning fair rule lists. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 4665–4669, New York, NY, USA. Association for Computing Machinery.

[3] Angelino, E., Larus-Stone, N., Alabi, D., Seltzer, M., and Rudin, C. (2017). Learning certifiably optimal rule lists. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 35–44. ACM.

[4] Angelino, E., Larus-Stone, N., Alabi, D., Seltzer, M., and Rudin, C. (2018). Learning certifiably optimal rule lists for categorical data. *Journal of Machine Learning Research*, 18(234):1–78.

[5] Angwin, J., Larson, J., Mattu, S., and Kirchner, L. (2016). Machine bias. *ProPublica, May*, 23.

[6] Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163.

[7] Dua, D. and Graff, C. (2017). UCI machine learning repository.

[8] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226. ACM.

References II

[9] Ferry, J., Aïvodji, U., Gambs, S., Huguet, M.-J., and Siala, M. (2021). Améliorer la généralisation de l'équité en apprentissage grâce à l'optimisation distributionnellement robuste. In *Rencontres des Jeunes Chercheurs en Intelligence Artificielle*.

[10] Ferry, J., Aïvodji, U., Gambs, S., Huguet, M.-J., and Siala, M. (2022a). Improving fairness generalization through a sample-robust optimization method. *Machine Learning*, S.I. on Safe & Fair ML.

[11] Ferry, J., Aïvodji, U., Gambs, S., Huguet, M.-J., and Siala, M. (2022b). Leveraging integer linear programming to learn optimal fair rule lists. In *19th International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*.

[12] Hardt, M., Price, E., Srebro, N., et al. (2016). Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323.

[13] Rivest, R. L. (1987). Learning decision lists. *Machine learning*, 2(3):229–246.

[14] Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. (2021). Interpretable machine learning: Fundamental principles and 10 grand challenges. *arXiv preprint arXiv:2103.11251*.

Example of the Equal Opportunity Metric

- We add an objective to $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$:
  - Objective: maximize $x^{TP_{\mathcal{E},p}} - x^{FP_{\mathcal{E},p}} + x^{TP_{\mathcal{E},u}} - x^{FP_{\mathcal{E},u}}$
- On the one hand, this optimization problem may take longer to solve than the simple feasibility problem defined earlier
- On the other hand:
  - If $ILP_{EOpp}(\delta, \mathcal{E}, L, U, \epsilon)$ is UNSAT, we can safely prune the subtree associated to δ in the prefix tree
  - **Otherwise, we now get an upper bound on the accuracy that any classification function consistant with δ's predictions can reach. This gives us a lower bound on `FairCORELS`'s objective function, which can be used to order the priority queue**
- Finally, we can leverage the ILP to guide exploration towards the prefixes whose predictions cause less conflict between accuracy and fairness, effectively speeding up exploration

The ILP model for Equalized Odds: $ILP_{EO}(\delta, \mathcal{E}, L, U, \epsilon)$

- Inputs: Prefix δ, dataset $\mathcal{E}$, accuracy lower and upper bounds L and U, unfairness tolerance ϵ
- Variables:

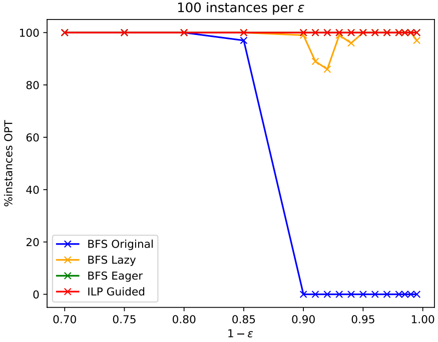
$$x^{TP_{\mathcal{E},p}} \in [TP^{\delta}_{\mathcal{E},p}, |\mathcal{E}^p \cap \mathcal{E}^+| - FN^{\delta}_{\mathcal{E},p}],\ x^{TP_{\mathcal{E},u}} \in [TP^{\delta}_{\mathcal{E},u}, |\mathcal{E}^u \cap \mathcal{E}^+| - FN^{\delta}_{\mathcal{E},u}],$$
$$x^{FP_{\mathcal{E},p}} \in [FP^{\delta}_{\mathcal{E},p}, |\mathcal{E}^p \cap \mathcal{E}^-| - TN^{\delta}_{\mathcal{E},p}],\ x^{FP_{\mathcal{E},u}} \in [FP^{\delta}_{\mathcal{E},u}, |\mathcal{E}^u \cap \mathcal{E}^-| - TN^{\delta}_{\mathcal{E},u}].$$

- Constraints:

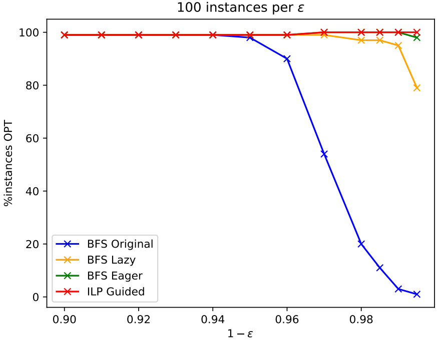
$$L \leq x^{TP_{\mathcal{E},p}} + x^{TP_{\mathcal{E},u}} + |\mathcal{E}^p \cap \mathcal{E}^-| - x^{FP_{\mathcal{E},p}} + |\mathcal{E}^u \cap \mathcal{E}^-| - x^{FP_{\mathcal{E},u}} \leq U \quad (3)$$
$$-C_2 \leq |\mathcal{E}^u \cap \mathcal{E}^-| \times x^{FP_{\mathcal{E},p}} - |\mathcal{E}^p \cap \mathcal{E}^-| \times x^{FP_{\mathcal{E},u}} \leq C_2 \quad (4)$$
$$-C_3 \leq |\mathcal{E}^p \cap \mathcal{E}^+| \times x^{TP_{\mathcal{E},u}} - |\mathcal{E}^u \cap \mathcal{E}^+| \times x^{TP_{\mathcal{E},p}} \leq C_3 \quad (5)$$

with $C_2 = \epsilon \times |\mathcal{E}^u \cap \mathcal{E}^-| \times |\mathcal{E}^p \cap \mathcal{E}^-|$ and $C_3 = \epsilon \times |\mathcal{E}^p \cap \mathcal{E}^+| \times |\mathcal{E}^u \cap \mathcal{E}^+|$

Experiments Results for the Equalized Odds Metric I



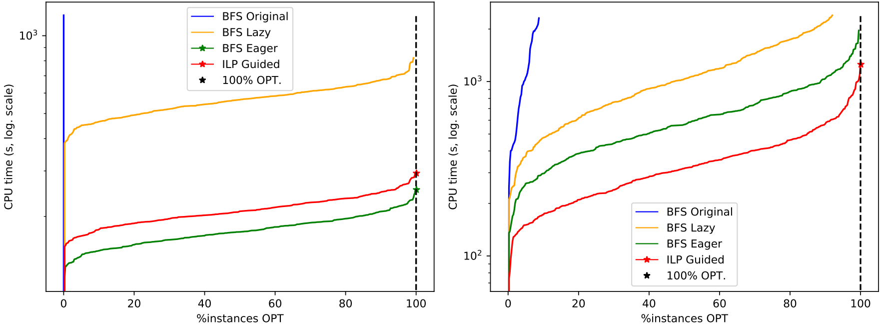
(a) COMPAS dataset



(b) German Credit dataset

Figure: Proportion of instances solved to optimality as a function of $1 - \epsilon$.

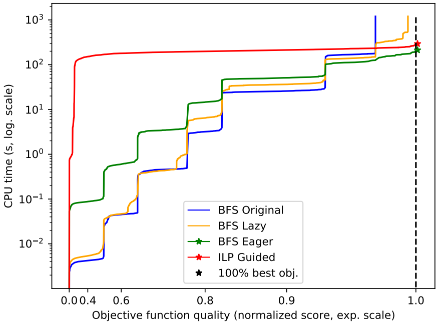
Experiments Results for the Equalized Odds Metric II



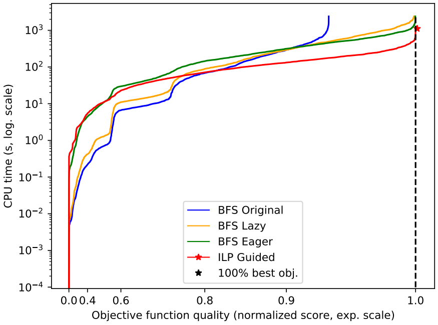
(a) COMPAS dataset　　(b) German Credit dataset

Figure: CPU time as a function of the proportion of instances solved to optimality, for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).

Experiments Results for the Equalized Odds Metric III

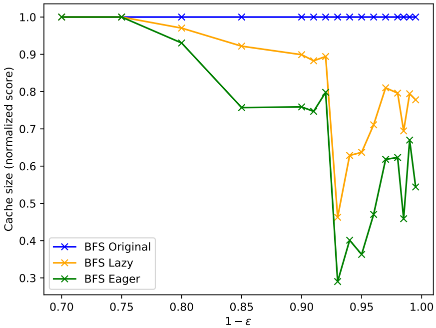


(a) COMPAS dataset

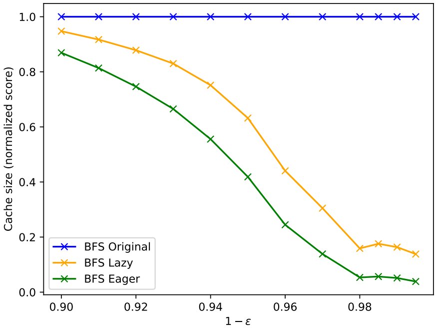


(b) German Credit dataset

Figure: Solving time as a function of the objective function quality normalized score, for high fairness requirements (unfairness tolerances ranging between 0.005 and 0.02).



(a) COMPAS dataset



(b) German Credit dataset

Figure: Relative cache size (#nodes) as a function of $1-\epsilon$.