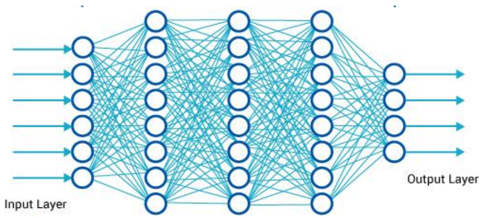
Strong statistical parity through fair synthetic data

Ivona Krchova, Michael Platzer, Paul Tiwald

Structured Synthetic Data

NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Mary	25y	female	Book	12€	4/2/19	8:12
John	72y	male	Pizza	34€	4/2/19	18:12
...						
Bill	18y	male	Swim	6€	4/4/19	10:02
Bill	18y	male	Shoes	123€	4/4/19	12:32



NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Kim	29y	female	Amazon	236€	4/4/19	12:32
Kim	29y	female	Zalando	36€	4/4/19	18:58
...						
Brian	82y	male	Beer	6€	4/2/19	21:32
Sue	24y	female	Sushi	12€	4/2/19	21:32

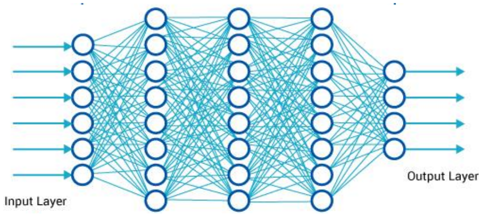
Real Data

Synthetic Data

One of the main use cases: Privacy-By-Design

NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Mary	25y	female	Book	12€	4/2/19	8:12
John	72y	male	Pizza	34€	4/2/19	18:12
...						
Bill	18y	male	Swim	6€	4/4/19	10:02
Bill	18y	male	Shoes	123€	4/4/19	12:32

Real Data



NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Kim	29y	female	Amazon	236€	4/4/19	12:32
Kim	29y	female	Zalando	36€	4/4/19	18:58
...						
Brian	82y	male	Beer	6€	4/2/19	21:32
Sue	24y	female	Sushi	12€	4/2/19	21:32

Synthetic Data

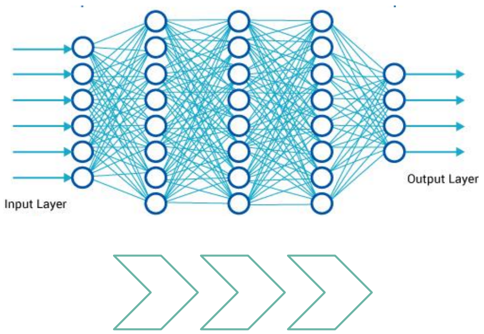
Privacy Protection

MOSTLY·AI

Can Synthetic Data help with Fairness-By-Design?

NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Mary	25y	female	Book	12€	4/2/19	8:12
John	72y	male	Pizza	34€	4/2/19	18:12
...						
Bill	18y	male	Swim	6€	4/4/19	10:02
Bill	18y	male	Shoes	123€	4/4/19	12:32

Real Data



NAME	AGE	GENDER	ITEM	EUR	DATE	TIME
Kim	29y	female	Amazon	236€	4/4/19	12:32
Kim	29y	female	Zalando	36€	4/4/19	18:58
...						
Brian	82y	male	Beer	6€	4/2/19	21:32
Sue	24y	female	Sushi	12€	4/2/19	21:32

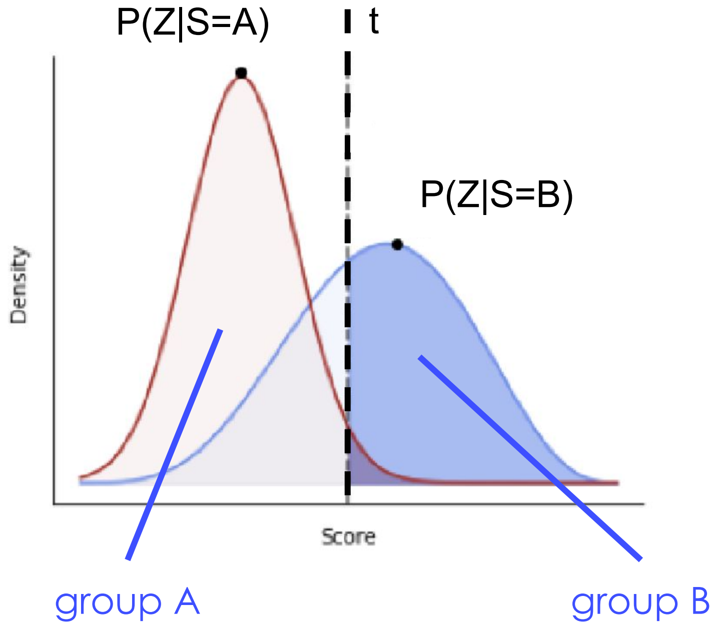
Synthetic Data

Fairness ?

Strong Statistical Parity

$$P(Z \geq t | S = s_i) = P(Z \geq t | S = s_j) \quad \text{for } i, j \in \{1, 2, \ldots, K\}, \quad \forall t \in Z$$

- Positive rates of sensitive groups S must be equal for **ALL** thresholds **t** of the downstream model:



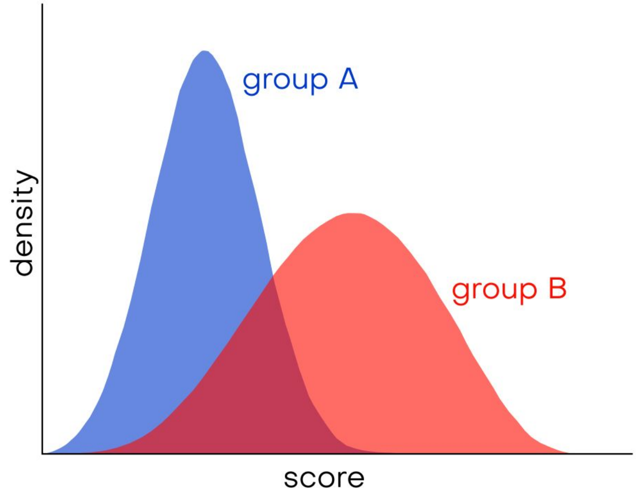
[Geometric Repair for Fair Classification at Any Decision Threshold]

Fair Synthetic Data subject to strong statistical parity

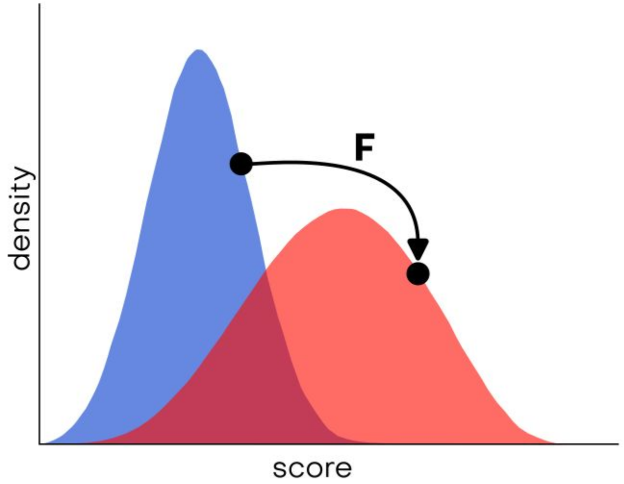
1
generate synthetic data

<features>	sensitive col	target
…	B	1
…	A	0
…	A	0
…	…	…

2
train classifier and calc. propensity scores for target column



3
learn transformation **F** to uplevel scores of group A



4
use upleveled scores to re-sample the target values for group B

<features>	sensitive col	fair target
…	B	1
…	A	0
…	A	1
…	…	…

MOSTLY·AI

Advantages

Flexibility: fairness measures applied **after** training the SD Generator

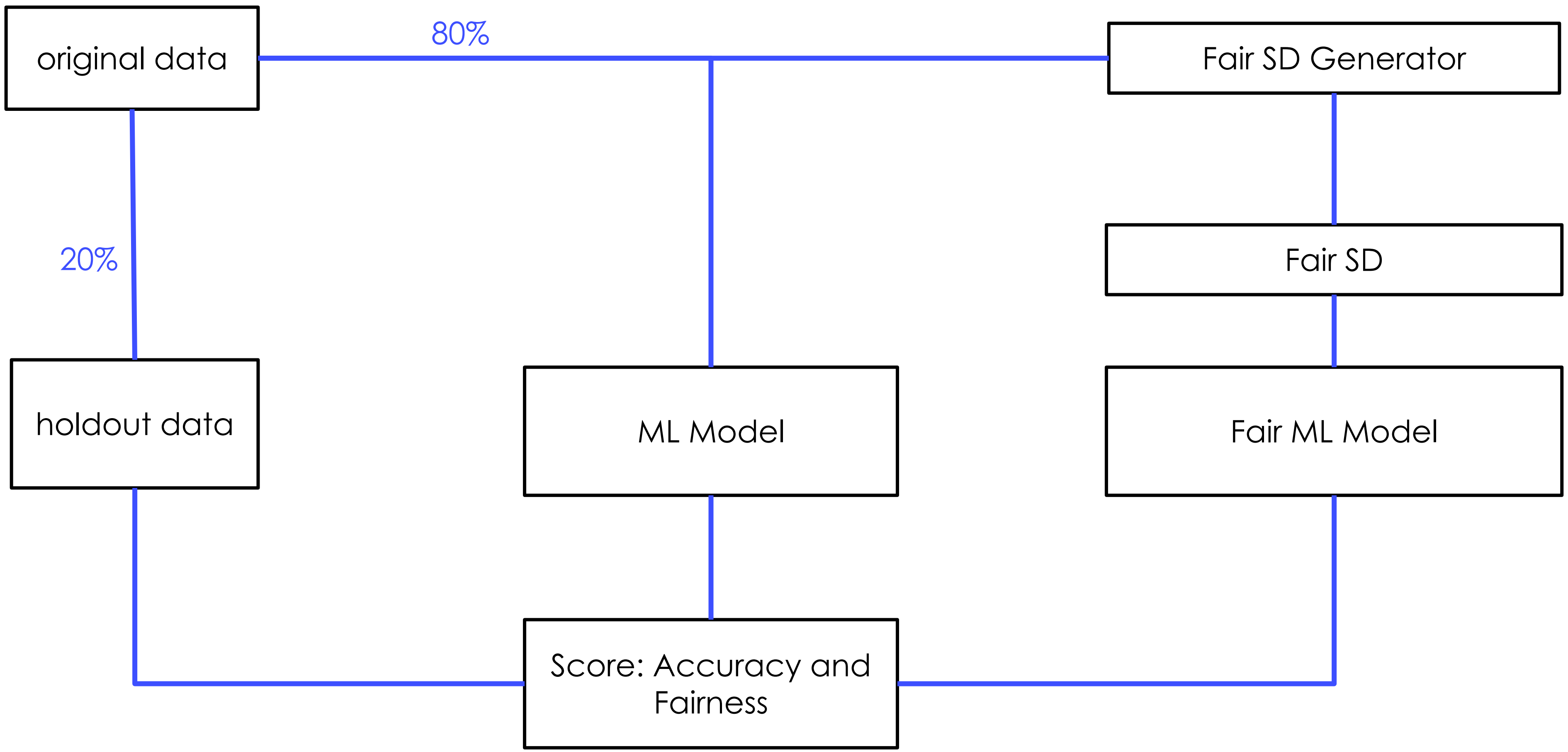
- no a-priori information about the data is needed
- select sensitive and target column without re-training
- tune scores/distributions to fit your needs (uplevel, downlevel, mix of both)
- tune strength of correction (fairness weight) without retraining the SD Generator

More robust fairness across thresholds in downstream task

Interpretable “fairness weight”:

- new score = $\lambda \cdot F(\text{score}) + (1-\lambda) \cdot \text{score}$

"Experimental Setup"



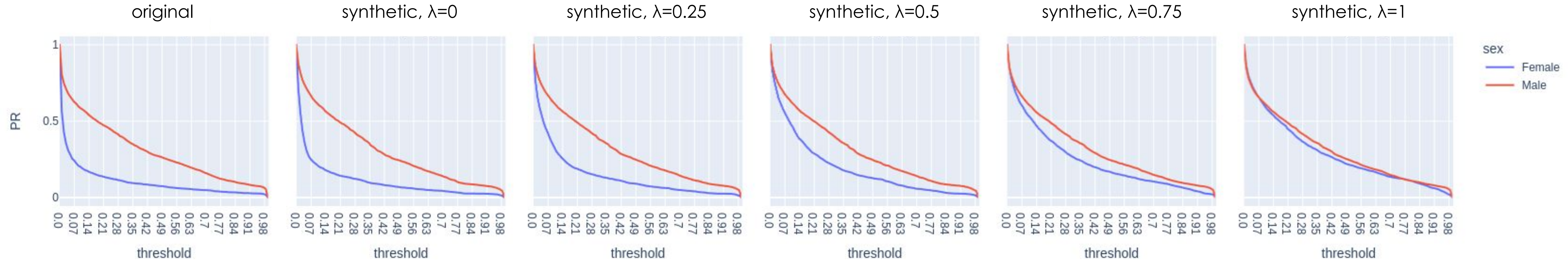
prerequisite: "reliable" SD Generator and downstream model

MOSTLY·AI

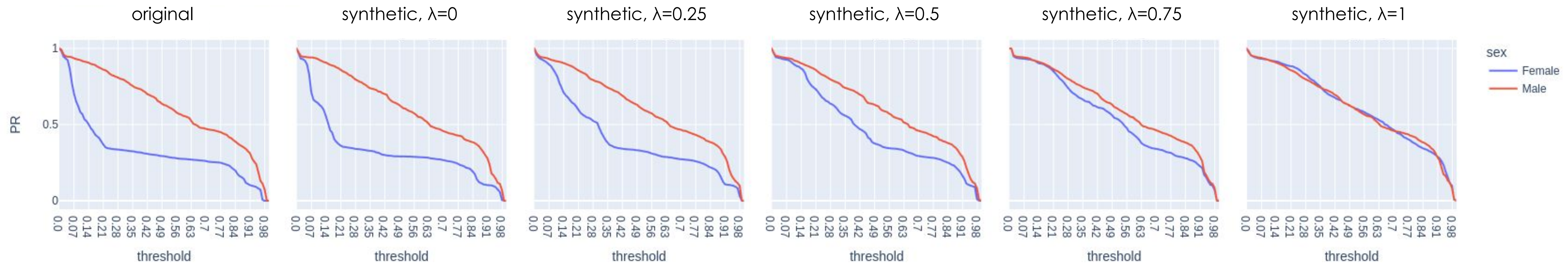
Positive rates of unprivileged groups “catch up”

… with increasing fairness weight λ

Dataset: Adult Census, Model: XGBoost

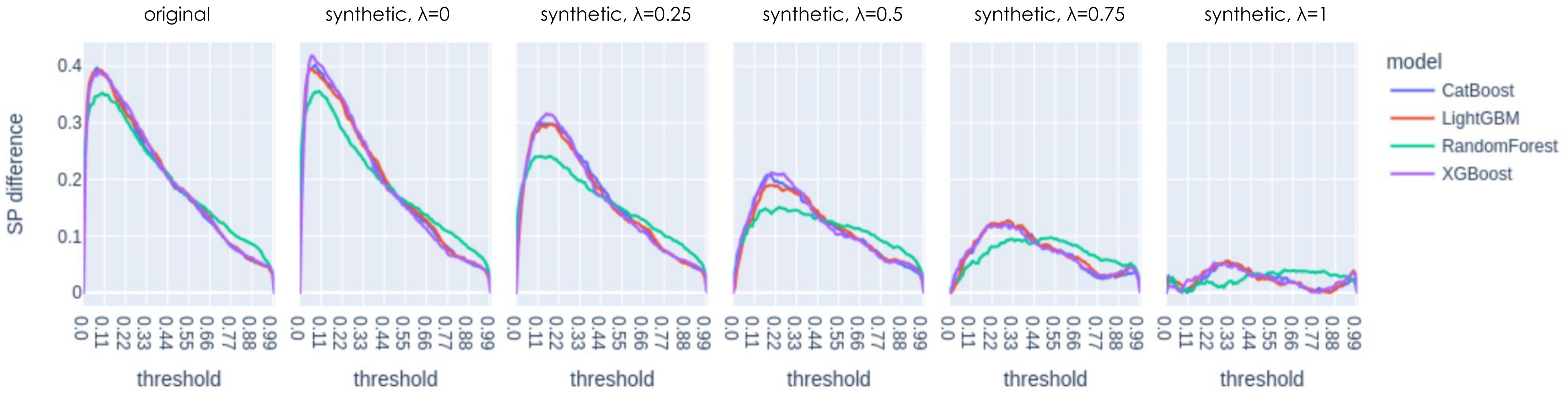


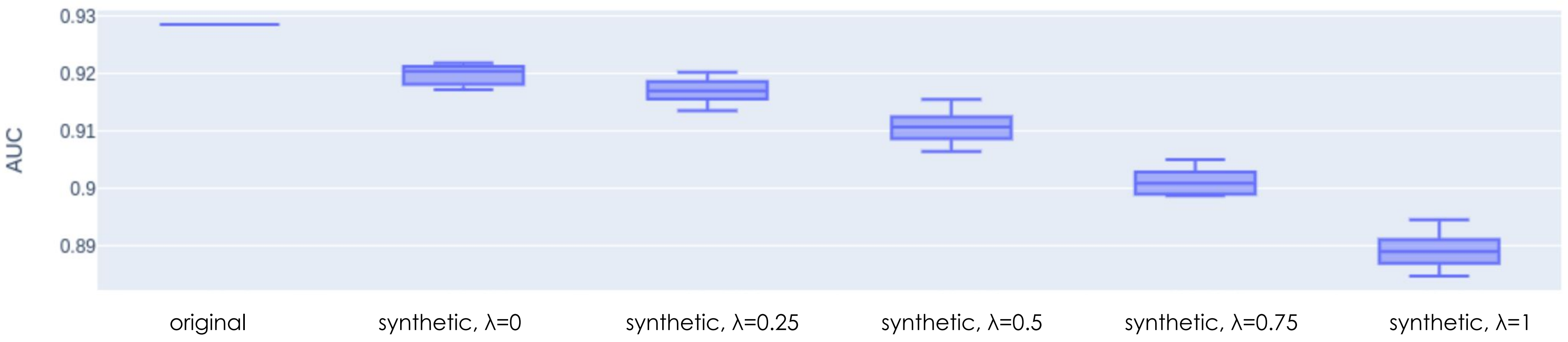
Dataset: Dutch Census, Model: LightGBM



MOSTLY·AI

Results - Adult Census Data Set





MOSTLY·AI