Towards Explainable Prediction Models on High-dimensional Behavioral & Textual data

by Yanou Ramon
supervisor: Prof. David Martens

Research Seminar – June 19, 2020 – 11am
Faculty of Business & Economics, University of Antwerp

ABOUT ME

YANOU RAMON

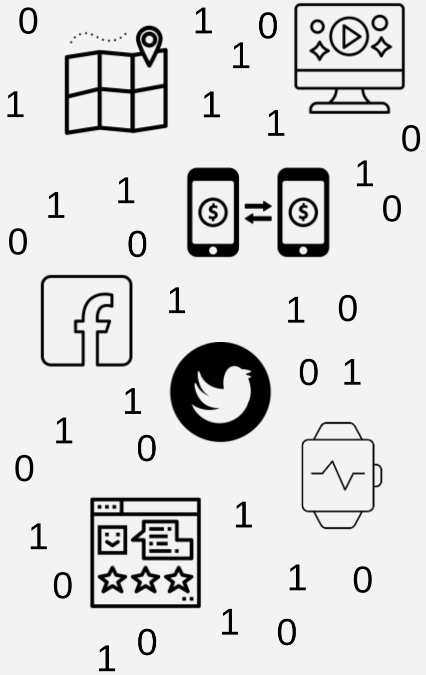
PhD student at University of Antwerp
Applied Data Mining Group - Prof. David Martens

Towards explainable prediction models on high-dimensional behavioral and textual data

M.Sc. in Business Engineering (Finance)
University of Antwerp

Big Data, Data Mining, Artificial Intelligence, programming etc.

DATA-DRIVEN DECISION-MAKING



Behavioral and textual data
(High-dimensional & sparse)

Prediction model

Predicted value of variable of interest

automated decisions or decision support

LOCATION DATA

smartphone sensor data (GPS locations), online "check-ins",...

Example applications:
- Ecommerce: efficient parcel delivery
- Psychological/behavioral profiling
- Customer relationship management
- Political party preference & orientation
- Daily habits, interests & preferences

LOCATION DATA

smartphone sensor data (GPS locations), online "check-ins",...

Example applications:

- Ecommerce: efficient parcel delivery
- Psychological/behavioral profiling
- Customer relationship management
- Political party preference & orientation
- Daily habits, interests & preferences

Contact-tracing apps raise privacy fears

Financial Times, April 2020

SOCIAL MEDIA & BROWSING DATA

Facebook/Instagram “likes”, Twitter posts, online reviews/blogposts, search queries,...

Example applications:

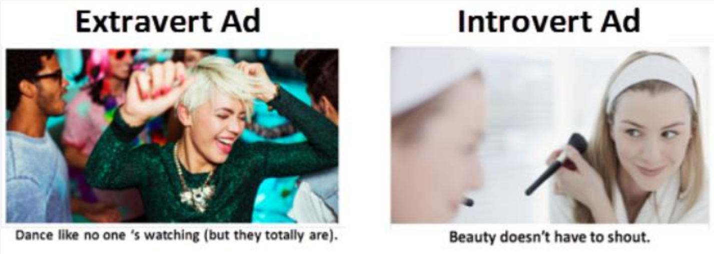
- Psychological/behavioral profiling
- Product interest & online targeted advertising
- Political party preference & orientation
- Behavioral credit scoring

SOCIAL MEDIA & BROWSING DATA

Facebook/Instagram "likes", Twitter posts, online reviews/blogposts, search queries,...

<u>Example applications:</u>

- Psychological/behavioral profiling
- Product interest & online targeted advertising
- Political party preference & orientation
- Behavioral credit scoring



Advertisers can target you psychologically based on a single Facebook like, study finds

Business Insider, November 2017; Matz et al., 2017

SOCIAL MEDIA & BROWSING DATA

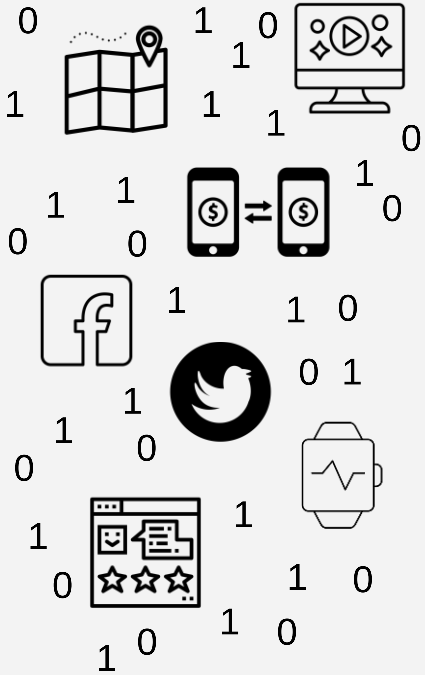
Facebook/Instagram “likes”, Twitter posts, online reviews/blogposts, search queries,…
But also: “metadata”

<u>Example applications:</u>

- Psychological/behavioral profiling
- Product interest & online targeted advertising
- Political party preference & orientation
- Behavioral credit scoring

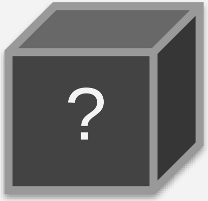
de Montjoye et al., 2013; Financial Times, March 2019

DATA-DRIVEN DECISION-MAKING



Behavioral and textual data
(High-dimensional & sparse)

Prediction model

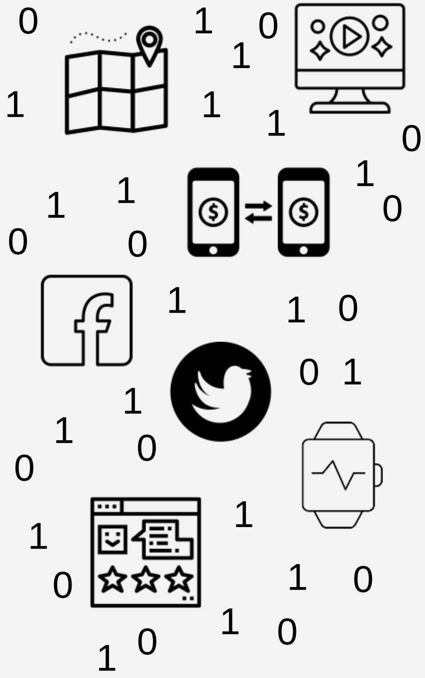


"Black Box"

⇒ ***Thousands of coefficients***
⇒ ***Nonlinear techniques***

Predicted value of variable of interest

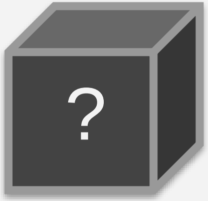
DATA-DRIVEN DECISION-MAKING



Behavioral and textual data
(High-dimensional & sparse)

"EXplainable Artificial Intelligence (XAI)"
"Interpretable Machine Learning"

Prediction model



"Black Box"

Predicted value of variable of interest

⇒ ***Thousands of coefficients***
⇒ ***Nonlinear techniques***

MOTIVATION

To what extent is the prediction (model) in line with expectations?

TRUST

Accurate

INSIGHT

Wrong

IMPROVE

(Martens, 2020)

EXPLAINING PREDICTION MODELS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global	Instance-level
Flexibility	Model-specific	Model-agnostic
Faithfulness	Intrinsic	Post-hoc
Output format	Rule, importance-ranked list, visualization, linear model,...	

EXPLAINING PREDICTION MODELS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global ★	Instance-level ★
Flexibility	Model-specific	Model-agnostic
Faithfulness	Intrinsic	Post-hoc
Output format	Rule, importance-ranked list, visualization, linear model,...	

EXPLAINING PREDICTION MODELS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global ★	Instance-level ★
Flexibility	Model-specific	Model-agnostic ★
Faithfulness	Intrinsic	Post-hoc
Output format	Rule, importance-ranked list, visualization, linear model,...	

EXPLAINING PREDICTION MODELS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global ★	Instance-level ★
Flexibility	Model-specific	Model-agnostic ★
Faithfulness	Intrinsic	Post-hoc ★
Output format	Rule, importance-ranked list, visualization, linear model,…	

EXPLAINING PREDICTION MODELS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global	Instance-level
Flexibility	Model-specific	Model-agnostic
Faithfulness	Intrinsic	Post-hoc
Output format	Rule, importance-ranked list, visualization, linear model,...	

OVERVIEW OF PROJECTS

I. **Deep Learning for Big, Sparse, Behavioral data**
De Cnudde et al., Big Data (2019)

II. **Instance-level explanation algorithms on behavioural and textual data: a counterfactual-oriented comparison**
Ramon et al., *Forthcoming in Advances in Data Analysis and Classification* (2020)

III. **Improving the cost of explainability for high-dimensional, sparse data using metafeatures-based rule-extraction**
Ramon et al., *Submitted to Machine Learning* (2020)

OVERVIEW OF PROJECTS

Towards explainable prediction models on high-dimensional behavioral and textual data

	GLOBAL	INSTANCE-LEVEL
Theoretical research	Metafeatures-based rule-extraction	Counterfactual explanation algorithms
Validation of methods in (business) applications		
Applied research	Psychological profiling/targeting	User study about explanation attributes
	Churn prediction	…
	…	

PROJECT 1

INSTANCE-LEVEL EXPLANATION ALGORITHMS ON BEHAVIORAL AND TEXTUAL DATA: A COUNTERFACTUAL-ORIENTED COMPARISON

Yanou Ramon, David Martens, Foster Provost, Theodoros Evgeniou

Forthcoming in Advances in Data Analysis and Classification (2020)


PROBLEM
STATEMENT

LOCATION DATA NYC: tourist or citizen?

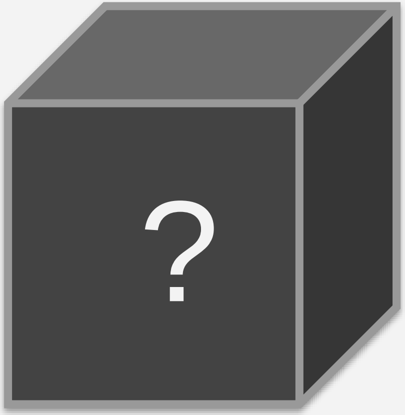
evidence "present" = active feature

	Columbia University	Time Square	DUMBO	…	Chelsea Market	*Target $\hat{y}$ Tourist*
Anna	1	1	1	…	0	***1***
Jack	1	0	0	…	1	***0***
…	…	…	…	…	…	***…***
Bill	0	0	1	…	0	***0***

➔ data matrix is very high-dimensional and sparse

	Columbia University	Time Square	DUMBO	…	Chelsea Market	*Target $\hat{y}$ Tourist*
Anna	1	1	1	…	0	***1***
Jack	1	0	0	…	1	***0***
…	…	…	…	…	…	***…***
Bill	0	0	1	…	0	***0***

LOCATION DATA NYC



$\hat{y} = 1$ ***if*** *tourist*
else $\hat{y} = 0$

“Black Box” model

$\Rightarrow$ Thousands of coefficients

$\Rightarrow$ Nonlinear techniques

(Local) interpretability issues

➔ Counterfactual explanations

COUNTERFACTUAL EXPLANATIONS

- Instance-level
- Causality within the model
- Minimal set of features such that the predicted class changes when “**removing**” them (setting value to zero)
- Very intuitive and comprehensible → contrastive nature “Why X rather than not-X?” (Miller, 2017)

COUNTERFACTUAL EXPLANATIONS

EXPLANATIONS help users to understand the relationship between the input (features) and the model's predicted output (target)

DIMENSIONS

Scope	Global	Instance-level ★
Flexibility	Model-specific	Model-agnostic ★
Faithfulness	Intrinsic	Post-hoc ★
Output format	Rule, importance-ranked list, visualization, linear model,...	

COUNTERFACTUAL EXPLANATIONS

Example: Tourist prediction using NYC location data

Anna visited 120 places last month
Anna was predicted as “tourist”

COUNTERFACTUAL EXPLANATIONS

Example: Tourist prediction using NYC location data

Anna visited 120 places last month
Anna was predicted as "tourist"

WHY?

COUNTERFACTUAL EXPLANATIONS

Example: Tourist prediction using NYC location data

Anna visited 120 places last month
Anna was predicted as "tourist"

		Columbia University	Time Square	DUMBO	...	Chelsea Market	Target $\hat{y}$ Tourist
$\mathbf{x}$	Anna	1	1	1	...	0	*1*
$\mathbf{z}_1$	*Anna (perturbed)*	*1*	*0*	*0*	...	*0*	*0*

IF Anna would **not** have visited **{Time Square, DUMBO}**,
THEN the predicted class changes from "tourist" to "NY citizen"

COUNTERFACTUAL ALGORITHMS

DESIDERATA

- Model-agnostic algorithm
- Find **minimum-sized** counterfactual explanation E for a single model prediction of instance $\mathbf{x}$

DESIDERATA

- Model-agnostic algorithm
- Find **minimum-sized** counterfactual explanation E for a single model prediction of instance $\mathbf{x}$

*More **comprehensible** (~cognitive limitations)*

*More **actionable**: e.g., "cloak" fewer online traces to get a desired outcome (not be targeted with ads of gay bars)*

FORMAL OBJECTIVE FUNCTION

- Original instance $\mathbf{x}$ vs perturbed instance $\mathbf{z}$

$$z_I = z = \begin{cases} \forall j \in I : z_j = 0 \\ \forall j \notin I : z_j = x_j \end{cases}$$

$\boldsymbol{E^*}$ = {Time Square, DUMBO}

$\boldsymbol{z^* = z_1}$

Example: NYC location data

		Columbia University	Time Square	DUMBO	...	Chelsea Market	Target $\hat{y}$ Tourist
$\mathbf{x}$	Anna	1	1	1	...	0	1
$\boldsymbol{z_1}$	Anna (perturbed)	1	0	0	...	0	0
$\boldsymbol{z_2}$	Anna (perturbed)	1	0	1	...	0	1

I forms a subset of the set of indices of the "active" features of $\mathbf{x}$

FORMAL OBJECTIVE FUNCTION

- Original instance $\mathbf{x}$ vs perturbed instance $\mathbf{z}$
- **Find $\mathbf{z}^*$ (or E^*) that is as close as possible to $\mathbf{x}$ and has a different predicted class**

$$A = \{\mathrm{z} | (\mathrm{z} = \underset{z}{\operatorname{argmin}}\ \mathrm{d(z,x)}) \wedge (\mathrm{f(z)} < t) \wedge (\forall x_j > 0 : z_j \in \{0, x_j\}) \wedge (\forall x_k = 0 : z_k = 0)\} \quad (2)$$

cosine distance *predicted class change* *only "active" features"are perturbed*

$$z^* = \underset{z \in A}{\operatorname{argmin}}\ \mathrm{f(z)} \quad (3)$$

FORMAL OBJECTIVE FUNCTION

tourist

NY citizen

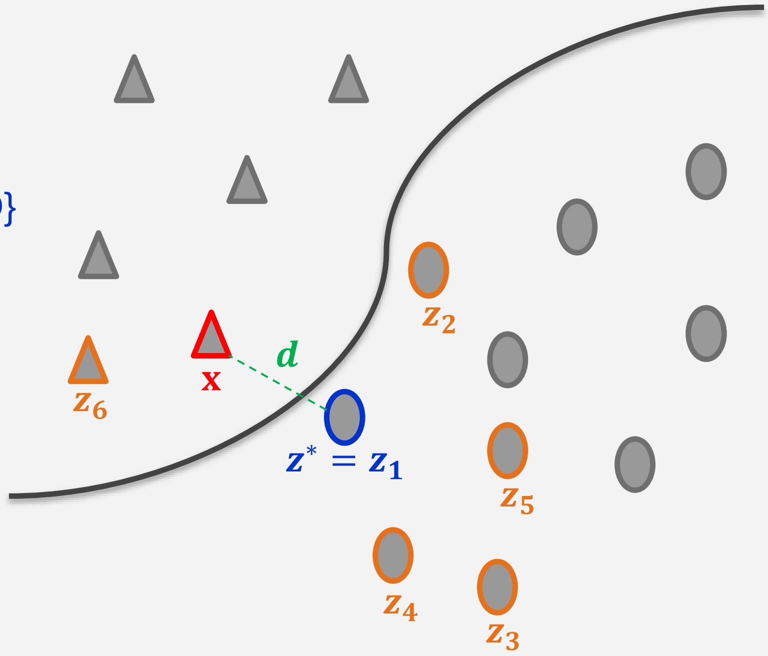
$z^* = x \setminus \{\text{Time Square, DUMBO}\}$

$E^* = \{\text{Time Square, DUMBO}\}$

original instance

perturbed instances

d cosine distance



WHY COMPLETE SEARCH FAILS

- Start with removing one feature and increase number of features in the subset until the predicted class changes
- Scales exponentially with active features m and required number of features k to be removed

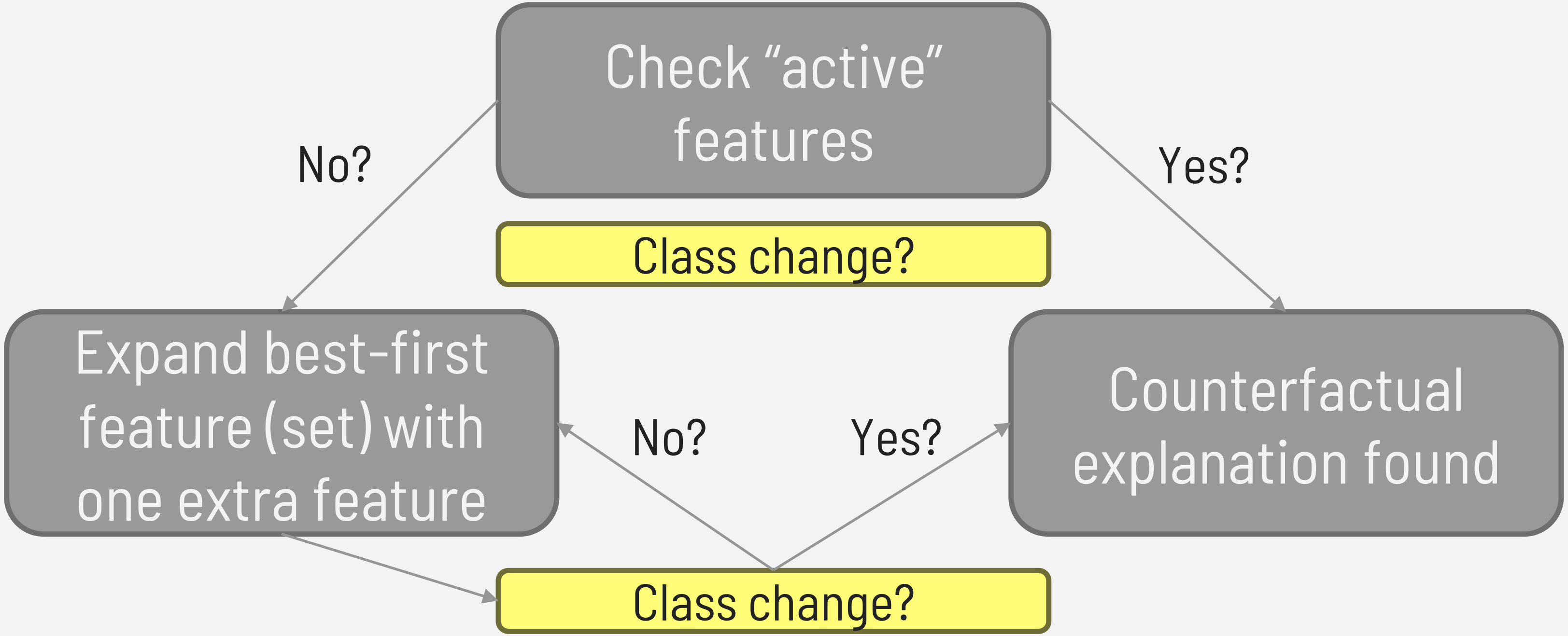
e.g., for an instance with m features, a combination of k features requires $\frac{m!}{(m-k)!k!}$ evaluations

BEST-FIRST SEARCH (SEDC)

- Explaining document classifications (Martens & Provost, 2013)
- Model-agnostic algorithm SEDC: heuristic best-first search
- Optimal for linear models

Implementation on https://github.com/yramon/edc

BEST-FIRST SEARCH (SEDC)



NOVEL HYBRID ALGORITHMS

Additive Feature Attribution (AFA) methods:

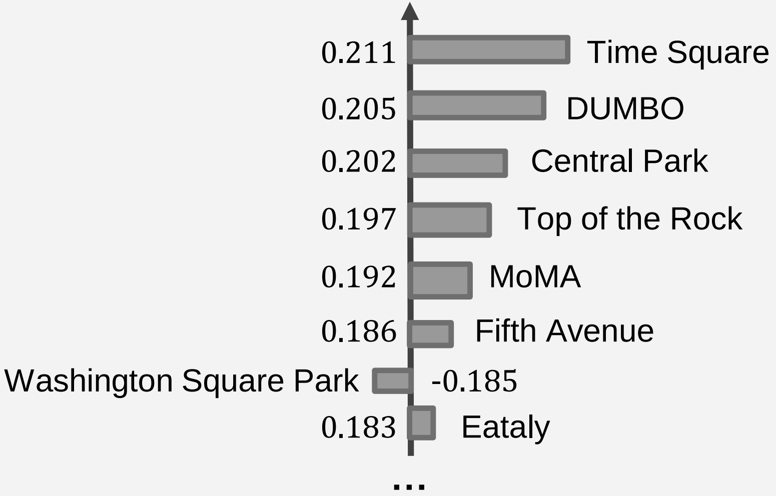
- LIME: Local Model-agnostic Explainer (Ribeiro et al., 2016)
- SHAP: Shapley Additive Explanations (Lundberg et al., 2018)

Output: Importance-ranked list

NOVEL HYBRID ALGORITHMS

LIME / SHAP

Example: Tourist prediction using NYC location data



NOVEL HYBRID ALGORITHMS

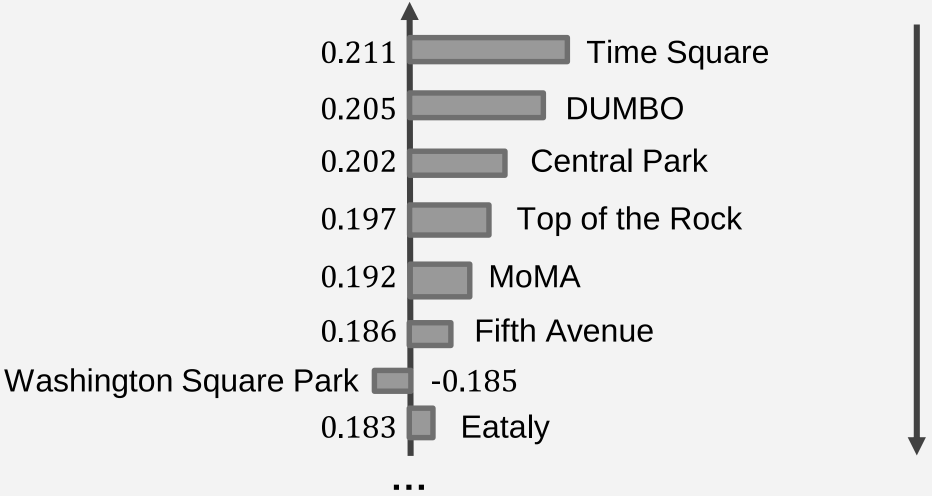
Originality: importance rankings may be an “intelligent” starting point for efficiently computing counterfactuals

$\Rightarrow$ Novel algorithms: **LIME–C** and **SHAP-C**

NOVEL HYBRID ALGORITHMS

LIME-C / SHAP-C

Example: Tourist prediction using NYC location data



Remove features with positive importance weight until the class changes

EXPERIMENTAL SETUP

Collect data sets and build models

Textual data: linear/rbf SVM

Behavioral data: LR/MLP

Table 1: **Data characteristics** of the data sets: data type (B:behavioral, T:textual), target variable, number of instances and features, imbalance b of the target, the sparsity p and the test set size (percentage of instances predicted as positive are placed in brackets). We use 20% of the data as test set. A * indicates that the number of positively predicted test instances used for the experiments was a random subset of 300. The average number of active features $\dot{m}_{lin}$ and $\dot{m}_{nonlin}$ are measured over the positively predicted test instances of respectively the linear and nonlinear model. The last column shows the reference. Note that we sort the data sets by increasing values of $\dot{m}_{lin}$.

Dataset	Type	Target	Instances	Features	b	p	Test set (%)	$\dot{m}_{lin}$	$\dot{m}_{nonlin}$	ref
Flickr*	B	comments	100,000	190,991	36.91%	99.99%	20,000 (20%)	2.02	2.96	[36]
Ecommerce*	B	gender	15,000	21,880	21.98%	99.99%	3,000 (15%)	2.60	2.67	[3]
Airline*	T	sentiment	14,640	5,183	16.14%	99.82%	2,928 (15%)	7.81	8.21	[2]
Twitter	T	topic	6,090	4,569	9.15%	99.74%	1,218 (10%)	9.52	9.35	[5]
Fraud*	B	fraudulent	858,131	107,345	6.4e-5%	99.99%	171,627(1%)	11.83	14.09	n.a.
YahooMovies*	B	gender	7,642	11,915	28.87%	99.76%	1,529 (20%)	25.24	25.00	[6]
TaFeng*	B	age	31,640	23,719	45.23%	99.90%	6,328 (15%)	44.32	37.24	[22]
KDD2015*	B	dropout	120,542	4,835	20.71%	99.67%	24,109 (20%)	49.01	46.40	[4]
20news	T	atheism	18,846	41,356	4.24%	99.84%	3,770 (5%)	67.96	62.77	[1]
Movielens_100k	B	gender	943	1,682	28.95%	93.69%	189 (25%)	68.73	73.42	[20]
Facebook*	B	gender	386,321	122,924	44.57%	99.94%	77,265 (30%)	83.03	84.55	[9]
Movielens_1m*	B	gender	6,040	3,706	28.29%	95.53%	1,208 (25%)	168.46	153.46	[20]
Libimseti*	B	gender	137,806	166,353	44.53%	99.93%	27,562 (30%)	229.16	226.97	[8]

Collect data sets and build models

Generate explanations for test instances

Textual data: linear/rbf SVM

Behavioral data: LR/MLP

SEDC

LIME-C

SHAP-C

Positively-predicted test instances

max. 2 minutes
max. 30 features

SEDC: max 50 iterations
LIME/SHAP-C: 5000 samples

EVALUATION CRITERIA

The goal is to find the minimum-sized counterfactual as fast as possible ➔ tradeoff between:

- **Effectiveness**
  - Percentage explained
  - Switching point: amount of features in explanation
- **Efficiency**
  - Computation time in seconds

Collect data sets and build models

Textual data: linear/rbf SVM

Behavioral data: LR/MLP

Generate explanations for test instances

SEDC

LIME-C

SHAP-C

Positively-predicted test instances

max. 2 minutes
max. 30 features

SEDC: max. 50 iterations
LIME/SHAP-C: 5000 samples

Evaluation

Percentage explained

Switching point

Computation time


RESULTS & CONCLUSION

EFFECTIVENESS

Table 2: Percentage explained (fraction of positively predicted instances for which a counterfactual smaller than 30 is found). For stochastic *LIME-C/SHAP-C*, these are average percentages over 5 runs. The best percentages are indicated in bold. The percentages are underlined if a method is significantly worse than the best method on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)
Flickr	**100**	99.33	99.33	**28.67**	28.33	24.33
Ecommerce	**100**	97.33	**100**	97.67	97.00	**99.67**
Airline	**100**	**100**	**100**	**100**	**100**	**100**
Twitter	**100**	**100**	**100**	**100**	**100**	**100**
Fraud	**100**	**100**	81.67	**100**	**100**	75.00
YahooMovies	**100**	**100**	**100**	98.67	**100**	**100**
TaFeng	**100**	**100**	**100**	93.33	**100**	**100**
KDD2015	**100**	**100**	**100**	99.67	**100**	97.67
20news	**100**	98.94	**100**	**100**	98.41	**100**
Movielens_100k	**100**	**100**	**100**	**100**	**100**	97.92
Facebook	**97.00**	95.33	92.67	70.33	**93.00**	87.67
Movielens_1m	**99.33**	99.00	96.67	90.00	**95.33**	**92.67**
Libimseti	**96.33**	91.00	89.00	78.00	**82.33**	70.67
Average	**99.44**	98.53	96.87	88.95	**91.88**	88.12
# wins	**13**	8	8	6	**10**	7

EFFECTIVENESS

Table 2: Percentage explained (fraction of positively predicted instances for which a counterfactual smaller than 30 is found). For stochastic *LIME-C/SHAP-C*, these are average percentages over 5 runs. The best percentages are indicated in bold. The percentages are underlined if a method is significantly worse than the best method on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)
Flickr	**100**	99.33	99.33	**28.67**	28.33	24.33
Ecommerce	**100**	97.33	**100**	97.67	97.00	**99.67**
Airline	**100**	**100**	**100**	**100**	**100**	**100**
Twitter	**100**	**100**	**100**	**100**	**100**	**100**
Fraud	**100**	**100**	<u>81.67</u>	**100**	**100**	<u>75.00</u>
YahooMovies	**100**	**100**	**100**	98.67	**100**	**100**
TaFeng	**100**	**100**	**100**	<u>93.33</u>	**100**	**100**
KDD2015	**100**	**100**	**100**	99.67	**100**	97.67
20news	**100**	98.94	**100**	**100**	98.41	**100**
Movielens_100k	**100**	**100**	**100**	**100**	**100**	97.92
Facebook	**97.00**	95.33	92.67	<u>70.33</u>	**93.00**	<u>87.67</u>
Movielens_1m	**99.33**	99.00	96.67	<u>90.00</u>	**95.33**	**92.67**
Libimseti	**96.33**	<u>91.00</u>	<u>89.00</u>	<u>78.00</u>	**82.33**	<u>70.67</u>
Average	**99.44**	98.53	96.87	88.95	**91.88**	88.12
# wins	**13**	8	8	6	**10**	7

EFFECTIVENESS

Table 2: Percentage explained (fraction of positively predicted instances for which a counterfactual smaller than 30 is found). For stochastic *LIME-C/SHAP-C*, these are average percentages over 5 runs. The best percentages are indicated in bold. The percentages are underlined if a method is significantly worse than the best method on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC (%)**	**LIME-C (%)**	**SHAP-C (%)**	**SEDC (%)**	**LIME-C (%)**	**SHAP-C (%)**
Flickr	**100**	99.33	99.33	**28.67**	28.33	24.33
Ecommerce	**100**	97.33	**100**	97.67	97.00	**99.67**
Airline	**100**	**100**	**100**	**100**	**100**	**100**
Twitter	**100**	**100**	**100**	**100**	**100**	**100**
Fraud	**100**	**100**	<u>81.67</u>	**100**	**100**	<u>75.00</u>
YahooMovies	**100**	**100**	**100**	98.67	**100**	**100**
TaFeng	**100**	**100**	**100**	<u>93.33</u>	**100**	**100**
KDD2015	**100**	**100**	**100**	99.67	**100**	97.67
20news	**100**	98.94	**100**	**100**	98.41	**100**
Movielens_100k	**100**	**100**	**100**	**100**	**100**	97.92
Facebook	**97.00**	95.33	92.67	<u>70.33</u>	**93.00**	<u>87.67</u>
Movielens_1m	**99.33**	99.00	96.67	<u>90.00</u>	**95.33**	**92.67**
Libimseti	**96.33**	<u>91.00</u>	<u>89.00</u>	<u>78.00</u>	**82.33**	<u>70.67</u>
Average	**99.44**	98.53	96.87	88.95	**91.88**	88.12
# wins	**13**	8	8	6	**10**	7

EFFECTIVENESS

Table 2: Percentage explained (fraction of positively predicted instances for which a counterfactual smaller than 30 is found). For stochastic *LIME-C/SHAP-C*, these are average percentages over 5 runs. The best percentages are indicated in bold. The percentages are underlined if a method is significantly worse than the best method on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)
Flickr	**100**	99.33	99.33	**28.67**	28.33	24.33
Ecommerce	**100**	97.33	**100**	97.67	97.00	**99.67**
Airline	**100**	**100**	**100**	**100**	**100**	**100**
Twitter	**100**	**100**	**100**	**100**	**100**	**100**
Fraud	**100**	**100**	<u>81.67</u>	**100**	**100**	<u>75.00</u>
YahooMovies	**100**	**100**	**100**	98.67	**100**	**100**
TaFeng	**100**	**100**	**100**	<u>93.33</u>	**100**	**100**
KDD2015	**100**	**100**	**100**	99.67	**100**	97.67
20news	**100**	98.94	**100**	**100**	98.41	**100**
Movielens_100k	**100**	**100**	**100**	**100**	**100**	97.92
Facebook	**97.00**	95.33	92.67	<u>70.33</u>	**93.00**	<u>87.67</u>
Movielens_1m	**99.33**	99.00	96.67	<u>90.00</u>	**95.33**	**92.67**
Libimseti	**96.33**	<u>91.00</u>	<u>89.00</u>	<u>78.00</u>	**82.33**	<u>70.67</u>
Average	**99.44**	98.53	96.87	88.95	**91.88**	88.12
# wins	**13**	8	8	6	**10**	7

EFFECTIVENESS

Table 2: Percentage explained (fraction of positively predicted instances for which a counterfactual smaller than 30 is found). For stochastic *LIME-C/SHAP-C*, these are average percentages over 5 runs. The best percentages are indicated in bold. The percentages are underlined if a method is significantly worse than the best method on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)	**SEDC** (%)	**LIME-C** (%)	**SHAP-C** (%)
Flickr	**100**	99.33	99.33	**28.67**	28.33	24.33
Ecommerce	**100**	97.33	**100**	97.67	97.00	**99.67**
Airline	**100**	**100**	**100**	**100**	**100**	**100**
Twitter	**100**	**100**	**100**	**100**	**100**	**100**
Fraud	**100**	**100**	81.67	**100**	**100**	75.00
YahooMovies	**100**	**100**	**100**	98.67	**100**	**100**
TaFeng	**100**	**100**	**100**	93.33	**100**	**100**
KDD2015	**100**	**100**	**100**	99.67	**100**	97.67
20news	**100**	98.94	**100**	**100**	98.41	**100**
Movielens_100k	**100**	**100**	**100**	**100**	**100**	97.92
Facebook	**97.00**	95.33	92.67	70.33	**93.00**	87.67
Movielens_1m	**99.33**	99.00	96.67	90.00	**95.33**	**92.67**
Libimseti	**96.33**	91.00	89.00	78.00	**82.33**	70.67
Average	**99.44**	98.53	96.87	88.95	**91.88**	88.12
# wins	**13**	8	8	6	**10**	7

EFFECTIVENESS

Table 3: Median and interquantile range of **switching point**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The switching point is measured over the subset of instances where *all* methods have found a switching point. The best (median) switching points are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

	Linear				Nonlinear			
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**Random**	**SEDC**	**LIME-C**	**SHAP-C**	**Random**
Flickr	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Ecommerce	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Airline	**1(1-2)**	**1(1-2)**	**1(1-2)**	$2(1-3)$	**1(1-1)**	**1(1-1)**	**1(1-1)**	$2(1-3)$
Twitter	**2(1-3)**	**2(1-3)**	**2(1-3)**	$3(2-5)$	**1(1-1)**	**1(1-1)**	**1(1-1)**	$3(2-5)$
Fraud	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**
YahooMovies	**2(1-4)**	**2(1-4)**	**2(1-4)**	$4(2-7)$	**1(1-3)**	$2(1-3)$	$2(1-3)$	$4(2-7)$
TaFeng	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5(3-11)$	**2(1-8)**	**2(1-3)**	**2(1-3.05)**	$6(3-11)$
KDD2015	**3(1-7)**	**3(1-7)**	**3(1-7)**	$8.5(3-17.25)$	**2(1-3)**	**2(1-3.25)**	**2(1-4)**	$4(3-17.25)$
20news	**2(1-4)**	**2(1-4)**	**2(1-4)**	$11(4-23.5)$	**1(1-3)**	**1(1-3)**	**1(1-3)**	$8(4-23.5)$
Movielens_100k	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5.5(3-10)$	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5(3-10)$
Facebook	**3(2-7)**	**3(2-7)**	**3(2-7)**	$8(4-18)$	$4(1-13)$	**2.8(1-4.2)**	$3(1.2-5.15)$	$9(4-18)$
Movielens_1m	**3(2-7)**	**3(2-7)**	**3(2-7)**	$8.5(4-18)$	**3(1-6)**	**3(1-6)**	**3(1-6)**	$7(4-18)$
Libimseti	**3(2-5.5)**	**3(2-5.7)**	**3(2-5.9)**	$28(13-48)$	**2(1-5)**	$4.2(2-8.8)$	$5.2(2.4-11.5)$	$19(13-48)$
# wins	**13**	**13**	**13**	3	**12**	11	10	3

EFFECTIVENESS

Table 3: Median and interquantile range of **switching point**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The switching point is measured over the subset of instances where *all* methods have found a switching point. The best (median) switching points are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

	Linear				**Nonlinear**			
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**Random**	**SEDC**	**LIME-C**	**SHAP-C**	**Random**
Flickr	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Ecommerce	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Airline	**1(1-2)**	**1(1-2)**	**1(1-2)**	2(1 – 3)	**1(1-1)**	**1(1-1)**	**1(1-1)**	2(1 – 3)
Twitter	**2(1-3)**	**2(1-3)**	**2(1-3)**	3(2 – 5)	**1(1-1)**	**1(1-1)**	**1(1-1)**	3(2 – 5)
Fraud	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**
YahooMovies	**2(1-4)**	**2(1-4)**	**2(1-4)**	4(2 – 7)	**1(1-3)**	2(1 – 3)	2(1 – 3)	4(2 – 7)
TaFeng	**2(1-4)**	**2(1-4)**	**2(1-4)**	5(3 – 11)	**2(1-8)**	**2(1-3)**	**2(1-3.05)**	6(3 – 11)
KDD2015	**3(1-7)**	**3(1-7)**	**3(1-7)**	8.5(3 – 17.25)	**2(1-3)**	**2(1-3.25)**	**2(1-4)**	4(3 – 17.25)
20news	**2(1-4)**	**2(1-4)**	**2(1-4)**	11(4 – 23.5)	**1(1-3)**	**1(1-3)**	**1(1-3)**	8(4 – 23.5)
Movielens_100k	**2(1-4)**	**2(1-4)**	**2(1-4)**	5.5(3 – 10)	**2(1-4)**	**2(1-4)**	**2(1-4)**	5(3 – 10)
Facebook	**3(2-7)**	**3(2-7)**	**3(2-7)**	8(4 – 18)	4(1 – 13)	**2.8(1-4.2)**	3(1.2 – 5.15)	9(4 – 18)
Movielens_1m	**3(2-7)**	**3(2-7)**	**3(2-7)**	8.5(4 – 18)	**3(1-6)**	**3(1-6)**	**3(1-6)**	7(4 – 18)
Libimseti	**3(2-5.5)**	**3(2-5.7)**	**3(2-5.9)**	28(13 – 48)	**2(1-5)**	4.2(2 – 8.8)	5.2(2.4 – 11.5)	19(13 – 48)
# wins	**13**	**13**	**13**	3	**12**	11	10	3

EFFECTIVENESS

Table 3: Median and interquantile range of **switching point**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The switching point is measured over the subset of instances where *all* methods have found a switching point. The best (median) switching points are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

	Linear				Nonlinear			
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**Random**	**SEDC**	**LIME-C**	**SHAP-C**	**Random**
Flickr	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Ecommerce	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-2)**
Airline	**1(1-2)**	**1(1-2)**	**1(1-2)**	$2(1-3)$	**1(1-1)**	**1(1-1)**	**1(1-1)**	$2(1-3)$
Twitter	**2(1-3)**	**2(1-3)**	**2(1-3)**	$3(2-5)$	**1(1-1)**	**1(1-1)**	**1(1-1)**	$3(2-5)$
Fraud	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**	**1(1-1)**
YahooMovies	**2(1-4)**	**2(1-4)**	**2(1-4)**	$4(2-7)$	**1(1-3)**	$2(1-3)$	$2(1-3)$	$4(2-7)$
TaFeng	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5(3-11)$	**2(1-8)**	**2(1-3)**	**2(1-3.05)**	$6(3-11)$
KDD2015	**3(1-7)**	**3(1-7)**	**3(1-7)**	$8.5(3-17.25)$	**2(1-3)**	**2(1-3.25)**	**2(1-4)**	$4(3-17.25)$
20news	**2(1-4)**	**2(1-4)**	**2(1-4)**	$11(4-23.5)$	**1(1-3)**	**1(1-3)**	**1(1-3)**	$8(4-23.5)$
Movielens_100k	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5.5(3-10)$	**2(1-4)**	**2(1-4)**	**2(1-4)**	$5(3-10)$
Facebook	**3(2-7)**	**3(2-7)**	**3(2-7)**	$8(4-18)$	$4(1-13)$	**2.8(1-4.2)**	$3(1.2-5.15)$	$9(4-18)$
Movielens_1m	**3(2-7)**	**3(2-7)**	**3(2-7)**	$8.5(4-18)$	**3(1-6)**	**3(1-6)**	**3(1-6)**	$7(4-18)$
Libimseti	**3(2-5.5)**	**3(2-5.7)**	**3(2-5.9)**	$28(13-48)$	**2(1-5)**	$4.2(2-8.8)$	$5.2(2.4-11.5)$	$19(13-48)$
# wins	**13**	**13**	**13**	3	**12**	11	10	3

EFFICIENCY

Table 4: Median and interquantile range of **computation time in seconds**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The computation time is measured over the subset of instances where *all* methods have found an explanation. The best (median) computation times are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

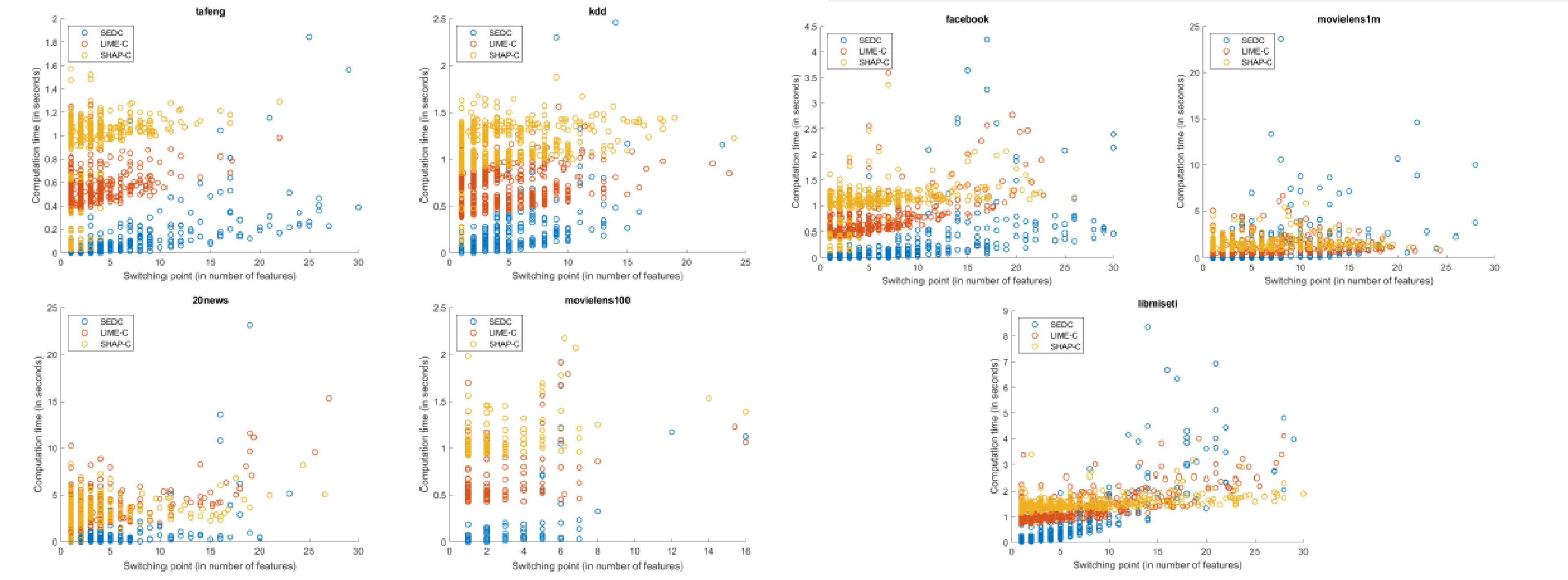
	Linear			Nonlinear		
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**SEDC**	**LIME-C**	**SHAP-C**
Flickr	**0.00(0.00-0.00)**	0.37(0.36 – 0.38)	0.07(0.07 – 0.08)	**0.00(0.00-0.00)**	0.39(0.39 – 0.42)	0.09(0.08 – 0.17)
Ecommerce	**0.00(0.00-0.00)**	0.82(0.77 – 0.86)	**0.06(0.05-0.07)**	**0.00(0.00-0.00)**	0.43(0.42 – 0.45)	0.04(0.03 – 0.04)
Airline	**0.00(0.00-0.02)**	0.97(0.84 – 1.09)	0.09(0.04 – 0.62)	**0.02(0.00-0.02)**	1.36(1.18 – 1.52)	0.13(0.04 – 0.84)
Twitter	**0.01(0.01-0.01)**	0.66(0.61 – 0.69)	0.22(0.07 – 0.49)	**0.01(0.00-0.01)**	0.69(0.65 – 0.71)	0.17(0.06 – 0.47)
Fraud	**0.00(0.00-0.00)**	0.44(0.41 – 0.46)	0.07(0.06 – 0.09)	**0.00(0.00-0.02)**	0.43(0.41 – 0.45)	0.08(0.07 – 0.15)
YahooMovies	**0.01(0.01-0.02)**	0.45(0.43 – 0.49)	0.94(0.87 – 0.99)	**0.05(0.03-0.12)**	1.88(1.82 – 1.96)	3.39(3.24 – 3.47)
TaFeng	**0.02(0.01-0.05)**	0.49(0.44 – 0.58)	1.03(0.98 – 1.08)	**0.02(0.00-0.06)**	0.49(0.45 – 0.58)	0.99(0.95 – 1.06)
KDD2015	**0.03(0.02-0.11)**	0.51(0.46 – 0.61)	1.03(0.97 – 1.07)	**0.06(0.03-0.16)**	0.81(0.75 – 0.91)	1.3(1.24 – 1.37)
20news	**0.13(0.03-0.44)**	3.34(2.11 – 4.35)	3.55(2.68 – 4.33)	**0.07(0.02-0.26)**	2.31(1.58 – 3.23)	2.61(2.09 – 3.22)
Movielens_100k	**0.02(0.02-0.07)**	0.51(0.47 – 0.74)	1.01(0.96 – 1.14)	**0.03(0.02-0.12)**	0.61(0.53 – 0.89)	1.10(1.03 – 1.31)
Facebook	**0.03(0.01-0.14)**	0.58(0.48 – 0.81)	1.09(1.03 – 1.19)	**0.04(0.01-0.19)**	0.54(0.48 – 0.63)	1.08(1.03 – 1.14)
Movielens_1m	**0.09(0.03-0.44)**	0.76(0.53 – 1.20)	1.15(1.01 – 1.47)	**0.13(0.03-0.48)**	0.79(0.61 – 1.14)	1.25(1.12 – 1.48)
Libimseti	**0.13(0.06-0.38)**	1.06(0.94 – 1.38)	1.38(1.3 – 1.55)	**0.16(0.08-0.39)**	1.04(0.93 – 1.29)	1.44(1.38 – 1.57)
# wins	13	0	0	13	0	0

EFFICIENCY

Table 4: Median and interquantile range of **computation time in seconds**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The computation time is measured over the subset of instances where *all* methods have found an explanation. The best (median) computation times are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**SEDC**	**LIME-C**	**SHAP-C**
Flickr	**0.00(0.00-0.00)**	0.37(0.36 – 0.38)	0.07(0.07 – 0.08)	**0.00(0.00-0.00)**	0.39(0.39 – 0.42)	0.09(0.08 – 0.17)
Ecommerce	**0.00(0.00-0.00)**	0.82(0.77 – 0.86)	**0.06(0.05-0.07)**	**0.00(0.00-0.00)**	0.43(0.42 – 0.45)	0.04(0.03 – 0.04)
Airline	**0.00(0.00-0.02)**	0.97(0.84 – 1.09)	0.09(0.04 – 0.62)	**0.02(0.00-0.02)**	1.36(1.18 – 1.52)	0.13(0.04 – 0.84)
Twitter	**0.01(0.01-0.01)**	0.66(0.61 – 0.69)	0.22(0.07 – 0.49)	**0.01(0.00-0.01)**	0.69(0.65 – 0.71)	0.17(0.06 – 0.47)
Fraud	**0.00(0.00-0.00)**	0.44(0.41 – 0.46)	0.07(0.06 – 0.09)	**0.00(0.00-0.02)**	0.43(0.41 – 0.45)	0.08(0.07 – 0.15)
YahooMovies	**0.01(0.01-0.02)**	0.45(0.43 – 0.49)	0.94(0.87 – 0.99)	**0.05(0.03-0.12)**	1.88(1.82 – 1.96)	3.39(3.24 – 3.47)
TaFeng	**0.02(0.01-0.05)**	0.49(0.44 – 0.58)	1.03(0.98 – 1.08)	**0.02(0.00-0.06)**	0.49(0.45 – 0.58)	0.99(0.95 – 1.06)
KDD2015	**0.03(0.02-0.11)**	0.51(0.46 – 0.61)	1.03(0.97 – 1.07)	**0.06(0.03-0.16)**	0.81(0.75 – 0.91)	1.3(1.24 – 1.37)
20news	**0.13(0.03-0.44)**	3.34(2.11 – 4.35)	3.55(2.68 – 4.33)	**0.07(0.02-0.26)**	2.31(1.58 – 3.23)	2.61(2.09 – 3.22)
Movielens_100k	**0.02(0.02-0.07)**	0.51(0.47 – 0.74)	1.01(0.96 – 1.14)	**0.03(0.02-0.12)**	0.61(0.53 – 0.89)	1.10(1.03 – 1.31)
Facebook	**0.03(0.01-0.14)**	0.58(0.48 – 0.81)	1.09(1.03 – 1.19)	**0.04(0.01-0.19)**	0.54(0.48 – 0.63)	1.08(1.03 – 1.14)
Movielens_1m	**0.09(0.03-0.44)**	0.76(0.53 – 1.20)	1.15(1.01 – 1.47)	**0.13(0.03-0.48)**	0.79(0.61 – 1.14)	1.25(1.12 – 1.48)
Libimseti	**0.13(0.06-0.38)**	1.06(0.94 – 1.38)	1.38(1.3 – 1.55)	**0.16(0.08-0.39)**	1.04(0.93 – 1.29)	1.44(1.38 – 1.57)
# wins	13	0	0	13	0	0

EFFICIENCY vs SWITCHING POINT



EFFICIENCY

Table 4: Median and interquantile range of **computation time in seconds**. For stochastic *LIME-C/SHAP-C*, this is the average median/range over 5 runs. The computation time is measured over the subset of instances where *all* methods have found an explanation. The best (median) computation times are indicated in bold. The values are underlined if a method is significantly worse than the best method (smallest median value) on a 1% significance level using a McNemar mid-p test [15].

	Linear			Nonlinear		
Dataset	**SEDC**	**LIME-C**	**SHAP-C**	**SEDC**	**LIME-C**	**SHAP-C**
Flickr	**0.00(0.00-0.00)**	0.37(0.36 – 0.38)	0.07(0.07 – 0.08)	**0.00(0.00-0.00)**	0.39(0.39 – 0.42)	0.09(0.08 – 0.17)
Ecommerce	**0.00(0.00-0.00)**	0.82(0.77 – 0.86)	**0.06(0.05-0.07)**	**0.00(0.00-0.00)**	0.43(0.42 – 0.45)	0.04(0.03 – 0.04)
Airline	**0.00(0.00-0.02)**	0.97(0.84 – 1.09)	0.09(0.04 – 0.62)	**0.02(0.00-0.02)**	1.36(1.18 – 1.52)	0.13(0.04 – 0.84)
Twitter	**0.01(0.01-0.01)**	0.66(0.61 – 0.69)	0.22(0.07 – 0.49)	**0.01(0.00-0.01)**	0.69(0.65 – 0.71)	0.17(0.06 – 0.47)
Fraud	**0.00(0.00-0.00)**	0.44(0.41 – 0.46)	0.07(0.06 – 0.09)	**0.00(0.00-0.02)**	0.43(0.41 – 0.45)	0.08(0.07 – 0.15)
YahooMovies	**0.01(0.01-0.02)**	0.45(0.43 – 0.49)	0.94(0.87 – 0.99)	**0.05(0.03-0.12)**	1.88(1.82 – 1.96)	3.39(3.24 – 3.47)
TaFeng	**0.02(0.01-0.05)**	0.49(0.44 – 0.58)	1.03(0.98 – 1.08)	**0.02(0.00-0.06)**	0.49(0.45 – 0.58)	0.99(0.95 – 1.06)
KDD2015	**0.03(0.02-0.11)**	0.51(0.46 – 0.61)	1.03(0.97 – 1.07)	**0.06(0.03-0.16)**	0.81(0.75 – 0.91)	1.3(1.24 – 1.37)
20news	**0.13(0.03-0.44)**	3.34(2.11 – 4.35)	3.55(2.68 – 4.33)	**0.07(0.02-0.26)**	2.31(1.58 – 3.23)	2.61(2.09 – 3.22)
Movielens_100k	**0.02(0.02-0.07)**	0.51(0.47 – 0.74)	1.01(0.96 – 1.14)	**0.03(0.02-0.12)**	0.61(0.53 – 0.89)	1.10(1.03 – 1.31)
Facebook	**0.03(0.01-0.14)**	0.58(0.48 – 0.81)	1.09(1.03 – 1.19)	**0.04(0.01-0.19)**	0.54(0.48 – 0.63)	1.08(1.03 – 1.14)
Movielens_1m	**0.09(0.03-0.44)**	0.76(0.53 – 1.20)	1.15(1.01 – 1.47)	**0.13(0.03-0.48)**	0.79(0.61 – 1.14)	1.25(1.12 – 1.48)
Libimseti	**0.13(0.06-0.38)**	1.06(0.94 – 1.38)	1.38(1.3 – 1.55)	**0.16(0.08-0.39)**	1.04(0.93 – 1.29)	1.44(1.38 – 1.57)
# wins	13	0	0	13	0	0

CONCLUSION

- **SEDC** most efficient and effective for small data instances, however
  - flaw in heuristic best-first for some nonlinear models

- **SHAP-C** overall good performance, however
  - problems with highly unbalanced data
  - computation time more sensitive to # active features than LIME-C
  - relatively worse effectiveness/efficiency

⇒ **LIME-C**: suitable alternative to SEDC because of good tradeoff
  - good effectiveness results for all data and models
  - low computation times
  - efficiency least sensitive to switching point

CONCLUSION

- **SEDC** most efficient and effective for small data instances, however
  - flaw in heuristic best-first for some nonlinear models

- **SHAP-C** overall good performance, however
  - problems with highly unbalanced data
  - computation time more sensitive to # active features than LIME-C
  - relatively worse effectiveness/efficiency

⇒ **LIME-C**: suitable alternative to SEDC because of good tradeoff
  - good effectiveness results for all data and models
  - low computation times
  - efficiency least sensitive to switching point

! Also addresses problem of setting complexity of LIME/SHAP explanation

PROJECT 2

IMPROVING THE COST OF EXPLAINABILITY FOR HIGH-DIMENSIONAL, SPARSE DATA USING METAFEATURES-BASED RULE-EXTRACTION

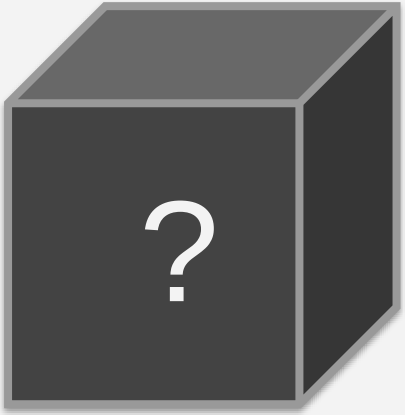
Yanou Ramon, David Martens, Theodoros Evgeniou, Stiene Praet

Submitted in Machine Learning (Special Issue on Feature Engineering)


PROBLEM
STATEMENT

	Columbia University	Time Square	DUMBO	…	Chelsea Market	*Target $\hat{y}$ Tourist*
Anna	1	1	1	…	0	***1***
Jack	1	0	0	…	1	***0***
…	…	…	…	…	…	***…***
Bill	0	0	1	…	0	***0***

LOCATION DATA NYC



"Black Box" model

$\Rightarrow$ **Thousands of coefficients**

$\Rightarrow$ **Nonlinear techniques**

$\hat{y} = 1$ ***if*** *tourist*

else $\hat{y} = 0$

(Global) comprehensibility issues
➔ Rule-extraction

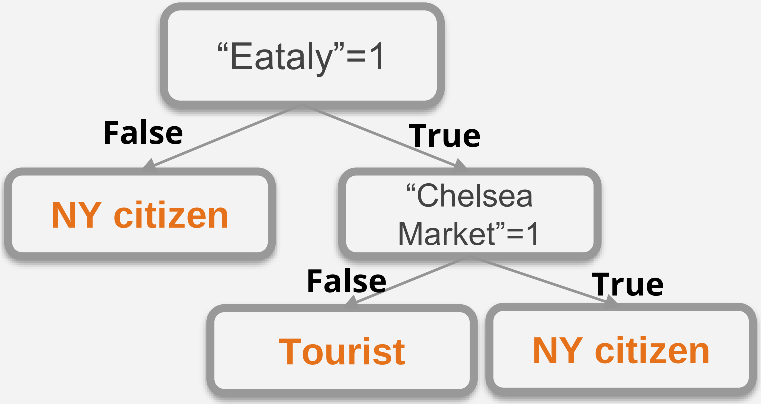
RULE-EXTRACTION

- **Train a comprehensible model ("white-box") to mimic the predictions of a more complex, highly accurate "black-box" model**
- Black-box model: all models on high-dimensional, sparse data
- Small decision trees and concise rule sets as "white-boxes"
- Black-box model predictions y^{BB} are used as new labels instead of the true labels y

RULE-EXTRACTION

	Columbia University	Time Square	DUMBO	…	Chelsea Market	*Target $\hat{y}$ Tourist*
Anna	1	1	1	…	0	***1***
Jack	1	0	0	…	1	***0***
…	…	…	…	…	…	***…***
Bill	0	0	1	…	0	***0***

LOCATION DATA NYC



Explains global classification behaviour over entire instance/feature space

CHALLENGES FOR HIGH-DIMENSIONAL, SPARSE DATA

Existing research focuses on low-dimensional, dense data

Challenges

1. **Complexity of extracted rules**
2. Computational complexity
3. Fine-grained feature comprehensibility

CHALLENGES FOR HIGH-DIMENSIONAL, SPARSE DATA

Existing research focuses on low-dimensional, dense data

Challenges

1. **Complexity of extracted rules**
2. Computational complexity
3. Fine-grained feature comprehensibility

➔ It is questionable whether the original fine-grained (FG) features are the best representation to achieve high explanation quality. This motivates our approach to use **"metafeatures"**.

METAFEATURES

Address sparsity of fine-grained features by mapping FG data onto a higher-level MF representation: $h(x): X_{FG} \rightarrow X_{MF} \subset \mathbb{R}^k$

Desired properties

1. Low dimensionality
2. High density
3. Faithfulness
4. Mutual exclusivity
5. Semantic comprehensibility

GENERATING METAFEATURES

Big Behavioral & Text Data	Metafeatures
Social media data (e.g., Facebook “Likes”)	Categories of Facebook “Likes” (e.g., Humor, Music, Art)
Transaction data	Spending categories (e.g., Gambling, Gift Shops)
Location data	Regions/venue types (e.g., Concert halls, Sports venues)
Textual data	Topics
Movie viewing data	Movie genres
Web browsing data	Words on a page/categories of URLs

Domain-based metafeatures vs data-driven metafeatures

MAIN CLAIM

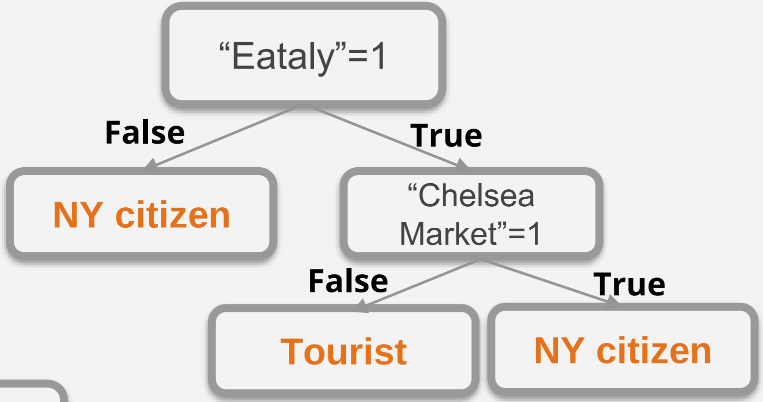
"Metafeatures" are more appropriate (↑ fidelity, ↑ stability) for extracting comprehensible rules from classifiers that are trained on high-dimensional, sparse data than the original fine-grained features

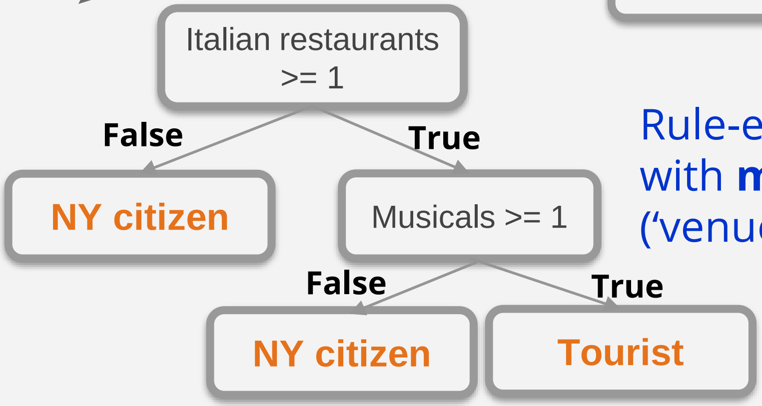
RULE-EXTRACTION

	Columbia University	Time Square	DUMBO	…	Chelsea Market	*Target $\hat{y}$ Tourist*
Anna	1	1	1	…	0	***1***
Jack	1	0	0	…	1	***0***
…	…	…	…	…	…	***…***
Bill	0	0	1	…	0	***0***

LOCATION DATA NYC

Rule-extraction with **fine-grained features**





Rule-extraction with **metafeatures** ('venue types')

PROPOSED METHODOLOGY

PROPOSED METHODOLOGY

1. Build classification model C_{BB} from labeled training data $\{X_{FG,train}, Y_{train}\}$
2. Predict labels y^{BB} for all data instances (train, test, validation)
3. Generate metafeatures X_{MF}
4. Extract cognitively simple rules using X_{MF} and y^{BB}
5. Evaluate the quality of explanation rules (fidelity, stability, accuracy)

GENERATING METAFEATURES

- Domain-based $X_{DomainMF}$
- Data-driven approach X_{DDMF}
  → approach based on Non-negative Matrix Factorization
  parameter of X_{DDMF} is k (number of generated metafeatures)
  $k \in [10, 1000]$

COGNITIVELY SIMPLE RULE-EXTRACTION

- CART decision tree algorithm (Scikit-learn library in Python)
- Based on Gini impurity
- Max. tree depth of 5 (~32 rules) in line with cognitive simplicity arguments and cognitive load theory

EVALUATION CRITERIA

- **Fidelity**: how well does the explanation model C_{WB} (extracted rules) approximate the underlying model C_{BB}?

(**“cost of explainability”**: 100% - fidelity is the loss in fidelity when replacing the black-box with an explanation model)

- **Explanation stability**: how stable is the explanation model over different training sessions with (slightly) different training sets?
- **Accuracy**: how well does the explanation model predict true labels y?


EXPERIMENTAL SETUP

DATA

Table 1 Characteristics of the data sets: data type (Type: behavioral/textual), classification task (Target), number of instances (Instances), number of features (Features), number of domain-based metafeatures (DomainMF), balance of the target b (fraction of instances with a "positive" class label), and sparsity of the data p (fraction of zero feature values in the data matrix).

Dataset	Type	Target	Instances	Features	DomainMF	b	p
Facebook	B	gender	6,733	5,357	50	32.42%	98.19%
Movielens1m	B	gender	6,040	3,883	18	28.29%	95.76%
Yahoomovies	B	gender	7,642	11,915	n.a.	71.13%	99.76%
Movielens100	B	gender	943	1,682	n.a.	71.05%	93.69%
Tafeng	B	gender	31,640	23,719	n.a.	45.23%	99.90%
Libimseti	B	gender	137,806	166,353	n.a.	44.53%	99.93%
20news	T	topic	18,846	41,356	n.a.	4.24%	99.87%
Airline	T	sentiment	14,640	5,183	n.a.	16.14%	99.82%
Flickr	B	comments	100,000	190,991	n.a.	36.91%	99.99%

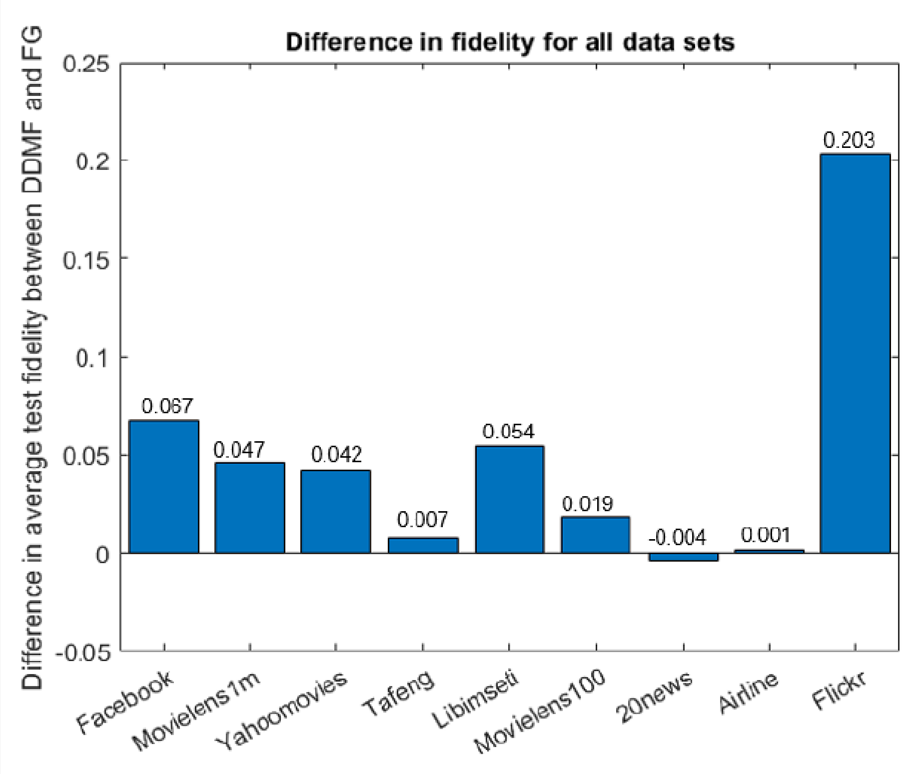
PREDICTION MODELS

Table 2 Performance of black-box classification models: accuracy, f-score, precision and recall. The last column shows the optimal hyperparameter value (regularization parameter C for L2-LR).

Dataset	accuracy	f-score	precision	recall	$\mathrm{HP}_{\mathrm{opt}}$
Facebook	85.97%	78.35%	79.91%	76.85%	0.01
Movielens1m	78.06%	61.31%	60.69%	61.95%	0.01
Yahoomovies	76.78%	83.51%	82.70%	84.33%	0.1
Tafeng	67.69%	64.98%	67.59%	62.55%	0.1
Libimseti	93.05%	92.53%	99.97%	86.11%	0.001
Movielens100	73.55%	81.48%	82.71%	80.29%	0.1
20news	96.66%	61.11%	60.74%	61.49%	100
Airline	89.58%	66.96%	64.51%	69.59%	1
Flickr	81.22%	75.36%	79.61%	71.54%	10

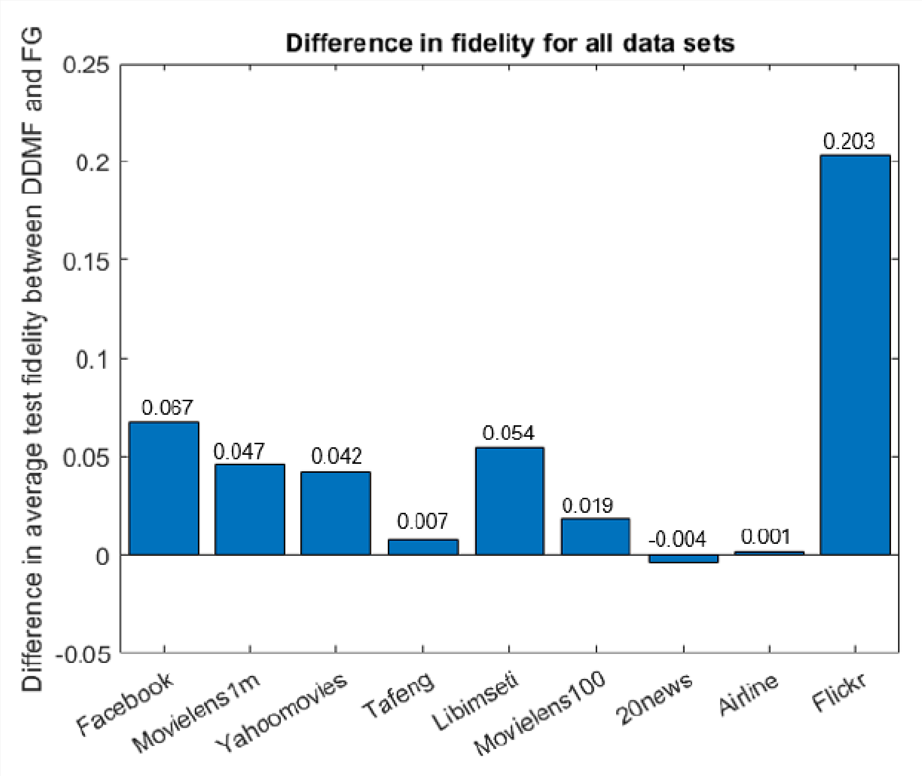

RESULTS & CONCLUSION

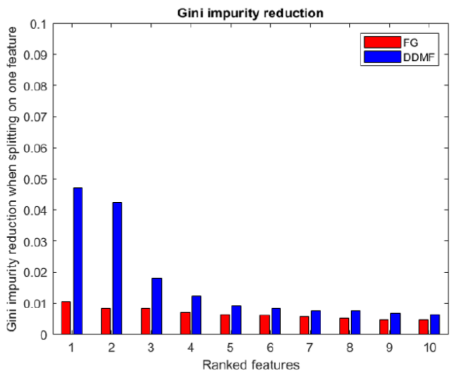
FIDELITY



Difference in fidelity for all data sets
Difference in average test fidelity between DDMF and FG
0.25
0.2
0.15
0.1
0.05
0
-0.05
0.067
0.047
0.042
0.007
0.054
0.019
-0.004
0.001
0.203
Facebook
Movielens1m
Yahoomovies
Tafeng
Libimseti
Movielens100
20news
Airline
Flickr

FIDELITY



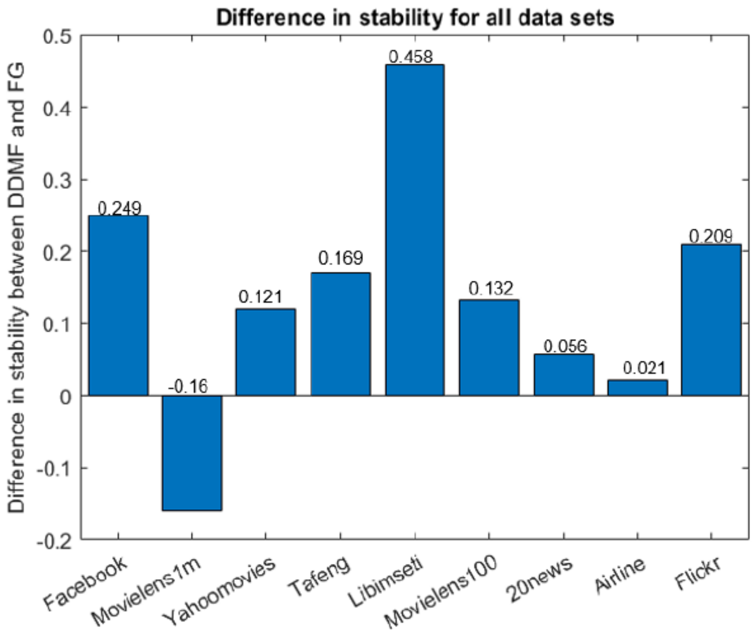


(g)

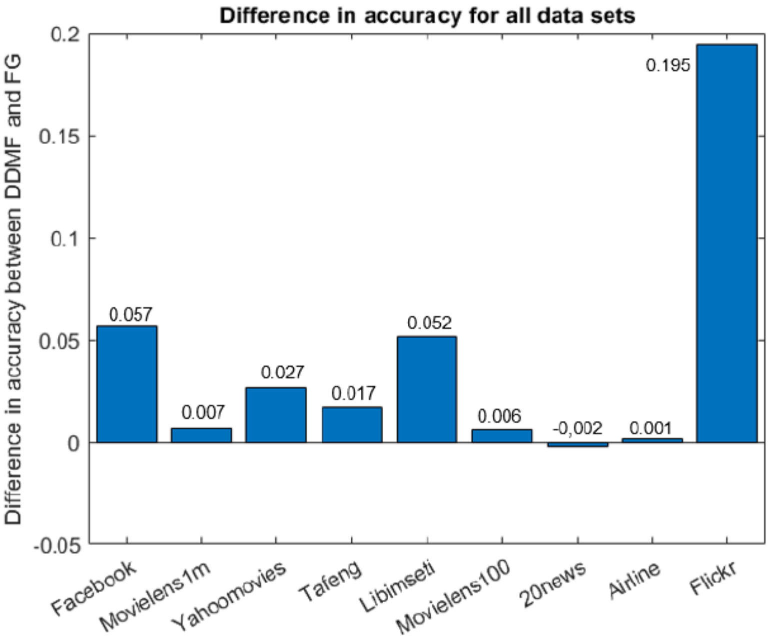
Flickr

Correlation between Gini impurity reduction ratio of best FG vs best MF and difference in fidelity: 0.929

STABILITY - ACCURACY



(c)



(d)

CONCLUSION

- Metafeatures-based rule-extraction leads to **better tradeoffs**:
  - Improved "cost of explainability": small trees/rules that explain a large(r) percentage of black-box predictions
    +5% fidelity, +15% stability, +5% accuracy

- Important tradeoff: increasing the complexity leads to increased fidelity but decreased stability

- Finetune k (or any *other* parameter of explanation model C_{WB}) to get desired fidelity/stability tradeoff


KEY TAKEAWAYS

OVERVIEW OF PROJECTS

I. **Deep Learning for Big, Sparse, Behavioral data**
De Cnudde et al., Big Data (2019)

II. **Instance-level explanation algorithms on behavioural and textual data: a counterfactual-oriented comparison**
Ramon et al., *Forthcoming in Advances in Data Analysis and Classification* (2020)

III. **Improving the cost of explainability for high-dimensional, sparse data using metafeatures-based rule-extraction**
Ramon et al., *Submitted to Machine Learning* (2020)

OVERVIEW OF PROJECTS

I. **Deep Learning for Big, Sparse, Behavioral data**
De Cnudde et al., Big Data (2019)

II. **Instance-level explanation algorithms on behavioural and textual data: a counterfactual-oriented comparison**
Ramon et al., *Forthcoming in Advances in Data Analysis and Classification* (2020)

→ **SEDC is most effective/efficient for data with small instances**
→ **LIME-C algorithm is a good alternative to SEDC algorithm for large data instances**

III. **Improving the cost of explainability for high-dimensional, sparse data using metafeatures-based rule-extraction**
Ramon et al., *Submitted to Machine Learning* (2020)

OVERVIEW OF PROJECTS

I. **Deep Learning for Big, Sparse, Behavioral data**
De Cnudde et al., Big Data (2019)

II. **Instance-level explanation algorithms on behavioural and textual data: a counterfactual-oriented comparison**
Ramon et al., *Forthcoming in Advances in Data Analysis and Classification* (2020)

III. **Improving the cost of explainability for high-dimensional, sparse data using metafeatures-based rule-extraction**
Ramon et al., *Submitted to Machine Learning* (2020)

→ **Metafeatures-based rule-extraction improves a key "cost of explainability": higher fidelity compared to rules using fine-grained features**

THANKS!

Further questions?

Mail: yanou.ramon@uantwerp.be
www.linkedin.com/in/yanou-ramon
https://yramon.github.io/
www.applieddatamining.com