Flexibly Mining Better Subgroups

Hoang Vu Nguyen
Jilles Vreeken

UNIVERSITÄT DES SAARLANDES

M2CI CLUSTER OF EXCELLENCE

max planck institut informatik

Question of the day

How can we **efficiently** discover the globally **optimal** cut points for **any** subgroup discovery objective function?

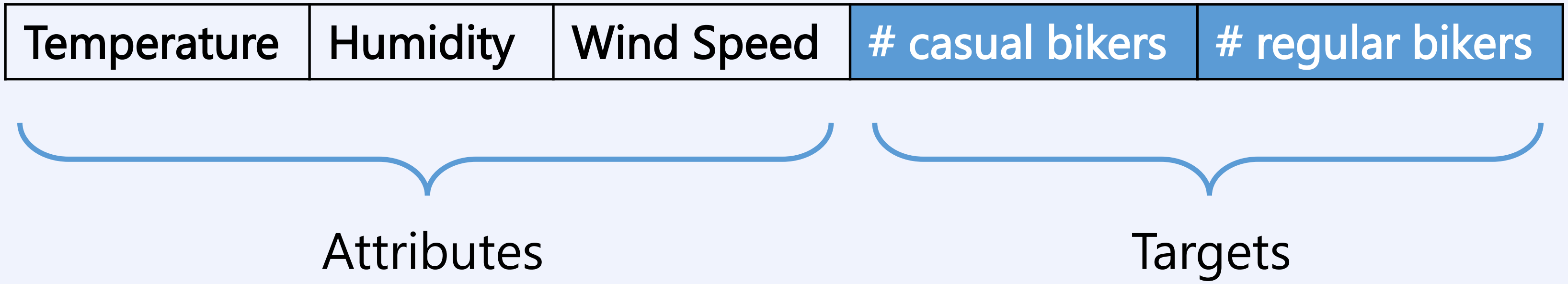
Question of the day

How can we **efficiently** discover the ~~globally~~ **optimal** cut points for **any** subgroup discovery objective function?

Question of the day

How can we efficiently discover the locally optimal cut points for any subgroup discovery objective function?

Subgroup Discovery



Temperature | Humidity | Wind Speed | # casual bikers | # regular bikers

Attributes

Targets

Find conditions on attributes such that distribution of the targets on the conditioned data is different from that of the global data

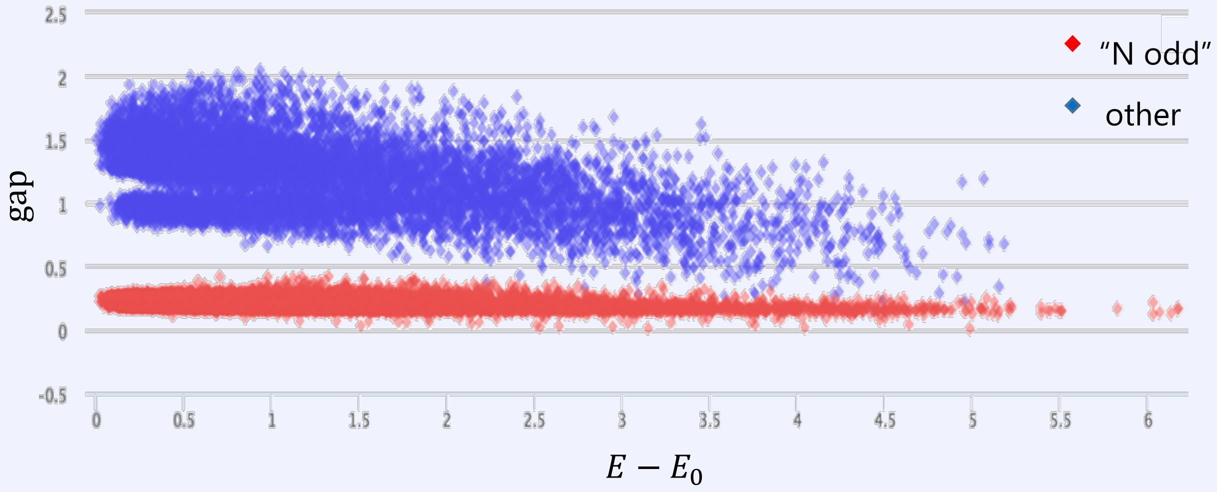
For example

- when $Temperature \leq 6$ there are fewer bikers than usual
- when $20 \leq Temperature \leq 25$ and $65 \leq Humidity \leq 75$ there are more bikers than usual

Example Subgroup

The number of gold atoms in a micro-cluster strongly determines its homo-lumo gap

$Condition = \{\text{“}N \text{ odd”}\}$



(together with Mario Boley, work in progress)

Binary Features

A condition on an attribute is essentially a binary feature

- subgroup discovery essentially relies on feature construction

For nominal data, extracting binary features is easy

- there are only $2^{|dom(A)|}$ features for each attribute A, after all

For numeric or ordinal data, this is much harder

- there are 2^n possible features for each attribute A
- standard approach is to simply use k equi-width or height bins

Eye of the beholder

	Univariate			Multivariate		
Measure	**Nominal**	**Ordinal**	**Numeric**	**Nominal**	**Ordinal**	**Numeric**
WRAcc	✓	✓	-	-	-	-
z-score	-	-	✓	-	-	-
Kullback-Leibler	✓	✓	-	✓	✓	-
Hellinger distance	✓	✓	-	✓	✓	-
Quadratic divergence	-	✓	✓	-	✓	✓

There exist very many quality measures

- each with specific properties, for target-specific data types

Discovering subgroups

Very complicated combinatorial problem

- humonguous search space
  all possible conditions on all possible attributes
- unstructured search space
  useful objective functions are not monotone/submodular

Standard approach

- naively binarise your data
- sample or search to discover top-k best subgroups

Discovering subgroups

Very complicated combinatorial problem

- humonguous search space
  all possible conditions on all possible attributes
- unstructured search space
  useful objective functions are not monotone/submodular

Standard approach

- naively binarise your data
- sample or search to discover top-k best subgroups

What we fix
in this paper

Quality measures

	Univariate			Multivariate		
Measure	**Nominal**	**Ordinal**	**Numeric**	**Nominal**	**Ordinal**	**Numeric**
WRAcc	✓	✓	-	-	-	-
z-score	-	-	✓	-	-	-
Kullback-Leibler	✓	✓	-	✓	✓	-
Hellinger distance	✓	✓	-	✓	✓	-
Quadratic divergence	-	✓	✓	-	✓	✓

Quality measures are highly specific to problem settings

- can we define a general and efficient algorithm to find cut points?

FLEXI

For attribute A, discover the **binary features**, i.e. grid g, that gives **maximal average quality** for objective ϕ

$$\arg\max_{g \in \mathcal{F}} \frac{1}{|g|} \sum_{i=1}^{|g|} \phi(b_g^i)$$

This leaves $|\mathcal{F}| = O(2^n)$ grids to evaluate...

- luckily, the search space is **structured**

(we also consider maximal **total** quality, but this leads to worse results)

Structure in space

Let g be the optimal partitioning of attribute A into k bins.

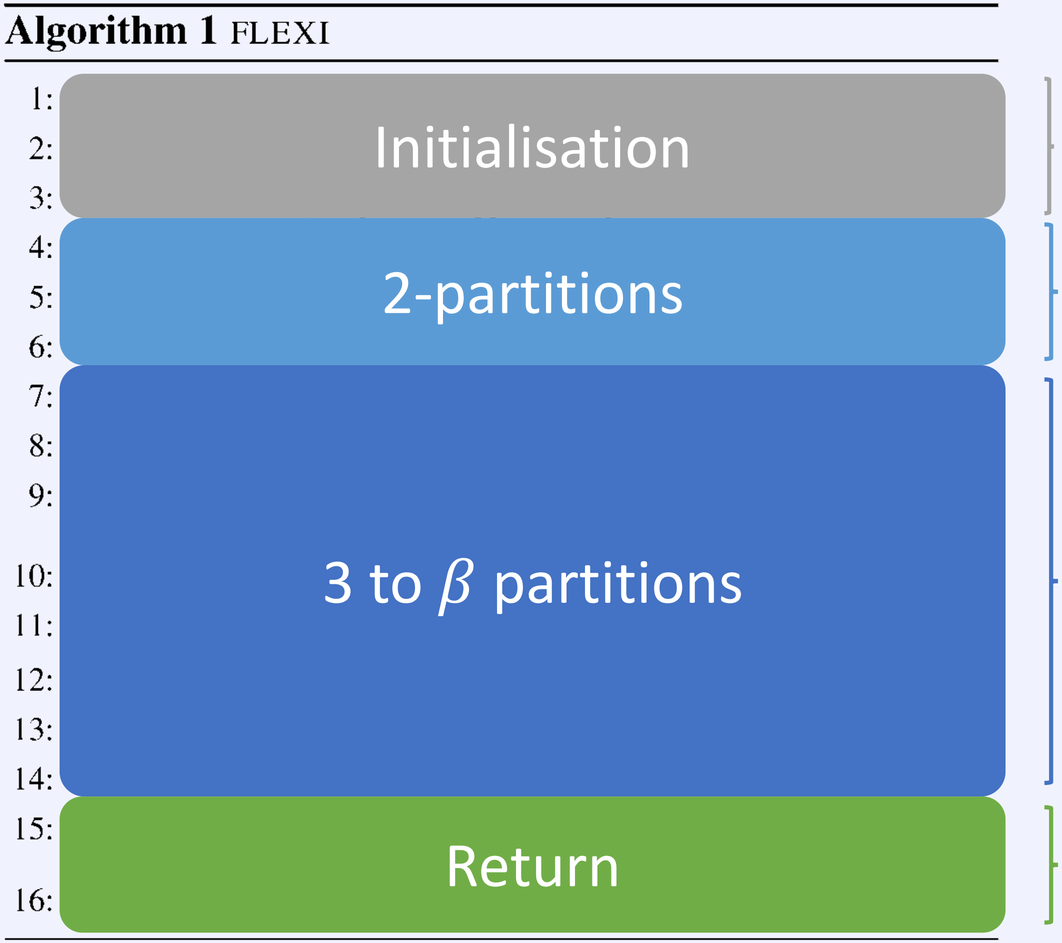
We observe

$$\sum_{i=1}^{k} \phi(b_g^i) \quad = \quad \phi\left(b_g^k\right) + \sum_{i=1}^{k-1} \phi(b_g^i)$$

This means that $\{b_g^1, \ldots, b_g^{k-1}\}$ is
the optimal partitioning of $A \leq l_g^k$ into $k-1$ bins.

We can use dynamic programming!

FLEXI, the algorithm



Algorithm 1 FLEXI

Initialisation

2-partitions

3 to β partitions

Return

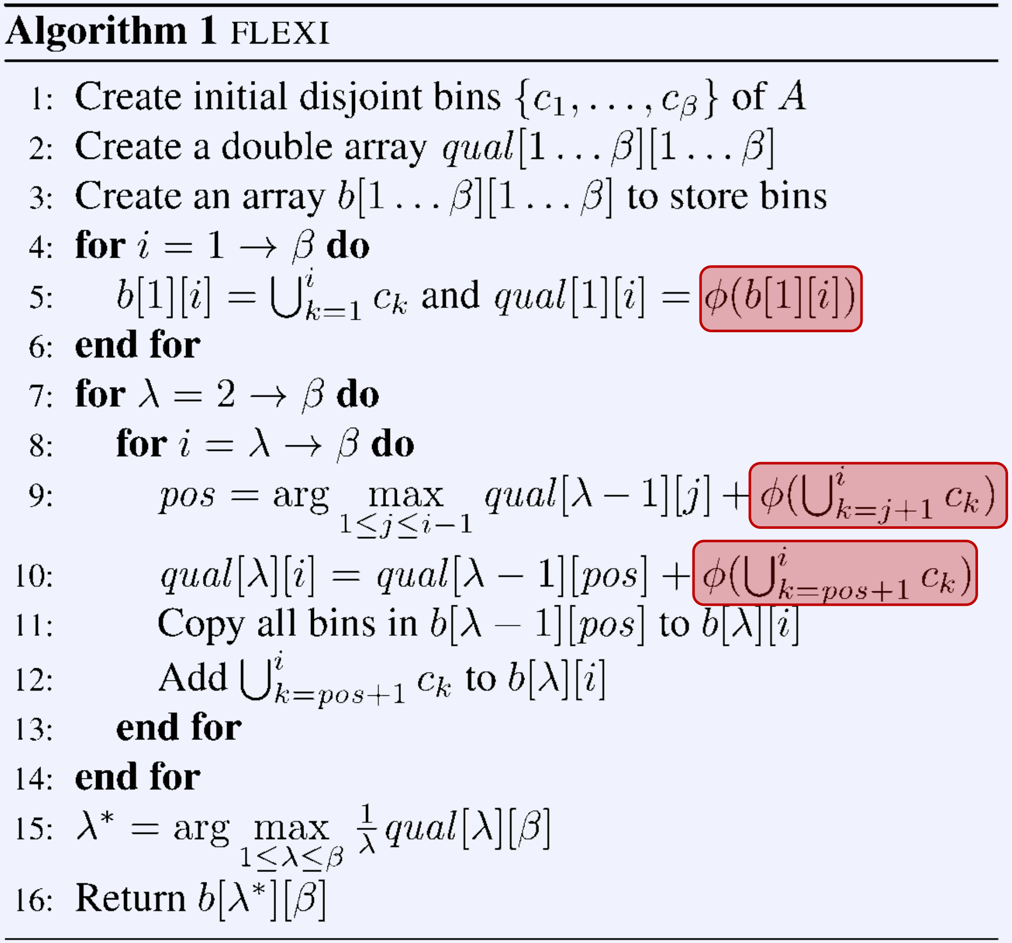
Initialise $\beta \ll |D|$ micro-bins

Compute 2-partition scores

Use dynamic programming to compute best scores for partitions into 3 to β parts

Identify and return best result

FLEXI, the algorithm



Algorithm 1 FLEXI

```
Create initial disjoint bins $\{c_1, \dots, c_\beta\}$ of $A$
Create a double array $qual[1 \dots \beta][1 \dots \beta]$
Create an array $b[1 \dots \beta][1 \dots \beta]$ to store bins
for $i = 1 \to \beta$ do
    $b[1][i] = \bigcup_{k=1}^{i} c_k$ and $qual[1][i] = \boxed{\phi(b[1][i])}$
end for
for $\lambda = 2 \to \beta$ do
    for $i = \lambda \to \beta$ do
        $pos = \arg \max_{1 \le j \le i-1} qual[\lambda - 1][j] + \boxed{\phi(\bigcup_{k=j+1}^{i} c_k)}$
        $qual[\lambda][i] = qual[\lambda - 1][pos] + \boxed{\phi(\bigcup_{k=pos+1}^{i} c_k)}$
        Copy all bins in $b[\lambda - 1][pos]$ to $b[\lambda][i]$
        Add $\bigcup_{k=pos+1}^{i} c_k$ to $b[\lambda][i]$
    end for
end for
$\lambda^* = \arg \max_{1 \le \lambda \le \beta} \frac{1}{\lambda} qual[\lambda][\beta]$
Return $b[\lambda^*][\beta]$
```

FLEXI can be used with any quality function ϕ

To ensure efficiency, we need a smart way to compute $\phi\left(\bigcup_{k=j}^{i} c_k\right)$

For five measures we show how to do this

Instantiating FLEXI$_w$

Weighted Relative Accuracy

- standard quality measure for single binary target

$$WRAcc(S) = \frac{s}{n}\left(\frac{s_+}{s} - \frac{n_+}{n}\right)$$

Compares the ratios of positive samples $\frac{s_+}{s}$ within subgroup S to that of the whole data, $\frac{n_+}{n}$

How can we efficiently pre-compute $WRAcc\left(\bigcup_{k=j}^{i} c_k\right)$?

Instantiating FLEXI$_w$

Pre-computing Weighted Relative Accuracies

```
1)  for $i = 1 \rightarrow \beta$ do
    $count[i]$ = number of positive labels in $D_{c_i}$
    compute $WRAcc(c_i)$ based on $count[i]$
2)  for $i = 2 \rightarrow \beta$ do
        $\theta = count[i]$
        for $j = i - 1 \rightarrow 1$ do
            $\theta = \theta + count[j]$
            set # of positive labels in $\bigcup_{k=j}^{i} c_k$ to $\theta$
            compute $WRAcc(\bigcup_{k=j}^{i} c_k)$
```

$O(n)$

$O(\beta^2)$

Done!

Instantiating FLEXI

We show how to instantiate

FLEXI$_w$	with WRAcc	at $O(n + \beta^2)$
FLEXI$_z$	with Z-score	at $O(n + \beta^2)$
FLEXI$_h$	with Hellinger distance	at $O(n\beta^2 d)$
FLEXI$_k$	with Kullback Leibler	at $O(n\beta^2 d)$
FLEXI$_q$	with quadratic divergence	at $O(n^2 d)$

As β is typically small, between 5 to 40,
the first four scale linear in n

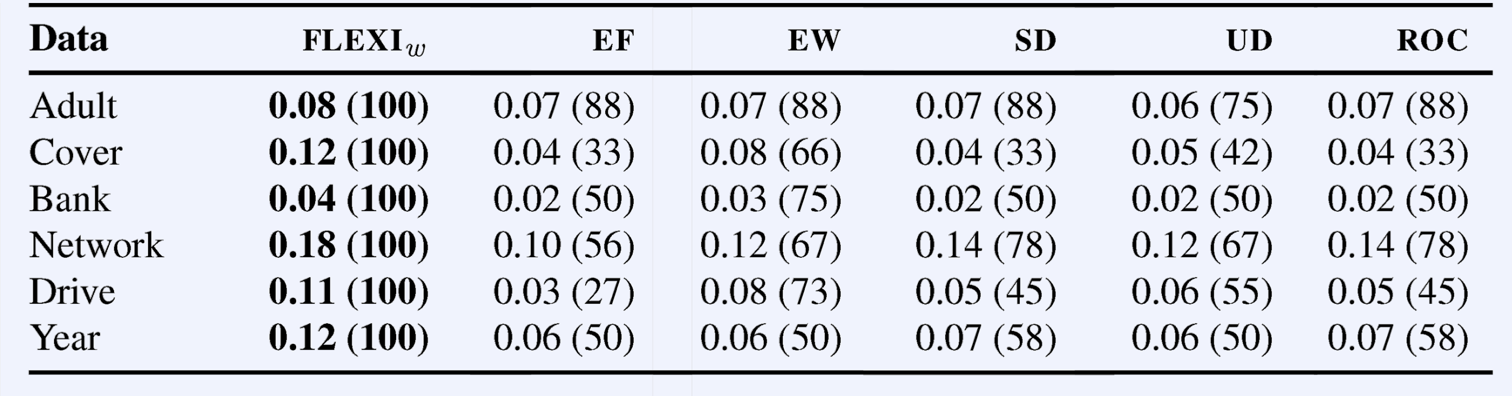
(see the paper for details on how to speed up FLEXI$_q$ using sampling)

Experiments

Experiments show that FLEXI outperforms the state of the art in **quality**, **flexibility**, and **efficiency**.

Experiments

Experiments show that FLEXI outperforms the state of the art in quality, flexibility, and efficiency.

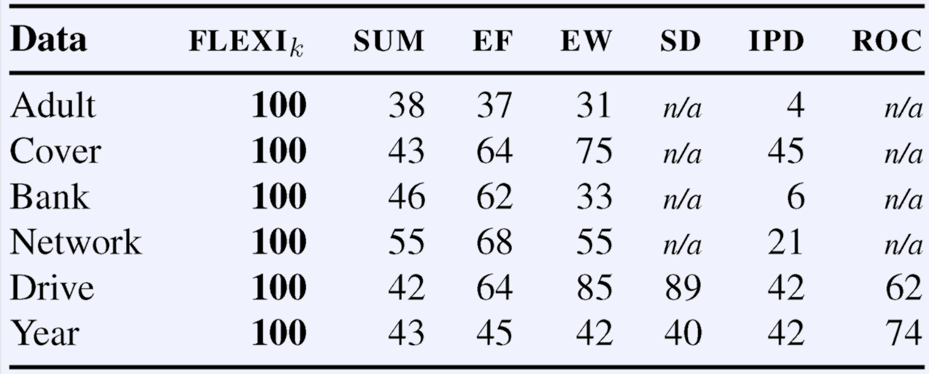


Data	**FLEXI$_w$**	**EF**	**EW**	**SD**	**UD**	**ROC**
Adult	**0.08 (100)**	0.07 (88)	0.07 (88)	0.07 (88)	0.06 (75)	0.07 (88)
Cover	**0.12 (100)**	0.04 (33)	0.08 (66)	0.04 (33)	0.05 (42)	0.04 (33)
Bank	**0.04 (100)**	0.02 (50)	0.03 (75)	0.02 (50)	0.02 (50)	0.02 (50)
Network	**0.18 (100)**	0.10 (56)	0.12 (67)	0.14 (78)	0.12 (67)	0.14 (78)
Drive	**0.11 (100)**	0.03 (27)	0.08 (73)	0.05 (45)	0.06 (55)	0.05 (45)
Year	**0.12 (100)**	0.06 (50)	0.06 (50)	0.07 (58)	0.06 (50)	0.07 (58)

Average quality for top-50 subgroups
(WRAcc)

Experiments

Experiments show that FLEXI outperforms the state of the art in quality, flexibility, and efficiency.

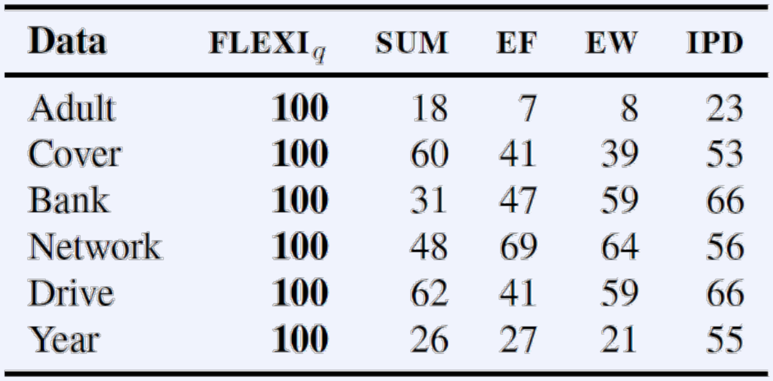


Data	FLEXI$_k$	SUM	EF	EW	SD	IPD	ROC
Adult	**100**	38	37	31	*n/a*	4	*n/a*
Cover	**100**	43	64	75	*n/a*	45	*n/a*
Bank	**100**	46	62	33	*n/a*	6	*n/a*
Network	**100**	55	68	55	*n/a*	21	*n/a*
Drive	**100**	42	64	85	89	42	62
Year	**100**	43	45	42	40	42	74

Average quality for top-50 subgroups
(Kullback-Leibler divergence)

Experiments

Experiments show that FLEXI outperforms the state of the art in quality, flexibility, and efficiency.

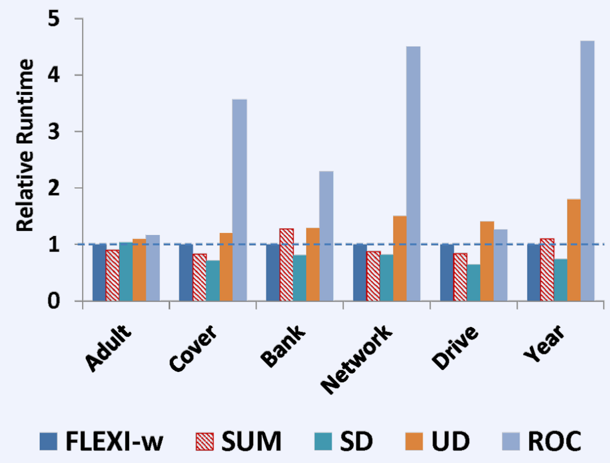


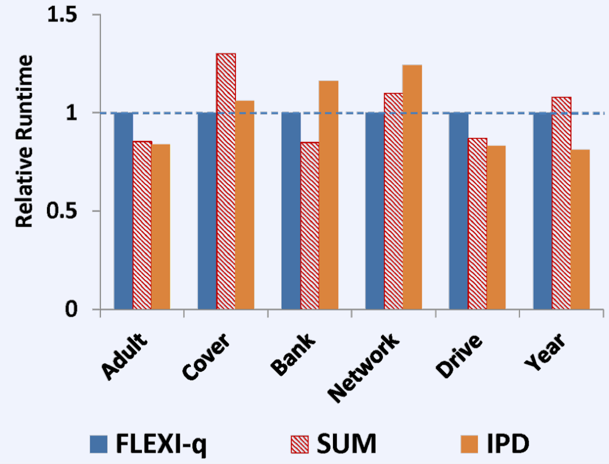
Data	FLEXI$_q$	SUM	EF	EW	IPD
Adult	**100**	18	7	8	23
Cover	**100**	60	41	39	53
Bank	**100**	31	47	59	66
Network	**100**	48	69	64	56
Drive	**100**	62	41	59	66
Year	**100**	26	27	21	55

Average quality for top-50 subgroups
(Quadratic divergence)

Experiments

Experiments show that FLEXI outperforms the state of the art in quality, flexibility, and efficiency.





Relative runtime to mine top-50 subgroups

Conclusions

We studied how to efficiently discover
high quality binary features for subgroup discovery

In short, FLEXI

- discovers binary features with maximal average quality
- highly flexible, operates with any objective function
- efficient due to dynamic programming
- complexity depends on ϕ, yet often linear in size of the data

Future work

- feature construction to allow sampling high quality subgroups

(source code available at: eda.mmci.uni-saarland.de/flexi)

Thank you!

We studied how to efficiently discover
high quality binary features for subgroup discovery

In short, FLEXI

- discovers binary features with maximal average quality
- highly flexible, operates with any objective function
- efficient due to dynamic programming
- complexity depends on ϕ, yet often linear in size of the data

Future work

- feature construction to allow sampling high quality subgroups

(source code available at: eda.mmci.uni-saarland.de/flexi)