Discrimination Discovery

The discrimination discovery task at a glance

Given a large database of historical decision records,

find discriminatory situations and practices.

S. Ruggieri, D. Pedreschi and F. Turini (2010). DCUBE: Discrimination discovery in databases. In SIGMOD, pp. 1127-1130.

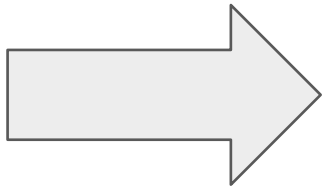
Discrimination discovery scenario

INPUT

Database of historical *decision records*

A *criterion* of (unlawful) discrimination

A set of *potentially discriminated* groups



OUTPUT

A subset of *decision records* and *potentially discriminated* people for which the *criterion* holds

D. Pedreschi, S. Ruggieri and F. Turini (2008). Discrimination-Aware Data Mining. In KDD, pp. 560-568.

The German credit score dataset

A small dataset used in many papers about discrimination
(like Zachary's karate club for networks people)

N = 1,000 records of bank account holders

Class label: good/bad creditor (grant or deny a loan)

Attributes: *numeric/interval-scaled*: duration of loan, amount requested, number of installments, age of requester, existing credits, number of dependents; *nominal*: result of past credits, purpose of credit, personal status, other parties, residence since, property magnitude, housing, job, other payment plans, own telephone, foreign worker; *ordinal*: checking status, saving status, employment

German credit score dataset: https://archive.ics.uci.edu/ml/datasets/Statlog+(German+Credit+Data)

Defining potentially discriminated (PD) groups

A subset of attribute values are **perceived as potentially discriminatory** based on background knowledge.

Potentially discriminated groups are people with those attribute values.

Examples:

- Women (misogyny)
- Ethnic minority (*racism*) or minority language
- Specific age range (*ageism*)
- Specific sexual orientation (*homophobia*)

Discrimination and combinations of attribute values

Discrimination can be a result of several joint characteristics (attribute values) which are not discriminatory by themselves

Thus, the object of discrimination should be described by a conjunction of attribute values:

Known as Itemsets

Association and classification rules

Association rules are if/then statements that help uncover relationships between seemingly unrelated data in a relational database.

In a classification rule, Y is a class item and X contains no class items.

$$\mathbf{X \rightarrow Y}$$

Definition: Association Rule

Let **D** be database of transactions e.g.

Transaction ID	Items
2000	A, B, C
1000	A, C
4000	A, D
5000	B, E, F

- Let ***I*** be the set of items that appear in the database, e.g., ***I = {A,B,C,D,E,F}***
- A **rule** is defined by **X → Y**, where **X⊂*I***, **Y⊂*I***, and **X∩Y=∅**
  - e.g.: **{B,C} → {A}** is a rule

Definition: Association Rule

- **Association Rule**
  - An implication expression of the form **X** → **Y**, where **X** and **Y** are non-overlapping itemsets
  - Example:
    {Milk, Diaper} → ***{Beer}***

- **Rule Evaluation Metrics**
  - **Support (s)**
    - Fraction of transactions that contain both **X** and **Y**
  - **Confidence (c)**
    - Measures how often items in **Y** appear in transactions that contain **X**

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Example:

$$\{\text{Milk}, \text{Diaper}\} \rightarrow \text{Beer}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

Computing support and confidence

TID	date items_bought	
100	10/10/99	{F,A,D,B}
200	15/10/99 {D,A,C,E,B}	
300	19/10/99	{C,A,B,E}
400	20/10/99	{B,A,D}

What is the ***support*** and ***confidence*** of the rule: {B,D} → {A}

- Support:
  - percentage of tuples that contain {A,B,D} = 75%
- Confidence:

$$\frac{\text{number of tuples that contain } \{A,B,D\}}{\text{number of tuples that contain } \{B,D\}} = 100\%$$

Association-rule mining task

Given a set of transactions **D**, the goal of association rule mining is to find **all** rules having

- support ≥ ***minsup*** threshold
- confidence ≥ ***minconf*** threshold

Beyond the scope of the current course!

Direct discrimination

Direct discrimination implies rules or procedures that impose ‘disproportionate burdens’ on minorities

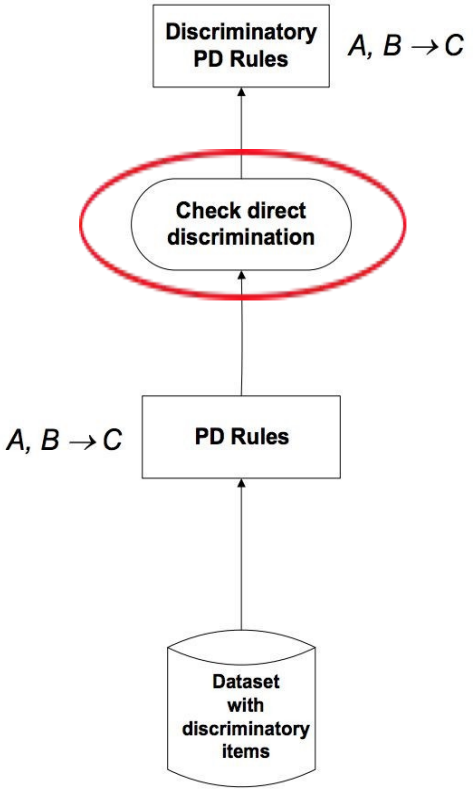
Potentially Discriminatory (PD) rules are any classification rule of the form:

A, B → C

where A is a PD group (B is called a "context")

Example:

gender="female", saving_status="no known savings"
→ credit=no



Favoritist PD rules

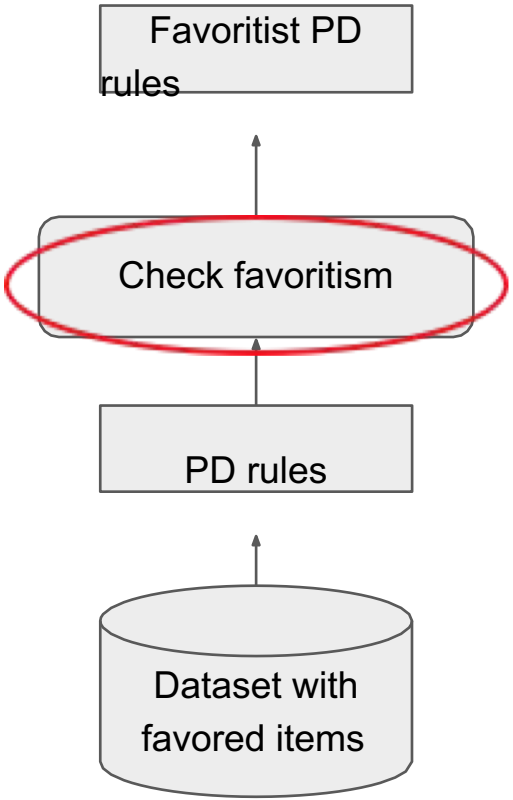
Is unveiled by looking at PD rules of the form

A, B → C

where C grants some benefit and A refers to a favored group.

Example:
gender="male", savings="no known savings"

→ credit=yes



Indirect discrimination

Indirect discrimination implies rules or procedures that impose ‘disproportionate burdens’ on minorities, though not explicitly using discriminatory attributes

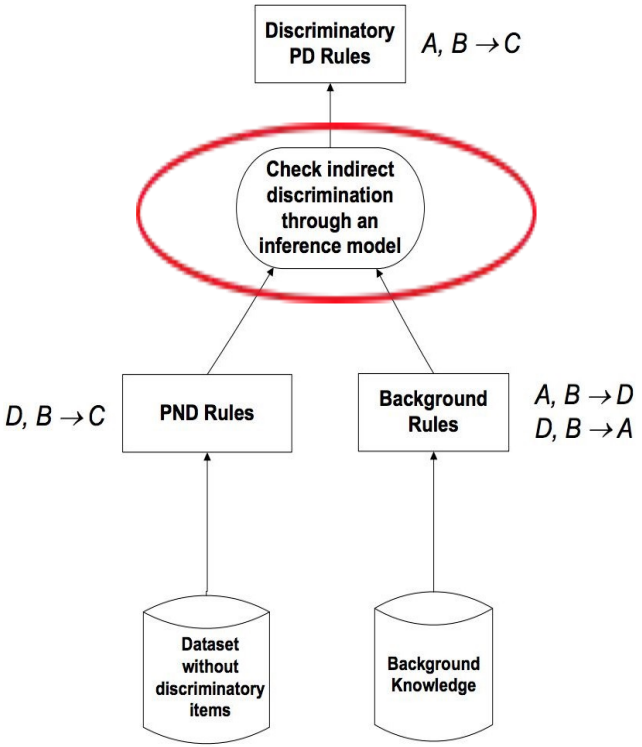
Potentially non-discriminatory (PND) rules may unveil discrimination, and are of the form:

D, B → C where D is a PND group

Example:

neighborhood="10451", city="NYC"

→ credit=no



Indirect discrimination example

Suppose we know that with high confidence:
(a) neighborhood=10451, city=NYC → benefit=deny

But we also know that with high confidence:
(b) neighborhood=10451, city=NYC → race=black

Hence:
(c) race=black, neighborhood=10451, city=NYC → benefit=deny

Rule (b) is background knowledge that allows us to infer (c), which shows that
rule (a) is indirectly discriminating against blacks

Evaluating PD rules through the extended lift

Remembering that conf(X → Y) = support(X → Y) / support(X)

We define the **extended lift with respect to B** of rule A, B → C as:

$$\text{elift}_B(A, B \rightarrow C) = \text{conf}(A, B \rightarrow C) / \text{conf}(B \rightarrow C)$$

The rules we care about are PD rules such that:

- A is a protected group (e.g. female, black)
- B is a context (e.g. lives in San Francisco)
- C is an outcome (usually negative, e.g., deny a loan)

The concept of α-protection

For a given threshold α, we say that PD rule A, B → C, involving a PD group A in a context B for an outcome C, is α-protective if:

$elift_B(A, B \rightarrow C) = conf(A, B \rightarrow C) / conf(B \rightarrow C) < \alpha$

Otherwise, when $elift_B(A, B \rightarrow C) >= \alpha$, then we say that A, B → C is an α-discriminatory rule

Relation of α-protection and group representation

For a given threshold α, we say that PD rule A, B → C,
involving a PD group A in a context B for a (usually bad) outcome C,
is α-protective if:

$$\text{elift}_B(A, B \rightarrow C) = \text{conf}(A, B \rightarrow C) / \text{conf}(B \rightarrow C) \leq \alpha$$

Note that:

$$\text{elift}_B(A, B \rightarrow C) = \text{elift}_B(B, C \rightarrow A) = \text{conf}(B, C \rightarrow A) / \text{conf}(B \rightarrow A)$$

This means extended lift is the ratio between the proportion of the disadvantaged group A in context B for (bad) outcome C, over the overall proportion of A in B.

Direct discrimination example

Rule (a):

city="NYC"
→ benefit=deny
with confidence 0.25

Rule (b):

race="black", city="NYC"
→ benefit=deny
with confidence 0.75 **elift 3.0**

Additional (discriminatory) element increases the rule confidence up to 3 times.

According to *α*-protection method, if the threshold *α*=3 is fixed then the rule (b) is classified as discriminatory

Real-world example from German credit dataset

Fixing α=3:

(B) saving status = "no known savings" → credit = deny conf. 0.18

(A) personal status = "female div/sep/mar",
saving status = "no known savings" → credit = deny conf. 0.27 **elift 1.52**

Rule A is α-protective.

Real-world example 2 from German credit dataset

Fixing α=3:

(B) purpose = "used car" → credit = deny — conf. 0.17

(A) age = "52.6+", personal status = "female div/sep/mar", purpose = "used car" → credit = deny — conf. 1.00 **elift 6.06**

Rule A is α-discriminatory.

Genuine occupational requirements

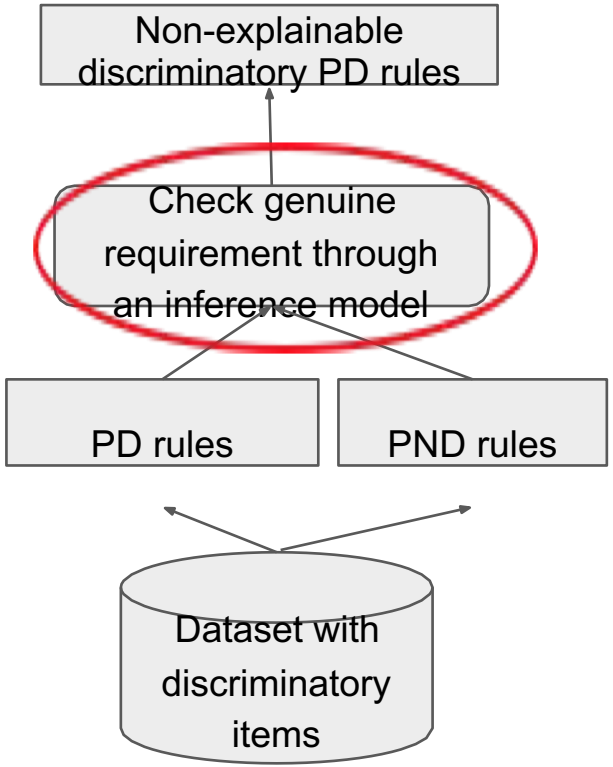
Supported by a PD rule of the form

$$A, B \rightarrow C$$

where C denies some benefit, we search for PND rules of the form

$$D, B \rightarrow C$$

such that D is a legitimate requirement, having the same effects of the PD rule



D. Pedreschi, S. Ruggieri and F. Turini (2009). Integrating induction and deduction for finding evidence of discrimination. In Proc. of International Conference on Artificial Intelligence and Law (pp. 157-166). ACM.

Example: genuine occupational requirement

(a) [A] gender="female", [B] city="NYC" → [C] hire=no conf. 0.58

(b) [D] drive_truck="false", [B] city="NYC" → [C] hire=no conf. 0.81

(c) [A] gender="female", [B] city="NYC" → [D] drive_truck=false conf. 0.91

Let $p \in [0, 1]$. Classification rule (a) A, B → C with A being a PD attribute, is a p.instance of a PND rule (b) D, B → C, if:

- D is a legitimate ground for the decision (i.e., accepted by law),
- conf(D, B→ C) ≥ p · conf(A, B → C), and
- conf(A, B → D) ≥ p.

If p is close to 1, there is no discrimination!

D. Pedreschi, S. Ruggieri and F. Turini (2009). Integrating induction and deduction for finding evidence of discrimination. In Proc. of International Conference on Artificial Intelligence and Law (pp. 157-166). ACM.

Limitations of classification rules approach

Legal limitations

Measuring group discrimination by aggregated values over **undifferentiated groups** is opposable in a court of law.

Take the context of women as the protected group and job hiring as the benefit. Approaches using aggregated values mix decisions for people that may be very different as per skills required for the job.

For example, do women have the same characteristics of men they are compared with? Or do they differ as per skills or other legally admissible reasons?

Limitations of classification rules approach (cont.)

Interpretational limitations

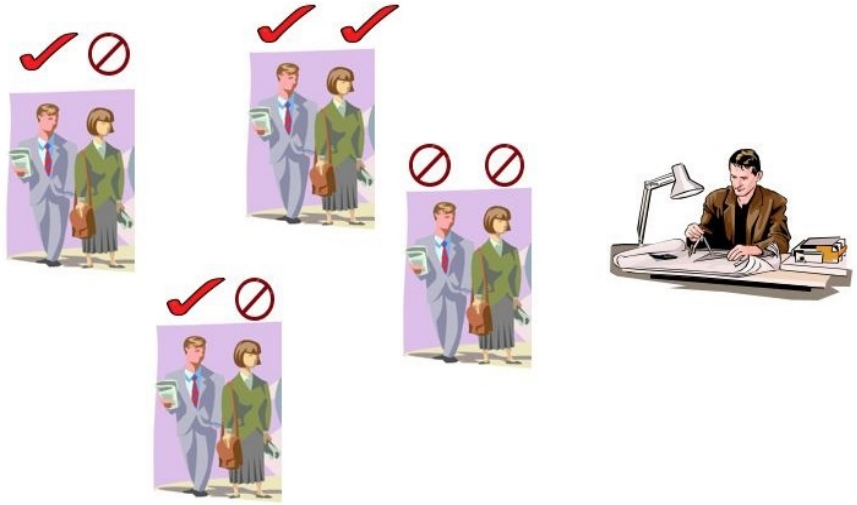
- The result of the knowledge discovery process is a large set of classification rules, which provide local niches of possible discrimination.
- No global description of who is discriminated and who is not.

Technical limitations

- Due to the use of frequent itemset mining, it can only consider nominal attributes and nominal decisions
- Interval-scaled attributes (age, income) and decisions (loan rate, wage) must be discretized as a pre-processing step.

Situation testing

- Legal approach for creating controlled experiments
- Matched pairs undergo the same situation, e.g. apply for a job
- Same characteristics apart from the discrimination ground (a black and a white, a male and a female etc.)



Idea: k-NN as situation testing

Given past decision records, for each member of the protected group with a negative decision outcome, k-NN is applied to search testers with similar, legally admissible, characteristics.

If decision outcomes between the testers of protected and unprotected groups are different, then there is discrimination.

k-NN as situation testing

Input: a dataset *R* of decision records

- For r ∈ *R*, dec(r) is the decision (discrete or continuous)
    - E.g., dec(r) is grant-benefit or deny-benefit
- P(R) is the set of protected-by-law groups, e.g., women
    - E.g., P(R) = {r ∈ *R* | r[gender]=female}
- U(R) = R \ P(R) is the rest of the dataset, e.g., men

Relax the "identical characteristics" of situation testing to a "similar characteristics" by using a distance function *d*

Distance function in k-NN

Distance *d(a,b)* is defined over attributes that are legally admissible for the purpose of taking the decision

$$d(\mathbf{r},\mathbf{s}) = \frac{\sum_{i=1}^{n} d_i(\mathbf{r}_i,\mathbf{s}_i)}{n}$$

Interval-scaled values are first standardized using the z-score $z_i(x) = (x - m_i)/s_i$ where m_i is the mean value.

Then distance between x,y is measured by the absolute difference of their z-scores:

$$d_i(x,y) = |z_i(x) - z_i(y)|.$$

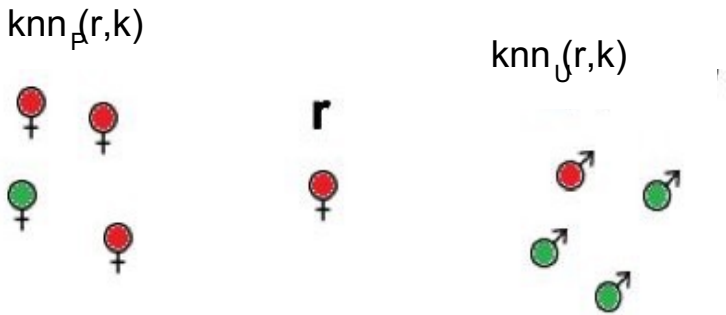
For nominal domains, distance is a binary function testing equality:

$$d_i(x,y) = 0 \text{ if } x = y, \text{ and } d_i(x,y) = 1 \text{ otherwise}$$

k-NN as situation testing (the algorithm)

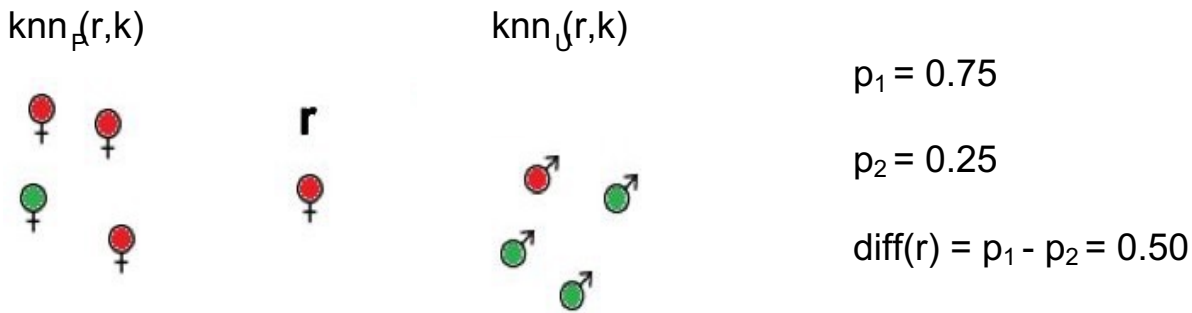
For r ∈ P(R), look at its k closest neighbors

- ... in the protected set
  - define p_1 = proportion with the same decision as r
- … in the unprotected set
  - define p_2 = proportion with the same decision as r

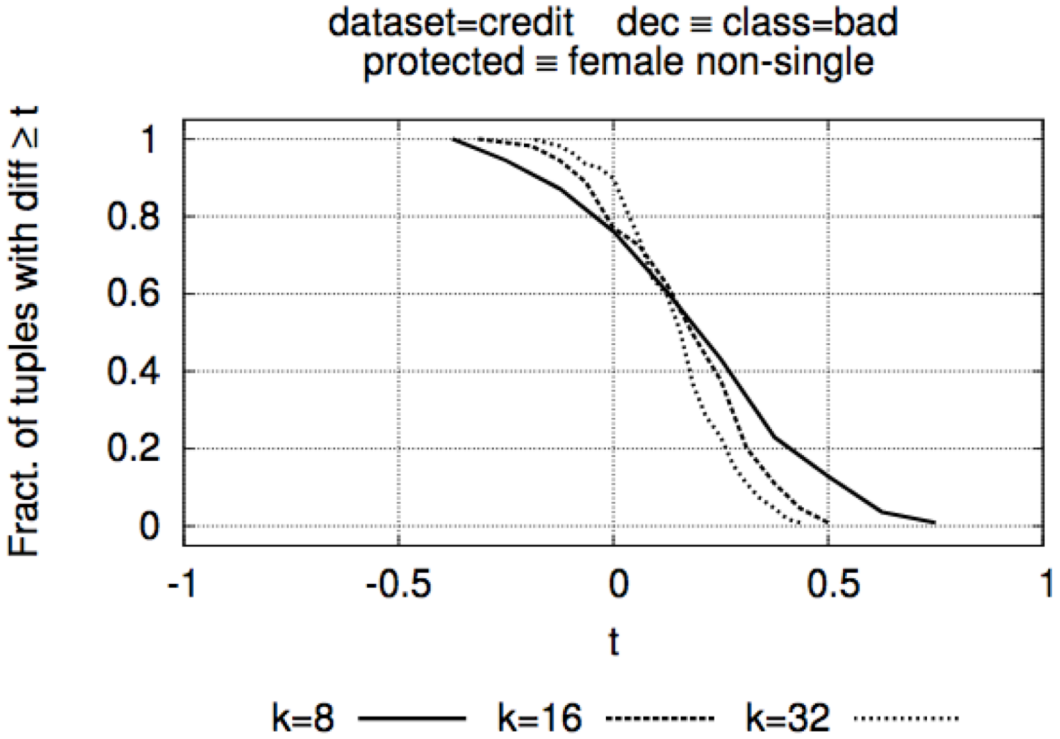


k-NN as situation testing (the algorithm)

- measure the degree of discrimination of the decision for r
  - define diff(r) = p_1 - p_2
- If decision=deny-benefit, and diff(r) ≥ t > 0, then we found discrimination around r



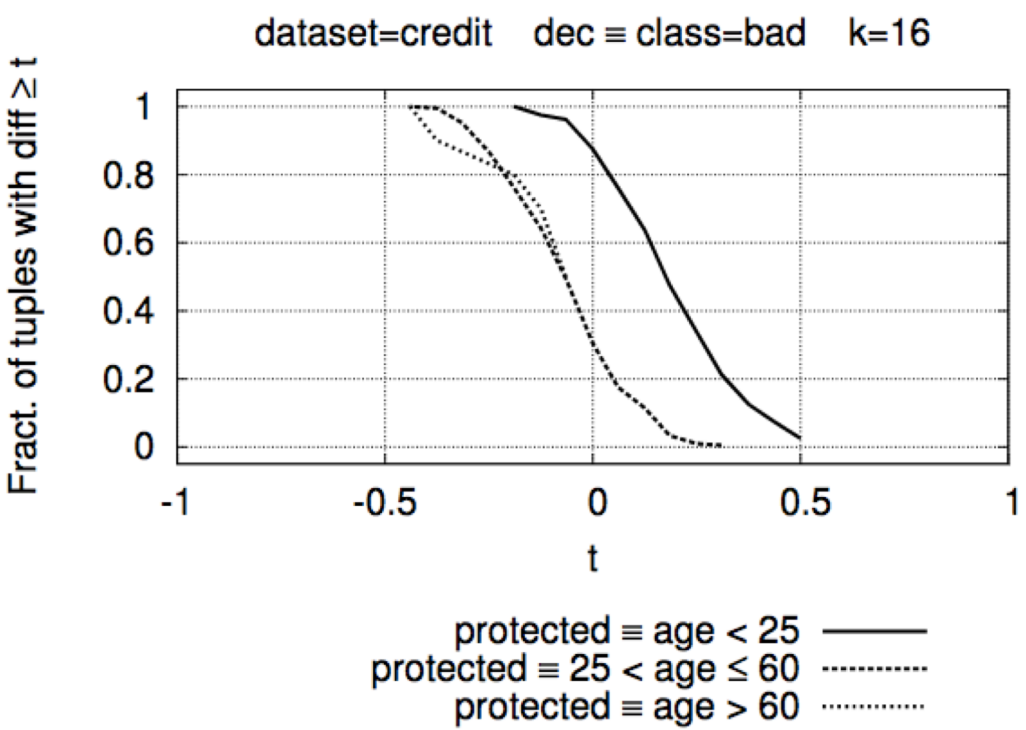
Results from German Credit Dataset



60% of non-single women have diff(r) ≥ 0.1

So, bad-debtors are at least 10% more frequent among k-most similar persons in protected group than among k-most similar persons in unprotected group.

Results from German Credit Dataset

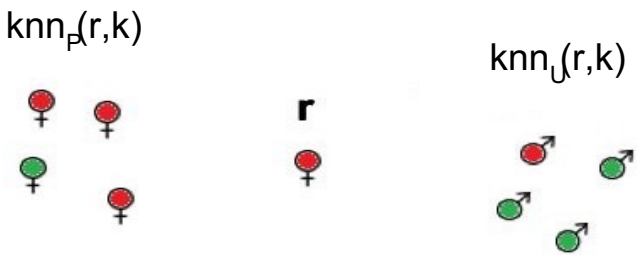


Cumulative distribution of diff() for protected groups defined on ranges of age.

The plot clearly shows that youngsters suffer from a higher bias towards the bad-debtor classification than middle-aged or older people.

Characterizing discrimination using k-NN

- For r ∈ P(R), set a new attribute: "t-discriminated"
  - If dec(r) = deny-benefit and diff(r) ≥ t, t-discriminated(r) := TRUE
    - Otherwise t-discriminated(r) := FALSE
- Example: for t=0.3 the sample r below is classified as t-discriminated



$p_1 = 0.75$

$p_2 = 0.25$

$diff(r) = p_1 - p_2 = 0.50$

How should t be chosen?

The answer depends on the law.

The U.K. legislation for sex discrimination (Sex Discrimination Act, 1975) sets t = 0.05, namely a 5% difference.

Discrimination discovery using k-NN

- To answer the question: under which conditions a protected group was t-discriminated?
- Create t-labeled version of a dataset P(R) by (i) including records of only protected people (ii) adding binary attribute disc (which is true if a person from a protected group is t-discriminated and false otherwise).
- Create a classifier (Decision Tree or Classification Rules) with training set P(R)
- Analyze the classifier to learn discrimination rules.

DiscoveryN$(\mathcal{R}, t)$ {
  $\mathcal{L} = \emptyset$
  for $\mathbf{r} \in P(\mathcal{R})$ {
    if($dec(\mathbf{r}) = \ominus$ and
      $diff(\mathbf{r}) \geq t$)
      $\mathbf{r}[\mathtt{disc}] = \mathtt{yes}$
    else
      $\mathbf{r}[\mathtt{disc}] = \mathtt{no}$
    $\mathcal{L} = \mathcal{L} \cup \{\mathbf{r}\}$
  }
  build a classifier on $\mathcal{L}$
}

Example discrimination rules found using DiscoveryN

- German credit dataset
  - protected = female non-single
  - 0.10-discriminated cases
- Decision tree model (C4.5)

```
num_dependents <= 1
|   credit_amount <= 2631: disc=yes (59.0/9.0)
|   credit_amount > 2631: disc=no (44.0/15.0)
num_dependents > 1: disc=no (6.0)

disc=yes:  Precision  0.847   Recall  0.769
```

Discriminated women had no dependents (children) and were asking for small amounts

- Classification rule model (RIPPER)

```
(credit_amount >= 3190) => disc=no (39.0/12.0)
(installment_commitment <= 2) and (residence_since >= 3)
                                        => disc=no (10.0/2.0)
 => disc=yes (60.0/9.0)

disc=yes:  Precision  0.85   Recall  0.785
```

Discriminated women were asking for small amounts and were either paying in many installments or had been resident for a short time

k-NN for discrimination prevention

- Goals of non-discriminating classifier:
  - Maximize classifier accuracy
    e.g., give credit to people who will pay
  - Minimize t-discriminated cases
    e.g., give credit to women if similar men would have been given credit
- Basic idea: t-correction of training set
  - **Flip the labels from negative to positive**
    for t-discriminated cases in the training set

$$
\begin{aligned}
&\textbf{PreventionN}(\mathcal{T},\ \mathcal{V},\ t)\ \{\\
&\quad \mathcal{T}' = \emptyset\\
&\quad \text{for } \mathbf{r} \in \mathcal{T}\ \{\\
&\qquad \mathbf{r}' = \mathbf{r}\\
&\qquad \text{if(}\ dec(\mathbf{r}) = \ominus \text{ and}\\
&\qquad\qquad protected(\mathbf{r}) \text{ and}\\
&\qquad\qquad\qquad diff(\mathbf{r}) \geq t\ )\\
&\qquad\quad \mathbf{r}'[\mathtt{dec}] = \oplus\\
&\qquad \mathcal{T}' = \mathcal{T}' \cup \{\mathbf{r}'\}\\
&\quad \}\\
&\quad \text{build classifiers on } \mathcal{T} \text{ and } \mathcal{T}'\\
&\quad \text{compare them on } \mathcal{V}\\
&\}
\end{aligned}
$$

k-NN for discrimination prevention (results)

Original Train Data

t-Corrected Data

	No pre-processing		0.10-correction	
classifier	accuracy	0.10-discr.	accuracy	0.10-discr.
C4.5	85.60%	4.24%	84.94%	1.07%
Naïve Bayes	82.46%	4.06%	82.33%	2.23%
Logistic	85.28%	6.61%	84.70%	0.61%
RIPPER	84.42%	5.24%	83.98%	3.94%
PART	85.20%	12.62%	84.00%	2.3%

Only the training set changes, the testing set is fixed

Accuracy shows a small decrease

0.10-discrimination reduces substantially

There may always be an accuracy-fairness tradeoff. But the approach balances them well.

Dataset: Income Predictions on Census Data (aka "Census Income" dataset) https://archive.ics.uci.edu/ml/datasets/Adult