Data Preprocessing Techniques for Classification with Discrimination

Faisal Kamiran - Toon Calders / 2011

- **Problem:** Training data exhibit unlawful discrimination
- **Task:**
  - → Optimize accuracy
  - → Classifier doesn't have discrimination on test data
- **Method:** Preprocess the data to remove discrimination
- **Application:** A binary sensitive attribute + 2-class classification problem

---

- **Discrimination:** Unfair or unequal treatment of individuals of a certain group based solely on their affiliation to that group
- **Discrimination-aware classification:**
  - → Input: A labeled dataset with one or more sensitive attributes
  - → Output: A classifier that doesn't correlate with the sensitive attribute
  - → Quality measurement: Accuracy and discrimination
- **Discrimination metric:**
  - → Sensitive attribute S / → b: Deprived community, w: Favored community
  - → Target attribute Class / → +: Desirable outcome, −: Non-desirable outcome
  - → Discrimination of a classifier (evaluation on the test set!)

$$disc_{S=b} = P(C(X) = + \mid X(S) = w) - P(C(X) = + \mid X(S) = b)$$

→ The deprived (b) and the favored (w) communities should have the same proportion of desirable (+) outcome for the classifier to be discrimination free ($disc_{S=b} = 0$)

⚠ Vocab: In the paper, a classifier is a classifier that only outputs the classes! → + / −
a Ranker is a classifier that outputs probabilities

- **Methods for preprocessing the data** (non-detailed version)
  - → **Suppression**: Remove S and the attributes correlated to it ⇒ Establish a baseline
  - → **Massaging the data**: Change the labels of some data points to remove discrimination
    A classifier/ranker is used to choose the data points whose labels we're going to change
  - → **Reweighing**: Weights are assigned to each data point ⇒ used for training a classifier
  - → **Sampling**: To be used when reweighing's not possible.
    Resample the combinations of $S = \{w, b\}$ and $Class = \{+, -\}$
    till making the dataset discrimination free

• b + : upsample	• w + : downsample
• b − : downsample	• w − : upsample

    2 proposed sampling methods → Uniform sampling (US)
    → Preferential sampling (PS)

- **Discrimination in a dataset**:
  → $disc_{S=b} = P(X(class)=+ \mid X(S)=w) - P(X(class)=+ \mid X(S)=b)$ } can be negative as well !!
  The prob of having a + outcome for someone in the favored group
  The prob of having a positive outcome for someone in the deprived group
  For discrimination of a classifier, the ≠ is:
  - → Instead of $X(class) \rightarrow C(X)$, C = the classifier
  - → The evaluation is done on the test set

- **Discrimination-aware classifier** → The accuracy of the classifier C is high
  → The discrimination of the classifier is low
  - 3 assumptions made in the article:
    - → A1: The goal is to learn a classifier → highest accuracy, 0 discrimination
    - → A2: Don't use the sensitive attribute at prediction time
    - → A3: The total ratio of + predictions of C should be similar to that of D

- **Some theory**
  - C dominates C' when { Accuracy(C) ≥ Accuracy(C') & Disc(C) ≤ Disc(C') }
  - C strictly dominates C' when one of these 2 inequalities is strict
  - DA-optimality (Accuracy / Discrimination): A classifier C in a set of classifiers $\mathcal{C}$ is DA-optimal if there is no other classifier in $\mathcal{C}$ that strictly dominates C

    In other words, there should be no classifier with both better accuracy and a lower discrimination than C (where one of them is strict)

- **Some notation** →
  - C_{all} = set of all classifiers
  - C^*_{all} = set of all classifiers with (same ratio of + cases) $P(C(X)=+ \mid X \in D) = P(X(Class)=+ \mid X \in D)$

- **Accuracy - Discrimination trade-off:**
  - A perfect classifier C^{perf} → $C^{perf}(X) = X(Class)$, $X \in D$

Theorem 1 *A classifier C is DA-optimal in C_{all} iff*

$$acc(C^{Perf}) - acc(C) = \frac{\min(d_b, d_w)}{d}\left(disc(C^{Perf}) - disc(C)\right)$$

A classifier C is DA-optimal in C^*_{all} iff

$$acc(C^{Perf}) - acc(C) = 2\frac{d_b}{d}\frac{d_w}{d}\left(disc(C^{Perf}) - disc(C)\right)$$

⇒ Discrimination-accuracy tradeoff for DA-optimal classifiers is linear

- **Imperfect classifiers** ⇒ The relationship remains linear
- Using a ranker allows for a relationship that is sub-linear

- **Solutions: Data preprocessing techniques**

1. **Massaging** → changing the labels to get a discrimination of 0.
   ↳ is a 2-step method

   - Step 1: Define M, the # of data points whose labels have to be changed for each group (b & w)

$$M = \frac{Disc(D) \times |D_b| \times |D_w|}{D}$$

   ⚠ M depends only on D

   ⚠ The distribution of labels doesn't change as the same # of points change labels + ↔ −

   - Step 2: Learn a ranker to choose which points to modify in each group (b & w)

   . We define two sets of points that are considered for label change
   ↳ promotion candidates (pr) = $\{x \in D \mid x(Class) = - \ \& \ x(S) = b\}$
   ↳ demotion candidates (dem) = $\{x \in D \mid x(Class) = + \ \& \ x(S) = w\}$

   . We build a ranker (R)
   ↳ We order pr in ascending order
   ↳ We order dem in descending order
   ↳ The points at the top of each class are close to the borderline, we take M points of each set and we change their labels

Using the def in the ranker, D is discrimination free ^_^

2. Reweighing → Weights are attached to the dataset to make it disc free

- **Idea**: In a discrimination-free dataset, the sensitive attribute and the label should be independent*

expected $\quad P_{exp}(S=b, X(Class)=+) = P(S=b) \cdot P(X(Class)=+)$

If the dataset is disc-free, the observed prob should = P_{exp}

We define weights for all cases: (+, b), (+, w), (−, b), (−, w)

$$W(X) = \frac{P_{exp}(S=X(S),\ Class=X(Class))}{P_{obs}(S=X(S),\ Class=X(Class))}$$

⚠ Reweighing only works with models that can use weights in the training

⚠ The authors define discrimination as the absence of correlation btwn the sensitive attribute and the label. Nonetheless, when talking about reweighing, they use independence !!!

3. Sampling → Instead of reweighing, over and under sample data points as not all models can use weights.

This method is a 2-steps method

Step 1: Calculate the weight for each case and multiply the # of occurences of each case by its corresponding weight to get the new # of samples

$$W(S, C) = \frac{|X \in D \mid X(S)=S| \times |X \in D \mid X(Class)=C|}{|D| \times |X(S)=S,\ X(Class)=C|}$$

- Step 2: Sample till getting the expected # of samples, using either uniforming sampling (US) or Preferential Sampling (PS)

Uniform Sampling (US): Sample data points with replacement from each case randomly until getting the expected # of samples

Preferential Sampling (PS):

- A ranker is used to rank data points among ≠ cases $(+,b), (+,w), (-,b), (-,w)$

PS starts from the original training dataset and iteratively duplicates (for the groups DP and FN) and removes objects (for the groups DN and FP) in the following way:

- Decreasing the size of a group is always done by removing the data objects closest to the boundary; i.e., the top elements in the ranked list.
- Increasing the sample size is done by duplication of the data object closest to the boundary. When an object has been duplicated, it is moved, together with its duplicate, to the bottom of the ranking. We repeat this procedure until the desired number of objects is obtained.

Conclusions:

- Removing the sensitive attribute from the data is not enough
- Massaging the data and Preferential Sampling usually work best
- The preprocessing works best when using this table classifiers down the line (decision trees, KNN with a small K)