Classification with Strategically Withheld Data[*][†]

Anilesh K. Krishnaswamy
Duke University
anilesh@cs.duke.edu

Haoming Li
University of Southern California
haoming.li@usc.edu

David Rein
Duke University
irving.rein@duke.edu

Hanrui Zhang
Duke University
hrzhang@cs.duke.edu

Vincent Conitzer
Duke University
conitzer@cs.duke.edu

ABSTRACT

Machine learning techniques can be useful in applications such as credit approval and college admission. However, to be classified more favorably in such contexts, an agent may decide to strategically withhold some of her features, such as bad test scores. This is a missing data problem with a twist: which data is missing *depends on the chosen classifier*, because the specific classifier is what may create the incentive to withhold certain feature values. We address the problem of training classifiers that are robust to this behavior.

We design three classification methods: MINCUT, HILL-CLIMBING (HC) and Incentive-Compatible Logistic Regression (IC-LR). We show that MINCUT is optimal when the true distribution of data is fully known. However, it can produce complex decision boundaries, and hence be prone to overfitting in some cases. Based on a characterization of truthful classifiers (i.e., those that give no incentive to strategically hide features), we devise a simpler alternative called HC which consists of a hierarchical ensemble of out-of-the-box classifiers, trained using a specialized hill-climbing procedure which we show to be convergent. For several reasons, MINCUT and HC are not effective in utilizing a large number of complementarily informative features. To this end, we present IC-LR, a modification of Logistic Regression that removes the incentive to strategically drop features. We also show that our algorithms perform well in experiments on real-world data sets, and present insights into their relative performance in different settings.

KEYWORDS

Classification, Game Theory, Missing Data, Mechanism Design

[*]Previously accepted at AAAI 2021.

[†]Supplement provided at the end of the paper. Full paper also available at https://arxiv.org/abs/2012.10203

1 INTRODUCTION

Applicants to most colleges in the US are required to submit their scores for at least one of the SAT and the ACT. Both tests are more or less equally popular, with close to two million taking each in 2016 [1]. Applicants usually take one of these two tests – whichever works to their advantage.[1] However, given the growing competitiveness of college admissions, many applicants now take both tests and then strategically decide whether to drop one of the scores (if they think it will hurt their application) or report both.[2] The key issue here is that it is impossible to distinguish between an applicant who takes both tests but reports only one, and an applicant that takes only one test—for example because the applicant simply took the one required by her school, the dates for the other test did not work with her schedule, or for other reasons that are not strategic in nature.[3]

Say a college wants to take a principled machine learning approach to making admission decisions based on the scores from these two tests. For simplicity, assume no other information is available. Assume that the college has enough historical examples that contain the scores of individuals (on whichever tests are taken, truthfully reported) along with the corresponding ideal (binary) admission decisions.[4] Based on this data, the college has to choose a decision function that determines which future applicants are accepted. If this function is known to the applicants, they are bound to strategize and use their knowledge of the decision function to decide the scores they report.[4] How can the classifier be trained to handle strategic reporting of scores at prediction time?

To see the intricacies of this problem, let us consider a simple example.

Example 1. *Say the scores for each of the two tests (SAT and ACT) take one of two values: h (for high) or l (for low). Let $*$ denote a missing value. Then there are eight possible inputs (excluding $(*, *)$ since at least one score is required): (h, h), (h, l), (l, h), (l, l), $(h, *)$, $(*, h)$, $(l, *)$ and $(*, l)$. Assume the natural distribution (without any withholding) over these inputs is known, and so are the conditional probabilities of the label $Y \in \{0, 1\}$, as shown below:*

Assume $Y = 1$ is the more desirable "accept" decision. Then, ideally, we would like to predict $\widehat{Y} = 1$ whenever $X \in \{(h, h), (h, l), (h, *), (*, h)\}$. However, the strategic reporting of scores at prediction time effectively means, for example, that an input $(*, h)$ cannot be assigned the accept decision of $\widehat{Y} = 1$ unless the same is done for (l, h) as well;

---

[1]https://www.princetonreview.com/college/sat-act

[2]https://blog.collegevine.com/should-you-submit-your-sat-act-scores/

[3]https://blog.prepscholar.com/do-you-need-to-take-both-the-act-and-sat

[4]We make these assumptions more generally throughout the paper.

Table 1: True distribution of inputs and targets:

X	(h,h)	(h,l)	(l,h)	(l,l)	$(h,*)$	$(*,h)$	$(l,*)$	$(*,l)$
$Pr(X)$	1/8	1/8	1/8	1/8	1/8	1/8	1/8	1/8
$Pr(Y=1 \mid X)$	0.9	0.7	0.3	0.1	0.6	0.6	0.2	0.2
$Pr(Y=0 \mid X)$	0.1	0.3	0.7	0.9	0.4	0.4	0.8	0.8

otherwise, someone with (l,h) would simply not report the first test, thereby misreporting $(*,h)$ and being accepted. Taking this into account, the classifier with minimum error is given by $\widehat{Y} = 1$ whenever $X \in \{(h,h),(h,l),(h,*)\}$.

There are many other settings where a similar problem arises. Many law schools now allow applicants to choose between the GRE and the traditional LSAT.[5] Recently, as a result of the COVID-19 pandemic, universities have implemented optional pass/fail policies, where students can choose to take some or all of their courses for pass/fail credit, as opposed to a standard letter grade that influences their GPA. They are often able to decide the status after already knowing their performance in the course. For credit scoring, some individuals might not report some of their information, especially if it is not mandatory by law [15].

The ability of strategic agents to withhold some of their features at prediction time poses a challenge only when the data used to train the classifier has some naturally missing components to begin with. For if not, the *principal* – e.g., the entity deciding on admissions – can reject all agents that withhold any of their features, thereby forcing them to reveal all features. We focus on how a principal can best train classifiers that are robust even when there is strategic withholding of data by agents. Our methods produce classifiers that eliminate the incentive for agents to withhold data.

Our contributions. We now describe the key questions we are facing, and how we answer them. Our model is described formally in *Section 2*. All proofs are in the Supplement.

If the true input distribution is known, can we compute the optimal classifier? (Section 3) We answer this question in the affirmative by showing that the problem of computing the optimal classifier (Theorem 1) in this setting reduces to the classical Min-cut problem [6]. This analysis gives us the MINCUT classifier, which can be computed on the empirical distribution, estimated using whatever data is available. However, since it can potentially give complex decision boundaries, it might not generalize well.

Are there simpler classifiers that are robust to strategic withholding of features? (Section 4) We first characterize the structure of classifiers that are "truthful", i.e., give no incentive to strategically hide features at prediction time (Theorem 2). Using this characterization, we devise a hill-climbing procedure (HC) to train a hierarchical ensemble of out-of-the-box classifiers and show that the procedure converges (Theorem 4) as long as we have black-box access to an agnostic learning oracle. We also analytically bound the generalization error of HC (Theorem 3). The ensemble of HC can be populated with any of the commonly used classifiers such as logistic regression, ANNs, etc.

Another truthful classifier we present is a modification of Logistic Regression. This method, called IC-LR (Incentive Compatible

Logistic Regression), works by encoding all features with positive values, and using positive regression coefficients – whereby it is in every agent's best interest to report all features truthfully. IC-LR uses Projected Gradient Descent for its training. The advantage of this method is that it can be directly to a large number of features.

How do our methods perform on real data sets? (Section 6) We conduct experiments on several real-world data sets to test the performance of our methods, comparing them to each other, as well as to other methods that handle missing data but ignore the strategic aspect of the problem. We see that our methods perform well overall, and uncover some interesting insights on their relative performance:

(1) When the number of features is small, HC is the most reliable across the board.
(2) When the number of features is small, and many of them are discrete/categorical (or suitably discretized), MINCUT and IC-LR perform better.
(3) If a large number of features must be used, IC-LR gives the best performance, although HC performs reasonably well with some simple feature selection techniques.

Related work. Our work falls broadly in the area of *strategic machine learning*, wherein a common assumption is that strategic agents can modify their features (i.e., misreport) in certain ways (normally at some cost), either to improve outcomes based on the classifier chosen by the principal [20] or to influence which classifier is chosen in the first place [9]. The main challenge in strategic machine learning, as in this paper, is the potential misalignment between the interests of the agents and the principal. Existing results in this line of work [4, 18, 25], often mainly theoretical, consider classifiers of a specific form, say linear, and ways of misreporting or modifying features in that context. Our results are different in that we focus on a specific type of strategic misreporting, i.e., withholding parts of the data, and devise general methods that are robust to this behavior that, in addition to having theoretical guarantees, can be tested practically. Some experimental results [20] do exist – but our work is quite different; for instance, we do not need to invent a cost function (as in Hardt et al. [20]). Another major difference is that we consider generalization in the presence of strategic behavior, while most previous work does not (except for a concurrent paper [39]), which studies the sample complexity of PAC learning in the presence of strategic behavior).

Our problem can also be viewed as an instance of *automated mechanism design with partial verification* [17, 23, 24, 34] where it is typically assumed that the feature space (usually called type space in mechanism design) is discrete and has reasonably small cardinality, and a prior distribution is known over the feature space. In contrast, the feature spaces considered in this paper consist of all possible combinations of potentially continuous feature values. Moreover, the population distribution can only be accessed by observing examples. Thus, common methodologies in automated mechanism design do not suffice for our setting.

A set of closely related (in particular, to Theorem 1) theoretical results are those of Zhang et al. [35, 36, 38] on the problem of distinguishing "good" agents from "bad" (where each produces a different distribution over a sample space, and the agent can misreport the set of n samples that she has drawn). However, our

work is different in that we consider the standard classification problem, we focus more on practical aspects, and we do not rely on the full knowledge of the input distribution.

Our work also finds a happy intersection between strategic machine learning and the literature on classification with missing data [27]. The problem we study is also connected to *adversarial classification* [7, 10]. We discuss these connections in more detail in the Supplement.

2 PRELIMINARIES

We now describe our model and the requisite notation.

Model with strategically withheld features: We have an input space $\mathcal{X}$, a label space $\mathcal{Y} = \{0, 1\}$, and a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ which models the population. A classifier $f : \mathcal{X} \rightarrow \mathcal{Y}$ maps a combination of features to a label. Let $F = [k] = \{1, \ldots, k\}$ be the set of features, each of which a data point may or may not have. For $x \in \mathcal{X}$, let x_i denote the value of its i-th feature ($x_i = *$ if x does not have feature $i \in [k]$). For any $S \subseteq [k]$, define $x|_S$ to be the projection of x onto S (i.e., retain features in S and drop those not in S):

$$(x|_S)_i = \begin{cases} x_i, & \text{if } i \in S \\ *, & \text{otherwise.} \end{cases}$$

We assume that data can be strategically manipulated at prediction (test) time in the following way: an agent whose true data point is x can report any other data point x' such that $x|_S = x'$ for some $S \subseteq [k]$. We use $\rightarrow$ to denote the relation between any such pair x, x' ($x \rightarrow x' \iff \exists S \subseteq [k] : x|_S = x'$). Note that $\rightarrow$ is transitive, i.e., for any $x_1, x_2, x_3 \in \mathcal{X}$, $x_1 \rightarrow x_2$ and $x_2 \rightarrow x_3 \implies x_1 \rightarrow x_3$.

We assume agents prefer label 1 to 0: in response to a classifier f, an agent with data point x will always withhold [6] features to receive label 1 if possible, i.e., the agent will report $x' \in \text{argmax}_{x'':x \rightarrow x''} f(x'')$. Incorporating such strategic behavior into the loss of a classifier f, we get

$$\ell_{\mathcal{D}}(f) = \Pr_{(x,y)\sim\mathcal{D}} \left[y \neq \max_{x':x \rightarrow x'} f(x') \right].$$

Truthful classifiers. We will also be interested in *truthful* classifiers, which provably eliminate incentives for such strategic manipulation. A classifier f is *truthful* if for any $x, x' \in \mathcal{X}$ where $x \rightarrow x'$, $f(x) \geq f(x')$. In other words, not withholding any features is always an optimal way to respond to a truthful classifier. As a result, the loss of any truthful classifier f in the presence of strategically withheld features has the standard form: $\ell_{\mathcal{D}}(f) = \Pr_{(x,y)\sim\mathcal{D}}[f(x) \neq y]$.

Note that the so-called Revelation Principle – which states that in the presence of strategic behavior, any classifier f is equivalent to a truthful classifier f' – holds in this case because the reporting structure is transitive.[7] In other words, we are guaranteed that, for any classifier f, there exists a truthful classifier f', such that for any $x \in \mathcal{X}$, $\max_{x':x \rightarrow x'} f(x') = f'(x)$. Therefore, we focus on truthful classifiers in our model, without loss of generality.

3 THE MINCUT CLASSIFIER

We first present a method for computing an optimal classifier *when the input distribution is fully known.*[8] Assuming $\mathcal{X}$ is finite, our goal is to characterize a classifier f^* which minimizes the loss $\ell_{\mathcal{D}}(.)$, for a known input distribution $\mathcal{D}$. As shorthand, define, for all $x \in \mathcal{X}$,

Definition 1. $\mathcal{D}^+(x) \coloneqq \Pr_{(x',y')\sim\mathcal{D}}[x' = x \wedge y' = 1]$, $\mathcal{D}^-(x) \coloneqq \Pr_{(x',y')\sim\mathcal{D}}[x' = x \wedge y' = 0]$.

The basic idea here is simple: to partition $\mathcal{X}$ into two sides, one labeled 1 and the other 0, where the error accrued for each $x \in \mathcal{X}$ is given by $\mathcal{D}^-(x)$ or $\mathcal{D}^+(x)$, according as x is labeled 1 or 0. Such a partition should crucially respect the constraints imposed by the strategic behavior of agents : if $x \rightarrow x'$, then either x is labeled 1 or x' is labeled 0.

Definition 2. Given $\mathcal{X}$ and $\mathcal{D}$, let $G(\mathcal{D}, \mathcal{X})$ be a directed capacitated graph with vertices $V = \mathcal{X} \cup \{s, t\}$, where the edges E and edge capacities u are defined as follows:

- For each $x \in \mathcal{X}$, there are edges (s, x) and (x, t) in E, with capacities $u(s, x) = \mathcal{D}^-(x)$ and $u(x, t) = \mathcal{D}^+(x)$.
- For all pairs $x, x' \in \mathcal{X}$ such that $x \rightarrow x'$, there is an edge $(x, x') \in E$ with capacity $u(x, x') = \infty$.

In terms of the graph defined above, computing the optimal classifier f^* we seek is equivalent to finding a minimum s-t cut on $G(\mathcal{D}, \mathcal{X})$. The intuition is that the edges from s and to t reflect the value gained from labeling an example 0 or 1, respectively; one of the edges must be cut, reflecting the loss of not assigning it to the corresponding side. Moreover, if $x \rightarrow x'$, then the corresponding edge with infinite capacity prevents the assigning of 0 to x and 1 to x'.

Theorem 1. *If $(S, \bar{S})$ is a minimum s-t cut of $G(\mathcal{D}, \mathcal{X})$ (where S is on the same side as s), then for the classifier $f^*(x) \coloneqq \mathbb{1}(x \in \bar{S})$, we have $\ell_{\mathcal{D}}(f^*) = \min_f \ell_{\mathcal{D}}(f)$.*

We note that, consequently, the optimal classifier can be computed in poly($|\mathcal{X}|$) time. In practice, it is natural to expect that we do not know $\mathcal{D}$ exactly, but have a finite number of samples from it. A more practical option is to apply Theorem 1 to the empirical distribution induced by the samples observed, and hope for the classifier computed from that to generalize to the true population distribution $\mathcal{D}$.

Implementing MINCUT. Given a set $\widehat{\mathcal{X}}$ of m i.i.d. samples from $\mathcal{D}$, let $\widehat{\mathcal{D}}$ be the corresponding empirical distribution over $\widehat{\mathcal{X}}$, and $\bar{\mathcal{X}} \coloneqq \widehat{\mathcal{X}} \cup \{x' : x' \rightarrow x, \exists x \in \widehat{\mathcal{X}}\}$. The MINCUT classifier is then obtained by applying Theorem 1 to $G(\widehat{\mathcal{D}}, \widehat{\mathcal{X}})$, and extending it to $\bar{\mathcal{X}}$ as and when required. Here, note that MINCUT runs in time poly(m) (and not poly($|\mathcal{X}|$)), since $G(\widehat{\mathcal{D}}, \widehat{\mathcal{X}})$ has m nodes, and checking if a test point is in $\bar{\mathcal{X}}$ takes poly(m) time.

In light of traditional wisdom, the smaller m is relative to $\mathcal{X}$, the larger the generalization error of MINCUT will be. We do not attempt a theoretical analysis in this regard, but note that when $\mathcal{X}$ is

---

[6]In practice, f might not be perfectly known, and agents might not be able to best respond. This problem does not arise for our methods, since they are truthul. For other classifiers, their accuracy may go up or down if agents fail to best-respond; but the assumption that agents best-respond is common in many such contexts.

[7]More details, including a formal proof, are in the Supplement.

---

[8]A theoretical companion paper [37] contains a more general version of the mincut-based algorithm. There, the goal is to compute an optimal classifier with possibly more than 2 outcomes given perfect knowledge of the entire population distribution. In this paper, we investigate the special case with only 2 outcomes (i.e., accept and reject), but do not assume prior knowledge about the population distribution.

large, the generalization error can be extremely large (see Example 2 in the Supplement). The reason for this is two-fold:

(1) Mincut can give complicated decision boundaries.
(2) Mincut is indecisive on samples not in $\bar{X}$.[9]

Therefore, a suitable discretization of features is sometimes useful (see Section 6). Note that Mincut is truthful, by virtue of the infinite capacity edges in Definition 2.

4 TRUTHFUL CLASSIFIERS AND HILL-CLIMBING

The other drawback of Mincut, related to the issue of generalization just discussed, is that it can be hard to interpret meaningfully in a practical setting. In this section, we devise a simpler alternative called Hill-Climbing. To help introduce this algorithm, we first present a characterization of truthful classifiers in our setting, since we can limit our focus to them without loss of generality (as discussed in Section 2). For shorthand, we use the following definition:

Definition 3 (F'-classifier). For a subset of features $F' \subseteq F$, a classifier f is said to be an F'-classifier if for all $x \in X$, we have $f(x) = f(x|_{F'})$, and if there exists $i \in F'$ such that $x_i = *$, then $f(x) = 0$.

In other words, an F'-classifier depends only on the values of the features in F', rejecting any x where any of these is empty. We can collect many such classifiers into an ensemble as follows:

Definition 4 (Max Ensemble). For a collection of classifiers $C = \{f_j\}$, its Max Ensemble classifier is given by $\text{Max}_C(.) := \max_j f_j(.)$.

This is equivalent to getting each agent to pick the most favorable classifier from among those in $\{f_j\}$. Now using the above definitions we have the following characterization of truthful classifiers:

Theorem 2. A classifier f is truthful iff $f(.) = \text{Max}_C(.)$ for a collection of classifiers $C = \{f_j\}$ such that, for some $\{F_j\} \subseteq 2^F$, each f_j is an F_j-classifier.

Now, for any truthful classifier f, we can bound the gap between its population loss $\ell_{\mathcal{D}}(f)$ and its empirical loss on a set of samples $\widehat{X}$ denoted by $\ell_{\widehat{X}}(f) := \frac{1}{m} \sum_{i \in [m]} |f(x_i) - y_i|$. Before stating a theorem to this end, we define the following entities: Let $\mathcal{H}$ be a base hypothesis space over X, and $n \in \{1, \ldots, 2^k\}$ be a parameter. Define $d := d_{\text{VC}}(\mathcal{H})$ to be the VC dimension of $\mathcal{H}$. Define $\bar{\mathcal{H}}$ as the set of all classifiers that can be written as the Max Ensemble of n classifiers in $\mathcal{H}$.

Theorem 3. Let $\widehat{X} = \{(x_i, y_i)\}_{i \in [m]}$ be m i.i.d. samples from $\mathcal{D}$. For any $f \in \bar{\mathcal{H}}$, for any $\delta > 0$, with probability at least $1 - \delta$, we have
$$\ell_{\mathcal{D}}(f) \le \ell_{\widehat{X}}(f) + O\left(\sqrt{\frac{dn \cdot \log dn \cdot \log m + \log(1/\delta)}{m}}\right).$$

It is easy to see that for any of the commonly used hypothesis spaces – say $\mathcal{H}$ consists of linear hypotheses – if a truthful classifier f is in $\mathcal{H}$, then so are the components of the Max Ensemble version of f as in Theorem 2. We have, however, stated Theorem 3 in slightly more general terms.

The Hill-Climbing *classifier.* We now present a hill-climbing approach with provable convergence and generalization guarantees. The Hill-Climbing classifier (henceforth HC) is of the same form as given by the characterization of truthful classifiers in Theorem 2.[10] Intuitively, the approach works by considering a hierarchy of classifiers, organized by the features involved. For example, consider a setting with $k = 3$ features. We make a choice as to what classifiers we use — say f_1 for input of the form $(x_1, *, *)$, f_2 for input of the form $(x_1, x_2, *)$, and f_3 for input of the form (x_1, x_2, x_3). Any agent with features 1 and 2 (but not 3), for example, should be able to report both features so as to be classified by f_2, or feature 2 to be classified by f_1 instead. So in effect, assuming full knowledge of the classifiers, each agent can check all of the classifiers and choose the most favorable one. Without loss of generality, we assume that when a data point does not have all the features required by a classifier, it is automatically rejected.

Algorithm 1 Hill-Climbing (HC) Classifier

Input: data set $\widehat{X} = \{(x_i, y_i)\}_{i \in [m]}$, n subsets $F_1, F_2, \ldots, F_n$ of F.

Initialize: $t \leftarrow 0$, $\{f_1^0, \ldots, f_n^0\}$.
while $\Delta > 0$ **do**
  for $i = 1, 2, \ldots, n$ **do**
    $S_i \leftarrow \{(x, y) \in \widehat{X} : f_j^t(x|_{F_j}) = 0, \forall j \ne i\}$.
    $f_i^{t+1} = \text{argmin}_{f \in \mathcal{H}} \sum_{(x,y) \in S_i} |f(x|_{F_i}) - y|$.
  end for
  $f^* \leftarrow \text{Max}_{\{f_1^{t+1}, \ldots, f_n^{t+1}\}}$; $\ell_t = \ell_{\widehat{X}}(f^*)$
  $\Delta \leftarrow \ell_t - \ell_{t-1}$; $t \leftarrow t + 1$
end while
Return: f^*.

In short, HC (defined formally in Algorithm 1) works as follows: first choose a hypothesis space $\mathcal{H}$, in order for Theorem 3 to apply. Then select n subsets of F (where n is a parameter), say $F_1, F_2, \ldots, F_n$. For each F_j, we learn a F_j-classifier, say f_j, from among those in H. Start by initializing these classifiers to any suitable $\{f_1^0, \ldots, f_n^0\}$. In each iterative step, each of the subclassifiers is updated to minimize the empirical loss on the samples that are rejected by all other classifiers. We next show that such an update procedure always converges. To do so, as far as our theoretical analysis goes, we assume we have black-box access to an agnostic learning oracle (Line 6 in Algorithm 1). After convergence, the HC classifier is obtained as the Max Ensemble of these classifiers. The generalization guarantee of Theorem 3 applies directly to the HC classifier.

Theorem 4. Algorithm 1 converges.

Connection with Mincut*:* The HC formulation given above can be thought of as a less complicated version of Mincut: some of the edge constraints are ignored with respect to learning the individual classifiers, and are instead factored in via the Max function. Say $F_1 \subset F_2$. For some x, it is possible that $f_1(x|_{F_1}) = 1$ and $f_2(x|_{F_1}) = 0$. In other words, the individual classifiers could violate the Mincut constraints, in order to learn classification functions that generalize well individually, and also collectively thanks to the combined HC training procedure.

---

[9]This is more likely to happen when using a large number of features.

[10]And, therefore, is truthful, and inherits Theorem 3.

Implementing HC:. In practice, the classifiers $\{f_1, f_2, \ldots, f_n\}$ in HC can be populated with any standard out-of-the-box methods such as logistic regression or neural networks, the choice of which can influence the performance of f. In Section 6, we test HC with a few such options. The assumption of having access to an agnostic learning oracle does not play a crucial role in practice, with standard training methods performing well enough to ensure convergence. Also, HC will converge in at most m (number of training examples) iterations, because in each iteration the number of correctly classified examples increases by at least one. (An iteration may need to train n individual classifiers.) This also means there is no difference between checking whether $\Delta > 0$ or $\Delta \geq 1/m$. In our experiments, we run HC using $\Delta \geq 10^{-4}$, and convergence is achieved pretty quickly (see the Supplement for exact details).

Choosing subsets: Note that we are free to choose any $F_1, F_2, \ldots, F_n$ to define HC. Its generalization (via Theorem 3), will depend on the choice of n. As more and more subsets of features are included (and further binning them based on their values), HC starts behaving more and more like MINCUT. In addition, using a large number of subsets increases the computational complexity of HC. In practice, therefore, the number of subsets must be limited somehow – we find that some simple strategies like the following work reasonably well: (a) selecting a few valuable features and taking all subsets of those features, (b) taking all subsets of size smaller than a fixed number k, say $k = 2$. In many practical situations, a few features (possibly putting their values in just a few bins) are often enough to get close to optimal accuracy, also improving interpretability (e.g., see Wang and Rudin [33] or Jung et al. [22]) The question of devising a more nuanced algorithm for selecting these subsets merits a separate investigation, and we leave this to future work.

5 INCENTIVE-COMPATIBLE LOGISTIC REGRESSION

As we just mentioned, it is challenging to directly apply HC and MINCUT to a large number of features. As we will see, we can address this challenge in various ways to still get very strong performance with HC. Moreover, HC enjoys remarkable generality, generalization and convergence guarantees. Nevertheless, we would like to have an algorithm that tries to make use of all the available features, while still being truthful. In this section, we present such an approach, which, as we show later in Section 6, indeed performs comparably to – and in some cases better than – MINCUT and HC.

Below we present a simple and truthful learning algorithm, Incentive-Compatible Logistic Regression (IC-LR), which is a truthful variant of classical gradient-based algorithms for logistic regression. Recall that in logistic regression, the goal is to learn a set of coefficients $\{\beta_i\}$, one for each feature $i \in F$, as well as an intercept β_0, such that for each data point (x, y), the predicted label $\hat{y}$ given by

$$\hat{y} = \mathbb{1}\left[\sigma(\beta_0 + \sum_{i \in F} x_i \cdot \beta_i) \geq 0.5\right]$$

fits y as well as possible, where $\sigma(t) = 1/(1+e^{-t})$ is the logistic function. Roughly speaking, IC-LR. (formally defined in Algorithm 2) works by restricting the coefficients $\{\beta_i\}$ in such a way that dropping a feature (i.e., setting x_i to 0) can never make the predicted

label larger. If, without loss of generality, all feature values x_i are nonnegative[11], then this is equivalent to: for each feature $i \in F$, the coefficient $\beta_i \geq 0$. IC-LR. enforces this nonnegativity constraint throughout the training procedure, by requiring a projection step after each gradient step, which projects the coefficients to the feasible nonnegative region by setting any negative coefficient to 0 (equivalently, an ℓ_1 projection).

Algorithm 2 Incentive-Compatible Logistic Regression

Input: data set $\widehat{X} = \{(x, y)\}$, learning rate $\{\eta_t\}$, $\delta \geq 0$.
Initialize: $t \leftarrow 0$, $\{\beta_0, \beta_1, \ldots, \beta_k\}$.
while $\Delta > \delta$ **do**
    $g_i \leftarrow 0$ for all $i \in \{0, 1, \ldots, k\}$
    for $(x, y) \in \widehat{X}$ **do**
        $g_0 \leftarrow g_0 + \sigma(\beta_0 + \sum_{i \in F} x_i \cdot \beta_i) - y$
        for $i \in F$ **do**
            $g_i \leftarrow g_i + (\sigma(\beta_0 + \sum_{i \in F} x_i \cdot \beta_i) - y) \cdot x_i$
        end for
    end for
    $\forall i \in \{0, 1, \ldots, k\}$, $\beta_i \leftarrow \max\{\beta_i - \eta_t \cdot g_i, 0\}$
    $f^*(x) := \mathbb{1}(\sigma(\beta_0 + \sum_{i \in F} \beta_i \cdot x_i) \geq 0.5)$
    $\ell_t = \ell_{\widehat{X}}(f^*)$; $\Delta \leftarrow \ell_t - \ell_{t-1}$; $t \leftarrow t + 1$
end while
Return: f^*.

One potential issue with IC-LR. is the following: if a certain feature $x_i \geq 0$ is negatively correlated with the positive classification label, then IC-LR is forced to ignore it (since it is constrained to use positive coefficients). To make good use of this feature, we can include an inverted copy $x_i' = \lambda - x_i$ (where λ is chosen such that $x_i' \geq 0$). We could also choose an apt discretization of such features (using cross-validation) and translate the discretized bins into separate binary variables. Such a discretization can account for more complex forms of correlation, e.g., a certain feature's being too high or too low me makes the positive label likelier. In practice, we find that the latter method does better. If such transformations are undesirable, perhaps for reasons of complexity or interpretability, HC methods are a safer bet.

6 EVALUATION

In this section, we show that, when strategic withholding is at play, MINCUT, HC and IC-LR perform well and provide a significant advantage over several out-of-the-box counterparts (that do not account for strategic behavior).

Datasets. Four credit approval datasets are obtained from the UCI repository [12], one each from Australia, Germany, Poland and Taiwan. As is common for credit approval datasets, they are imbalanced to various degrees. In order to demonstrate the performance of classifiers in a standard, controlled setting, we balance them by random undersampling. There is a dedicated community [3] that looks at the issue of imbalanced learning. We do not delve into these issues in our paper, and evaluate our methods on both balanced and imbalanced datasets (see the Supplement for the latter).

---

[11]If not, they can be suitably translated.

Table 2: Data set summary statistics (num. = numerical, cat. = categorical)

Data set	Size	Total # of features	Size after balancing	Features after restriction
Australia	690	15	614	2 num., 2 cat.
Germany	1000	20	600	1 num., 3 cat.
Poland	5910	64	820	4 num.
Taiwan	30,000	23	13,272	4 ordinal

In addition, to demonstrate the challenge of high-dimensional data imposed on some of the classification methods, the experiments are run on the datasets (a) restricted to 4 features,[12] and (b) with all available features. The basic characteristics of the datasets are summarized in Table 2 – note that there is enough variation in terms of the types of features present. We then randomly remove a fraction $\epsilon = 0, 0.1, \ldots, 0.5$ of all feature values in each dataset to simulate data that is missing "naturally" – i.e., not due to strategic withholding.

Testing. We test all methods under two ways of reporting: "truthful", i.e., all features are reported as is, and "strategic", i.e., some features might be withheld if it leads to a better outcome. We measure the test accuracy of each classifier, averaged over N=100 runs, with randomness over the undersampling and the data that is randomly chosen to be missing, to simulate data missing for non-strategic reasons. Other metrics, and details about implementing and training the classifiers, are discussed in the Supplement. It is important to note that for testing any method, we have to, in effect, compute the best response of each data point toward the classifier. Since the methods we propose are truthful, this is a trivial task. But for other methods, this might not be easy, thereby limiting what baselines can be used.

Classifiers. We evaluate our proposed methods, Mincut, HC with logistic regression (HC (LR)) and neural networks (HC (ANN)) as subclassifiers, and incentive-compatible logistic regression (IC-LR), against several baseline methods.

First, they will be compared against three out-of-the-box baseline classifiers: logistic regression (LR), neural networks (ANN) and random forest (RF). We select LR for its popularity in credit approval applications; we select ANN for it being the best-performing individual classifier on some credit approval datasets [26]; we select RF for it being the best-performing homogeneous ensemble on some credit approval datasets [26], as HC can be viewed as a homogeneous ensemble method. For the sake of exposition, we present numbers just for baselines based on LR, as they perform relatively better.

Second, for the purposes of comparison, we include Maj – predict the *majority* label if examples with the exact same feature values appeared in the training set, and reject if not – which can be thought of as a non-strategic counterpart of Mincut. We also include k-nearest neighbors (kNN) as a baseline, since it is closely related to Maj.

Table 3: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, 4 features

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.792	**.792**	.639	.639	.659	.659	.648	.648
Mincut	.770	.770	.580	.580	.501	.501	.652	**.652**
IC-LR	.788	.788	.654	**.654**	.639	.639	.499	.499
Imp(LR)	.796	.791	**.663**	.580	**.714**	**.660**	**.670**	.618
R-F(LR)	**.808**	.545	.631	.508	.670	.511	.665	.590

These out-of-the-box classifiers need help dealing with missing data, whether they are missing naturally at training and test time or strategically at test time, and to this end, we employ (a) Imp: mean/mode imputation [26], and (b) R-F: reduced-feature modeling [29], for each of them.

When the dataset has a large number of features, Mincut and IC-LR can be directly applied. For HC, we assist it in two ways: (a) by selecting 4 features based on the training data, denoted by fs (feature selection),[13] and (b) by choosing a limited number of small subsets (30 with 1 feature and 30 with 2 features), denoted by app (approximation). Note that since our proposed methods are truthful, we can assume that features are reported as is. However, for all out-of-the-box classifiers, except Imp(LR), it is infeasible to simulate strategic withholding of feature values, due to the enormous number of combinations of features.

Last but not least, we test all methods with the discretization of continuous features (into categorical ones) [14], for reasons given in earlier sections.

6.1 Results

For want of space, we report results only for $\epsilon = 0.2$. We also limit our exposition of HC, Imp and R-F methods to those based on logistic regression, as these perform better than their ANN/RF/kNN counterparts. For a comprehensive compilation of all results, along with standard deviation numbers, please refer to the Supplement.

With a small number of features (Table 3): As expected, the out-of-the-box baselines perform well under truthful reporting, but not with strategic reporting. Our methods are robust to strategic withholding, and in line with the earlier discussion on the potential issues faced by Mincut and IC-LR (in Sections 3 and 5), we see that **(a)** HC(LR) performs most consistently, and **(b)** in some cases, Mincut (e.g., Poland) and IC-LR (e.g., Taiwan) do not do well.

With discretization (Table 4): As expected, discretization of numerical features into binary categories improves the performance of Mincut and IC-LR, for reasons explained in Sections 3 and 5 respectively. We also see some benefit from discretization for HC(LR) when the features are mostly continuous (e.g., Poland), and less so when they are already discrete (e.g., Taiwan).

With a large number of features (Table 5): We see broadly similar trends here, except that in the case with discretization, IC-LR performs much better than before (e.g., Poland). The reason for this

---

[12]According to ANOVA F-value evaluated before dropping any feature values.

[13]Such a technique can be applied to other methods too – the results (see the Supplement) are not very different from those in Tables 4.

Table 4: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, 4 features ("w/ disc." stands for "with discretization of features")

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR) w/ disc.	.794	.794	.641	.641	.692	.692	.650	**.650**
Mincut w/ disc.	.789	.789	.629	.629	.692	.692	.649	.649
IC-LR w/ disc.	**.800**	**.800**	.651	**.651**	.698	**.698**	.646	.646
Imp(LR) w/ disc.	.799	.762	**.652**	.577	**.719**	.631	**.686**	.541
R-F(LR) w/ disc.	.796	.542	.633	.516	.708	.522	.684	.587

Table 5: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, all features

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCFs(LR)	.795	.795	.625	.625	.678	.678	.648	.648
HCApp(LR)	.777	.777	.617	.617	.658	.658	.638	.638
Mincut	.496	.496	.499	.499	.499	.499	.499	.499
IC-LR	.798	.798	.654	**.654**	.607	.607	.588	.588
HCFs(LR) w/ disc.	.794	.794	.632	.632	.694	.694	.649	.649
HCApp(LR) w/ disc.	.782	.782	.620	.620	.724	.724	.644	.644
Mincut w/ disc.	.534	.534	.503	.503	.499	.499	.550	.550
IC-LR w/ disc.	.805	**.805**	.653	.653	.773	**.773**	.667	**.667**
Imp(LR)	.802	.701	**.663**	.523	.729	.507	.657	.501
Imp(LR) w/ disc.	**.809**	.723	.659	.554	**.783**	.503	**.697**	.501

7 CONCLUSION

In this paper, we studied the problem of classification when each agent at prediction time can strategically withhold some of its features to obtain a more favorable outcome. We devised classification methods (Mincut, HC and IC-LR) that are robust to this behavior, and in addition, characterized the space of all possible truthful classifiers in our setting. We tested our methods on real-world data sets, showing that they outperform out-of-the-box methods that do not account for the aforementioned strategic behavior.

An immediate question that follows is relaxing the assumption of having access to truthful training data – for example, one could ask what the best incentive-compatible classifier is given that the training data consists of best responses to a known classifier f; or, one could consider an online learning model where the goal is to bound the overall loss over time. A much broader question for future work is to develop a more general theory of robustness to missing data that naturally includes the case of strategic withholding.

is that IC-LR is able to use all the available features once they are discretized into binary categories. However, without discretization, HC methods are more reliable (e.g., Poland and Taiwan).

On the out-of-the-box baselines: • *Imputation-based methods* are sensitive vis-á-vis the mean/mode values used. There is incentive to drop a certain feature if the imputed value is a positive signal. If there are many such features, then these methods perform poorly, as seen in Table 5 (cf. Table 3, Australia). If the imputed values do not give a clear signal (e.g., when the distribution of each feature value is not skewed), there is a high variance in the performance of these methods (see the Supplement). In some cases, the benchmarks perform as well as, or slightly better than, our incentive-compatible classifiers. For example, in Table 3, for the Australia and Poland data sets, the accuracy of Imp(LR) and that of HC(LR) are within 0.001 of each other. This happens because the imputed values are, in these cases (but not in most of our other cases), negative indicators of the positive label, and therefore there is generally no incentive to strategically drop features. • *Reduced-Feature modeling*, despite performing well under truthful reporting, allows too many examples to be accepted under strategic reporting, which hurts its performance. This is true especially for smaller ϵ, as each subclassifier has fewer examples to train on, giving several viable options for strategic withholding.

We note here that the variance (in the accuracy achieved) produced by our methods, since they are robust to strategic withholding, is much smaller than that of the baseline methods (exact numbers are deferred to the Supplement).

REFERENCES

[1] Caralee J. Adams. 2017. *In Race for Test-Takers, ACT Outscores SAT—for Now.* https://www.edweek.org/ew/articles/2017/05/24/in-race-for-test-takers-act-outscores-sat--for.html

[2] N Chawla, N Japkowicz, and A Kolcz. 2003. ICML'2003 Workshop on Learning from Imbalanced Data Sets (II). In *Proceedings available at http://www. site. uottawa. ca/nat/Workshop2003/workshop2003. html.*

[3] Nitesh V Chawla, Nathalie Japkowicz, and Aleksander Kotcz. 2004. Special issue on learning from imbalanced data sets. *ACM SIGKDD explorations newsletter* 6, 1 (2004), 1–6.

[4] Yiling Chen, Chara Podimata, Ariel D Procaccia, and Nisarg Shah. 2018. Strategyproof linear regression in high dimensions. In *Proceedings of the 2018 ACM Conference on Economics and Computation.* 9–26.

[5] Bryan Conroy, Larry Eshelman, Cristhian Potes, and Minnan Xu-Wilson. 2016. A dynamic ensemble approach to robust classification in the presence of missing data. *Machine Learning* 102, 3 (2016), 443–463.

[6] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. 2009. *Introduction to algorithms.* MIT press.

[7] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, and Deepak Verma. 2004. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.* 99–108.

[8] Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. 2015. Multiclass learnability and the ERM principle. *The Journal of Machine Learning Research* 16, 1 (2015), 2377–2404.

[9] Ofer Dekel, Felix Fischer, and Ariel D Procaccia. 2010. Incentive compatible regression learning. *J. Comput. System Sci.* 76, 8 (2010), 759–777.

[10] Ofer Dekel, Ohad Shamir, and Lin Xiao. 2010. Learning to classify with missing and corrupted features. *Machine learning* 81, 2 (2010), 149–178.

[11] Thomas G Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation* 10, 7 (1998), 1895–1923.

[12] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[13] Charles Elkan. 2001. The foundations of cost-sensitive learning. In *International joint conference on artificial intelligence*, Vol. 17. Lawrence Erlbaum Associates Ltd, 973–978.

[14] Usama M. Fayyad and Keki B. Irani. 1993. Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence. Chambéry, France, August 28 - September 3, 1993*, Ruzena Bajcsy (Ed.). Morgan Kaufmann, 1022–1029. http://ijcai.org/Proceedings/93-2/Papers/022.pdf

[15] Raquel Florez-Lopez. 2010. Effects of Missing Data in Credit Risk Scoring. A Comparative Analysis of Methods to Achieve Robustness in the Absence of Sufficient Data. *The Journal of the Operational Research Society* 61, 3 (2010), 486–501.

[16] Amir Globerson and Sam Roweis. 2006. Nightmare at test time: robust learning by feature deletion. In *Proceedings of the 23rd international conference on Machine learning.* 353–360.

[17] Jerry R Green and Jean-Jacques Laffont. 1986. Partially verifiable information and mechanism design. *The Review of Economic Studies* 53, 3 (1986), 447–456.

[18] Nika Haghtalab, Nicole Immorlica, Brendan Lucier, and Jack Wang. 2020. Maximizing Welfare with Incentive-Aware Evaluation Mechanisms. In *29th International Joint Conference on Artificial Intelligence.*

[19] David J Hand and Christoforos Anagnostopoulos. 2013. When is the area under the receiver operating characteristic curve an appropriate measure of classifier performance? *Pattern Recognition Letters* 34, 5 (2013), 492–495.

[20] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. 2016. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science.* 111–122.

[21] N Japkowicz. 2000. AAAI'2000 Workshop on Learning from Imbalanced Data Sets. *AAAI Tech Report, No. WS-00-05* (2000).

[22] Jongbin Jung, Connor Concannon, Ravi Shroff, Sharad Goel, and Daniel G Goldstein. 2017. Simple rules for complex decisions. *Available at SSRN 2919024* (2017).

[23] Andrew Kephart and Vincent Conitzer. 2015. Complexity of mechanism design with signaling costs. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems.* 357–365.

[24] Andrew Kephart and Vincent Conitzer. 2016. The revelation principle for mechanism design with reporting costs. In *Proceedings of the 2016 ACM Conference on Economics and Computation.* 85–102.

[25] Jon Kleinberg and Manish Raghavan. 2019. How Do Classifiers Induce Agents to Invest Effort Strategically?. In *Proceedings of the 2019 ACM Conference on Economics and Computation.* 825–844.

[26] Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C Thomas. 2015. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research* 247, 1 (2015), 124–136.

[27] Benjamin Marlin. 2008. *Missing data problems in machine learning.* Ph.D. Dissertation. Citeseer.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[29] Maytal Saar-Tsechansky and Foster Provost. 2007. Handling missing values when applying classification models. *Journal of machine learning research* 8, Jul (2007), 1623–1657.

[30] Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms.* Cambridge university press.

[31] Umar Syed and Ben Taskar. 2010. Semi-supervised learning with adversarially missing label information. In *Advances in Neural Information Processing Systems.* 2244–2252.

[32] Yevgeniy Vorobeychik and Murat Kantarcioglu. 2018. Adversarial machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 12, 3 (2018).

[33] Fulton Wang and Cynthia Rudin. 2015. Falling rule lists. In *Artificial Intelligence and Statistics.* 1013–1022.

[34] Lan Yu. 2011. Mechanism design with partial verification and revelation principle. *Autonomous Agents and Multi-Agent Systems* 22, 1 (2011), 217–223.

[35] Hanrui Zhang, Yu Cheng, and Vincent Conitzer. 2019. Distinguishing Distributions When Samples Are Strategically Transformed. In *Advances in Neural Information Processing Systems.* 3187–3195.

[36] Hanrui Zhang, Yu Cheng, and Vincent Conitzer. 2019. When samples are strategically selected. In *International Conference on Machine Learning.* 7345–7353.

[37] Hanrui Zhang, Yu Cheng, and Vincent Conitzer. 2021. Automated Mechanism Design for Classification with Partial Verification. In *Proceedings of the AAAI Conference on Artificial Intelligence.*

[38] Hanrui Zhang, Yu Cheng, and Vincent Conitzer. 2021. Classification with Few Tests through Self-Selection. In *Proceedings of the AAAI Conference on Artificial Intelligence.*

[39] Hanrui Zhang and Vincent Conitzer. 2021. Incentive-Aware PAC Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence.*

A FURTHER RELATED WORK

Our work can, in a way, be thought of as studying an adversarial classification [see, e.g., 32] problem – in particular, a decision-time, white-box, targeted attack on binary classifiers, assuming that the only strategy available to the attacker is to remove feature values, and the attacker's goal is to maximize the number of instances classified as positive. In this regard, what we study is similar in spirit to some of the existing literature [10, 16, 31] on adversarial classification.

For example, [16] consider a problem where, at test time, the attacker can set up to a certain number of features (say pixels in an image) to zero for each instance individually in a way that is most harmful to the classifier chosen. To be robust to such attacks, they devise convex programming based methods that avoid depending on small sets of features to learn the class structure. Our work is different in that we take a more game-theoretic approach to designing classifiers (including ensemble-based ones) that are fully resistant to the strategic withholding of features by agents (that prefer being labeled positively). Moreover, we make no assumptions on the actual structure of the feature space.

Our work is also related to the literature on classification with missing data (Batista and Monard 2003; Marlin 2008). We devise methods that can deal with the strategic withholding by agents of some of their features, against a backdrop of missing data caused by natural reasons (e.g., nonstationary distributions of which features are present, or input sensor failures). The HC method can be viewed as an ensemble method for missing data [5] that is strategy-proof against the aforementioned strategic behavior. We also study the performance of other standard, non-strategic classification methods for missing data in the strategic setting, including predictive value imputation and reduced-feature modeling [29].

B PROOFS

Theorem 1. *If $(S, \bar{S})$ is a minimum s-t cut of $G(\mathcal{D}, \mathcal{X})$ (where S is on the same side as s), then for the classifier $f^*(x) := \mathbb{1}(x \in \bar{S})$, we have $\ell_{\mathcal{D}}(f^*) = \min_f \ell_{\mathcal{D}}(f)$.*

Proof. First observe that any classifier f can be viewed equivalently as a subset of $\mathcal{X}$, given by

$$\{x \in \mathcal{X} \mid f(x) = 1\}.$$

Below, we use these interpretations, i.e., as a function or a subset, of a classifier interchangeably.

The loss of a truthful classifier f can then be written as

$$
\begin{aligned}
\ell_{\mathcal{D}}(f) &= \Pr_{(x,y)\sim\mathcal{D}}[(x \in f \wedge y = 0) \vee (x \notin f \wedge y = 1)] \\
&= \Pr_{(x,y)\sim\mathcal{D}}[x \in f \wedge y = 0] + \Pr_{(x,y)\sim\mathcal{D}}[x \notin f \wedge y = 1] \\
&= \sum_{x \in f} \Pr_{(x',y)\sim\mathcal{D}}[x' = x \wedge y = 0] + \sum_{x \notin f} \Pr_{(x',y)\sim\mathcal{D}}[x' = x \wedge y = 1] \\
&= \sum_{x \in f} \mathcal{D}^-(x) + \sum_{x \notin f} \mathcal{D}^+(x),
\end{aligned}
$$

where the last line follows from Definition 1.

Therefore, our goal is to solve the following optimization problem:

$$
\begin{aligned}
\min_{f \subseteq \mathcal{X}} \quad & \sum_{x \in f} \mathcal{D}^-(x) + \sum_{x \notin f} \mathcal{D}^+(x) \\
\text{s.t.} \quad & x' \in f \implies x \in f \qquad \forall x, x' \in \mathcal{X} \text{ where } x \to x'.
\end{aligned}
$$

Consider the following min-cut formulation (using Definition 2): Let $G(\mathcal{D}, \mathcal{X})$ be a directed capacitated graph, with vertices $V = \mathcal{X} \cup \{s, t\}$, with edges E and edge capacities u defined as follows:

- For each $x \in \mathcal{X}$, there is an edge $(s, x) \in E$ with capacity $\mathcal{D}^-(x)$, and an edge $(x, t) \in E$ with capacity $\mathcal{D}^+(x)$.
- For each pair $x, x' \in \mathcal{X}$ where $x \to x'$, there is an edge $(x', x) \in E$ with capacity ∞.

Observe that each finite-capacity s-t cut $(S, \bar{S})$ corresponds bijectively to a truthful classifier $f := \mathbb{1}(x \in \bar{S} \setminus \{t\})$. Moreover, the capacity of the cut is given precisely by

$$\sum_{x \in \bar{S} \cap \mathcal{X}} \mathcal{D}^-(x) + \sum_{x \in S \cap \mathcal{X}} \mathcal{D}^+(x) = \sum_{x \in f} D^-(x) + \sum_{x \notin f} D^+(x) = \ell_{\mathcal{D}}(f).$$

Therefore, any s-t min-cut corresponds to an optimal classifier f^*, which can be computed "efficiently" (i.e., in time $\text{poly}(|\mathcal{X}|)$) using any efficient max-flow algorithm given complete knowledge of $\mathcal{D}$. □

Theorem 2. *A classifier f is truthful iff $f(.) = \text{Max}_C(.)$ for a collection of classifiers $C = \{f_j\}$ such that, for some $\{F_j\} \subseteq 2^F$, each f_j is an F_j-classifier .*

PROOF. Recall that the revelation principle holds in our setting (as mentioned in Section Section 2, also see Proposition 1). It therefore suffices to characterize all direct revelation classifiers. For any (not necessarily truthful) classifier f, consider its direct revelation implementation f', which maps feature values x to the most desirable label the data point can get by dropping features, i.e.,

$$f'(x) = \max_{x':x \to x'} f(x').$$

We argue below that f' has the desired form.

Observe that depending on which features a data point x has, f can be decomposed into 2^k subclassifiers, denoted $\{f_F\}_{F \subseteq [k]}$. The label of x is then determined in the following way: let F_x be the set of features possessed by x, i.e.,

$$F_x = \{i \in [k] \mid x_i \neq *\}.$$

Then

$$f(x) = f_{F_x}(x).$$

Moreover, observe that (1) f_F effectively depends only on $x|_F$ (i.e., $f_F(x) = f_F(x|_F)$), since f_F only acts on those data points where all features not in F are missing, and (2) without loss of generality, f_F rejects any data point with a missing feature $i \in F$, since f_F never acts on a data point where such a feature $i \in F$ is missing. Now consider how f' works on a data point x. For any $F \subseteq F_x$, by dropping all features not in F, x can report $x|_F$. Moreover, for any such $F \subseteq F_x$, $f(x|_F) = f_F(x|_F)$. f' outputs 1 for x, iff there exists $F \subseteq F_x$, such that $f_F(x|_F) = 1$. One can therefore write f' in the following way: for any $x \in \mathcal{X}$,

$$f'(x) = \max_{F \subseteq [k]} f_F(x),$$

as desired. □

Theorem 3. *Let $\widehat{\mathcal{X}} = \{(x_i, y_i)\}_{i \in [m]}$ be m i.i.d. samples from $\mathcal{D}$. For any $f \in \bar{\mathcal{H}}$, for any $\delta > 0$, with probability at least $1 - \delta$, we have*

$$\ell_{\mathcal{D}}(f) \leq \ell_{\widehat{\mathcal{X}}}(f) + O\left(\sqrt{\frac{dn \cdot \log dn \cdot \log m + \log(1/\delta)}{m}}\right).$$

PROOF. Recall that $\bar{\mathcal{H}}$ is defined as the set of all classifiers that can be written as the MAX Ensemble of n classifiers in $\mathcal{H}$. Given the classical VC inequality [e.g., 30, Theorem 6.11], we only need to bound the VC dimension of $\bar{\mathcal{H}}$, and show that

$$d_{VC}(\bar{\mathcal{H}}) = O(dn \cdot \log dn),$$

where d is the VC dimension of $\mathcal{H}$. To this end, observe that each $f \in \bar{\mathcal{H}}$ is essentially a decision tree with $n + 1$ leaves, where each leaf is associated with a binary label, and each internal node corresponds to a classifier in $\mathcal{H}$. To be precise, f can be computed in the following way: for any $x \in \mathcal{X}$, if $f_1(x) = 1$, then $f(x) = 1$; otherwise, if $f_2(x) = 1$, then $f(x) = 1$, etc. It is known [see, e.g., 8, Section 5.2] that the class of all such decision trees with $n + 1$ leaves, which is a superset of $\bar{\mathcal{H}}$, has VC dimension $O(dn \log dn)$. As a result, $d_{VC}(\bar{\mathcal{H}}) = O(dn \log dn)$, and the theorem follows. □

Theorem 4. *Algorithm 1 converges.*

PROOF. Given $f = \text{MAX}_{\{f_1^t, f_2^t, \ldots, f_n^t\}}$, consider a single update step for, say, f_1^t. As in Algorithm 1, define:

$$S_1 = \{(x, y) \in \widehat{\mathcal{X}} : f_j^t(x|_{F_j}) = 0, \forall j \neq 1\},$$
$$S_{-1} = S \setminus S_1.$$

Then we perform the update as follows:

$$f_1^{t+1} = \text{argmin}_{h \in \mathcal{H}} \sum_{(x,y) \in S_1} |h(x|_{F_1}) - y|.$$

Let $f' = \text{MAX}_{\{f_1^{t+1}, f_2^t, \ldots, f_n^t\}}$. Now, the loss calculated for f' is

$$\begin{aligned}
\ell_{\widehat{\mathcal{X}}}(f') &= \frac{1}{m}(|S_1| \cdot \ell_{S_1}(f') + |S_{-1}| \cdot \ell_{S_{-1}}(f')) \\
&= \frac{1}{m}(|S_1| \cdot \ell_{S_1}(f_1^{t+1}) + |S_{-1}| \cdot \ell_{S_{-1}}(f')) \\
&= \frac{1}{m}(|S_1| \cdot \ell_{S_1}(f_1^{t+1}) + |S_{-1}| \cdot \ell_{S_{-1}}(f)) \\
&\leq \frac{1}{m}(|S_1| \cdot \ell_{S_1}(f_1^t) + |S_{-1}| \cdot \ell_{S_{-1}}(f)) \\
&= \ell_{\widehat{\mathcal{X}}}(f).
\end{aligned}$$

The inequality in the above sequence of steps follows from the fact that f_j^{t+1} accrues a lower loss on S_1 than f_j^t by definition, and that the classification outcomes for any $(x, y) \in S_{-1}$ is the same for f and f'.

If we treat $\ell_{\widehat{X}}(f)$ as a potential function, we can see that it can only decrease with each step, and therefore, the algorithm has to converge at some point. □

C   REVELATION PRINCIPLE

There are many results in the literature on partial verification as to the validity of the revelation principle in various settings [17, 23, 24, 34]. For our purposes, as mentioned in Section 2, when the reporting structure is given by a partial order (i.e., it is transitive, meaning for any $x_1, x_2, x_3, x_1 \to x_2$ and $x_2 \to x_3 \implies x_1 \to x_3$), the revelation principle holds. Below we give a quick proof for why this is the case in our setting.

Proposition 1. *For any classifier $f : X \to \{0, 1\}$, there is a truthful classifier f' such that after misreporting, f and f' output the same label for all $x \in X$, i.e.,*

$$f'(x) = \max_{x':x \to x'} f(x').$$

Proof. Below we explicitly construct f'. Let f' be such that for $x \in X$,

$$f'(x) = \max_{x':x \to x'} f(x).$$

Clearly f' and f output the same label after strategic manipulation. We only need to show f' is truthful, i.e., for any $x_1, x_2 \in X$ where $x_1 \to x_2, f'(x_1) \geq f'(x_2)$. Let $X_1 = \{x' : x_1 \to x'\}$ and $X_2 = \{x' : x_2 \to x'\}$. Recall that $\to$ is transitive and $x_1 \to x_2$, so $X_1 \supseteq X_2$. Now we have

$$f'(x_1) = \max_{x \in X_1} f(x) \geq \max_{x \in X_2} f(x) = f'(x_2). \qquad\qquad □$$

Note that the above proof crucially depends on transitivity of the reporting structure. In fact, if the reporting structure $\to$ is not transitive, then the revelation principle in general does not hold. To see why this is the case, suppose $\to$ is not transitive, and let x_1, x_2, x_3 be such that $x_1 \to x_2, x_2 \to x_3$, and $x_1 \not\to x_3$. Suppose we want to assign label 0 to x_1, and label 1 to x_2 and x_3, then the only way to achieve that is to implement a classifier f where $f(x_1) = f(x_2) = 0$ and $f(x_3) = 1$. However, this classifier is not truthful, since x_2 always misreports as x_3 in order to be accepted.

D   OTHER OBSERVATIONS

D.1   Regarding MINCUT

Naturally, the test error of MINCUT depends on X and m. For example, If X is discrete and small, one would expect that MINCUT is almost optimal given enough samples. However, when X is large or even infinite, the generalization gap can be extremely large. To see why this is true, consider the following example:

Example 2. *Say we are given a feature space with two features, each of which can take any real value between 0 and 1. Let the marginal distribution of $\mathcal{D}$ on X be the uniform distribution over $X = \{(x, y), (x, *), (*, y) \mid x, y \in [0, 1]\}$. When we see a new data point (x, y), unless we already have $(x, y), (x, *)$ or $(*, y)$ in the set of samples (which happens with probability 0), we know absolutely nothing about the label of (x, y), and therefore by no means we can expect f' to predict the label of (x, y) correctly — in fact, f' will always assign label 0 to such a data point.*

D.2   On truthful classifiers and hill-climbing

Below, we make a few remarks regarding the generalization bound (Theorem 3) for HC.

- Observe that the generalization gap depends polynomially on the number of subclassifiers n. Without additional restrictions, n can be as large as 2^k leading to a gap which is exponential in k. This suggests that in practice, to achieve any meaningful generalization guarantee, one has to restrict the number of subclassifiers used. In fact, we do run our algorithm on a small set of features in Section 6.
- Recall that the class of linear classifiers in the k-dimensional Euclidean space has VC dimension $k + 1$. So, if we restrict all subclassifiers to be linear, and require that the number of subclassifiers n to be constant, then Theorem 3 implies that with high probability, the generalization gap is

$$\widetilde{O}\left(\sqrt{\frac{k}{m}}\right),$$

where k is the number of features, m is the number of samples, and $\widetilde{O}$ hides a logarithmic factor. Our algorithms are practicable in this kind of regime.

E EXPERIMENTS

E.1 Implementation details

In our implementation, we use Python's Scikit-learn (0.22.1) package [28] of classifiers and other machine learning packages whenever possible. The categorical features in the datasets are one-hot encoded. To help ensure the convergence of gradient-based classifiers, we then standardize features by removing the mean and scaling to unit variance.

For imputation-based classifiers, we use mean/mode imputation: mean for numerical and ordinal features, and mode for categorical features. For reduced-feature-based classifiers, we default to reject if the test data point's set of available features was unseen in the training process. For classification methods involving Fayyad and Irani's MDLP discretization algorithm, we use a modified version of `Discretization-MDLPC`, licensed under GPL-3[14].

The performance of each classifier under each setting is evaluated with Nx2-fold cross-validation [11]: training on 50% of the data and testing on the remaining 50%; repeat N = 100 times. To tune the parameters for the classifiers, we perform grid search over a subset of the parameter space considered by [26], in a 5-fold cross-validation on every training set of the (outer) Nx2-cross-validation loop.

E.2 Additional experimental results

We evaluate our methods on datasets with 1) 4 selected features, same for all runs (Table 6 to 29), 2) 4 selected features, based on the training data at each run (Table 30 to 53), and 3) all available features (Table 54 to 77). In addition, we evaluate our methods both with and without balancing the datasets through random undersampling. This is denoted by "balanced datasets" when we undersample before the experiment, and "unbalanced datasets" when we do not. We vary ϵ from 0 to .5 and report classifier accuracy and AUC (when applicable).

Comparing Figure 1 and 3, it appears that there is no significant difference in relative accuracy across the various methods when applied to balanced and imbalanced datasets. However, comparing, for example, Table 8 and 20, we observe that when the dataset is highly unbalanced, classifiers based on imputation and reduced-feature modeling, when faced with strategic reporting, tend to accept everything and yield a considerably high accuracy. Many other general issues regarding the use of accuracy as a metric on unbalanced datasets are known [2, 3, 21]. In practice, thresholding methods are sometimes used to determine a proper threshold for binary prediction in such cases [13, 26].

Therefore, in addition to accuracy, we evaluate our approach with area under the receiver operating characteristic curve (AUC). AUC becomes a useful metric when doing imbalanced classification because often a balance of false-positive and false-negative rates is desired, and it is invariant to the threshold used in binary classification.[15] For Mincut, its receiver operating characteristic curve is undefined because it does not output probabilistic predictions; for Hill-Climbing, we take the maximum of the probabilistic predictions across all applicable classifiers to be the Hill-Climbing classifier's probabilistic prediction for a data point, and obtain AUC from that. From Table 24 to 29, we observe that our three proposed methods generally yield a AUC as good as, if not higher than, imputation- and reduced-feature-based classifiers on imbalanced datasets (also Figure 4). The same holds for balanced datasets too (Figure 2).

For completeness, we also include the performance of classifiers based on imputation and reduced-feature modeling, with discretization. As expected, common classifiers are generally less prone to overfitting than Mincut, and discretizing the feature space only limits their performance.

The number of iteration the training of Hill-Climbing classifiers takes to converge varies by dataset, but usually in no more than than 5 passes through all the subclassifiers. For the balanced Australia dataset, for example, HC(LR) takes an average of 4.28 iterations to converge (median 3); HC(LR) w/ disc. takes an average of 2.45 (median 2); HC(ANN): takes an average of 3.56 (median 3); HC(ANN) w/ disc. takes an average of 2.62 (median 2).

---

[14]`Discretization-MDLPC` codebase: https://github.com/navicto/Discretization-MDLPC.
[15]... although AUC's global perspective assumes implicitly that all thresholds are equally probable. This is often criticized as not plausible in credit scoring [19]

Figure 1: Selected classifier accuracy w/ strategic behavior, balanced datasets, pre-selected 4 features
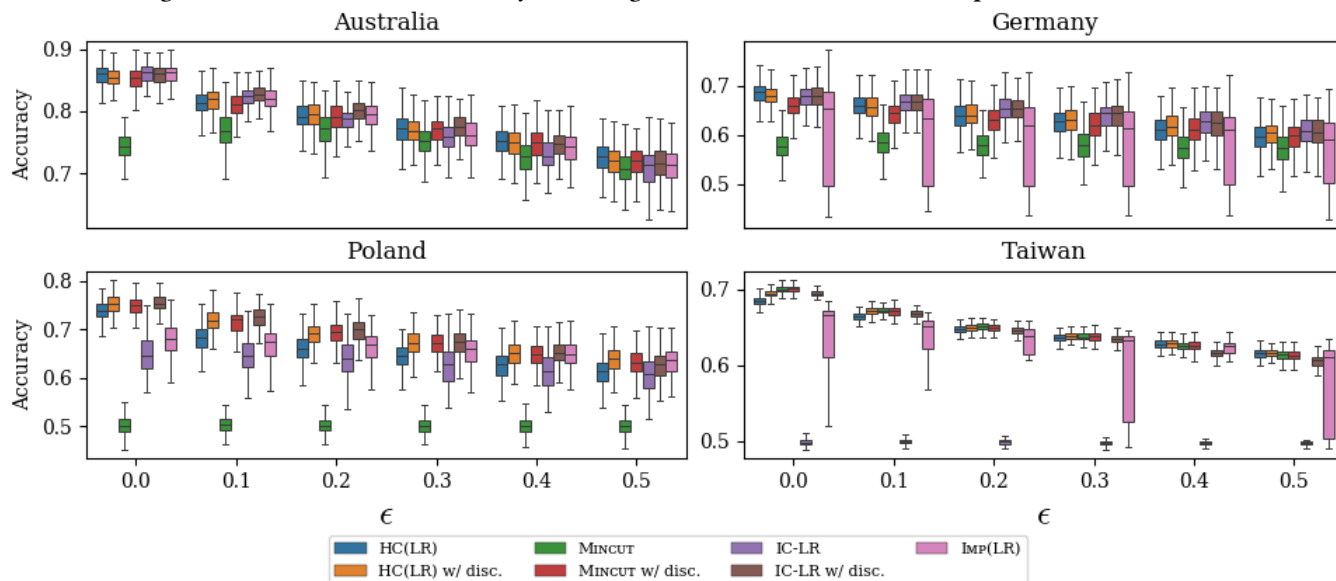
Figure 2: Selected classifier AUC w/ strategic behavior, balanced datasets, pre-selected 4 features
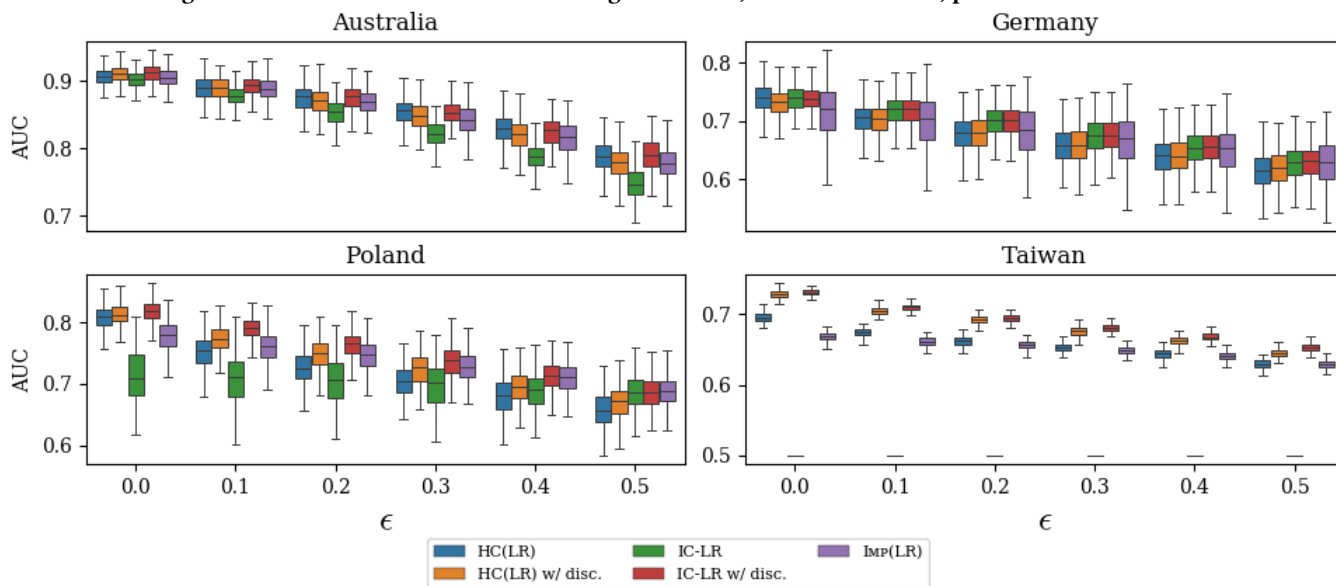
Figure 3: Selected classifier accuracy w/ strategic behavior, unbalanced datasets, pre-selected 4 features
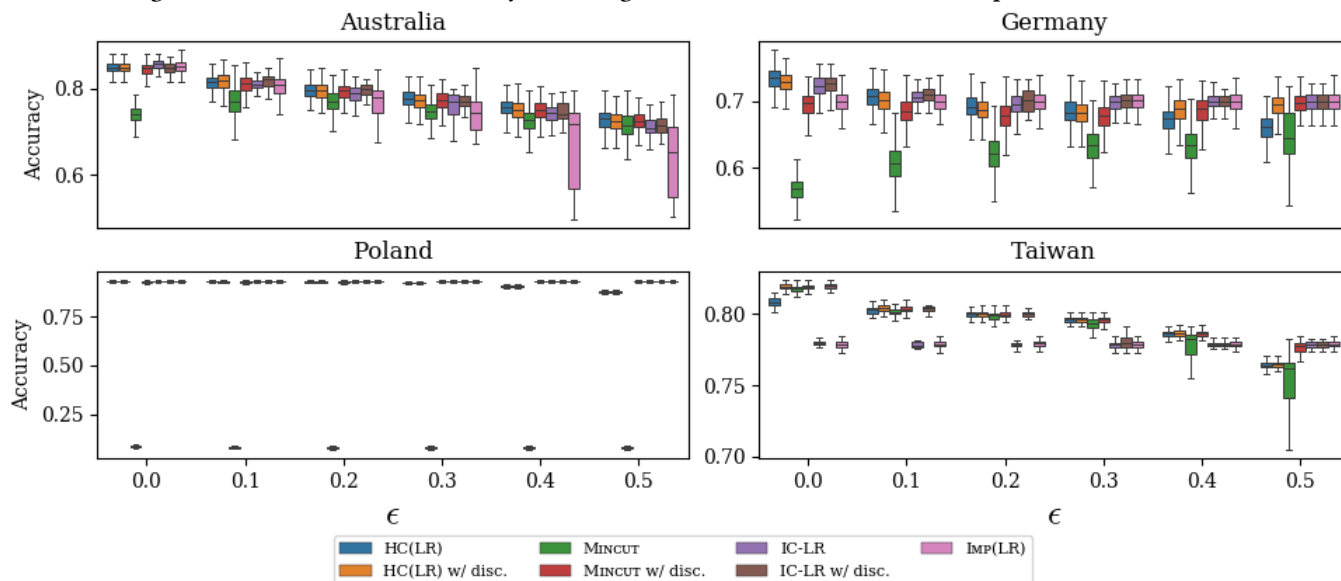
Figure 4: Selected classifier AUC w/ strategic behavior, unbalanced datasets, pre-selected 4 features
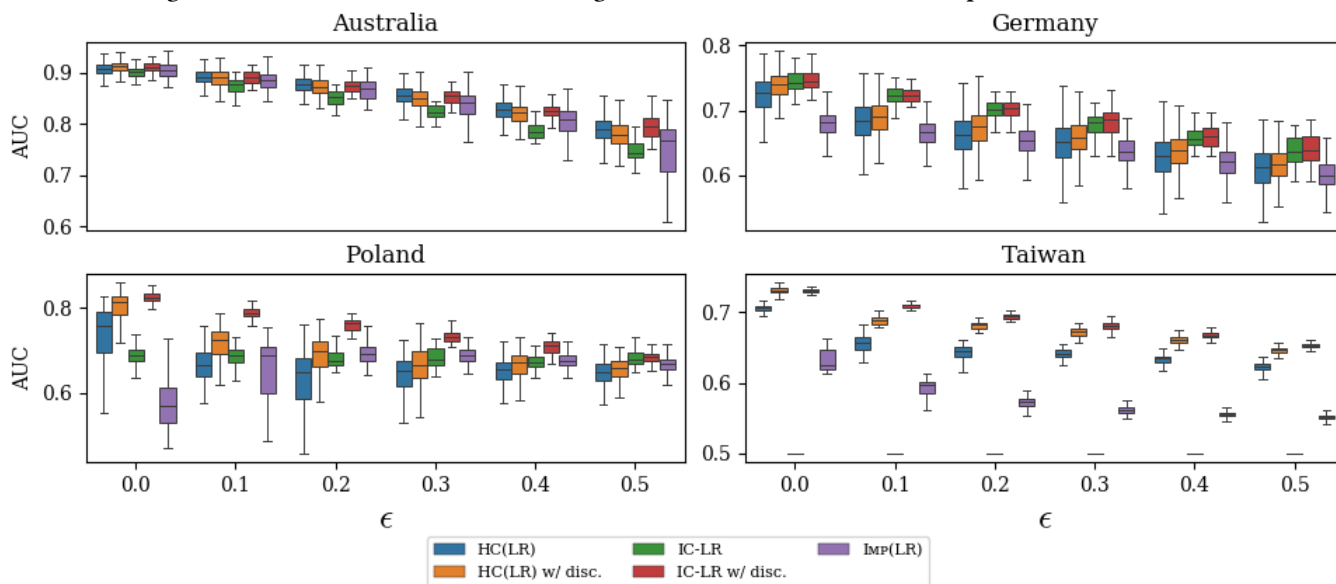
Figure 5: Selected classifier accuracy w/ strategic behavior, balanced datasets, each classifier selects 4 best features
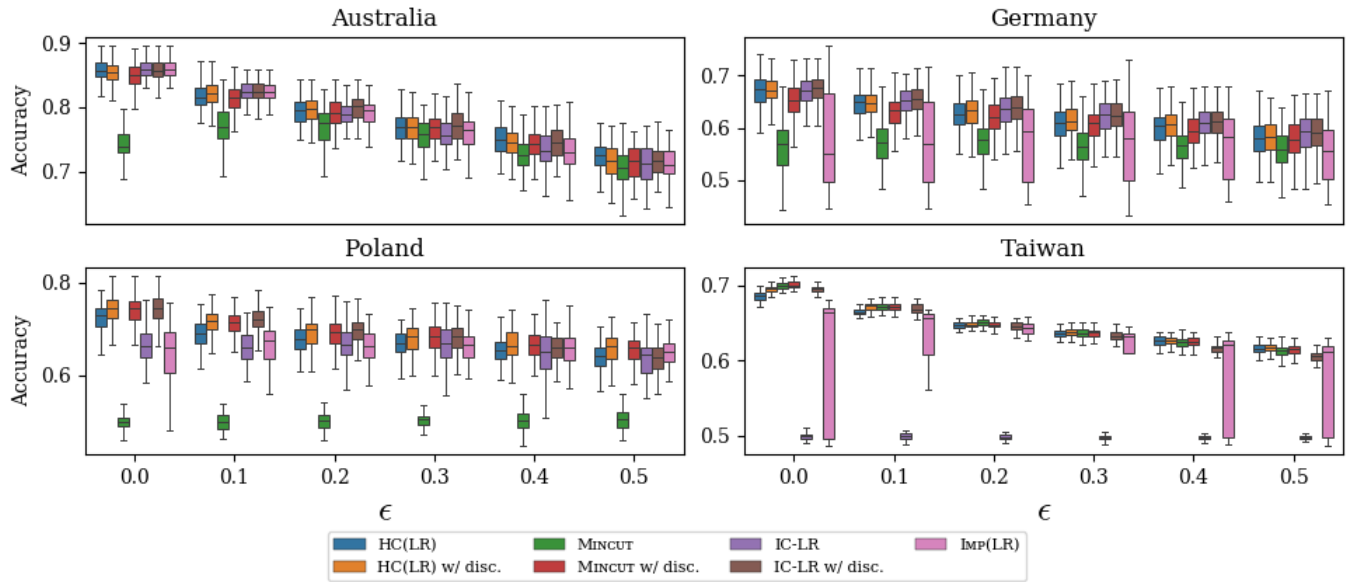
Figure 6: Selected classifier AUC w/ strategic behavior, balanced datasets, each classifier selects 4 best features
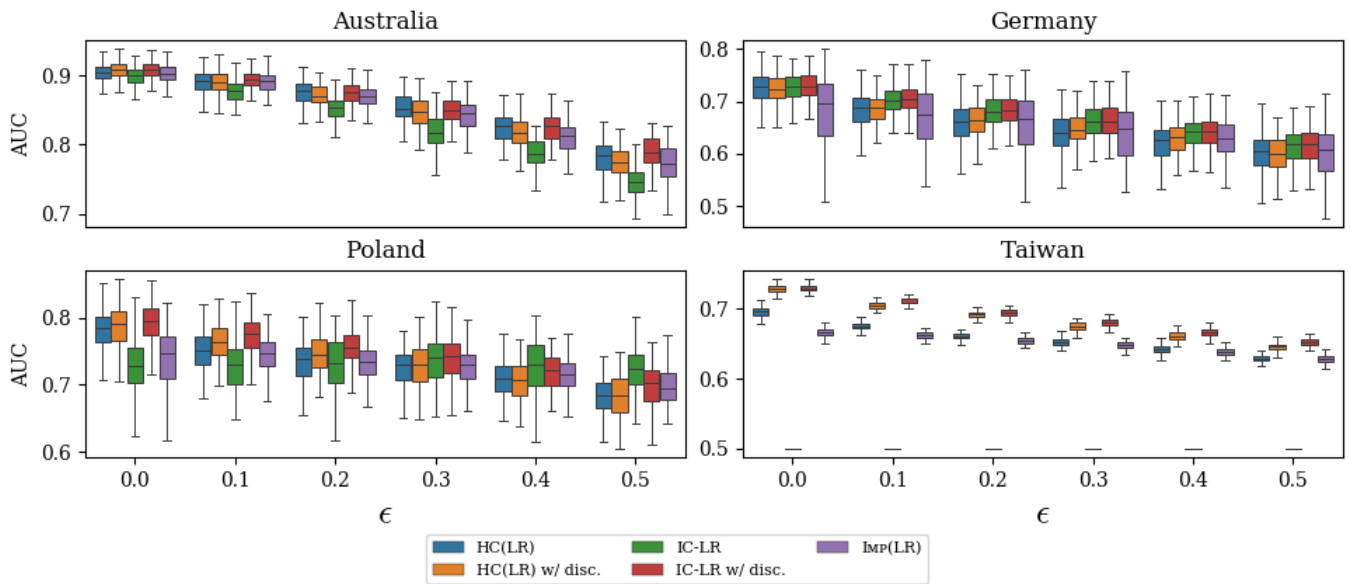
Figure 7: Selected classifier accuracy w/ strategic behavior, unbalanced datasets, each classifier selects 4 best features
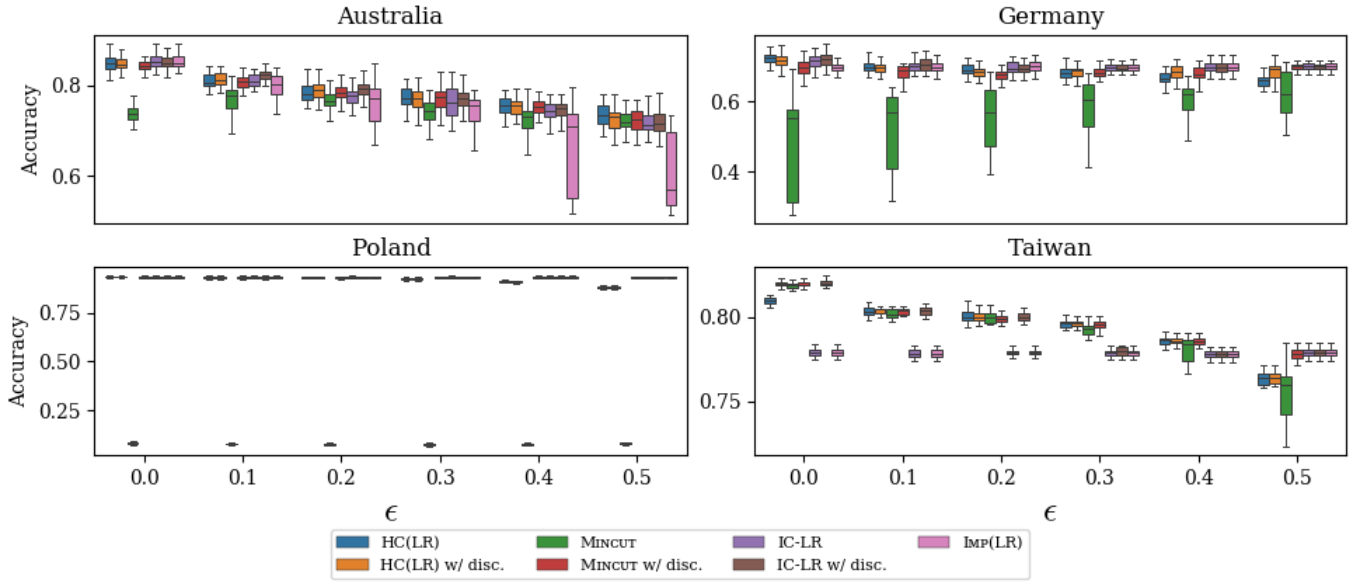
Figure 8: Selected classifier AUC w/ strategic behavior, unbalanced datasets, each classifier selects 4 best features
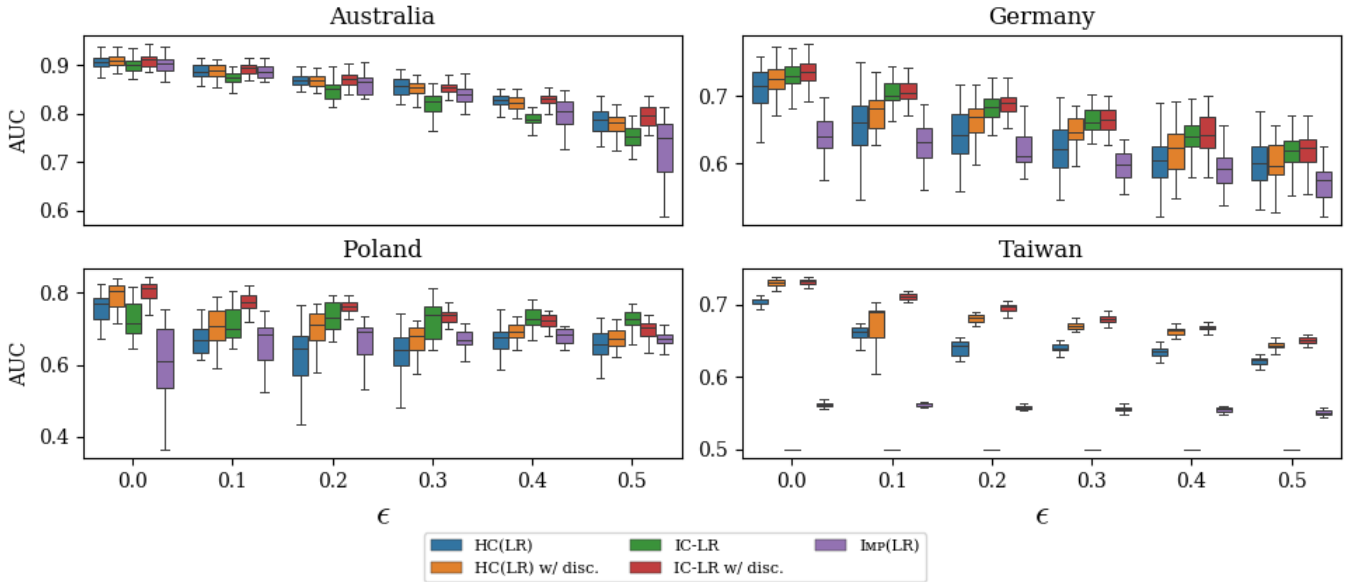
Figure 9: Selected classifier accuracy w/ strategic behavior, balanced datasets, all features
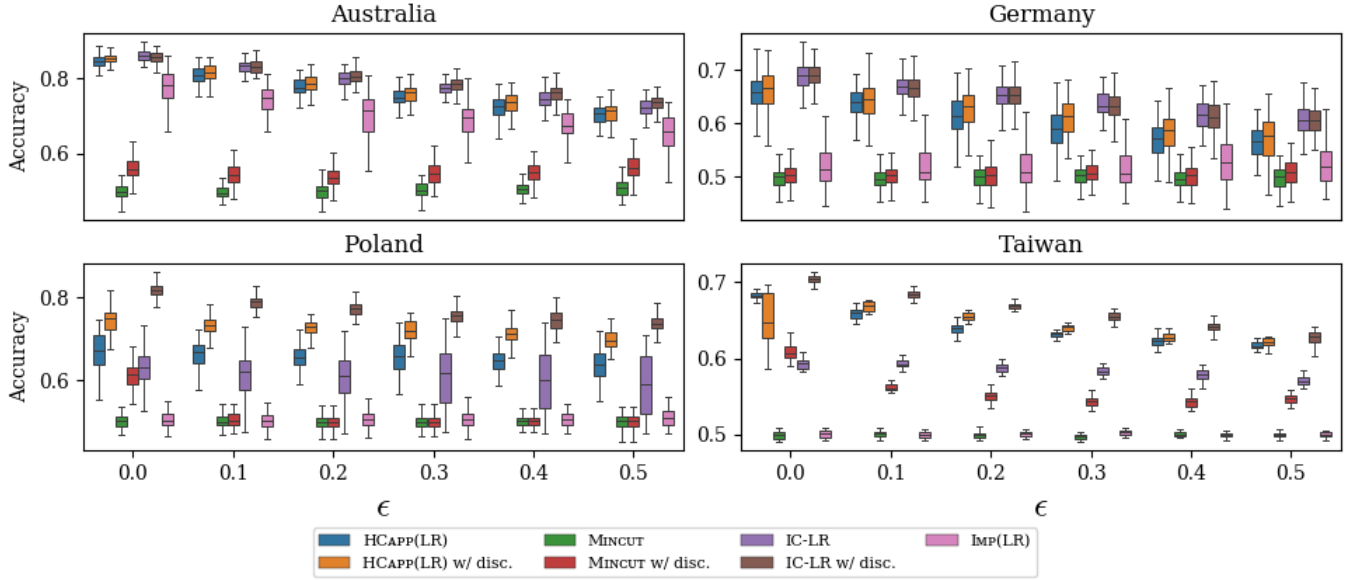
Figure 10: Selected classifier AUC w/ strategic behavior, balanced datasets, all features
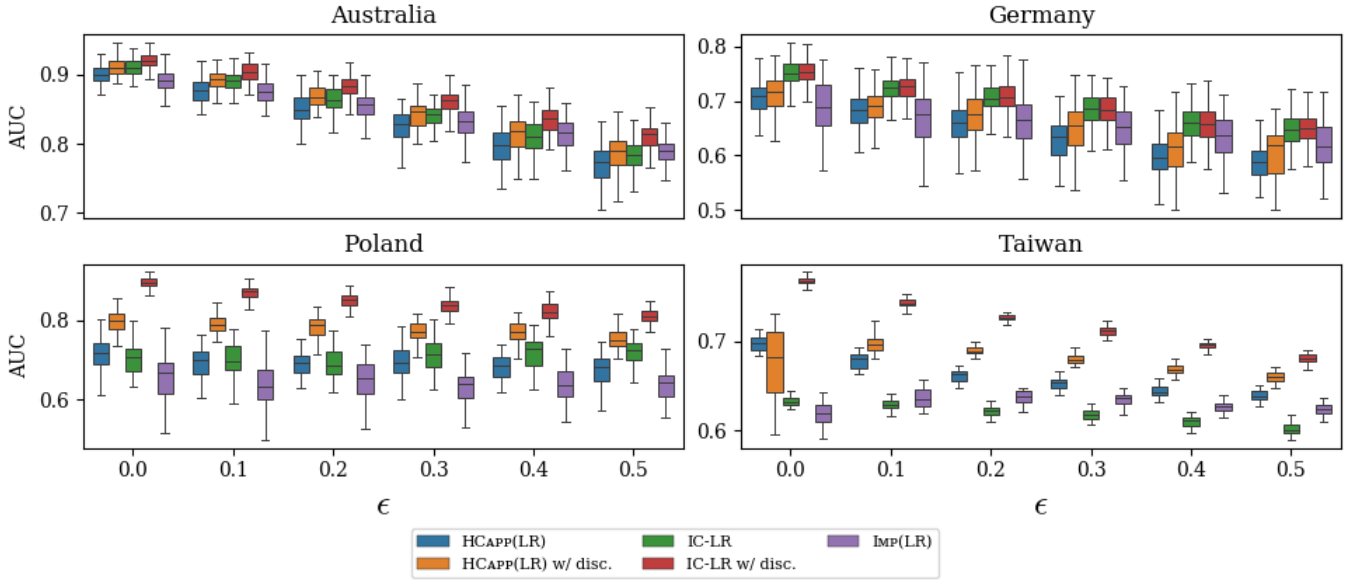
Figure 11: Selected classifier accuracy w/ strategic behavior, unbalanced datasets, all features
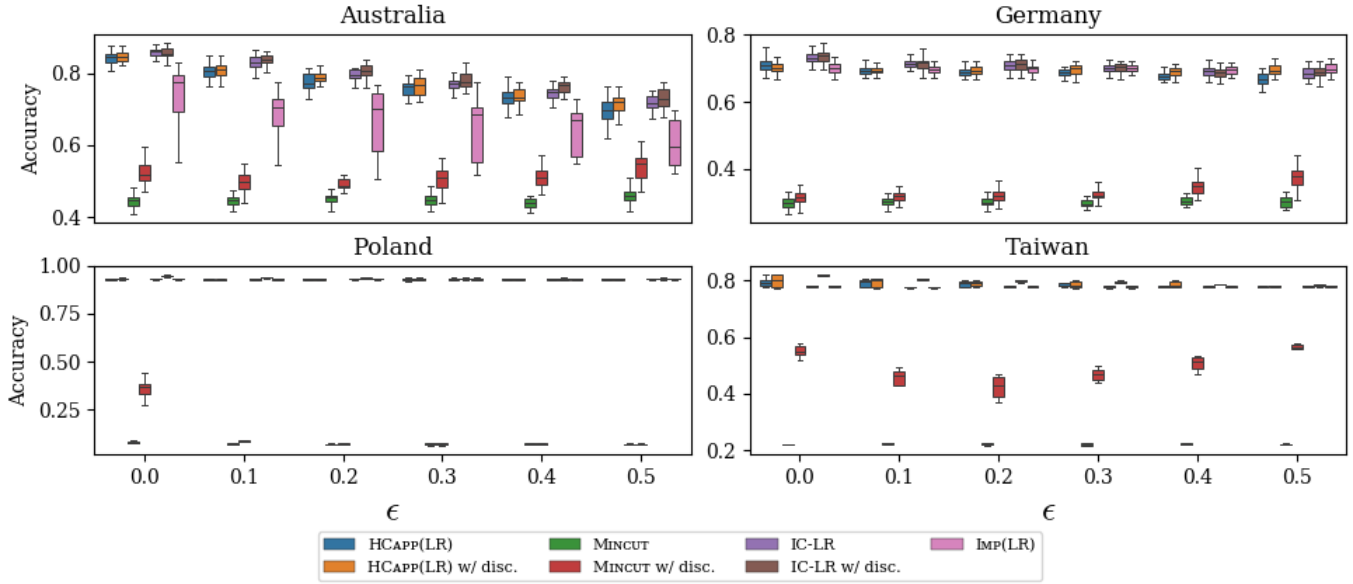
Figure 12: Selected classifier AUC w/ strategic behavior, unbalanced datasets, all features

Table 6: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.859 (.016)	.859 (.016)	.686 (.024)	.686 (.024)	.739 (.022)	.739 (.022)	.685 (.006)	.685 (.006)
HC(LR) w/ disc.	.855 (.016)	.855 (.016)	.679 (.024)	.679 (.024)	.753 (.019)	.753 (.019)	.695 (.005)	.695 (.005)
HCApp(LR)	.860 (.017)	.860 (.017)	.685 (.028)	.685 (.028)	.733 (.023)	.733 (.023)	.649 (.018)	.649 (.018)
HCApp(LR) w/ disc.	.852 (.030)	.852 (.030)	.672 (.030)	.672 (.030)	.745 (.019)	.745 (.019)	.637 (.009)	.637 (.009)
HC(ANN)	.859 (.015)	.859 (.015)	.691 (.023)	.691 (.023)	.729 (.025)	.729 (.025)	.688 (.006)	.688 (.006)
HC(ANN) w/ disc.	.849 (.015)	.849 (.015)	.683 (.023)	.683 (.023)	.754 (.019)	.754 (.019)	.696 (.006)	.696 (.006)
Mincut	.743 (.021)	.743 (.021)	.577 (.028)	.577 (.028)	.501 (.018)	.501 (.018)	.700 (.005)	.700 (.005)
Mincut w/ disc.	.852 (.018)	.852 (.018)	.658 (.027)	.658 (.027)	.748 (.020)	.748 (.020)	.701 (.005)	.701 (.005)
IC-LR	**.862** (.015)	**.862** (.015)	.677 (.025)	.677 (.025)	.650 (.036)	.650 (.036)	.499 (.004)	.499 (.004)
IC-LR w/ disc.	.858 (.015)	.858 (.015)	.680 (.023)	.680 (.023)	.754 (.018)	.754 (.018)	.695 (.004)	.695 (.004)
IC-LR w/ neg.	.861 (.013)	.861 (.013)	.691 (.024)	.691 (.024)	.660 (.038)	.660 (.038)	.687 (.005)	.687 (.005)
Maj	.745 (.021)	.745 (.021)	.579 (.027)	.579 (.027)	.501 (.018)	.501 (.018)	.701 (.005)	.701 (.005)
Maj w/ disc.	.854 (.018)	.854 (.018)	.653 (.026)	.653 (.026)	.749 (.020)	.749 (.020)	.701 (.005)	.701 (.005)
Imp(LR)	.861 (.014)	**.862** (.015)	.692 (.022)	.601 (.097)	.738 (.021)	.676 (.039)	.689 (.006)	.637 (.051)
Imp(LR) w/ disc.	.859 (.016)	.855 (.025)	.682 (.025)	.593 (.093)	**.756** (.019)	.742 (.034)	.695 (.005)	.500 (.004)
Imp(ANN)	.859 (.015)	.859 (.019)	**.693** (.022)	.602 (.098)	.729 (.024)	.664 (.039)	.699 (.005)	.576 (.027)
Imp(ANN) w/ disc.	.849 (.016)	.828 (.038)	.683 (.027)	.596 (.093)	**.756** (.019)	.739 (.036)	.697 (.005)	.503 (.016)
Imp(RF)	.820 (.018)	.698 (.108)	.651 (.026)	.562 (.079)	.747 (.018)	.736 (.023)	.701 (.005)	.531 (.030)
Imp(RF) w/ disc.	.856 (.017)	.849 (.029)	.668 (.026)	.591 (.090)	.750 (.020)	.734 (.039)	**.702** (.005)	.520 (.032)
Imp(κNN)	.852 (.017)	.800 (.085)	.672 (.027)	.585 (.090)	.726 (.023)	.647 (.048)	.674 (.039)	.515 (.024)
Imp(κNN) w/ disc.	.850 (.023)	.815 (.080)	.670 (.032)	.581 (.086)	.741 (.032)	.724 (.051)	.673 (.062)	.497 (.048)
R-F(LR)	.861 (.014)	.861 (.014)	.692 (.022)	.692 (.022)	.738 (.021)	.738 (.021)	.689 (.006)	.689 (.006)
R-F(LR) w/ disc.	.859 (.016)	.859 (.016)	.682 (.025)	.682 (.025)	**.756** (.019)	**.756** (.019)	.695 (.005)	.695 (.005)
R-F(ANN)	.859 (.015)	.859 (.015)	**.693** (.022)	**.693** (.022)	.728 (.026)	.729 (.024)	.699 (.005)	.699 (.005)
R-F(ANN) w/ disc.	.849 (.016)	.849 (.016)	.684 (.024)	.684 (.024)	.755 (.019)	**.756** (.019)	.697 (.005)	.697 (.006)
R-F(RF)	.820 (.019)	.820 (.018)	.651 (.026)	.651 (.026)	.747 (.019)	.747 (.018)	.701 (.005)	.701 (.005)
R-F(RF) w/ disc.	.855 (.017)	.855 (.017)	.668 (.026)	.668 (.026)	.750 (.020)	.750 (.020)	**.702** (.005)	**.702** (.005)
R-F(κNN)	.852 (.017)	.852 (.018)	.672 (.026)	.673 (.027)	.727 (.023)	.726 (.023)	.674 (.040)	.675 (.038)
R-F(κNN) w/ disc.	.848 (.027)	.849 (.023)	.668 (.034)	.669 (.033)	.737 (.042)	.739 (.038)	.671 (.065)	.674 (.061)

Table 7: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.814 (.020)	.814 (.020)	.659 (.027)	.659 (.027)	.681 (.028)	.681 (.028)	.664 (.005)	.664 (.005)
HC(LR) w/ disc.	.818 (.020)	.818 (.020)	.657 (.028)	.657 (.028)	.718 (.021)	.718 (.021)	.671 (.005)	.671 (.005)
HCApp(LR)	.817 (.020)	.817 (.020)	.669 (.026)	.669 (.026)	.694 (.027)	.694 (.027)	.664 (.007)	.664 (.007)
HCApp(LR) w/ disc.	.817 (.022)	.817 (.022)	.660 (.028)	.660 (.028)	.722 (.023)	.722 (.023)	.669 (.009)	.669 (.009)
HC(ANN)	.803 (.024)	.803 (.024)	.647 (.029)	.647 (.029)	.664 (.028)	.664 (.028)	.665 (.006)	.665 (.006)
HC(ANN) w/ disc.	.796 (.026)	.796 (.026)	.634 (.033)	.634 (.033)	.705 (.026)	.705 (.026)	.671 (.005)	.671 (.005)
Mincut	.769 (.033)	.769 (.033)	.584 (.030)	.584 (.030)	.502 (.017)	.502 (.017)	.672 (.005)	.672 (.005)
Mincut w/ disc.	.811 (.021)	.811 (.021)	.642 (.029)	.642 (.029)	.715 (.022)	.715 (.022)	.671 (.005)	.671 (.005)
IC-LR	.823 (.018)	.823 (.018)	.664 (.027)	.664 (.027)	.645 (.037)	.645 (.037)	.500 (.004)	.500 (.004)
IC-LR w/ disc.	**.827** (.018)	**.827** (.018)	.665 (.025)	.665 (.025)	.725 (.021)	**.725** (.021)	.669 (.005)	.669 (.005)
IC-LR w/ neg.	.816 (.017)	.816 (.017)	**.680** (.021)	**.680** (.021)	.661 (.035)	.661 (.035)	.676 (.007)	**.676** (.007)
Maj	.689 (.025)	.726 (.078)	.548 (.026)	.536 (.048)	.502 (.017)	.502 (.017)	.686 (.005)	.565 (.033)
Maj w/ disc.	.816 (.021)	.682 (.096)	.620 (.029)	.545 (.051)	.728 (.023)	.604 (.059)	.693 (.005)	.564 (.037)
Imp(LR)	.821 (.019)	.821 (.026)	.677 (.024)	.594 (.089)	.726 (.024)	.668 (.037)	.676 (.006)	.627 (.047)
Imp(LR) w/ disc.	.822 (.017)	.795 (.048)	.664 (.025)	.594 (.085)	**.735** (.021)	.676 (.077)	.690 (.005)	.549 (.030)
Imp(ANN)	.821 (.019)	.819 (.027)	.678 (.023)	.596 (.089)	.718 (.026)	.655 (.039)	.692 (.005)	.557 (.006)
Imp(ANN) w/ disc.	.823 (.019)	.774 (.066)	.667 (.027)	.596 (.086)	**.735** (.022)	.679 (.072)	.693 (.006)	.559 (.019)
Imp(RF)	.794 (.022)	.658 (.116)	.636 (.029)	.560 (.068)	**.735** (.021)	.655 (.057)	.687 (.005)	.541 (.031)
Imp(RF) w/ disc.	.821 (.018)	.779 (.062)	.659 (.027)	.593 (.085)	.732 (.022)	.666 (.078)	**.695** (.006)	.544 (.031)
Imp(κNN)	.817 (.021)	.753 (.086)	.653 (.029)	.579 (.079)	.712 (.025)	.636 (.047)	.662 (.029)	.552 (.033)
Imp(κNN) w/ disc.	.812 (.028)	.755 (.098)	.649 (.036)	.581 (.079)	.721 (.031)	.656 (.084)	.650 (.072)	.539 (.057)
R-F(LR)	**.827** (.019)	.543 (.073)	.651 (.026)	.507 (.026)	.696 (.027)	.506 (.024)	.674 (.008)	.528 (.039)
R-F(LR) w/ disc.	.823 (.020)	.535 (.069)	.650 (.028)	.511 (.032)	.730 (.022)	.515 (.036)	.689 (.005)	.528 (.040)
R-F(ANN)	.825 (.020)	.544 (.072)	.649 (.028)	.507 (.026)	.696 (.027)	.505 (.021)	.680 (.006)	.520 (.035)
R-F(ANN) w/ disc.	.813 (.022)	.542 (.075)	.642 (.041)	.513 (.033)	.727 (.024)	.515 (.031)	.689 (.006)	.523 (.038)
R-F(RF)	.793 (.021)	.537 (.060)	.624 (.028)	.506 (.024)	.719 (.020)	.510 (.027)	.687 (.005)	.527 (.034)
R-F(RF) w/ disc.	.821 (.019)	.543 (.074)	.642 (.030)	.513 (.031)	.724 (.025)	.516 (.032)	.693 (.006)	.523 (.033)
R-F(κNN)	.810 (.021)	.528 (.060)	.624 (.030)	.505 (.026)	.683 (.025)	.505 (.024)	.661 (.030)	.523 (.035)
R-F(κNN) w/ disc.	.808 (.024)	.526 (.060)	.631 (.034)	.511 (.032)	.710 (.030)	.513 (.035)	.659 (.047)	.520 (.033)

Table 8: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, pre-selected 4 features.

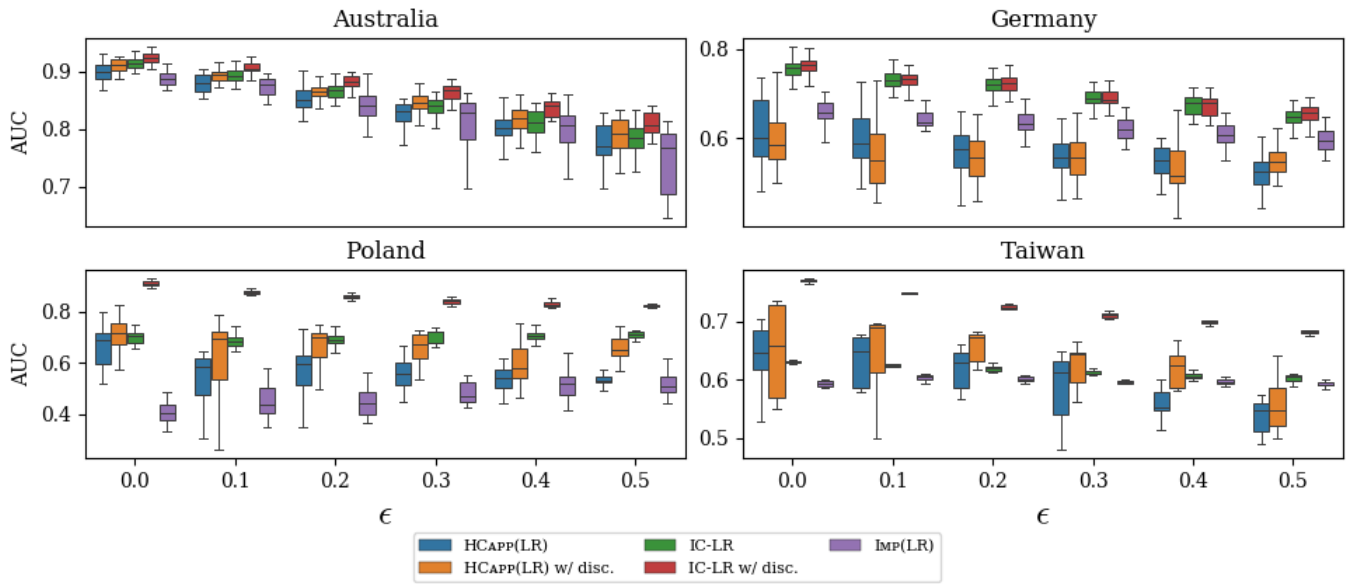| Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.792 (.022)	.792 (.022)	.639 (.029)	.639 (.029)	.660 (.029)	.660 (.029)	.648 (.006)	.648 (.006)
HC(LR) w/ disc.	.795 (.023)	.795 (.023)	.641 (.027)	.641 (.027)	.692 (.025)	.692 (.025)	.650 (.005)	.650 (.005)
HCApp(LR)	.794 (.021)	.794 (.021)	.652 (.027)	.652 (.027)	.670 (.027)	.670 (.027)	.646 (.005)	.646 (.005)
HCApp(LR) w/ disc.	.795 (.024)	.795 (.024)	.649 (.027)	.649 (.027)	.694 (.024)	.694 (.024)	.648 (.005)	.648 (.005)
HC(ANN)	.775 (.028)	.775 (.028)	.624 (.029)	.624 (.029)	.640 (.035)	.640 (.035)	.647 (.006)	.647 (.006)
HC(ANN) w/ disc.	.770 (.032)	.770 (.032)	.614 (.036)	.614 (.036)	.680 (.027)	.680 (.027)	.648 (.006)	.648 (.006)
Mincut	.770 (.031)	.770 (.031)	.580 (.031)	.580 (.031)	.501 (.016)	.501 (.016)	.651 (.005)	.651 (.005)
Mincut w/ disc.	.790 (.024)	.790 (.024)	.630 (.030)	.630 (.030)	.692 (.025)	.692 (.025)	.649 (.006)	.649 (.006)
IC-LR	.788 (.019)	.788 (.019)	.653 (.028)	.653 (.028)	.639 (.039)	.639 (.039)	.499 (.004)	.499 (.004)
IC-LR w/ disc.	.800 (.021)	**.800** (.021)	.651 (.029)	.651 (.029)	.698 (.025)	**.698** (.025)	.646 (.006)	.646 (.006)
IC-LR w/ neg.	.793 (.024)	.793 (.024)	.661 (.027)	**.661** (.027)	.652 (.035)	.652 (.035)	.671 (.009)	**.671** (.009)
Maj	.664 (.025)	.674 (.099)	.536 (.028)	.526 (.043)	.501 (.017)	.501 (.018)	.678 (.005)	.560 (.034)
Maj w/ disc.	.795 (.023)	.625 (.094)	.603 (.031)	.532 (.049)	.714 (.024)	.564 (.052)	.688 (.005)	.566 (.034)
Imp(LR)	.797 (.021)	.791 (.034)	.663 (.026)	.582 (.082)	.714 (.027)	.660 (.036)	.670 (.008)	.618 (.047)
Imp(LR) w/ disc.	.800 (.021)	.763 (.066)	.652 (.025)	.580 (.083)	.720 (.024)	.630 (.088)	.686 (.006)	.536 (.026)
Imp(ANN)	.799 (.021)	.789 (.037)	**.665** (.025)	.583 (.083)	.705 (.031)	.644 (.043)	.686 (.005)	.536 (.005)
Imp(ANN) w/ disc.	.800 (.021)	.747 (.085)	.652 (.029)	.580 (.084)	.719 (.024)	.636 (.084)	.688 (.005)	.543 (.024)
Imp(RF)	.772 (.023)	.630 (.101)	.623 (.031)	.560 (.065)	**.721** (.021)	.601 (.065)	.679 (.005)	.549 (.031)
Imp(RF) w/ disc.	.797 (.023)	.745 (.088)	.643 (.028)	.578 (.080)	.716 (.025)	.622 (.086)	**.689** (.005)	.542 (.027)
Imp(κNN)	.789 (.023)	.722 (.085)	.636 (.029)	.570 (.070)	.695 (.026)	.623 (.047)	.659 (.021)	.559 (.036)
Imp(κNN) w/ disc.	.786 (.028)	.705 (.119)	.628 (.041)	.574 (.072)	.696 (.047)	.608 (.087)	.656 (.044)	.563 (.050)
R-F(LR)	**.808** (.020)	.545 (.063)	.630 (.028)	.508 (.026)	.670 (.030)	.511 (.026)	.665 (.009)	.590 (.038)
R-F(LR) w/ disc.	.797 (.021)	.541 (.067)	.633 (.027)	.516 (.037)	.709 (.022)	.522 (.041)	.684 (.005)	.588 (.029)
R-F(ANN)	.805 (.022)	.542 (.061)	.628 (.028)	.509 (.026)	.664 (.037)	.509 (.026)	.668 (.008)	.548 (.047)
R-F(ANN) w/ disc.	.793 (.021)	.542 (.062)	.630 (.031)	.514 (.035)	.708 (.025)	.520 (.039)	.683 (.006)	.542 (.041)
R-F(RF)	.777 (.024)	.537 (.050)	.608 (.031)	.508 (.025)	.699 (.022)	.519 (.029)	.679 (.005)	.575 (.027)
R-F(RF) w/ disc.	.799 (.022)	.545 (.063)	.626 (.028)	.515 (.036)	.707 (.024)	.525 (.043)	.687 (.005)	.586 (.026)
R-F(κNN)	.780 (.024)	.535 (.055)	.602 (.031)	.505 (.025)	.658 (.027)	.512 (.027)	.659 (.018)	.585 (.029)
R-F(κNN) w/ disc.	.775 (.024)	.537 (.063)	.617 (.032)	.513 (.036)	.685 (.033)	.519 (.036)	.650 (.037)	.587 (.027)

Table 9: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.771 (.023)	.771 (.023)	.625 (.029)	.625 (.029)	.642 (.029)	.642 (.029)	.636 (.006)	.636 (.006)
HC(LR) w/ disc.	.769 (.024)	.769 (.024)	.629 (.030)	.629 (.030)	.670 (.027)	.670 (.027)	.639 (.006)	.639 (.006)
HCAPP(LR)	.773 (.019)	.773 (.019)	.634 (.028)	.634 (.028)	.645 (.027)	.645 (.027)	.636 (.006)	.636 (.006)
HCAPP(LR) w/ disc.	.772 (.021)	.772 (.021)	.631 (.031)	.631 (.031)	.672 (.027)	.672 (.027)	.638 (.006)	.638 (.006)
HC(ANN)	.753 (.028)	.753 (.028)	.610 (.030)	.610 (.030)	.621 (.034)	.621 (.034)	.636 (.005)	.636 (.005)
HC(ANN) w/ disc.	.748 (.031)	.748 (.031)	.598 (.035)	.598 (.035)	.659 (.028)	.659 (.028)	.638 (.006)	.638 (.006)
MINCUT	.752 (.028)	.752 (.028)	.580 (.032)	.580 (.032)	.500 (.018)	.500 (.018)	.638 (.006)	.638 (.006)
MINCUT w/ disc.	.770 (.024)	.770 (.024)	.620 (.031)	.620 (.031)	.670 (.027)	.670 (.027)	.638 (.006)	.638 (.006)
IC-LR	.759 (.023)	.759 (.023)	.636 (.035)	.636 (.035)	.624 (.041)	.624 (.041)	.497 (.004)	.497 (.004)
IC-LR w/ disc.	.775 (.021)	**.775** (.021)	.638 (.031)	.638 (.031)	.673 (.030)	**.673** (.030)	.634 (.007)	.634 (.007)
IC-LR w/ neg.	.771 (.022)	.771 (.022)	.640 (.029)	**.640** (.029)	.645 (.041)	.645 (.041)	.660 (.013)	**.660** (.013)
MAJ	.650 (.027)	.653 (.099)	.532 (.027)	.524 (.035)	.500 (.019)	.498 (.019)	.668 (.006)	.559 (.041)
MAJ w/ disc.	.775 (.026)	.615 (.094)	.592 (.033)	.534 (.047)	.701 (.024)	.559 (.057)	.680 (.005)	.561 (.041)
IMP(LR)	.774 (.023)	.758 (.048)	.651 (.027)	.580 (.076)	.701 (.026)	.654 (.038)	.659 (.011)	.603 (.054)
IMP(LR) w/ disc.	.770 (.025)	.721 (.087)	.636 (.030)	.587 (.074)	**.707** (.027)	.604 (.087)	.681 (.005)	.519 (.021)
IMP(ANN)	.775 (.023)	.752 (.050)	**.652** (.027)	.582 (.077)	.692 (.032)	.640 (.048)	.677 (.005)	.521 (.005)
IMP(ANN) w/ disc.	.770 (.023)	.708 (.098)	.635 (.034)	.586 (.076)	.705 (.027)	.612 (.086)	**.682** (.005)	.527 (.023)
IMP(RF)	.749 (.025)	.625 (.098)	.609 (.031)	.556 (.063)	.706 (.022)	.567 (.060)	.669 (.006)	.554 (.039)
IMP(RF) w/ disc.	.771 (.024)	.709 (.097)	.628 (.032)	.582 (.073)	.705 (.025)	.599 (.089)	.681 (.005)	.538 (.038)
IMP(κNN)	.766 (.025)	.663 (.106)	.623 (.032)	.565 (.066)	.681 (.025)	.614 (.052)	.650 (.015)	.559 (.042)
IMP(κNN) w/ disc.	.760 (.032)	.666 (.119)	.608 (.041)	.567 (.065)	.685 (.046)	.587 (.081)	.640 (.039)	.560 (.049)
R-F(LR)	**.788** (.023)	.584 (.072)	.608 (.028)	.512 (.029)	.651 (.027)	.524 (.031)	.654 (.010)	.613 (.017)
R-F(LR) w/ disc.	.777 (.028)	.577 (.077)	.615 (.030)	.520 (.038)	.697 (.024)	.550 (.051)	.678 (.005)	.601 (.010)
R-F(ANN)	.784 (.024)	.578 (.070)	.606 (.029)	.512 (.028)	.646 (.033)	.515 (.026)	.656 (.009)	.570 (.044)
R-F(ANN) w/ disc.	.776 (.027)	.595 (.081)	.611 (.032)	.516 (.036)	.695 (.025)	.547 (.051)	.676 (.006)	.558 (.044)
R-F(RF)	.760 (.024)	.580 (.059)	.592 (.028)	.512 (.028)	.683 (.020)	.543 (.035)	.669 (.006)	.592 (.011)
R-F(RF) w/ disc.	.783 (.028)	.610 (.083)	.609 (.031)	.522 (.038)	.701 (.024)	.558 (.052)	.680 (.005)	.599 (.010)
R-F(κNN)	.768 (.026)	.575 (.069)	.582 (.031)	.509 (.029)	.647 (.025)	.532 (.032)	.652 (.013)	.598 (.020)
R-F(κNN) w/ disc.	.759 (.032)	.586 (.079)	.598 (.032)	.519 (.039)	.681 (.027)	.547 (.048)	.653 (.019)	.600 (.017)

Table 10: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.751 (.024)	.751 (.024)	.610 (.030)	.610 (.030)	.626 (.030)	.626 (.030)	.628 (.006)	.628 (.006)
HC(LR) w/ disc.	.748 (.025)	.748 (.025)	.617 (.030)	.617 (.030)	.650 (.026)	.650 (.026)	.628 (.006)	.628 (.006)
HCApp(LR)	.752 (.023)	.752 (.023)	.618 (.029)	.618 (.029)	.629 (.029)	.629 (.029)	.627 (.006)	.627 (.006)
HCApp(LR) w/ disc.	.746 (.023)	.746 (.023)	.620 (.032)	.620 (.032)	.657 (.024)	**.657** (.024)	.628 (.006)	.628 (.006)
HC(ANN)	.731 (.028)	.731 (.028)	.593 (.031)	.593 (.031)	.601 (.035)	.601 (.035)	.628 (.007)	.628 (.007)
HC(ANN) w/ disc.	.726 (.030)	.726 (.030)	.586 (.037)	.586 (.037)	.640 (.029)	.640 (.029)	.627 (.006)	.627 (.006)
Mincut	.727 (.028)	.727 (.028)	.575 (.033)	.575 (.033)	.499 (.017)	.499 (.017)	.626 (.007)	.626 (.007)
Mincut w/ disc.	.746 (.027)	.746 (.027)	.610 (.033)	.610 (.033)	.648 (.028)	.648 (.028)	.626 (.007)	.626 (.007)
IC-LR	.731 (.027)	.731 (.027)	.623 (.035)	**.623** (.035)	.614 (.037)	.614 (.037)	.497 (.003)	.497 (.003)
IC-LR w/ disc.	.747 (.022)	.747 (.022)	.621 (.036)	.621 (.036)	.651 (.027)	.651 (.027)	.617 (.006)	.617 (.006)
IC-LR w/ neg.	.753 (.026)	**.753** (.026)	.620 (.040)	.620 (.040)	.632 (.046)	.632 (.046)	.649 (.013)	**.649** (.013)
Maj	.651 (.027)	.634 (.104)	.532 (.028)	.524 (.037)	.499 (.018)	.500 (.020)	.659 (.006)	.555 (.045)
Maj w/ disc.	.760 (.026)	.612 (.097)	.582 (.033)	.534 (.049)	**.689** (.025)	.563 (.060)	.671 (.005)	.557 (.046)
Imp(LR)	.757 (.024)	.733 (.052)	**.635** (.028)	.579 (.071)	.688 (.025)	.645 (.036)	.648 (.012)	.602 (.049)
Imp(LR) w/ disc.	.741 (.025)	.690 (.094)	.622 (.031)	.577 (.070)	.679 (.029)	.576 (.079)	.671 (.006)	.512 (.021)
Imp(ANN)	.759 (.025)	.728 (.055)	**.635** (.028)	.581 (.073)	.676 (.036)	.625 (.052)	.668 (.005)	.512 (.004)
Imp(ANN) w/ disc.	.742 (.026)	.679 (.103)	.623 (.033)	.578 (.070)	.677 (.031)	.584 (.078)	**.672** (.005)	.518 (.026)
Imp(RF)	.729 (.027)	.636 (.095)	.603 (.032)	.561 (.061)	.685 (.021)	.560 (.055)	.660 (.006)	.553 (.045)
Imp(RF) w/ disc.	.742 (.024)	.675 (.105)	.614 (.032)	.574 (.069)	.678 (.028)	.573 (.080)	.671 (.006)	.518 (.026)
Imp(κNN)	.744 (.026)	.633 (.108)	.606 (.033)	.559 (.061)	.662 (.025)	.578 (.062)	.641 (.012)	.555 (.045)
Imp(κNN) w/ disc.	.726 (.034)	.637 (.115)	.591 (.042)	.561 (.060)	.657 (.048)	.572 (.076)	.641 (.026)	.550 (.048)
R-F(LR)	**.771** (.025)	.631 (.069)	.590 (.032)	.523 (.032)	.627 (.034)	.540 (.037)	.642 (.010)	.615 (.011)
R-F(LR) w/ disc.	.761 (.026)	.645 (.077)	.599 (.032)	.527 (.042)	.683 (.025)	.580 (.055)	.669 (.006)	.602 (.006)
R-F(ANN)	.766 (.025)	.619 (.074)	.587 (.031)	.522 (.030)	.619 (.038)	.523 (.033)	.643 (.010)	.603 (.027)
R-F(ANN) w/ disc.	.758 (.030)	.637 (.075)	.587 (.033)	.520 (.037)	.677 (.028)	.561 (.055)	.668 (.007)	.596 (.028)
R-F(RF)	.747 (.026)	.625 (.056)	.582 (.031)	.523 (.030)	.661 (.023)	.560 (.031)	.660 (.006)	.598 (.007)
R-F(RF) w/ disc.	.767 (.025)	.670 (.070)	.593 (.032)	.527 (.042)	.685 (.024)	.581 (.054)	.670 (.006)	.602 (.006)
R-F(κNN)	.754 (.027)	.628 (.063)	.570 (.034)	.518 (.032)	.629 (.028)	.559 (.035)	.640 (.012)	.601 (.016)
R-F(κNN) w/ disc.	.747 (.029)	.646 (.075)	.585 (.033)	.526 (.038)	.666 (.028)	.577 (.055)	.637 (.020)	.601 (.016)

Table 11: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.726 (.025)	.726 (.025)	.596 (.029)	.596 (.029)	.612 (.030)	.612 (.030)	.616 (.006)	.616 (.006)
HC(LR) w/ disc.	.719 (.025)	.719 (.025)	.601 (.030)	.601 (.030)	.638 (.028)	**.638** (.028)	.616 (.005)	.616 (.005)
HCAPP(LR)	.727 (.026)	**.727** (.026)	.600 (.028)	.600 (.028)	.617 (.029)	.617 (.029)	.616 (.007)	.616 (.007)
HCAPP(LR) w/ disc.	.720 (.027)	.720 (.027)	.603 (.032)	**.603** (.032)	.636 (.026)	.636 (.026)	.617 (.005)	.617 (.005)
HC(ANN)	.709 (.028)	.709 (.028)	.578 (.032)	.578 (.032)	.588 (.036)	.588 (.036)	.615 (.006)	.615 (.006)
HC(ANN) w/ disc.	.701 (.032)	.701 (.032)	.570 (.036)	.570 (.036)	.626 (.030)	.626 (.030)	.615 (.006)	.615 (.006)
MINCUT	.706 (.028)	.706 (.028)	.572 (.033)	.572 (.033)	.500 (.018)	.500 (.018)	.613 (.008)	.613 (.008)
MINCUT w/ disc.	.719 (.024)	.719 (.024)	.596 (.031)	.596 (.031)	.632 (.029)	.632 (.029)	.613 (.008)	.613 (.008)
IC-LR	.708 (.031)	.708 (.031)	.603 (.038)	**.603** (.038)	.604 (.039)	.604 (.039)	.497 (.003)	.497 (.003)
IC-LR w/ disc.	.717 (.028)	.717 (.028)	.603 (.038)	**.603** (.038)	.625 (.028)	.625 (.028)	.606 (.008)	.606 (.008)
IC-LR w/ neg.	.716 (.028)	.716 (.028)	.600 (.034)	.600 (.034)	.619 (.045)	.619 (.045)	.640 (.014)	**.640** (.014)
MAJ	.649 (.028)	.618 (.106)	.534 (.029)	.526 (.040)	.500 (.018)	.497 (.019)	.646 (.006)	.551 (.049)
MAJ w/ disc.	.734 (.027)	.607 (.102)	.571 (.030)	.534 (.047)	.667 (.025)	.567 (.068)	.657 (.005)	.552 (.050)
IMP(LR)	.733 (.025)	.705 (.051)	.616 (.029)	.571 (.064)	**.670** (.026)	.631 (.037)	.636 (.014)	.581 (.054)
IMP(LR) w/ disc.	.707 (.030)	.643 (.099)	.603 (.033)	.564 (.065)	.647 (.036)	.560 (.069)	.658 (.005)	.505 (.015)
IMP(ANN)	.734 (.025)	.699 (.062)	**.617** (.029)	.571 (.065)	.660 (.035)	.606 (.060)	.655 (.005)	.505 (.004)
IMP(ANN) w/ disc.	.708 (.030)	.633 (.104)	.600 (.040)	.563 (.066)	.645 (.036)	.563 (.070)	**.659** (.005)	.511 (.026)
IMP(RF)	.705 (.026)	.635 (.090)	.588 (.032)	.555 (.055)	.660 (.023)	.551 (.055)	.647 (.006)	.549 (.049)
IMP(RF) w/ disc.	.707 (.030)	.633 (.105)	.595 (.034)	.562 (.063)	.648 (.034)	.558 (.070)	.658 (.005)	.509 (.022)
IMP(κNN)	.719 (.028)	.626 (.102)	.588 (.032)	.552 (.054)	.646 (.026)	.568 (.061)	.626 (.013)	.546 (.047)
IMP(κNN) w/ disc.	.692 (.036)	.609 (.103)	.573 (.043)	.544 (.056)	.624 (.050)	.547 (.066)	.622 (.027)	.548 (.050)
R-F(LR)	**.744** (.026)	.660 (.050)	.572 (.030)	.532 (.034)	.613 (.033)	.559 (.035)	.627 (.010)	.611 (.008)
R-F(LR) w/ disc.	.737 (.028)	.668 (.069)	.584 (.032)	.525 (.041)	.665 (.025)	.587 (.057)	.657 (.005)	.604 (.005)
R-F(ANN)	.735 (.028)	.641 (.060)	.569 (.031)	.530 (.034)	.598 (.036)	.531 (.037)	.625 (.011)	.605 (.015)
R-F(ANN) w/ disc.	.730 (.029)	.650 (.071)	.569 (.032)	.515 (.035)	.656 (.028)	.566 (.054)	.654 (.009)	.595 (.021)
R-F(RF)	.722 (.026)	.650 (.041)	.566 (.030)	.533 (.035)	.636 (.025)	.569 (.028)	.647 (.006)	.600 (.006)
R-F(RF) w/ disc.	.743 (.025)	.683 (.062)	.583 (.031)	.524 (.041)	.666 (.025)	.584 (.056)	.657 (.005)	.604 (.005)
R-F(κNN)	.731 (.029)	.659 (.051)	.555 (.033)	.528 (.034)	.618 (.028)	.577 (.031)	.627 (.012)	.599 (.014)
R-F(κNN) w/ disc.	.722 (.029)	.668 (.064)	.572 (.033)	.526 (.040)	.651 (.028)	.585 (.048)	.626 (.015)	.599 (.014)

Table 12: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.906 (.013)	.906 (.013)	.739 (.026)	.739 (.026)	.807 (.022)	.807 (.022)	.696 (.007)	.696 (.007)
HC(LR) w/ disc.	**.911** (.013)	**.911** (.013)	.732 (.024)	.732 (.024)	.813 (.019)	.813 (.019)	.729 (.006)	.729 (.006)
HCApp(LR)	.910 (.013)	.910 (.013)	.732 (.027)	.732 (.027)	.795 (.023)	.795 (.023)	.649 (.014)	.649 (.014)
HCApp(LR) w/ disc.	**.911** (.019)	**.911** (.019)	.720 (.027)	.720 (.027)	.786 (.021)	.786 (.021)	.646 (.010)	.646 (.010)
HC(ANN)	.908 (.013)	.908 (.013)	.753 (.023)	.753 (.023)	.800 (.023)	.800 (.023)	.701 (.010)	.701 (.010)
HC(ANN) w/ disc.	.909 (.013)	.909 (.013)	.736 (.024)	.736 (.024)	.815 (.018)	.815 (.018)	.731 (.005)	.731 (.005)
IC-LR	.901 (.013)	.901 (.013)	.739 (.023)	.739 (.023)	.714 (.040)	.714 (.040)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	**.911** (.013)	**.911** (.013)	.739 (.022)	.739 (.022)	.818 (.019)	.818 (.019)	.731 (.004)	.731 (.004)
IC-LR w/ neg.	.904 (.012)	.904 (.012)	**.762** (.020)	**.762** (.020)	.728 (.044)	.728 (.044)	.704 (.005)	.704 (.005)
Imp(LR)	.906 (.013)	.905 (.014)	.756 (.022)	.717 (.042)	.806 (.022)	.776 (.028)	.704 (.006)	.667 (.009)
Imp(LR) w/ disc.	.910 (.013)	.897 (.016)	.739 (.024)	.688 (.053)	**.819** (.020)	.794 (.031)	.731 (.005)	.579 (.025)
Imp(ANN)	.907 (.013)	.905 (.014)	.757 (.022)	.717 (.041)	.801 (.022)	.771 (.024)	.732 (.009)	.683 (.013)
Imp(ANN) w/ disc.	.910 (.013)	.898 (.016)	.742 (.023)	.692 (.052)	.818 (.019)	.793 (.030)	.731 (.005)	.587 (.003)
Imp(RF)	.885 (.015)	.875 (.024)	.709 (.025)	.660 (.084)	**.819** (.017)	.810 (.018)	**.734** (.005)	.582 (.022)
Imp(RF) w/ disc.	**.911** (.014)	.899 (.017)	.712 (.025)	.667 (.063)	.811 (.020)	.789 (.031)	.731 (.005)	.527 (.075)
Imp(κNN)	.904 (.015)	.874 (.018)	.717 (.026)	.663 (.078)	.794 (.021)	.771 (.027)	.716 (.017)	.636 (.041)
Imp(κNN) w/ disc.	.901 (.019)	.885 (.023)	.715 (.029)	.651 (.075)	.800 (.026)	.778 (.038)	.712 (.024)	.520 (.082)
R-F(LR)	.906 (.013)	.906 (.013)	.756 (.022)	.756 (.022)	.806 (.022)	.806 (.022)	.704 (.006)	.704 (.006)
R-F(LR) w/ disc.	.910 (.013)	.910 (.013)	.739 (.024)	.739 (.024)	**.819** (.020)	**.819** (.020)	.731 (.005)	.731 (.005)
R-F(ANN)	.907 (.013)	.907 (.013)	.757 (.022)	.757 (.022)	.801 (.023)	.801 (.022)	.731 (.009)	.732 (.009)
R-F(ANN) w/ disc.	.910 (.013)	.910 (.013)	.742 (.023)	.742 (.023)	.818 (.019)	.818 (.019)	.731 (.005)	.731 (.005)
R-F(RF)	.885 (.015)	.885 (.015)	.709 (.025)	.709 (.025)	**.819** (.017)	**.819** (.017)	**.734** (.005)	**.734** (.005)
R-F(RF) w/ disc.	.910 (.014)	**.911** (.014)	.712 (.026)	.712 (.026)	.811 (.020)	.811 (.021)	.731 (.005)	.731 (.005)
R-F(κNN)	.904 (.015)	.904 (.015)	.717 (.026)	.717 (.026)	.794 (.021)	.794 (.021)	.716 (.017)	.715 (.017)
R-F(κNN) w/ disc.	.902 (.017)	.901 (.019)	.714 (.031)	.714 (.030)	.802 (.024)	.802 (.024)	.712 (.024)	.713 (.025)

Table 13: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.889 (.018)	.889 (.018)	.704 (.028)	.704 (.028)	.751 (.029)	.751 (.029)	.674 (.006)	.674 (.006)
HC(LR) w/ disc.	.888 (.018)	.888 (.018)	.702 (.027)	.702 (.027)	.774 (.022)	.774 (.022)	.704 (.006)	.704 (.006)
HCApp(LR)	.893 (.016)	**.893** (.016)	.713 (.030)	.713 (.030)	.753 (.028)	.753 (.028)	.674 (.007)	.674 (.007)
HCApp(LR) w/ disc.	.891 (.017)	.891 (.017)	.703 (.028)	.703 (.028)	.771 (.023)	.771 (.023)	.703 (.006)	.703 (.006)
HC(ANN)	.886 (.018)	.886 (.018)	.707 (.028)	.707 (.028)	.730 (.032)	.730 (.032)	.682 (.008)	.682 (.008)
HC(ANN) w/ disc.	.876 (.019)	.876 (.019)	.681 (.032)	.681 (.032)	.765 (.023)	.765 (.023)	.706 (.006)	.706 (.006)
IC-LR	.878 (.016)	.878 (.016)	.718 (.025)	.718 (.025)	.709 (.040)	.709 (.040)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.893 (.015)	**.893** (.015)	.720 (.025)	.720 (.025)	.790 (.020)	**.790** (.020)	.710 (.005)	**.710** (.005)
IC-LR w/ neg.	.891 (.014)	.891 (.014)	**.740** (.021)	**.740** (.021)	.725 (.040)	.725 (.040)	.691 (.007)	.691 (.007)
Imp(LR)	.893 (.015)	.888 (.017)	.738 (.025)	.701 (.042)	.792 (.025)	.760 (.029)	.694 (.007)	.661 (.007)
Imp(LR) w/ disc.	**.894** (.016)	.859 (.029)	.720 (.026)	.676 (.050)	.800 (.020)	.757 (.038)	.725 (.005)	.677 (.027)
Imp(ANN)	.893 (.015)	.888 (.016)	.739 (.025)	.702 (.041)	.786 (.026)	.753 (.026)	.721 (.008)	.678 (.011)
Imp(ANN) w/ disc.	.892 (.016)	.859 (.029)	.724 (.026)	.679 (.050)	.799 (.020)	.758 (.038)	**.726** (.005)	.677 (.026)
Imp(RF)	.866 (.018)	.850 (.031)	.688 (.028)	.657 (.064)	**.805** (.020)	.773 (.024)	.717 (.005)	.634 (.039)
Imp(RF) w/ disc.	**.894** (.016)	.857 (.044)	.695 (.029)	.660 (.057)	.791 (.021)	.741 (.051)	.725 (.005)	.662 (.044)
Imp(κNN)	.885 (.018)	.860 (.027)	.699 (.029)	.658 (.065)	.776 (.024)	.747 (.025)	.708 (.012)	.663 (.028)
Imp(κNN) w/ disc.	.884 (.021)	.845 (.042)	.691 (.033)	.638 (.070)	.779 (.026)	.722 (.063)	.700 (.027)	.596 (.083)
R-F(LR)	.880 (.018)	.798 (.067)	.697 (.028)	.609 (.055)	.758 (.029)	.591 (.099)	.693 (.006)	.546 (.101)
R-F(LR) w/ disc.	.884 (.019)	.789 (.070)	.701 (.028)	.617 (.061)	.790 (.021)	.654 (.100)	.723 (.006)	.570 (.121)
R-F(ANN)	.878 (.019)	.817 (.056)	.688 (.030)	.624 (.051)	.758 (.029)	.632 (.084)	.706 (.008)	.588 (.088)
R-F(ANN) w/ disc.	.876 (.021)	.815 (.043)	.687 (.036)	.626 (.059)	.786 (.021)	.695 (.061)	.724 (.006)	.633 (.076)
R-F(RF)	.860 (.019)	.799 (.056)	.674 (.029)	.611 (.055)	.788 (.021)	.664 (.085)	.717 (.005)	.592 (.072)
R-F(RF) w/ disc.	.880 (.019)	.803 (.057)	.681 (.029)	.629 (.055)	.777 (.023)	.681 (.068)	.722 (.005)	.574 (.094)
R-F(κNN)	.871 (.019)	.787 (.056)	.664 (.031)	.598 (.044)	.742 (.026)	.640 (.056)	.706 (.013)	.644 (.037)
R-F(κNN) w/ disc.	.869 (.022)	.772 (.070)	.676 (.033)	.610 (.047)	.767 (.027)	.671 (.064)	.701 (.022)	.636 (.057)

Table 14: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.875 (.018)	.875 (.018)	.678 (.031)	.678 (.031)	.727 (.028)	.727 (.028)	.662 (.007)	.662 (.007)
HC(LR) w/ disc.	.870 (.019)	.870 (.019)	.679 (.030)	.679 (.030)	.748 (.025)	.748 (.025)	.693 (.007)	.693 (.007)
HCApp(LR)	.876 (.018)	.876 (.018)	.691 (.029)	.691 (.029)	.731 (.027)	.731 (.027)	.661 (.007)	.661 (.007)
HCApp(LR) w/ disc.	.871 (.020)	.871 (.020)	.684 (.028)	.684 (.028)	.746 (.024)	.746 (.024)	.691 (.006)	.691 (.006)
HC(ANN)	.865 (.021)	.865 (.021)	.675 (.032)	.675 (.032)	.702 (.036)	.702 (.036)	.670 (.008)	.670 (.008)
HC(ANN) w/ disc.	.850 (.026)	.850 (.026)	.655 (.035)	.655 (.035)	.735 (.027)	.735 (.027)	.691 (.007)	.691 (.007)
IC-LR	.853 (.019)	.853 (.019)	.700 (.027)	.700 (.027)	.706 (.040)	.706 (.040)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.875 (.018)	.875 (.018)	.700 (.027)	.700 (.027)	.764 (.023)	**.764** (.023)	.694 (.006)	**.694** (.006)
IC-LR w/ neg.	**.884** (.011)	**.884** (.011)	**.726** (.024)	**.726** (.024)	.718 (.042)	.718 (.042)	.688 (.005)	.688 (.005)
Imp(LR)	.879 (.017)	.868 (.019)	.720 (.026)	.681 (.042)	.782 (.026)	.746 (.026)	.690 (.006)	.657 (.006)
Imp(LR) w/ disc.	.877 (.014)	.832 (.029)	.701 (.026)	.658 (.047)	.780 (.024)	.710 (.059)	**.721** (.005)	.682 (.011)
Imp(ANN)	.880 (.016)	.868 (.019)	.722 (.026)	.683 (.042)	.775 (.027)	.739 (.024)	.714 (.007)	.673 (.009)
Imp(ANN) w/ disc.	.875 (.014)	.832 (.029)	.705 (.027)	.659 (.048)	.778 (.024)	.711 (.059)	**.721** (.005)	.682 (.011)
Imp(RF)	.850 (.019)	.821 (.040)	.668 (.033)	.641 (.053)	**.789** (.022)	.733 (.031)	.709 (.006)	.638 (.033)
Imp(RF) w/ disc.	.876 (.016)	.831 (.031)	.679 (.028)	.645 (.051)	.772 (.024)	.701 (.061)	.720 (.005)	.671 (.024)
Imp(κNN)	.868 (.019)	.842 (.031)	.681 (.030)	.640 (.057)	.757 (.026)	.725 (.028)	.702 (.010)	.655 (.025)
Imp(κNN) w/ disc.	.861 (.024)	.815 (.036)	.669 (.038)	.628 (.060)	.754 (.033)	.673 (.076)	.698 (.018)	.618 (.057)
R-F(LR)	.866 (.019)	.778 (.066)	.668 (.032)	.599 (.051)	.726 (.032)	.589 (.091)	.687 (.007)	.629 (.031)
R-F(LR) w/ disc.	.864 (.018)	.746 (.078)	.678 (.028)	.613 (.058)	.766 (.023)	.633 (.089)	.717 (.005)	.666 (.031)
R-F(ANN)	.863 (.020)	.801 (.050)	.662 (.032)	.607 (.050)	.724 (.035)	.638 (.071)	.694 (.007)	.635 (.028)
R-F(ANN) w/ disc.	.858 (.018)	.793 (.048)	.669 (.030)	.621 (.053)	.761 (.024)	.676 (.048)	.716 (.005)	.666 (.023)
R-F(RF)	.846 (.021)	.775 (.057)	.650 (.034)	.593 (.055)	.762 (.023)	.647 (.070)	.708 (.006)	.645 (.020)
R-F(RF) w/ disc.	.858 (.019)	.785 (.049)	.661 (.029)	.627 (.057)	.756 (.024)	.661 (.065)	.716 (.005)	.658 (.033)
R-F(κNN)	.845 (.020)	.742 (.055)	.635 (.033)	.567 (.042)	.707 (.029)	.620 (.044)	.700 (.009)	.658 (.023)
R-F(κNN) w/ disc.	.839 (.022)	.724 (.068)	.655 (.033)	.593 (.052)	.739 (.031)	.647 (.053)	.695 (.018)	.659 (.023)

Table 15: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.855 (.020)	.855 (.020)	.658 (.032)	.658 (.032)	.704 (.027)	.704 (.027)	.653 (.006)	.653 (.006)
HC(LR) w/ disc.	.848 (.021)	.848 (.021)	.659 (.032)	.659 (.032)	.724 (.029)	.724 (.029)	.676 (.008)	.676 (.008)
HCApp(LR)	.854 (.018)	.854 (.018)	.667 (.031)	.667 (.031)	.702 (.030)	.702 (.030)	.653 (.006)	.653 (.006)
HCApp(LR) w/ disc.	.847 (.018)	.847 (.018)	.661 (.030)	.661 (.030)	.721 (.027)	.721 (.027)	.677 (.006)	.677 (.006)
HC(ANN)	.843 (.023)	.843 (.023)	.651 (.032)	.651 (.032)	.674 (.039)	.674 (.039)	.656 (.006)	.656 (.006)
HC(ANN) w/ disc.	.826 (.031)	.826 (.031)	.628 (.037)	.628 (.037)	.709 (.030)	.709 (.030)	.679 (.006)	.679 (.006)
IC-LR	.822 (.019)	.822 (.019)	.676 (.028)	.676 (.028)	.697 (.038)	.697 (.038)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.853 (.017)	.853 (.017)	.677 (.029)	.677 (.029)	.737 (.027)	**.737** (.027)	.681 (.006)	.681 (.006)
IC-LR w/ neg.	**.865** (.017)	**.865** (.017)	.700 (.029)	**.700** (.029)	.716 (.041)	.716 (.041)	.685 (.005)	**.685** (.005)
Imp(LR)	.862 (.019)	.842 (.027)	.703 (.029)	.667 (.041)	.769 (.025)	.728 (.026)	.685 (.005)	.649 (.005)
Imp(LR) w/ disc.	.855 (.017)	.792 (.049)	.682 (.030)	.646 (.045)	.764 (.026)	.681 (.055)	**.716** (.006)	.674 (.009)
Imp(ANN)	.862 (.019)	.842 (.026)	**.706** (.028)	.668 (.040)	.761 (.027)	.721 (.024)	.705 (.006)	.663 (.009)
Imp(ANN) w/ disc.	.853 (.017)	.792 (.048)	.684 (.031)	.647 (.045)	.761 (.025)	.681 (.055)	**.716** (.006)	.675 (.008)
Imp(RF)	.829 (.021)	.794 (.040)	.650 (.033)	.629 (.049)	**.770** (.022)	.695 (.044)	.700 (.006)	.652 (.012)
Imp(RF) w/ disc.	.854 (.018)	.791 (.048)	.662 (.032)	.635 (.048)	.759 (.024)	.678 (.054)	.714 (.006)	.671 (.011)
Imp(κNN)	.848 (.023)	.806 (.047)	.663 (.033)	.623 (.057)	.740 (.026)	.701 (.031)	.692 (.008)	.641 (.026)
Imp(κNN) w/ disc.	.838 (.027)	.769 (.067)	.646 (.040)	.615 (.056)	.735 (.041)	.649 (.067)	.685 (.017)	.616 (.044)
R-F(LR)	.848 (.020)	.775 (.068)	.643 (.030)	.595 (.047)	.704 (.030)	.611 (.072)	.679 (.005)	.636 (.013)
R-F(LR) w/ disc.	.843 (.023)	.751 (.074)	.658 (.032)	.609 (.051)	.752 (.023)	.653 (.075)	.709 (.006)	.672 (.012)
R-F(ANN)	.844 (.021)	.790 (.050)	.640 (.032)	.599 (.046)	.705 (.035)	.636 (.053)	.681 (.006)	.634 (.022)
R-F(ANN) w/ disc.	.839 (.024)	.785 (.045)	.650 (.033)	.611 (.046)	.748 (.024)	.673 (.043)	.709 (.006)	.668 (.011)
R-F(RF)	.828 (.021)	.757 (.063)	.629 (.030)	.588 (.053)	.738 (.023)	.648 (.060)	.697 (.006)	.652 (.012)
R-F(RF) w/ disc.	.842 (.023)	.779 (.048)	.647 (.032)	.621 (.046)	.745 (.023)	.665 (.057)	.709 (.006)	.671 (.012)
R-F(κNN)	.825 (.023)	.740 (.049)	.610 (.032)	.562 (.043)	.691 (.026)	.627 (.043)	.690 (.008)	.649 (.016)
R-F(κNN) w/ disc.	.819 (.026)	.729 (.057)	.632 (.034)	.594 (.050)	.726 (.027)	.650 (.045)	.689 (.009)	.651 (.017)

Table 16: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.828 (.023)	.828 (.023)	.639 (.034)	.639 (.034)	.681 (.030)	.681 (.030)	.644 (.007)	.644 (.007)
HC(LR) w/ disc.	.820 (.024)	.820 (.024)	.641 (.032)	.641 (.032)	.695 (.027)	.695 (.027)	.663 (.006)	.663 (.006)
HCApp(LR)	.827 (.021)	.827 (.021)	.645 (.033)	.645 (.033)	.682 (.030)	.682 (.030)	.644 (.006)	.644 (.006)
HCApp(LR) w/ disc.	.817 (.022)	.817 (.022)	.642 (.032)	.642 (.032)	.699 (.026)	.699 (.026)	.663 (.006)	.663 (.006)
HC(ANN)	.810 (.028)	.810 (.028)	.626 (.037)	.626 (.037)	.646 (.040)	.646 (.040)	.647 (.007)	.647 (.007)
HC(ANN) w/ disc.	.794 (.034)	.794 (.034)	.604 (.037)	.604 (.037)	.683 (.028)	.683 (.028)	.664 (.007)	.664 (.007)
IC-LR	.788 (.021)	.788 (.021)	.654 (.029)	.654 (.029)	.690 (.033)	.690 (.033)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.825 (.020)	.825 (.020)	.656 (.028)	.656 (.028)	.712 (.025)	**.712** (.025)	.668 (.006)	.668 (.006)
IC-LR w/ neg.	**.848** (.017)	**.848** (.017)	**.689** (.030)	**.689** (.030)	.700 (.039)	.700 (.039)	.678 (.007)	**.678** (.007)
Imp(LR)	.844 (.019)	.814 (.028)	.686 (.029)	.652 (.040)	**.753** (.027)	.709 (.026)	.679 (.005)	.641 (.006)
Imp(LR) w/ disc.	.829 (.020)	.762 (.038)	.665 (.032)	.629 (.042)	.736 (.028)	.644 (.058)	**.707** (.006)	.660 (.009)
Imp(ANN)	.845 (.019)	.814 (.028)	.688 (.029)	.653 (.040)	.745 (.030)	.704 (.026)	.696 (.007)	.652 (.009)
Imp(ANN) w/ disc.	.828 (.020)	.762 (.038)	.666 (.032)	.629 (.042)	.733 (.028)	.644 (.057)	**.707** (.006)	.660 (.009)
Imp(RF)	.809 (.024)	.768 (.040)	.640 (.033)	.619 (.044)	.746 (.022)	.679 (.028)	.692 (.007)	.650 (.009)
Imp(RF) w/ disc.	.829 (.020)	.762 (.038)	.646 (.034)	.619 (.043)	.731 (.028)	.642 (.057)	.705 (.006)	.657 (.011)
Imp(κNN)	.826 (.022)	.775 (.045)	.642 (.035)	.609 (.050)	.718 (.027)	.675 (.033)	.682 (.008)	.625 (.026)
Imp(κNN) w/ disc.	.812 (.033)	.738 (.059)	.623 (.044)	.598 (.053)	.717 (.033)	.622 (.065)	.678 (.015)	.607 (.045)
R-F(LR)	.826 (.023)	.782 (.051)	.622 (.034)	.595 (.041)	.676 (.038)	.623 (.053)	.667 (.006)	.631 (.009)
R-F(LR) w/ disc.	.821 (.024)	.768 (.052)	.638 (.036)	.611 (.038)	.731 (.026)	.669 (.048)	.694 (.007)	.662 (.008)
R-F(ANN)	.821 (.024)	.781 (.040)	.618 (.034)	.594 (.039)	.675 (.040)	.623 (.047)	.667 (.007)	.633 (.012)
R-F(ANN) w/ disc.	.818 (.024)	.779 (.035)	.630 (.037)	.602 (.039)	.728 (.026)	.668 (.040)	.693 (.007)	.661 (.009)
R-F(RF)	.805 (.025)	.753 (.049)	.612 (.034)	.592 (.044)	.705 (.026)	.646 (.040)	.684 (.007)	.648 (.009)
R-F(RF) w/ disc.	.821 (.023)	.784 (.034)	.629 (.035)	.616 (.038)	.728 (.027)	.673 (.044)	.693 (.007)	.661 (.008)
R-F(κNN)	.805 (.024)	.746 (.043)	.591 (.037)	.570 (.041)	.666 (.030)	.633 (.036)	.676 (.008)	.639 (.015)
R-F(κNN) w/ disc.	.802 (.027)	.746 (.040)	.616 (.036)	.590 (.043)	.708 (.028)	.656 (.039)	.674 (.009)	.637 (.016)

Table 17: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, balanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.788 (.023)	.788 (.023)	.616 (.033)	.616 (.033)	.657 (.030)	.657 (.030)	.630 (.006)	.630 (.006)
HC(LR) w/ disc.	.779 (.024)	.779 (.024)	.618 (.031)	.618 (.031)	.670 (.029)	.670 (.029)	.645 (.006)	.645 (.006)
HCApp(LR)	.787 (.025)	.787 (.025)	.620 (.032)	.620 (.032)	.661 (.031)	.661 (.031)	.631 (.007)	.631 (.007)
HCApp(LR) w/ disc.	.777 (.027)	.777 (.027)	.620 (.031)	.620 (.031)	.670 (.027)	.670 (.027)	.646 (.006)	.646 (.006)
HC(ANN)	.773 (.029)	.773 (.029)	.600 (.035)	.600 (.035)	.624 (.041)	.624 (.041)	.632 (.006)	.632 (.006)
HC(ANN) w/ disc.	.761 (.031)	.761 (.031)	.581 (.036)	.581 (.036)	.661 (.032)	.661 (.032)	.646 (.006)	.646 (.006)
IC-LR	.747 (.023)	.747 (.023)	.629 (.029)	.629 (.029)	.686 (.032)	.686 (.032)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.790 (.024)	.790 (.024)	.631 (.029)	.631 (.029)	.686 (.027)	.686 (.027)	.653 (.006)	.653 (.006)
IC-LR w/ neg.	.816 (.022)	**.816** (.022)	.660 (.028)	**.660** (.028)	.708 (.035)	**.708** (.035)	.672 (.006)	**.672** (.006)
Imp(LR)	**.819** (.021)	.778 (.030)	.663 (.032)	.628 (.040)	**.736** (.027)	.688 (.026)	.672 (.006)	.630 (.006)
Imp(LR) w/ disc.	.791 (.021)	.710 (.052)	.641 (.032)	.607 (.039)	.698 (.036)	.611 (.052)	**.695** (.005)	.642 (.011)
Imp(ANN)	.818 (.021)	.778 (.030)	**.665** (.032)	.629 (.040)	.729 (.030)	.683 (.028)	.684 (.007)	.637 (.008)
Imp(ANN) w/ disc.	.790 (.021)	.710 (.052)	.639 (.035)	.605 (.041)	.694 (.036)	.610 (.053)	**.695** (.005)	.643 (.011)
Imp(RF)	.782 (.024)	.735 (.037)	.622 (.034)	.602 (.040)	.720 (.025)	.655 (.027)	.681 (.006)	.640 (.008)
Imp(RF) w/ disc.	.791 (.021)	.710 (.052)	.627 (.033)	.600 (.040)	.696 (.036)	.610 (.051)	.693 (.005)	.642 (.011)
Imp(κNN)	.796 (.026)	.741 (.045)	.620 (.036)	.592 (.047)	.703 (.029)	.650 (.043)	.668 (.010)	.607 (.028)
Imp(κNN) w/ disc.	.772 (.035)	.691 (.067)	.604 (.046)	.576 (.049)	.674 (.041)	.584 (.061)	.662 (.020)	.591 (.043)
R-F(LR)	.787 (.025)	.765 (.032)	.594 (.034)	.585 (.037)	.652 (.037)	.621 (.039)	.649 (.006)	.621 (.007)
R-F(LR) w/ disc.	.779 (.031)	.750 (.036)	.622 (.036)	.597 (.037)	.706 (.028)	.658 (.031)	.672 (.006)	.644 (.008)
R-F(ANN)	.781 (.025)	.754 (.034)	.590 (.034)	.580 (.036)	.646 (.040)	.607 (.043)	.647 (.006)	.622 (.011)
R-F(ANN) w/ disc.	.778 (.031)	.751 (.036)	.606 (.037)	.578 (.037)	.702 (.028)	.652 (.032)	.670 (.007)	.642 (.008)
R-F(RF)	.768 (.024)	.730 (.035)	.588 (.033)	.581 (.038)	.670 (.027)	.635 (.032)	.663 (.006)	.636 (.008)
R-F(RF) w/ disc.	.787 (.029)	.758 (.033)	.618 (.035)	.598 (.036)	.706 (.028)	.660 (.031)	.672 (.007)	.644 (.008)
R-F(κNN)	.770 (.026)	.737 (.035)	.570 (.036)	.567 (.042)	.646 (.031)	.631 (.031)	.656 (.008)	.624 (.012)
R-F(κNN) w/ disc.	.766 (.029)	.736 (.039)	.597 (.037)	.580 (.040)	.687 (.031)	.636 (.043)	.654 (.009)	.623 (.014)

Table 18: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.848 (.015)	.848 (.015)	.736 (.017)	.736 (.017)	.931 (.003)	.931 (.003)	.808 (.003)	.808 (.003)
HC(LR) w/ disc.	.848 (.013)	.848 (.013)	.730 (.015)	.730 (.015)	.931 (.004)	.931 (.004)	.819 (.002)	.819 (.002)
HCApp(LR)	.849 (.014)	.849 (.014)	.720 (.017)	.720 (.017)	.931 (.003)	.931 (.003)	.808 (.013)	.808 (.013)
HCApp(LR) w/ disc.	.834 (.036)	.834 (.036)	.718 (.015)	.718 (.015)	**.932** (.003)	**.932** (.003)	.819 (.002)	.819 (.002)
HC(ANN)	.850 (.013)	.850 (.013)	.740 (.018)	.740 (.018)	.931 (.003)	.931 (.003)	.819 (.002)	.819 (.002)
HC(ANN) w/ disc.	.848 (.014)	.848 (.014)	.730 (.017)	.730 (.017)	.931 (.003)	.931 (.003)	.819 (.002)	.819 (.002)
Mincut	.739 (.020)	.739 (.020)	.568 (.018)	.568 (.018)	.085 (.004)	.085 (.004)	.818 (.002)	.818 (.002)
Mincut w/ disc.	.845 (.014)	.845 (.014)	.696 (.019)	.696 (.019)	.929 (.004)	.929 (.004)	.819 (.002)	.819 (.002)
IC-LR	**.855** (.012)	**.855** (.012)	.725 (.015)	.725 (.015)	.931 (.003)	.931 (.003)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.848 (.012)	.848 (.012)	.727 (.015)	.727 (.015)	.931 (.003)	.931 (.003)	.819 (.002)	.819 (.002)
IC-LR w/ neg.	**.855** (.012)	**.855** (.012)	.740 (.015)	.740 (.015)	.931 (.003)	.931 (.003)	.807 (.003)	.807 (.003)
Maj	.738 (.020)	.738 (.020)	.553 (.019)	.553 (.019)	.085 (.004)	.085 (.004)	.818 (.002)	.818 (.002)
Maj w/ disc.	.847 (.014)	.847 (.014)	.686 (.020)	.686 (.020)	.928 (.004)	.928 (.004)	.819 (.002)	.819 (.002)
Imp(LR)	.853 (.013)	.822 (.097)	.744 (.017)	.700 (.014)	.930 (.003)	.931 (.003)	.808 (.003)	.779 (.002)
Imp(LR) w/ disc.	.850 (.013)	.798 (.102)	.731 (.015)	.700 (.014)	**.932** (.003)	.931 (.003)	**.820** (.002)	.779 (.002)
Imp(ANN)	.849 (.013)	.816 (.096)	.744 (.016)	.700 (.014)	.930 (.003)	.931 (.003)	.819 (.002)	.779 (.002)
Imp(ANN) w/ disc.	.848 (.014)	.776 (.096)	.729 (.016)	.700 (.014)	**.932** (.004)	.931 (.003)	.819 (.002)	.779 (.002)
Imp(RF)	.818 (.015)	.637 (.095)	.705 (.017)	.700 (.014)	.929 (.003)	.931 (.003)	.818 (.002)	.779 (.002)
Imp(RF) w/ disc.	.847 (.015)	.792 (.103)	.719 (.016)	.700 (.014)	.930 (.004)	.931 (.003)	.819 (.002)	.779 (.002)
Imp(κNN)	.845 (.016)	.747 (.111)	.727 (.018)	.700 (.014)	.931 (.003)	.931 (.003)	.811 (.007)	.774 (.036)
Imp(κNN) w/ disc.	.841 (.022)	.749 (.120)	.717 (.020)	.699 (.016)	.929 (.004)	.926 (.057)	.811 (.010)	.771 (.061)
R-F(LR)	.853 (.013)	.853 (.013)	.744 (.017)	**.744** (.017)	.930 (.003)	.930 (.003)	.808 (.003)	.808 (.003)
R-F(LR) w/ disc.	.850 (.013)	.850 (.013)	.731 (.015)	.731 (.015)	**.932** (.003)	**.932** (.003)	**.820** (.002)	**.820** (.002)
R-F(ANN)	.849 (.013)	.849 (.013)	**.745** (.016)	**.744** (.017)	.930 (.003)	.930 (.003)	.819 (.002)	.819 (.002)
R-F(ANN) w/ disc.	.848 (.014)	.848 (.014)	.729 (.015)	.730 (.015)	.931 (.004)	.931 (.003)	**.820** (.002)	.819 (.002)
R-F(RF)	.818 (.015)	.818 (.016)	.705 (.016)	.706 (.016)	.929 (.003)	.929 (.003)	.818 (.002)	.818 (.002)
R-F(RF) w/ disc.	.846 (.015)	.847 (.015)	.719 (.016)	.719 (.016)	.930 (.004)	.930 (.004)	.819 (.002)	.819 (.002)
R-F(κNN)	.845 (.016)	.845 (.017)	.727 (.018)	.728 (.018)	.931 (.003)	.931 (.003)	.811 (.007)	.810 (.007)
R-F(κNN) w/ disc.	.842 (.019)	.841 (.023)	.717 (.020)	.717 (.021)	.929 (.005)	.929 (.004)	.810 (.012)	.807 (.042)

Table 19: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.814 (.019)	.814 (.019)	.708 (.017)	.708 (.017)	.930 (.004)	.930 (.004)	.803 (.003)	.803 (.003)
HC(LR) w/ disc.	.818 (.020)	.818 (.020)	.701 (.018)	.701 (.018)	.929 (.004)	.929 (.004)	.804 (.003)	**.804** (.003)
HCApp(LR)	.807 (.014)	.807 (.014)	.710 (.013)	.710 (.013)	.930 (.003)	.930 (.003)	.795 (.010)	.795 (.010)
HCApp(LR) w/ disc.	.815 (.020)	.815 (.020)	.702 (.018)	.702 (.018)	.930 (.003)	.930 (.003)	.804 (.003)	**.804** (.003)
HC(ANN)	.799 (.022)	.799 (.022)	.700 (.020)	.700 (.020)	.930 (.004)	.930 (.004)	.803 (.003)	.803 (.003)
HC(ANN) w/ disc.	.797 (.025)	.797 (.025)	.693 (.017)	.693 (.017)	.930 (.004)	.930 (.004)	.802 (.003)	.802 (.003)
Mincut	.770 (.032)	.770 (.032)	.606 (.028)	.606 (.028)	.080 (.004)	.080 (.004)	.802 (.003)	.802 (.003)
Mincut w/ disc.	.811 (.021)	.811 (.021)	.686 (.021)	.686 (.021)	.928 (.004)	.928 (.004)	.804 (.002)	**.804** (.002)
IC-LR	.809 (.018)	.809 (.018)	.706 (.013)	.706 (.013)	**.931** (.003)	**.931** (.003)	.778 (.002)	.778 (.002)
IC-LR w/ disc.	.819 (.019)	**.819** (.019)	.710 (.014)	.710 (.014)	.930 (.003)	.930 (.003)	.804 (.002)	**.804** (.002)
IC-LR w/ neg.	.814 (.020)	.814 (.020)	.728 (.015)	**.728** (.015)	**.931** (.003)	**.931** (.003)	.804 (.003)	**.804** (.003)
Maj	.677 (.024)	.734 (.073)	.489 (.021)	.637 (.030)	.077 (.004)	.142 (.120)	.811 (.003)	.782 (.006)
Maj w/ disc.	.814 (.019)	.703 (.080)	.643 (.028)	.686 (.019)	.922 (.005)	.929 (.004)	.815 (.003)	.782 (.007)
Imp(LR)	.818 (.019)	.775 (.098)	**.735** (.016)	.700 (.014)	.930 (.004)	.930 (.004)	.805 (.003)	.779 (.002)
Imp(LR) w/ disc.	.825 (.018)	.744 (.105)	.723 (.016)	.699 (.015)	**.931** (.003)	.930 (.003)	.816 (.003)	.779 (.002)
Imp(ANN)	.821 (.018)	.768 (.099)	**.735** (.016)	.700 (.014)	.930 (.004)	.930 (.004)	.815 (.003)	.779 (.002)
Imp(ANN) w/ disc.	**.826** (.016)	.735 (.107)	.722 (.016)	.699 (.015)	.930 (.003)	.930 (.003)	**.817** (.003)	.779 (.002)
Imp(RF)	.794 (.020)	.638 (.094)	.696 (.019)	.700 (.014)	.928 (.004)	.930 (.004)	.813 (.003)	.779 (.003)
Imp(RF) w/ disc.	.822 (.019)	.732 (.108)	.708 (.018)	.699 (.015)	.929 (.004)	.930 (.003)	**.817** (.002)	.779 (.002)
Imp(κNN)	.813 (.019)	.703 (.100)	.719 (.019)	.700 (.014)	.930 (.004)	.930 (.004)	.808 (.007)	.779 (.002)
Imp(κNN) w/ disc.	.813 (.023)	.698 (.122)	.707 (.023)	.695 (.022)	.927 (.006)	.930 (.003)	.807 (.010)	.776 (.018)
R-F(LR)	.825 (.017)	.570 (.037)	.715 (.017)	.697 (.014)	.930 (.004)	.930 (.004)	.806 (.003)	.779 (.002)
R-F(LR) w/ disc.	.818 (.019)	.571 (.046)	.713 (.018)	.696 (.015)	.930 (.004)	.930 (.003)	**.817** (.002)	.779 (.002)
R-F(ANN)	.823 (.017)	.572 (.039)	.715 (.017)	.697 (.014)	.930 (.004)	.930 (.004)	.811 (.003)	.779 (.002)
R-F(ANN) w/ disc.	.817 (.018)	.576 (.053)	.711 (.018)	.697 (.015)	.930 (.003)	.930 (.003)	.814 (.002)	.779 (.002)
R-F(RF)	.795 (.019)	.570 (.037)	.687 (.019)	.697 (.014)	.927 (.004)	.930 (.004)	.813 (.003)	.779 (.003)
R-F(RF) w/ disc.	.817 (.018)	.576 (.052)	.698 (.019)	.696 (.015)	.927 (.004)	.930 (.003)	.816 (.002)	.779 (.003)
R-F(κNN)	.807 (.020)	.563 (.034)	.701 (.018)	.696 (.015)	.930 (.004)	.930 (.004)	.808 (.006)	.779 (.002)
R-F(κNN) w/ disc.	.805 (.024)	.565 (.035)	.699 (.021)	.696 (.015)	.928 (.004)	.930 (.003)	.809 (.006)	.779 (.003)

Table 20: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.796 (.019)	.796 (.019)	.694 (.019)	.694 (.019)	.929 (.003)	.929 (.003)	.800 (.002)	**.800** (.002)
HC(LR) w/ disc.	.794 (.020)	.794 (.020)	.688 (.020)	.688 (.020)	.929 (.003)	.929 (.003)	.800 (.002)	**.800** (.002)
HCApp(LR)	.795 (.019)	.795 (.019)	.692 (.016)	.692 (.016)	.929 (.004)	.929 (.004)	.789 (.011)	.789 (.011)
HCApp(LR) w/ disc.	.789 (.022)	.789 (.022)	.690 (.019)	.690 (.019)	.928 (.003)	.928 (.003)	.800 (.002)	**.800** (.002)
HC(ANN)	.780 (.026)	.780 (.026)	.682 (.021)	.682 (.021)	.929 (.003)	.929 (.003)	.800 (.003)	**.800** (.003)
HC(ANN) w/ disc.	.771 (.027)	.771 (.027)	.686 (.021)	.686 (.021)	.929 (.003)	.929 (.003)	.800 (.002)	**.800** (.002)
Mincut	.769 (.026)	.769 (.026)	.621 (.028)	.621 (.028)	.078 (.004)	.078 (.004)	.798 (.004)	.798 (.004)
Mincut w/ disc.	.794 (.020)	.794 (.020)	.678 (.022)	.678 (.022)	.929 (.004)	.929 (.004)	.800 (.002)	**.800** (.002)
IC-LR	.788 (.021)	.788 (.021)	.696 (.019)	.696 (.019)	**.931** (.004)	**.931** (.004)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.796 (.017)	.796 (.017)	.701 (.018)	.701 (.018)	**.931** (.004)	**.931** (.004)	.800 (.002)	**.800** (.002)
IC-LR w/ neg.	.797 (.018)	**.797** (.018)	.717 (.017)	**.717** (.017)	**.931** (.004)	**.931** (.004)	.800 (.003)	**.800** (.003)
Maj	.652 (.024)	.698 (.086)	.471 (.022)	.678 (.026)	.074 (.003)	.568 (.091)	.809 (.003)	.780 (.003)
Maj w/ disc.	.798 (.022)	.647 (.076)	.632 (.035)	.694 (.017)	.922 (.005)	**.931** (.003)	**.814** (.002)	.781 (.004)
Imp(LR)	.803 (.020)	.740 (.095)	**.728** (.016)	.700 (.014)	.930 (.003)	**.931** (.003)	.802 (.003)	.779 (.002)
Imp(LR) w/ disc.	.799 (.019)	.709 (.105)	.718 (.017)	.700 (.015)	.930 (.004)	.930 (.003)	.813 (.003)	.779 (.002)
Imp(ANN)	.806 (.018)	.729 (.093)	.727 (.016)	.700 (.014)	**.931** (.003)	**.931** (.003)	.813 (.003)	.779 (.002)
Imp(ANN) w/ disc.	.798 (.019)	.700 (.107)	.716 (.017)	.700 (.015)	.930 (.004)	.930 (.003)	.813 (.002)	.779 (.002)
Imp(RF)	.775 (.021)	.623 (.081)	.690 (.019)	.700 (.014)	.928 (.003)	**.931** (.003)	.811 (.003)	.780 (.003)
Imp(RF) w/ disc.	.797 (.019)	.691 (.109)	.706 (.018)	.700 (.015)	.929 (.003)	.930 (.003)	**.814** (.003)	.779 (.003)
Imp(κNN)	.792 (.020)	.675 (.095)	.711 (.020)	.698 (.016)	.930 (.004)	**.931** (.003)	.807 (.005)	.779 (.003)
Imp(κNN) w/ disc.	.789 (.023)	.670 (.112)	.698 (.028)	.694 (.029)	.928 (.004)	.930 (.003)	.803 (.027)	.778 (.005)
R-F(LR)	**.807** (.019)	.577 (.039)	.700 (.019)	.696 (.014)	.929 (.003)	.929 (.003)	.803 (.003)	.779 (.003)
R-F(LR) w/ disc.	.798 (.020)	.568 (.032)	.704 (.018)	.697 (.015)	.928 (.004)	.929 (.003)	.813 (.003)	.780 (.003)
R-F(ANN)	.804 (.019)	.576 (.039)	.698 (.020)	.696 (.014)	.929 (.003)	.929 (.003)	.805 (.003)	.778 (.002)
R-F(ANN) w/ disc.	.795 (.020)	.576 (.045)	.701 (.019)	.698 (.015)	.929 (.003)	.929 (.003)	.809 (.003)	.778 (.002)
R-F(RF)	.780 (.020)	.577 (.040)	.675 (.019)	.696 (.015)	.924 (.003)	.928 (.003)	.810 (.003)	.782 (.005)
R-F(RF) w/ disc.	.799 (.020)	.576 (.043)	.689 (.019)	.698 (.015)	.925 (.004)	.929 (.003)	.813 (.003)	.784 (.007)
R-F(κNN)	.781 (.024)	.567 (.032)	.691 (.020)	.695 (.014)	.929 (.004)	.929 (.003)	.806 (.004)	.781 (.004)
R-F(κNN) w/ disc.	.780 (.024)	.567 (.036)	.692 (.020)	.697 (.015)	.927 (.005)	.929 (.003)	.806 (.004)	.782 (.005)

Table 21: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.775 (.021)	.775 (.021)	.685 (.019)	.685 (.019)	.923 (.004)	.923 (.004)	.796 (.003)	.796 (.003)
HC(LR) w/ disc.	.771 (.023)	.771 (.023)	.682 (.021)	.682 (.021)	.923 (.004)	.923 (.004)	.796 (.003)	.796 (.003)
HCApp(LR)	.776 (.021)	**.776** (.021)	.690 (.017)	.690 (.017)	.923 (.003)	.923 (.003)	.789 (.010)	.789 (.010)
HCApp(LR) w/ disc.	.771 (.026)	.771 (.026)	.694 (.018)	.694 (.018)	.922 (.003)	.922 (.003)	.796 (.003)	.796 (.003)
HC(ANN)	.755 (.026)	.755 (.026)	.673 (.022)	.673 (.022)	.923 (.004)	.923 (.004)	.796 (.003)	.796 (.003)
HC(ANN) w/ disc.	.747 (.031)	.747 (.031)	.692 (.020)	.692 (.020)	.923 (.004)	.923 (.004)	.796 (.003)	.796 (.003)
Mincut	.746 (.024)	.746 (.024)	.634 (.027)	.634 (.027)	.079 (.004)	.079 (.004)	.792 (.006)	.792 (.006)
Mincut w/ disc.	.772 (.023)	.772 (.023)	.677 (.022)	.677 (.022)	**.930** (.003)	**.930** (.003)	.796 (.003)	.796 (.003)
IC-LR	.762 (.030)	.762 (.030)	.699 (.014)	.699 (.014)	**.930** (.003)	**.930** (.003)	.778 (.003)	.778 (.003)
IC-LR w/ disc.	.773 (.020)	.773 (.020)	.700 (.014)	.700 (.014)	**.930** (.003)	**.930** (.003)	.782 (.007)	.782 (.007)
IC-LR w/ neg.	.770 (.024)	.770 (.024)	.717 (.016)	**.717** (.016)	**.930** (.003)	**.930** (.003)	.797 (.003)	**.797** (.003)
Maj	.642 (.026)	.659 (.084)	.477 (.023)	.693 (.020)	.079 (.004)	.729 (.008)	.807 (.003)	.780 (.003)
Maj w/ disc.	.783 (.022)	.630 (.076)	.631 (.036)	.698 (.015)	.923 (.005)	**.930** (.003)	.811 (.002)	.781 (.004)
Imp(LR)	.783 (.021)	.709 (.090)	**.723** (.017)	.701 (.014)	**.930** (.003)	**.930** (.003)	.799 (.003)	.779 (.003)
Imp(LR) w/ disc.	.773 (.022)	.668 (.104)	.711 (.018)	.699 (.015)	**.930** (.003)	**.930** (.003)	.808 (.003)	.779 (.003)
Imp(ANN)	.784 (.021)	.695 (.093)	**.723** (.017)	.701 (.014)	**.930** (.003)	**.930** (.003)	.808 (.003)	.779 (.003)
Imp(ANN) w/ disc.	.772 (.021)	.657 (.105)	.710 (.017)	.699 (.015)	**.930** (.003)	**.930** (.003)	**.812** (.002)	.779 (.003)
Imp(RF)	.755 (.024)	.615 (.078)	.688 (.021)	.701 (.014)	.926 (.004)	**.930** (.003)	.809 (.003)	.780 (.003)
Imp(RF) w/ disc.	.773 (.020)	.655 (.105)	.702 (.018)	.699 (.015)	.928 (.003)	**.930** (.003)	.811 (.003)	.779 (.003)
Imp(κNN)	.770 (.024)	.635 (.090)	.704 (.022)	.697 (.019)	**.930** (.004)	**.930** (.003)	.803 (.005)	.779 (.003)
Imp(κNN) w/ disc.	.763 (.030)	.644 (.102)	.693 (.030)	.689 (.038)	.928 (.005)	.928 (.021)	.802 (.009)	.775 (.020)
R-F(LR)	**.791** (.022)	.604 (.050)	.688 (.020)	.693 (.015)	.923 (.004)	.923 (.004)	.797 (.003)	.776 (.003)
R-F(LR) w/ disc.	.783 (.022)	.596 (.057)	.697 (.019)	.696 (.016)	.922 (.004)	.923 (.004)	.807 (.002)	.781 (.007)
R-F(ANN)	.788 (.023)	.601 (.051)	.689 (.020)	.693 (.015)	.923 (.004)	.923 (.004)	.797 (.003)	.774 (.003)
R-F(ANN) w/ disc.	.780 (.024)	.613 (.061)	.696 (.019)	.696 (.016)	.923 (.004)	.923 (.004)	.802 (.003)	.774 (.003)
R-F(RF)	.765 (.023)	.609 (.048)	.666 (.021)	.692 (.015)	.915 (.004)	.920 (.004)	.804 (.003)	.782 (.007)
R-F(RF) w/ disc.	.785 (.023)	.631 (.068)	.683 (.022)	.696 (.016)	.919 (.004)	.923 (.004)	.807 (.003)	.783 (.008)
R-F(κNN)	.770 (.025)	.597 (.049)	.686 (.020)	.693 (.015)	.923 (.004)	.923 (.004)	.800 (.004)	.780 (.005)
R-F(κNN) w/ disc.	.767 (.026)	.600 (.057)	.686 (.022)	.696 (.016)	.921 (.004)	.923 (.004)	.798 (.010)	.779 (.006)

Table 22: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.756 (.023)	.756 (.023)	.673 (.021)	.673 (.021)	.908 (.005)	.908 (.005)	.786 (.002)	.786 (.002)
HC(LR) w/ disc.	.749 (.025)	.749 (.025)	.686 (.023)	.686 (.023)	.908 (.005)	.908 (.005)	.786 (.002)	.786 (.002)
HCApp(LR)	.759 (.023)	**.759** (.023)	.675 (.018)	.675 (.018)	.908 (.004)	.908 (.004)	.779 (.010)	.779 (.010)
HCApp(LR) w/ disc.	.749 (.025)	.749 (.025)	.692 (.017)	.692 (.017)	.908 (.004)	.908 (.004)	.786 (.002)	.786 (.002)
HC(ANN)	.733 (.027)	.733 (.027)	.669 (.022)	.669 (.022)	.908 (.004)	.908 (.004)	.786 (.002)	.786 (.002)
HC(ANN) w/ disc.	.724 (.033)	.724 (.033)	.694 (.019)	.694 (.019)	.909 (.005)	.909 (.005)	.786 (.003)	.786 (.003)
Mincut	.725 (.029)	.725 (.029)	.633 (.028)	.633 (.028)	.079 (.004)	.079 (.004)	.777 (.013)	.777 (.013)
Mincut w/ disc.	.749 (.025)	.749 (.025)	.685 (.024)	.685 (.024)	**.931** (.004)	**.931** (.004)	.785 (.006)	.785 (.006)
IC-LR	.740 (.027)	.740 (.027)	.700 (.013)	.700 (.013)	**.931** (.003)	**.931** (.003)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.744 (.023)	.744 (.023)	.699 (.014)	.699 (.014)	.930 (.003)	.930 (.003)	.779 (.002)	.779 (.002)
IC-LR w/ neg.	.746 (.026)	.746 (.026)	.708 (.014)	**.708** (.014)	**.931** (.003)	**.931** (.003)	.795 (.002)	**.795** (.002)
Maj	.644 (.027)	.634 (.083)	.501 (.024)	.697 (.018)	.093 (.004)	.823 (.007)	.804 (.003)	.779 (.002)
Maj w/ disc.	.767 (.026)	.620 (.077)	.640 (.032)	.699 (.015)	.924 (.004)	**.931** (.003)	.808 (.002)	.780 (.003)
Imp(LR)	.763 (.023)	.678 (.089)	**.716** (.017)	.700 (.014)	**.931** (.003)	**.931** (.003)	.796 (.003)	.779 (.002)
Imp(LR) w/ disc.	.748 (.025)	.637 (.093)	.708 (.016)	.701 (.014)	.929 (.004)	.930 (.003)	.806 (.003)	.779 (.002)
Imp(ANN)	.763 (.022)	.659 (.093)	.715 (.017)	.700 (.014)	**.931** (.003)	**.931** (.003)	.805 (.002)	.779 (.002)
Imp(ANN) w/ disc.	.748 (.024)	.627 (.093)	.706 (.016)	.701 (.014)	.930 (.003)	.930 (.003)	**.809** (.002)	.779 (.002)
Imp(RF)	.736 (.024)	.604 (.074)	.685 (.020)	.700 (.014)	.924 (.004)	**.931** (.003)	.805 (.002)	.779 (.002)
Imp(RF) w/ disc.	.749 (.023)	.627 (.093)	.700 (.018)	.701 (.014)	.929 (.004)	.930 (.003)	.808 (.002)	.779 (.002)
Imp(κNN)	.748 (.025)	.615 (.080)	.696 (.026)	.692 (.028)	.930 (.004)	**.931** (.004)	.801 (.006)	.778 (.003)
Imp(κNN) w/ disc.	.734 (.038)	.626 (.089)	.686 (.040)	.682 (.054)	.927 (.007)	.927 (.030)	.800 (.013)	.764 (.064)
R-F(LR)	**.775** (.024)	.645 (.055)	.675 (.021)	.686 (.016)	.908 (.004)	.908 (.005)	.784 (.003)	.767 (.003)
R-F(LR) w/ disc.	.764 (.024)	.634 (.064)	.694 (.018)	.697 (.015)	.907 (.005)	.908 (.004)	.795 (.003)	.776 (.007)
R-F(ANN)	.769 (.025)	.630 (.056)	.675 (.021)	.686 (.016)	.908 (.004)	.908 (.004)	.782 (.003)	.765 (.002)
R-F(ANN) w/ disc.	.762 (.025)	.640 (.063)	.694 (.018)	.697 (.015)	.908 (.005)	.908 (.004)	.790 (.003)	.766 (.003)
R-F(RF)	.749 (.025)	.649 (.047)	.654 (.022)	.682 (.016)	.896 (.005)	.901 (.005)	.791 (.003)	.775 (.005)
R-F(RF) w/ disc.	.771 (.024)	.670 (.065)	.682 (.021)	.696 (.015)	.905 (.005)	.908 (.004)	.794 (.003)	.777 (.008)
R-F(κNN)	.755 (.027)	.649 (.056)	.674 (.020)	.684 (.017)	.908 (.004)	.908 (.004)	.787 (.006)	.771 (.006)
R-F(κNN) w/ disc.	.755 (.029)	.652 (.064)	.681 (.023)	.695 (.016)	.907 (.005)	.908 (.005)	.787 (.008)	.772 (.008)

Table 23: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.729 (.026)	.729 (.026)	.660 (.021)	.660 (.021)	.877 (.005)	.877 (.005)	.764 (.003)	.764 (.003)
HC(LR) w/ disc.	.724 (.025)	.724 (.025)	.692 (.023)	.692 (.023)	.880 (.013)	.880 (.013)	.765 (.003)	.765 (.003)
HCApp(LR)	.734 (.021)	**.734** (.021)	.666 (.021)	.666 (.021)	.877 (.005)	.877 (.005)	.759 (.008)	.759 (.008)
HCApp(LR) w/ disc.	.730 (.023)	.730 (.023)	.692 (.023)	.692 (.023)	.881 (.014)	.881 (.014)	.764 (.002)	.764 (.002)
HC(ANN)	.714 (.030)	.714 (.030)	.663 (.021)	.663 (.021)	.877 (.005)	.877 (.005)	.765 (.003)	.765 (.003)
HC(ANN) w/ disc.	.706 (.032)	.706 (.032)	.694 (.019)	.694 (.019)	.880 (.013)	.880 (.013)	.765 (.003)	.765 (.003)
Mincut	.714 (.031)	.714 (.031)	.647 (.039)	.647 (.039)	.101 (.130)	.101 (.130)	.752 (.025)	.752 (.025)
Mincut w/ disc.	.724 (.025)	.724 (.025)	.696 (.020)	.696 (.020)	.930 (.006)	.930 (.006)	.771 (.021)	.771 (.021)
IC-LR	.709 (.026)	.709 (.026)	.700 (.015)	.700 (.015)	**.931** (.003)	**.931** (.003)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.715 (.022)	.715 (.022)	.700 (.015)	.700 (.015)	**.931** (.003)	**.931** (.003)	.779 (.002)	.779 (.002)
IC-LR w/ neg.	.714 (.029)	.714 (.029)	.706 (.017)	**.706** (.017)	**.931** (.003)	**.931** (.003)	.792 (.002)	**.792** (.002)
Maj	.643 (.027)	.615 (.079)	.532 (.023)	.699 (.015)	.125 (.005)	.878 (.005)	.800 (.003)	.779 (.002)
Maj w/ disc.	.745 (.026)	.608 (.076)	.649 (.028)	.699 (.014)	.924 (.005)	**.931** (.003)	**.805** (.003)	.780 (.002)
Imp(LR)	.736 (.023)	.628 (.084)	**.710** (.016)	.700 (.014)	.930 (.003)	**.931** (.003)	.793 (.003)	.779 (.002)
Imp(LR) w/ disc.	.719 (.029)	.600 (.073)	.704 (.017)	.701 (.014)	.930 (.004)	**.931** (.003)	.804 (.003)	.779 (.002)
Imp(ANN)	.734 (.022)	.613 (.081)	.709 (.016)	.700 (.014)	.930 (.003)	**.931** (.003)	.801 (.003)	.779 (.002)
Imp(ANN) w/ disc.	.720 (.028)	.594 (.071)	.703 (.016)	.701 (.014)	**.931** (.003)	**.931** (.003)	.803 (.003)	.779 (.002)
Imp(RF)	.711 (.024)	.593 (.064)	.684 (.019)	.700 (.014)	.920 (.004)	**.931** (.003)	.802 (.003)	.779 (.002)
Imp(RF) w/ disc.	.721 (.027)	.596 (.071)	.698 (.017)	.701 (.014)	.930 (.004)	**.931** (.003)	.804 (.003)	.779 (.002)
Imp(κNN)	.723 (.029)	.602 (.071)	.688 (.030)	.689 (.035)	.929 (.004)	.930 (.008)	.798 (.007)	.777 (.007)
Imp(κNN) w/ disc.	.702 (.040)	.595 (.071)	.677 (.052)	.680 (.060)	.928 (.005)	**.931** (.003)	.795 (.021)	.774 (.026)
R-F(LR)	**.749** (.024)	.667 (.042)	.657 (.020)	.670 (.018)	.876 (.005)	.877 (.005)	.761 (.003)	.747 (.003)
R-F(LR) w/ disc.	.744 (.027)	.667 (.061)	.689 (.021)	.696 (.019)	.878 (.011)	.879 (.011)	.770 (.003)	.758 (.006)
R-F(ANN)	.740 (.026)	.647 (.047)	.657 (.020)	.670 (.018)	.877 (.005)	.877 (.005)	.756 (.004)	.745 (.003)
R-F(ANN) w/ disc.	.735 (.029)	.649 (.060)	.691 (.022)	.696 (.019)	.879 (.011)	.879 (.011)	.765 (.003)	.748 (.004)
R-F(RF)	.729 (.024)	.668 (.035)	.639 (.021)	.665 (.018)	.861 (.006)	.865 (.006)	.767 (.003)	.757 (.005)
R-F(RF) w/ disc.	.748 (.025)	.688 (.057)	.679 (.024)	.695 (.020)	.876 (.011)	.878 (.011)	.769 (.003)	.759 (.006)
R-F(κNN)	.732 (.027)	.669 (.043)	.654 (.021)	.666 (.019)	.876 (.005)	.877 (.005)	.764 (.005)	.751 (.007)
R-F(κNN) w/ disc.	.730 (.029)	.677 (.058)	.673 (.026)	.689 (.025)	.877 (.011)	.878 (.011)	.764 (.007)	.751 (.011)

Table 24: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.907 (.012)	.907 (.012)	.723 (.033)	.723 (.033)	.729 (.087)	.729 (.087)	.705 (.004)	.705 (.004)
HC(LR) w/ disc.	**.911** (.012)	**.911** (.012)	.738 (.020)	.738 (.020)	.797 (.048)	.797 (.048)	.730 (.007)	.730 (.007)
HCApp(LR)	.910 (.012)	.910 (.012)	.694 (.034)	.694 (.034)	.729 (.067)	.729 (.067)	.674 (.026)	.674 (.026)
HCApp(LR) w/ disc.	.904 (.030)	.904 (.030)	.685 (.031)	.685 (.031)	.781 (.019)	.781 (.019)	.717 (.004)	.717 (.004)
HC(ANN)	.908 (.012)	.908 (.012)	.745 (.026)	.745 (.026)	.795 (.050)	.795 (.050)	.729 (.014)	.729 (.014)
HC(ANN) w/ disc.	.910 (.012)	.910 (.012)	.747 (.019)	.747 (.019)	.818 (.040)	.818 (.040)	.730 (.007)	.730 (.007)
IC-LR	.901 (.011)	.901 (.011)	.744 (.015)	.744 (.015)	.693 (.027)	.693 (.027)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.910 (.011)	.910 (.011)	.746 (.015)	.746 (.015)	.824 (.013)	.824 (.013)	.730 (.004)	.730 (.004)
IC-LR w/ neg.	.905 (.012)	.905 (.012)	**.765** (.014)	**.765** (.014)	.697 (.033)	.697 (.033)	.704 (.004)	.704 (.004)
Imp(LR)	.907 (.012)	.895 (.036)	.761 (.017)	.680 (.020)	.766 (.026)	.579 (.062)	.705 (.004)	.625 (.027)
Imp(LR) w/ disc.	.910 (.012)	.868 (.070)	.744 (.016)	.644 (.019)	.825 (.013)	.608 (.070)	.731 (.004)	.573 (.032)
Imp(ANN)	.908 (.012)	.895 (.035)	.762 (.017)	.681 (.020)	.809 (.016)	.728 (.014)	.735 (.009)	.653 (.018)
Imp(ANN) w/ disc.	.910 (.012)	.868 (.070)	.747 (.016)	.647 (.019)	**.827** (.014)	.625 (.062)	.732 (.004)	.587 (.003)
Imp(RF)	.886 (.013)	.841 (.106)	.711 (.017)	.578 (.037)	.813 (.012)	.689 (.055)	**.736** (.004)	.542 (.041)
Imp(RF) w/ disc.	**.911** (.013)	.826 (.189)	.717 (.018)	.604 (.034)	.808 (.019)	.556 (.085)	.731 (.004)	.521 (.077)
Imp(κNN)	.906 (.013)	.833 (.122)	.724 (.020)	.562 (.051)	.755 (.020)	.523 (.017)	.709 (.020)	.528 (.056)
Imp(κNN) w/ disc.	.903 (.016)	.827 (.136)	.709 (.026)	.562 (.048)	.752 (.042)	.495 (.031)	.706 (.026)	.487 (.078)
R-F(LR)	.907 (.012)	.907 (.012)	.761 (.017)	.761 (.017)	.766 (.026)	.766 (.026)	.705 (.004)	.705 (.004)
R-F(LR) w/ disc.	.910 (.012)	.910 (.012)	.744 (.016)	.744 (.016)	.825 (.013)	.825 (.013)	.731 (.004)	.731 (.004)
R-F(ANN)	.908 (.012)	.908 (.012)	.763 (.017)	.762 (.017)	.809 (.016)	.809 (.016)	.735 (.008)	.734 (.009)
R-F(ANN) w/ disc.	.910 (.012)	.910 (.012)	.747 (.016)	.747 (.016)	**.827** (.014)	**.827** (.014)	.732 (.004)	.732 (.004)
R-F(RF)	.886 (.013)	.886 (.013)	.711 (.017)	.711 (.017)	.813 (.012)	.813 (.012)	**.736** (.004)	**.736** (.004)
R-F(RF) w/ disc.	**.911** (.013)	**.911** (.013)	.717 (.018)	.717 (.018)	.808 (.019)	.809 (.019)	.732 (.004)	.732 (.004)
R-F(κNN)	.905 (.013)	.905 (.013)	.724 (.021)	.724 (.020)	.756 (.020)	.755 (.020)	.709 (.020)	.710 (.020)
R-F(κNN) w/ disc.	.903 (.018)	.903 (.016)	.707 (.027)	.709 (.025)	.758 (.038)	.754 (.039)	.705 (.028)	.707 (.026)

Table 25: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.890 (.018)	.890 (.018)	.682 (.033)	.682 (.033)	.653 (.072)	.653 (.072)	.655 (.013)	.655 (.013)
HC(LR) w/ disc.	.890 (.016)	.890 (.016)	.687 (.031)	.687 (.031)	.715 (.041)	.715 (.041)	.681 (.020)	.681 (.020)
HCApp(LR)	.887 (.014)	.887 (.014)	.684 (.039)	.684 (.039)	.646 (.081)	.646 (.081)	.637 (.025)	.637 (.025)
HCApp(LR) w/ disc.	.886 (.014)	.886 (.014)	.665 (.039)	.665 (.039)	.685 (.093)	.685 (.093)	.682 (.016)	.682 (.016)
HC(ANN)	.886 (.016)	.886 (.016)	.690 (.031)	.690 (.031)	.682 (.063)	.682 (.063)	.673 (.010)	.673 (.010)
HC(ANN) w/ disc.	.876 (.018)	.876 (.018)	.670 (.046)	.670 (.046)	.738 (.028)	.738 (.028)	.688 (.017)	.688 (.017)
IC-LR	.874 (.015)	.874 (.015)	.723 (.016)	.723 (.016)	.687 (.028)	.687 (.028)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.891 (.013)	.891 (.013)	.724 (.015)	.724 (.015)	.790 (.013)	**.790** (.013)	.709 (.004)	**.709** (.004)
IC-LR w/ neg.	**.893** (.012)	**.893** (.012)	**.746** (.015)	**.746** (.015)	.699 (.033)	.699 (.033)	.690 (.003)	.690 (.003)
Imp(LR)	.892 (.015)	.873 (.043)	.741 (.018)	.666 (.020)	.760 (.028)	.658 (.065)	.694 (.006)	.593 (.014)
Imp(LR) w/ disc.	**.893** (.015)	.832 (.075)	.723 (.019)	.633 (.019)	.807 (.014)	.634 (.070)	.732 (.004)	.675 (.022)
Imp(ANN)	**.893** (.014)	.873 (.041)	.743 (.018)	.667 (.020)	.801 (.017)	.722 (.015)	.724 (.006)	.599 (.013)
Imp(ANN) w/ disc.	.892 (.015)	.833 (.075)	.726 (.019)	.635 (.018)	**.810** (.014)	.640 (.064)	**.733** (.004)	.677 (.020)
Imp(RF)	.865 (.016)	.826 (.073)	.685 (.021)	.592 (.048)	.799 (.014)	.646 (.059)	.720 (.005)	.624 (.041)
Imp(RF) w/ disc.	.892 (.016)	.817 (.112)	.696 (.022)	.605 (.030)	.795 (.017)	.595 (.075)	**.733** (.004)	.672 (.028)
Imp(κNN)	.885 (.016)	.824 (.103)	.703 (.022)	.582 (.047)	.740 (.021)	.532 (.024)	.704 (.014)	.624 (.042)
Imp(κNN) w/ disc.	.881 (.021)	.798 (.124)	.682 (.030)	.571 (.045)	.743 (.043)	.506 (.045)	.702 (.018)	.565 (.065)
R-F(LR)	.884 (.014)	.798 (.069)	.709 (.021)	.616 (.049)	.748 (.024)	.587 (.090)	.696 (.005)	.538 (.101)
R-F(LR) w/ disc.	.886 (.016)	.787 (.070)	.711 (.020)	.638 (.048)	.799 (.015)	.703 (.045)	.725 (.004)	.560 (.135)
R-F(ANN)	.883 (.015)	.822 (.059)	.707 (.022)	.627 (.046)	.770 (.019)	.683 (.060)	.714 (.006)	.603 (.087)
R-F(ANN) w/ disc.	.883 (.016)	.822 (.045)	.704 (.020)	.637 (.050)	.788 (.015)	.728 (.039)	.725 (.004)	.667 (.051)
R-F(RF)	.863 (.015)	.795 (.061)	.679 (.022)	.621 (.049)	.782 (.013)	.551 (.053)	.720 (.005)	.561 (.077)
R-F(RF) w/ disc.	.880 (.017)	.801 (.060)	.685 (.023)	.637 (.050)	.758 (.021)	.599 (.043)	.724 (.004)	.533 (.117)
R-F(κNN)	.873 (.017)	.785 (.060)	.678 (.024)	.611 (.043)	.725 (.021)	.524 (.030)	.703 (.013)	.624 (.054)
R-F(κNN) w/ disc.	.871 (.021)	.774 (.077)	.681 (.026)	.618 (.049)	.734 (.031)	.523 (.035)	.701 (.018)	.610 (.063)

Table 26: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.876 (.017)	.876 (.017)	.661 (.031)	.661 (.031)	.626 (.080)	.626 (.080)	.642 (.011)	.642 (.011)
HC(LR) w/ disc.	.872 (.017)	.872 (.017)	.668 (.035)	.668 (.035)	.682 (.071)	.682 (.071)	.676 (.014)	.676 (.014)
HCApp(LR)	.875 (.016)	.875 (.016)	.652 (.043)	.652 (.043)	.608 (.075)	.608 (.075)	.630 (.021)	.630 (.021)
HCApp(LR) w/ disc.	.867 (.021)	.867 (.021)	.646 (.040)	.646 (.040)	.657 (.094)	.657 (.094)	.678 (.014)	.678 (.014)
HC(ANN)	.866 (.021)	.866 (.021)	.654 (.037)	.654 (.037)	.674 (.053)	.674 (.053)	.654 (.007)	.654 (.007)
HC(ANN) w/ disc.	.853 (.024)	.853 (.024)	.637 (.043)	.637 (.043)	.701 (.036)	.701 (.036)	.679 (.012)	.679 (.012)
IC-LR	.850 (.017)	.850 (.017)	.701 (.016)	.701 (.016)	.679 (.021)	.679 (.021)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.873 (.014)	.873 (.014)	.702 (.017)	.702 (.017)	.760 (.016)	**.760** (.016)	.694 (.004)	**.694** (.004)
IC-LR w/ neg.	.879 (.013)	**.879** (.013)	**.725** (.018)	**.725** (.018)	.688 (.027)	.688 (.027)	.683 (.005)	.683 (.005)
Imp(LR)	.879 (.016)	.853 (.045)	.722 (.021)	.653 (.022)	.757 (.030)	.681 (.043)	.684 (.005)	.573 (.014)
Imp(LR) w/ disc.	.874 (.015)	.797 (.081)	.705 (.021)	.621 (.019)	.796 (.014)	.647 (.062)	.725 (.004)	.668 (.013)
Imp(ANN)	**.880** (.016)	.853 (.044)	.724 (.021)	.654 (.022)	.790 (.018)	.712 (.017)	.709 (.007)	.578 (.020)
Imp(ANN) w/ disc.	.873 (.015)	.797 (.080)	.708 (.021)	.622 (.019)	**.798** (.014)	.648 (.059)	**.726** (.004)	.670 (.011)
Imp(RF)	.850 (.017)	.794 (.077)	.665 (.023)	.604 (.032)	.781 (.015)	.591 (.056)	.711 (.005)	.640 (.027)
Imp(RF) w/ disc.	.873 (.016)	.795 (.084)	.681 (.023)	.604 (.027)	.786 (.016)	.624 (.064)	**.726** (.004)	.666 (.010)
Imp(κNN)	.869 (.018)	.809 (.087)	.681 (.026)	.584 (.042)	.726 (.018)	.533 (.029)	.698 (.010)	.626 (.039)
Imp(κNN) w/ disc.	.861 (.022)	.765 (.115)	.660 (.036)	.570 (.043)	.729 (.038)	.513 (.043)	.692 (.021)	.564 (.059)
R-F(LR)	.870 (.017)	.782 (.065)	.680 (.024)	.601 (.045)	.739 (.021)	.611 (.074)	.690 (.005)	.633 (.030)
R-F(LR) w/ disc.	.867 (.017)	.761 (.070)	.690 (.023)	.619 (.049)	.783 (.016)	.662 (.078)	.718 (.004)	.674 (.022)
R-F(ANN)	.867 (.018)	.808 (.048)	.674 (.026)	.611 (.043)	.751 (.023)	.674 (.050)	.699 (.005)	.656 (.014)
R-F(ANN) w/ disc.	.861 (.018)	.799 (.040)	.678 (.026)	.618 (.048)	.773 (.018)	.693 (.048)	.718 (.004)	.686 (.008)
R-F(RF)	.850 (.017)	.771 (.064)	.656 (.024)	.595 (.049)	.756 (.017)	.524 (.032)	.710 (.005)	.643 (.021)
R-F(RF) w/ disc.	.861 (.018)	.782 (.046)	.667 (.024)	.622 (.052)	.738 (.021)	.583 (.056)	.718 (.004)	.664 (.024)
R-F(κNN)	.846 (.019)	.735 (.060)	.649 (.027)	.574 (.041)	.710 (.020)	.530 (.031)	.697 (.009)	.639 (.030)
R-F(κNN) w/ disc.	.844 (.020)	.724 (.080)	.660 (.028)	.593 (.052)	.720 (.027)	.516 (.026)	.692 (.016)	.634 (.026)

Table 27: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.855 (.019)	.855 (.019)	.648 (.034)	.648 (.034)	.634 (.067)	.634 (.067)	.641 (.008)	.641 (.008)
HC(LR) w/ disc.	.849 (.020)	.849 (.020)	.656 (.032)	.656 (.032)	.649 (.076)	.649 (.076)	.669 (.012)	.669 (.012)
HCApp(LR)	.854 (.016)	.854 (.016)	.644 (.040)	.644 (.040)	.602 (.093)	.602 (.093)	.628 (.023)	.628 (.023)
HCApp(LR) w/ disc.	.846 (.019)	.846 (.019)	.646 (.043)	.646 (.043)	.621 (.094)	.621 (.094)	.669 (.012)	.669 (.012)
HC(ANN)	.841 (.024)	.841 (.024)	.635 (.037)	.635 (.037)	.660 (.054)	.660 (.054)	.645 (.007)	.645 (.007)
HC(ANN) w/ disc.	.830 (.026)	.830 (.026)	.622 (.039)	.622 (.039)	.666 (.037)	.666 (.037)	.668 (.012)	.668 (.012)
IC-LR	.822 (.013)	.822 (.013)	.680 (.021)	.680 (.021)	.683 (.023)	.683 (.023)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.853 (.015)	.853 (.015)	.683 (.022)	.683 (.022)	.733 (.015)	**.733** (.015)	.681 (.006)	**.681** (.006)
IC-LR w/ neg.	**.867** (.014)	**.867** (.014)	**.709** (.019)	**.709** (.019)	.694 (.029)	.694 (.029)	.674 (.004)	.674 (.004)
Imp(LR)	.863 (.018)	.824 (.050)	.705 (.022)	.638 (.022)	.749 (.028)	.685 (.031)	.675 (.005)	.563 (.013)
Imp(LR) w/ disc.	.853 (.017)	.764 (.081)	.682 (.023)	.607 (.020)	.778 (.016)	.640 (.056)	.717 (.004)	.665 (.012)
Imp(ANN)	.862 (.018)	.824 (.050)	.707 (.022)	.638 (.021)	**.780** (.018)	.703 (.017)	.695 (.006)	.565 (.015)
Imp(ANN) w/ disc.	.852 (.017)	.764 (.081)	.684 (.023)	.608 (.020)	**.780** (.016)	.640 (.055)	**.718** (.004)	.665 (.012)
Imp(RF)	.828 (.022)	.757 (.088)	.649 (.027)	.594 (.030)	.762 (.018)	.626 (.047)	.703 (.005)	.651 (.012)
Imp(RF) w/ disc.	.853 (.018)	.761 (.087)	.663 (.025)	.594 (.025)	.768 (.017)	.625 (.056)	.717 (.004)	.660 (.012)
Imp(κNN)	.847 (.022)	.766 (.091)	.660 (.027)	.578 (.043)	.715 (.019)	.545 (.039)	.688 (.008)	.619 (.036)
Imp(κNN) w/ disc.	.838 (.026)	.734 (.099)	.634 (.037)	.559 (.040)	.711 (.041)	.523 (.054)	.686 (.015)	.577 (.050)
R-F(LR)	.854 (.019)	.779 (.068)	.659 (.026)	.603 (.040)	.721 (.022)	.618 (.076)	.682 (.005)	.638 (.010)
R-F(LR) w/ disc.	.849 (.020)	.753 (.077)	.670 (.024)	.617 (.044)	.759 (.017)	.635 (.086)	.710 (.004)	.673 (.013)
R-F(ANN)	.849 (.020)	.796 (.047)	.655 (.025)	.605 (.039)	.728 (.023)	.664 (.051)	.685 (.005)	.649 (.008)
R-F(ANN) w/ disc.	.844 (.020)	.791 (.038)	.653 (.028)	.606 (.042)	.740 (.017)	.659 (.048)	.710 (.004)	.676 (.008)
R-F(RF)	.831 (.021)	.755 (.060)	.636 (.026)	.592 (.045)	.732 (.019)	.545 (.030)	.700 (.005)	.650 (.012)
R-F(RF) w/ disc.	.843 (.020)	.776 (.052)	.654 (.025)	.617 (.040)	.719 (.026)	.553 (.088)	.710 (.004)	.672 (.012)
R-F(κNN)	.826 (.023)	.730 (.061)	.627 (.027)	.571 (.037)	.693 (.023)	.550 (.032)	.685 (.008)	.637 (.018)
R-F(κNN) w/ disc.	.825 (.022)	.729 (.055)	.643 (.028)	.589 (.048)	.705 (.026)	.549 (.041)	.682 (.011)	.632 (.024)

Table 28: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia Tru.	Australia Str.	Germany Tru.	Germany Str.	Poland Tru.	Poland Str.	Taiwan Tru.	Taiwan Str.
HC(LR)	.827 (.020)	.827 (.020)	.627 (.036)	.627 (.036)	.647 (.043)	.647 (.043)	.633 (.006)	.633 (.006)
HC(LR) w/ disc.	.820 (.022)	.820 (.022)	.636 (.030)	.636 (.030)	.663 (.044)	.663 (.044)	.660 (.007)	.660 (.007)
HCApp(LR)	.829 (.018)	.829 (.018)	.623 (.042)	.623 (.042)	.640 (.054)	.640 (.054)	.626 (.017)	.626 (.017)
HCApp(LR) w/ disc.	.820 (.020)	.820 (.020)	.634 (.026)	.634 (.026)	.665 (.054)	.665 (.054)	.659 (.008)	.659 (.008)
HC(ANN)	.809 (.026)	.809 (.026)	.612 (.038)	.612 (.038)	.658 (.032)	.658 (.032)	.637 (.006)	.637 (.006)
HC(ANN) w/ disc.	.796 (.029)	.796 (.029)	.606 (.033)	.606 (.033)	.642 (.034)	.642 (.034)	.659 (.008)	.659 (.008)
IC-LR	.785 (.018)	.785 (.018)	.656 (.016)	.656 (.016)	.675 (.022)	.675 (.022)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.826 (.016)	.826 (.016)	.659 (.017)	.659 (.017)	.709 (.017)	**.709** (.017)	.667 (.005)	**.667** (.005)
IC-LR w/ neg.	**.846** (.015)	**.846** (.015)	**.688** (.016)	**.688** (.016)	.690 (.027)	.690 (.027)	.665 (.004)	.665 (.004)
Imp(LR)	.842 (.019)	.794 (.050)	.683 (.023)	.620 (.024)	.736 (.026)	.675 (.024)	.665 (.004)	.559 (.015)
Imp(LR) w/ disc.	.827 (.020)	.728 (.078)	.663 (.024)	.592 (.019)	.756 (.021)	.622 (.054)	**.709** (.004)	.651 (.014)
Imp(ANN)	.842 (.019)	.795 (.049)	.685 (.023)	.621 (.024)	**.764** (.018)	.688 (.016)	.679 (.005)	.559 (.017)
Imp(ANN) w/ disc.	.826 (.020)	.727 (.078)	.664 (.024)	.592 (.019)	.759 (.022)	.622 (.053)	**.709** (.004)	.652 (.013)
Imp(RF)	.806 (.022)	.737 (.070)	.634 (.028)	.584 (.029)	.735 (.019)	.625 (.025)	.694 (.005)	.649 (.009)
Imp(RF) w/ disc.	.826 (.022)	.726 (.080)	.645 (.025)	.582 (.023)	.748 (.022)	.615 (.053)	.708 (.005)	.648 (.012)
Imp(κNN)	.822 (.023)	.744 (.076)	.638 (.029)	.568 (.038)	.700 (.023)	.546 (.045)	.677 (.008)	.604 (.039)
Imp(κNN) w/ disc.	.811 (.027)	.703 (.094)	.615 (.039)	.548 (.036)	.695 (.036)	.517 (.047)	.671 (.017)	.571 (.046)
R-F(LR)	.830 (.021)	.786 (.052)	.633 (.026)	.597 (.036)	.704 (.023)	.639 (.050)	.670 (.005)	.633 (.006)
R-F(LR) w/ disc.	.824 (.023)	.768 (.050)	.652 (.026)	.615 (.035)	.736 (.020)	.667 (.040)	.696 (.005)	.662 (.006)
R-F(ANN)	.826 (.022)	.787 (.035)	.630 (.027)	.596 (.034)	.704 (.025)	.659 (.033)	.670 (.005)	.640 (.006)
R-F(ANN) w/ disc.	.820 (.022)	.779 (.034)	.635 (.029)	.596 (.034)	.707 (.024)	.643 (.036)	.696 (.004)	.663 (.006)
R-F(RF)	.806 (.022)	.752 (.047)	.618 (.027)	.592 (.039)	.700 (.019)	.565 (.029)	.686 (.005)	.646 (.009)
R-F(RF) w/ disc.	.822 (.022)	.779 (.039)	.640 (.027)	.613 (.030)	.704 (.024)	.608 (.053)	.696 (.005)	.662 (.005)
R-F(κNN)	.806 (.023)	.747 (.046)	.606 (.028)	.573 (.038)	.676 (.023)	.573 (.029)	.671 (.007)	.629 (.015)
R-F(κNN) w/ disc.	.805 (.025)	.745 (.048)	.627 (.029)	.588 (.043)	.687 (.024)	.575 (.043)	.668 (.009)	.624 (.019)

Table 29: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, unbalanced datasets, pre-selected 4 features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.788 (.024)	.788 (.024)	.611 (.033)	.611 (.033)	.645 (.033)	.645 (.033)	.622 (.008)	.622 (.008)
HC(LR) w/ disc.	.779 (.024)	.779 (.024)	.616 (.027)	.616 (.027)	.655 (.030)	.655 (.030)	.646 (.006)	.646 (.006)
HCApp(LR)	.788 (.023)	.788 (.023)	.608 (.039)	.608 (.039)	.644 (.035)	.644 (.035)	.618 (.013)	.618 (.013)
HCApp(LR) w/ disc.	.780 (.024)	.780 (.024)	.617 (.033)	.617 (.033)	.658 (.026)	.658 (.026)	.644 (.006)	.644 (.006)
HC(ANN)	.776 (.027)	.776 (.027)	.596 (.033)	.596 (.033)	.643 (.030)	.643 (.030)	.625 (.009)	.625 (.009)
HC(ANN) w/ disc.	.768 (.028)	.768 (.028)	.583 (.032)	.583 (.032)	.623 (.031)	.623 (.031)	.645 (.007)	.645 (.007)
IC-LR	.746 (.022)	.746 (.022)	.638 (.022)	.638 (.022)	.684 (.023)	.684 (.023)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.795 (.024)	.795 (.024)	.641 (.023)	.641 (.023)	.683 (.016)	.683 (.016)	.653 (.004)	**.653** (.004)
IC-LR w/ neg.	**.821** (.022)	**.821** (.022)	**.669** (.021)	**.669** (.021)	.701 (.027)	**.701** (.027)	.650 (.005)	.650 (.005)
Imp(LR)	.815 (.021)	.750 (.053)	.662 (.023)	.601 (.023)	.728 (.026)	.666 (.023)	.651 (.005)	.554 (.013)
Imp(LR) w/ disc.	.791 (.023)	.679 (.080)	.641 (.025)	.578 (.022)	.727 (.024)	.595 (.051)	**.696** (.005)	.637 (.011)
Imp(ANN)	.815 (.021)	.750 (.052)	.664 (.024)	.602 (.022)	**.749** (.020)	.676 (.016)	.660 (.006)	.553 (.014)
Imp(ANN) w/ disc.	.790 (.023)	.679 (.080)	.640 (.027)	.574 (.024)	.729 (.024)	.595 (.050)	**.696** (.005)	.637 (.011)
Imp(RF)	.778 (.024)	.698 (.067)	.618 (.028)	.572 (.028)	.710 (.019)	.616 (.022)	.686 (.005)	.643 (.007)
Imp(RF) w/ disc.	.791 (.023)	.679 (.080)	.627 (.027)	.569 (.024)	.720 (.024)	.590 (.050)	.694 (.005)	.635 (.010)
Imp(κNN)	.789 (.028)	.699 (.082)	.616 (.030)	.557 (.034)	.684 (.023)	.543 (.047)	.665 (.009)	.597 (.041)
Imp(κNN) w/ disc.	.772 (.037)	.655 (.090)	.595 (.039)	.541 (.036)	.669 (.042)	.513 (.047)	.658 (.015)	.559 (.043)
R-F(LR)	.792 (.024)	.767 (.030)	.609 (.027)	.589 (.033)	.682 (.025)	.643 (.032)	.652 (.005)	.624 (.006)
R-F(LR) w/ disc.	.785 (.027)	.751 (.037)	.633 (.028)	.605 (.029)	.707 (.020)	.655 (.026)	.673 (.005)	.645 (.006)
R-F(ANN)	.786 (.024)	.759 (.031)	.604 (.028)	.583 (.032)	.680 (.027)	.642 (.030)	.652 (.005)	.628 (.006)
R-F(ANN) w/ disc.	.782 (.028)	.749 (.034)	.610 (.032)	.575 (.032)	.671 (.024)	.620 (.030)	.673 (.005)	.645 (.006)
R-F(RF)	.771 (.025)	.732 (.036)	.599 (.028)	.583 (.033)	.667 (.019)	.574 (.022)	.667 (.005)	.638 (.007)
R-F(RF) w/ disc.	.787 (.027)	.754 (.033)	.625 (.028)	.601 (.029)	.684 (.021)	.624 (.032)	.673 (.005)	.645 (.006)
R-F(κNN)	.769 (.025)	.739 (.032)	.583 (.028)	.573 (.036)	.655 (.021)	.586 (.023)	.651 (.007)	.618 (.013)
R-F(κNN) w/ disc.	.770 (.029)	.736 (.038)	.608 (.030)	.576 (.040)	.657 (.024)	.576 (.035)	.648 (.009)	.612 (.018)

Table 30: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.859 (.015)	.859 (.015)	.671 (.030)	.671 (.030)	.726 (.028)	.726 (.028)	.686 (.006)	.686 (.006)
HC(LR) w/ disc.	.855 (.017)	.855 (.017)	.674 (.027)	.674 (.027)	.741 (.031)	.741 (.031)	.695 (.005)	.695 (.005)
HCApp(LR)	.860 (.015)	.860 (.015)	.673 (.032)	.673 (.032)	.709 (.032)	.709 (.032)	.647 (.018)	.647 (.018)
HCApp(LR) w/ disc.	.851 (.028)	.851 (.028)	.653 (.051)	.653 (.051)	.729 (.034)	.729 (.034)	.634 (.006)	.634 (.006)
HC(ANN)	.859 (.014)	.859 (.014)	.679 (.027)	.679 (.027)	.718 (.026)	.718 (.026)	.687 (.006)	.687 (.006)
HC(ANN) w/ disc.	.847 (.016)	.847 (.016)	.676 (.025)	.676 (.025)	.739 (.030)	.739 (.030)	.696 (.005)	.696 (.005)
Mincut	.743 (.024)	.743 (.024)	.564 (.047)	.564 (.047)	.500 (.017)	.500 (.017)	.700 (.005)	.700 (.005)
Mincut w/ disc.	.850 (.018)	.850 (.018)	.650 (.036)	.650 (.036)	.739 (.031)	.739 (.031)	**.701** (.005)	**.701** (.005)
IC-LR	.861 (.014)	.861 (.014)	.671 (.027)	.671 (.027)	.666 (.036)	.666 (.036)	.499 (.004)	.499 (.004)
IC-LR w/ disc.	.858 (.015)	.858 (.015)	.675 (.025)	.675 (.025)	**.742** (.030)	**.742** (.030)	.695 (.005)	.695 (.005)
IC-LR w/ neg.	**.862** (.015)	**.862** (.015)	**.682** (.025)	**.682** (.025)	.675 (.038)	.675 (.038)	.686 (.006)	.686 (.006)
Maj	.746 (.025)	.746 (.025)	.565 (.047)	.565 (.047)	.500 (.017)	.500 (.017)	**.701** (.005)	**.701** (.005)
Maj w/ disc.	.854 (.017)	.854 (.017)	.648 (.036)	.648 (.036)	.739 (.032)	.739 (.032)	**.701** (.005)	**.701** (.005)
Imp(LR)	.860 (.014)	.859 (.031)	**.682** (.026)	.580 (.090)	.725 (.026)	.643 (.069)	.688 (.006)	.606 (.079)
Imp(LR) w/ disc.	.858 (.015)	.853 (.036)	.676 (.024)	.583 (.090)	**.742** (.031)	.678 (.100)	.695 (.005)	.499 (.004)
Imp(ANN)	.859 (.015)	.857 (.031)	.681 (.025)	.580 (.090)	.719 (.026)	.659 (.045)	.698 (.005)	.578 (.024)
Imp(ANN) w/ disc.	.847 (.016)	.827 (.042)	.676 (.024)	.583 (.090)	.739 (.030)	.675 (.096)	.696 (.006)	.504 (.018)
Imp(RF)	.819 (.019)	.700 (.103)	.645 (.027)	.538 (.062)	.728 (.032)	.654 (.085)	**.701** (.005)	.527 (.028)
Imp(RF) w/ disc.	.854 (.017)	.844 (.039)	.661 (.028)	.577 (.087)	.739 (.031)	.666 (.101)	**.701** (.005)	.519 (.033)
Imp(κNN)	.853 (.018)	.803 (.070)	.657 (.030)	.563 (.079)	.717 (.029)	.617 (.056)	.675 (.038)	.516 (.025)
Imp(κNN) w/ disc.	.846 (.023)	.809 (.084)	.660 (.034)	.568 (.084)	.728 (.047)	.662 (.100)	.678 (.054)	.490 (.050)
R-F(LR)	.860 (.014)	.860 (.014)	**.682** (.026)	**.682** (.026)	.725 (.026)	.725 (.026)	.688 (.006)	.688 (.006)
R-F(LR) w/ disc.	.858 (.015)	.858 (.015)	.676 (.024)	.676 (.024)	**.742** (.031)	**.742** (.031)	.695 (.005)	.695 (.005)
R-F(ANN)	.859 (.014)	.858 (.015)	.681 (.025)	.681 (.025)	.719 (.027)	.719 (.029)	.698 (.005)	.698 (.006)
R-F(ANN) w/ disc.	.847 (.017)	.847 (.017)	.676 (.024)	.676 (.024)	.740 (.030)	.738 (.030)	.695 (.005)	.696 (.005)
R-F(RF)	.819 (.019)	.820 (.019)	.644 (.027)	.644 (.026)	.728 (.032)	.729 (.031)	**.701** (.005)	**.701** (.005)
R-F(RF) w/ disc.	.854 (.018)	.854 (.017)	.662 (.028)	.661 (.028)	.740 (.031)	.740 (.032)	**.701** (.005)	**.701** (.005)
R-F(κNN)	.852 (.017)	.851 (.017)	.659 (.031)	.659 (.031)	.717 (.028)	.716 (.028)	.680 (.031)	.670 (.044)
R-F(κNN) w/ disc.	.847 (.021)	.848 (.023)	.660 (.034)	.660 (.035)	.726 (.050)	.727 (.050)	.676 (.056)	.679 (.053)

Table 31: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.817 (.020)	.817 (.020)	.645 (.029)	.645 (.029)	.690 (.029)	.690 (.029)	.665 (.006)	.665 (.006)
HC(LR) w/ disc.	.821 (.019)	.821 (.019)	.646 (.028)	.646 (.028)	.714 (.030)	.714 (.030)	.672 (.005)	.672 (.005)
HCApp(LR)	.817 (.021)	.817 (.021)	.652 (.028)	.652 (.028)	.693 (.029)	.693 (.029)	.665 (.006)	.665 (.006)
HCApp(LR) w/ disc.	.817 (.020)	.817 (.020)	.645 (.038)	.645 (.038)	.716 (.029)	.716 (.029)	.670 (.009)	.670 (.009)
HC(ANN)	.804 (.024)	.804 (.024)	.633 (.032)	.633 (.032)	.672 (.032)	.672 (.032)	.665 (.006)	.665 (.006)
HC(ANN) w/ disc.	.798 (.024)	.798 (.024)	.628 (.033)	.628 (.033)	.701 (.031)	.701 (.031)	.672 (.006)	.672 (.006)
Mincut	.768 (.037)	.768 (.037)	.573 (.039)	.573 (.039)	.500 (.018)	.500 (.018)	.673 (.006)	.673 (.006)
Mincut w/ disc.	.814 (.020)	.814 (.020)	.629 (.033)	.629 (.033)	.713 (.028)	.713 (.028)	.672 (.006)	.672 (.006)
IC-LR	.825 (.018)	.825 (.018)	.649 (.033)	.649 (.033)	.664 (.035)	.664 (.035)	.499 (.005)	.499 (.005)
IC-LR w/ disc.	.826 (.017)	**.826** (.017)	.653 (.029)	.653 (.029)	.719 (.030)	**.719** (.030)	.670 (.006)	.670 (.006)
IC-LR w/ neg.	.824 (.018)	.824 (.018)	.661 (.025)	**.661** (.025)	.667 (.034)	.667 (.034)	.680 (.007)	**.680** (.007)
Maj	.685 (.029)	.717 (.084)	.544 (.038)	.544 (.046)	.500 (.018)	.500 (.018)	.687 (.006)	.566 (.031)
Maj w/ disc.	.818 (.021)	.678 (.098)	.615 (.038)	.543 (.045)	.721 (.028)	.616 (.071)	.695 (.005)	.565 (.034)
Imp(LR)	.822 (.019)	.822 (.035)	**.662** (.027)	.574 (.082)	.722 (.026)	.658 (.062)	.678 (.006)	.616 (.069)
Imp(LR) w/ disc.	.824 (.017)	.797 (.048)	.654 (.026)	.576 (.082)	**.731** (.027)	.600 (.103)	.691 (.005)	.512 (.039)
Imp(ANN)	.823 (.017)	.819 (.035)	**.662** (.026)	.575 (.082)	.719 (.026)	.660 (.051)	.694 (.006)	.556 (.007)
Imp(ANN) w/ disc.	.824 (.017)	.784 (.051)	.655 (.025)	.574 (.083)	.729 (.027)	.615 (.096)	.693 (.006)	.563 (.028)
Imp(RF)	.793 (.020)	.637 (.113)	.625 (.029)	.540 (.060)	.720 (.028)	.609 (.073)	.689 (.006)	.541 (.032)
Imp(RF) w/ disc.	.821 (.020)	.775 (.078)	.644 (.029)	.565 (.080)	.730 (.027)	.592 (.098)	**.697** (.005)	.551 (.031)
Imp(κNN)	.815 (.019)	.740 (.094)	.636 (.030)	.556 (.070)	.710 (.029)	.618 (.060)	.669 (.027)	.553 (.030)
Imp(κNN) w/ disc.	.813 (.038)	.756 (.107)	.635 (.034)	.560 (.073)	.711 (.049)	.596 (.098)	.661 (.062)	.557 (.051)
R-F(LR)	**.831** (.019)	.539 (.070)	.640 (.029)	.506 (.029)	.699 (.027)	.509 (.027)	.676 (.009)	.528 (.038)
R-F(LR) w/ disc.	.826 (.019)	.537 (.071)	.640 (.029)	.513 (.037)	.721 (.029)	.518 (.045)	.690 (.006)	.525 (.038)
R-F(ANN)	.828 (.019)	.537 (.067)	.637 (.029)	.506 (.025)	.694 (.031)	.509 (.025)	.681 (.008)	.518 (.032)
R-F(ANN) w/ disc.	.818 (.019)	.536 (.065)	.631 (.038)	.510 (.032)	.720 (.028)	.519 (.044)	.691 (.006)	.514 (.025)
R-F(RF)	.793 (.022)	.534 (.060)	.614 (.028)	.506 (.026)	.703 (.028)	.511 (.028)	.688 (.006)	.529 (.034)
R-F(RF) w/ disc.	.822 (.021)	.536 (.067)	.629 (.031)	.511 (.034)	.720 (.028)	.520 (.045)	.694 (.005)	.525 (.035)
R-F(κNN)	.811 (.025)	.529 (.068)	.611 (.029)	.506 (.027)	.682 (.031)	.509 (.031)	.672 (.012)	.521 (.032)
R-F(κNN) w/ disc.	.807 (.025)	.527 (.064)	.624 (.034)	.514 (.038)	.706 (.036)	.510 (.033)	.656 (.054)	.520 (.035)

Table 32: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.794 (.023)	.794 (.023)	.624 (.032)	.624 (.032)	.677 (.031)	.677 (.031)	.648 (.005)	.648 (.005)
HC(LR) w/ disc.	.795 (.022)	.795 (.022)	.631 (.031)	.631 (.031)	.693 (.032)	.693 (.032)	.649 (.005)	.649 (.005)
HCApp(LR)	.793 (.019)	.793 (.019)	.631 (.032)	.631 (.032)	.680 (.030)	.680 (.030)	.648 (.005)	.648 (.005)
HCApp(LR) w/ disc.	.794 (.021)	.794 (.021)	.629 (.042)	.629 (.042)	.692 (.032)	.692 (.032)	.648 (.005)	.648 (.005)
HC(ANN)	.778 (.025)	.778 (.025)	.614 (.031)	.614 (.031)	.654 (.033)	.654 (.033)	.648 (.005)	.648 (.005)
HC(ANN) w/ disc.	.771 (.025)	.771 (.025)	.608 (.036)	.608 (.036)	.679 (.036)	.679 (.036)	.647 (.005)	.647 (.005)
Mincut	.770 (.030)	.770 (.030)	.576 (.039)	.576 (.039)	.501 (.018)	.501 (.018)	.651 (.006)	.651 (.006)
Mincut w/ disc.	.793 (.022)	.793 (.022)	.620 (.030)	.620 (.030)	.692 (.032)	.692 (.032)	.648 (.006)	.648 (.006)
IC-LR	.789 (.019)	.789 (.019)	.632 (.034)	.632 (.034)	.665 (.043)	.665 (.043)	.498 (.004)	.498 (.004)
IC-LR w/ disc.	.801 (.020)	**.801** (.020)	.637 (.030)	.637 (.030)	.696 (.029)	**.696** (.029)	.646 (.007)	.646 (.007)
IC-LR w/ neg.	.794 (.020)	.794 (.020)	**.659** (.029)	**.659** (.029)	.655 (.042)	.655 (.042)	.671 (.008)	**.671** (.008)
Maj	.664 (.030)	.686 (.087)	.537 (.033)	.533 (.046)	.500 (.017)	.501 (.021)	.678 (.006)	.564 (.032)
Maj w/ disc.	.795 (.025)	.637 (.089)	.604 (.037)	.534 (.044)	.706 (.029)	.576 (.061)	.688 (.005)	.568 (.032)
Imp(LR)	.795 (.019)	.790 (.026)	.649 (.028)	.571 (.073)	.713 (.028)	.654 (.055)	.667 (.008)	.622 (.053)
Imp(LR) w/ disc.	.800 (.019)	.767 (.041)	.638 (.028)	.571 (.075)	**.719** (.027)	.569 (.090)	.686 (.005)	.504 (.022)
Imp(ANN)	.799 (.019)	.789 (.035)	.649 (.028)	.570 (.073)	.714 (.026)	.655 (.049)	.685 (.005)	.536 (.005)
Imp(ANN) w/ disc.	.801 (.019)	.752 (.069)	.638 (.030)	.572 (.075)	.717 (.027)	.576 (.083)	**.689** (.005)	.538 (.016)
Imp(RF)	.771 (.025)	.638 (.096)	.613 (.032)	.541 (.056)	.709 (.028)	.589 (.068)	.679 (.005)	.553 (.031)
Imp(RF) w/ disc.	.797 (.020)	.750 (.073)	.628 (.033)	.568 (.073)	.716 (.026)	.566 (.082)	**.689** (.005)	.540 (.025)
Imp(κNN)	.789 (.021)	.720 (.086)	.624 (.031)	.553 (.064)	.697 (.030)	.610 (.058)	.659 (.016)	.556 (.035)
Imp(κNN) w/ disc.	.791 (.024)	.714 (.111)	.613 (.040)	.554 (.067)	.697 (.046)	.563 (.086)	.652 (.045)	.550 (.042)
R-F(LR)	**.809** (.022)	.548 (.057)	.617 (.029)	.505 (.025)	.681 (.027)	.515 (.032)	.664 (.008)	.588 (.041)
R-F(LR) w/ disc.	.801 (.024)	.541 (.059)	.620 (.032)	.514 (.038)	.704 (.028)	.520 (.041)	.685 (.005)	.584 (.030)
R-F(ANN)	.807 (.022)	.546 (.057)	.615 (.029)	.504 (.024)	.677 (.029)	.514 (.031)	.667 (.007)	.549 (.046)
R-F(ANN) w/ disc.	.796 (.026)	.550 (.064)	.616 (.034)	.514 (.040)	.703 (.028)	.523 (.043)	.683 (.006)	.548 (.044)
R-F(RF)	.776 (.026)	.543 (.046)	.597 (.031)	.504 (.025)	.685 (.027)	.521 (.033)	.678 (.005)	.579 (.023)
R-F(RF) w/ disc.	.803 (.024)	.560 (.069)	.614 (.032)	.515 (.038)	.703 (.030)	.527 (.045)	.687 (.005)	.587 (.023)
R-F(κNN)	.780 (.023)	.538 (.057)	.591 (.033)	.501 (.023)	.666 (.027)	.513 (.031)	.659 (.018)	.583 (.028)
R-F(κNN) w/ disc.	.777 (.029)	.536 (.055)	.608 (.036)	.514 (.039)	.679 (.035)	.515 (.033)	.658 (.029)	.592 (.026)

Table 33: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.770 (.022)	.770 (.022)	.608 (.032)	.608 (.032)	.668 (.028)	.668 (.028)	.636 (.006)	.636 (.006)
HC(LR) w/ disc.	.769 (.023)	.769 (.023)	.616 (.032)	.616 (.032)	.680 (.030)	.680 (.030)	.637 (.006)	.637 (.006)
HCApp(LR)	.772 (.022)	.772 (.022)	.614 (.033)	.614 (.033)	.670 (.027)	.670 (.027)	.635 (.006)	.635 (.006)
HCApp(LR) w/ disc.	.769 (.022)	.769 (.022)	.616 (.041)	.616 (.041)	.681 (.030)	.681 (.030)	.637 (.006)	.637 (.006)
HC(ANN)	.756 (.028)	.756 (.028)	.599 (.034)	.599 (.034)	.644 (.032)	.644 (.032)	.635 (.006)	.635 (.006)
HC(ANN) w/ disc.	.748 (.027)	.748 (.027)	.590 (.036)	.590 (.036)	.664 (.033)	.664 (.033)	.636 (.006)	.636 (.006)
Mincut	.756 (.026)	.756 (.026)	.566 (.039)	.566 (.039)	.505 (.016)	.505 (.016)	.636 (.006)	.636 (.006)
Mincut w/ disc.	.767 (.022)	.767 (.022)	.606 (.034)	.606 (.034)	.682 (.033)	**.682** (.033)	.637 (.006)	.637 (.006)
IC-LR	.760 (.026)	.760 (.026)	.622 (.037)	.622 (.037)	.667 (.044)	.667 (.044)	.499 (.009)	.499 (.009)
IC-LR w/ disc.	.774 (.025)	**.774** (.025)	.621 (.037)	.621 (.037)	.682 (.030)	**.682** (.030)	.633 (.007)	.633 (.007)
IC-LR w/ neg.	.770 (.027)	.770 (.027)	.631 (.033)	**.631** (.033)	.671 (.034)	.671 (.034)	.660 (.010)	**.660** (.010)
Maj	.663 (.040)	.649 (.099)	.530 (.034)	.526 (.037)	.503 (.015)	.501 (.019)	.666 (.008)	.553 (.039)
Maj w/ disc.	.770 (.028)	.608 (.085)	.588 (.037)	.533 (.045)	.699 (.027)	.568 (.070)	.679 (.005)	.556 (.039)
Imp(LR)	.773 (.023)	.758 (.038)	**.633** (.031)	.567 (.069)	.708 (.026)	.654 (.057)	.658 (.010)	.599 (.059)
Imp(LR) w/ disc.	.772 (.023)	.719 (.094)	.624 (.033)	.569 (.071)	.710 (.027)	.567 (.097)	.679 (.006)	.505 (.026)
Imp(ANN)	.777 (.022)	.750 (.047)	.632 (.031)	.567 (.069)	.709 (.025)	.651 (.059)	.677 (.005)	.520 (.005)
Imp(ANN) w/ disc.	.771 (.023)	.702 (.105)	.622 (.036)	.571 (.071)	.709 (.029)	.585 (.093)	**.680** (.005)	.527 (.027)
Imp(RF)	.751 (.023)	.630 (.101)	.600 (.032)	.543 (.055)	.697 (.026)	.563 (.066)	.668 (.006)	.549 (.038)
Imp(RF) w/ disc.	.769 (.024)	.694 (.112)	.616 (.036)	.565 (.068)	**.711** (.028)	.564 (.089)	**.680** (.005)	.533 (.035)
Imp(κNN)	.763 (.027)	.666 (.105)	.606 (.035)	.546 (.058)	.691 (.029)	.594 (.077)	.648 (.013)	.561 (.042)
Imp(κNN) w/ disc.	.762 (.034)	.694 (.113)	.601 (.042)	.552 (.061)	.694 (.040)	.579 (.091)	.652 (.030)	.551 (.048)
R-F(LR)	**.788** (.025)	.576 (.071)	.598 (.033)	.514 (.030)	.669 (.025)	.534 (.038)	.653 (.011)	.609 (.018)
R-F(LR) w/ disc.	.776 (.027)	.569 (.074)	.606 (.032)	.522 (.040)	.695 (.028)	.556 (.060)	.676 (.005)	.596 (.011)
R-F(ANN)	.783 (.026)	.573 (.069)	.596 (.031)	.514 (.029)	.666 (.029)	.533 (.043)	.655 (.009)	.568 (.043)
R-F(ANN) w/ disc.	.773 (.028)	.578 (.076)	.599 (.034)	.517 (.035)	.692 (.028)	.556 (.056)	.674 (.006)	.560 (.042)
R-F(RF)	.764 (.026)	.572 (.057)	.587 (.031)	.513 (.031)	.670 (.027)	.550 (.039)	.668 (.006)	.590 (.012)
R-F(RF) w/ disc.	.777 (.028)	.588 (.075)	.599 (.033)	.525 (.040)	.696 (.029)	.568 (.063)	.678 (.005)	.595 (.011)
R-F(κNN)	.768 (.029)	.570 (.068)	.575 (.033)	.508 (.027)	.655 (.028)	.537 (.043)	.651 (.013)	.594 (.021)
R-F(κNN) w/ disc.	.757 (.031)	.579 (.079)	.592 (.036)	.522 (.043)	.674 (.032)	.557 (.057)	.648 (.021)	.595 (.022)

Table 34: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia Tru.	Australia Str.	Germany Tru.	Germany Str.	Poland Tru.	Poland Str.	Taiwan Tru.	Taiwan Str.
HC(LR)	.751 (.024)	.751 (.024)	.598 (.034)	.598 (.034)	.656 (.030)	.656 (.030)	.627 (.006)	.627 (.006)
HC(LR) w/ disc.	.747 (.026)	.747 (.026)	.604 (.033)	.604 (.033)	.667 (.030)	.667 (.030)	.627 (.006)	.627 (.006)
HCApp(LR)	.753 (.024)	**.753** (.024)	.602 (.033)	.602 (.033)	.656 (.030)	.656 (.030)	.627 (.006)	.627 (.006)
HCApp(LR) w/ disc.	.747 (.026)	.747 (.026)	.601 (.039)	.601 (.039)	.670 (.029)	.670 (.029)	.627 (.006)	.627 (.006)
HC(ANN)	.734 (.027)	.734 (.027)	.585 (.036)	.585 (.036)	.626 (.038)	.626 (.038)	.626 (.007)	.626 (.007)
HC(ANN) w/ disc.	.727 (.029)	.727 (.029)	.578 (.035)	.578 (.035)	.654 (.035)	.654 (.035)	.626 (.006)	.626 (.006)
Mincut	.726 (.028)	.726 (.028)	.566 (.035)	.566 (.035)	.502 (.022)	.502 (.022)	.624 (.007)	.624 (.007)
Mincut w/ disc.	.744 (.027)	.744 (.027)	.595 (.034)	.595 (.034)	.667 (.031)	.667 (.031)	.625 (.007)	.625 (.007)
IC-LR	.734 (.030)	.734 (.030)	.606 (.038)	.606 (.038)	.646 (.052)	.646 (.052)	.498 (.007)	.498 (.007)
IC-LR w/ disc.	.745 (.026)	.745 (.026)	.608 (.036)	**.608** (.036)	.659 (.031)	.659 (.031)	.617 (.006)	.617 (.006)
IC-LR w/ neg.	.751 (.026)	.751 (.026)	.608 (.039)	**.608** (.039)	.677 (.042)	**.677** (.042)	.652 (.012)	**.652** (.012)
Maj	.651 (.042)	.624 (.104)	.531 (.030)	.525 (.036)	.498 (.020)	.498 (.019)	.657 (.007)	.555 (.046)
Maj w/ disc.	.752 (.030)	.604 (.097)	.574 (.036)	.530 (.043)	.685 (.028)	.573 (.073)	.670 (.004)	.556 (.046)
Imp(LR)	.755 (.023)	.726 (.053)	**.619** (.030)	.568 (.062)	.697 (.027)	.652 (.054)	.649 (.013)	.582 (.062)
Imp(LR) w/ disc.	.740 (.027)	.671 (.107)	.611 (.031)	.567 (.063)	.696 (.027)	.558 (.087)	.670 (.005)	.500 (.004)
Imp(ANN)	.755 (.021)	.718 (.056)	**.619** (.030)	.569 (.062)	**.700** (.026)	.649 (.058)	.667 (.005)	.510 (.005)
Imp(ANN) w/ disc.	.741 (.026)	.657 (.115)	.609 (.032)	.566 (.063)	.691 (.031)	.571 (.086)	**.671** (.005)	.511 (.012)
Imp(RF)	.725 (.026)	.623 (.101)	.590 (.032)	.549 (.050)	.681 (.024)	.563 (.065)	.659 (.006)	.554 (.045)
Imp(RF) w/ disc.	.742 (.026)	.652 (.117)	.604 (.031)	.561 (.061)	.695 (.027)	.562 (.083)	.670 (.005)	.520 (.032)
Imp(κNN)	.738 (.028)	.625 (.110)	.594 (.033)	.546 (.054)	.680 (.026)	.584 (.073)	.637 (.015)	.551 (.047)
Imp(κNN) w/ disc.	.727 (.035)	.640 (.116)	.587 (.041)	.553 (.058)	.680 (.044)	.564 (.084)	.635 (.031)	.545 (.052)
R-F(LR)	**.766** (.027)	.625 (.061)	.582 (.030)	.526 (.031)	.653 (.027)	.572 (.045)	.641 (.010)	.614 (.010)
R-F(LR) w/ disc.	.758 (.025)	.622 (.074)	.588 (.034)	.526 (.040)	.684 (.027)	.609 (.055)	.668 (.005)	.599 (.006)
R-F(ANN)	.762 (.029)	.614 (.065)	.578 (.029)	.524 (.031)	.649 (.031)	.552 (.048)	.643 (.010)	.597 (.031)
R-F(ANN) w/ disc.	.753 (.027)	.624 (.079)	.580 (.032)	.521 (.039)	.678 (.029)	.585 (.060)	.667 (.006)	.593 (.028)
R-F(RF)	.743 (.028)	.625 (.056)	.574 (.030)	.524 (.030)	.654 (.026)	.576 (.040)	.658 (.006)	.596 (.009)
R-F(RF) w/ disc.	.762 (.025)	.650 (.071)	.586 (.033)	.527 (.040)	.685 (.026)	.612 (.053)	.669 (.005)	.599 (.006)
R-F(κNN)	.749 (.029)	.621 (.063)	.565 (.031)	.520 (.032)	.652 (.026)	.584 (.045)	.641 (.012)	.601 (.016)
R-F(κNN) w/ disc.	.747 (.028)	.635 (.081)	.574 (.036)	.526 (.038)	.667 (.032)	.600 (.054)	.642 (.018)	.599 (.014)

Table 35: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.723 (.026)	.723 (.026)	.580 (.034)	.580 (.034)	.642 (.031)	.642 (.031)	.616 (.006)	.616 (.006)
HC(LR) w/ disc.	.718 (.026)	.718 (.026)	.582 (.034)	.582 (.034)	.657 (.032)	.657 (.032)	.617 (.006)	.617 (.006)
HCApp(LR)	.725 (.025)	.725 (.025)	.582 (.034)	.582 (.034)	.644 (.028)	.644 (.028)	.616 (.006)	.616 (.006)
HCApp(LR) w/ disc.	.717 (.027)	.717 (.027)	.579 (.038)	.579 (.038)	.659 (.031)	**.659** (.031)	.617 (.006)	.617 (.006)
HC(ANN)	.704 (.030)	.704 (.030)	.567 (.034)	.567 (.034)	.618 (.033)	.618 (.033)	.616 (.006)	.616 (.006)
HC(ANN) w/ disc.	.701 (.029)	.701 (.029)	.558 (.040)	.558 (.040)	.641 (.029)	.641 (.029)	.616 (.006)	.616 (.006)
Mincut	.705 (.031)	.705 (.031)	.559 (.038)	.559 (.038)	.506 (.022)	.506 (.022)	.613 (.008)	.613 (.008)
Mincut w/ disc.	.714 (.025)	.714 (.025)	.578 (.036)	.578 (.036)	.655 (.031)	.655 (.031)	.614 (.009)	.614 (.009)
IC-LR	.711 (.031)	.711 (.031)	.588 (.037)	**.588** (.037)	.636 (.045)	.636 (.045)	.497 (.005)	.497 (.005)
IC-LR w/ disc.	.717 (.023)	.717 (.023)	.588 (.036)	**.588** (.036)	.638 (.031)	.638 (.031)	.605 (.008)	.605 (.008)
IC-LR w/ neg.	.726 (.026)	**.726** (.026)	.587 (.037)	.587 (.037)	.655 (.038)	.655 (.038)	.638 (.015)	**.638** (.015)
Maj	.653 (.042)	.621 (.099)	.529 (.030)	.520 (.035)	.503 (.019)	.503 (.022)	.647 (.006)	.553 (.049)
Maj w/ disc.	.732 (.030)	.607 (.093)	.560 (.034)	.525 (.041)	.681 (.029)	.581 (.073)	.657 (.005)	.554 (.049)
Imp(LR)	.734 (.024)	.707 (.043)	**.598** (.033)	.553 (.054)	.684 (.026)	.640 (.058)	.634 (.013)	.584 (.053)
Imp(LR) w/ disc.	.706 (.029)	.641 (.098)	.589 (.035)	.551 (.055)	.681 (.029)	.543 (.076)	**.659** (.005)	.503 (.019)
Imp(ANN)	.733 (.025)	.700 (.048)	**.598** (.032)	.553 (.054)	**.686** (.026)	.634 (.064)	.656 (.005)	.505 (.004)
Imp(ANN) w/ disc.	.704 (.030)	.627 (.104)	.581 (.041)	.545 (.057)	.680 (.031)	.556 (.077)	**.659** (.006)	.515 (.032)
Imp(RF)	.706 (.030)	.627 (.094)	.576 (.029)	.538 (.046)	.667 (.029)	.565 (.061)	.648 (.006)	.551 (.048)
Imp(RF) w/ disc.	.704 (.029)	.628 (.103)	.583 (.035)	.546 (.053)	.681 (.028)	.545 (.074)	.657 (.005)	.516 (.034)
Imp(κNN)	.711 (.032)	.616 (.102)	.575 (.035)	.538 (.044)	.666 (.027)	.577 (.067)	.627 (.010)	.555 (.047)
Imp(κNN) w/ disc.	.690 (.037)	.612 (.103)	.570 (.041)	.535 (.049)	.655 (.054)	.555 (.076)	.626 (.024)	.545 (.047)
R-F(LR)	**.742** (.028)	.650 (.056)	.565 (.031)	.531 (.031)	.643 (.029)	.590 (.037)	.627 (.010)	.611 (.009)
R-F(LR) w/ disc.	.736 (.029)	.655 (.064)	.569 (.033)	.528 (.039)	.675 (.031)	.626 (.047)	.657 (.005)	.602 (.006)
R-F(ANN)	.733 (.030)	.633 (.059)	.563 (.031)	.529 (.031)	.634 (.034)	.566 (.044)	.626 (.012)	.603 (.017)
R-F(ANN) w/ disc.	.730 (.028)	.633 (.066)	.559 (.035)	.518 (.037)	.669 (.034)	.604 (.060)	.655 (.008)	.597 (.022)
R-F(RF)	.725 (.028)	.645 (.045)	.560 (.031)	.530 (.032)	.641 (.027)	.581 (.031)	.648 (.006)	.599 (.007)
R-F(RF) w/ disc.	**.742** (.027)	.669 (.056)	.570 (.034)	.529 (.039)	.677 (.030)	.630 (.046)	.657 (.005)	.602 (.006)
R-F(κNN)	.729 (.031)	.652 (.052)	.551 (.033)	.528 (.029)	.641 (.030)	.602 (.036)	.626 (.010)	.599 (.012)
R-F(κNN) w/ disc.	.726 (.028)	.656 (.064)	.560 (.032)	.529 (.037)	.664 (.030)	.623 (.043)	.626 (.015)	.599 (.011)

Table 36: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.904 (.013)	.904 (.013)	.725 (.033)	.725 (.033)	.781 (.028)	.781 (.028)	.696 (.007)	.696 (.007)
HC(LR) w/ disc.	.909 (.013)	**.909** (.013)	.724 (.027)	.724 (.027)	.787 (.032)	.787 (.032)	.729 (.006)	.729 (.006)
HCApp(LR)	.908 (.013)	.908 (.013)	.723 (.032)	.723 (.032)	.763 (.032)	.763 (.032)	.646 (.013)	.646 (.013)
HCApp(LR) w/ disc.	.909 (.022)	**.909** (.022)	.699 (.056)	.699 (.056)	.758 (.037)	.758 (.037)	.644 (.009)	.644 (.009)
HC(ANN)	.906 (.013)	.906 (.013)	.737 (.028)	.737 (.028)	.778 (.028)	.778 (.028)	.700 (.009)	.700 (.009)
HC(ANN) w/ disc.	.907 (.013)	.907 (.013)	.727 (.025)	.727 (.025)	.788 (.031)	.788 (.031)	.731 (.005)	.731 (.005)
IC-LR	.900 (.013)	.900 (.013)	.728 (.025)	.728 (.025)	.729 (.037)	.729 (.037)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.909 (.013)	**.909** (.013)	.730 (.026)	.730 (.026)	.791 (.032)	.791 (.032)	.730 (.005)	.730 (.005)
IC-LR w/ neg.	.904 (.014)	.904 (.014)	**.744** (.027)	**.744** (.027)	.741 (.041)	.741 (.041)	.702 (.007)	.702 (.007)
Imp(LR)	.904 (.014)	.903 (.015)	.741 (.027)	.684 (.063)	.786 (.027)	.734 (.050)	.703 (.005)	.665 (.009)
Imp(LR) w/ disc.	.909 (.013)	.896 (.021)	.729 (.026)	.662 (.072)	.791 (.032)	.728 (.085)	.730 (.005)	.500 (.000)
Imp(ANN)	.905 (.013)	.902 (.014)	.741 (.027)	.684 (.062)	.779 (.027)	.749 (.027)	.731 (.009)	.680 (.014)
Imp(ANN) w/ disc.	.908 (.013)	.896 (.021)	.730 (.025)	.664 (.071)	.790 (.031)	.738 (.060)	.730 (.005)	.587 (.003)
Imp(RF)	.884 (.015)	.870 (.041)	.698 (.028)	.641 (.076)	**.794** (.033)	.771 (.042)	**.734** (.005)	.580 (.023)
Imp(RF) w/ disc.	.909 (.014)	.896 (.044)	.702 (.027)	.640 (.078)	.786 (.032)	.717 (.094)	.730 (.005)	.527 (.077)
Imp(κNN)	.900 (.016)	.869 (.032)	.703 (.031)	.620 (.094)	.773 (.032)	.738 (.040)	.717 (.015)	.628 (.041)
Imp(κNN) w/ disc.	.894 (.022)	.878 (.054)	.704 (.031)	.618 (.094)	.770 (.038)	.703 (.102)	.710 (.027)	.498 (.086)
R-F(LR)	.904 (.014)	.904 (.014)	.741 (.027)	.741 (.027)	.786 (.027)	.786 (.027)	.703 (.005)	.703 (.005)
R-F(LR) w/ disc.	.909 (.013)	**.909** (.013)	.729 (.026)	.729 (.026)	.791 (.032)	.791 (.032)	.730 (.005)	.730 (.005)
R-F(ANN)	.905 (.013)	.905 (.013)	.741 (.027)	.741 (.027)	.779 (.027)	.780 (.027)	.730 (.009)	.731 (.010)
R-F(ANN) w/ disc.	.908 (.013)	.908 (.013)	.730 (.025)	.730 (.025)	.790 (.031)	.790 (.031)	.730 (.005)	.730 (.005)
R-F(RF)	.885 (.015)	.884 (.015)	.698 (.028)	.698 (.028)	**.794** (.033)	**.794** (.034)	**.734** (.005)	**.734** (.005)
R-F(RF) w/ disc.	**.910** (.014)	.909 (.014)	.703 (.028)	.703 (.028)	.786 (.032)	.785 (.032)	.730 (.005)	.730 (.005)
R-F(κNN)	.901 (.015)	.900 (.015)	.703 (.031)	.702 (.033)	.772 (.030)	.772 (.030)	.718 (.017)	.715 (.018)
R-F(κNN) w/ disc.	.893 (.022)	.894 (.022)	.704 (.031)	.702 (.033)	.770 (.038)	.770 (.038)	.711 (.021)	.712 (.020)

Table 37: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.892 (.015)	.892 (.015)	.685 (.033)	.685 (.033)	.750 (.030)	.750 (.030)	.675 (.007)	.675 (.007)
HC(LR) w/ disc.	.890 (.017)	.890 (.017)	.686 (.028)	.686 (.028)	.764 (.031)	.764 (.031)	.705 (.006)	.705 (.006)
HCApp(LR)	.892 (.015)	.892 (.015)	.694 (.032)	.694 (.032)	.748 (.029)	.748 (.029)	.675 (.006)	.675 (.006)
HCApp(LR) w/ disc.	.890 (.017)	.890 (.017)	.685 (.041)	.685 (.041)	.758 (.030)	.758 (.030)	.704 (.005)	.704 (.005)
HC(ANN)	.888 (.016)	.888 (.016)	.688 (.032)	.688 (.032)	.734 (.035)	.734 (.035)	.680 (.006)	.680 (.006)
HC(ANN) w/ disc.	.877 (.019)	.877 (.019)	.669 (.033)	.669 (.033)	.757 (.033)	.757 (.033)	.706 (.006)	.706 (.006)
IC-LR	.878 (.016)	.878 (.016)	.701 (.027)	.701 (.027)	.728 (.038)	.728 (.038)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.894 (.014)	.894 (.014)	.704 (.028)	.704 (.028)	.774 (.030)	**.774** (.030)	.710 (.005)	**.710** (.005)
IC-LR w/ neg.	**.895** (.013)	**.895** (.013)	**.720** (.031)	**.720** (.031)	.731 (.035)	.731 (.035)	.695 (.007)	.695 (.007)
Imp(LR)	.894 (.014)	.889 (.018)	.717 (.029)	.670 (.054)	.781 (.025)	.744 (.032)	.697 (.006)	.662 (.006)
Imp(LR) w/ disc.	**.895** (.013)	.863 (.021)	.703 (.027)	.648 (.061)	.782 (.029)	.698 (.086)	.726 (.005)	.674 (.027)
Imp(ANN)	**.895** (.014)	.889 (.018)	.718 (.029)	.670 (.054)	.776 (.027)	.744 (.027)	.723 (.009)	.676 (.013)
Imp(ANN) w/ disc.	.893 (.014)	.863 (.021)	.705 (.027)	.649 (.061)	.782 (.028)	.710 (.067)	**.727** (.005)	.681 (.020)
Imp(RF)	.866 (.016)	.850 (.034)	.670 (.030)	.629 (.064)	**.784** (.031)	.753 (.036)	.718 (.005)	.628 (.040)
Imp(RF) w/ disc.	.893 (.016)	.860 (.027)	.680 (.030)	.630 (.065)	.776 (.029)	.680 (.097)	.726 (.005)	.666 (.041)
Imp(κNN)	.880 (.022)	.848 (.050)	.676 (.034)	.619 (.075)	.764 (.031)	.727 (.039)	.710 (.013)	.664 (.023)
Imp(κNN) w/ disc.	.872 (.039)	.838 (.043)	.676 (.033)	.610 (.075)	.758 (.036)	.647 (.106)	.703 (.028)	.614 (.066)
R-F(LR)	.883 (.017)	.803 (.066)	.682 (.032)	.602 (.054)	.754 (.028)	.600 (.102)	.694 (.006)	.584 (.077)
R-F(LR) w/ disc.	.887 (.016)	.786 (.074)	.686 (.030)	.614 (.055)	.773 (.030)	.659 (.106)	.725 (.006)	.589 (.123)
R-F(ANN)	.880 (.018)	.795 (.074)	.670 (.035)	.598 (.060)	.750 (.030)	.642 (.089)	.705 (.007)	.598 (.090)
R-F(ANN) w/ disc.	.879 (.017)	.805 (.056)	.672 (.035)	.601 (.061)	.769 (.030)	.691 (.069)	.725 (.006)	.624 (.077)
R-F(RF)	.862 (.018)	.800 (.055)	.659 (.031)	.603 (.054)	.764 (.031)	.667 (.077)	.717 (.005)	.612 (.054)
R-F(RF) w/ disc.	.881 (.018)	.805 (.049)	.666 (.032)	.617 (.057)	.761 (.030)	.679 (.071)	.723 (.005)	.598 (.084)
R-F(κNN)	.869 (.021)	.718 (.114)	.650 (.031)	.569 (.049)	.734 (.032)	.615 (.068)	.709 (.010)	.621 (.050)
R-F(κNN) w/ disc.	.865 (.024)	.704 (.123)	.661 (.035)	.583 (.058)	.748 (.031)	.623 (.081)	.695 (.033)	.627 (.050)

Table 38: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.876 (.017)	.876 (.017)	.659 (.038)	.659 (.038)	.735 (.035)	.735 (.035)	.661 (.005)	.661 (.005)
HC(LR) w/ disc.	.872 (.016)	.872 (.016)	.664 (.032)	.664 (.032)	.744 (.033)	.744 (.033)	.692 (.006)	.692 (.006)
HCApp(LR)	.876 (.017)	.876 (.017)	.668 (.036)	.668 (.036)	.735 (.034)	.735 (.034)	.662 (.006)	.662 (.006)
HCApp(LR) w/ disc.	.871 (.017)	.871 (.017)	.662 (.044)	.662 (.044)	.742 (.032)	.742 (.032)	.691 (.006)	.691 (.006)
HC(ANN)	.868 (.021)	.868 (.021)	.657 (.033)	.657 (.033)	.718 (.036)	.718 (.036)	.666 (.007)	.666 (.007)
HC(ANN) w/ disc.	.850 (.023)	.850 (.023)	.641 (.036)	.641 (.036)	.732 (.034)	.732 (.034)	.691 (.006)	.691 (.006)
IC-LR	.852 (.018)	.852 (.018)	.681 (.030)	.681 (.030)	.731 (.041)	.731 (.041)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.874 (.015)	.874 (.015)	.683 (.029)	.683 (.029)	.755 (.032)	**.755** (.032)	.694 (.006)	**.694** (.006)
IC-LR w/ neg.	**.881** (.015)	**.881** (.015)	**.721** (.026)	**.721** (.026)	.725 (.042)	.725 (.042)	.689 (.005)	.689 (.005)
Imp(LR)	.878 (.015)	.869 (.016)	.703 (.031)	.659 (.051)	.772 (.029)	.732 (.031)	.689 (.005)	.655 (.006)
Imp(LR) w/ disc.	.877 (.014)	.833 (.024)	.686 (.028)	.638 (.057)	.773 (.030)	.670 (.073)	**.721** (.005)	.680 (.011)
Imp(ANN)	.879 (.015)	.869 (.016)	.704 (.031)	.661 (.051)	.770 (.029)	.735 (.029)	.713 (.007)	.668 (.010)
Imp(ANN) w/ disc.	.875 (.014)	.833 (.024)	.688 (.028)	.639 (.058)	.771 (.030)	.678 (.066)	**.721** (.005)	.682 (.009)
Imp(RF)	.849 (.019)	.821 (.036)	.656 (.034)	.623 (.059)	**.774** (.030)	.727 (.039)	.707 (.006)	.637 (.033)
Imp(RF) w/ disc.	.876 (.016)	.832 (.025)	.663 (.031)	.622 (.061)	.766 (.029)	.662 (.073)	.719 (.006)	.672 (.014)
Imp(κNN)	.861 (.024)	.836 (.040)	.664 (.035)	.616 (.066)	.753 (.030)	.714 (.040)	.700 (.009)	.653 (.026)
Imp(κNN) w/ disc.	.857 (.028)	.803 (.057)	.649 (.041)	.602 (.067)	.743 (.041)	.617 (.091)	.692 (.021)	.621 (.051)
R-F(LR)	.866 (.019)	.788 (.059)	.657 (.033)	.584 (.053)	.735 (.029)	.605 (.101)	.687 (.005)	.626 (.035)
R-F(LR) w/ disc.	.867 (.018)	.759 (.066)	.666 (.032)	.604 (.055)	.755 (.032)	.645 (.091)	.717 (.006)	.665 (.034)
R-F(ANN)	.862 (.020)	.792 (.065)	.651 (.034)	.589 (.052)	.731 (.029)	.638 (.089)	.693 (.006)	.639 (.019)
R-F(ANN) w/ disc.	.860 (.020)	.792 (.055)	.654 (.036)	.601 (.058)	.750 (.033)	.676 (.061)	.717 (.006)	.656 (.048)
R-F(RF)	.847 (.022)	.777 (.053)	.641 (.034)	.592 (.048)	.744 (.030)	.657 (.067)	.707 (.006)	.642 (.026)
R-F(RF) w/ disc.	.861 (.020)	.792 (.045)	.650 (.033)	.612 (.051)	.746 (.032)	.671 (.069)	.716 (.006)	.655 (.037)
R-F(κNN)	.840 (.021)	.694 (.096)	.626 (.037)	.551 (.044)	.714 (.030)	.611 (.060)	.698 (.012)	.649 (.022)
R-F(κNN) w/ disc.	.835 (.025)	.681 (.100)	.643 (.038)	.580 (.054)	.726 (.036)	.616 (.071)	.697 (.013)	.647 (.028)

Table 39: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.853 (.022)	.853 (.022)	.641 (.038)	.641 (.038)	.726 (.028)	.726 (.028)	.653 (.006)	.653 (.006)
HC(LR) w/ disc.	.846 (.024)	.846 (.024)	.646 (.036)	.646 (.036)	.728 (.031)	.728 (.031)	.675 (.007)	.675 (.007)
HCApp(LR)	.853 (.022)	.853 (.022)	.647 (.038)	.647 (.038)	.727 (.027)	.727 (.027)	.653 (.006)	.653 (.006)
HCApp(LR) w/ disc.	.846 (.023)	.846 (.023)	.645 (.042)	.645 (.042)	.728 (.031)	.728 (.031)	.677 (.006)	.677 (.006)
HC(ANN)	.841 (.023)	.841 (.023)	.634 (.036)	.634 (.036)	.700 (.033)	.700 (.033)	.655 (.006)	.655 (.006)
HC(ANN) w/ disc.	.825 (.028)	.825 (.028)	.615 (.036)	.615 (.036)	.718 (.034)	.718 (.034)	.677 (.006)	.677 (.006)
IC-LR	.820 (.024)	.820 (.024)	.662 (.035)	.662 (.035)	.737 (.036)	.737 (.036)	.502 (.014)	.502 (.014)
IC-LR w/ disc.	.850 (.021)	.850 (.021)	.662 (.035)	.662 (.035)	.739 (.030)	**.739** (.030)	.680 (.006)	.680 (.006)
IC-LR w/ neg.	**.862** (.016)	**.862** (.016)	.676 (.036)	**.676** (.036)	.737 (.029)	.737 (.029)	.682 (.006)	**.682** (.006)
Imp(LR)	.860 (.018)	.842 (.024)	.683 (.035)	.641 (.049)	**.768** (.024)	.727 (.028)	.684 (.005)	.648 (.005)
Imp(LR) w/ disc.	.854 (.019)	.798 (.043)	.668 (.037)	.622 (.056)	.762 (.029)	.664 (.073)	**.714** (.005)	.672 (.009)
Imp(ANN)	.860 (.018)	.841 (.024)	**.685** (.034)	.643 (.049)	.767 (.024)	.730 (.027)	.705 (.008)	.661 (.008)
Imp(ANN) w/ disc.	.853 (.020)	.798 (.043)	.669 (.037)	.622 (.057)	.760 (.029)	.668 (.072)	**.714** (.006)	.673 (.009)
Imp(RF)	.832 (.023)	.797 (.038)	.638 (.036)	.602 (.057)	.760 (.029)	.691 (.062)	.698 (.006)	.649 (.013)
Imp(RF) w/ disc.	.855 (.020)	.797 (.043)	.647 (.035)	.609 (.054)	.754 (.029)	.659 (.070)	.711 (.006)	.667 (.011)
Imp(κNN)	.838 (.026)	.795 (.070)	.642 (.038)	.596 (.061)	.743 (.031)	.700 (.048)	.691 (.009)	.634 (.029)
Imp(κNN) w/ disc.	.829 (.036)	.766 (.073)	.630 (.046)	.587 (.063)	.738 (.035)	.629 (.083)	.687 (.015)	.619 (.040)
R-F(LR)	.846 (.019)	.767 (.081)	.633 (.036)	.585 (.046)	.720 (.027)	.623 (.083)	.678 (.005)	.634 (.012)
R-F(LR) w/ disc.	.842 (.021)	.733 (.093)	.647 (.037)	.602 (.051)	.739 (.031)	.651 (.088)	.707 (.006)	.671 (.011)
R-F(ANN)	.842 (.020)	.784 (.053)	.628 (.036)	.581 (.044)	.718 (.029)	.651 (.062)	.680 (.006)	.630 (.021)
R-F(ANN) w/ disc.	.838 (.024)	.776 (.054)	.638 (.041)	.597 (.045)	.736 (.031)	.668 (.060)	.706 (.006)	.667 (.013)
R-F(RF)	.829 (.021)	.757 (.055)	.621 (.035)	.583 (.051)	.722 (.030)	.653 (.070)	.695 (.006)	.649 (.013)
R-F(RF) w/ disc.	.838 (.022)	.766 (.055)	.636 (.037)	.610 (.045)	.732 (.031)	.664 (.072)	.706 (.006)	.669 (.014)
R-F(κNN)	.819 (.024)	.705 (.076)	.601 (.036)	.551 (.042)	.700 (.031)	.631 (.050)	.690 (.008)	.649 (.017)
R-F(κNN) w/ disc.	.811 (.028)	.703 (.076)	.624 (.041)	.573 (.054)	.714 (.032)	.638 (.061)	.686 (.010)	.644 (.021)

Table 40: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.827 (.022)	.827 (.022)	.623 (.035)	.623 (.035)	.709 (.029)	.709 (.029)	.642 (.006)	.642 (.006)
HC(LR) w/ disc.	.817 (.023)	.817 (.023)	.627 (.035)	.627 (.035)	.707 (.027)	.707 (.027)	.661 (.006)	.661 (.006)
HCApp(LR)	.827 (.021)	.827 (.021)	.628 (.036)	.628 (.036)	.710 (.029)	.710 (.029)	.643 (.006)	.643 (.006)
HCApp(LR) w/ disc.	.818 (.022)	.818 (.022)	.625 (.041)	.625 (.041)	.707 (.027)	.707 (.027)	.662 (.006)	.662 (.006)
HC(ANN)	.807 (.027)	.807 (.027)	.612 (.037)	.612 (.037)	.680 (.039)	.680 (.039)	.644 (.007)	.644 (.007)
HC(ANN) w/ disc.	.794 (.030)	.794 (.030)	.598 (.038)	.598 (.038)	.696 (.031)	.696 (.031)	.662 (.007)	.662 (.007)
IC-LR	.788 (.025)	.788 (.025)	.640 (.034)	.640 (.034)	.728 (.038)	.728 (.038)	.501 (.010)	.501 (.010)
IC-LR w/ disc.	.826 (.022)	.826 (.022)	.641 (.033)	.641 (.033)	.720 (.026)	.720 (.026)	.667 (.006)	.667 (.006)
IC-LR w/ neg.	**.846** (.022)	**.846** (.022)	.663 (.035)	**.663** (.035)	.748 (.030)	**.748** (.030)	.678 (.005)	**.678** (.005)
Imp(LR)	.841 (.020)	.810 (.028)	**.665** (.033)	.629 (.044)	**.758** (.025)	.714 (.028)	.678 (.005)	.639 (.006)
Imp(LR) w/ disc.	.828 (.020)	.754 (.050)	.647 (.034)	.609 (.048)	.744 (.031)	.642 (.068)	**.706** (.006)	.657 (.009)
Imp(ANN)	.840 (.020)	.809 (.026)	**.665** (.033)	.630 (.044)	**.758** (.024)	.716 (.029)	.694 (.007)	.648 (.008)
Imp(ANN) w/ disc.	.827 (.021)	.754 (.049)	.647 (.034)	.609 (.049)	.742 (.031)	.644 (.066)	**.706** (.006)	.659 (.009)
Imp(RF)	.807 (.024)	.762 (.038)	.624 (.037)	.599 (.047)	.742 (.028)	.687 (.033)	.691 (.006)	.648 (.009)
Imp(RF) w/ disc.	.827 (.022)	.753 (.049)	.633 (.036)	.601 (.047)	.739 (.031)	.641 (.066)	.704 (.005)	.657 (.009)
Imp(κNN)	.814 (.028)	.755 (.065)	.627 (.035)	.592 (.051)	.731 (.027)	.675 (.050)	.680 (.009)	.622 (.021)
Imp(κNN) w/ disc.	.805 (.037)	.720 (.077)	.618 (.043)	.585 (.056)	.719 (.041)	.605 (.081)	.674 (.016)	.589 (.051)
R-F(LR)	.823 (.024)	.781 (.042)	.611 (.034)	.587 (.041)	.703 (.027)	.654 (.046)	.666 (.006)	.630 (.008)
R-F(LR) w/ disc.	.816 (.023)	.759 (.055)	.627 (.040)	.601 (.041)	.724 (.026)	.672 (.057)	.693 (.006)	.660 (.008)
R-F(ANN)	.818 (.025)	.777 (.041)	.606 (.034)	.582 (.042)	.700 (.029)	.656 (.042)	.666 (.006)	.630 (.013)
R-F(ANN) w/ disc.	.814 (.025)	.771 (.041)	.617 (.039)	.590 (.041)	.720 (.027)	.672 (.045)	.692 (.006)	.659 (.009)
R-F(RF)	.802 (.026)	.749 (.043)	.601 (.035)	.585 (.042)	.700 (.027)	.660 (.044)	.682 (.006)	.647 (.009)
R-F(RF) w/ disc.	.816 (.024)	.775 (.038)	.621 (.038)	.603 (.039)	.719 (.026)	.674 (.052)	.692 (.006)	.659 (.008)
R-F(κNN)	.798 (.028)	.725 (.048)	.584 (.035)	.557 (.042)	.689 (.028)	.646 (.042)	.675 (.008)	.638 (.012)
R-F(κNN) w/ disc.	.792 (.027)	.728 (.060)	.601 (.043)	.576 (.048)	.703 (.031)	.652 (.046)	.673 (.010)	.634 (.017)

Table 41: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, balanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.782 (.026)	.782 (.026)	.601 (.037)	.601 (.037)	.684 (.030)	.684 (.030)	.629 (.006)	.629 (.006)
HC(LR) w/ disc.	.775 (.026)	.775 (.026)	.600 (.036)	.600 (.036)	.683 (.032)	.683 (.032)	.646 (.006)	.646 (.006)
HCApp(LR)	.783 (.026)	.783 (.026)	.603 (.037)	.603 (.037)	.685 (.029)	.685 (.029)	.629 (.005)	.629 (.005)
HCApp(LR) w/ disc.	.775 (.027)	.775 (.027)	.599 (.039)	.599 (.039)	.683 (.032)	.683 (.032)	.646 (.006)	.646 (.006)
HC(ANN)	.765 (.033)	.765 (.033)	.589 (.035)	.589 (.035)	.656 (.031)	.656 (.031)	.631 (.006)	.631 (.006)
HC(ANN) w/ disc.	.758 (.030)	.758 (.030)	.572 (.039)	.572 (.039)	.671 (.032)	.671 (.032)	.646 (.006)	.646 (.006)
IC-LR	.744 (.028)	.744 (.028)	.614 (.035)	.614 (.035)	.720 (.036)	.720 (.036)	.501 (.007)	.501 (.007)
IC-LR w/ disc.	.790 (.023)	.790 (.023)	.614 (.035)	.614 (.035)	.700 (.030)	.700 (.030)	.653 (.006)	.653 (.006)
IC-LR w/ neg.	**.819** (.021)	**.819** (.021)	**.639** (.031)	**.639** (.031)	.724 (.031)	**.724** (.031)	.672 (.005)	**.672** (.005)
Imp(LR)	.811 (.024)	.771 (.031)	.638 (.036)	.602 (.044)	.746 (.026)	.698 (.027)	.672 (.005)	.629 (.006)
Imp(LR) w/ disc.	.788 (.023)	.713 (.047)	.622 (.037)	.584 (.047)	.727 (.032)	.618 (.059)	**.696** (.005)	.642 (.009)
Imp(ANN)	.812 (.023)	.771 (.031)	**.639** (.036)	.603 (.044)	**.747** (.026)	.700 (.028)	.685 (.007)	.635 (.007)
Imp(ANN) w/ disc.	.788 (.023)	.713 (.047)	.619 (.040)	.581 (.051)	.725 (.031)	.619 (.058)	**.696** (.005)	.643 (.008)
Imp(RF)	.782 (.025)	.736 (.036)	.606 (.036)	.581 (.045)	.725 (.031)	.667 (.033)	.682 (.006)	.641 (.007)
Imp(RF) w/ disc.	.787 (.024)	.712 (.046)	.609 (.038)	.577 (.047)	.725 (.031)	.618 (.058)	.694 (.005)	.642 (.008)
Imp(κNN)	.781 (.029)	.725 (.063)	.602 (.040)	.572 (.050)	.716 (.032)	.659 (.046)	.668 (.009)	.605 (.029)
Imp(κNN) w/ disc.	.765 (.039)	.685 (.069)	.591 (.048)	.561 (.053)	.704 (.039)	.590 (.067)	.662 (.015)	.581 (.043)
R-F(LR)	.784 (.028)	.757 (.034)	.590 (.034)	.575 (.036)	.681 (.029)	.649 (.037)	.649 (.006)	.621 (.007)
R-F(LR) w/ disc.	.778 (.030)	.744 (.039)	.605 (.040)	.581 (.040)	.702 (.031)	.669 (.034)	.673 (.006)	.645 (.007)
R-F(ANN)	.778 (.030)	.747 (.036)	.586 (.035)	.568 (.036)	.676 (.036)	.641 (.038)	.646 (.007)	.620 (.009)
R-F(ANN) w/ disc.	.776 (.029)	.741 (.037)	.592 (.041)	.563 (.042)	.698 (.032)	.663 (.034)	.671 (.006)	.641 (.010)
R-F(RF)	.770 (.030)	.727 (.039)	.581 (.034)	.571 (.038)	.678 (.029)	.651 (.032)	.664 (.006)	.636 (.007)
R-F(RF) w/ disc.	.783 (.029)	.748 (.035)	.601 (.039)	.582 (.038)	.703 (.031)	.670 (.033)	.672 (.006)	.644 (.007)
R-F(κNN)	.766 (.031)	.715 (.045)	.565 (.035)	.555 (.037)	.672 (.033)	.646 (.034)	.655 (.008)	.622 (.013)
R-F(κNN) w/ disc.	.762 (.029)	.715 (.045)	.581 (.038)	.564 (.043)	.688 (.031)	.654 (.039)	.653 (.009)	.620 (.014)

Table 42: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.848 (.019)	.848 (.019)	.725 (.016)	.725 (.016)	**.932** (.003)	**.932** (.003)	.809 (.002)	.809 (.002)
HC(LR) w/ disc.	.850 (.014)	.850 (.014)	.716 (.018)	.716 (.018)	**.932** (.003)	**.932** (.003)	**.820** (.003)	**.820** (.003)
HCApp(LR)	.851 (.016)	.851 (.016)	.719 (.017)	.719 (.017)	**.932** (.003)	**.932** (.003)	.809 (.009)	.809 (.009)
HCApp(LR) w/ disc.	.841 (.033)	.841 (.033)	.709 (.016)	.709 (.016)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
HC(ANN)	.851 (.015)	.851 (.015)	.727 (.016)	.727 (.016)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
HC(ANN) w/ disc.	.848 (.015)	.848 (.015)	.720 (.016)	.720 (.016)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
Mincut	.735 (.027)	.735 (.027)	.472 (.135)	.472 (.135)	.085 (.004)	.085 (.004)	.818 (.003)	.818 (.003)
Mincut w/ disc.	.845 (.014)	.845 (.014)	.695 (.025)	.695 (.025)	.931 (.004)	.931 (.004)	**.820** (.003)	**.820** (.003)
IC-LR	**.855** (.014)	**.855** (.014)	.715 (.020)	.715 (.020)	.931 (.003)	.931 (.003)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.852 (.015)	.852 (.015)	.720 (.018)	.720 (.018)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
IC-LR w/ neg.	**.855** (.014)	**.855** (.014)	.731 (.016)	.731 (.016)	.931 (.003)	.931 (.003)	.808 (.003)	.808 (.003)
Maj	.735 (.027)	.735 (.027)	.464 (.129)	.464 (.129)	.084 (.004)	.084 (.004)	.819 (.003)	.819 (.003)
Maj w/ disc.	.847 (.015)	.847 (.015)	.689 (.029)	.689 (.029)	.929 (.005)	.929 (.005)	.819 (.003)	.819 (.003)
Imp(LR)	.854 (.014)	.827 (.093)	**.733** (.015)	.698 (.013)	.931 (.003)	.931 (.003)	.809 (.002)	.779 (.002)
Imp(LR) w/ disc.	.852 (.014)	.817 (.092)	.720 (.018)	.698 (.013)	.931 (.003)	.931 (.003)	**.820** (.003)	.779 (.002)
Imp(ANN)	.850 (.015)	.823 (.092)	**.733** (.015)	.698 (.013)	.931 (.003)	.931 (.003)	**.820** (.003)	.779 (.002)
Imp(ANN) w/ disc.	.848 (.014)	.786 (.091)	.721 (.017)	.698 (.013)	.931 (.003)	.931 (.003)	**.820** (.003)	.779 (.002)
Imp(RF)	.820 (.015)	.666 (.097)	.700 (.018)	.698 (.013)	.930 (.005)	.931 (.003)	.819 (.003)	.779 (.002)
Imp(RF) w/ disc.	.848 (.015)	.803 (.090)	.708 (.022)	.698 (.013)	.931 (.004)	.931 (.003)	.819 (.003)	.779 (.002)
Imp(κNN)	.846 (.017)	.745 (.112)	.719 (.022)	.698 (.013)	.931 (.003)	.931 (.003)	.812 (.007)	.769 (.053)
Imp(κNN) w/ disc.	.842 (.020)	.759 (.119)	.708 (.022)	.698 (.013)	.930 (.004)	.931 (.003)	.808 (.013)	.761 (.097)
R-F(LR)	.854 (.014)	.854 (.014)	**.733** (.015)	**.733** (.015)	.931 (.003)	.931 (.003)	.809 (.002)	.809 (.002)
R-F(LR) w/ disc.	.852 (.014)	.852 (.014)	.720 (.018)	.720 (.018)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
R-F(ANN)	.850 (.015)	.850 (.015)	**.733** (.016)	.732 (.014)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
R-F(ANN) w/ disc.	.849 (.015)	.848 (.014)	.720 (.018)	.722 (.018)	.931 (.003)	.931 (.003)	**.820** (.003)	**.820** (.003)
R-F(RF)	.820 (.015)	.820 (.016)	.701 (.017)	.700 (.018)	.930 (.005)	.930 (.005)	.819 (.003)	.819 (.003)
R-F(RF) w/ disc.	.848 (.014)	.848 (.015)	.709 (.021)	.710 (.021)	.931 (.004)	.931 (.004)	.819 (.003)	.819 (.003)
R-F(κNN)	.844 (.018)	.846 (.018)	.719 (.022)	.721 (.020)	.930 (.003)	.931 (.003)	.812 (.006)	.812 (.006)
R-F(κNN) w/ disc.	.841 (.028)	.841 (.028)	.706 (.023)	.706 (.023)	.930 (.004)	.930 (.004)	.807 (.011)	.804 (.016)

Table 43: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.810 (.014)	.810 (.014)	.699 (.018)	.699 (.018)	.930 (.004)	.930 (.004)	.803 (.003)	.803 (.003)
HC(LR) w/ disc.	.814 (.016)	.814 (.016)	.695 (.017)	.695 (.017)	.929 (.004)	.929 (.004)	.804 (.003)	**.804** (.003)
HCApp(LR)	.814 (.015)	.814 (.015)	.702 (.016)	.702 (.016)	.930 (.004)	.930 (.004)	.795 (.013)	.795 (.013)
HCApp(LR) w/ disc.	.814 (.019)	.814 (.019)	.699 (.016)	.699 (.016)	.929 (.004)	.929 (.004)	.804 (.003)	**.804** (.003)
HC(ANN)	.794 (.021)	.794 (.021)	.688 (.021)	.688 (.021)	.930 (.004)	.930 (.004)	.803 (.003)	.803 (.003)
HC(ANN) w/ disc.	.792 (.020)	.792 (.020)	.692 (.017)	.692 (.017)	.930 (.004)	.930 (.004)	.802 (.003)	.802 (.003)
Mincut	.770 (.030)	.770 (.030)	.512 (.107)	.512 (.107)	.081 (.004)	.081 (.004)	.802 (.003)	.802 (.003)
Mincut w/ disc.	.808 (.016)	.808 (.016)	.684 (.019)	.684 (.019)	.928 (.004)	.928 (.004)	.804 (.002)	**.804** (.002)
IC-LR	.812 (.014)	.812 (.014)	.701 (.016)	.701 (.016)	**.931** (.004)	**.931** (.004)	.778 (.003)	.778 (.003)
IC-LR w/ disc.	.822 (.015)	**.822** (.015)	.704 (.019)	.704 (.019)	.930 (.004)	.930 (.004)	.804 (.003)	**.804** (.003)
IC-LR w/ neg.	.811 (.016)	.811 (.016)	.720 (.017)	**.720** (.017)	**.931** (.004)	**.931** (.004)	.802 (.003)	.802 (.003)
Maj	.678 (.021)	.744 (.060)	.410 (.094)	.591 (.070)	.077 (.004)	.137 (.116)	.810 (.002)	.779 (.003)
Maj w/ disc.	.811 (.021)	.697 (.082)	.664 (.031)	.689 (.017)	.922 (.006)	.929 (.004)	**.816** (.002)	.779 (.005)
Imp(LR)	.816 (.015)	.760 (.104)	**.726** (.015)	.700 (.015)	**.931** (.004)	**.931** (.004)	.804 (.002)	.778 (.003)
Imp(LR) w/ disc.	.824 (.015)	.740 (.103)	.716 (.019)	.700 (.015)	.930 (.004)	**.931** (.004)	**.816** (.002)	.778 (.003)
Imp(ANN)	.820 (.016)	.755 (.104)	**.726** (.016)	.700 (.015)	**.931** (.004)	**.931** (.004)	**.816** (.002)	.778 (.003)
Imp(ANN) w/ disc.	.824 (.015)	.732 (.103)	.717 (.017)	.700 (.015)	**.931** (.004)	**.931** (.004)	**.816** (.002)	.778 (.003)
Imp(RF)	.794 (.020)	.635 (.091)	.686 (.023)	.699 (.015)	.929 (.005)	**.931** (.004)	.813 (.003)	.778 (.003)
Imp(RF) w/ disc.	.821 (.018)	.726 (.106)	.707 (.019)	.700 (.015)	.929 (.004)	**.931** (.004)	**.816** (.002)	.778 (.003)
Imp(κNN)	.810 (.018)	.679 (.098)	.712 (.019)	.698 (.015)	.930 (.004)	**.931** (.004)	.810 (.006)	.778 (.003)
Imp(κNN) w/ disc.	.818 (.018)	.694 (.120)	.704 (.024)	.697 (.020)	.927 (.007)	**.931** (.004)	.811 (.006)	.778 (.003)
R-F(LR)	**.828** (.018)	.576 (.051)	.709 (.015)	.696 (.014)	.930 (.004)	.930 (.004)	.804 (.003)	.778 (.003)
R-F(LR) w/ disc.	.820 (.015)	.570 (.046)	.703 (.017)	.696 (.015)	.929 (.003)	.930 (.004)	**.816** (.003)	.778 (.003)
R-F(ANN)	.823 (.017)	.581 (.055)	.708 (.015)	.696 (.015)	.930 (.004)	.930 (.004)	.810 (.003)	.778 (.003)
R-F(ANN) w/ disc.	.818 (.017)	.570 (.037)	.703 (.019)	.696 (.014)	.930 (.004)	.930 (.004)	.814 (.003)	.778 (.003)
R-F(RF)	.796 (.018)	.578 (.050)	.681 (.019)	.696 (.015)	.927 (.005)	.930 (.004)	.813 (.003)	.778 (.003)
R-F(RF) w/ disc.	.816 (.022)	.573 (.042)	.695 (.018)	.696 (.015)	.927 (.005)	.930 (.004)	**.816** (.002)	.778 (.003)
R-F(κNN)	.807 (.019)	.564 (.043)	.696 (.019)	.695 (.016)	.930 (.005)	.930 (.004)	.809 (.005)	.778 (.003)
R-F(κNN) w/ disc.	.802 (.022)	.564 (.040)	.696 (.019)	.695 (.015)	.929 (.004)	.930 (.004)	.810 (.004)	.778 (.003)

Table 44: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.787 (.020)	.787 (.020)	.693 (.018)	.693 (.018)	.929 (.003)	.929 (.003)	.800 (.003)	.800 (.003)
HC(LR) w/ disc.	.788 (.020)	.788 (.020)	.681 (.017)	.681 (.017)	.928 (.003)	.928 (.003)	.800 (.003)	.800 (.003)
HCApp(LR)	.791 (.020)	.791 (.020)	.692 (.022)	.692 (.022)	.929 (.003)	.929 (.003)	.791 (.010)	.791 (.010)
HCApp(LR) w/ disc.	.788 (.020)	.788 (.020)	.691 (.017)	.691 (.017)	.929 (.003)	.929 (.003)	.800 (.003)	.800 (.003)
HC(ANN)	.775 (.019)	.775 (.019)	.676 (.024)	.676 (.024)	.929 (.003)	.929 (.003)	.801 (.003)	**.801** (.003)
HC(ANN) w/ disc.	.768 (.021)	.768 (.021)	.692 (.022)	.692 (.022)	.929 (.003)	.929 (.003)	.801 (.003)	**.801** (.003)
Mincut	.767 (.024)	.767 (.024)	.550 (.083)	.550 (.083)	.083 (.018)	.083 (.018)	.800 (.003)	.800 (.003)
Mincut w/ disc.	.788 (.020)	.788 (.020)	.676 (.020)	.676 (.020)	.929 (.004)	.929 (.004)	.799 (.003)	.799 (.003)
IC-LR	.777 (.023)	.777 (.023)	.696 (.017)	.696 (.017)	**.931** (.003)	**.931** (.003)	.779 (.003)	.779 (.003)
IC-LR w/ disc.	.791 (.020)	.791 (.020)	.693 (.016)	.693 (.016)	**.931** (.003)	**.931** (.003)	.800 (.003)	.800 (.003)
IC-LR w/ neg.	.793 (.020)	**.793** (.020)	.711 (.019)	**.711** (.019)	**.931** (.003)	**.931** (.003)	.801 (.003)	**.801** (.003)
Maj	.649 (.028)	.711 (.073)	.420 (.094)	.669 (.033)	.075 (.003)	.573 (.085)	.810 (.003)	.781 (.004)
Maj w/ disc.	.797 (.021)	.679 (.068)	.652 (.043)	.699 (.017)	.921 (.005)	.930 (.003)	.814 (.003)	.782 (.006)
Imp(LR)	.796 (.020)	.734 (.100)	**.720** (.019)	.701 (.016)	**.931** (.003)	**.931** (.003)	.802 (.003)	.779 (.003)
Imp(LR) w/ disc.	.794 (.020)	.681 (.112)	.708 (.016)	.701 (.016)	.930 (.003)	**.931** (.003)	.814 (.003)	.779 (.003)
Imp(ANN)	.798 (.021)	.722 (.096)	.719 (.018)	.701 (.016)	**.931** (.003)	**.931** (.003)	.813 (.002)	.779 (.003)
Imp(ANN) w/ disc.	.793 (.021)	.674 (.112)	.707 (.016)	.701 (.016)	**.931** (.003)	**.931** (.003)	.814 (.003)	.779 (.003)
Imp(RF)	.765 (.017)	.601 (.070)	.686 (.019)	.700 (.016)	.928 (.004)	**.931** (.003)	.812 (.003)	.780 (.004)
Imp(RF) w/ disc.	.795 (.020)	.675 (.111)	.699 (.017)	.701 (.016)	.929 (.004)	**.931** (.003)	**.815** (.003)	.779 (.003)
Imp(κNN)	.786 (.017)	.653 (.093)	.704 (.021)	.698 (.018)	.930 (.003)	**.931** (.003)	.807 (.005)	.780 (.004)
Imp(κNN) w/ disc.	.780 (.030)	.644 (.106)	.687 (.025)	.685 (.055)	.928 (.006)	**.931** (.003)	.803 (.011)	.768 (.042)
R-F(LR)	**.810** (.021)	.580 (.035)	.699 (.019)	.698 (.017)	.929 (.003)	.929 (.003)	.803 (.004)	.779 (.004)
R-F(LR) w/ disc.	.800 (.019)	.573 (.038)	.698 (.018)	.699 (.017)	.929 (.003)	.929 (.003)	.814 (.003)	.781 (.006)
R-F(ANN)	.805 (.021)	.578 (.031)	.693 (.020)	.697 (.017)	.929 (.003)	.929 (.003)	.806 (.003)	.778 (.003)
R-F(ANN) w/ disc.	.799 (.021)	.581 (.043)	.697 (.020)	.700 (.016)	.929 (.003)	.929 (.003)	.810 (.003)	.778 (.003)
R-F(RF)	.778 (.021)	.575 (.030)	.672 (.024)	.697 (.017)	.924 (.004)	.928 (.003)	.811 (.003)	.782 (.007)
R-F(RF) w/ disc.	.801 (.020)	.590 (.048)	.681 (.023)	.700 (.016)	.926 (.004)	.929 (.003)	.813 (.003)	.785 (.010)
R-F(κNN)	.782 (.019)	.577 (.039)	.692 (.020)	.697 (.017)	.928 (.004)	.929 (.003)	.806 (.005)	.782 (.007)
R-F(κNN) w/ disc.	.784 (.024)	.576 (.039)	.689 (.023)	.699 (.017)	.928 (.003)	.929 (.003)	.807 (.005)	.781 (.006)

Table 45: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia Tru.	Australia Str.	Germany Tru.	Germany Str.	Poland Tru.	Poland Str.	Taiwan Tru.	Taiwan Str.
HC(LR)	.774 (.023)	.774 (.023)	.679 (.019)	.679 (.019)	.924 (.004)	.924 (.004)	.796 (.002)	.796 (.002)
HC(LR) w/ disc.	.771 (.025)	.771 (.025)	.682 (.019)	.682 (.019)	.923 (.005)	.923 (.005)	.796 (.002)	.796 (.002)
HCApp(LR)	.779 (.024)	**.779** (.024)	.683 (.018)	.683 (.018)	.924 (.004)	.924 (.004)	.791 (.009)	.791 (.009)
HCApp(LR) w/ disc.	.776 (.022)	.776 (.022)	.688 (.016)	.688 (.016)	.924 (.005)	.924 (.005)	.796 (.002)	.796 (.002)
HC(ANN)	.755 (.029)	.755 (.029)	.671 (.021)	.671 (.021)	.924 (.004)	.924 (.004)	.796 (.002)	.796 (.002)
HC(ANN) w/ disc.	.746 (.030)	.746 (.030)	.692 (.017)	.692 (.017)	.924 (.004)	.924 (.004)	.796 (.002)	.796 (.002)
Mincut	.741 (.027)	.741 (.027)	.590 (.073)	.590 (.073)	.084 (.017)	.084 (.017)	.791 (.007)	.791 (.007)
Mincut w/ disc.	.770 (.025)	.770 (.025)	.680 (.020)	.680 (.020)	.930 (.005)	.930 (.005)	.795 (.004)	.795 (.004)
IC-LR	.762 (.035)	.762 (.035)	.697 (.013)	.697 (.013)	**.931** (.004)	**.931** (.004)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.771 (.025)	.771 (.025)	.694 (.015)	.694 (.015)	**.931** (.004)	**.931** (.004)	.781 (.006)	.781 (.006)
IC-LR w/ neg.	.777 (.024)	.777 (.024)	.707 (.016)	**.707** (.016)	**.931** (.004)	**.931** (.004)	.798 (.003)	**.798** (.003)
Maj	.645 (.031)	.684 (.078)	.442 (.076)	.685 (.026)	.078 (.004)	.731 (.008)	.807 (.002)	.780 (.002)
Maj w/ disc.	.774 (.021)	.657 (.073)	.653 (.032)	.695 (.016)	.922 (.005)	**.931** (.004)	**.811** (.002)	.780 (.002)
Imp(LR)	.781 (.018)	.722 (.080)	**.710** (.015)	.697 (.013)	**.931** (.004)	**.931** (.004)	.799 (.003)	.779 (.002)
Imp(LR) w/ disc.	.771 (.017)	.688 (.100)	.702 (.017)	.697 (.013)	.930 (.004)	**.931** (.004)	.809 (.003)	.779 (.002)
Imp(ANN)	.783 (.019)	.706 (.087)	**.710** (.015)	.697 (.013)	**.931** (.004)	**.931** (.004)	.809 (.003)	.779 (.002)
Imp(ANN) w/ disc.	.769 (.019)	.676 (.103)	.704 (.016)	.697 (.013)	.930 (.003)	**.931** (.004)	**.811** (.002)	.779 (.002)
Imp(RF)	.751 (.025)	.632 (.084)	.674 (.021)	.697 (.013)	.926 (.004)	**.931** (.004)	.809 (.003)	.779 (.002)
Imp(RF) w/ disc.	.770 (.017)	.679 (.104)	.691 (.017)	.697 (.013)	.929 (.004)	**.931** (.004)	**.811** (.002)	.779 (.002)
Imp(κNN)	.767 (.020)	.653 (.092)	.694 (.019)	.692 (.020)	.930 (.004)	.930 (.004)	.804 (.004)	.779 (.002)
Imp(κNN) w/ disc.	.758 (.030)	.677 (.099)	.681 (.041)	.684 (.046)	.928 (.007)	**.931** (.004)	.804 (.005)	.766 (.056)
R-F(LR)	**.788** (.021)	.611 (.052)	.688 (.017)	.692 (.014)	.924 (.005)	.924 (.004)	.798 (.003)	.776 (.002)
R-F(LR) w/ disc.	.774 (.024)	.597 (.051)	.694 (.018)	.695 (.013)	.923 (.004)	.924 (.004)	.807 (.002)	.782 (.007)
R-F(ANN)	.785 (.021)	.613 (.056)	.688 (.017)	.690 (.015)	.924 (.004)	.924 (.004)	.797 (.003)	.775 (.002)
R-F(ANN) w/ disc.	.777 (.023)	.614 (.055)	.694 (.018)	.695 (.013)	.924 (.004)	.924 (.004)	.802 (.003)	.775 (.002)
R-F(RF)	.759 (.021)	.624 (.050)	.656 (.022)	.689 (.015)	.916 (.004)	.921 (.004)	.804 (.002)	.781 (.005)
R-F(RF) w/ disc.	.777 (.025)	.628 (.063)	.682 (.021)	.695 (.013)	.920 (.004)	.924 (.004)	.807 (.002)	.783 (.006)
R-F(κNN)	.768 (.024)	.606 (.053)	.678 (.018)	.689 (.016)	.923 (.004)	.924 (.004)	.800 (.004)	.778 (.005)
R-F(κNN) w/ disc.	.763 (.027)	.608 (.056)	.687 (.023)	.694 (.014)	.922 (.004)	.924 (.004)	.796 (.017)	.780 (.005)

Table 46: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.755 (.021)	.755 (.021)	.669 (.018)	.669 (.018)	.910 (.004)	.910 (.004)	.786 (.002)	.786 (.002)
HC(LR) w/ disc.	.753 (.023)	.753 (.023)	.685 (.022)	.685 (.022)	.908 (.004)	.908 (.004)	.786 (.002)	.786 (.002)
HCApp(LR)	.756 (.021)	**.756** (.021)	.675 (.018)	.675 (.018)	.910 (.004)	.910 (.004)	.779 (.010)	.779 (.010)
HCApp(LR) w/ disc.	.753 (.023)	.753 (.023)	.685 (.022)	.685 (.022)	.909 (.004)	.909 (.004)	.786 (.002)	.786 (.002)
HC(ANN)	.734 (.028)	.734 (.028)	.670 (.022)	.670 (.022)	.909 (.004)	.909 (.004)	.786 (.003)	.786 (.003)
HC(ANN) w/ disc.	.723 (.033)	.723 (.033)	.689 (.024)	.689 (.024)	.909 (.004)	.909 (.004)	.786 (.002)	.786 (.002)
Mincut	.725 (.033)	.725 (.033)	.604 (.051)	.604 (.051)	.081 (.010)	.081 (.010)	.777 (.022)	.777 (.022)
Mincut w/ disc.	.751 (.021)	.751 (.021)	.680 (.024)	.680 (.024)	.931 (.004)	.931 (.004)	.782 (.014)	.782 (.014)
IC-LR	.741 (.029)	.741 (.029)	.698 (.015)	.698 (.015)	**.932** (.004)	**.932** (.004)	.778 (.002)	.778 (.002)
IC-LR w/ disc.	.746 (.021)	.746 (.021)	.696 (.016)	.696 (.016)	.931 (.004)	.931 (.004)	.778 (.002)	.778 (.002)
IC-LR w/ neg.	.734 (.039)	.734 (.039)	.705 (.017)	**.705** (.017)	**.932** (.004)	**.932** (.004)	.795 (.003)	**.795** (.003)
Maj	.628 (.025)	.628 (.077)	.479 (.063)	.689 (.020)	.092 (.004)	.822 (.007)	.803 (.003)	.779 (.002)
Maj w/ disc.	.760 (.035)	.619 (.075)	.645 (.041)	.696 (.017)	.923 (.006)	.931 (.004)	**.808** (.003)	.779 (.002)
Imp(LR)	.759 (.020)	.660 (.099)	**.706** (.019)	.698 (.015)	**.932** (.004)	.931 (.004)	.796 (.003)	.778 (.002)
Imp(LR) w/ disc.	.747 (.020)	.636 (.097)	.701 (.017)	.698 (.015)	.930 (.004)	.931 (.004)	.805 (.003)	.778 (.002)
Imp(ANN)	.759 (.021)	.642 (.095)	.704 (.019)	.698 (.015)	.931 (.004)	.931 (.004)	.805 (.003)	.778 (.002)
Imp(ANN) w/ disc.	.746 (.019)	.637 (.098)	.701 (.016)	.698 (.015)	.931 (.004)	.931 (.004)	**.808** (.003)	.778 (.002)
Imp(RF)	.730 (.024)	.612 (.075)	.670 (.022)	.698 (.015)	.925 (.005)	.931 (.004)	.805 (.003)	.779 (.002)
Imp(RF) w/ disc.	.745 (.020)	.621 (.093)	.691 (.020)	.698 (.015)	.929 (.004)	.931 (.004)	**.808** (.003)	.778 (.002)
Imp(κNN)	.746 (.026)	.641 (.091)	.688 (.020)	.693 (.020)	.931 (.004)	.931 (.004)	.803 (.004)	.778 (.002)
Imp(κNN) w/ disc.	.736 (.030)	.633 (.091)	.682 (.039)	.685 (.054)	.927 (.007)	.931 (.004)	.797 (.011)	.770 (.034)
R-F(LR)	.769 (.024)	.642 (.055)	.671 (.017)	.684 (.015)	.909 (.005)	.910 (.004)	.785 (.003)	.766 (.003)
R-F(LR) w/ disc.	.765 (.020)	.628 (.061)	.687 (.020)	.692 (.019)	.908 (.005)	.909 (.004)	.794 (.003)	.777 (.007)
R-F(ANN)	.763 (.023)	.625 (.055)	.669 (.018)	.683 (.016)	.909 (.004)	.909 (.004)	.782 (.003)	.764 (.002)
R-F(ANN) w/ disc.	.763 (.023)	.631 (.064)	.689 (.021)	.691 (.018)	.909 (.004)	.909 (.004)	.789 (.003)	.766 (.003)
R-F(RF)	.743 (.021)	.642 (.047)	.648 (.018)	.678 (.017)	.895 (.007)	.902 (.005)	.791 (.002)	.775 (.006)
R-F(RF) w/ disc.	**.771** (.021)	.676 (.065)	.676 (.024)	.690 (.019)	.906 (.006)	.909 (.004)	.794 (.003)	.777 (.007)
R-F(κNN)	.750 (.027)	.646 (.057)	.672 (.018)	.682 (.018)	.908 (.005)	.909 (.004)	.788 (.004)	.770 (.005)
R-F(κNN) w/ disc.	.752 (.027)	.654 (.064)	.681 (.022)	.692 (.018)	.907 (.005)	.909 (.004)	.786 (.008)	.770 (.007)

Table 47: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.735 (.025)	.735 (.025)	.660 (.022)	.660 (.022)	.877 (.005)	.877 (.005)	.764 (.004)	.764 (.004)
HC(LR) w/ disc.	.725 (.026)	.725 (.026)	.683 (.029)	.683 (.029)	.878 (.011)	.878 (.011)	.764 (.004)	.764 (.004)
HCApp(LR)	.738 (.023)	**.738** (.023)	.662 (.020)	.662 (.020)	.877 (.005)	.877 (.005)	.759 (.009)	.759 (.009)
HCApp(LR) w/ disc.	.724 (.024)	.724 (.024)	.683 (.028)	.683 (.028)	.878 (.011)	.878 (.011)	.764 (.004)	.764 (.004)
HC(ANN)	.720 (.033)	.720 (.033)	.667 (.019)	.667 (.019)	.877 (.005)	.877 (.005)	.764 (.004)	.764 (.004)
HC(ANN) w/ disc.	.714 (.026)	.714 (.026)	.690 (.019)	.690 (.019)	.879 (.011)	.879 (.011)	.764 (.003)	.764 (.003)
Mincut	.721 (.029)	.721 (.029)	.622 (.064)	.622 (.064)	.087 (.016)	.087 (.016)	.754 (.020)	.754 (.020)
Mincut w/ disc.	.725 (.026)	.725 (.026)	.696 (.017)	.696 (.017)	.930 (.003)	.930 (.003)	.773 (.024)	.773 (.024)
IC-LR	.712 (.031)	.712 (.031)	.700 (.012)	.700 (.012)	**.931** (.003)	**.931** (.003)	.779 (.003)	.779 (.003)
IC-LR w/ disc.	.715 (.027)	.715 (.027)	.699 (.012)	.699 (.012)	.930 (.003)	.930 (.003)	.779 (.003)	.779 (.003)
IC-LR w/ neg.	.704 (.038)	.704 (.038)	**.702** (.011)	**.702** (.011)	**.931** (.003)	**.931** (.003)	.793 (.003)	**.793** (.003)
Maj	.648 (.032)	.623 (.083)	.518 (.054)	.698 (.013)	.126 (.005)	.879 (.005)	.800 (.003)	.779 (.003)
Maj w/ disc.	.746 (.031)	.614 (.075)	.641 (.040)	.700 (.012)	.924 (.004)	.930 (.003)	**.804** (.002)	.779 (.003)
Imp(LR)	.733 (.021)	.618 (.085)	.700 (.017)	.700 (.012)	**.931** (.003)	.930 (.003)	.793 (.003)	.779 (.003)
Imp(LR) w/ disc.	.721 (.032)	.569 (.059)	.696 (.016)	.700 (.012)	.929 (.003)	.930 (.003)	.803 (.003)	.779 (.003)
Imp(ANN)	.733 (.019)	.604 (.080)	.700 (.015)	.700 (.012)	.930 (.003)	.930 (.003)	.802 (.003)	.779 (.003)
Imp(ANN) w/ disc.	.725 (.026)	.569 (.058)	.698 (.013)	.700 (.012)	.930 (.003)	.930 (.003)	**.804** (.003)	.779 (.003)
Imp(RF)	.714 (.023)	.571 (.056)	.675 (.023)	.700 (.012)	.920 (.004)	.930 (.003)	.802 (.003)	.779 (.003)
Imp(RF) w/ disc.	.724 (.023)	.568 (.058)	.691 (.017)	.700 (.012)	.929 (.004)	.930 (.003)	**.804** (.002)	.779 (.003)
Imp(κNN)	.721 (.031)	.599 (.073)	.684 (.023)	.682 (.046)	.929 (.003)	.930 (.003)	.796 (.012)	.776 (.010)
Imp(κNN) w/ disc.	.706 (.058)	.578 (.071)	.679 (.045)	.689 (.045)	.928 (.003)	.930 (.003)	.790 (.031)	.771 (.027)
R-F(LR)	**.750** (.023)	.659 (.042)	.655 (.019)	.672 (.016)	.877 (.005)	.877 (.005)	.760 (.004)	.746 (.004)
R-F(LR) w/ disc.	.743 (.026)	.658 (.053)	.682 (.022)	.690 (.019)	.878 (.011)	.879 (.011)	.769 (.003)	.757 (.005)
R-F(ANN)	.739 (.026)	.639 (.047)	.658 (.018)	.673 (.017)	.877 (.005)	.877 (.005)	.756 (.004)	.745 (.004)
R-F(ANN) w/ disc.	.738 (.029)	.634 (.049)	.685 (.021)	.691 (.019)	.879 (.011)	.879 (.011)	.764 (.004)	.747 (.004)
R-F(RF)	.730 (.028)	.658 (.043)	.634 (.020)	.667 (.016)	.862 (.005)	.866 (.005)	.766 (.003)	.755 (.005)
R-F(RF) w/ disc.	.749 (.024)	.673 (.051)	.669 (.027)	.689 (.020)	.875 (.011)	.878 (.011)	.769 (.003)	.758 (.006)
R-F(κNN)	.731 (.029)	.661 (.037)	.657 (.018)	.670 (.016)	.876 (.005)	.876 (.005)	.764 (.005)	.751 (.006)
R-F(κNN) w/ disc.	.731 (.025)	.670 (.042)	.674 (.027)	.684 (.026)	.876 (.011)	.878 (.011)	.763 (.007)	.752 (.006)

Table 48: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.906 (.017)	.906 (.017)	.710 (.031)	.710 (.031)	.748 (.061)	.748 (.061)	.705 (.005)	.705 (.005)
HC(LR) w/ disc.	.910 (.014)	.910 (.014)	.725 (.021)	.725 (.021)	.788 (.052)	.788 (.052)	.730 (.005)	.730 (.005)
HCApp(LR)	.910 (.014)	.910 (.014)	.684 (.044)	.684 (.044)	.745 (.064)	.745 (.064)	.672 (.032)	.672 (.032)
HCApp(LR) w/ disc.	.905 (.038)	.905 (.038)	.638 (.057)	.638 (.057)	.727 (.047)	.727 (.047)	.718 (.004)	.718 (.004)
HC(ANN)	.907 (.015)	.907 (.015)	.728 (.037)	.728 (.037)	.785 (.040)	.785 (.040)	.717 (.012)	.717 (.012)
HC(ANN) w/ disc.	.909 (.014)	.909 (.014)	.733 (.019)	.733 (.019)	.800 (.029)	.800 (.029)	.729 (.009)	.729 (.009)
IC-LR	.900 (.014)	.900 (.014)	.730 (.020)	.730 (.020)	.725 (.048)	.725 (.048)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.911 (.014)	.911 (.014)	.735 (.018)	.735 (.018)	**.803** (.029)	**.803** (.029)	.731 (.004)	.731 (.004)
IC-LR w/ neg.	.904 (.016)	.904 (.016)	**.751** (.018)	**.751** (.018)	.727 (.048)	.727 (.048)	.705 (.005)	.705 (.005)
Imp(LR)	.907 (.015)	.897 (.034)	.749 (.019)	.640 (.030)	.777 (.027)	.605 (.095)	.705 (.005)	.560 (.012)
Imp(LR) w/ disc.	.911 (.014)	.875 (.062)	.734 (.018)	.606 (.033)	**.803** (.029)	.482 (.124)	.731 (.004)	.500 (.000)
Imp(ANN)	.907 (.015)	.897 (.034)	.748 (.019)	.640 (.028)	.798 (.020)	.723 (.032)	.729 (.012)	.646 (.016)
Imp(ANN) w/ disc.	.911 (.014)	.875 (.062)	.734 (.017)	.608 (.034)	**.803** (.029)	.626 (.064)	.732 (.004)	.588 (.003)
Imp(RF)	.888 (.014)	.848 (.102)	.705 (.021)	.561 (.039)	.794 (.033)	.666 (.063)	**.737** (.004)	.535 (.046)
Imp(RF) w/ disc.	**.912** (.015)	.847 (.164)	.709 (.021)	.569 (.045)	.789 (.030)	.447 (.117)	.732 (.004)	.528 (.074)
Imp(κNN)	.902 (.018)	.842 (.106)	.712 (.025)	.507 (.020)	.739 (.031)	.502 (.004)	.708 (.019)	.471 (.042)
Imp(κNN) w/ disc.	.890 (.028)	.837 (.113)	.698 (.030)	.509 (.042)	.731 (.052)	.490 (.044)	.714 (.017)	.473 (.063)
R-F(LR)	.907 (.015)	.907 (.015)	.749 (.019)	.749 (.019)	.777 (.027)	.777 (.027)	.705 (.005)	.705 (.005)
R-F(LR) w/ disc.	.911 (.014)	.911 (.014)	.734 (.018)	.734 (.018)	**.803** (.029)	**.803** (.029)	.731 (.004)	.731 (.004)
R-F(ANN)	.907 (.015)	.907 (.014)	.748 (.019)	.748 (.019)	.797 (.020)	.798 (.019)	.723 (.013)	.725 (.014)
R-F(ANN) w/ disc.	.911 (.014)	.910 (.014)	.734 (.018)	.734 (.018)	**.803** (.028)	**.803** (.029)	.732 (.004)	.732 (.004)
R-F(RF)	.887 (.014)	.888 (.014)	.705 (.020)	.704 (.020)	.794 (.033)	.794 (.034)	**.737** (.004)	**.737** (.004)
R-F(RF) w/ disc.	**.912** (.014)	**.912** (.015)	.708 (.021)	.708 (.021)	.790 (.030)	.789 (.030)	.732 (.004)	.732 (.004)
R-F(κNN)	.902 (.016)	.903 (.016)	.712 (.024)	.713 (.023)	.738 (.030)	.742 (.031)	.710 (.015)	.710 (.013)
R-F(κNN) w/ disc.	.892 (.033)	.893 (.034)	.697 (.030)	.695 (.034)	.726 (.049)	.733 (.048)	.714 (.018)	.710 (.022)

Table 49: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.888 (.018)	.888 (.018)	.656 (.043)	.656 (.043)	.652 (.081)	.652 (.081)	.661 (.009)	.661 (.009)
HC(LR) w/ disc.	.888 (.015)	.888 (.015)	.669 (.046)	.669 (.046)	.696 (.069)	.696 (.069)	.670 (.034)	.670 (.034)
HCApp(LR)	.888 (.017)	.888 (.017)	.660 (.050)	.660 (.050)	.659 (.072)	.659 (.072)	.632 (.069)	.632 (.069)
HCApp(LR) w/ disc.	.890 (.013)	.890 (.013)	.645 (.067)	.645 (.067)	.664 (.102)	.664 (.102)	.676 (.025)	.676 (.025)
HC(ANN)	.884 (.014)	.884 (.014)	.663 (.036)	.663 (.036)	.644 (.092)	.644 (.092)	.675 (.010)	.675 (.010)
HC(ANN) w/ disc.	.874 (.015)	.874 (.015)	.651 (.045)	.651 (.045)	.707 (.048)	.707 (.048)	.691 (.014)	.691 (.014)
IC-LR	.875 (.013)	.875 (.013)	.705 (.021)	.705 (.021)	.709 (.042)	.709 (.042)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	**.893** (.011)	**.893** (.011)	.707 (.020)	.707 (.020)	.770 (.026)	**.770** (.026)	.711 (.005)	**.711** (.005)
IC-LR w/ neg.	**.893** (.013)	**.893** (.013)	**.728** (.020)	**.728** (.020)	.727 (.043)	.727 (.043)	.691 (.004)	.691 (.004)
Imp(LR)	.891 (.014)	.869 (.049)	**.728** (.022)	.628 (.030)	.764 (.029)	.652 (.082)	.693 (.005)	.563 (.003)
Imp(LR) w/ disc.	**.893** (.013)	.825 (.090)	.710 (.024)	.587 (.037)	.788 (.028)	.594 (.083)	.731 (.002)	.661 (.026)
Imp(ANN)	.891 (.014)	.868 (.049)	**.728** (.024)	.630 (.029)	.787 (.022)	.713 (.018)	.720 (.006)	.600 (.008)
Imp(ANN) w/ disc.	.891 (.013)	.826 (.084)	.711 (.025)	.589 (.036)	**.790** (.030)	.633 (.059)	**.732** (.002)	.676 (.018)
Imp(RF)	.863 (.017)	.827 (.071)	.674 (.031)	.568 (.040)	.777 (.034)	.612 (.051)	.717 (.003)	.608 (.044)
Imp(RF) w/ disc.	**.893** (.017)	.817 (.104)	.689 (.023)	.562 (.043)	.773 (.028)	.512 (.111)	**.732** (.003)	.673 (.019)
Imp(κNN)	.874 (.019)	.798 (.123)	.687 (.027)	.522 (.036)	.728 (.027)	.502 (.004)	.704 (.010)	.553 (.054)
Imp(κNN) w/ disc.	.874 (.023)	.763 (.143)	.673 (.029)	.523 (.037)	.722 (.038)	.497 (.036)	.702 (.016)	.513 (.064)
R-F(LR)	.882 (.014)	.798 (.060)	.698 (.021)	.590 (.059)	.745 (.028)	.563 (.106)	.698 (.003)	.487 (.121)
R-F(LR) w/ disc.	.885 (.014)	.781 (.065)	.699 (.020)	.628 (.049)	.780 (.027)	.676 (.063)	.724 (.004)	.552 (.125)
R-F(ANN)	.881 (.015)	.792 (.097)	.694 (.024)	.594 (.066)	.762 (.021)	.636 (.114)	.711 (.007)	.559 (.109)
R-F(ANN) w/ disc.	.881 (.014)	.806 (.053)	.691 (.023)	.603 (.066)	.764 (.031)	.700 (.055)	.725 (.004)	.642 (.076)
R-F(RF)	.861 (.015)	.791 (.055)	.670 (.025)	.602 (.052)	.758 (.033)	.531 (.035)	.717 (.003)	.539 (.067)
R-F(RF) w/ disc.	.879 (.016)	.803 (.046)	.677 (.026)	.616 (.055)	.739 (.031)	.576 (.069)	.725 (.004)	.474 (.127)
R-F(κNN)	.863 (.018)	.669 (.121)	.667 (.027)	.536 (.045)	.712 (.032)	.503 (.004)	.704 (.013)	.566 (.064)
R-F(κNN) w/ disc.	.862 (.021)	.654 (.119)	.665 (.031)	.549 (.057)	.701 (.046)	.501 (.004)	.702 (.013)	.536 (.039)

Table 50: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.870 (.015)	.870 (.015)	.643 (.036)	.643 (.036)	.621 (.090)	.621 (.090)	.640 (.011)	.640 (.011)
HC(LR) w/ disc.	.869 (.016)	.869 (.016)	.664 (.027)	.664 (.027)	.701 (.047)	.701 (.047)	.676 (.015)	.676 (.015)
HCApp(LR)	.869 (.018)	.869 (.018)	.641 (.045)	.641 (.045)	.603 (.098)	.603 (.098)	.633 (.018)	.633 (.018)
HCApp(LR) w/ disc.	.867 (.019)	.867 (.019)	.619 (.061)	.619 (.061)	.699 (.049)	.699 (.049)	.676 (.015)	.676 (.015)
HC(ANN)	.865 (.021)	.865 (.021)	.634 (.044)	.634 (.044)	.654 (.094)	.654 (.094)	.649 (.010)	.649 (.010)
HC(ANN) w/ disc.	.850 (.023)	.850 (.023)	.616 (.045)	.616 (.045)	.691 (.034)	.691 (.034)	.680 (.011)	.680 (.011)
IC-LR	.847 (.018)	.847 (.018)	.683 (.021)	.683 (.021)	.733 (.041)	.733 (.041)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.872 (.016)	.872 (.016)	.687 (.020)	.687 (.020)	.761 (.020)	**.761** (.020)	.696 (.006)	**.696** (.006)
IC-LR w/ neg.	**.878** (.017)	**.878** (.017)	**.710** (.022)	**.710** (.022)	.741 (.036)	.741 (.036)	.682 (.004)	.682 (.004)
Imp(LR)	.874 (.017)	.842 (.054)	.706 (.022)	.615 (.031)	.768 (.022)	.655 (.074)	.683 (.004)	.558 (.003)
Imp(LR) w/ disc.	.870 (.018)	.780 (.105)	.685 (.020)	.576 (.033)	.781 (.018)	.625 (.070)	.726 (.005)	.666 (.018)
Imp(ANN)	.875 (.017)	.843 (.052)	.706 (.022)	.616 (.031)	**.786** (.016)	.708 (.018)	.707 (.008)	.572 (.008)
Imp(ANN) w/ disc.	.868 (.018)	.783 (.097)	.687 (.019)	.577 (.035)	.785 (.018)	.644 (.058)	**.728** (.005)	.672 (.014)
Imp(RF)	.842 (.018)	.764 (.102)	.658 (.023)	.566 (.037)	.774 (.016)	.576 (.046)	.712 (.005)	.641 (.021)
Imp(RF) w/ disc.	.867 (.018)	.769 (.133)	.665 (.021)	.560 (.031)	.768 (.020)	.576 (.107)	.727 (.005)	.661 (.015)
Imp(κNN)	.858 (.019)	.773 (.123)	.666 (.024)	.525 (.039)	.722 (.028)	.504 (.004)	.698 (.011)	.589 (.048)
Imp(κNN) w/ disc.	.843 (.034)	.721 (.135)	.641 (.036)	.525 (.045)	.707 (.051)	.499 (.010)	.688 (.016)	.524 (.042)
R-F(LR)	.869 (.020)	.776 (.065)	.670 (.025)	.584 (.049)	.742 (.020)	.591 (.078)	.690 (.004)	.625 (.046)
R-F(LR) w/ disc.	.868 (.019)	.740 (.086)	.680 (.025)	.620 (.046)	.771 (.021)	.666 (.062)	.719 (.004)	.672 (.027)
R-F(ANN)	.863 (.018)	.781 (.064)	.662 (.029)	.587 (.055)	.755 (.016)	.655 (.085)	.700 (.005)	.649 (.020)
R-F(ANN) w/ disc.	.861 (.019)	.796 (.056)	.666 (.028)	.593 (.054)	.757 (.023)	.692 (.032)	.721 (.004)	.685 (.013)
R-F(RF)	.848 (.019)	.767 (.064)	.648 (.027)	.588 (.040)	.747 (.017)	.515 (.020)	.711 (.006)	.644 (.016)
R-F(RF) w/ disc.	.863 (.018)	.764 (.055)	.661 (.031)	.621 (.049)	.727 (.026)	.593 (.056)	.720 (.004)	.657 (.051)
R-F(κNN)	.837 (.022)	.651 (.107)	.645 (.031)	.521 (.032)	.709 (.023)	.509 (.013)	.696 (.011)	.619 (.026)
R-F(κNN) w/ disc.	.844 (.020)	.640 (.102)	.650 (.031)	.535 (.049)	.712 (.029)	.505 (.012)	.694 (.012)	.621 (.021)

Table 51: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.856 (.019)	.856 (.019)	.621 (.042)	.621 (.042)	.630 (.070)	.630 (.070)	.639 (.007)	.639 (.007)
HC(LR) w/ disc.	.850 (.019)	.850 (.019)	.643 (.036)	.643 (.036)	.652 (.077)	.652 (.077)	.668 (.010)	.668 (.010)
HCAPP(LR)	.856 (.020)	.856 (.020)	.614 (.050)	.614 (.050)	.638 (.065)	.638 (.065)	.633 (.016)	.633 (.016)
HCAPP(LR) w/ disc.	.850 (.019)	.850 (.019)	.612 (.058)	.612 (.058)	.654 (.081)	.654 (.081)	.668 (.010)	.668 (.010)
HC(ANN)	.844 (.023)	.844 (.023)	.604 (.040)	.604 (.040)	.681 (.056)	.681 (.056)	.642 (.008)	.642 (.008)
HC(ANN) w/ disc.	.831 (.028)	.831 (.028)	.594 (.048)	.594 (.048)	.657 (.039)	.657 (.039)	.666 (.012)	.666 (.012)
IC-LR	.822 (.022)	.822 (.022)	.660 (.028)	.660 (.028)	.724 (.048)	.724 (.048)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.853 (.017)	.853 (.017)	.662 (.026)	.662 (.026)	.732 (.024)	.732 (.024)	.679 (.005)	**.679** (.005)
IC-LR w/ neg.	**.866** (.015)	**.866** (.015)	**.687** (.023)	**.687** (.023)	.733 (.041)	**.733** (.041)	.673 (.006)	.673 (.006)
IMP(LR)	.861 (.017)	.823 (.050)	.680 (.023)	.595 (.023)	.755 (.027)	.665 (.040)	.673 (.005)	.559 (.015)
IMP(LR) w/ disc.	.854 (.016)	.770 (.077)	.659 (.026)	.562 (.024)	.758 (.025)	.590 (.073)	.716 (.004)	.659 (.014)
IMP(ANN)	.862 (.017)	.825 (.045)	.681 (.024)	.596 (.024)	**.772** (.022)	.695 (.020)	.694 (.006)	.563 (.019)
IMP(ANN) w/ disc.	.853 (.015)	.771 (.073)	.661 (.027)	.560 (.027)	.760 (.025)	.607 (.059)	**.718** (.004)	.662 (.012)
IMP(RF)	.826 (.020)	.765 (.086)	.630 (.026)	.554 (.031)	.749 (.027)	.614 (.048)	.703 (.005)	.651 (.012)
IMP(RF) w/ disc.	.850 (.021)	.769 (.075)	.646 (.026)	.549 (.028)	.750 (.025)	.578 (.079)	.717 (.004)	.657 (.011)
IMP(κNN)	.835 (.026)	.758 (.091)	.642 (.027)	.524 (.029)	.711 (.025)	.509 (.019)	.687 (.007)	.601 (.046)
IMP(κNN) w/ disc.	.831 (.022)	.749 (.070)	.616 (.047)	.518 (.035)	.693 (.052)	.503 (.011)	.680 (.015)	.550 (.052)
R-F(LR)	.854 (.020)	.801 (.042)	.647 (.021)	.584 (.045)	.724 (.028)	.626 (.083)	.680 (.004)	.638 (.008)
R-F(LR) w/ disc.	.848 (.019)	.774 (.050)	.664 (.027)	.613 (.042)	.744 (.029)	.625 (.095)	.708 (.004)	.671 (.010)
R-F(ANN)	.847 (.021)	.792 (.053)	.644 (.025)	.594 (.046)	.737 (.026)	.678 (.072)	.684 (.005)	.648 (.007)
R-F(ANN) w/ disc.	.841 (.022)	.796 (.036)	.636 (.040)	.586 (.039)	.721 (.029)	.654 (.049)	.709 (.004)	.674 (.007)
R-F(RF)	.824 (.024)	.760 (.051)	.623 (.027)	.581 (.049)	.722 (.026)	.551 (.033)	.700 (.005)	.649 (.013)
R-F(RF) w/ disc.	.841 (.025)	.780 (.043)	.649 (.031)	.607 (.044)	.705 (.030)	.550 (.088)	.709 (.004)	.670 (.010)
R-F(κNN)	.818 (.027)	.719 (.061)	.625 (.025)	.540 (.034)	.698 (.033)	.543 (.030)	.681 (.010)	.632 (.022)
R-F(κNN) w/ disc.	.818 (.026)	.705 (.074)	.639 (.030)	.551 (.046)	.687 (.028)	.554 (.042)	.678 (.016)	.632 (.019)

Table 52: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia Tru.	Australia Str.	Germany Tru.	Germany Str.	Poland Tru.	Poland Str.	Taiwan Tru.	Taiwan Str.
HC(LR)	.826 (.017)	.826 (.017)	.603 (.040)	.603 (.040)	.665 (.043)	.665 (.043)	.634 (.007)	.634 (.007)
HC(LR) w/ disc.	.820 (.017)	.820 (.017)	.617 (.033)	.617 (.033)	.687 (.035)	.687 (.035)	.662 (.007)	.662 (.007)
HCApp(LR)	.826 (.017)	.826 (.017)	.600 (.040)	.600 (.040)	.664 (.041)	.664 (.041)	.624 (.021)	.624 (.021)
HCApp(LR) w/ disc.	.820 (.017)	.820 (.017)	.603 (.049)	.603 (.049)	.686 (.035)	.686 (.035)	.662 (.007)	.662 (.007)
HC(ANN)	.806 (.023)	.806 (.023)	.581 (.043)	.581 (.043)	.683 (.030)	.683 (.030)	.638 (.006)	.638 (.006)
HC(ANN) w/ disc.	.800 (.025)	.800 (.025)	.577 (.041)	.577 (.041)	.655 (.039)	.655 (.039)	.661 (.008)	.661 (.008)
IC-LR	.788 (.019)	.788 (.019)	.641 (.027)	.641 (.027)	.729 (.033)	.729 (.033)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.829 (.017)	.829 (.017)	.645 (.030)	.645 (.030)	.722 (.018)	.722 (.018)	.668 (.004)	**.668** (.004)
IC-LR w/ neg.	**.846** (.017)	**.846** (.017)	**.673** (.024)	**.673** (.024)	.741 (.030)	**.741** (.030)	.665 (.005)	.665 (.005)
Imp(LR)	.840 (.020)	.788 (.054)	.671 (.021)	.592 (.028)	.756 (.025)	.673 (.035)	.665 (.005)	.558 (.014)
Imp(LR) w/ disc.	.825 (.020)	.724 (.082)	.649 (.024)	.556 (.030)	.757 (.022)	.621 (.039)	**.709** (.005)	.652 (.013)
Imp(ANN)	.839 (.021)	.790 (.054)	.672 (.021)	.592 (.028)	**.766** (.023)	.684 (.020)	.679 (.007)	.555 (.017)
Imp(ANN) w/ disc.	.823 (.021)	.724 (.080)	.651 (.024)	.555 (.030)	.759 (.022)	.627 (.039)	**.709** (.005)	.654 (.013)
Imp(RF)	.801 (.024)	.734 (.083)	.620 (.033)	.553 (.034)	.733 (.021)	.631 (.035)	.696 (.005)	.651 (.007)
Imp(RF) w/ disc.	.825 (.021)	.724 (.080)	.635 (.025)	.547 (.029)	.746 (.025)	.611 (.044)	**.709** (.005)	.651 (.012)
Imp(κNN)	.812 (.026)	.727 (.106)	.626 (.028)	.522 (.033)	.710 (.024)	.511 (.017)	.679 (.007)	.605 (.042)
Imp(κNN) w/ disc.	.791 (.039)	.645 (.122)	.602 (.040)	.507 (.032)	.686 (.036)	.498 (.027)	.673 (.017)	.545 (.048)
R-F(LR)	.825 (.019)	.797 (.033)	.621 (.024)	.584 (.033)	.717 (.024)	.663 (.044)	.670 (.004)	.634 (.007)
R-F(LR) w/ disc.	.822 (.020)	.777 (.037)	.637 (.030)	.601 (.037)	.734 (.022)	.681 (.036)	.696 (.005)	.663 (.007)
R-F(ANN)	.820 (.019)	.790 (.036)	.618 (.024)	.581 (.030)	.724 (.023)	.681 (.030)	.670 (.005)	.638 (.008)
R-F(ANN) w/ disc.	.818 (.021)	.782 (.035)	.608 (.039)	.569 (.041)	.709 (.026)	.652 (.038)	.697 (.005)	.664 (.006)
R-F(RF)	.802 (.021)	.751 (.049)	.607 (.028)	.584 (.037)	.699 (.025)	.585 (.040)	.687 (.005)	.648 (.007)
R-F(RF) w/ disc.	.820 (.022)	.780 (.036)	.628 (.029)	.598 (.033)	.700 (.025)	.605 (.068)	.697 (.005)	.662 (.007)
R-F(κNN)	.796 (.024)	.720 (.046)	.597 (.027)	.562 (.039)	.687 (.026)	.588 (.050)	.673 (.007)	.629 (.016)
R-F(κNN) w/ disc.	.800 (.021)	.731 (.043)	.613 (.031)	.539 (.040)	.680 (.029)	.585 (.053)	.668 (.010)	.625 (.030)

Table 53: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, unbalanced datasets, each classifier selects 4 best features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HC(LR)	.788 (.024)	.788 (.024)	.596 (.041)	.596 (.041)	.657 (.039)	.657 (.039)	.622 (.006)	.622 (.006)
HC(LR) w/ disc.	.779 (.023)	.779 (.023)	.598 (.035)	.598 (.035)	.674 (.028)	.674 (.028)	.643 (.006)	.643 (.006)
HCApp(LR)	.787 (.025)	.787 (.025)	.599 (.038)	.599 (.038)	.656 (.042)	.656 (.042)	.617 (.013)	.617 (.013)
HCApp(LR) w/ disc.	.779 (.023)	.779 (.023)	.593 (.040)	.593 (.040)	.675 (.027)	.675 (.027)	.643 (.006)	.643 (.006)
HC(ANN)	.777 (.027)	.777 (.027)	.575 (.037)	.575 (.037)	.659 (.033)	.659 (.033)	.624 (.004)	.624 (.004)
HC(ANN) w/ disc.	.770 (.025)	.770 (.025)	.556 (.030)	.556 (.030)	.631 (.040)	.631 (.040)	.643 (.006)	.643 (.006)
IC-LR	.751 (.022)	.751 (.022)	.618 (.025)	.618 (.025)	.722 (.029)	.722 (.029)	.500 (.000)	.500 (.000)
IC-LR w/ disc.	.796 (.022)	.796 (.022)	.620 (.028)	.620 (.028)	.697 (.023)	.697 (.023)	.651 (.005)	.651 (.005)
IC-LR w/ neg.	**.819** (.021)	**.819** (.021)	**.643** (.026)	**.643** (.026)	.731 (.028)	**.731** (.028)	.652 (.005)	**.652** (.005)
Imp(LR)	.809 (.023)	.729 (.063)	.641 (.027)	.571 (.026)	.742 (.018)	.665 (.029)	.652 (.005)	.554 (.012)
Imp(LR) w/ disc.	.794 (.021)	.658 (.082)	.622 (.027)	.548 (.029)	.731 (.022)	.594 (.049)	**.696** (.006)	.632 (.014)
Imp(ANN)	.809 (.022)	.732 (.060)	.641 (.026)	.572 (.027)	**.754** (.016)	.676 (.020)	.661 (.006)	.552 (.015)
Imp(ANN) w/ disc.	.792 (.022)	.659 (.080)	.621 (.027)	.544 (.032)	.733 (.023)	.596 (.048)	**.696** (.006)	.633 (.013)
Imp(RF)	.781 (.028)	.683 (.078)	.602 (.031)	.544 (.030)	.715 (.021)	.621 (.028)	.684 (.006)	.640 (.006)
Imp(RF) w/ disc.	.790 (.023)	.659 (.080)	.606 (.026)	.536 (.027)	.726 (.023)	.591 (.048)	.695 (.006)	.631 (.013)
Imp(κNN)	.782 (.031)	.681 (.078)	.598 (.040)	.512 (.032)	.689 (.028)	.506 (.014)	.663 (.011)	.574 (.049)
Imp(κNN) w/ disc.	.757 (.036)	.600 (.105)	.578 (.037)	.506 (.023)	.675 (.032)	.497 (.030)	.656 (.016)	.553 (.047)
R-F(LR)	.790 (.024)	.765 (.038)	.599 (.029)	.576 (.036)	.696 (.022)	.655 (.039)	.651 (.006)	.622 (.005)
R-F(LR) w/ disc.	.785 (.025)	.746 (.045)	.615 (.031)	.586 (.034)	.710 (.022)	.672 (.028)	.673 (.006)	.645 (.006)
R-F(ANN)	.784 (.026)	.755 (.041)	.595 (.030)	.567 (.032)	.692 (.026)	.662 (.030)	.650 (.006)	.623 (.005)
R-F(ANN) w/ disc.	.783 (.025)	.751 (.036)	.584 (.033)	.557 (.034)	.678 (.026)	.636 (.039)	.674 (.006)	.646 (.006)
R-F(RF)	.774 (.024)	.732 (.034)	.591 (.030)	.573 (.031)	.673 (.019)	.581 (.028)	.665 (.007)	.634 (.006)
R-F(RF) w/ disc.	.787 (.024)	.757 (.034)	.607 (.031)	.580 (.030)	.685 (.026)	.636 (.036)	.673 (.006)	.645 (.005)
R-F(κNN)	.769 (.028)	.728 (.033)	.574 (.032)	.557 (.034)	.669 (.024)	.592 (.035)	.652 (.008)	.615 (.012)
R-F(κNN) w/ disc.	.767 (.023)	.727 (.038)	.594 (.032)	.543 (.037)	.659 (.023)	.586 (.039)	.649 (.007)	.615 (.015)

Table 54: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.848 (.016)	.848 (.016)	.655 (.040)	.655 (.040)	.662 (.068)	.662 (.068)	.680 (.013)	.680 (.013)
HCApp(LR) w/ disc.	.851 (.038)	.851 (.038)	.648 (.063)	.648 (.063)	.741 (.045)	.741 (.045)	.651 (.035)	.651 (.035)
Mincut	.498 (.019)	.498 (.019)	.498 (.020)	.498 (.020)	.500 (.017)	.500 (.017)	.499 (.005)	.499 (.005)
Mincut w/ disc.	.562 (.030)	.562 (.030)	.504 (.020)	.504 (.020)	.611 (.032)	.611 (.032)	.608 (.011)	.608 (.011)
IC-LR	**.862** (.015)	**.862** (.015)	.691 (.024)	.691 (.024)	.628 (.048)	.628 (.048)	.587 (.024)	.587 (.024)
IC-LR w/ disc.	.857 (.016)	.857 (.016)	.690 (.023)	.690 (.023)	**.817** (.017)	**.817** (.017)	**.705** (.005)	**.705** (.005)
IC-LR w/ neg.	.860 (.015)	.860 (.015)	**.693** (.021)	**.693** (.021)	.565 (.063)	.565 (.063)	.669 (.008)	.669 (.008)
Imp(LR)	.851 (.016)	.768 (.071)	.687 (.023)	.524 (.043)	.746 (.022)	.508 (.032)	.669 (.006)	.501 (.005)
Imp(LR) w/ disc.	.856 (.014)	.829 (.034)	.688 (.021)	.556 (.067)	.814 (.018)	.505 (.026)	**.705** (.005)	.501 (.005)

Table 55: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCAPP(LR)	.808 (.022)	.808 (.022)	.640 (.029)	.640 (.029)	.658 (.042)	.658 (.042)	.658 (.007)	.658 (.007)
HCAPP(LR) w/ disc.	.812 (.040)	.812 (.040)	.634 (.052)	.634 (.052)	.734 (.024)	.734 (.024)	.669 (.006)	.669 (.006)
MINCUT	.497 (.020)	.497 (.020)	.498 (.019)	.498 (.019)	.501 (.018)	.501 (.018)	.500 (.004)	.500 (.004)
MINCUT w/ disc.	.542 (.026)	.542 (.026)	.502 (.019)	.502 (.019)	.503 (.018)	.503 (.018)	.560 (.008)	.560 (.008)
IC-LR	.833 (.018)	.833 (.018)	.666 (.029)	.666 (.029)	.609 (.064)	.609 (.064)	.592 (.008)	.592 (.008)
IC-LR w/ disc.	.831 (.021)	.831 (.021)	.666 (.025)	.666 (.025)	.788 (.016)	**.788** (.016)	.684 (.006)	**.684** (.006)
IC-LR w/ neg.	**.839** (.020)	**.839** (.020)	.670 (.023)	**.670** (.023)	.550 (.055)	.550 (.055)	.666 (.007)	.666 (.007)
IMP(LR)	.822 (.021)	.738 (.051)	**.672** (.028)	.522 (.037)	.739 (.024)	.507 (.033)	.665 (.005)	.500 (.004)
IMP(LR) w/ disc.	.832 (.021)	.774 (.047)	.668 (.024)	.555 (.061)	**.795** (.018)	.501 (.019)	**.701** (.004)	.500 (.004)

Table 56: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCAPP(LR)	.778 (.026)	.778 (.026)	.616 (.038)	.616 (.038)	.656 (.030)	.656 (.030)	.639 (.008)	.639 (.008)
HCAPP(LR) w/ disc.	.785 (.042)	.785 (.042)	.620 (.052)	.620 (.052)	.727 (.021)	.727 (.021)	.648 (.032)	.648 (.032)
MINCUT	.498 (.022)	.498 (.022)	.501 (.022)	.501 (.022)	.498 (.018)	.498 (.018)	.500 (.004)	.500 (.004)
MINCUT w/ disc.	.535 (.030)	.535 (.030)	.504 (.023)	.504 (.023)	.498 (.017)	.498 (.017)	.550 (.007)	.550 (.007)
IC-LR	.801 (.021)	.801 (.021)	.653 (.027)	.653 (.027)	.603 (.064)	.603 (.064)	.585 (.009)	.585 (.009)
IC-LR w/ disc.	.806 (.020)	**.806** (.020)	.651 (.031)	.651 (.031)	.773 (.018)	**.773** (.018)	.669 (.004)	**.669** (.004)
IC-LR w/ neg.	.806 (.023)	**.806** (.023)	.661 (.028)	**.661** (.028)	.538 (.068)	.538 (.068)	.663 (.006)	.663 (.006)
IMP(LR)	.803 (.020)	.700 (.062)	**.662** (.030)	.520 (.043)	.729 (.020)	.508 (.025)	.659 (.005)	.500 (.004)
IMP(LR) w/ disc.	**.811** (.019)	.722 (.075)	.657 (.027)	.549 (.065)	**.782** (.018)	.503 (.018)	**.698** (.005)	.500 (.004)

Table 57: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCAPP(LR)	.752 (.022)	.752 (.022)	.589 (.038)	.589 (.038)	.655 (.042)	.655 (.042)	.631 (.007)	.631 (.007)
HCAPP(LR) w/ disc.	.757 (.037)	.757 (.037)	.607 (.043)	.607 (.043)	.715 (.039)	.715 (.039)	.627 (.043)	.627 (.043)
MINCUT	.503 (.020)	.503 (.020)	.502 (.020)	.502 (.020)	.499 (.016)	.499 (.016)	.498 (.004)	.498 (.004)
MINCUT w/ disc.	.546 (.030)	.546 (.030)	.508 (.022)	.508 (.022)	.499 (.016)	.499 (.016)	.543 (.007)	.543 (.007)
IC-LR	.776 (.019)	.776 (.019)	.635 (.026)	.635 (.026)	.607 (.074)	.607 (.074)	.584 (.005)	.584 (.005)
IC-LR w/ disc.	**.784** (.021)	**.784** (.021)	.634 (.026)	.634 (.026)	.757 (.021)	**.757** (.021)	.655 (.006)	**.655** (.006)
IC-LR w/ neg.	.781 (.017)	.781 (.017)	**.647** (.032)	**.647** (.032)	.530 (.054)	.530 (.054)	.652 (.007)	.652 (.007)
IMP(LR)	.779 (.019)	.688 (.051)	**.647** (.030)	.517 (.038)	.725 (.021)	.505 (.020)	.649 (.006)	.502 (.004)
IMP(LR) w/ disc.	**.784** (.020)	.685 (.086)	.640 (.027)	.537 (.055)	**.771** (.024)	.503 (.015)	**.691** (.004)	.502 (.004)

Table 58: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.724 (.031)	.724 (.031)	.571 (.038)	.571 (.038)	.643 (.034)	.643 (.034)	.622 (.008)	.622 (.008)
HCApp(LR) w/ disc.	.732 (.037)	.732 (.037)	.582 (.042)	.582 (.042)	.708 (.038)	.708 (.038)	.614 (.044)	.614 (.044)
Mincut	.503 (.021)	.503 (.021)	.496 (.020)	.496 (.020)	.500 (.015)	.500 (.015)	.500 (.004)	.500 (.004)
Mincut w/ disc.	.550 (.035)	.550 (.035)	.502 (.021)	.502 (.021)	.500 (.015)	.500 (.015)	.544 (.007)	.544 (.007)
IC-LR	.747 (.026)	.747 (.026)	.616 (.028)	.616 (.028)	.597 (.075)	.597 (.075)	.579 (.007)	.579 (.007)
IC-LR w/ disc.	.760 (.025)	**.760** (.025)	.613 (.030)	.613 (.030)	.747 (.026)	**.747** (.026)	.640 (.008)	.640 (.008)
IC-LR w/ neg.	.746 (.038)	.746 (.038)	.623 (.025)	**.623** (.025)	.522 (.058)	.522 (.058)	.648 (.009)	**.648** (.009)
Imp(LR)	**.765** (.023)	.670 (.050)	**.628** (.026)	.531 (.042)	.719 (.023)	.507 (.022)	.646 (.006)	.502 (.012)
Imp(LR) w/ disc.	.763 (.025)	.660 (.090)	.621 (.025)	.550 (.052)	**.763** (.019)	.502 (.016)	**.684** (.004)	.500 (.004)

Table 59: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.703 (.028)	.703 (.028)	.566 (.030)	.566 (.030)	.634 (.043)	.634 (.043)	.616 (.008)	.616 (.008)
HCApp(LR) w/ disc.	.704 (.042)	.704 (.042)	.572 (.044)	.572 (.044)	.694 (.036)	.694 (.036)	.621 (.007)	.621 (.007)
Mincut	.508 (.023)	.508 (.023)	.499 (.022)	.499 (.022)	.498 (.018)	.498 (.018)	.500 (.004)	.500 (.004)
Mincut w/ disc.	.564 (.038)	.564 (.038)	.509 (.027)	.509 (.027)	.498 (.018)	.498 (.018)	.546 (.007)	.546 (.007)
IC-LR	.723 (.025)	.723 (.025)	.606 (.030)	.606 (.030)	.590 (.071)	.590 (.071)	.571 (.007)	.571 (.007)
IC-LR w/ disc.	.735 (.024)	**.735** (.024)	.606 (.028)	.606 (.028)	.738 (.019)	**.738** (.019)	.627 (.010)	.627 (.010)
IC-LR w/ neg.	.712 (.044)	.712 (.044)	.607 (.036)	**.607** (.036)	.508 (.044)	.508 (.044)	.642 (.008)	**.642** (.008)
Imp(LR)	**.737** (.021)	.653 (.057)	**.615** (.027)	.523 (.039)	.712 (.022)	.511 (.025)	.639 (.007)	.502 (.009)
Imp(LR) w/ disc.	.730 (.023)	.649 (.083)	.610 (.027)	.542 (.049)	**.758** (.019)	.513 (.026)	**.674** (.005)	.500 (.004)

Table 60: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.901 (.013)	.901 (.013)	.703 (.037)	.703 (.037)	.697 (.090)	.697 (.090)	.693 (.018)	.693 (.018)
HCApp(LR) w/ disc.	.907 (.045)	.907 (.045)	.700 (.065)	.700 (.065)	.791 (.052)	.791 (.052)	.675 (.040)	.675 (.040)
IC-LR	.911 (.013)	.911 (.013)	.751 (.024)	.751 (.024)	.705 (.037)	.705 (.037)	.634 (.007)	.634 (.007)
IC-LR w/ disc.	.921 (.011)	**.921** (.011)	.753 (.024)	.753 (.024)	**.895** (.012)	**.895** (.012)	**.768** (.005)	**.768** (.005)
IC-LR w/ neg.	.913 (.012)	.913 (.012)	**.761** (.026)	**.761** (.026)	.691 (.026)	.691 (.026)	.719 (.007)	.719 (.007)
Imp(LR)	.910 (.012)	.892 (.016)	.752 (.025)	.687 (.047)	.801 (.021)	.654 (.062)	.721 (.006)	.619 (.013)
Imp(LR) w/ disc.	**.922** (.011)	.919 (.012)	.748 (.024)	.691 (.050)	**.895** (.012)	.797 (.069)	**.768** (.005)	.550 (.037)

Table 61: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.878 (.016)	.878 (.016)	.681 (.033)	.681 (.033)	.686 (.056)	.686 (.056)	.679 (.009)	.679 (.009)
HCApp(LR) w/ disc.	.888 (.043)	.888 (.043)	.677 (.056)	.677 (.056)	.791 (.025)	.791 (.025)	.697 (.010)	.697 (.010)
IC-LR	.892 (.014)	.892 (.014)	.723 (.025)	.723 (.025)	.702 (.041)	.702 (.041)	.629 (.006)	.629 (.006)
IC-LR w/ disc.	.904 (.013)	**.904** (.013)	.723 (.025)	.723 (.025)	.869 (.017)	**.869** (.017)	.743 (.006)	**.743** (.006)
IC-LR w/ neg.	.901 (.014)	.901 (.014)	**.732** (.019)	**.732** (.019)	.680 (.045)	.680 (.045)	.715 (.005)	.715 (.005)
Imp(LR)	.894 (.015)	.877 (.016)	.729 (.029)	.671 (.049)	.792 (.024)	.636 (.058)	.719 (.005)	.637 (.010)
Imp(LR) w/ disc.	**.907** (.013)	.890 (.017)	.727 (.025)	.677 (.048)	**.874** (.017)	.680 (.119)	**.764** (.005)	.697 (.021)

Table 62: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.849 (.024)	.849 (.024)	.656 (.039)	.656 (.039)	.688 (.037)	.688 (.037)	.662 (.007)	.662 (.007)
HCApp(LR) w/ disc.	.864 (.043)	.864 (.043)	.663 (.056)	.663 (.056)	.785 (.025)	.785 (.025)	.682 (.038)	.682 (.038)
IC-LR	.865 (.018)	.865 (.018)	.706 (.029)	.706 (.029)	.692 (.037)	.692 (.037)	.622 (.006)	.622 (.006)
IC-LR w/ disc.	.883 (.016)	**.883** (.016)	.707 (.029)	.707 (.029)	.850 (.018)	**.850** (.018)	.727 (.004)	**.727** (.004)
IC-LR w/ neg.	.880 (.017)	.880 (.017)	**.722** (.028)	**.722** (.028)	.686 (.045)	.686 (.045)	.711 (.004)	.711 (.004)
Imp(LR)	.878 (.018)	.854 (.027)	.714 (.030)	.662 (.044)	.785 (.019)	.649 (.050)	.713 (.004)	.637 (.008)
Imp(LR) w/ disc.	**.890** (.016)	.861 (.026)	.713 (.030)	.666 (.049)	**.859** (.015)	.634 (.121)	**.758** (.004)	.697 (.011)

Table 63: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.826 (.022)	.826 (.022)	.628 (.040)	.628 (.040)	.690 (.048)	.690 (.048)	.653 (.007)	.653 (.007)
HCApp(LR) w/ disc.	.839 (.042)	.839 (.042)	.647 (.047)	.647 (.047)	.768 (.044)	.768 (.044)	.664 (.054)	.664 (.054)
IC-LR	.839 (.018)	.839 (.018)	.685 (.027)	.685 (.027)	.713 (.041)	.713 (.041)	.618 (.006)	.618 (.006)
IC-LR w/ disc.	.861 (.016)	**.861** (.016)	.684 (.028)	.684 (.028)	.836 (.020)	**.836** (.020)	.711 (.005)	**.711** (.005)
IC-LR w/ neg.	.861 (.018)	**.861** (.018)	**.708** (.026)	**.708** (.026)	.698 (.052)	.698 (.052)	.702 (.007)	.702 (.007)
Imp(LR)	.860 (.018)	.832 (.023)	.700 (.030)	.649 (.041)	.781 (.021)	.636 (.046)	.706 (.005)	.635 (.007)
Imp(LR) w/ disc.	**.868** (.017)	.831 (.021)	.694 (.027)	.649 (.044)	**.850** (.022)	.603 (.091)	**.751** (.005)	.689 (.008)

Table 64: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.796 (.030)	.796 (.030)	.599 (.043)	.599 (.043)	.678 (.046)	.678 (.046)	.645 (.008)	.645 (.008)
HCApp(LR) w/ disc.	.812 (.042)	.812 (.042)	.615 (.046)	.615 (.046)	.765 (.045)	.765 (.045)	.650 (.056)	.650 (.056)
IC-LR	.812 (.024)	.812 (.024)	.658 (.032)	.658 (.032)	.718 (.044)	.718 (.044)	.610 (.007)	.610 (.007)
IC-LR w/ disc.	.836 (.023)	**.836** (.023)	.658 (.033)	.658 (.033)	.822 (.023)	**.822** (.023)	.696 (.007)	**.696** (.007)
IC-LR w/ neg.	.835 (.021)	.835 (.021)	**.677** (.026)	**.677** (.026)	.713 (.045)	.713 (.045)	.695 (.006)	.695 (.006)
Imp(LR)	.844 (.022)	.810 (.033)	.674 (.031)	.633 (.041)	.771 (.026)	.638 (.044)	.699 (.006)	.627 (.008)
Imp(LR) w/ disc.	**.847** (.021)	.795 (.036)	.670 (.030)	.633 (.043)	**.838** (.019)	.600 (.091)	**.742** (.005)	.676 (.009)

Table 65: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, balanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.771 (.028)	.771 (.028)	.590 (.033)	.590 (.033)	.674 (.049)	.674 (.049)	.639 (.007)	.639 (.007)
HCApp(LR) w/ disc.	.783 (.041)	.783 (.041)	.604 (.049)	.604 (.049)	.748 (.042)	.748 (.042)	.661 (.006)	.661 (.006)
IC-LR	.784 (.022)	.784 (.022)	.648 (.031)	.648 (.031)	.721 (.031)	.721 (.031)	.602 (.007)	.602 (.007)
IC-LR w/ disc.	.810 (.020)	.810 (.020)	.649 (.029)	.649 (.029)	.811 (.022)	**.811** (.022)	.681 (.005)	.681 (.005)
IC-LR w/ neg.	.814 (.018)	**.814** (.018)	**.669** (.028)	**.669** (.028)	.700 (.046)	.700 (.046)	.687 (.006)	**.687** (.006)
Imp(LR)	**.818** (.019)	.786 (.026)	.657 (.032)	.618 (.044)	.766 (.024)	.639 (.042)	.692 (.007)	.624 (.007)
Imp(LR) w/ disc.	**.818** (.018)	.759 (.036)	.654 (.031)	.618 (.044)	**.828** (.020)	.650 (.065)	**.731** (.006)	.663 (.009)

Table 66: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.0$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.825 (.072)	.825 (.072)	.709 (.021)	.709 (.021)	.931 (.003)	.931 (.003)	.794 (.016)	.794 (.016)
HCApp(LR) w/ disc.	.838 (.055)	.838 (.055)	.702 (.018)	.702 (.018)	.933 (.006)	.933 (.006)	.798 (.022)	.798 (.022)
Mincut	.444 (.016)	.444 (.016)	.299 (.016)	.299 (.016)	.078 (.004)	.078 (.004)	.222 (.003)	.222 (.003)
Mincut w/ disc.	.523 (.031)	.523 (.031)	.316 (.017)	.316 (.017)	.364 (.040)	.364 (.040)	.550 (.020)	.550 (.020)
IC-LR	**.858** (.013)	**.858** (.013)	.733 (.017)	.733 (.017)	.931 (.003)	.931 (.003)	.779 (.003)	.779 (.003)
IC-LR w/ disc.	**.858** (.014)	**.858** (.014)	.736 (.016)	.736 (.016)	.948 (.003)	**.948** (.003)	**.819** (.002)	**.819** (.002)
IC-LR w/ neg.	**.858** (.012)	**.858** (.012)	**.746** (.016)	**.746** (.016)	.931 (.003)	.931 (.003)	.801 (.005)	.801 (.005)
Imp(LR)	.850 (.014)	.733 (.093)	.743 (.017)	.701 (.016)	.930 (.004)	.931 (.003)	.811 (.003)	.779 (.003)
Imp(LR) w/ disc.	.855 (.013)	.786 (.089)	.735 (.018)	.701 (.016)	**.952** (.004)	.931 (.003)	**.819** (.002)	.779 (.003)

Table 67: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.1$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.804 (.024)	.804 (.024)	.696 (.018)	.696 (.018)	.929 (.003)	.929 (.003)	.791 (.012)	.791 (.012)
HCApp(LR) w/ disc.	.809 (.021)	.809 (.021)	.694 (.013)	.694 (.013)	.930 (.003)	.930 (.003)	.792 (.015)	.792 (.015)
Mincut	.447 (.014)	.447 (.014)	.303 (.012)	.303 (.012)	.071 (.002)	.071 (.002)	.224 (.002)	.224 (.002)
Mincut w/ disc.	.500 (.025)	.500 (.025)	.318 (.015)	.318 (.015)	.085 (.006)	.085 (.006)	.459 (.027)	.459 (.027)
IC-LR	.828 (.021)	.828 (.021)	.714 (.016)	.714 (.016)	.929 (.003)	.929 (.003)	.776 (.002)	.776 (.002)
IC-LR w/ disc.	**.838** (.015)	**.838** (.015)	.713 (.018)	.713 (.018)	.938 (.003)	**.938** (.003)	.805 (.002)	**.805** (.002)
IC-LR w/ neg.	.831 (.018)	.831 (.018)	.728 (.017)	**.728** (.017)	.929 (.002)	.929 (.002)	.798 (.009)	.798 (.009)
Imp(LR)	.826 (.015)	.684 (.072)	**.733** (.014)	.697 (.012)	.929 (.004)	.929 (.002)	.805 (.002)	.776 (.002)
Imp(LR) w/ disc.	.836 (.016)	.709 (.084)	.724 (.017)	.697 (.012)	**.945** (.002)	.929 (.002)	**.815** (.002)	.776 (.002)

Table 68: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.2$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.775 (.023)	.775 (.023)	.692 (.016)	.692 (.016)	.931 (.003)	.931 (.003)	.789 (.010)	.789 (.010)
HCApp(LR) w/ disc.	.786 (.018)	.786 (.018)	.695 (.015)	.695 (.015)	.931 (.003)	.931 (.003)	.789 (.009)	.789 (.009)
Mincut	.450 (.023)	.450 (.023)	.303 (.014)	.303 (.014)	.069 (.003)	.069 (.003)	.221 (.003)	.221 (.003)
Mincut w/ disc.	.495 (.024)	.495 (.024)	.321 (.019)	.321 (.019)	.070 (.003)	.070 (.003)	.424 (.038)	.424 (.038)
IC-LR	.793 (.023)	.793 (.023)	.709 (.018)	.709 (.018)	.931 (.003)	.931 (.003)	.779 (.003)	.779 (.003)
IC-LR w/ disc.	.804 (.022)	**.804** (.022)	.712 (.017)	.712 (.017)	.936 (.003)	**.936** (.003)	.799 (.003)	.799 (.003)
IC-LR w/ neg.	.798 (.018)	.798 (.018)	**.725** (.018)	**.725** (.018)	.931 (.003)	.931 (.003)	.803 (.003)	**.803** (.003)
Imp(LR)	.802 (.019)	.662 (.088)	**.725** (.017)	.698 (.014)	.929 (.003)	.931 (.003)	.805 (.004)	.779 (.003)
Imp(LR) w/ disc.	**.808** (.019)	.672 (.102)	.720 (.017)	.698 (.014)	**.943** (.004)	.931 (.003)	**.813** (.003)	.779 (.003)

Table 69: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.3$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.756 (.022)	.756 (.022)	.688 (.014)	.688 (.014)	.931 (.004)	.931 (.004)	.785 (.006)	.785 (.006)
HCApp(LR) w/ disc.	.768 (.027)	.768 (.027)	.696 (.016)	.696 (.016)	.931 (.004)	.931 (.004)	.787 (.010)	.787 (.010)
Mincut	.445 (.023)	.445 (.023)	.299 (.013)	.299 (.013)	.069 (.004)	.069 (.004)	.221 (.003)	.221 (.003)
Mincut w/ disc.	.509 (.032)	.509 (.032)	.324 (.015)	.324 (.015)	.069 (.004)	.069 (.004)	.468 (.019)	.468 (.019)
IC-LR	.770 (.023)	.770 (.023)	.700 (.017)	.700 (.017)	.931 (.004)	.931 (.004)	.779 (.003)	.779 (.003)
IC-LR w/ disc.	.782 (.022)	**.782** (.022)	.703 (.014)	.703 (.014)	.933 (.005)	**.933** (.005)	.794 (.004)	.794 (.004)
IC-LR w/ neg.	.764 (.035)	.764 (.035)	**.716** (.018)	**.716** (.018)	.931 (.004)	.931 (.004)	.798 (.004)	**.798** (.004)
Imp(LR)	.788 (.024)	.651 (.080)	**.716** (.019)	.701 (.013)	.929 (.005)	.931 (.004)	.800 (.003)	.779 (.003)
Imp(LR) w/ disc.	**.792** (.024)	.650 (.089)	.713 (.017)	.701 (.013)	**.941** (.004)	.931 (.004)	**.810** (.002)	.779 (.003)

Table 70: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.4$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.733 (.026)	.733 (.026)	.676 (.013)	.676 (.013)	.929 (.004)	.929 (.004)	.779 (.002)	.779 (.002)
HCApp(LR) w/ disc.	.736 (.026)	.736 (.026)	.691 (.020)	.691 (.020)	.930 (.003)	.930 (.003)	.785 (.009)	.785 (.009)
Mincut	.439 (.014)	.439 (.014)	.305 (.015)	.305 (.015)	.070 (.003)	.070 (.003)	.221 (.002)	.221 (.002)
Mincut w/ disc.	.514 (.036)	.514 (.036)	.350 (.024)	.350 (.024)	.070 (.003)	.070 (.003)	.507 (.022)	.507 (.022)
IC-LR	.744 (.019)	.744 (.019)	.690 (.017)	.690 (.017)	.930 (.003)	.930 (.003)	.779 (.002)	.779 (.002)
IC-LR w/ disc.	.762 (.019)	**.762** (.019)	.689 (.018)	.689 (.018)	.932 (.003)	**.932** (.003)	.788 (.003)	.788 (.003)
IC-LR w/ neg.	.747 (.026)	.747 (.026)	.704 (.016)	**.704** (.016)	.930 (.003)	.930 (.003)	.795 (.002)	**.795** (.002)
Imp(LR)	**.769** (.020)	.637 (.064)	**.709** (.012)	.696 (.015)	.927 (.003)	.930 (.003)	.797 (.002)	.779 (.002)
Imp(LR) w/ disc.	.766 (.019)	.619 (.064)	.704 (.014)	.696 (.015)	**.938** (.003)	.930 (.003)	**.806** (.003)	.779 (.002)

Table 71: Our methods vs. the rest: mean classifier accuracy for $\epsilon = 0.5$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.696 (.033)	.696 (.033)	.669 (.019)	.669 (.019)	.930 (.003)	.930 (.003)	.779 (.002)	.779 (.002)
HCApp(LR) w/ disc.	.707 (.047)	.707 (.047)	.695 (.016)	.695 (.016)	.931 (.003)	.931 (.003)	.780 (.002)	.780 (.002)
Mincut	.457 (.023)	.457 (.023)	.304 (.016)	.304 (.016)	.068 (.003)	.068 (.003)	.222 (.002)	.222 (.002)
Mincut w/ disc.	.542 (.038)	.542 (.038)	.377 (.030)	.377 (.030)	.068 (.003)	.068 (.003)	.567 (.009)	.567 (.009)
IC-LR	.716 (.025)	.716 (.025)	.686 (.018)	.686 (.018)	.931 (.002)	.931 (.002)	.780 (.002)	.780 (.002)
IC-LR w/ disc.	.730 (.028)	**.730** (.028)	.689 (.018)	.689 (.018)	.934 (.002)	**.934** (.002)	.783 (.003)	.783 (.003)
IC-LR w/ neg.	.716 (.040)	.716 (.040)	**.699** (.022)	**.699** (.022)	.932 (.002)	.932 (.002)	.792 (.003)	**.792** (.003)
Imp(LR)	**.751** (.022)	.606 (.067)	**.699** (.018)	.698 (.017)	.929 (.003)	.932 (.003)	.795 (.002)	.780 (.002)
Imp(LR) w/ disc.	.741 (.016)	.596 (.072)	**.699** (.018)	.698 (.017)	**.937** (.003)	.932 (.003)	**.804** (.002)	.780 (.002)

Table 72: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.0$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.881 (.077)	.881 (.077)	.616 (.069)	.616 (.069)	.635 (.142)	.635 (.142)	.639 (.059)	.639 (.059)
HCApp(LR) w/ disc.	.900 (.067)	.900 (.067)	.597 (.067)	.597 (.067)	.703 (.080)	.703 (.080)	.649 (.081)	.649 (.081)
IC-LR	.915 (.010)	.915 (.010)	.758 (.017)	.758 (.017)	.701 (.024)	.701 (.024)	.631 (.004)	.631 (.004)
IC-LR w/ disc.	.924 (.010)	**.924** (.010)	.762 (.017)	.762 (.017)	.911 (.010)	**.911** (.010)	**.770** (.003)	**.770** (.003)
IC-LR w/ neg.	.917 (.010)	.917 (.010)	**.774** (.018)	**.774** (.018)	.708 (.028)	.708 (.028)	.718 (.003)	.718 (.003)
Imp(LR)	.915 (.011)	.867 (.065)	.767 (.020)	.657 (.028)	.796 (.025)	.412 (.045)	.724 (.003)	.594 (.006)
Imp(LR) w/ disc.	**.926** (.010)	.904 (.041)	.758 (.017)	.653 (.024)	**.913** (.009)	.422 (.154)	**.770** (.003)	.562 (.014)

Table 73: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.1$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.878 (.020)	.878 (.020)	.599 (.062)	.599 (.062)	.529 (.121)	.529 (.121)	.634 (.042)	.634 (.042)
HCApp(LR) w/ disc.	.892 (.013)	.892 (.013)	.563 (.072)	.563 (.072)	.620 (.169)	.620 (.169)	.638 (.076)	.638 (.076)
IC-LR	.893 (.013)	.893 (.013)	.731 (.020)	.731 (.020)	.684 (.024)	.684 (.024)	.627 (.006)	.627 (.006)
IC-LR w/ disc.	.907 (.011)	**.907** (.011)	.731 (.018)	.731 (.018)	.878 (.007)	**.878** (.007)	.748 (.001)	**.748** (.001)
IC-LR w/ neg.	.900 (.011)	.900 (.011)	**.748** (.019)	**.748** (.019)	.700 (.028)	.700 (.028)	.715 (.007)	.715 (.007)
Imp(LR)	.901 (.015)	.854 (.062)	.743 (.020)	.645 (.021)	.770 (.027)	.445 (.063)	.722 (.005)	.604 (.006)
Imp(LR) w/ disc.	**.909** (.014)	.871 (.054)	.734 (.021)	.642 (.024)	**.892** (.011)	.451 (.134)	**.769** (.003)	.690 (.013)

Table 74: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.2$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.849 (.030)	.849 (.030)	.568 (.052)	.568 (.052)	.572 (.091)	.572 (.091)	.619 (.036)	.619 (.036)
HCApp(LR) w/ disc.	.866 (.015)	.866 (.015)	.563 (.063)	.563 (.063)	.656 (.104)	.656 (.104)	.648 (.045)	.648 (.045)
IC-LR	.867 (.015)	.867 (.015)	.721 (.020)	.721 (.020)	.699 (.038)	.699 (.038)	.619 (.005)	.619 (.005)
IC-LR w/ disc.	.882 (.013)	**.882** (.013)	.723 (.019)	.723 (.019)	.860 (.008)	**.860** (.008)	.724 (.007)	**.724** (.007)
IC-LR w/ neg.	.878 (.015)	.878 (.015)	**.741** (.018)	**.741** (.018)	.708 (.036)	.708 (.036)	.705 (.003)	.705 (.003)
Imp(LR)	.878 (.014)	.821 (.061)	.729 (.025)	.635 (.025)	.772 (.017)	.448 (.056)	.710 (.003)	.602 (.005)
Imp(LR) w/ disc.	**.885** (.015)	.824 (.071)	.720 (.023)	.627 (.022)	**.883** (.012)	.519 (.093)	**.756** (.003)	.690 (.004)

Table 75: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.3$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCApp(LR)	.827 (.023)	.827 (.023)	.562 (.046)	.562 (.046)	.557 (.062)	.557 (.062)	.585 (.061)	.585 (.061)
HCApp(LR) w/ disc.	.849 (.018)	.849 (.018)	.555 (.051)	.555 (.051)	.654 (.076)	.654 (.076)	.621 (.037)	.621 (.037)
IC-LR	.842 (.018)	.842 (.018)	.691 (.021)	.691 (.021)	.705 (.027)	.705 (.027)	.613 (.004)	.613 (.004)
IC-LR w/ disc.	.863 (.016)	**.863** (.016)	.690 (.023)	.690 (.023)	.843 (.010)	**.843** (.010)	.711 (.005)	**.711** (.005)
IC-LR w/ neg.	.861 (.017)	.861 (.017)	**.712** (.021)	**.712** (.021)	.711 (.033)	.711 (.033)	.697 (.003)	.697 (.003)
Imp(LR)	.864 (.017)	.803 (.064)	.699 (.023)	.620 (.025)	.769 (.020)	.477 (.043)	.702 (.004)	.597 (.004)
Imp(LR) w/ disc.	**.870** (.019)	.792 (.087)	.688 (.021)	.613 (.020)	**.875** (.010)	.575 (.089)	**.749** (.005)	.684 (.006)

Table 76: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.4$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCAPP(LR)	.804 (.026)	.804 (.026)	.548 (.038)	.548 (.038)	.539 (.047)	.539 (.047)	.557 (.028)	.557 (.028)
HCAPP(LR) w/ disc.	.820 (.022)	.820 (.022)	.532 (.053)	.532 (.053)	.599 (.085)	.599 (.085)	.611 (.049)	.611 (.049)
IC-LR	.811 (.022)	.811 (.022)	.673 (.022)	.673 (.022)	.708 (.022)	.708 (.022)	.607 (.006)	.607 (.006)
IC-LR w/ disc.	.838 (.016)	.838 (.016)	.673 (.023)	.673 (.023)	.832 (.011)	**.832** (.011)	.699 (.004)	**.699** (.004)
IC-LR w/ neg.	.841 (.021)	**.841** (.021)	**.694** (.023)	**.694** (.023)	.716 (.024)	.716 (.024)	.691 (.004)	.691 (.004)
IMP(LR)	.843 (.015)	.796 (.041)	.689 (.024)	.609 (.026)	.759 (.018)	.511 (.055)	.695 (.004)	.598 (.005)
IMP(LR) w/ disc.	**.844** (.017)	.781 (.049)	.678 (.021)	.605 (.026)	**.854** (.010)	.629 (.063)	**.741** (.004)	.674 (.008)

Table 77: Our methods vs. the rest: mean classifier AUC for $\epsilon = 0.5$, unbalanced datasets, all features.

Classifier	Australia		Germany		Poland		Taiwan	
	Tru.	Str.	Tru.	Str.	Tru.	Str.	Tru.	Str.
HCAPP(LR)	.775 (.034)	.775 (.034)	.526 (.036)	.526 (.036)	.537 (.034)	.537 (.034)	.538 (.029)	.538 (.029)
HCAPP(LR) w/ disc.	.780 (.066)	.780 (.066)	.552 (.038)	.552 (.038)	.651 (.054)	.651 (.054)	.559 (.043)	.559 (.043)
IC-LR	.784 (.027)	.784 (.027)	.651 (.022)	.651 (.022)	.713 (.021)	.713 (.021)	.603 (.008)	.603 (.008)
IC-LR w/ disc.	.810 (.020)	.810 (.020)	.655 (.025)	.655 (.025)	.822 (.014)	**.822** (.014)	.682 (.003)	**.682** (.003)
IC-LR w/ neg.	**.818** (.026)	**.818** (.026)	**.673** (.019)	**.673** (.019)	.721 (.021)	.721 (.021)	.682 (.005)	**.682** (.005)
IMP(LR)	.816 (.015)	.748 (.055)	.667 (.027)	.596 (.024)	.763 (.016)	.521 (.043)	.687 (.004)	.593 (.005)
IMP(LR) w/ disc.	.813 (.018)	.713 (.077)	.658 (.029)	.590 (.024)	**.845** (.013)	.663 (.057)	**.732** (.004)	.660 (.003)