2011 11th IEEE International Conference on Data Mining

Class Imbalance, Redux

Byron C. Wallace* †, Kevin Small*, Carla E. Brodley† and Thomas A. Trikalinos *
*ICRHPS, Tufts Medical Center, Boston, MA
†Dept. of Computer Science, Tufts University, Medford, MA

***Abstract*—Class imbalance (i.e., scenarios in which classes are unequally represented in the training data) occurs in many real-world learning tasks. Yet despite its practical importance, there is no established theory of class imbalance, and existing methods for handling it are therefore not well motivated. In this work, we approach the problem of imbalance from a probabilistic perspective, and from this vantage identify dataset characteristics (such as dimensionality, sparsity, etc.) that exacerbate the problem. Motivated by this theory, we advocate the approach of bagging an ensemble of classifiers induced over balanced bootstrap training samples, arguing that this strategy will often succeed where others fail. Thus in addition to providing a theoretical understanding of class imbalance, corroborated by our experiments on both simulated and real datasets, we provide practical guidance for the data mining practitioner working with imbalanced data.**

***Keywords*-Classification, class imbalance**

I. Introduction and Motivation

In the context of classification, *class imbalance* refers to the scenario in which the number of instances from each class is (perhaps extremely) unequal. Imbalance is common in real-world learning tasks, for example: detecting oil spills [14], text classification, and many medical applications [5], to name a few. The problem of imbalance is exacerbated by the fact that in imbalanced scenarios, the minority class is typically of primary interest. That is, misclassification costs are typically asymmetric so as to emphasize correct classification of minority instances (e.g., cancer detection). Unfortunately, discriminative models induced over imbalanced datasets tend to fare poorly in terms of their predictive accuracy with respect to the minority class (i.e., such models generally suffer from low recall) [1].

Learning under imbalance is thus an important problem in data mining due to the prevalence of imbalance in real-world tasks and the relatively poor performance achieved by existing learning algorithms on such datasets. Indeed, the problem of inducing classifiers over imbalanced datasets with asymmetric costs was recently designated as one of '10 challenging problems in data mining research' [21]. The inherently pragmatic nature of the problem has motivated a significant amount of methodological research into learning under imbalance, of which there are at least three surveys [9], [8], [12]. Yet while many methods have been proposed to handle imbalance, there has been relatively little attempt to elucidate the underlying mechanisms that cause discriminative models to fail when faced with imbalanced datasets. Because the conditions that lead to poor classifier performance under imbalance are not well understood, it is not clear which (if any) of the myriad existing algorithms for mitigating the effects of imbalance ought to be employed for a given task. Consequently, when faced with imbalance, the data mining practitioner is left with little guidance regarding how to proceed.

More specifically, many techniques to address imbalance have been proposed in the literature, particularly for the binary classification case in which the aim is to induce a model to discriminate the minority from the majority class. These methods include re-sampling techniques such as: *undersampling*, i.e., training the model on an equal number of examples from both classes by discarding majority training instances; creating synthetic minorities (SMOTE) [4]; and algorithmic approaches that use cost-sensitive variants of algorithms (e.g., weighted SVM). All of these methods are heuristic and have been found to improve classifier performance under imbalance in some cases. Due to a dearth of theoretical understanding regarding the problem of imbalance, it is not clear when which method will be effective for a given task/dataset. In this article, we theoretically motivate and empirically justify the use of the simple undersampling strategy for imbalanced datasets under particular conditions (e.g., high-dimensionality). This work thus provides an explanation for the otherwise surprising observation that undersampling tends often to outperform what are ostensibly more advanced techniques (e.g., SMOTE) [11].

While effective, undersampling is problematic in that it's a high-variance strategy: classifiers induced over different bootstrap samples will tend to have significantly different predictive performances. To ameliorate this problem, one can use the *bagging* [2] variance-reduction ensemble method. Bagging reduces classifier variance by creating an ensemble of predictors over independently drawn bootstrap training samples. The strategy of bagging classifiers induced over balanced bootstrap training sets has been independently proposed several times (e.g,. [13], [19], [10], [17]), but *why* and *when* it should outperform other methods has been largely unexplored. In this work we provide such an explanation, and we conclude that *in almost all imbalanced scenarios, practitioners should bag classifiers induced over balanced bootstrap samples*. Specifically, we contend that while algorithmic approaches to handling class imbalance often improve performance, sampling approaches (such as

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.33

undersampling) will usually perform better. We show that cost-sensitive approaches that look to improve performance achieved under imbalance by, for example, modifying the relative costs of false negatives to false positives in an objective function, will generally be effective only when the training dataset is not separable.

The primary contributions of this work are as follows. We develop a probabilistic theory to quantify the effects of imbalance on the induction of empirical-loss minimizing models (e.g., SVMs). We show that under a few weak assumptions, such models will necessarily be biased toward the minority class, explaining the observed degradation in recall over test datasets. Furthermore, we decompose this bias into sub-components, some of which reflect properties of the training sample, and others that modify the empirical loss calculation. In light of this decomposition, we analyze several popular methods for handling imbalance, and discuss under what conditions one can expect them to work. We theoretically motivate, and experimentally demonstrate the efficacy of, the simple but robust strategy of bagging classifiers induced over balanced, bootstrap samples under various learning conditions. By providing a probabilistically motivated theory of imbalance, the implications of which are borne out both in our simulation and empirical experiments, we shed new light on a long-standing problem and provide disciplined guidance to practitioners facing imbalance.

II. A Theoretical Analysis of Imbalance

In supervised classification, we are given an observed training set $\mathcal{D}$ over which a predictive model c is to be induced. Typically, c is constructed so as to optimize a specified objective function (equivalently, minimize some loss function) over the points comprising $\mathcal{D}$. More precisely, let us assume that given $\mathcal{D}$, the aim is to induce a linear classifier that minimizes the empirical error over $\mathcal{D}$.

The primary assumption we will make in this work is that the observed positive and negative instances ($\mathcal{D}^+$ and $\mathcal{D}^-$) are drawn from two independent, latent distributions: P and G, respectively. Without loss of generality, we assume that positive instances constitute the minority class. Under this 'two-sample' assumption, it is readily apparent why a discriminative model induced over the joint observed distribution $\mathcal{D} = \mathcal{D}^+ \cup \mathcal{D}^-$ achieves poor recall: the positive distribution P is under-represented and hence poorly characterized, while we are likely to have encountered 'outlying' negative examples due the comparatively large number of observations drawn from G. We are therefore likely to induce a separator that is skewed toward the minority class (i.e., closer to the minority points than it should be), resulting in poor predictive performance over hold-out instances from this class.

This intuition is illustrated in Figure 1, a synthetic example in which the $\times$s represent the minority class and the ■s the majority: the corresponding latent Gaussians (P and G) from which these samples were drawn are also shown. The dotted line ($\hat{w}$) is the hypothesis induced over the training instances, while the solid line (w^*) depicts the optimal separator of the underlying distributions. In this case, the former is clearly skewed toward the minority class. We will now formalize the intuition captured in Figure 1 more precisely.

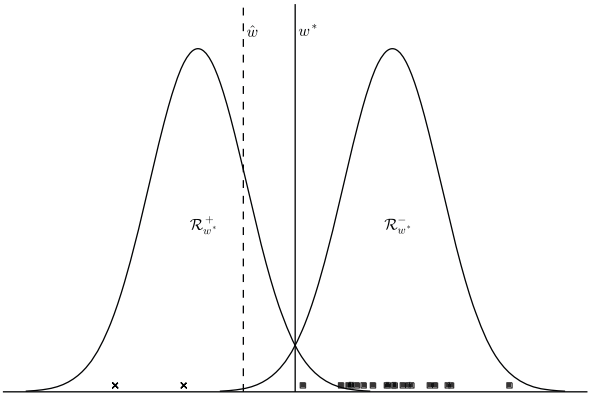


Figure 1: The bias of a linear separator induced over an imbalanced empirical sample in a one-dimensional example. Here the underlying distributions are shown, as well as a training sample comprising a few instances from the minority class (the $\times$s) and ten times as many from the majority class (the ■s). The solid line, w^*, is the optimal separator, w.r.t. the underlying distributions; the dotted line, $\hat{w}$, is the max-margin loss-minimizing separator induced over the empirical sample. Note that $\hat{w}$ is bias toward the minority class, w.r.t. w^*.

A. The Bias of Empirically Estimated Separators

Here we begin with the 'two-sample' scenario introduced above. We restrict ourselves to the task of inducing a loss-minimizing separating hyperplane w that splits the input (feature) space into two half-spaces, $\mathcal{R}^w_+$ and $\mathcal{R}^w_-$. Instances that fall in the $\mathcal{R}^w_+$ region (left side of Figure 1) are predicted to belong to the minority (positive) class, and those in $\mathcal{R}^w_-$ to the majority (negative). Aside from empirical loss minimization (over a training set $\mathcal{D}$), we make no further restrictions regarding the parameter estimation procedure. Indeed, there may be an infinite number of equivalent planes (i.e., loss-minimizing weight vectors) for a given training dataset; we assume only that the selected $\hat{w}$ is one of these. As a notational convenience, we superscript $\mathcal{R}$s with the planes that delineate them. We assume that the costs of false positives and false negatives are known, and denote these as $\mathcal{C}_{\text{fp}}$ and $\mathcal{C}_{\text{fn}}$, respectively.

Let us assume that the objective when training a classifier is to induce a separating plane w^* that minimizes loss with respect to the true, latent distributions P and G. Note that this assumption is consistent with the standard metrics for classifier evaluation under imbalance, which are typically averages of *rates*. The most widely used of these metrics is F-measure, a weighted harmonic mean of specificity and

recall.[1] Another popular metric for imbalanced datasets is the G-mean [14], which is the square root of the product of the accuracy achieved on the respective classes. By definition, these metrics are independent of the prevalence of the minority class, and thus one is tacitly ignoring the joint distribution observed in $\mathcal{D}$. Alternatively, this objective can be viewed as learning under the minimax assumption, in which case we attempt to minimize the maximum loss under an arbitrary covariate shift [15]. Stated precisely, we define the optimal plane as follows

$$w^* \leftarrow \operatorname*{argmin}_w \mathcal{L}^*(w) \tag{1}$$

where $\mathcal{L}^*(w)$ is the loss with respect to the true, latent distributions, i.e.:

$$\mathcal{L}^*(w) \leftarrow \mathcal{C}_{\text{fn}} \int_{\mathcal{R}_-^w} P(x)\mathrm{d}x + \mathcal{C}_{\text{fp}} \int_{\mathcal{R}_+^w} G(x)\mathrm{d}x \tag{2}$$

Thus w^* is the *ideal* separator w.r.t. the underlying distributions. Now consider the effect of imbalance. We denote the prevalence of the minority class by π (note that $\pi < .5$). Further, we will use $\mathcal{D}_\pi$ to denote the distribution over all datasets drawn from P and G with minority prevalence π. Then the expected empirical loss of an arbitrary w is

$$\mathbb{E}_{\mathcal{D}_\pi}[\mathcal{L}(w)] = \pi\mathcal{C}_{\text{fn}} \int_{\mathcal{R}_-^w} P(x)\mathrm{d}x + (1-\pi)\mathcal{C}_{\text{fp}} \int_{\mathcal{R}_+^w} G(x)\mathrm{d}x \tag{3}$$

We can also consider the empirical loss incurred over a particular dataset, $\mathcal{D}$:

$$\begin{aligned} \mathcal{L}_{\mathcal{D}}(w) &= \mathcal{C}_{\text{fn}}|\{x|x \in \mathcal{D}^+ \wedge x \in \mathcal{R}_-^w\}| + \\ &\quad \mathcal{C}_{\text{fp}}|\{x|x \in \mathcal{D}^- \wedge x \in \mathcal{R}_+^w\}| \end{aligned} \tag{4}$$

We denote by $\hat{w}$ a plane that minimizes empirical error over a particular draw from $\mathcal{D}_\pi$. We now argue that minimizing the empirical loss will (probabilistically) result in a plane that is skewed toward the minority class, with respect to w^*. In particular, we analyze the specific conditions under which this is the case, and show that the problem is exacerbated by imbalance. More specifically, we are interested in the conditions under which the induced region delineating the positive instance space $\mathcal{R}_+^{\hat{w}}$ is smaller than the corresponding region induced by w^* with respect to the loss over the underlying distributions, i.e.,

$$\mathcal{R}_+^{\hat{w}} < \mathcal{R}_+^{w^*} \tag{5}$$

Equation 5 formalizes our notion of bias. The question, then, is: when can we expect the induced separator to be biased? Proposition 1 states the necessary and sufficient condition for skew. Put plainly, this condition is intuitive. The induced separator will be biased if and only if any plane that designates a half-space for the minority class that is larger than the half-space delineated by w^* also incurs greater empirical loss on the training sample $\mathcal{D}$. More formally:

Proposition 1. *Fix a training dataset $\mathcal{D}$. Then, $\mathcal{R}_+^{\hat{w}} < \mathcal{R}_+^{w^*}$ iff $\exists w^\gamma$ s.t. $\forall w' \in \{w : \mathcal{R}_+^w \geq \mathcal{R}_+^{w^*}\}, \mathcal{L}_{\mathcal{D}}(w^\gamma) < \mathcal{L}_{\mathcal{D}}(w')$.*

Proof: Both directions of the implication are straightforward. Start with sufficiency. Assume that $\mathcal{R}_+^{\hat{w}} < \mathcal{R}_+^{w^*}$. For contradiction, further assume that $\neg\exists w^\gamma$ s.t. $\forall w' \in \{w : \mathcal{R}_+^w \geq \mathcal{R}_+^{w^*}\}, \mathcal{L}_{\mathcal{D}}(w^\gamma) < \mathcal{L}_{\mathcal{D}}(w')$. But $\hat{w}$ is just such a w^γ. It delineates a region $\mathcal{R}_+^{\hat{w}}$ that reduces the feature-space region circumscribing the minority class by some γ, by assumption. Further, by construction $\neg\exists w'$ $\mathcal{L}_{\mathcal{D}}(w^\gamma) < \mathcal{L}_{\mathcal{D}}(w')$. Thus we have our desired contradiction.

Now consider the other direction, i.e., necessity. Assume that $\neg\exists w^\gamma$ s.t. $\forall w' \in \{w : \mathcal{R}_+^w \geq \mathcal{R}_+^{w^*}\}, \mathcal{L}_{\mathcal{D}}(w^\gamma) < \mathcal{L}_{\mathcal{D}}(w')$. Then it must be the case that $\mathcal{R}_+^{\hat{w}} < \mathcal{R}_+^{w^*}$; were it not, there would $\exists\gamma$ s.t. $\mathcal{L}_{\mathcal{D}}(w^\gamma) < \mathcal{L}_{\mathcal{D}}(\hat{w})$, which is a contradiction because $\hat{w}$ minimizes empirical loss, by definition. ■

The question now becomes: when is such a w^γ likely to exist (equivalently, when can we expect the induced classifier to be biased)?. For latent distributions P and G, and training datasets with minority prevalence π, a w^γ will more likely exist than not when:

$$(1-\pi)\mathcal{C}_{\text{fp}} \int_{\mathcal{R}_{w^*}^+} G(x)\mathrm{d}x > \pi\mathcal{C}_{\text{fn}} \int_{\mathcal{R}_{w^*}^-} P(x)\mathrm{d}x \tag{6}$$

i.e., when w^* would incur a greater empirical cost than some alternative hypothesis w^γ because of the disproportionate contribution of false positives to this cost (i.e., the l.h.s. of Equation 6). In such cases, shifting w^* toward the minority class will reduce the empirical cost over $\mathcal{D}$, giving rise to a biased, empirical loss-minimizing hypothesis. In the following few sections, we will discuss methods for handling imbalance in light of this quantification of bias.

B. Why Weighted Empirical Cost Minimization is not Sufficient

Equation 6 decomposes the likelihood of inducing a biased separator into three sub-components: prevalence (π), costs quantifying mistakes made on instances belonging to the respective classes, and (latent) distributional characteristics. It would seem that the straight-forward strategy to handling imbalance, then, would be to fiddle with the $\mathcal{C}_{\text{fp}}$ and $\mathcal{C}_{\text{fn}}$ variables – in particular to penalize false negatives more heavily than false positives, or otherwise modifying the objective function to achieve this implicitly. We will refer to the family of methods that attempt to mitigate the effects of imbalance by assigning different costs to false

[1] When precision is used instead of specificity, F_2 is not independent of prevalence.

positives/negatives during induction as *weighted empirical cost minimizing learners*. Many methods of this type have been proposed in the literature, e.g., [17], [20].

However, modifying the empirical cost structure will often have no effect at all. In particular, *if the instances comprising the classes in the training dataset are separable, modifying the cost of false negatives relative to that of false positives in the objective function will not reduce bias*. This is trivially true; increasing the cost of false negatives will not budge the induced $\hat{w}$ if there are none in the first place.

Furthermore, one can quantify the conditions under which modifying the empirical cost of false negatives/positives will be effective. Consider that this can reduce bias (Equation 5) if and only if it affects the empirical lost incurred over $\mathcal{D}$ (Equation 4). For the moment, let $\mathcal{C}_{\text{fp}} = \mathcal{C}_{\text{fp}} = 1$. Denote the empirical loss-minimizing plane induced in this case by $\hat{w}_1$. Increasing the cost of a false negative to β times that of a false positive will produce a different plane only if there exists a point closer to the majority half-space than $\hat{w}_1$, i.e., if $\hat{w}_1$ results in at least one false negative. If no such point exists, $\hat{w}_1$ will already be loss-minimizing, regardless of β.

The probability that such a point will have been observed in $\mathcal{D}$ is

$$\pi|\mathcal{D}| \int_{\mathcal{R}^-_{\hat{w}_1}} P(x)\mathrm{d}x \tag{7}$$

As the degree of imbalance increases (i.e., π decreases), the probability that using weighted empirical cost minimization to counter imbalance will be effective in reducing bias decreases. Equation 7 also suggests that as the size of the training set increases, such strategies will become more effective, in general. Both of these observations are borne out in our simulation experiments (Section IV). The characteristics of P will also contribute to the (in)effectiveness of cost-sensitive induction procedures, e.g., if P happens to be dense around the space of w^*, then weighting will be efficacious.

C. Remarks on SMOTE

One of the most popular strategies for countering imbalance is the Synthetic Minority Oversampling TEchnique (SMOTE) [4]. SMOTE is ostensibly a sampling strategy, insofar as it ultimately produces a balanced dataset on which to induce a model, but we argue that with respect to imbalance, SMOTE behaves similarly to the weighted empirical cost minimizing learners discussed above. SMOTE works by interpolating the observed minority instances with one another to create 'new', synthetic minority instances. In particular, this is done as follows. For each minority instance x^i, find the k minority points in $\mathcal{D}$ to which it is nearest. Now create synthetic minority points from x^i by selecting one of these neighbors x^n at random and creating a value for each feature j that falls on a random point along the line connecting x^i_j and x^n_j.

Due to the interpolation mechanism for creating synthetic instances, no pseudo-minority point produced via SMOTE will ever be located outside of the convex hull enclosing the observed minority instances. This observation implies that the probability that SMOTE will reduce bias during induction over an imbalanced dataset is exactly the same as for the weighted empirical loss minimizing techniques (Equation 7), i.e., SMOTE should work in cases that weighted empirical loss minimizing methods work.

III. The Case for Undersampling and Bagging

We'll now present arguments in favor of the undersampling plus bagging strategy for mitigating imbalance in light of the preceding discussion.

A. Why Does Undersampling Work?

The idea of throwing away most of one's data in order to induce a model seems anathema to statistical inference, as generally the best strategy is to exploit all available information. In spite of this, undersampling has proven effective in the case of imbalance, more often than not outperforming more advanced methods [11], [14], [6], [17]. The notion of 'outperforming', of course, pre-supposes a metric of interest. Most of the empirical work in the imbalanced literature uses a weighted harmonic mean of recall and specificity (or recall and precision), and we will follow this convention here. Generally it assumed that recall is more important than accuracy on the majority class; how much so will depend on the task at hand.

Undersampling is effective despite its simplicity because it reduces the probability that the induced separator will be biased. More specifically, consider the inequality expressed in Equation 6, which quantifies the condition under which we are likely to induce a biased $\hat{w}$. Removing majority instances from $\mathcal{D}$ until $|\mathcal{D}^+| = |\mathcal{D}^-|$ effectively removes π from this equation. Thus, for a separator induced over an undersampled dataset, the condition under which we expect a biased plane becomes:

$$\mathcal{C}_{\text{fp}} \int_{\mathcal{R}^+_{w^*}} G(x)\mathrm{d}x > \mathcal{C}_{\text{fn}} \int_{\mathcal{R}^-_{w^*}} P(x)\mathrm{d}x \tag{8}$$

Crucially, this removes the imbalance component from the inequality (it becomes π on both sides). To illustrate the effects of this, recall the toy example depicted in Figure 1, in which training instances are drawn disproportionately from two latent one-dimensional Gaussians. In Figure 2, we draw 10 planes induced over balanced samples taken from the training set. All of these are less biased (closer to w^*) than the separator induced over the entire training dataset ($\hat{w}$). However, one can see that this is also a high-variance procedure – different re-samplings beget very different planes. We will now discuss how to mitigate this property via bagging [2].

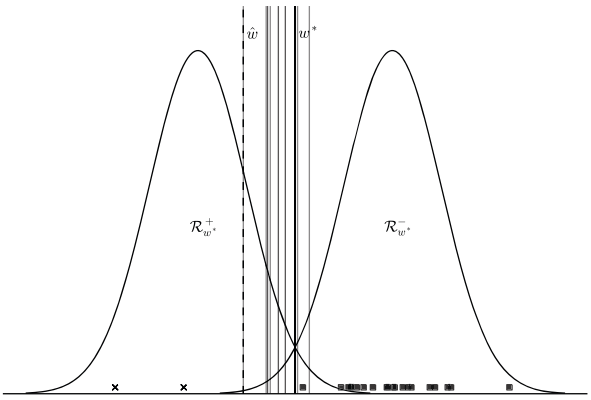


Figure 2: The effect of undersampling on separator induction. $\hat{w}$ is the (biased) plane induced over the entire dataset, w^* the optimal plane (w.r.t. the underlying distributions) and the light grey lines depict the lines induced over independently drawn balanced bootstrap samples of the training data. Note that all of these are less biased (nearer w^*) than $\hat{w}$.

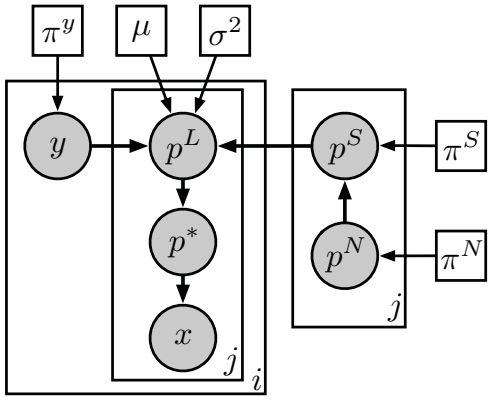


Figure 3: The plate diagram corresponding to our simulation scenario. See text for details.

B. Bagging for Imbalance

Bagging is a method of aggregating classifiers induced over independently drawn *boostrap* samples [2]. *Bootstrapping* is a sampling mechanism that has traditionally been used to estimate the (true) standard error of a summary statistic calculated over an empirical sample $\mathcal{D}$ by calculating this statistic over n independently drawn 'bootstrap' samples taken from $\mathcal{D}$.

Bagging is a natural extension of the bootstrapping technique for predictive models that works as follows. We build an ensemble comprising B models, each induced over a bootstrapped sample of the training data. When a new instance is to be classified, each model makes a prediction, and the final, aggregate prediction is taken as the majority vote. Typically, bootstrap samples are drawn at random and i.i.d. from the original sample [7], and thus reflect the distributional characteristics of the original dataset. In our case, this would mean each sample would be imbalanced. This is undesirable because it would create bootstraps equally likely to beget biased classifiers. Indeed, the bagging methods proposed for imbalance advocate taking balanced samples, with the exception of Hido and Kashima [10], who propose 'roughly balanced' samples as a 'better motivated' (statistically) approach. However, balanced sampling *is* stastically appropriate, given our aim.

In particular, constructing balanced bootstrap samples is statistically appropriate for the case of classifier induction; specifically it is an instance of the *two-sample* case, described by Efron [7]. We observe a joint distribution $\mathcal{D}$ drawn from P and G, disproportionately. We are interested in estimating a separator w.r.t. these distributions, independent of the imbalance in the observed sample. In particular, during the induction of a discriminative model, we are implicitly estimating properties of P and G. In the case of empirical error minimizing linear separators, we are effectively estimating the density of points around the intersecting tails of the distributions. Two-sample bootstrapping provides a direct mechanism for estimating this. In general, bagging will improve classifier performance when the individual members comprising the ensemble are high-variance – this is exactly the case with classifiers induced from different undersampled training datasets.

IV. Simulations

A. Simulation Framework

We performed simulation experiments to systematically explore the empirical implications of the preceding sections. We constructed a simple generative model for creating instances that allowed us to experiment with various factors that, given our theoretical assumptions and above derivations, ought to influence the relative performance of various techniques for handling imbalance. The objective here is to use this simple model to elucidate the conditions under which different undersampling techniques might be effective.

We are more specifically interested in those cases in which undersampling, and/or bagging classifiers induced over undersampled datasets (hereupon referred to simply as bagging) outperforms other strategies for learning under imbalance. In particular, we consider SMOTE [4], and weighted-SVM. Obviously, there are many other existing techniques for handling imbalance with which we could have experimented, but the selected approaches are: 1) commonly used and 2) prototypical, in the sense that other techniques tend to be special cases or hybrids of these.

The generative model we used in our simulations is described by the plate diagram in Figure 3 (note that all variables denoted by π parameterize Bernoulli distributions). This is essentially Naïve Bayes de-constructed; when generating an instance x, the probability of observing a given feature is conditioned only on the label assigned to x, independent of the other attributes comprising x. In this

simulation, we consider only binary features, i.e., each feature is either '1' or '0'. We will denote feature j of instance i by x_{ij}. The label, y, is '1' (i.e., a minority instance) with probability π^y. We associate with each (non-noisy) feature a *polarity*, which is drawn from a normal distribution. The mean, μ, of this distribution determines how strongly features correlate with their labels – thus a larger μ here implies an 'easier' task.

Aside from the prevalence π^y, there are two parameters of particular interest: π^N, which dictates the amount of feature-noise, and π^S, which we call 'sparsity'. The former encodes the expected proportion of features that will contain no information regarding the label of the instances in which they are observed, i.e., features for which it is the case that $p(x_{ij} = 1|y_i) = p(x_{ij})$. The latter (sparsity) encodes how sparse the generated instances will be; all other details aside, any given feature will be observed in an arbitrary instance (independent of its label) with probability $\leq 1 - \pi^S$.

To summarize, we use a simple multinomial model to generate data. This generative model allows us to explore the effects of various parameters of input, including: dimensionality (d, which is external to the plate diagram, as it is set prior to generation); the degree of imbalance in the dataset (π^y); the proportion of uninformative features in the data (π^N); and the sparsity of the data π^S.

B. Results From Simulation Experiments

We now present a series of experiments that explore the effects of altering the parameters outlined above to support the arguments presented in earlier sections. In all experiments shown, we fixed μ at .6, with a relatively tight σ^2 of .02 – i.e., the results shown are for datasets in which non-noisy features are relatively strongly correlated with classes, though not overwhelmingly so. In all experiments, we generated both a training and a test set with the same parameters.

SMOTE requires specifying the percentage of synthetic minorities to be added to the dataset; for example, setting this to 100% will effectively double the minority class size by adding synthetic instances. For our purposes, we ran experiments with this parameter set across a few orders of magnitude (100% and 1000%) and display results for the best performing parameter.[2] For **weighted**-SVM, we used LibSVM's [3] implementation, and similarly experimented with a few orders of magnitude (100, 1000) for the parameter expressing the cost of false negatives relative to false positives, again showing the best result achieved for each experiment.

For **undersampling**, we threw majority instances away at random until the training set was balanced. Finally, for **bagging**, we built an ensemble of 11 classifiers induced over independently constructed undersampled datasets,[3] and predictions were taken as a majority vote over these. Both undersampling and bagging therefore include a stochastic element. We thus performed 10 independent iterations of each experiment with these methods to assess variability (error bars in the plots show best and worst performance over these runs).

Classifier evaluation over imbalanced datasets is inherently tricky. In practice, the relative costs of false positives (negatives) would have to be somehow elicited from the domain expert, and a weighted metric reflecting these costs could then be used to assess performance. For this work, we don't have these costs explicitly, and thus we take the standard approach of using a weighted harmonic mean of recall and specificity. Specifically, we use F_2, in which recall is considered twice as important as specificity.[4]

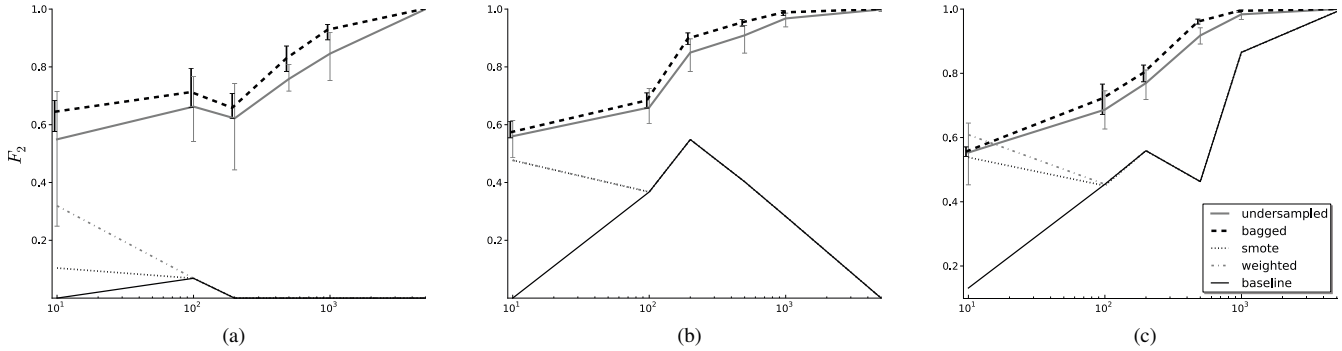
The first experiment we conducted considered the effect of increasing dimensionality on the induced classifiers' performances. According to Section II, increased dimensionality should lead to decreased utility of the empirical-cost adjustment strategies (SMOTE and weighted-SVM), because in general as dimensionality increases, so too will the likelihood of the training data being separable, modulo the prevalence π^y and training sample size $|\mathcal{D}|$. For these experiments, we set the sparsity parameter π^S to .5, and we did not include any noisy features, i.e., all features were informative. The results are shown in Figure 4. In all cases, SMOTE and weighted-SVM both improve performance relative to baseline SVM at lower dimensionalities, but their relative performance regresses to the baseline as the dimensionality increases, as we predicted. The 'hump' seen in the first two cases appears because up to a certain dimensionality, the additional informative features increase model recall. However because of the low prevalence, the classifier is unable to learn which are the features associated with the minority class in higher dimensions. In (c), the prevalence appears to be sufficiently high to ameliorate this issue. Note that bagging not only reduces variance with respect to only undersampling (as can be seen in the corresponding error bars), but also performs better, on average.

The second experiment shown examines the relationship between the training set size ($|\mathcal{D}|$) and classifier performance. Figure 5 plots this against performance, again for three prevalences (.05, .1 and .2). In the first case, when the prevalence is low, the undersampled and bagged approaches consistently dominate, until the training set size reaches 3000, at which point weighted-SVM manages to catch up. When the degree of imbalance is less extreme, e.g., .1 and .2, the empirical weighted cost methods more quickly achieve performance comparable to the sampling strategies. This is precisely what we would expect in light of Equation 7. Note that undersampling and bagging again dominate, and

[2]These almost universally performed the same.

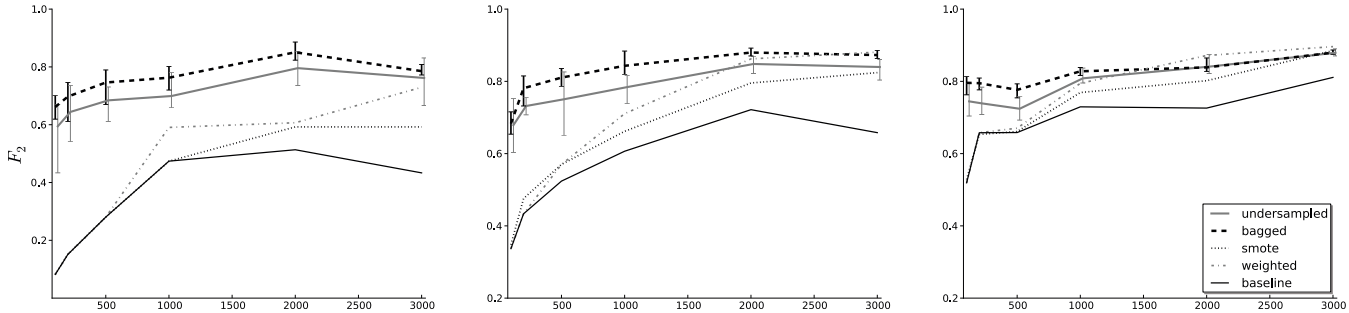
[3]The committee size of 11 was arbitrarily selected.

[4]We use specificity rather than the more popular precision when calculating F_2 because this version of the metric is independent of prevalence – note that this is similar to the 'G-mean' [14].



(a) (b) (c)

Figure 4: Simulation experiments investigating the relationship between dimensionality and F_2. Dimensionality runs across the x-axis (log-scale) from 10 to 1000 dimensions. The plots show results for experiments with varying levels of minority prevalence π^y. In particular, from left to right: $\pi^y = .05$, $\pi^y = .1$, $\pi^y = .2$. For all experiments, the training and test set comprise 100 and 1000 examples, respectively. See text for further details.



(a) (b) (c)

Figure 5: Simulation experiments investigating the relationship between training set size and F_2. In all experiments, the dimensionality of the feature space is fixed at 100. The minority prevalences running from left to right are $\pi^y = .05$, $\pi^y = .1$, $\pi^y = .2$.

again the latter both performs better and reduces variance, compared to undersampling alone.

We also considered the relationship between empirical error on the training set and the performance of the induced classifier. Our hypothesis was that empirical weighted cost strategies (e.g., weighted-SVM and SMOTE) would be effective in countering imbalance only when the baseline SVM incurred empirical error on the training set. This hypothesis follows directly from the discussion in Section II-B, and the intuition is clear. Strategies that upweight the cost of, e.g., false negatives w.r.t. false positives will work only insofar as they may push the separating plane demarcating the minority space until it encompasses the minority instance in the training set nearest the majority class. Once this outlying minority instance is correctly classified, modifying empirical costs will have no effect. Nor will SMOTEing work in this case, because due to the interpolative method of point generation, any synthetic point will necessarily fall inside of this outlying minority instance.

To explore this conjecture empirically, we ran experiments over 50 synthetic datasets generated at random. We drew the parameters dictating various properties of the dataset at random from sets of values we thought reasonable. In particular: for dimensionality, we drew uniformly from $\{10,$

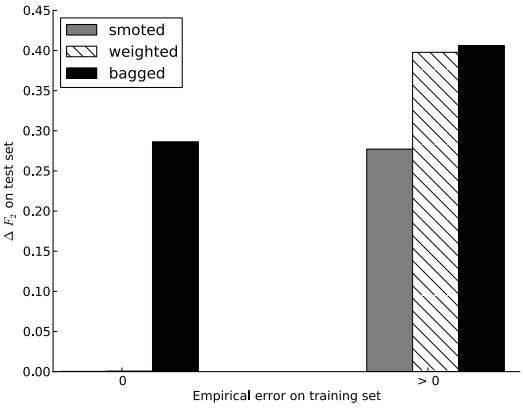


Figure 6: The y-axis is the average difference (improvement) in F_2 between the corresponding method and baseline SVM over hold-out test sets (ΔF_2 – when this difference is large, the corresponding method for handling imbalance was effective in that it improved performance). Three methods are shown: SMOTEd, weighted-SVM and bagged. (Results for undersampled were similar to bagged). On the left-hand side of the plot, the average ΔF_2 is shown for the methods in datasets for which there was 0 empirical error (i.e., separable datasets). Note that the empirical weighted cost strategies provide no benefit over baseline, but bagging is efficacious. Results for cases where there was empirical error on the training set are shown on the right-hand side. In these cases, SMOTE and weighted-SVM are competetive with bagging.

100, 500, 1000, 2000}, and for training set size from {100, 200, 300, 500, 1000}.[5] We drew π^N (noise) uniformly from {0, .2, .3, .4, .5, .6, .7, .8}, π^S (sparsity) uniformly from {0, .2, .5, .6, .7, .8, .95. 99}, μ (polarity) from {.55, .6, .65} and π^y (prevalence) from {.05, .1, .15, .2}.

Figure 6 displays summary results from these 50 datasets. In particular, we show the average improvement, in terms of F_2, achieved by each of the strategies with respect to baseline SVM. The left-hand side corresponds to datasets over which baseline SVM achieved perfect accuracy, while the right-hand side plots results for datasets on which the baseline SVM incurred empirical error. In the former case, neither SMOTEing nor weighting has any effect on classifier performance, but bagging does; when there is empirical error, however, both weighted SVM and SMOTEing are effective, thus supporting our conjecture. The message here is that bagging is effective even when a standard SVM can perfectly separate the training dataset, whereas empirical weighted cost strategies are not.

To summarize our results over synthetic datasets, we have shown that: 1) bagging/undersampling consistently outperformed other strategies in terms of predictive performance on synthetically generated imbalanced data (bagging doing so with lower variance than undersamping alone), and 2) as expected per our discussion in Section II, undersampling/bagging was particularly effective, relative to SMOTE and weighted-SVM, in cases when the training dataset was not separable. Moreover, as is implied by Equation 7, this relative performance was observed to correlate with the prevalence (π) and the training set size ($\mathcal{D}$).

V. Empirical Results on Benchmark Datasets

We now experiment with 'real' datasets to see if the pattern observed in the synthetic case (above) holds. In particular, we used 16 datasets with varying degrees of imbalance; 13 of these were taken from the UCI dataset repostiory, the other 3 from real-world biomedical text classification tasks. The datasets are summarized in Table I. Here we define the 'sparsity' of a dataset as 1 minus (the expected proportion of features observed in a given instance drawn from that dataset). Sparsity is particularly relevant to textual data, wherein every word is relatively rare, and the vectors representing documents tend therefore to be sparse.

We first randomly split all of the datasets shown in Table I into train and test sets, comprising 10% and 90% of the corresponding datasets, respectively. We then conducted the same experimental analysis as was described for the simulated data case in Section IV.

Figure 7 plots F_2 against dimensionality for all of the learners across all datasets. The most striking feature of this plot is the departure of bagging/undersampling at extreme dimensionalities: the difference in F_2 becomes substantial at dimensionalities of 10^4. At this point, as in our simulations, both SMOTE and weighted-SVM regress to baseline SVM. Another consistent pattern that emerges is that bagging again performs comparably (and often better) than undersampling alone and has a lower variance.

[5]In all cases we tested over a few thousand generated instances.

name	N	d	π	sparsity
car	1728	21	.040	.714
cmc	1473	24	.226	.625
ecoli	336	9	.104	.222
german	1000	61	.300	.721
glass	214	9	.079	0
haberman	306	3	.265	0
letter-a	20000	16	.039	0
letter-vowel	20000	16	.194	0
nursery	12960	27	.025	.704
pima	768	8	.349	0
splice	3190	287	.241	.790
vehicle	846	18	.251	0
yeast	1484	9	.289	.111
proton beam	4751	1025	.051	.993
copd	1600	6526	.122	.989
micro-nutrients	4010	11524	.064	.992

Table I: Characteristics of the datasets we used in our experiments. The top 13 are taken from the UCI dataset repository; the bottom 3 are real-world datasets from a biomedical text classification task.

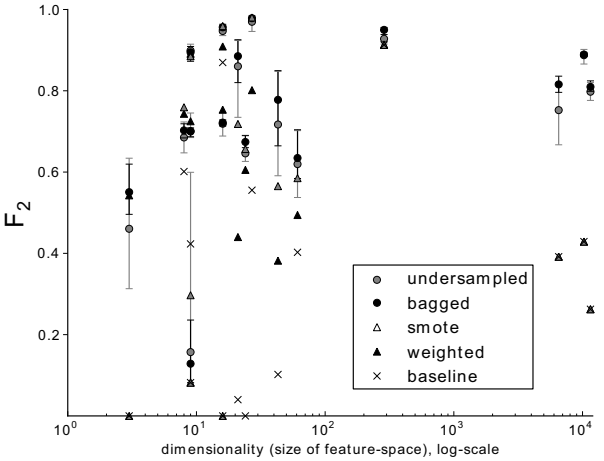


Figure 7: F_2 over test-sets for the datasets summarized in Table I. Note that for the very high dimensional datasets, undersampling and bagging dominate (the latter again having lower variance).

Figure 7 is somewhat difficult to parse, and, further, it is restricted to one dimension (dimensionality), despite the fact that the other characteristics of interest (e.g., π) are not fixed, as they were in the simulations by construction. For interpretative purposes, we therefore performed an analysis on these empirical results to explore the effect of the different dataset characteristics on the respective methods for handling imbalance. More specifically, we evaluated the association between the recall of the five techniques and four characteristics of the datasets (prevalence, log-transformed training set size, log-transformed number of dimensions, and sparsity). Briefly, we used a two-level generalized linear mixed-effects regression that allows for between-classifier

correlations within each dataset, and for common effects of the characteristics of interest across datasets. We modeled the dependency of classifier performance (recall) on each characteristic with interaction terms. Such hierarchical regressions are often used to explore which factors affect the relative performance of diagnostic tests [18].

Figure 8 displays the results from this analysis. Figure 8(a) shows how the predicted mean recall of SMOTE and bagging (i.e., predicted by the model induced over the empirical results) change for each classifier induced as a function of the dataset characteristics of interest. For each characteristic, we hold the values for the others constant; the red lines demarcate this fixed spot for each characteristic. The trends are as we expect: SMOTE works well when prevalence and training set size are large, but poorly when they are small, as is predicted by Equation 7. Similarly, SMOTE works comparatively well in lower dimensionality and sparsity (these can be seen as properties of the underlying distribution, P). In Figure 8(b), we show the estimated coefficient of each of the aforementioned dataset characteristics in terms of their effect on the difference between the performance of bagged and that of the empirical weighted cost methods (SMOTE and weighted-SVM). The circles and squares correspond to these point estimates for SMOTE versus bagged and weighted versus bagged, respectively, and the horizontal bars depict the 95% confidence interval. The directions of these coefficients are as expected, given our theoretical exposition and our simulation experiments; both SMOTE and weighted-SVM perform better (worse), w.r.t. the undersampled bagging approach, as the prevalence (π) and the training set size ($\mathcal{D}$) increase (decrease). The reverse holds for dimensionality and sparsity: as these decrease, the effectiveness of the empirical weighted cost methods decreases, too (relative to bagging).

In the low-dimensional case, the empirical cost weighting strategies (SMOTE, weighted) are competetive with undersampling and bagging. In higher-dimensions, however, these strategies regress to the baseline.

VI. Related Work

Broadly, techniques for mitigating the effects of imbalance fall into two categories: re-sampling methods (e.g., [19], [14]) and methods that alter the empirical error function being optimized over the training set to emphasize recall (e.g., [16], [20]). The former class of methods have often been observed to out-perform more sophisticated approaches to handling imbalance (e.g., SMOTE). However, despite the method of subsequently aggregating an ensemble of these classifiers being (in our view) a natural next step, this bagging approach is often overlooked by researchers. Indeed, the most comprehensive empirical comparison of strategies to mitigate the effects of imbalance [11] did not include bagged classifiers induced on bootstrap (undersampled) training sets, despite undersampling performing the best of all methods, overall. However, while it has not gained widespread adoption by practitioners, bagging for imbalance has been previously proposed [13], [19], [10], [17]. Our main objective in this work, aside from attaining a better formal understanding of the imbalanced problem in general, was to elucidate the conditions in which this strategy is effective.

In addition to the utility of bagging for handling imbalance, the results presented in this work corroborate, and provide explanation for, previously reported observations. For example, Japkowicz [12] observed that as sample size increases, imbalance becomes less of a problem, in general. One can see why this is the case under our model: eventually a sufficient number of draws are made from the minority class, and it can thus be adequately characterized. (This is supported by Figure 5). Those interested in a more detailed summary of existing methods for handling imbalance should consult one of the existing surveys (e.g., [9]).

VII. Conclusions

We have provided a probabilistic interpretation of the effects class imbalance has on discriminative models. We ran simulation experiments to corroborate this theory. On this interpretation, we demonstrated the scenarios in which empirical error minimizing (linear) classifiers induced over imbalanced datasets will likely induce a biased separator. Furthermore, we quantified the conditions when weighted empirical cost methods for mitigating the effects of imbalance, such as weighted-SVM and SMOTE,[6] will likely fail to improve performance.

It follows from the probabilistic interpretation of class imbalance developed in this paper that re-sampling methods, specifically undersampling, ought to be used to handle imbalance in most scenarios (as opposed to strategies that modify the objective function maximized during classifier induction to penalize false negatives more than false positives). Further, bagging should be used to reduce the variance of this approach. We motivated this advice theoretically and experimentally, and highlighted that this is in agreement with much of the prior experimental work investigating methods for handling imbalance.

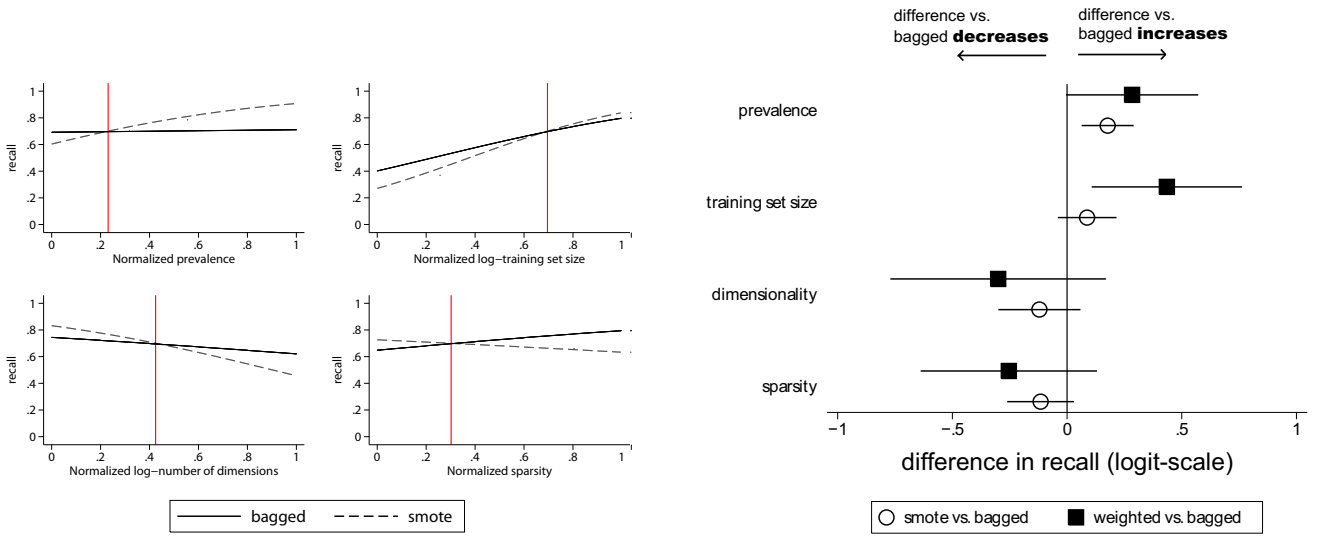
References

[1] R. Akbani, S. Kwek, and N. Japkowicz. Applying support vector machines to imbalanced datasets. In *ECML*, pages 39–50, 2004.

[2] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] C.-C. Chang and C. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1–27:27, 2011.

[6] We re-iterate that while SMOTE technically affects the training class distributions, it effectively behaves like a empirical cost weighted technique, due to its method for generating synthetic minority instances: see Section II-C.



(a) (b)

Figure 8: Results from a regression analysis of our empirical results. The direction of the effects of the considered dataset characteristics agree with our theoretical assessment and our simulation experiments. See text for discussion.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[5] G. Cohen, M. Hilario, H. Sax, S. Hugonnet, and A. Geissbuhler. Learning from imbalanced data in surveillance of nosocomial infection. *Artificial Intelligence in Medicine*, 37(1):7–18, 2006.

[6] C. Drummond and R. C. Holte. C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling. In *Proceedings of the ICML Workshop on Learning from Imbalanced Datasets II*, 2003.

[7] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. CRC Press, 1993.

[8] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou. On the class imbalance problem. In *ICNC*, pages 192–201, 2008.

[9] H. He. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[10] S. Hido and H. Kashima. Roughly balanced bagging for imbalanced data. In *SDM*, pages 143–152, 2008.

[11] J. V. Hulse, T. M. Khoshgoftaar, and A. Napolitano. Experimental perspectives on learning from imbalanced data. In *ICML*, pages 935–942, 2007.

[12] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449, 2002.

[13] P. Kang and S. Cho. EUS SVMs: Ensemble of under-sampled svms for data imbalance problems. In *ICONIP*, pages 837–846, 2006.

[14] M. Kubat and S. Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *ICML*, pages 179–186, 1997.

[15] G. R. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan. A robust minimax approach to classification. *Journal of Machine Learning Research*, 3:555–582, 2002.

[16] W. Liu and S. Chawla. A quadratic mean based supervised learning model for managing data skewness. In *SDM*, 2011.

[17] X.-Y. Liu, J. Wu, and Z.-H. Zhou. Exploratory under-sampling for class-imbalance learning. In *ICDM*, pages 965–969, 2006.

[18] J. Reitsma, A. Glas, A. Rutjes, R. Scholten, P. Bossuyt, and A. Zwinderman. Bivariate analysis of sensitivity and specificity produces informative summary measures in diagnostic reviews. *Journal of Clinical Epidemiology*, 58(10):982–990, 2005.

[19] D. Tao, X. Tang, X. Li, and X. Wu. Asymmetric bassing and random subspace for support vector machines-based relevance feedback in information retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1088–1099, 2006.

[20] S. Wu, K. Lin, C. Chen, and M. Chen. Asymmetric support vector machines: Low false-positive learning under the user tolerance. In *KDD*, pages 749–757. ACM, 2008.

[21] W. Yang and X. Wu. 10 challenging problems in data mining research. *International Journal of Information Technology Decision Making*, 5(4):597–604, 2006.