An Improved Categorization of Classifier's Sensitivity on Sample Selection Bias

Wei Fan[1] Ian Davidson[2] Bianca Zadrozny[1] Philip S. Yu[1]

[1] IBM T.J.Watson Research, Hawthorne, NY 10532
{weifan,psyu,zadrozny}@us.ibm.com

[2] Department of Computer Science, State University of New York, Albany, NY 12222
davidson@cs.albany.edu

Abstract

A recent paper categorizes classifier learning algorithms according to their sensitivity to a common type of sample selection bias where the chance of an example being selected into the training sample depends on its feature vector $\mathbf{x}$ *but not (directly) on its class label* y*. A classifier learner is categorized as "local" if it is insensitive to this type of sample selection bias, otherwise, it is considered "global". In that paper, the true model is not clearly distinguished from the model that the algorithm outputs. In their discussion of Bayesian classifiers, logistic regression and hard-margin support vector machines, the true model (or the model that generates the true class label for every example) is implicitly assumed to be contained in the model space of the learner, and the true class probabilities and model estimated class probabilities are assumed to asymptotically converge as the training data set size increases. However, in the discussion of naive Bayes, decision trees and soft-margin support vector machines, the model space is assumed not to contain the true model, and these three classification algorithms are instead argued to be "global learners". Here we argue that most classifier learning algorithms including those just discussed may or may not be affected by sample selection bias; this will depend on the dataset as well as the heuristics or inductive bias implied by the classifier learning algorithm and their appropriateness to the particular dataset. We make use of our earlier experimental results and produce additional results to illustrate our claims.*

1 Introduction

A common assumption made in data mining is that the training and test sets are drawn from the same stationary distribution. However, in practice this rarely happens. Recent work published in the data mining community has addressed situations where the learned concept (i.e. the relationship between the class label and the example description) drifts. In this paper, we look at the related and prevalent problem that most data mining practitioners face: sample selection bias. Sample selection bias occurs when the concept remains static, but the training set is biased in that the chance of encountering an example is not the same in the training and test sets. For example, in the application of data mining to direct marketing, it is common practice to build models of the response of customers to a particular offer using only the customers that have received the offer in the past as the training set, and then to apply the model to the entire customer database. Because these offers are usually not given at random, the training set is not drawn from the same population as the test set. Therefore, a model learned using this training set may not perform well for the entire population of customers.

Sample selection bias of all types has been an active research topic. A large body of related work will be reviewed in Section 5. In this paper, we concentrate on recent work [Zadrozny, 2004] that focuses on a common type of bias (defined in the next paragraph) and categorizes inductive learners into two types, "local" and "global", based on their sensitivity to this type of sample selection bias. We clarify some of the important notations and concepts implied in [Zadrozny, 2004] and make improvements on the categorization of classifiers. We believe these improvements establish a general relationship between sample selection bias and the accuracy of inductive learners, and create a foundation for future work on general methods to detect the sensitivity of an algorithm to sample selection bias in real-world applications.

1.1 Sample Selection Bias

Assume that the event $s = 1$ denotes that a labeled example $(\mathbf{x}, y)$ is selected from the domain $\mathcal{D}$ of examples into the training set D, and that $s = 0$ denotes that $(\mathbf{x}, y)$ is not chosen. When constructing a classification model, we only have access to examples where $s = 1$. In [Zadrozny, 2004], four different types of sample selection bias are clearly discussed according to the dependency of s on $\mathbf{x}$ and y.

The **first case** is that s is independent from both $\mathbf{x}$ and y, i.e., $P(s|\mathbf{x}, y) = P(s)$. In other words, the selection variable s is a random variable completely independent from both the feature vector $\mathbf{x}$ and the true class label y. In the **second case**, the selection bias s is dependent on the feature vector $\mathbf{x}$ and it is conditionally independent of the true class label y given $\mathbf{x}$, i.e., $P(s|\mathbf{x}, y) = P(s|\mathbf{x})$. This type of sample selection naturally exists. For example, in the direct marketing case mentioned earlier, the customers are selected into the sample based on whether or not they have received the offer in the past. Because the decision to send an offer is based on the known characteristics of the customers (that is, $\mathbf{x}$) before seeing the response (that is, y) then the bias will be of this type. This type of bias also occurs in medical data where a treatment is not given at random, but the patients receive the treatment according to their symptoms which are contained in the example description (i.e. the $\mathbf{x}$ values). Therefore, the population that received the treatment in the past is usually not a random sample of the population that could potentially receive the treatment in the future. In the **third case**, the selection bias is dependent only on the true class label y, and is independent from the feature vector $\mathbf{x}$, i.e., $P(s|\mathbf{x}, y) = P(s|y)$. This occurs when there is a correlation between the label value and the likelihood of appearance in the database. For example, people with higher income may be less inclined to answer a survey about income. Thus, if we are trying to learn to predict an individual's income category using survey data, this type of bias is likely to occur. In the **last case**, there is no assumption about any independence of s given $\mathbf{x}$ and y, and both the example description and its label influence whether the example will be chosen into the training s et. Note that in all cases the test set examples are assumed to be unbiased, since the model will be used on the entire population. For the remainder of this paper we shall focus on the second type of sample bias as it is believed to be a prevalent problem [Zadrozny, 2004].

1.2 Existing Classifier Categorization and its Limitations

In [Zadrozny, 2004], inductive learners are categorized into two types, either "local" or "global", according to their dependency/sensitivity to sample selection bias.

Definition 1.1 *The output of a* ***Local Learner*** *depends asymptotically only on $P(y|\mathbf{x})$. [Zadrozny, 2004]*

Definition 1.2 *The output of a* ***Global Learner*** *depends asymptotically both on $P(\mathbf{x})$ and $P(y|\mathbf{x})$. [Zadrozny, 2004]*

In the above definition, "the output of the learner" refers to the classifier constructed from the training set by a particular learner. These definitions were discussed in the context of the second type of sample bias [Zadrozny, 2004]. In [Zadrozny, 2004], several popular learners including Bayesian classifiers, naive Bayes, decision trees, logistic regressions as well as soft and hard margin support vector machines, are categorized as always being either "local" or "global" . This original categorization is independent from the particular application problem and only dependent on the learner. However, several strong assumptions are implicitly made, as discussed below, regarding the model space of the classifiers and the interpretation of $P(y|\mathbf{x})$ used in the definition of "local" and "global".

In the discussion of Bayesian classifiers, logistic regression, and hard-margin support vector machine always to be sample bias independent "local" classifiers, the following two assumptions are made implicitly.

Assumption 1.1 *The learner outputs a classifier θ that produces the probability $P(y|\mathbf{x}, \theta)$ to approximate/estimate the model independent true probability $P(y|\mathbf{x})$.*

Assumption 1.2 *For* ***Bayesian classifiers, logistic regression, and hard-margin support vector machine:*** *The learner is a consistent estimator of the true probabilities $P(y|\mathbf{x})$. That is, the estimated probability $P(y|\mathbf{x}, \theta)$ equals the true model independent probability $P(y|\mathbf{x})$ when the training data is exhaustive. Formally, $\forall \mathbf{x} \lim_{|D|\to\infty} P(y|\mathbf{x}, \theta) = P(y|\mathbf{x})$.*

In this notation, θ denotes the classifier or "the output of the learner" constructed from training data D. (A summary of all notations is in Figure 2.) The above assumptions are strong since the individuality of different problems and different datasets is not considered. In this paper, we relax these assumptions and take these differences into account. We argue that Bayesian classifiers, logistic regression and hard-margin support vector machines can be either "local" or "global" classifiers. This is dependent on the particular dataset to which these learners are applied.

On the other hand, when naive Bayes, decision tree and soft-margin support vector machines are characterized as "global" classifiers in [Zadrozny, 2004], the following assumption is made that replaces Assumption 1.2.

Assumption 1.3 *For* ***naive Bayes, decision tree, and soft-margin support vector machines:*** *The learner's model space does not contain the true model and hence is inconsistent. Formally, $\exists \mathbf{x} \lim_{|D|\to\infty} P(y|\mathbf{x}, \theta) \neq P(y|\mathbf{x})$.*

Similarly, this assumption is also strong because the differences between datasets are not considered either. We can also argue that naive Bayes, decision trees and soft-margin support vector machines could be either "local" (invariant to sample bias) or "global" (affected by sample bias) when the differences between datasets are accounted for. We generalize this argument to say that most of the known inductive learners could behave either as "local" or "global" learners, depending on the particular dataset as well as the inductive bias implied by the learning algorithm. When the true model of the particular dataset is contained in the model space, the learner would be local. Nonetheless, when the true model is not contained in the model space, the learner would be global.

2 Restrictions of Existing Categorization

To fully understand the assumptions and the practical implications of the "global" and "local" categorization, it is necessary to formally define some of the important notations and quantities, particularly, $P(\mathbf{x})$ and $P(y|\mathbf{x})$ used in Definitions 1.1 and 1.2.

2.1 Formal Definitions

In [Zadrozny, 2004], $P(\mathbf{x})$ is defined as "a global distribution over the entire input space", and a global classifier is

- $\mathbf{x}$ is feature vector, y is class label, and $s = 1$ denotes that an example $(\mathbf{x}, y)$ is selected into the training set D.
- $P(s = 1|\mathbf{x}, y)$ formally describes sample selection bias, and it denotes the probability that an example $(\mathbf{x}, y)$ is selected into the training set.
- $P(s = 1|\mathbf{x}, y) = P(s = 1|\mathbf{x})$ is true for the second type of sample selection bias where s is only dependent on the feature vector $\mathbf{x}$ and it is independent from class label y.
- $P(\mathbf{x})$ is the probability distribution of feature vector $\mathbf{x}$ and it is not related to either class label y or sample selection bias s.
- $P(y|\mathbf{x})$ denotes the true conditional probability for a feature vector $\mathbf{x}$ to be a member of class y. It is completely determined by the true concept or true function that an inductive learner is to model. The true function is typically unknown unless the dataset is synthesized. $P(y|\mathbf{x})$ is independent from training data as well as sample selection bias.
- Θ is the model space assumed by a learner.
- θ is a classifier constructed by a learner by searching in the model space Θ given training data D. By definition, θ is dependent on both Θ and D.
- $P(y|\mathbf{x}, \theta)$ denotes the probability for an example $\mathbf{x}$ to be of class y, as estimated by a classifier θ. Typically, $P(y|\mathbf{x}, \theta) \neq P(y|\mathbf{x})$, in other words, the estimated probability may not be equal to the true probability.
- Since θ is dependent on both Θ and D, we define $P(y|\mathbf{x}, D, \Theta)$ to represent the same probability as $P(y|\mathbf{x}, \theta)$, i.e., $P(y|\mathbf{x}, D, \Theta) = P(y|\mathbf{x}, \theta)$.

Figure 1. Summary of Symbols and Concepts

affected by sample selection bias "because the bias changes $P(\mathbf{x})$". Formally, $P(\mathbf{x})$ is the probability distribution solely as a function of the feature vector $\mathbf{x}$, and it is independent of class labels y. (Strictly speaking, if any feature x_i within the feature vector $\mathbf{x}$ is a continuous variable, $P(\mathbf{x})$ is a density function.) For example, assume that there are only two binary-valued features and each unique combination of feature values happens with equal chance. In this case, $\forall \mathbf{x}, P(\mathbf{x}) = 0.25$. Given the new formal definition in this paper, $P(\mathbf{x})$ is a problem dependent quantity and is independent from sample selection bias. Borrowing the notation of sample selection bias introduced in [Zadrozny, 2004], a global classifier's dependence on sample selection bias is best formalized through the dependence on $P(s = 1|\mathbf{x})$ rather than on $P(\mathbf{x})$. In general, $P(s = 1|\mathbf{x})$ is not related to $P(\mathbf{x})$.

Definition 2.1 ***Improved and General Definition of Global Learners****: A global learner's output depends asymptotically on both $P(s = 1|\mathbf{x}, y)$ and $P(y|\mathbf{x})$. Under the second type of sample selection bias, it depends on $P(s = 1|\mathbf{x})$ and $P(y|\mathbf{x})$.*

An important distinction between definitions 2.1 and 1.2 is that in the improved definition, a global learner's output is shown to directly be influenced by the chance of an instance being selected in the training set or the selection bias. However, in the original definition 1.2, the influence on the global learner is formulated on the chance of occurring a feature vector, that is actually independent from sample selection bias.

In [Zadrozny, 2004], $P(y|\mathbf{x})$ "refers to many local distributions, one for each value of $\mathbf{x}$". Strictly speaking, $P(y|\mathbf{x})$ denotes the true conditional probability distribution or the posterior probability distribution for a feature vector $\mathbf{x}$ to be a member of class y. The true class label for $\mathbf{x}$ is generated according to $P(y|\mathbf{x})$. $P(y|\mathbf{x})$ is completely determined by some unknown true function, and is unrelated to either the training data D or the model space of the inductive learner. Obviously, the true probability $P(y|\mathbf{x})$ is independent from both the sample selection bias due to training data D and the inductive bias due to choice of hypotheses space of a particular algorithm. In reality, for most practical applications where the dataset is not synthesized, the true probability distribution $P(y|\mathbf{x})$ is not known either before or after training some model. The availability of true probability $P(y|\mathbf{x})$ to model is very different from the true class label y. Class labels y's are provided in the training data, and are essential for inductive learners to construct classifiers. However, the true probability $P(y|\mathbf{x})$ is not normally given. Even if a feature vector $\mathbf{x}$ in the training data has class label y, it is strongly biased to assume that $P(y|\mathbf{x}) = 1$ in general. One instance of $(\mathbf{x}, y)$ is just a single observation. By definition, $P(y|\mathbf{x})$ is the probability to observe class label y when $\mathbf{x}$ is sampled repeatedly.

2.2 Classifiers as Approximators of $P(y|\mathbf{x})$

Most inductive learning algorithms construct classifiers to either directly or indirectly measure, approximate and output the true probability $P(y|\mathbf{x})$ by searching for one or a set of hypotheses that are consistent with the training data D in some hypotheses space Θ specific to each learning algorithm. The choice of hypothesis is called inductive bias [Mitchell, 1997]. The model space for decision tree learning algorithm is the complete set of trees that tests each feature of the feature vector in different orders and with different splitting conditions. However, the model for logistic regression is the set of logistic regression formulas with different coefficients corresponding to each feature. For classification algorithms that do not directly output conditional probabilities, they either output a score (specific to each algorithm) or predict the "most likely" class label. The scores can be normalized into conditional probability estimates. When scoring is unavailable, we interpret the estimated probability for the predicted class as 1, and 0 for all others. In summary, classifiers can be represented as approximators of the true conditional probability.

Formally, we use the notation $P(y|\mathbf{x}, D, \Theta)$ to denote a classifier constructed by some learner to approximate

$P(y|\mathbf{x})$. In this notation, D is the training set of examples with $s = 1$ or $D = \{\forall(\mathbf{x}, y) \in \mathcal{D} \wedge s = 1\}$, Θ is the model space implied by the learning algorithm, such as the complete set of decisions trees. The learning algorithm searches for a model (or an ensemble of hypotheses) $\theta \in \Theta$ that is consistent with the training data D. When there are multiple hypotheses consistent with the training data D, preferences are given to certain models. We could use θ to replace the dependency on both D and Θ in the notation, i.e., we define $P(y|\mathbf{x}, \theta) = P(y|\mathbf{x}, D, \Theta)$. In our notation, θ is the "output of the learner" used in the original definitions of "local" and "global" or Definitions 1.1 and 1.2. The chosen model θ specifies the exact procedures to compute and approximate the true probability $P(y|\mathbf{x})$. If θ is a decision tree, it specifies the order of feature tests, the threshold value for continuous variable tests, and the number of examples belonging to different classes at the leaf nodes.

2.3 Learner Consistency at Estimating Probabilities

The classifier θ is trained to estimate/approximate the true probability $P(y|\mathbf{x})$. However, the estimated probability $P(y|\mathbf{x}, \theta)$ may not be equal to $P(y|\mathbf{x})$, if the true model or a model that produces $P(y|\mathbf{x})$ is not contained in the model space of the learner. We say that a learner is consistent if the learning algorithm can find a model θ that is equivalent to the true model at producing class conditional probabilities given an exhaustive training data set. Formally, a learner is consistent if it can find a model θ from an exhaustive number of examples such that $\forall \mathbf{x} \lim_{|D| \to \infty} P(y|\mathbf{x}, \theta) = P(y|\mathbf{x})$. For example, if a decision tree algorithm is used to approximate a non-vertical and non-horizontal linear function that separate the space into two classes, it will never be able to find a tree with 0 error rate. At best, a decision tree approximates the linear function with steps. Clearly, the consistency of a particular learner depends on the dataset that it is applied to. In other words, the same learner could be consistent for some dataset but inconsistent for others. Verification of learner consistency for an arbitrary dataset is a difficult problem. To the best of our knowledge, there is no published work to exhaustively test a learner's consistency for an arbitrary dataset. A complete answer is impossible for realistic problems with infinite number of examples and unknown true function, such as mortgage application and catalog campaigns. Based on the above analysis, we propose to relax Assumptions 1.2 and 1.3.

Assumption 2.1 ***Relaxed assumption of Assumption 1.2 and 1.3*** *We do not assume the learner's consistency, i.e., it could be either consistent or inconsistent.*

2.4 Limited Utility of $P(y|\mathbf{x}, s = 1) = P(y|\mathbf{x})$

It is true by the definition of the second type of sample selection bias that $P(s|y, \mathbf{x}) = P(s|\mathbf{x})$ in [Zadrozny, 2004]. Re-writing by the definition of conditional probability, it becomes $\frac{P(s,y,\mathbf{x})}{P(y,\mathbf{x})} = \frac{P(s,\mathbf{x})}{P(\mathbf{x})}$. Re-arranging the denominators and using the definition of conditional probabilities, it becomes obvious that $P(y|s = 1, \mathbf{x}) = P(y|\mathbf{x})$. Following the definition of training dataset D, the dependency on $s = 1$ can be replaced by a dependency on the training set D, i.e., $P(y|D, \mathbf{x}) = P(y|\mathbf{x})$.

$P(y|s = 1, \mathbf{x}) = P(y|\mathbf{x})$ is used in [Zadrozny, 2004] to argue that some algorithms, e.g., Bayesian classifier, logistic regression and SVM, are independent from the second type of sample selection bias. However, this is only true under assumption 1.2 that the learner is consistent. However, as discussed in Section 2.2, the estimated probability is actually $P(y|\mathbf{x}, \theta)$, and we generally cannot assume the learners are consistent for many realistic problems. One deeper interpretation is that any consistent learner is "sufficiently and necessarily" local under the second type of sample selection bias since $\big(P(y|s = 1, \mathbf{x}) = P(y|\mathbf{x})\big) \Longleftrightarrow \big(P(s = 1|\mathbf{x}, y) = P(s = 1|y)\big)$.

3 Analysis of Popular Algorithms

In [Zadrozny, 2004], Bayesian classifiers, logistic regression, and hard-margin support vector machines are argued to be "local" classifiers that are insensitive to sample selection bias. However, naive Bayes, decision trees, and soft-margin support vector machines have been argued as "local" learners that are sensitive to sample selection bias. We argue that all these learners can be either global or local, and this depends on both the learner and dataset, but not on the learner itself.

3.1 Bayesian Classifiers

In the analysis of Bayesian classifiers to be "local" [Zadrozny, 2004], the following equation is used

$$\frac{P(\mathbf{x}|y, s = 1)P(y|s = 1)}{P(\mathbf{x}|s = 1)} = P(y|\mathbf{x}, s = 1) = P(y|\mathbf{x})$$

The above analysis does not consider the dependency on the model space of the learner or Θ. For a Bayesian classifier, Θ describes exactly how to estimate $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x}|s = 1)$ from the training data D with sample selection bias. In fact, $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x}|s = 1)$ are $P(\mathbf{x}|y, s = 1, \Theta)$ and $P(\mathbf{x}|s = 1, \Theta)$ respectively. By definition of D, the dependency on $s = 1$ can be replaced by dependency on D. These two probabilities can be then represented as $P(\mathbf{x}|y, D, \Theta)$ and $P(\mathbf{x}|D, \Theta)$ instead. As an example, suppose that we have a dataset with all categorical features. Obviously, $P(\mathbf{x}|y, D, \Theta)$ is the ratio of all examples in the training data D with class label y that also have feature vector $\mathbf{x}$. Now, the problem is to decide what numbers to divide to calculate this ratio. For simplicity, we assume that each feature vector $\mathbf{x}$ is unique. In this case, if the choice is to consider every feature x_i in the feature vector $\mathbf{x}$, then $P(\mathbf{x}|y, D, \Theta) = \frac{1}{|D_y|}$ if $\mathbf{x}$ has class label y, otherwise 0. In this notation, D_y is the subset of examples in the training dataset D that have class label y. $P(\mathbf{x}|D, \Theta)$ is $\frac{1}{|D|}$ because each feature vector $\mathbf{x}$ is assumed to be unique. Taking everything into consideration, $P(y|\mathbf{x}, D, \Theta) = 1$ if $\mathbf{x}$ has class label y or 0 otherwise. This computation is straightforward but not very useful, since there is no generalization and it is equivalent to "rote learning". The problems come from the strong assertions to consider every feature x_j in the feature vector $\mathbf{x}$. In reality, we normally only consider a "subset" of features in the feature vector $\mathbf{x}$ in order for the algorithm to generalize. The exact subset of features to consider depends on the training data D. Since Θ actually represents these inevitable choices, the dependency on Θ cannot be ignored. This problem becomes more complicated when some features are continuous, as there are infinite number of choices to either discretize the continuous features or split them into halves. In summary, due to the inevitable assertions and choices, it is generally very hard to compute $P(y|\mathbf{x})$ exactly for Bayesian classifiers. As a matter of fact, we still compute $P(y|\mathbf{x}, D, \Theta)$, and neither the dependency on D nor the dependence on Θ can be removed.

We provide an example to show that Bayesian classifier can also be "global", as opposed to the analysis in [Zadrozny, 2004] that it is always "local". Assume that the feature vector can be decomposed into two disjoint subsets $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$. Let the true conditional probability only depend on $\mathbf{x}_1$, i.e., $P(y|\mathbf{x}) = P(y|\mathbf{x}_1)$, and the selector variable depend only on $\mathbf{x}_2$, i.e., $P(s = 1|\mathbf{x}) = P(s = 1|\mathbf{x}_2)$. Under this situation, the choice of features to compute $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x}|s = 1)$ decides how much the sample selection bias will influence these estimated quantities from the data. If luckily, only those features $\in \mathbf{x}_1$ are taken into account to compute $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x}|s = 1)$, then the effect of sample selection bias will be rather small. However, if any features $\in \mathbf{x}_2$ are chosen to compute $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x}|s = 1)$, the estimated probability will reflect sample selection bias. In reality however, we normally do not know if the feature vector could be decomposed into the two disjoint subsets. In sophisticated situations, s for different examples could depend on a different feature subset. In the same time, the choice of features to compute $P(\mathbf{x}|y, s = 1)$ and $P(\mathbf{x})$ could be rather arbitrary. Because of limited number of examples in the training set, some combination of feature values could result in very few or even no examples in the training set. When this happens, the estimated probability can be statistically unreliable due to trivial sample size.

3.2 Naive Bayes Classifier

In [Zadrozny, 2004], naive Bayes is argued to a global classifier that is affected by sample selection bias since in general $\frac{P(x_1|y,s=1)\ldots P(x_n|y,s=1)P(y|s=1)}{P(\mathbf{x}|s=1)} \neq \frac{P(\mathbf{x}|y,s=1)P(y|s=1)}{P(\mathbf{x}|s=1)}$. However, this does not imply that the naive Bayes classifier is always a "global classifier" for any datasets. For some datasets where the independence assumption holds true, naive Bayes computes exactly $\frac{P(\mathbf{x}|y,s=1)P(y|s=1)}{P(\mathbf{x}|s=1)} = P(y|\mathbf{x}, s = 1)$, which is $P(y|\mathbf{x})$ according to the assumption of the second type of sample selection bias. A simple example that satisfies the independence assumption is one whose true label is only dependent on one feature, and all other features are irrelevant, i.e., $\exists i, P(y|\mathbf{x}) = P(y|x_i)$.

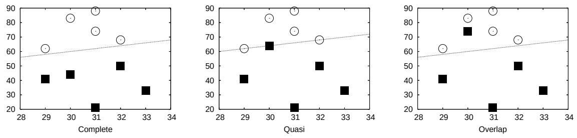


Figure 2. Logistic Regression

3.3 Logistic Regression

In [Zadrozny, 2004], logistic regression is described as

$$P(y = 1|\mathbf{x}, s = 1) = \frac{1}{1 + \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n)}$$

Logistic regression is argued to be free from the second type of sample selection bias in [Zadrozny, 2004] as "because we are assuming that y is independent of s given $\mathbf{x}$ we have that $P(y = 1|\mathbf{x}, s = 1) = P(y = 1|\mathbf{x})$". This assumption ignores the effect of D on θ, or the set of parameters β_i in this case. A further proof would be required to justify that β_i's are not affected by $s = 1$ for any dataset D. However, this could be very difficult since $\beta_i's$ are computed from D by minimizing log-likelihood function through Newton's method. In [Zadrozny, 2004], a simple example of one independent variable is shown to justify the claim that logistic regression is not affected by sample selection bias. Although this example is indeed correct, it is over simplified. The true function for the given example is a simple linear separable function $P(1|x \geq -0.75) = 1$.

The likelihood equation of logistic regression, both binary response and ordinal response (> 2 classes), is not guaranteed to have a finite solution. The existence of maximum likelihood estimate depends on the configuration of chosen training examples [So, 1999]. There are three mutually exclusive and exhaustive categories, i.e., complete separation, quasicomplete separation and overlap. Consider a binary response model of classes $y = 1$ and $y = 2$. In **complete separation**, there exists a vector $\mathbf{b}$ that correctly allocates all observations to their response groups, that is $\begin{cases} \mathbf{b}'\mathbf{x} > 0 & y = 1 \\ \mathbf{b}'\mathbf{x} < 0 & y = 2 \end{cases}$ This corresponds to the simplest case of "linear separatability" where the maximum likelihood estimate does not exist and there exists multiple vectors $\mathbf{b}$'s that can equally separate the data points completely. The exact solution for the same labeled training set, i.e., which particular $\mathbf{b}$'s is chosen, depends on the implementation. During the iterative process to fit the logistic regression model, the negative log-likelihood decreases to 0. As long as all the data points in the domain $\mathbf{x}$ are linearly separatable, logistic regression is completely free from any sample selection bias including the second type of sample selection bias as discussed in [Zadrozny, 2004]. Although the exact vector $\mathbf{b}$ selected by the iterative process may still be dependent on the sample selection bias. However, this difference is insignificant since each satisfying $\mathbf{b}$ separates the data points equally well. In the **quasicomplete separation** case, the data points are not "linearly" separable, and there exists a unique vector $\mathbf{b}$ such that $\begin{cases} \mathbf{b}'\mathbf{x} \geq 0 & y = 1 \\ \mathbf{b}'\mathbf{x} \leq 0 & y = 2 \end{cases}$. Similar to the complete separation case, the maximum likelihood estimate does not exist. However, during the iterative process, the log-likelihood does not diminish to 0 as in the case of complete separation. If neither complete separation nor quasicomplete separation exists, there is an **overlap** of sample points. For every vector $\mathbf{b}$ drawn in the sample space, there is always a sample point of different classes on the same side of the vector. In this case, maximum likelihood exists and is unique although there could be multiple vectors $\mathbf{b}$ that equally converge to the maximum likelihood. Both quasicomplete and overlapping cases are sensitive to the second type of sample selection bias. As shown in Figure 2, quasicomplete is sensitive to the placement sample points that fall exactly on the vector $\mathbf{b}$. If we introduce on additional example that crosses to the "wrong" side of $\mathbf{b}$, the problem becomes "overlapping". For overlapping problems, all satisfying vectors $\mathbf{b}$ forms an envelope. The introduction of one example that crosses to the "wrong" side of this envelope will change the maximum likelihood.

3.4 Decision Trees

Decision trees are argued to be "global" classifier independent from the dataset and problem in [Zadrozny, 2004]. However, a decision tree could also be a "local" classifier. The decision path of a tree tests a sequence of features starting from the root of tree to the current node. Without loss of generality, assume that decision path is $(x_1, x_2, \ldots, x_k)$, which is a true subset of the full feature vector, or $\subset \mathbf{x} = (x_1, x_2, \ldots, x_n)$. Assume that each feature is categorical. Then $P(y|\mathbf{x}, s) = P(y|\mathbf{x})$ implies $P(y|x_1, x_2, \ldots, x_k, s) = P(y|x_1, x_2, \ldots, x_k)$ if $x_{k+1}, \ldots, x_n$ are irrelevant at predicting the class label y but may be exclusively used to determine if the instances are selected into the training set.

3.5 Support Vector Machines (SVM)

In [Zadrozny, 2004], the hard-margin SVM algorithm is argued to be a local learner, while the soft-margin SVM algorithm is argued to be a global learner. The hard-margin SVM algorithms learn the parameters a and b describing a linear decision rule, $h(\mathbf{x}) = \text{sign}(a \cdot \mathbf{x} + b)$ whose sign determines the label of an example, so that the smallest distance between each training example and the decision boundary, i.e. the margin, is maximized. Given a sample of examples $(\mathbf{x_i}, y_i)$, where $y_i \in \{-1, 1\}$, it accomplishes margin maximization by solving the following optimization problem: minimize: $V(a, b) = \frac{1}{2} a \cdot a$, subject to: $\forall i: \ y_i[a \cdot \mathbf{x_i} + b] \geq 1$. The constraint requires that all examples in the training set are classified correctly, i.e., that the data can be separated by a hyperplane. If this is indeed the case, sample selection bias will not asymptotically affect the output of the hard-margin SVM algorithm as argued by [Zadrozny, 2004]. However, because this algorithm does not have a solution if the data is not linearly separable it cannot be applied in most practical cases.

The soft-margin SVM algorithms introduces slack variables $\xi_i > 0$ for each example $(\mathbf{x_i}, y_i)$. The optimization is changed to minimize: $V(a, b, \xi) = \frac{1}{2} a \cdot a + C \sum_{i=1}^{n} \xi_i$, subject to: $\forall i: \ y_i[a \cdot \mathbf{x_i} + b] \geq 1 - \xi_i, \ \xi_i > 0$ If a training example lies on the wrong side of the decision boundary, the corresponding ξ_i is greater than 1. Therefore, $\sum_{i=1}^{n} \xi_i$ is an upper bound on the number of training errors. The factor C is a parameter that allows one to trade off training error and model complexity.

In [Zadrozny, 2004] it is argued that sample selection bias affects the soft-margin SVM because it can change the sum of ξ_i values by making regions of the feature space denser than others. When this sum is changed, the decision boundary is also changed. Therefore, the soft-margin SVM algorithm is characterized as a global leaner. While this is true in general, like other learners, the soft-margin SVM algorithm can also behave as a local learner depending on the specific dataset used. In particular, if the data is linearly separable, the sum of ξ_i values will be always zero and sample selection bias will not asymptotically affect the output. This is also the case if the minimum of the sum is not changed by the bias (for example if the bias only affects examples that are on the correct side of the boundary).

4 Experimental Studies

We empirically validate the claims in the paper. The thesis is that a classifier learner cannot always be global or local. In fact, whether it is local or global depends on the combination of the data set, modeling assumptions of the learner and their appropriateness to model the particular dataset. In [Zadrozny, 2004], it has already been shown that decision tree and naive Bayes classifiers are global and affected by sample bias, and that logistic regression and hard-margin SVM are local and not be effected by sample bias. It remains to show that naive Bayes and decision trees can also be local and not be affected by sample bias, while logistic regression and hard-margin SVM can be still effected by sample bias.

Naive Bayes and Decision Tree Can be Local Classifiers We construct an artificial data set for naive Bayes where $\exists i, P(y|x) = P(y|x_i)$ as described in Section 3.2. We generate 100 random datasets of 1000 data points described by 20 Boolean variables (T or F). For each data set, one of the 20 variables is selected to determine the class label. Each feature value is generated with equal likelihood. For the chosen variable x_i that decides the class label, we set $P(y = +|x_i) = 1/i$. We then generate test sets in exactly the same way. To produce the biased training sets, we select a variable x_j among the remaining variables (i.e. excluding x_i) and set sample selection bias $P(s = 1|x_j = \mathrm{T})$ randomly in between 0 and 1. No other variable effects the

Figure 3. Examples where naive Bayes and Decision trees Behave as Local Learners

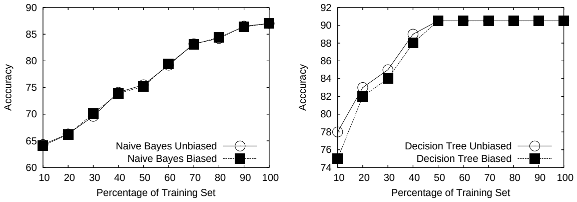
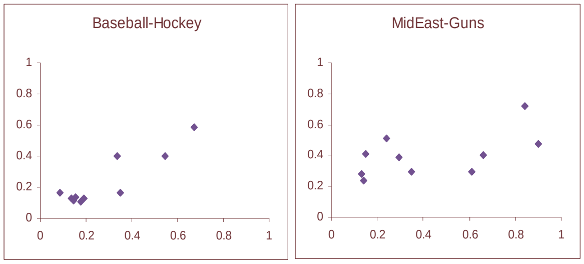


Figure 4. Typical scatter plots of mean number of top ten word for newsgroup data sets



sample selection bias. Though quite simple, it is a clear example of the situation where the true model is contained in the model space of the learner. The comparison of the naive Bayes models constructed from biased and unbiased training sets are shown in the left plot of Figure 3. It clearly demonstrates that naive Bayes in this circumstance is not effected by sample bias and hence is a local learner.

For decision trees, we create the situation discussed in Section 3.4. We randomly divide the 20 Boolean feature vector into two equal sized separate disjoint subsets of variables $\mathbf{x}_i$ and $\mathbf{x}_j$. The class probability is only dependent on $\mathbf{x}_i$ not on $\mathbf{x}_j$ as for all combination of attributes values for $\mathbf{x}_i$ we set $P(y = +|\mathbf{x}_i)$ randomly as either 0.1 or 0.9. To introduce sample selection bias, we choose one variable x_j from the subset $\mathbf{x}_j$ i.e. $x_j \in \mathbf{x}_j$ and set the selection bias as $P(s = 1|x_j = \mathrm{T}) = 1/j$, no other variable effects the sample selection. We generated 100 training and test sets with 1000 examples each as before. The results are summarized in the right plot of Figure 3. Clearly, the learner is not effected by sample bias since the true model is in the model space and the sample bias does not effect the tree generated.

Text Datasets For the newsgroup datasets, the training and test data sets are drawn from similar but not identical distributions. These data sets are of particular importance since they contain sample bias, and it is very likely that the unknown true model may not be in the model space of logistic regression, decision tree, naive Bayes or support vector machine classifiers. Since the chance of encountering a particular class is the same in the training and test sets, hence $P(y|s = 1) = P(y)$. We show below when $P(y|s = 1) = P(y)$, the only possible selection bias is due to the feature vector $\mathbf{x}$ but not y.

$$\begin{aligned}
& P(s=1|\mathbf{x},y) && \text{Bayes Theorem} \\
&= \frac{P(y,\mathbf{x}|s=1)P(s=1)}{P(\mathbf{x},y)} && \text{Product Rule} \\
&= \frac{P(\mathbf{x}|s=1).P(y|\mathbf{x},s=1)P(s=1)}{P(\mathbf{x},y)} && \text{No Class Bias} \\
&= \frac{P(\mathbf{x}|s=1).P(y|\mathbf{x})P(s=1)}{P(x,y)} && \text{Conditional Probability} \\
&= \frac{P(\mathbf{x},s=1).P(\mathbf{x},y)P(s=1)}{P(s=1)P(\mathbf{x})P(\mathbf{x},y)} && \text{Cancellation} \\
&= \frac{P(\mathbf{x},s=1)}{P(\mathbf{x})} && \text{Conditional Probability} \\
&= P(s=1|\mathbf{x})
\end{aligned}$$

We perform experiments on the 20 Newsgroups datasets using the standard “by-date” division into training (60%) and test (40%) sets based on posting date. This division creates a temporal bias. For example, in the Guns newsgroup the word “Waco” occurs extensively in news articles in the training set but not in the test set, as interest in the topic fades. This can be seen visually by examining the relative frequency of word occurrences (the attribute values) in the training and test sets in the scatter plots of Figure 4. This verifies the existence of sample selection bias, and based on our derivation above, it is the second type of sample selection bias. We used the tool Rainbow to extract features from news articles. The feature vector for a document consists of the frequencies of top ten words by selecting words with highest mutual information with the class variable. On the other hand, the unbiased training and tests sets were created by randomly shuffling all of the newsgroup data and selecting training and test sets irrespective of the news item posting date. The results shown in Table 1 illustrate that the hard-margin SVM and logistic regression learners can be adversely effected by sample bias (i.e. the Baseball-Hock, MidEast-Guns and Mac-Religion data sets) as well as being invariant to sample bias (the other three data sets).

5 Related Work

The sample selection bias problem has received a great deal of attention in econometrics. There it appears mostly because data are collected through surveys. Very often people that respond to a survey are self-selected, so they do not constitute a random sample of the general population. In Nobel-prize winning work, [Heckman, 1979] has developed a two-step procedure for correcting sample selection bias in linear regression models, which are commonly used in econometrics. The key insight in Heckman’s work is that if we can estimate the probability that an observation is selected into the sample, we can use this probability estimate to correct the model. The drawback of his procedure is that it is only applicable to linear regression models In the statistics literature, the related problem of missing data has been considered extensively [Little and Rubin, 2002]. However, they are generally concerned with cases in which some of the features of an example are missing, and not with cases in which whole examples are missing. The literature in this area distinguishes between different types of

Technique	Baseball	Christian	MCyles	MidEast	MidEast	Mac
	vs Hock.	vs Sale	vs Guns	vs Guns	vs Elec	vs Relig.
NB	22.2% (24.6%)	12.0% (12.3%)	11.5% (10.5%)	11.0% (10.7%)	17.4% (17.2%)	19.4% (21.1%)
DT(C4.5)	13.3% (15.7%)	8.7% (7.9%)	8.2% (10.8%)	10.5% (20.3%)	17.4% (17.2%)	14.1% (18.4%)
LogR	11.9% (11.4%)	8.1% (8.0%)	7.8% (8.1%)	10.15% (10.3%)	17.4% (17.2%)	11.1% (10.7%)
SVM	24.8% (26.1%)	8.2% (8.4%)	15.4% (14.2%)	18.4% (21.4%)	17.4% (17.2%)	22.9% (23.6%)

Table 1. Error of various techniques on newsgroup data from unbiased training sample and biased training sample (shown in parentheses)

missing data mechanisms: missing completely at random (MCAR), missing at random (MAR) and not missing at random (NMAR). Different imputation and weighting methods appropriate for each type of mechanism have been developed. More recently, the sample selection bias problem has begun to receive attention from the machine learning and data mining communities. The publication extended in this paper [Zadrozny, 2004] presents a new categorization of the behavior of learning algorithms under sample selection bias (global learners vs. local learners) and analyzes how a number of well-known classifier learning methods are affected by sample selection bias. The main shortcoming of this work is that they do not consider the effects of incorrect modeling assumptions on the behavior of the classifier learner under sample selection bias. In other words, the work implicitly assumes that the data is drawn from a distribution that could be perfectly fit by the model. Smith and Elkan [Smith and Elkan, 2004] provide a systematic characterization of the different types of sample selection bias and examples of real-world situation where they arise. For the characterization, they use a Bayesian network representation that describes the dependence of the selection mechanism on observable and non-observable features and on the class label. They also present an overview of existing learning algorithms from the statistics and econometrics literature that are appropriate for each situation. Finally, Rosset et al. [Rosset et al., 2005] consider the situation where the sample selection bias depends on the true label and present an algorithm based on the method of moments to learn in the presence of this type of bias.

6 Conclusion and Future Work

Addressing sample selection bias is necessary for data mining in the real world for applications such as merchandise promotion, clinical trial, charity donation, etc. One very important problem is to study the effect of sample selection bias on inductive learners. One recent work categorizes several learners sensitivity on one very common form of sample selection bias, where the chance to select an example into the training set depends on feature vector $\mathbf{x}$ but not directly on class label y. A learner is categorized as "local" if it is insensitive to this type of sample selection bias. Bayesian classifier, logistic regression and hard-margin SVM's are argued to be "local", with a strong and implicit assumption that the true model is contained in the model space of these learners regardless of the dataset that they are applied to. On the other hand, a learner is categorized as "global" if it is sensitive to sample selection. Decision trees, naive Bayes and soft-margin SVM are argued to be always "global", with a strong and different assumption that the true model is not contained in their model space.

In this paper, we formalize the definitions of sample selection bias, and make the local and global categorizations more rigorous. In particular, we distinguish the difference between true generative probability and model estimated probability. We generalize these important categorizations by relaxing the assumptions made previously. We argue formally and by examples that most of these popular learners could be either local or global, and it all depends on the combination of the dataset and the learner. In general, a learner could be local for some datasets and global for others. In addition, we also show formally and by examples that when there is no class label bias, the only possible sample selection bias comes from the dependency on feature vectors.

Future Work Our paper raises the question of how to categorize classifiers when there is no pre-assumption or prior knowledge about if the true model is contained in the model space of a learner. In our future work, we will propose a method based on bias-variance decomposition to quantify the sensitivity to selection bias.

References

[Heckman, 1979] Heckman, J. (1979). Sample selection bias as a specification error. *Econometrica*, 47:153–161.

[Little and Rubin, 2002] Little, R. and Rubin, D. (2002). *Statistical Analysis with Missing Data*. Wiley, 2nd edition.

[Mitchell, 1997] Mitchell, T. M. (1997). *Machine Learning*. McGraw Hill.

[Rosset et al., 2005] Rosset, S., Zhu, J., Zou, H., and Hastie, T. (2005). A method for inferring label sampling mechanisms in semi-supervised learning. In *Advances in Neural Information Processing Systems 17*, pages 1161–1168. MIT Press.

[Smith and Elkan, 2004] Smith, A. and Elkan, C. (2004). A bayesian network framework for reject inference. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 286–295.

[So, 1999] So, Y. (1999). A tutorial on logistic regression. Technical report, SAS Institute Inc, Cary, NC.

[Zadrozny, 2004] Zadrozny, B. (2004). Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21th International Conference on Machine Learning*.