Contents

1 Binary classification under sample selection bias

Matthias Hein

The problem of general sample selection bias is studied from a decision-theoretic perspective in the case of binary classification. We show necessary and sufficient conditions for the equivalence of the Bayes classifiers of training and test distribution and give bounds for the excess risk if they disagree. Moreover, we show without any assumptions on the type of sample selection bias that the knowledge about unlabeled data allows one to identify regions where the sign of the regression functions of training and test is guaranteed to coincide.

In the second part we use the insights gained from the theoretical analysis. We provide a nonparametric framework for learning under general sample selection bias motivated by a modified cluster assumption. The connection to semi-supervised learning is discussed. Further, we present experimental results for data sets with explicit control of the selection bias.

1.1 Introduction

In econometrics and sociology it is widely accepted that often the sample one uses for learning or estimation comes from a different distribution than the one used in testing. In the machine learning community only very recently this problem has been discussed [Zadrozny, 2004, Smith and Elkan, 2004]. The reason for this might be that one can argue that sample selection bias only occurs due to a bad choice of the training set. We agree that this can be the reason for sample selection bias, however there are problems where even the most careful choice of the training set would not prevent sample selection bias. One example is the prediction of the income of people based on a questionnaire. Usually the richer people are the less they tend to answer such a questionnaire. Clearly the prediction based on the data of the questionnaire will be biased towards low income. Another case is when we have only training data from some proportion of the test population. This occurs if a bank wants to predict if someone who is applying for a loan will eventually repay it. The credit bank has only data from customers whose loan has been approved. This set of customers will be generally a biased sample of the whole population or

the set of potential customers.

In the machine learning literature so far the main emphasis has been laid on a special kind of sample selection bias, the so called covariate shift [Shimodeira, 2000, Sugiyama and Müller, 2005, Huang et al., 2007], where the conditional distribution $p(y|x)$ of training and test distribution is the same. For the general sample selection bias problem several parametric models have been proposed in the econometrics literature, see e.g. Heckman [1979], Winship and Mare [1992], Dubin and Rivers [1989]. In this article we study the general scenario of sample selection bias, in particular we derive necessary and sufficient conditions for the equivalence of the Bayes classifiers of training and test distributions. Moreover, we analyze the situation where one has access to an unlabeled sample of the test distribution which can be either a part of the training data which has not been labeled or an independent sample of the marginal test distribution. A similar approach with the goal of identifying the possible range of probability measures responsible for the training data without making any prior assumptions on the sampling process has been studied by Manski and Horowitz [Manski, 1989, Horowitz and Manski, 2006].

Originating from this analysis we propose a new nonparametric principle to deal with sample selection bias in the case where one has access to unlabeled test data. The setting where one has unlabeled test data is similar to semi-supervised learning. However, in semi-supervised learning one assumes that training and test data come from the same distribution. We show that implementing the new principle via adaptive regularization leads to an algorithm which is similar to existing ones for semi-supervised learning [Zhu et al., 2003, Zhou et al., 2004]. Whereas the performance is similar when training and test data come from the same distribution, the new algorithm performs better in cases where also the conditional distribution changes. Therefore this algorithm can also be seen as an extension of semi-supervised learning which is robust to sample selection bias.

1.2 Model for sample selection bias

In this chapter we consider binary classification. Our goal is to learn a classifier $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are the input and output domain. For binary classification we have $\mathcal{Y} = \{-1, +1\}$. We assume that there exists a (stationary) distribution P on $\mathcal{X} \times \mathcal{Y}$, the true distribution of X and Y. However, we are only given a biased sample. This can be formally described using a random binary selection variable s, where $s = 1$ means that we accept the point for the training sample and $s = 0$ means that we will not observe it in the training phase. Of interest is how $p(y|x, s = 1)$, the conditional distribution of the training sample, behaves with respect to the true conditional distribution $p(y|x)$.

We always assume in the following that both probability measures P_{tr} and P_{te} have densities p_{tr} and p_{te} with respect to some dominating measure e.g. if $\mathcal{X} = \mathbb{R}^d$ we take as the dominating measure the Lebesgue measure. Thus we avoid an overly technical presentation. However, the results still hold in the general case.

In order to keep the presentation as clear as possible we will keep the explicit dependency on the selection variable s. We give the following dictionary to be consistent with the notation in the rest of the book.

$$\begin{aligned} \text{training distribution} \quad & p_{\text{tr}}(y,x) = p(y,x|s=1) \\ \text{test distribution} \quad & p_{\text{te}}(y,x) = p(y,x) \end{aligned}$$

We further assume in the following that the sampling of training and test data is done i.i.d. from $p(y,x|s=1)$ and $p(y,x)$ respectively.

A central role will be played by $p(s=1|x,y)$, that is the probability that a given joint pair (x,y) is observed in the training sample. The following relationships can be derived by straightforward application of Bayes' rule,

$$\begin{aligned} p(y|x,s=1) &= \frac{p(y|x)p(s=1|x,y)}{p(s=1|+,x)p(+|x)+p(s=1|-,x)p(-|x)}, \\ p(x|s=1) &= \frac{p(s=1|x,+)p(+,x)+p(s=1|x,-)p(-,x)}{\int_X p(s=1|x,+)p(+,x)+p(s=1|x,-)p(-,x)\,dx}, \end{aligned} \tag{1.1}$$

where we have introduced the shorthand notation $+$ and $-$ for $y=1$ and $y=-1$ e.g. $p(+,x)$ for $p(y=1,x)$ and $p(-,x)$ for $p(y=-1,x)$. Since we can estimate $p(y|x,s=1$ and $p(x|s=1)$ from the training data, it is more interesting to express the quantities of the test data in terms of the training data.

$$\begin{aligned} p(y|x) &= \frac{p(y|x,s=1)p(x|s=1)p(s=1)+p(y|x,s=0)p(x|s=0)p(s=0)}{p(x|s=1)p(s=1)+p(x|s=0)p(s=0)} \\ &= \frac{p(s=1|x)}{p(s=1|y,x)}p(y|x,s=1) \\ p(x) &= p(x|s=0)p(s=0)+p(x|s=1)p(s=1) = \frac{p(s=1)}{p(s=1|x)}p(x|s=1). \end{aligned} \tag{1.2}$$

Using these relations one can characterize different special cases of sample selection bias.

Random selection This is the case we usually assume to be true in standard binary classification. The selection is completely random, that is independent of x and y. This implies $p(s=1|y,x) = p(s=1)$. Obviously we have in this case $p(y|x,s=1) = p(y|x)$ and $p(x|s=1) = p(x)$ so that the distributions of training and test data are identical.

Class conditionally independent selection In this case the selection is independent of the class label y given the feature x or equivalently given x the knowledge of the selection variable s gives no information about the class label y. Due to this property this scenario is sometimes called 'missing at random (MAR)'. We have

$$p(s|x,y) = p(s|x) \quad \Longleftrightarrow \quad p(y|x,s) = p(y|x).$$

The conditional probabilities of training and test data agree and therefore also the Bayes classifiers of training and test data. However, the marginal distribution of the training data is in general different,

$$p(x|s=1) = \frac{p(s=1|x)p(x)}{\int_X p(s=1|x)p(x)\,dx}.$$

Sometimes this scenario is also called *covariate shift*, see Shimodeira [2000], Sugiyama and Müller [2005]. Note that the support of the training data has to be contained in the support of the test data. Under covariate shift we can trust the labels we are given, but the true/test marginal distribution is different from the training distribution. In this case often reweighting of the loss function is done in order to get an unbiased estimate of true loss, see e.g. Manski [1977], Shimodeira [2000], Huang et al. [2007]. We will come back to this issue in a later section.

Class dependent selection In this case the selection variable s is independent of the feature x given the label y,

$$p(s|x,y) = p(s|y) \quad \Longleftrightarrow \quad p(x|y,s) = p(x|y).$$

This means that the class conditional distributions stay the same. However, the class probabilities $p(y|s=1)$ and $p(y)$ of training and test data differ and thus the class conditional probabilities $p(y|x,s=1)$ and $p(y|x)$ are different as well. In particular one has

$$p(y|s=1) = \frac{p(s=1|y)p(y)}{p(s=1|+)p(+) + p(s=1|-)p(-)},$$
$$p(y|x,s=1) = \frac{p(s=1|y)p(y|x)}{p(s=1|+)p(+|x) + p(s=1|-)p(-|x)}.$$

Having knowledge about $p(s=1|y)$ or equivalently the true class probabilities $p(y)$ one can easily correct for the modification of $p(y|x,s=1)$. Namely by setting $p(+|x) = p(-|x) = \frac{1}{2}$ we observe that the threshold for a Bayes-optimal decision with respect to the test distribution is given by

$$\gamma = \frac{p(s=1|+)}{p(s=1|+) + p(s=1|-)},$$

that is we decide for $+$ if $p(+|x,s=1) > \gamma$ and for $-$ otherwise. This problem is closely related to cost-sensitive learning, see Elkan [2001]. Suppose that $c_{-1,1}$ denotes the cost of predicting the positive class when the negative is true and $c_{1,-1}$ the corresponding opposite cost. It is then easy to show that the Bayes optimal threshold γ, that is one predicts $+$ if $p(+|x) > \gamma$, is given by

$$\gamma = \frac{c_{-1,1}}{c_{-1,1} + c_{1,-1}}.$$

We observe that both expressions are equal if we identify $c_{-1,1} = p(s=1|+)$ and $c_{1,-1} = p(s=1|-)$. Thus the costs tell us how we should change the training

distribution such that for the test distribution we can decide with the normal threshold $\frac{1}{2}$.

Note, that in practice one often artificially balances the classes for training in order to get a better estimate of the decision boundary, in particular if the classes are very unbalanced. The process of balancing can be equivalently seen as a class dependent selection. However, the correction for this simple form of sample selection bias is straightforward using the modified threshold which we introduced above.

The general case Sample selection bias is a very general model for differing training and test distributions. In this paragraph we will analyze conditions on the probability measures P_{tr} and P_{te} such that P_{tr} can be seen as selected from P_{te}. Not all different training and test distributions can be modelled in such a way. The first basic requirement is the *support condition*: the support of the probability measure of the training data has to be a subset of the support of the probability measure of the test data.

Suppose that the support condition holds and we are given the densities of the joint measures $p_{\mathrm{tr}}(y,x)$ and $p_{\mathrm{te}}(y,x)$ of training and test distribution. Does there exist a sampling mechanism such that one can see p_{tr} as $p_{\mathrm{tr}}(y,x) = p(y,x|s=1)$ and $p_{\mathrm{te}}(y,x) = p(y,x)$? We can check this using

$$p(y,x) = p(y,x|s=0)p(s=0) + p(y,x|s=1)p(s=1).$$

The part $p(y,x|s=0)p(s=0)$ can be modelled arbitrarily. We are searching for a nontrivial solution with $p(s=1) > 0$. For every $(y,x) \in \mathcal{Y} \times \mathcal{X}$, we require

$$p(y,x) - p(s=1)p(y,x|s=1) \geq 0.$$

Thus the *selection condition*, a necessary and sufficient requirement that p_{tr} can be seen as generated by selecting from p_{te}, can be stated as

$$\sup_{(y,x)\in\mathcal{Y}\times\mathcal{X}} \frac{p_{\mathrm{tr}}(y,x)}{p_{\mathrm{te}}(y,x)} < \infty,$$

where we are slightly sloppy regarding the supremum[1]. If the above condition holds then we can model P_{tr} as being selected from P_{te}. The probability of selection $p(s=1)$ is upper bounded as $p(s=1) \leq \inf_{y,x} \frac{p_{\mathrm{te}}(y,x)}{p_{\mathrm{tr}}(y,x)}$.

Note that the support condition rules already out some cases. Namely $p_{\mathrm{tr}}(y|x) > 0$ is not possible if $p_{\mathrm{te}}(y|x) = 0$. Secondly, suppose $\mathcal{X} = \mathbb{R}^d$ and both measures have a marginal density with respect to the Lebesgue measure. Then the selection condition rules out cases where $p_{\mathrm{te}}(x) = 0$ and $p_{\mathrm{tr}}(x) > 0$. But also the tails of training and test distribution have to be well-behaved. Suppose both have a Gaussian density

1. The support condition can be equivalently formulated that for any measurable set A it holds $\mathrm{P}_{\mathrm{te}}(A) = 0 \;\Rightarrow\; P_{\mathrm{tr}}(A) = 0$. Thus P_{tr} is absolutely continuous with respect to P_{te} which implies by the Radon-Nikodym theorem that there exists a density $f \in L_1(\mathcal{X})$ such that $\mathrm{P}_{\mathrm{tr}}(A) = \int_A f \, d\mathrm{P}_{\mathrm{te}}$. Then the selection condition is given by, $\|f\|_\infty < \infty$.

with different means but equal covariance. Then the quotient $\frac{p_{\mathrm{tr}}(x)}{p_{\mathrm{te}}(x)}$ can not be upper bounded on $\mathbb{R}^d$. On the positive side one can make the following statement.

Lemma 1.1 *Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^d$ and suppose the probability measures P_{tr} and P_{te} have continuous marginal densities with respect to the Lebesgue measure. Let further the support condition hold. If $p_{\mathrm{te}}(x) > 0$ for all $x \in \mathcal{X}$ and $\sup_{x\in\mathcal{X}} \max_{y\in\{-1,1\}} \frac{p_{\mathrm{tr}}(y|x)}{p_{\mathrm{te}}(y|x)} < \infty$ then P_{tr} and P_{te} can be modelled in the sample selection framework.*

Proof: We decompose the selection condition into

$$\sup_{(y,x)\in Y\times\mathcal{X}} \frac{p_{\mathrm{tr}}(y,x)}{p_{\mathrm{te}}(y,x)} \leq \sup_{x\in\mathcal{X}} \frac{p_{\mathrm{tr}}(x)}{p_{\mathrm{te}}(x)} \sup_{x\in\mathcal{X}} \max_{y\in\{-1,1\}} \frac{p_{\mathrm{tr}}(y|x)}{p_{\mathrm{te}}(y|x)}.$$

The first supremum is finite since both $p_{\mathrm{tr}}(x)$ and $p_{\mathrm{te}}(x)$ are continuous and therefore both achieve their maximum and minimum due to compactness of $\mathcal{X}$ with $\inf_x p_{\mathrm{te}}(x) > 0$ by assumption. The second supremum is finite by assumption. Let us finally discuss the situation where the support of training and test distribution differ. There are in principle two situations. If the training distribution has probability mass on a set where the test distribution has not, then the information about this set is completely useless for learning on the remaining test set without making assumptions on the relation of training and test distribution. For us this means that we can safely discard this information and instead work with the probability measure $\mathrm{P}_{\mathrm{tr}}\Big(y, x | x \in \mathrm{supp}\,(\mathrm{P}_{\mathrm{te}})\Big)$, where $\mathrm{supp}\,(\mathrm{P}_{\mathrm{te}})$ is the support of the test distribution. On the other hand if the test distribution has probability mass where the training distribution has not then we cannot hope to make any useful predictions on this portion of the test distribution without any further assumptions on how training and test data have been generated. The support condition seems therefore not to be too restrictive.

1.3 Necessary and sufficient conditions for the equivalence of the Bayes classifier

The essential element for classification is the conditional distribution $p(y|x)$ We have seen in the previous section that in the case of covariate shift one has $p(y|x, s = 1) = p(y|x)$. The goal of this section is to analyze the general case of sample selection bias. In particular, we are interested under which conditions the Bayes classifier of training and test data agree. Since this is a much weaker condition than equivalence of the conditional distribution $p(y|x)$, this is usually said to be the reason why classification is easier than regression. Moreover, we give an exact expression of the excess error of the Bayes classifier of the training distribution compared to the error of the Bayes classifier of the test distribution. This will allow us to characterize cases where sample selection bias does not matter substantially.

We define the regression functions η_{tr} and η_{te} and the Bayes classifiers b_{tr} of b_{te}

of the training and test distribution as,

$$\eta_{\rm tr}(x) = 2p(+|x, s = 1) - 1, \qquad \eta_{\rm te}(x) = 2p(+|x) - 1,$$
$$b_{\rm tr}(x) = \operatorname{sign} \eta_{\rm tr}(x) \qquad b_{\rm te}(x) = \operatorname{sign} \eta_{\rm te}(x).$$

A necessary and sufficient condition that the Bayes classifiers agree is,

$$\eta_{\rm tr}(x)\, \eta_{\rm te}(x) \geq 0, \quad \forall\, x \in X.$$

Using essentially Equation 1.2 one can then derive necessary and sufficient conditions for the equivalence of the Bayes classifiers of training and test distribution. We give all results in terms of quantities related to the training distribution since this is the distribution we have access to. The statement about equivalence will depend on the *selection index*, which measures the amount of bias in the labels at a given point.

Definition 1.2 *The* ***selection index*** $s(x) : \mathcal{X} \to [-1, 1]$ *is defined as*

$$s(x) = \frac{p(s = 1|+, x) - p(s = 1|-, x)}{p(s = 1|+, x) + p(s = 1|-, x)}.$$

The following theorem will state the equivalence of the Bayes classifiers in terms of the selection index.

Theorem 1.3 *Let* $p(s = 1|y, x) > 0$ *for all* $x \in \mathcal{X}$ *and* $y \in \{-1, 1\}$. *The regression function of the test data* $\eta_{\rm te}$ *can be expressed as*

$$\eta_{\rm te}(x) = \frac{\eta_{\rm tr}(x) - s(x)}{1 - s(x)\eta_{\rm tr}(x)}.$$

The Bayes classifiers $b_{\rm te}$ *and* $b_{\rm tr}$ *of test and training distribution agree at* x *if and only if*

$$\big|\eta_{\rm tr}(x)\big| \geq \operatorname{sign}\Big(\eta_{\rm tr}(x)\Big)\, s(x).$$

Moreover the risk of the Bayes classifier $b_{\rm tr}$ *of the training distribution* $p(y, x|s = 1)$ *with respect to the test distribution* $p(y, x)$ *is given as*

$$R\big(b_{\rm tr}\big) = R\big(b_{\rm te}\big) + \int\limits_{\{x \mid |\eta_{\rm tr}(x)| < \operatorname{sign}(\eta_{\rm tr}(x))\, s(x)\}} \left|\frac{\eta_{\rm tr}(x) - s(x)}{1 - s(x)\eta_{\rm tr}(x)}\right| p_{\rm te}(x)\, dx.$$

Proof: Using $p(+|x) = p(+|x, s = 1)\frac{p(s=1|x)}{p(s=1|+,x)}$ we arrive after a straightforward calculation at

$$p(+|x) = \frac{p(+|x, s = 1)p(s = 1|-, x)}{p(s = 1|+, x) - p(+|x, s = 1)[p(s = 1|+, x) - p(s = 1|-, x)]}.$$

Using now $p(+|x) = \frac{\eta_{\text{te}}(x)+1}{2}$ and $p(+|x, s = 1) = \frac{\eta_{\text{tr}}(x)+1}{2}$ we get the result

$$\eta_{\text{te}}(x) = \frac{\eta_{\text{tr}}(x) - s(x)}{1 - s(x)\eta_{\text{tr}}(x)}.$$

Equivalence of the Bayes classifiers is given if $\eta_{\text{tr}}(x)\eta_{\text{te}}(x) \geq 0$ for all $x \in \mathcal{X}$. Since $s(x) \in [-1, 1]$ we have $s(x)\eta_{\text{tr}}(x) \leq 1$ and thus,

$$\eta_{\text{tr}}(x)\eta_{\text{te}}(x) \geq 0 \quad \Leftrightarrow \quad \eta_{\text{tr}}(x)^2 \geq \eta_{\text{tr}}(x)s(x),$$

which gives the desired result. Finally, the risk $R(f)$ of a function $f : \mathcal{X} \to \{-1, +1\}$ with respect to the test distribution is given as:

$$R(f) = R(b_{\text{te}}) + \mathbf{E}_X \left[I_{f(X)\eta_{\text{te}}(X)<0} |\eta_{\text{te}}(X)| \right].$$

Note, that $\eta_{\text{tr}}(x)\eta_{\text{te}}(x) < 0$ is equivalent to $b_{\text{tr}}(x)\eta_{\text{te}}(x) < 0$. Plugging in $b_{\text{tr}}(x)$ for the function f and the expressions for η_{te} and $\eta_{\text{tr}}(x)\eta_{\text{te}}(x) < 0$ finishes the proof.
The interpretation of Theorem 1.3 is not straightforward. From the form of the regression function η_{te} of the test distribution it becomes clear that $s(x)$ quantifies the amount of bias in the labels. If $s(x) \to \pm 1$ (we do not allow $s(x) = \pm 1$) then the training data is maximally biased. If $s(x)$ is positive, one has a bias towards the positive class and vice versa. Figure 1.1 shows the dependency of η_{te} on the selection index $s(x)$ and the regression function of the training data. Two statements can

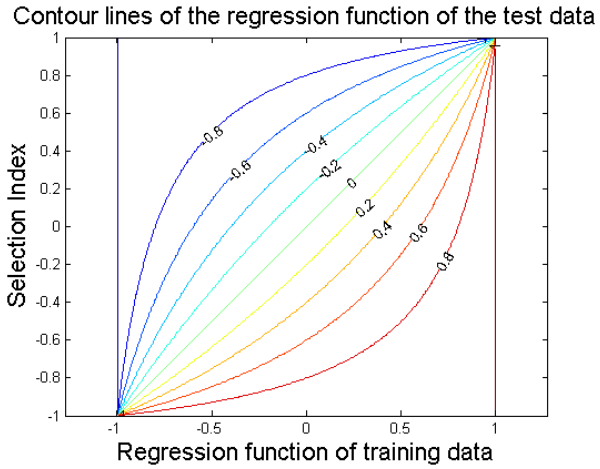

Figure 1.1 Contour lines of η_{te} in dependency of the selection index $s(x)$ and the regression function of the training data.

be made. We have $p(+|x) = p(+|x, s = 1)$ or equivalently $\eta_{\text{tr}}(x) = \eta_{\text{te}}(x)$ if and only if the selection index $s(x)$ is zero, that is $p(s = 1|+, x) = p(s = 1|-, x)$. This is the special case of sample selection bias often called covariate shift, where the labels are missing at random. However, this is a much stronger condition

then the derived condition for the equivalence of the Bayes classifiers. There as one could have expected we only need that the selection index $s(x)$ is zero at the decision boundary defined as $\{x \in \mathcal{X} \,|\, p(y|x) = \frac{1}{2}\}$. Away from the decision boundary one can allow for nonzero values of the selection index $s(x)$, that is $p(s = 1|+, x) \neq p(s = 1|-, x)$. In the "easy" regions where the training distribution is noise-free that is $\eta_{\text{tr}}(x) = \pm 1$ all non-zeros values for $p(s = 1|y, x)$ are allowed. However, note that e.g. $\eta_{\text{tr}}(x) = 1 \Leftrightarrow p(+|x, s = 1) = 1$ is only equivalent to $p(+|x) = 1$ if $p(s = 1|-, x) > 0$. In general if $p(s = 1|+, x) = 0$ or $p(s = 1|-, x) = 0$ then no statements about the conditional test distribution $p(y|x)$ can be made using knowledge about the conditional training distribution $p(y|x, s = 1)$.

If one has upper and lower bounds on $p(s = 1|y, x)$, one can derive the following corollary which gives an easier bound on the excess risk $R\big(b_{\text{tr}}\big) - R\big(b_{\text{te}}\big)$ than Theorem 1.3.

Corollary 1.4 *Assume $|s(x)| \leq \delta$ for all $x \in X$. Then the Bayes classifiers b_{te} and b_{tr} agree at x if $\big|\eta_{\text{tr}}\big| \geq \text{sign}\Big(\eta_{\text{tr}}(x)\Big)\delta$. The risk of the Bayes classifier of the training distribution b_{tr} with respect to the test distribution can be upper bounded as*

$$R\big(b_{\text{tr}}\big) \leq R\big(b_{\text{te}}\big) + \delta\, \text{P}_{\text{te}}\Big(\big|\eta_{\text{tr}}\big| < \delta\Big)$$

Proof: The Bayes classifier b_{tr} makes an error if $\big|\eta_{\text{tr}}\big| \geq \text{sign}\Big(\eta_{\text{tr}}(x)\Big)\delta$. Suppose $\eta_{\text{tr}}(x) > 0$, then an error happens if $\eta_{\text{tr}}(x) < s(x)$ and $s(x) > 0$. We have $|\eta_{\text{te}}(x)| = \frac{s(x) - \eta_{\text{tr}}(x)}{1 - s(x)\eta_{\text{tr}}(x)}$. A straightforward analysis shows that $|\eta_{\text{te}}(x)|$ is monotonically decreasing with increasing $\eta_{\text{tr}}(x)$. Therefore the maximum of $|\eta_{\text{te}}(x)|$ is attained at $\eta_{\text{tr}}(x) = 0$ and its value is δ. The same bound can be derived for the other case, which finishes the proof.
This corollary has a nice and easy interpretation. If the sampling process is not too nasty, that is δ is small, and the probability mass of the test distribution around the decision boundary of the training distribution is small, then using the Bayes classifier of the training distribution is not much worse than the Bayes classifier of the test distribution.

One can also tackle the problem from a different direction. Similar to cost-sensitive learning the optimal decision threshold under sample selection bias for $p(y|x, s = 1)$ with respect to the test distribution will in general not be $\frac{1}{2}$. In other words one can also define a new threshold function which leads then to an optimal decision with respect to the test distribution but *not* with respect to the training distribution. This can be done through knowledge about $p(s = 1|y, x)$.

Theorem 1.5 *Define the threshold function*

$$\text{Thresh}(x) = \frac{2\, p(s = 1|+, x)}{p(s = 1|+, x) + p(s = 1|-, x)},$$

and the new regression function $\overline{\eta_{\mathrm{tr}}}$ *of the training distribution as*

$$\overline{\eta_{\mathrm{tr}}}(x) = 2p(+|x, s = 1) - \mathrm{Thresh}(x).$$

If $p(s = 1|y, x) > 0, \forall\, x \in \mathcal{X}$*, then the new Bayes classifier* $\overline{b_{\mathrm{tr}}}(x) = \operatorname{sign} \overline{\eta_{\mathrm{tr}}}(x)$ *of the training distribution and the Bayes classifier of the test distribution* $b_{\mathrm{te}}(x)$ *agree for all* $x \in \mathcal{X}$*.*

Proof: Set $p(+|x) = p(-|x) = \frac{1}{2}$ in Equation 1.1, then one has $p(+|x, s = 1) = \mathrm{Thresh}(x)$.
Of course given only information about the training distribution there is no way to get any information about $p(s = 1|y, x)$. But this result indicates how one can improve the performance under sample selection bias if more information about the sampling mechanism is available.

1.4 Bounding the selection index via unlabeled data

In the last section we indicated how bounds on the selection index can help to identify parts of the the regression function η_{te}. By identification we mean that given complete knowledge about $p(y|x, s = 1)$ we can at least be sure about the sign of the regression function η_{te} of the test distribution in some regions and thus predict the correct label. The process of so called "partial identification of probability measures" has been pioneered by Manski [Manski, 1989, Horowitz and Manski, 2006].

In this section we will analyze the value of unlabeled data in order to determine bounds on the selection index. We will distinguish two situations. In the first one we assume that we know $p(y, x|s = 1)$ and we are given the marginal density $p(x)$ of the test distribution. Both quantities can be estimated consistently from a training sample $(X^{\mathrm{tr}}, Y^{\mathrm{tr}})$ and an independent unlabeled test sample X^{te}. In the second one we know $p(y|x, s = 1)$, the marginal density $p(x|s = 0)$ of the sample points which have not been selected to be labeled and the probability $p(s = 1)$ of being selected. This corresponds to a setting where we have an unlabeled sample $\{X_i^{\mathrm{te}}\}_{i=1,\ldots,T}$ of size T and then a subset of size S is being selected to be labeled yielding the training sample of labeled data $\{(X_j^{\mathrm{tr}}, Y_j^{\mathrm{tr}})\}_{j=1,\ldots,S}$ and a set of unlabeled data $\{X_i\}_{i=S+1,\ldots,T}$ where we assume without loss of generality that the data has been reordered after the selection. Note that $p(s = 1)$ can then be estimated via the ration S/T.

This distinction seems at first to be rather artificial. We illustrate both cases with an example. The first one corresponds e.g. to a test study of a new medical treatment. There one has information about the patients which decided to participate in the study. But usually no information is stored about the patients who refused to take part in the study. However, one might know the distribution of people where this medical treatment is supposed to be applied. This could be either the whole population or a certain subset. The second case where one has unlabeled data is

usually more generic. Assume a credit bank wants to assess how well their selection of customers works out. Potential customers are all persons who applied for a loan in the bank. The bank has labeled data of the customers who have been given a credit and they also have data about the customers who did not get one.

We see that both cases can occur in practice. In the second case one has the probability of selection $p(s = 1)$ as an important additional piece of information. We will see that without this information the knowledge about the marginal density $p(x)$ does not help to gain information about the selection index. However, given that we know $p(s = 1)$ also in the first case, then both cases are completely equivalent. This can be easily seen from

$$p(x) = p(x|s = 0)p(s = 0) + p(x|s = 1)p(s = 1),$$

where knowledge about $p(x|s = 1)$, $p(x)$ and $p(s = 1)$ identifies $p(x|s = 0)$ and vice versa. The following lemma restricts the selection index using information about $p(x)$ and $p(s = 1)$.

Lemma 1.6 *The selection index $s(x)$ can be bounded as,*

$$\operatorname{sign}(\eta_{\text{tr}}(x))s(x) \geq \frac{1 - p(s = 1|x)}{p(s = 1|x)}.$$

Thus the Bayes classifier of training and test data agree at $x \in \mathcal{X}$, if

$$|\eta_{\text{tr}}(x)| \geq \frac{1 - p(s = 1|x)}{p(s = 1|x)}.$$

Proof: One can decompose

$$p(s = 1|x) = p(s = 1|+, x)p(+|x) + p(s = 1|-, x)p(-|x).$$

Thus with $\lambda = p(+|x)$ we get $p(s = 1|+, x) = \frac{1}{\lambda}[p(s = 1|x) - (1 - \lambda)p(s = 1|-, x)]$ and plugging this into the expression for the selection index we can lower bound $s(x)$ for $\lambda \geq \frac{1}{2}$ as,

$$\begin{aligned} s(x) &= \frac{p(s = 1|x) - p(s = 1|-, x)}{p(s = 1|x) - (1 - 2\lambda)p(s = 1|-, x)} \geq \frac{p(s = 1|x) - p(s = 1|-, x)}{p(s = 1|x)} \\ &\geq \frac{p(s = 1|x) - 1}{p(s = 1|x)}. \end{aligned}$$

Therefore, for $\eta_{\text{te}} > 0$ the selection bias towards negative labels is lower bounded by $\frac{p(s=1|x)-1}{p(s=1|x)}$. Thus, if $\eta_{\text{tr}} < 0$ and $\eta_{\text{tr}} < \frac{p(s=1|x)-1}{p(s=1|x)}$ then we can be sure that also $\eta_{\text{te}} < 0$. The other direction follows by considering the case $\eta_{\text{te}} < 0$.

The second assertion follows directly from Theorem 1.3. However, the following proof is quite instructive. We have

$$p(+|x) = p(+|x, s = 0)p(s = 0|x) + p(+|x, s = 1)p(s = 1|x).$$

In particular

$$p(+|x, s = 1)p(s = 1|x) \leq p(+|x) \leq 1 - p(s = 1|x) + p(+|x, s = 1)p(s = 1|x).$$

Thus given that $p(+|x, s = 1) \geq \frac{1}{2}$ we have to ensure that $p(+|x) \geq \frac{1}{2}$ which using the inequality holds if $p(+|x, s = 1) \geq \frac{1}{2\,p(s=1|x)}$ or equivalently $\eta_{\text{tr}}(x) \geq \frac{1-p(s=1|x)}{p(s=1|x)}$. The other direction can be done similarly.
Note that $p(s = 1|x) = \frac{p(x|s=1)}{p(x)} p(s = 1)$ and therefore the quantity in the lower bound can be computed using the available knowledge about the marginal test density and the selection probability. Further, note that the bound is only nontrivial given that $p(s = 1|x) < \frac{1}{2}$. At a first glance, it might seem odd why the bound for the selection index has this strange form. This has a simple explanation. If $\eta_{\text{te}}(x) > 0$ and we have positive selection bias, then clearly $\eta_{\text{tr}}(x) > 0$ and vice versa. Therefore the selection index needs only be bounded with respect to the label of the training data e.g. if $\eta_{\text{tr}}(x) > 0$ then it could be that $\eta_{\text{te}} < 0$ and we have a positive selection bias. Thus only an upper bound on the positive selection bias is required.

Lemma 1.6 shows that using unlabeled data we can be sure about our estimated function wherever $|\eta_{\text{tr}}|$ is sufficiently large. This result holds without making any assumption on the form of the selection. Unfortunately, the bound is only nontrivial if $p(s = 1|x) < \frac{1}{2}$ or equivalently $p(x) < 2\,p(x|s = 1)p(s = 1)$. This condition holds in regions where the marginal test density is rather small with respect to the marginal training density. Thus the total mass of the test distribution of the region where this condition holds might be quite small.

1.5 Classifiers of small and large capacity

Until now we have analyzed how the Bayes optimal classifiers of training and test data are related. In this section we will discuss the difference of classifiers of small and large capacity in the case of sample selection bias. The first statement is an easy corollary of Theorem 1.3. Let us first recall the definition of a Bayes consistent classifier.

Definition 1.7 *A **Bayes or universally consistent classifier** is a sequence of classifiers f_n for which for every $\epsilon > 0$ and every probability measure on $\mathcal{X} \times \mathcal{Y}$,*

$$\lim_{n\to\infty} \mathrm{P}(R(f_n) - R(b) > \epsilon) = 0,$$

where n denotes the sample size, f_n is the selected classifier for a sample of size n and b is the Bayes classifier.

Corollary 1.8 *Let $p(s = 1|y, x) > 0$ for all $x \in \mathcal{X}$ and $y \in \{-1, 1\}$. Any Bayes consistent classifier trained on the biased sample is also Bayes consistent for the*

test distribution if and only if

$$\forall\, x \in \mathcal{X}, \quad \big|\eta_{\mathrm{tr}}(x)\big| \geq \operatorname{sign}\Big(\eta_{\mathrm{tr}}(x)\Big)\, s(x). \tag{1.3}$$

Formulating this (almost) trivial corollary in simple terms: at least in the asymptotic regime it does not matter if we train our classifier with the biased sample or the unbiased sample if condition 1.3 holds. The only criterion we have to fulfill is that we use a Bayes consistent classifier. For several classifiers Bayes consistency has been shown e.g. KNN-classifiers [Devroye et al., 1996] or the SVM with a Gaussian kernel [Steinwart, 2002] and many more results are known. A Bayes consistent classifier has asymptotically maximal capacity because in the limit as the sample size goes to infinity any target function can be learned.

For the moment we assume that the Bayes classifiers of training and test distribution are equal e.g. as in the covariate shift problem. What happens now if one uses a classifier of smaller capacity? A simple example shows that classifiers of small capacity can perform arbitrary badly even if the conditional distribution of training and test data agrees. As a classifier of large capacity we take the SVM with a Gaussian kernel and for the one with small capacity the SVM with a linear kernel. We use the checkerboard data illustrated in Figure 1.2. The sample selection bias is in this case just a covariate shift between training and test distribution. Since it is

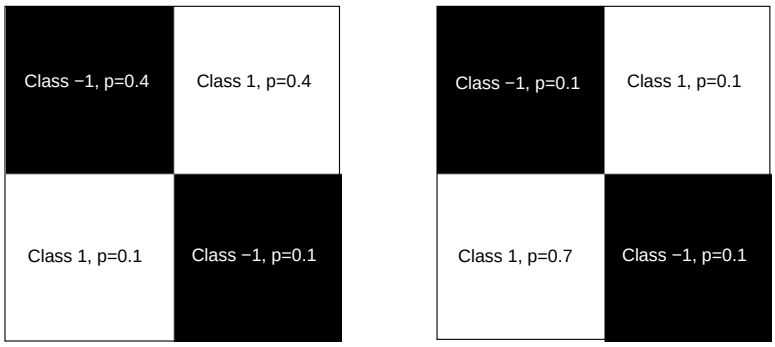

Figure 1.2 The checkerboard data as an example of covariate shift. Left: the training distribution, Right: the test distribution. Each square is sampled uniformly. The probability of each square is denoted by p.

noise-free, the optimal Bayes error is zero in that case. We test the learned classifier once on samples drawn from the training distribution and once on samples from the test distribution. Table 1.3 shows the mean errors together with the standard deviation over 20 runs for 200 training points. The test error of the SVM with Gaussian kernel increases significantly but the performance is still reasonable. The results of the linear SVM are hopeless. The linear SVM is significantly worse than

	Error on Train. Dist.	Error on Test. Dist.
SVM with linear kernel	28.8 ± 8.1	72.4 ± 17.0
SVM with Gaussian kernel	5.3 ± 2.6	7.9 ± 4.5

Figure 1.3 The mean error over 20 runs of the training and test data for the checkerboard data of Figure 1.2 for a SVM with linear and Gaussian kernel with 200 datapoints.

random guessing. Such a phenomenon is also known as *Anti-Learning*. It is obvious that one could modify the test distribution such that the error of the linear SVM would be even worse. It becomes clear from this simple experiment that classifiers of small capacity are much more sensitive to sample selection bias than classifiers of large capacity. This can also seen directly by comparing the loss with respect to training and test distribution:

$$\text{Training dist.}: \quad \mathbf{E}_{\text{tr}}\left[l(f(X),Y)\right] = \int_X p(x|s=1) \int_Y l(f(x),y)\, p(y|x,s=1)\, dy\, dx,$$
$$\text{Test dist.}: \quad \mathbf{E}_{\text{te}}\left[l(f(X),Y)\right] = \int_X p(x) \int_Y l(f(x),y)\, p(y|x)\, dy\, dx,$$

A classifier of large capacity can fit the function (almost) pointwise and therefore only the term $\int_Y l(f(x),y)\, p(y|x,s=1)\, dy$ matters. If we assume that the Bayes classifiers of training and test distribution agree then minimization of this part will lead in both cases to the same value of f at x. The weighting with $p(x|s=1)$ or $p(x)$ does then not matter anymore for determining the optimal function. A classifier of small capacity can only fit a limited amount of data and pointwise minimization is not possible. Therefore one has to minizmize $\mathbf{E}_{\text{tr}}\left[l(f(X),Y)\right]$ globally. In that case it matters a lot how the errors are weighted and therefore the weighting with $p(x|s=1)$ instead of $p(x)$ can lead to huge differences in the minimizer. In particular, for classifiers of small capacity it is therefore very important to reweight the loss with $g(x) = \frac{p(x)}{p(x|s=1)}$ given that one has information about the marginal test distribution $p(x)$:

$$\tfrac{1}{n}\textstyle\sum_{i=1}^n l(f(X_i),Y_i) \quad \longrightarrow \quad \tfrac{1}{n}\textstyle\sum_{i=1}^n l(f(X_i),Y_i) g(X_i).$$

For classifiers of large capacity the reweighting is (asymptotically) neither improving the results nor does it harm if one has covariate shift as sample selection bias. See Shimodeira [2000], Zadrozny [2004], Sugiyama and Müller [2005] and Chapters **??**, **??**, **??** and **??** for more on reweighting in this volume. For the SVM using different kernels one has to distinguish between kernels which lead to Bayes consistency and those which do not. In this respect the statement of Zadrozny [2004] that linear SVM's are asymptotically affected by sample selection bias can be sharpened to that SVM's are asymptotically affected by sample selection bias even if the Bayes classifiers agree if one is not using a kernel which leads to a Bayes consistent classifier.

Up to now we dealt with the special case of covariate shift. In the case of general sample selection bias it is not clear if reweighting is a good strategy. Note, that we know from Lemma 1.6 the larger $p(s=1|x) = \frac{p(x|s=1)}{p(x)} p(s=1)$ the more we

are sure about that $\text{sign}(\eta_{\text{tr}}(x)) = \text{sign}(\eta_{\text{te}}(x))$. However, the reweighting factor $g(x) = \frac{p(x)}{p(x|s=1)}$ is reciprocal to $p(s = 1|x)$ which means that by reweighting one downweights the regions of $\mathcal{X}$ where one is sure about that $\text{sign}(\eta_{\text{tr}}(x)) = \text{sign}(\eta_{\text{te}}(x))$. On the other hand one increases the weight of regions where one does not know if the signs of η_{tr} and η_{te} agree. It remains a point for future work to resolve this apparent contradiction.

1.6 A nonparametric framework for general sample selection bias using adaptive regularization

Basically we have seen in the last sections that without further assumptions on the nature of the sample selection bias there is no way to find the correct sign of $\eta_{\text{te}}(x)$. Using unlabeled data from the test distribution and information about the selection probability we could show that in regions where $|\eta_{\text{tr}}(x)| > \frac{1-p(s=1|x)}{p(s=1|x)}$ we can be sure that $\text{sign}(\eta_{\text{tr}}(x)) = \text{sign}(\eta_{\text{te}}(x))$. In order to make any assertions about the remaining regions we have to make assumptions how the selection bias was generated. The existing approaches for the general case of sample selection bias make explicit parametric assumptions on the relationship between training and test distribution e.g. the bivariate probit model of Heckman [Heckman, 1979] or other more general models [Dubin and Rivers, 1989]. It is often questionable if these assumptions hold in real data. A natural assumption should be one which is general enough to be true for a large class of data sets. In this remaining part we propose a nonparametric principle to deal with general sample selection bias under the assumption that one has a sample of unlabeled data from the test distribution and eventually knows the selection probability $p(s = 1)$. Both assumptions are fulfilled in the traditional setting of sample selection bias, see Section 1.4 for a discussion. The main underlying principle will be a modified cluster assumption. The cluster assumption has been proposed in semi-supervised learning (SSL) and can be formulated as follows.

Cluster assumption: Two points which can be connected by a path through high-density regions are likely to have the same label.

In semi-supervised learning one usually assumes that labeled and unlabeled data come from the same distribution. In the case of sample selection bias it makes only sense to use the cluster structure of the unlabeled data from the test distribution. Therefore we modify slightly the cluster assumption of SSL

Modified Cluster assumption: Two points which can be connected by a path through high-density regions *of the test data* are likely to have the same label.

We think that the modified cluster assumption is quite natural and holds for a large class of data sets.

The other important question is which part of the labels of the training data we should use. In principle, we know by Lemma 1.6 that without any assumptions we can only trust the sign of the regression function of the training distribution if $|\eta_{\text{tr}}(x)|$ exceeds a certain threshold. This implies that in the worst case we should

not use any information on Y of the training data in regions where $|\eta_{\text{tr}}(x)|$ is below the threshold. On the other hand if one has random selection or the labels are missing at random then we have $p(y|x, s = 1) = p(y|x)$ and it would be not reasonable to discard any label information. Both ways can be integrated into the learning framework using different weights in the loss function.

As learning framework we will use regularized empirical risk minimization,

$$f_n = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} l(f(X_i), Y_i)\, \gamma(X_i) + \mu\, \Omega(f),$$

where $\Omega : \mathcal{F} \to \mathbb{R}_+$ denotes the regularization functional, l the loss function and $\gamma : \mathcal{X} \to \mathbb{R}_+$ a weighting function. In order to implement the modified cluster assumption we need a regularizer Ω which enforces the cluster structure in the *test* data, that is it should prefer functions which are almost constant on the clusters and are allowed to change in between. A similar regularization principle is used in semi-supervised learning (SSL), where one uses unlabeled data to build graph-based regularizers which adapt to the cluster structure of training *and* test data, see Bousquet et al. [2004]. We will show that the adaptation to the cluster structure of the test data leads to a modification of an existing learning algorithm for SSL. Our experiments indicate that this modification leads to robustness against sample selection bias.

1.6.1 Adaptive graph-based regularization

Our input space $\mathcal{X}$ will be in the following always a compact subset of the d-dimensional Euclidean space $\mathbb{R}^d$. A regularizer which implements the cluster assumption for SSL where training and test distribution are equal can be built using a graph based on *training* and *test* data, see Bousquet et al. [2004], Hein [2006].

- take the sample of test and training data $\{X_i\}_{i=1}^n$ as the set of vertices,
- edge weight $w(X_i, X_j) = \frac{1}{h^d} k(\|X_i - X_j\| / h)$ if $\|X_i - X_j\| \leq h$, otherwise no edge, where $k : \mathbb{R}_+ \to \mathbb{R}_+$ is the kernel function, $h > 0$ is the neighborhood parameter of the resulting graph and d the dimension of the input space.

The data-dependent graph-based regularization functional is then defined as:

$$\tilde{S}_{n,h,\lambda}(f) = \frac{1}{2n^2\, h^2} \sum_{i,j=1}^{n} \frac{w(X_i, X_j)}{(d(X_i) d(U_j))^{\lambda}} (f(X_i) - f(X_j))^2,$$

where $d(X_i) = \frac{1}{n} \sum_{j=1}^{n} w(X_i, X_j)$ is the degree function. The parameter $\lambda > 0$ controls the influence of the density as can be seen from the following theorem.

Theorem 1.9 *[Hein, 2006] Let $\{X_i\}_{i=1}^n$ be an i.i.d. sample of a probability measure* P *on a compact set $\mathcal{X} \subset \mathbb{R}^d$. If $f \in C^3(\mathcal{X})$ and $h \to 0$ and $nh^d / \log n \to \infty$, then*

almost surely

$$\lim_{n\to\infty} \tilde{S}_{h,n,\lambda}(f) = \frac{C_2}{2C_1^{\lambda}} \int_{\mathcal{X}} \|\nabla f\|^2 \, p(x)^{2-2\lambda} \, dx,$$

where C_1, C_2 are constants depending on the kernel function k and the dimension d.

This theorem has been strengthened to uniform convergence over the class of Hölder functions on $\mathcal{X}$ and still holds when the data lies on a low-dimensional submanifold M, see Hein [2006]. Since $\|\nabla f\|$ is weighted by the density, this functional is only small if the function varies only very little in high-density regions, whereas variations in low-density regions are hardly penalized. However, in our setting it cannot be applied directly since training and test data are not from the same distribution. In order to implement the cluster structure of the *test* data we need the density of the test data in the limit functional. One can achieve this via reweighting of the regularization functional $\tilde{S}_{n,h,\lambda}(f)$. For simplicity we set $\lambda = 0$ in the following. But the results can be generalized to all values of λ. We have the following setting:

- $\{X_i^{\mathrm{tr}}\}_{i=1}^n$ from the training distribution $p(x|s=1)$,
- $\{X_j^{\mathrm{te}}\}_{j=1}^m$ from the test distribution $p(x)$.
- concatenated sample

$U = \{X_1^{\mathrm{tr}}, \ldots, X_n^{\mathrm{tr}}, X_1^{\mathrm{te}}, \ldots, X_m^{\mathrm{te}}\}$, $l = n + m$.

Then we define two kernel density estimators based on $\{X_i^{\mathrm{tr}}\}_{i=1}^n$ and $\{X_i^{\mathrm{te}}\}_{i=1}^m$,

- $d_{X^{\mathrm{tr}}}(x) = \frac{1}{n\, h_n^d} \sum_{i=1}^n k\big(\, \|x - X_i\|_{/}\, h_n\big)$,
- $d_{X^{\mathrm{te}}}(x) = \frac{1}{m\, h_m^d} \sum_{i=1}^m k\big(\, \|x - Z_i\|_{/}\, h_m\big)$,

where h_m and h_n are the bandwidth of the kernel density estimators. One can use alternatively any other (consistent) density estimator. An estimate[2] of the reweighting function $g(x) = \frac{p(x)}{p(x|s=1)}$ can be computed as $\hat{g}(x) = \frac{d_{X^{\mathrm{te}}}(x)}{d_{X^{\mathrm{tr}}}(x)}$ on the training points. We define:

$$\phi(U_i) = \begin{cases} \hat{g}(U_i) & , \quad \text{if} \quad i \le n, \quad \text{training points} \\ 1 & , \quad i > n, \qquad \text{test points.} \end{cases}$$

Moreover, we define the adaptive regularization functional which implements the modified cluster assumption as:

$$S_{l,h}(f) = \frac{1}{2l^2\, h^2} \sum_{i,j=1}^{l} w_{ij}\, \phi(U_i)\, \phi(U_j)\, (f(U_i) - f(U_j))^2,$$

where the weights $w_{ij} = w(U_i, U_j)$ are defined as before with a common scaling function h.

2. Estimates of a certain function g will be denoted by $\hat{g}$.

Theorem 1.10 *Let $\mathcal{X} \subset \mathbb{R}^d$ be compact and f, $p(x)$, $p(x|s=1) \in C^3(\mathcal{X})$. Furthermore let $p(x)$ and $p(x|s=1)$ be lower bounded and $p(x|s=1)$ be absolutely continuous with respect to $p(x)$, then if*

$$n \text{ finite, } m \to \infty,\ h_m \to 0 \text{ such that } m h_m^d \to \infty \text{ and } l\, h^d \to \infty.$$

or

$$n \to \infty,\ h_n \to 0 \text{ such that } n h_n^d / \log n \to \infty,$$
$$m \to \infty,\ h_m \to 0 \text{ such that } m h_m^d / \log m \to \infty,$$
$$\text{and } h = \max\{h_n, h_m\},$$

it holds almost surely,

$$\lim_{l \to \infty} S_{h,l}(f) = \frac{C_2}{2} \int_{\mathcal{X}} \|\nabla f\|^2 \, p(x)^2 \, dx,$$

where C_2 are constants depending on the kernel function k.

Proof: We sketch the proof which is similar to Hein [2006]. First one shows that $\hat{g}(U_i)$ and $g(U_i)$ are close with high probability. Furthermore the functional $S_{l,h}(f)$ can be decomposed in two one-sample U-statistics and one two-sample U-statistic. Then one uses Bernstein-type large deviation inequalities for U-statistics to show the convergence.

1.6.2 The learning problem

We have shown that the adaptive regularization functional $S_{l,h}(f)$ adapts to the cluster structure of the test data as desired. Similar to existing SSL algorithms, see Zhou et al. [2004], we formulate now the learning problem as a regularized least squares problem:

$$F = \operatorname*{arg\,min}_{f \in \mathbb{R}^l} \sum_{i=1}^{n} (f(X_i) - Y_i)^2 \hat{\gamma}(X_i) + \mu \, S_{l,h}(f), \tag{1.4}$$

where we reweight the loss with different functions $\hat{\gamma}$. In the functional $S_{l,h}(f)$ we use weights of the form

$$w'(U_i, U_j) = \begin{cases} 0 & \text{, if} \quad i, j \leq n, \\ w(U_i, U_j) & \text{, otherwise.} \end{cases}$$

The solution of this regularized least squares problem can be computed as the solution of the linear system

$$(\hat{\Gamma} + \mu \, \Delta_l') F = \hat{\Gamma} Y,$$

where $\Delta_l' = D' - W'$ is the graph Laplacian of the graph with weights $w'(U_i, U_j)$ and degree function $d'(U_i) = \sum_{j=1}^{l} w'(U_i, U_j)$. D' and $\hat{\Gamma}$ denote the diagonal matrices with the functions d', $\hat{\gamma}$ on the diagonal. Note that we have merged the remaining

factors of n, l and h in $S_{l,h}(f)$ into the regularization constant μ.

Three different weighting functions $\hat{\gamma}$ will be used in the loss function in the following.

- *Standard (SL):* $\hat{\gamma}(x) = 1$, standard least squares loss.
- *Reweighting I (RL1):* $\hat{\gamma}(x) = \hat{g}(x)$, if the sample selection type is random or the labels are misssing at random, this reweighting leads to an unbiased estimate of the true loss.
- *Reweighting II (RL2):* Let $\hat{f}$ be a classifier only based on the training data. Then we define for $c > 0$,

$$\hat{\gamma}(X_i) = \begin{cases} \hat{g}(X_i) & \text{, if } \quad |\hat{f}(X_i)| \geq c\, \frac{\hat{g}(X_i) - p(s=1)}{p(s=1)}, \\ 0 & \text{, otherwise} \end{cases} .$$

Note that $\frac{\frac{p(x)}{p(x|s=1)} - p(s=1)}{p(s=1)} = \frac{1 - p(s=1|x)}{p(s=1|x)}$. The last weighting function is motivated by Lemma 1.6 and only keeps the labels which are not potentially misleading. As a classifier $\hat{f}$ for the training data we use the SVM with a Gaussian kernel and the squared hinge loss since the minimizer of the squared hinge loss is given by the regression function

$$\eta_{\mathrm{tr}} = \arg\min_f \mathbf{E}_{\mathrm{P}_{\mathrm{tr}}} \left[\max\{0, 1 - Y\, f(X)\}^2 \right].$$

1.6.3 Difference to semi-supervised learning

The algorithm in Equation (1.4) looks very similar to existing SSL algorithms. However, there is a fundamental difference between SSL and our framework. Namely in SSL one assumes that training and test data come from the same distribution. A large class of SSL algorithms transfer the labels to the unlabeled points by propagating them along the data, thereby using manifold and cluster structure of the data. Since training and test data come from the same distribution, the cluster structure obviously coincides for training and test data. However, under sample selection bias this assumption need not hold. In general the cluster structure of training and test data will be different. Therefore under sample selection bias only the cluster structure of the *test* data should be used in order to propagate the labels. Therefore we have set in the proposed algorithm the adjacency matrix between the training points to zero. Thereby we ensure that label information only propagates along the test data. We would like to emphasize that the change of the adjacency matrix does not affect the limit of the regularization functional stated in Theorem 1.10.

The change of the adjacency matrix mainly makes a difference if the number of training points is larger or at least on the scale of the number of test points. In this case the proposed algorithm can also be seen as a robust extension of SSL. In the sense that the algorithm is robust to small differences between test and training distribution and performs as good as standard SSL when training and test distribution are equal. In the extreme case of SSL where one has only a few labeled

points but lots of unlabeled ones the difference of both approaches is negligible, since it is then likely that a labeled point is only connected to unlabeled points. We refer to Chawla and Karakoulas [2005] for further discussion of the relation of SSL and learning under sample selection bias.

1.7 Experiments

We have done experiments on a specific toy data set, where different types of sample selection bias could be easily simulated. We compare all combinations of the different losses, SL and RL1 and RL2, and standard and reweighted regularization functional, $\tilde{S}(f)$ resp. $S(f)$, abbreviated as SR and AR. The combination $SL+SR$ is very similar to existing SSL algorithms, see Zhu et al. [2003], Zhou et al. [2004].

In all experiments we use a symmetric kNN-graph with $k = \{5, 10, 20, 40\}$ and Gaussian weights, where the σ of the Gaussian is chosen as the average kNN-distance. The parameters h_n and h_m for the kernel density estimation are set as the average kNN-distance [3] for $k_n = \log(n) + 10$ and $k_m = \log(m) + 10$. For the regularization parameter μ we use $\log_{10} \mu = \{-4, -2, 0, 2, 4\}$. The best parameters are found by cross-validation. In order to be consistent with the loss we use for learning, we use for the cross-validation the same loss, that is SL, $RL1$ or $RL2$. In all experiments the total number of training and test points is fixed to 1000. All experiments are repeated 20 times. For numerical stability and to limit the influence of outliers we cut off the estimate of $\hat{g}$ used in $RL1$ and $RL2$ at 10 and 0.1. For the weighting function γ of the loss $RL2$ we need to determine the classifier $\hat{f}$ and the constant c. The classifier $\hat{f}$ is a SVM with squared hinge loss where we set the error parameter C to $C = 10$ in all experiments. We use the implementation described in Chapelle [2007]. The parameter c is determined in the following way. We choose the largest c such that at least half of the labels of the positive and negative class are used. Here we have a certain trade-off between keeping labeled data and discarding it in regions where we do not trust the labels.

In all experiments the test distribution is the same. The test class conditional distributions are two two-dimensional Gaussians of isotropic variance $\sigma = 0.6$. The means are at $(-1, 0)$ and $(1, 0)$. The class probabilities are equal, that is $p(+) = p(-) = 0.5$. The distribution is shown in Figure 1.4. We will always explore the two scenarios of unlabeled data discussed in Section 1.4.

- Unlabeled data type 1: As training data we have a sample from $p(y, x|s = 1)$. As unlabeled data we are given an independent sample of the marginal test density $p(x)$.
- Unlabeled data type 2: We are given a sample of the marginal test density $p(x)$.

3. This choice for h_n and h_m satisfies the condition of Theorem 1.10 since the kNN-distance R_k is prop. to $\left(\frac{k}{n}\right)^{\frac{1}{d}}$.

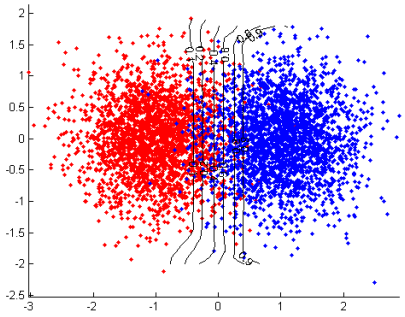

Figure 1.4 A sample of 5000 points of the test distribution with contour lines of $p(+|x)$.

Some of them are selected to be labeled via $p(s = 1|x)$ and the labels are drawn from $p(y|x, s = 1)$. This will be the training set. The rest of the sample which was not selected will be the unlabeled data. It has distribution $p(y, x|s = 0)$. The marginal test density is estimated in this case with all the samples.

We will always report results for both cases. The amount of unlabeled data of type 1 is chosen such that the amount of training and test data is equal for both cases. Moreover, apart from the test error for the parameters chosen by cross validation on the training set, we also report the minimal test error over all parameters. This is done in order to check if model selection works by cross validation. We will see in the case of general sample selection bias that this is not the case.

1.7.1 Random selection

This is the ideal learning scenario. Test and Training data come from the same distribution. Here using the losses SL and $RL1$ and the regularization SR and AR should not make any difference except that with RL and AR we expect more variance since the estimation of $\hat{g}(x) = \frac{d_{X^{\mathrm{te}}}(x)}{d_{X^{\mathrm{tr}}}(x)}$ is noisy. Our experimental results verify this fact. Astonishingly also using the loss $RL2$ does not lead to a significant reduction of performance despite the fact that we discard up to 50% of the labels. The reason is possibly that this dataset has a cluster structure and therefore our SSL-type algorithm performs well even with only a small number of labeled points.

1.7.2 Covariate Shift

In this scenario the conditional distributions of training and test data agree: $p(y|x, s = 1) = p(y|x)$. However, the marginal distribution $p(x|s = 1)$ and $p(x)$ differ. The selection probability $p(s = 1|x)$ has the form,

$$p(s = 1|x) = \begin{cases} \frac{8}{10}\frac{|x_1|}{1+|x_1|} & \text{, if} \quad x_1 < 0, \\ \frac{5}{10}\frac{|x_1|}{1+|x_1|} & \text{, otherwise.} \end{cases}$$

Table 1.1 Results for the random selection shift.

Unlabeled 1	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	$\mathbf{4.1 \pm 0.7}$	$\mathbf{4.1 \pm 0.8}$	$\mathbf{4.1 \pm 0.7}$	$\mathbf{4.1 \pm 0.8}$	$\mathbf{4.1 \pm 0.8}$	$\mathbf{4.1 \pm 0.7}$
Error from CV	$\mathbf{4.4 \pm 0.8}$	$\mathbf{4.4 \pm 0.8}$	4.7 ± 0.9	4.5 ± 0.8	4.5 ± 0.8	4.7 ± 1.0
Unlabeled 2	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	$\mathbf{4.3 \pm 1.0}$	$\mathbf{4.3 \pm 1.0}$	4.4 ± 1.1	$\mathbf{4.3 \pm 0.9}$	4.3 ± 1.0	4.4 ± 1.1
Error from CV	4.9 ± 1.1	$\mathbf{4.8 \pm 1.0}$	4.9 ± 1.1	4.9 ± 1.1	4.9 ± 1.0	5.0 ± 1.1

This form of selection implies that only a small number of points are sampled in the region of the decision boundary. Moreover, we sample more points of the red class than of the blue class, see Figure 1.5. This corresponds to a scenario where the training set was generated by only selecting cases where the label is obvious and where one has more samples of one class than the other one, despite in the test case both classes occur equally often.

The results show that neither one of the combinations is significantly better. As in the case of random selection the combinations with $RL2$-loss are slightly worse which is due to their reduced amount of labels they use. The loss on unlabeled data of type 2 is slightly higher than for type 2. The reason is that $p(x|s = 0)$ is more concentrated on the decision boundary than $p(x)$ and therefore one has more label noise.

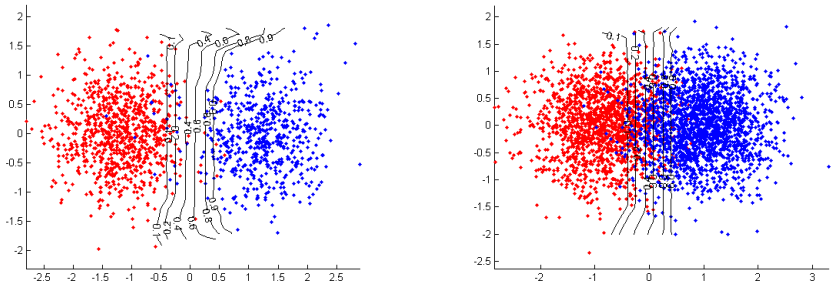

Figure 1.5 Covariate Shift: Left: The training set of the covariate shift data drawn from $p(y, x|s = 1)$, Right: The set of unlabeled points of type 2. Both plots show also the contour lines of $p(y|x, s = 1)$ resp. $p(y|x, s = 0)$ (the differences come from interpolation in Matlab).

Table 1.2 Results for the covariate shift.

Unlabeled 1	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	$\mathbf{4.7 \pm 0.9}$	$\mathbf{4.7 \pm 0.8}$	5.0 ± 0.9	$\mathbf{4.7 \pm 0.9}$	$\mathbf{4.7 \pm 0.8}$	4.9 ± 0.8
Error from CV	$\mathbf{5.3 \pm 0.9}$	5.4 ± 1.1	5.7 ± 1.0	5.4 ± 0.9	5.4 ± 0.8	5.6 ± 1.1
Unlabeled 2	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	$\mathbf{6.1 \pm 1.1}$	6.2 ± 1.1	6.5 ± 1.2	6.2 ± 1.0	$\mathbf{6.1 \pm 1.0}$	6.4 ± 1.2
Error from CV	6.8 ± 1.3	6.9 ± 1.4	7.2 ± 1.2	$\mathbf{6.7 \pm 1.1}$	$\mathbf{6.7 \pm 1.1}$	7.1 ± 1.2

1.7.3 General sample selection bias

In this case both the conditional probability and the marginal density differ between training and test data. The selection probability $p(s = 1|x)$ is given by

$$p(s = 1|x) = \frac{1}{2} \frac{\langle w, x_1 \rangle^2}{\langle w, x_1 \rangle^2 + 2} + \frac{1}{10}$$

and the conditional probability for the selected samples is given by

$$p(y|x, s = 1) = \theta(x)\, p_{\text{mod}}(y|x) + (1 - \theta(x))\, p(y|x),$$

where $\theta(x) = 0.8 \exp(-\frac{\|x\|^2}{20\sigma_2^2})$ with $\sigma_2 = 0.4$ and $p_{\text{mod}}(y|x)$ is the conditional distribution generated by two Gaussians with means at $(1, 1)$ and $(-1, -1)$ and variance $\sigma_2 = 0.4$ as class conditional probabilities $p(x|+)$ and $p(x|-)$ and equal class probabilities $p(+) = p(-) = 0.5$. Thus we can see the training conditional distribution $p(y|x, s = 1)$ as the test conditional distribution which is perturbed by another conditional distribution near the origin.

The results show that the use of the $RL2$-loss performs in this case significantly better than all other combinations of loss and regularizers. The adaptive regularizer is slightly better than the standard regularizers for the $RL2$-loss but the difference is not significant. The success of the $RL2$-loss is basically due to the fact that labels are discarded where we are not sure about them. This also helps to select the correct model as we can see from the minimal possible test errors over all parameters. All combinations of loss and regularizer have roughly the same minimal test error. The problem is that one can not identify the correct model using cross-validation on the training set since the conditional probability of the training data $p(y|x, s = 1)$ and the test data $p(y|x)$ differ. Therefore the $RL2$ loss outperforms the other losses since it discards labels in regions where we are not sure about them.

1.8 Conclusion

We have discussed the general problem of sample selection bias from a decision theoretic perspective. We have shown that the information about unlabeled data helps to restrict the difference between training and test distribution. It remains an open question if there exist other ways of characterizing additional information

Table 1.3 Results for general sample selection bias.

Unlabeled 1	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	5.0 ± 0.8	5.2 ± 0.9	4.9 ± 0.7	5.0 ± 0.8	5.0 ± 0.9	$\mathbf{4.8 \pm 0.8}$
Error from CV	7.3 ± 2.0	7.4 ± 1.7	6.4 ± 3.8	7.6 ± 1.8	7.5 ± 1.6	$\mathbf{5.3 \pm 0.8}$
Unlabeled 2	SL+SR	RL1+SR	RL2+SR	SL+AR	RL1 +AR	RL2 + AR
Min. Test Err.	5.7 ± 1.4	5.8 ± 1.3	5.6 ± 1.0	5.5 ± 1.3	5.5 ± 1.3	$\mathbf{5.2 \pm 0.8}$
Error from CV	9.0 ± 3.0	9.1 ± 3.1	6.3 ± 1.1	9.1 ± 3.2	9.6 ± 3.0	$\mathbf{6.0 \pm 0.9}$

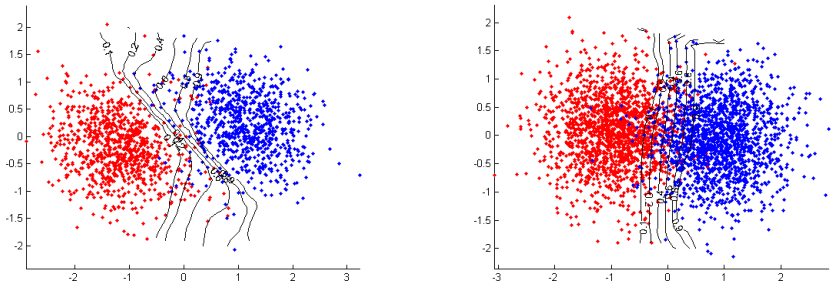

Figure 1.6 General Sample Selection: Left: The training data for the general sample selection problem, Right: Unlabeled data of type 2 with labels drawn from $p(y|x, s = 0)$. Both plots show also the contour lines of $p(y|x, s = 1)$ resp. $p(y|x, s = 0)$.

about the learning problem which could restrict the type of sample selection bias.

The problem of general sample selection bias cannot be solved without additional assumptions on the data generating process. Another open question is the characterization of natural assumptions on how training and test data are related. We have discussed a modified cluster assumption which seems reasonable for a large class of datasets. Another interesting direction would be the integration of causal relationships into a model about sample selection bias.

Acknowledgements

First of all I would like to thank Klaus Robert Müller for introducing me to the problem of sample selection bias. Furthermore, I would like to thank Arthur Gretton, Steffen Bickel and the organizers and participants of the NIPS workshop "When training and test distributions are different" for helpful and interesting discussions.

References

O. Bousquet, O. Chapelle, and M. Hein. Measure based regularization. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, volume 16. MIT Press, 2004.

O. Chapelle. Training a support vector machine in the primal. *Neural Computation*, 19:1155–1178, 2007.

N. V. Chawla and G. Karakoulas. Learning from labeled and unlabeled data: an empirical study across techniques and domains. *J. of Art. Int. Res.*, 23:331–366, 2005.

L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*. Springer, New York, 1996.

J. A. Dubin and D. Rivers. Selection bias in linear regression, logit and probit models. *Sociological Methods and Research*, 18:360–390, 1989.

C. Elkan. The foundations of cost-sensitive learning. In *Proc. of the 17th Int. Joint Conf. on AI (IJCAI)*, pages 973–978, 2001.

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

M. Hein. Uniform convergence of adaptive graph-based regularization. In G. Lugosi and H. Simon, editors, *Proc. of the 19th Conf. on Learning Theory (COLT)*, pages 50–64, 2006.

J. L. Horowitz and C. F. Manski. Identification and estimation of statistical functionals using incomplete data. *Journal of Econometrics*, 132:445–459, 2006.

J. Huang, A. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt, and T. Hofmann, editors, *Advances in Neural Information Processing Systems 19*, pages 601–608, 2007.

C. F. Manski. The estimation of choice probabilities from choice based samples. *Econometrica*, 45:1977–1988, 1977.

C. F. Manski. Anatomy of the selection problem. *J. of Human Resources*, 18:343–360, 1989.

H. Shimodeira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90:227–244, 2000.

A. Smith and C. Elkan. A Bayesian network framework for reject inference. In *Proc. of the 10th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 286–295, 2004.

I. Steinwart. Support vector machines are universally consistent. *J. Complexity*, 18:768–791, 2002.

M. Sugiyama and K.-R. Müller. Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, 23(4):249–279, 2005.

C. Winship and R. D. Mare. Models for sample selection bias. *Ann. Rev. Soc.*, 18:327–350, 1992.

B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In R. Greiner and D. Schuurmans, editors, *Proc. of the 21st Int. Conf. on Machine Learning (ICML)*, pages 114–122, 2004.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, volume 16, pages 321–328, 2004.

X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In T. Fawcett and N. Mishra, editors, *Proc. of the 20nd Int. Conf. on Machine Learning (ICML)*, 2003.