Instance Weighting for Patient-Specific Risk Stratification Models

Jen J. Gong
Massachusetts Institute of
Technology
jengong@mit.edu

Thoralf M. Sundt
Massachusetts General
Hospital, Division of Cardiac
Surgery
tsundt@partners.org

James D. Rawn
Brigham and Women's
Hospital, Division of Cardiac
Surgery
jrawn@partners.org

John V. Guttag
Massachusetts Institute of
Technology
guttag@mit.edu

ABSTRACT

Accurate risk models for adverse outcomes can provide important input to clinical decision-making. Surprisingly, one of the main challenges when using machine learning to build clinically useful risk models is the small amount of data available. Risk models need to be developed for specific patient populations, specific institutions, specific procedures, and specific outcomes. With each exclusion criterion, the amount of relevant training data decreases, until there is often an insufficient amount to learn an accurate model. This difficulty is compounded by the large class imbalance that is often present in medical applications.

In this paper, we present an approach to address the problem of small data using *transfer learning* methods in the context of developing risk models for cardiac surgeries. We explore ways to build surgery-specific and hospital-specific models (the *target task*) using information from other kinds of surgeries and other hospitals (*source tasks*). We propose a novel method to weight examples based on their similarity to the target task training examples to take advantage of the useful examples while discounting less relevant ones.

We show that incorporating appropriate source data in training can lead to improved performance over using only target task training data, and that our method of instance weighting can lead to further improvements. Applied to a surgical risk stratification task, our method, which used data from two institutions, performed comparably to the risk model published by the Society for Thoracic Surgeons, which was developed and tested on over one hundred thousand surgeries from hundreds of institutions.

Categories and Subject Descriptors

I.2.1 [Artificial Intelligence]: Applications and Expert Systems—*Medicine and Science*

Keywords

Risk stratification models; transfer learning; instance weighting

1. INTRODUCTION

In this work, we focus on developing hospital- and procedure-specific risk stratification models for adverse events after cardiac surgery. Risk stratification models are used to determine a patient's category of risk, such as high-risk or low-risk. Although this limits the data that can be used, previous work in the field of cardiac surgery, and in the context of other clinical questions, has shown that models that account for institution- and procedure-specific characteristics can have improved performance over those that do not [26, 13, 19, 11]. Even at the largest of medical centers, the amount of data available for a specific surgery is small relative to the number of potentially relevant features.

We hypothesize that for a hospital- and procedure-specific task, there is still information to be gained from the thousands of other cardiac surgeries, at the same institution and at others. To tackle the problem of small data, we make use of these auxiliary data in a *transfer learning* framework to improve performance on our task of interest. In the transfer learning framework, the task of interest is referred to as the *target task*, while the related tasks are called *source tasks* [18]. The data pertaining to the target task are said to lie in the target domain, and the source data to lie in the source domain.

In this paper, we use an instance-transfer method, where examples from the source data are used to augment examples from the target training data to improve the model. We use instance weighting, with the goal of making the most of the available source data while discounting less relevant instances. Instance weighting seeks to shift the source data distribution towards the target data distribution [18]. One way to compute the weights is to directly estimate the source and target data probability distributions. However, because

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
KDD '15, August 10–13, 2015, Sydney, NSW, Australia.
© 2015 ACM. ISBN 978-1-4503-3664-2/15/08 ...\$15.00.
DOI: <http://dx.doi.org/10.1145/2783258.2783397>.

our target tasks are specified by procedure and hospital, the amount of relevant target data for each task is very small (e.g., 191 isolated mitral valve replacements at one of the hospitals in our study). In addition, the large class imbalance means that there are few adverse outcomes to learn from (e.g., 9 deaths among the 191 patients). Therefore, accurately estimating the target data distribution from the empirical data is infeasible.

Instead, we propose two instance-weighting approaches based on characteristics of the target data. Our work is related to the Kernel Mean Matching (KMM) method proposed in [10] and the Kullback-Liebler Importance Estimation Procedure (KLIEP) proposed in [25]. We discuss these methods more in Section 2.

We use data collected for the Society for Thoracic Surgeons (STS) National Adult Cardiac Surgery Database from two large hospitals (Hospital 1 and Hospital 2) in the United States over the years 2001 to 2011. We evaluate our method on two binary adverse outcomes: 1) *operative mortality* (+1 if death occurred during the hospitalization or within 30 days after discharge, -1 otherwise) and 2) *prolonged length of stay (LOS)* (+1 if > 14 days post-operative stay in the hospital, -1 otherwise). We formulate each task as a binary classification problem, where predicted labels describe the patient as high-risk or low-risk.

In our application, the target task is risk-stratifying patients receiving a particular surgical procedure (e.g. isolated aortic valve replacements) at a specific hospital. The source data are *other* cardiac surgeries from the *same* hospital and all of the cardiac surgeries from the other hospital. We consider 12 target tasks, determined by three factors: 1) whether the surgery occurred at Hospital 1 or Hospital 2, 2) whether it was one of three types of surgeries: isolated aortic valve replacement (AVR), isolated mitral valve replacement (MVR), and isolated mitral valve repair (MVRepair), and 3) whether it was for the outcome of operative mortality or prolonged LOS.

The STS has developed and published several widely used risk algorithms. These algorithms cover seven different procedure types (listed in Table 1). They include isolated coronary artery bypass grafting procedures (CABG), isolated aortic and mitral valve procedures, and CABG plus mitral or CABG plus aortic valve procedures [22, 17, 23]. These models do not cover some of the more complex surgeries. For example, in our data, over 30% of the surgeries and over 50% of the cases of operative mortality and prolonged LOS are not covered by an STS risk model (Table 1). Each of these complex surgeries is relatively rare, and therefore there are limited data available for building such models. Additionally, because the STS models are built using a large multi-institutional data set, they are not optimized for specific institutions.

In Section 2, we describe related work in this field. We then describe the methods we use in this paper in Section 3. In Section 4, we describe the setup of our experiments. Finally, in Section 5, we compare our methods with existing baselines on data from the STS database. Our results show that our methods can result in improved risk stratification performance relative to methods using only task-specific data, and that our method outperforms the KLIEP weighting scheme.

2. RELATED WORK

In traditional machine learning methods, one assumes that the training and test data lie in the same feature space and are drawn from the same distribution. In the transfer learning framework, these assumptions are violated. The data for the task of interest (*target data*) and data from another similar task (*source data*) come from different distributions. However, these distributions are assumed to be strongly related. When this assumption does not hold, it can result in worse performance than not using the source data (*negative transfer*). In [21], Rosenstein *et al.* showed that this negative effect is particularly noticeable when the target data are small. How to correctly ascertain which data are likely to be useful for the target task is still an open question in the field of transfer learning.

Previous work on using transfer learning to improve performance of clinical risk stratification models has focused on feature-representation transfer [26] and parameter-transfer [13]. In [26], the authors show that including hospital-specific features can improve performance on the task of interest. They also include data from related tasks in training, but they do not consider weighting the auxiliary examples based on the similarity to the target task. In [13], the authors show that adapting a global model to a local institution can lead to improved performance over simply using the global model.

In contrast, our work focuses on an instance-weighting approach. Feature-representation approaches are less relevant in our application because all of our data come from the STS database and lie in the same feature space. Parameter-transfer approaches require training a separate model using only the target data. In our application, however, we do not always have enough target data to do this; for the task of risk-stratifying patients receiving isolated mitral valve repairs (MV Repair) for the outcome of operative mortality, for example, there are only two adverse events in the data.

Previous work using instance-transfer has incorporated related data in 1) the *inductive* setting, and 2) the *transductive* setting [18]. In the first case, methods make use of available labeled target examples [3, 12, 14, 27]. In the latter case, there are either no labeled target examples, or very few available for training. One common framework in the transductive setting is that of *covariate shift*, where $p_{\text{source}}(x, y)$ and $p_{\text{target}}(x, y)$ are assumed to differ only in the marginal and not in the conditional [24, 31, 10, 25, 8, 28, 29]. These methods use unsupervised approaches to weight the source data such that the marginal distribution, $p_{\text{weighted-source}}(x)$, is closer to the target domain marginal distribution, $p_{\text{target}}(x)$. Because many of the target tasks we consider have a small number of adverse outcomes, we follow the covariate shift framework.

The covariate shift framework is still relevant in a supervised classification setting, where one is interested in the conditional distribution $p(y|x)$, as shown in [24]. For each target task, we consider only source tasks with the same outcome as the target task. In our dataset, the target and source data differ either in the procedure performed, the hospital the procedure was performed at, or both. The distributions of the covariates in these groups will differ. For example, patients receiving different surgeries might have different age distributions or exhibit different risk factors, and patients receiving surgeries at different hospitals might demonstrate demographic differences.

Table 1: Available data broken down by hospital and procedure. Numbers of surgeries and adverse outcomes associated with each surgery are displayed. Percentages are taken with respect to the total of each column; they show what percentage of the total number of surgeries or adverse outcomes at each hospital each surgical procedure consists of. Named procedures are covered by the STS risk models. Over 30% of the surgeries at each hospital and over half of the cases of operative mortality and prolonged LOS are not covered by an existing STS risk model (Other). Highlighted rows identify the tasks we evaluate our methods on.

Surgical Procedure	Hospital 1			Hospital 2		
	N	Operative Mortality	Prolonged LOS	N	Operative Mortality	Prolonged LOS
Isolated CABG	4,002 (39.8%)	58 (16.6%)	252 (20.5%)	4,044 (33.1%)	75 (15.6%)	304 (17.5%)
Isolated AVR	917 (9.1%)	19 (5.4%)	68 (5.5%)	1,602 (13.1%)	45 (9.4%)	152 (8.7%)
Isolated MVR	191 (1.9%)	9 (2.6%)	41 (3.3%)	240 (2.0%)	13 (2.7%)	40 (2.3%)
Isolated MV Repair	314 (3.1%)	2 (0.6%)	9 (0.7%)	877 (7.2%)	13 (2.7%)	49 (2.8%)
AVR + CABG	840 (8.4%)	35 (10.0%)	115 (9.4%)	958 (7.8%)	44 (9.1%)	173 (9.9%)
MVR + CABG	79 (0.8%)	7 (2.0%)	21 (1.7%)	99 (0.8%)	5 (1.0%)	31 (1.8%)
MV Repair + CABG	149 (1.5%)	8 (2.3%)	22 (1.8%)	393 (3.2%)	16 (3.3%)	77 (4.4%)
Other	3,565 (35.4%)	211 (60.5%)	701 (57.0%)	4,006 (32.8%)	270 (56.1%)	915 (52.6%)
Total	10,057	349	1,229	12,219	481	1,741

The small number of target task examples precludes us from using methods that require estimating the probability densities of the target and source data. Instead, we computed the instance weights without estimating the probability distributions. This is similar to instance-weighting methods such as Kernel Mean Matching, proposed in [10], and the Kullback-Liebler Importance Estimation Procedure, proposed in [25]. Unlike these methods, and the methods proposed in [28] and [29], our method of instance weighting does not require solving an additional optimization problem to compute the weights.

Kernel Mean Matching (KMM) [10] computes the weights for each of the training examples (to move the weighted source data distribution closer to the target data distribution) by minimizing the distance between the mean of the target data and the mean of the source data in a reproducing kernel Hilbert space. The authors demonstrate that weighting instances using the KMM method improves performance over the unweighted approach in several applications [10]. However, KMM requires choosing an appropriate kernel width, σ , and previous work has demonstrated that the performance of KMM is sensitive to the choice of this hyper-parameter [25, 8]. In addition, KMM does not incorporate a model selection procedure within the importance estimation algorithm.

The Kullback-Liebler Importance Estimation Procedure (KLIEP) [25] uses a chosen set of basis functions and the data to determine the best weights. In contrast to the KMM method, KLIEP incorporates a model selection procedure. However, the authors note that this cross-validation procedure may not work well in the case of small sample sizes. The cross-validation is done over the test input samples in [25], and in their application, small data is not an issue. However, in our work we use the target training data to determine the weights, and the number of examples is small relative to the dimensionality of the data (≈ 250 features). This can be seen from Table 1. Because of this, we seek to reduce the number of hyper-parameters that must be cross-validated for in our methods.

3. METHODS

In this section, we present two instance-weighting methods: 1) *Weighted Instances*, and 2) *Weighted Clusters*. The methods we propose use the squared Euclidean distance between each training example (source *and* target) and the mean of the target training examples to compute the instance weights. This distance is measured using the features provided in the STS database. We calculated these weights after discretizing and binarizing continuous features and eliminating features (described in detail in Section 4). Because the features were binary, the squared Euclidean distance was equivalent to the Hamming distance.

3.1 Weighted Instances

Let S denote the source data and T denote the target training data for a given task. Let $\mathbf{x}_i \in \{0, 1\}^m$ denote the feature vector of example i , $\forall i \in S \cup T$. Let d_i be the

Figure 1: Diagram of the Weighted Instances scheme. Orange circles denote source examples, while blue triangles denote target examples. The instance weight for each example (target and source) is computed by dividing the distance of the farthest target example from the target centroid by the distance of the example in question.

distance of $\mathbf{x}_i$ from the mean of the target training data μ_T , and $\max_{i \in T}(d_i)$ be the distance of the farthest target example from the mean. In the *Weighted Instances* method, the weight I_i for instance i is then

$$I_i = \frac{\max_{i \in T}(d_i)}{d_i}, \forall i \in S \cup T. \quad (1)$$

This scheme is diagrammed in Figure 1 and the procedure is shown in Algorithm 1.

We use this weighting scheme so that examples that more closely resemble the target data (i.e., examples that lie closer to the target centroid) are given a higher weight (> 1), while examples that lie further than the farthest target example are given a lower weight (< 1).

Algorithm 1 Weighted Instances Algorithm.

Input: Feature vectors $\mathbf{x}_i \in \{0, 1\}^m, \forall i \in S \cup T$, where S is the source data and T is the target training data.

Output: Instance weights $I_i, \forall i \in S \cup T$

- 1: Compute the mean of T : $\mu_T \leftarrow \frac{1}{|T|} \sum_{i \in T} \mathbf{x}_i$
 - 2: **for** $i \in S \cup T$ **do**
 - 3: $d_i \leftarrow \sum_{j=1}^m (x_i^j - \mu_T^j)^2$
 - 4: **end for**
 - 5: **for** $i \in S \cup T$ **do**
 - 6: $I_i \leftarrow \frac{\max_{i \in T}(d_i)}{d_i}$
 - 7: **end for**
-

3.2 Weighted Clusters

We also investigated another approach, which we call *Weighted Clusters*. This method falls under the framework of multiple source transfer learning. Methods in this area seek to take maximum advantage of data from multiple source domains to improve performance on a target task [6, 15, 30, 7]. The multiple source framework is of interest because assuming structure in the source data can make the instance-weighting scheme more robust. That is, the weight of each individual instance is now moderated by the location of its neighbors. If the assumed structure is accurate, this could lead to less overfitting when calculating the instance weights.

For the Weighted Clusters approach, we used K -means clustering to discover structure in the source data. Exam-

Algorithm 2 Weighted Clusters Algorithm.

Input: Feature vectors $\mathbf{x}_i \in \{0, 1\}^m, \forall i \in S \cup T$, where S is the source data and T is the target training data.

Output: Instance weights $I_i, \forall i \in S \cup T$

- 1: Compute the mean of T : $\mu_T \leftarrow \frac{1}{|T|} \sum_{i \in T} \mathbf{x}_i$
 - 2: Cluster S using K-means 5 times. Take the clustering assignment $C_i \in \{1, \dots, K\}$ for each instance $i \in S$ with the minimum sum of within-cluster variance (dissimilarity).
 - 3: **for** $i \in T$ **do**
 - 4: $d_i \leftarrow \sum_{j=1}^m (x_i^j - \mu_T^j)^2$
 - 5: **end for**
 - 6: **for** $i \in S$ **do**
 - 7: $d_i \leftarrow \sum_{j=1}^m (\mu_{C_i}^j - \mu_T^j)^2$
 - 8: **end for**
 - 9: **for** $i \in S \cup T$ **do**
 - 10: $I_i \leftarrow \frac{\max_{i \in T}(d_i)}{d_i}$
 - 11: **end for**
-

ples from each cluster were assigned weights according to the distance of the cluster's mean to the mean of the target training data. We evaluated this approach for $K = 1, 2, 3$. These values allow us to explore structure in the source data without assuming too specific of a structure (large K). For each value of K , we repeated the clustering five times and selected the clustering with the minimum within-cluster dissimilarity. This procedure is detailed in Algorithm 2. Just as in the Weighted Instances approach, target examples were assigned weights based on their distance to the target mean. This allows the weights of the target examples to account for possible outliers.

4. EXPERIMENTAL SETUP

Because we are interested in developing hospital- and procedure-specific models, we evaluated our method on several specific groups of surgeries at each of the hospitals. We ran 12 experiments, each testing our method on one clinical outcome at a single hospital for a given procedure. We evaluated our method on three procedures for which there are established risk models: isolated aortic valve replacements (AVR), isolated mitral valve replacements (MVR), and isolated mitral valve repairs (MV Repairs). These procedures are highlighted in Table 1. We considered two outcomes, operative mortality and prolonged LOS. Figure 2 shows the relationship between the target data and source data for the task of risk-stratifying isolated AVRs at Hospital 1 for either outcome of interest. The source data in this case include all of the other data from Hospital 1 and all of the data from Hospital 2.

We compared our approach of using source data to augment the target task training data with instance weighting, to five baselines: 1) *Target-only*, where only target task data are used in training; 2) *Source-only*, where only source task data are used in training; 3) *Target + Source (Unweighted)*, where source and target data are used in training without weights; 4) the STS model; and 5) the KLIEP weighting method proposed in [25].

Figure 2: Diagram of the relationship between the source (orange) and target (blue) data, for the target task of risk-stratifying patients receiving an isolated AVR at Hospital 1. Source data include the isolated AVRs from Hospital 2 and the other cardiac surgeries from both hospitals.

4.1 Data and Preprocessing

Our data come from two hospitals and were prepared for the Society for Thoracic Surgeons (STS) Adult Cardiac Surgery database. The data were collected over the years 2001-2011. The STS database went through several data version changes through this period. We addressed this by manually consolidating variables with different names but the same meanings, and eliminating inconsistent variables. Table 1 shows the size of the hospitals and the occurrence of some of the common adverse outcomes, broken down by procedure. We considered only patients on full cardiopulmonary bypass.

The STS risk models [22, 17, 23] were built using interaction features and expert input. In contrast, we used the raw STS features. We binarized categorical features and discretized and binarized continuous features according to quintile cutoffs. We ended up with approximately 250 features.

4.2 Feature selection

We sought to minimize overfitting by using a dimensionality reduction technique. We did this using the Conditional Mutual Information Maximization (CMIM) filtering method [5]. This iterative feature selection method greedily adds features based on how much information is contributed beyond the contribution of already selected features. We used different thresholds in the CMIM algorithm at which to stop incorporating new features, depending upon the amount of available training data. We used a threshold of 0.001 for methods that only used target training data, and a threshold of 0.0001 for methods that added source data to training. The algorithm stops adding features when the conditional mutual information added by the best new feature falls below the threshold.

We experimented with the use of L1-regularized logistic regression (without CMIM) for the Target-only and Target + Source (Unweighted) baselines. The results were generally worse than the L2-regularized method, or comparable, and are not presented here. L1-regularization also results in a less interpretable model than L2-regularization, even with the use of CMIM for feature selection. While L1-regularization encourages a sparser solution, when there are highly correlated variables, the model weights are less interpretable than those from L2-regularized logistic regression. Although only one of several highly correlated features might be needed to achieve good predictive performance, driving the weights of other features to zero can present a deceptive picture to those interested in investigating ways to change outcomes as well as to predict them. In contrast, although CMIM is a greedy dimensionality reduction method, the scores for all of the features can be used to determine which features in each iteration provided the most information.

4.3 Model Development

In all of our experiments, we used L2-regularized logistic regression to predict a patient’s risk for a given adverse outcome. We used L2-regularization to prevent overfitting. The equation for L2-regularized logistic regression is as follows:

$$\min_{\mathbf{w}} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^n \log \left(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i} \right). \quad (2)$$

Because of the high class imbalance in our data, we used an asymmetric cost parameter so that misclassification of an adverse event penalized the objective function much more than misclassification of a non-adverse event. In effect, this means assigning a different cost parameter C to the non-adverse versus adverse events.

$$\min_{\mathbf{w}} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C_+ \sum_{i: y_i = +1} \log \left(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i} \right) + C_- \sum_{i: y_i = -1} \log \left(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i} \right) \quad (3)$$

We used 5-fold cross-validation to select the best value for C_- . We searched for the value in the range 10^{-7} to 10^2 in powers of 10. Because of the small amounts of data, we set the asymmetric cost parameter ($\frac{C_+}{C_-}$) to the class imbalance (i.e., the ratio of the number of non-adverse events to the number of adverse events).

All models were trained using the LIBLINEAR package [4], and we incorporated our instance-weights using LIBLINEAR’s built-in instance-weighting functionality.

4.4 Evaluation

We used leave-one-out cross-validation (LOOCV) to evaluate our method. In each iteration, one example from the target data was used for testing, and all of the others, plus any available source data, were used for training. We used LOOCV because it maximizes the amount of target training data available, and it is the most generalizable approach across different target tasks. The number of adverse outcomes in the target task data can range from the single digits to the hundreds (e.g., 2 deaths and 9 cases of prolonged LOS out of 314 isolated MVR at Hospital 1, to 152 cases of prolonged LOS out of 1,602 isolated AVR at Hospital 2).

We assessed performance using the overall precision and recall. These metrics were calculated by setting the threshold for high-risk at the 80th percentile. Using the upper quintile of risk as a cutoff for high-risk is a common practice in the medical literature [1, 2, 20]. We also computed the estimated area under the receiver operating characteristic curve (AUC) [9].

We used McNemar’s test, a statistical test that is applied to two by two contingency tables [16], to evaluate the statistical significance of our results.

5. RESULTS

In this section, we first demonstrate that Target + Source (Unweighted) can perform better than the Target-only and Source-only approaches. These results are presented in Tables 2 and 3. We then show that the weighting schemes have the potential to improve performance beyond the Target + Source (Unweighted) approach. Finally, we present statistical significance results comparing the weighting methods with the unweighted method. These results are presented in Tables 4 and 5.

5.1 Adding Source to Target

The results for each experiment in Tables 2 and 3 show that making use of source data can often increase the performance over the Target-only approach. In some cases, making use of source data is necessary to train a model that performs better than a random classifier for a given task.

Table 2: Operative Mortality: Comparing Target + Source (Unweighted) to Random, Target-only, Source-only, and STS baselines. Estimated precision, recall, and AUC are shown for Isolated AVR, Isolated MVR, and Isolated MV Repair procedures at each hospital. The best precision, recall, and AUC are in bold. Random precision (incidence in the population), recall (0.20), and AUC (0.50) are shown in the top row for each experiment. The Target-only performance for Isolated MV Repair at Hospital 1 is missing because there were not enough positive examples to develop a model (only two adverse events). Target + Source (Unweighted) performance is better than Target-only performance on all tasks, and it is comparable to STS performance for all six tasks (p-values from McNemar’s test > 0.05).

Procedure	Method	Hospital 1			Hospital 2		
		Precision	Recall	AUC	Precision	Recall	AUC
Isolated AVR	Random	0.0207	0.20	0.50	0.0281	0.20	0.50
	Target-only	0.0435	0.2632	0.6043	0.0469	0.3333	0.6234
	Source-only	0.0383	0.3684	0.6718	0.0719	0.5111	0.7456
	Target + Source (Unweighted)	0.0492	0.4737	0.6993	0.0781	0.5556	0.7525
	STS	0.0383	0.3684	0.7230	0.0781	0.5556	0.7528
Isolated MVR	Random	0.0471	0.20	0.50	0.0542	0.20	0.50
	Target-only	0.0263	0.1111	0.4029	0.1458	0.5385	0.8489
	Source-only	0.1053	0.4444	0.6636	0.1667	0.6154	0.8580
	Target + Source (Unweighted)	0.1053	0.4444	0.6575	0.1667	0.6154	0.8553
	STS	0.1053	0.4444	0.6770	0.2083	0.7692	0.8919
Isolated MV Repair	Random	0.0064	0.20	0.50	0.0148	0.20	0.50
	Target-only	–	–	–	0.0514	0.6923	0.8347
	Source-only	0.0317	1	0.9615	0.0629	0.8462	0.8884
	Target + Source (Unweighted)	0.0317	1	0.9599	0.0686	0.9231	0.9077
	STS	0.0317	1	0.9840	0.0629	0.8462	0.9221

Table 3: Prolonged Length of Stay (LOS > 14 days): Comparing Target + Source (Unweighted) to Random, Target-only, Source-only, and STS baselines. Estimated precision, recall, and AUC are shown for Isolated AVR, Isolated MVR, and Isolated MV Repair procedures at each hospital. The best precision, recall, and AUC are in bold. Random precision (incidence in the population), recall (0.20), and AUC (0.50) are shown in the top row for each experiment. Target + Source (Unweighted) performance is better than Target-only performance on most tasks, and it is comparable to STS performance for all six tasks (p-values from McNemar’s test > 0.05).

Procedure	Method	Hospital 1			Hospital 2		
		Precision	Recall	AUC	Precision	Recall	AUC
Isolated AVR	Random	0.0743	0.20	0.50	0.0949	0.20	0.50
	Target-only	0.2022	0.5441	0.7612	0.2000	0.4211	0.7154
	Source-only	0.2186	0.5882	0.7661	0.2344	0.4934	0.7559
	Target + Source (Unweighted)	0.2404	0.6471	0.7790	0.2375	0.5000	0.7591
	STS	0.2186	0.5882	0.7827	0.2313	0.4868	0.7537
Isolated MVR	Random	0.2158	0.20	0.50	0.1674	0.20	0.50
	Target-only	1	0.1220	0.2840	0.4375	0.5250	0.7327
	Source-only	0.5526	0.5122	0.7401	0.3958	0.4750	0.7602
	Target + Source (Unweighted)	0.5789	0.5366	0.7324	0.4167	0.5000	0.7666
	STS	0.6316	0.5854	0.7874	0.3750	0.4500	0.7534
Isolated MV Repair	Random	0.0287	0.20	0.50	0.0559	0.20	0.50
	Target-only	0.1270	0.8889	0.8189	0.1371	0.4898	0.7531
	Source-only	0.1111	0.7778	0.8623	0.1486	0.5306	0.7851
	Target + Source (Unweighted)	0.0952	0.6667	0.8656	0.1486	0.5306	0.7787
	STS	0.0952	0.6667	0.8477	0.1486	0.5306	0.8020

The performance of a random classifier for each of the tasks is shown in the top row for each experiment. Precision of a random classifier is the incidence of the outcome. By design, the recall of our random classifier is 0.2 (since we consider the upper quintile to be high-risk), and the AUC of a random classifier is 0.5. Comparing the Target-only performances to these values, we can see that in some cases, the Target-only performance falls below random. For example, in the case of operative mortality for isolated MVR at Hos-

pital 1, the precision of the Target-only classifier is 0.0263, below the random precision of 0.0471 (Table 2). Similarly, the recall falls below 0.2 and the AUC falls below 0.5.

In 9 out of the 12 cases, the Source-only approach outperformed the Target-only approach. It might seem surprising that the Source-only approach does not always outperform the Target-only approach, given that there is so much more source data available. However, this demonstrates that the

Table 4: Operative Mortality: Comparing performance of weighted methods versus Target + Source (Unweighted). Estimated precision, recall, and AUC are shown for Isolated AVR, Isolated MVR, and Isolated MV Repair procedures at each hospital. The best precision, recall, and AUC are in bold. McNemar’s test p-values for differences between weighted methods and Target + Source (Unweighted) are shown for each task.

Procedure	Method	Hospital 1				Hospital 2			
		Precision	Recall	AUC	McNemar’s	Precision	Recall	AUC	McNemar’s
Isolated AVR	Target + Source (Unweighted)	0.0492	0.4737	0.6993	–	0.0781	0.5556	0.7525	–
	KLIEP-weighted	0.0383	0.3684	0.6630	0.70	0.0813	0.5778	0.7362	0.88
	Weighted Instances	0.0437	0.4211	0.6784	0.82	0.0750	0.5333	0.7529	0.76
	Weighted Clusters, K = 1	0.0492	0.4737	0.6915	1	0.0781	0.5556	0.7554	1
	Weighted Clusters, K = 2	0.0437	0.4211	0.7208	0.84	0.0781	0.5556	0.7719	1
	Weighted Clusters, K = 3	0.0546	0.5263	0.7036	0.73	0.0813	0.5778	0.7736	0.79
Isolated MVR	Target + Source (Unweighted)	0.1053	0.4444	0.6575	–	0.1667	0.6154	0.8553	–
	KLIEP-weighted	0.0789	0.3333	0.5965	0.81	0.1875	0.6923	0.8234	0.72
	Weighted Instances	0.1053	0.4444	0.6502	1	0.1667	0.6154	0.8665	1
	Weighted Clusters, K = 1	0.1053	0.4444	0.6703	1	0.1667	0.6154	0.8523	1
	Weighted Clusters, K = 2	0.1053	0.4444	0.6471	1	0.2083	0.7692	0.8695	0.29
	Weighted Clusters, K = 3	0.1053	0.4444	0.6429	1	0.1667	0.6154	0.8506	1
Isolated MV Repair	Target + Source (Unweighted)	0.0317	1	0.9599	–	0.0686	0.9231	0.9077	–
	KLIEP-weighted	0.0159	0.5000	0.8878	0.76	0.0514	0.6923	0.8228	0.61
	Weighted Instances	0.0317	1	0.9535	1	0.0629	0.8462	0.8813	0.84
	Weighted Clusters, K = 1	0.0317	1	0.9663	1	0.0629	0.8462	0.8998	0.82
	Weighted Clusters, K = 2	0.0317	1	0.9631	1	0.0629	0.8462	0.9086	0.83
	Weighted Clusters, K = 3	0.0317	1	0.9583	1	0.0629	0.8462	0.8930	0.84

relevance of the data to the task of interest is an important factor.

Notably, using the available source data in addition to the target training data can lead to performances comparable to that of the STS isolated valve model, which makes use of data from hundreds of institutions and has over one hundred thousand surgeries available for training data [17]. The Target + Source (Unweighted) approach was not significantly different from the STS model in any of the experiments we did for either test, despite the fact that we had far less training data available (results not shown here).

5.2 Instance Weighting

In the previous section, we established that adding source data to target training data can improve performance on our tasks of interest. In this section, we investigate the added benefit of weighting instances using our two proposed schemes, Weighted Instances and Weighted Clusters. In addition to comparing against the Target + Source (Unweighted) baseline, we compared our methods to the KLIEP instance-weighting method. We used the code provided by the authors of [25]. We did not directly compare our method to the Kernel Mean Matching approach, because [25] demonstrated that the performance of KLIEP was comparable to that of KMM. In addition, KLIEP incorporates a parameter selection process, whereas KMM does not.

From Tables 4 and 5, we see that in 10 of the 12 experiments, either Weighted Instances or Weighted Clusters performed at least as well as the Target + Source (Unweighted) and KLIEP baselines in precision and recall.

Figure 3 compares the results from the Target + Source (Unweighted) approach to each of the weighting schemes we investigated: 1) Weighted Instances, 2) Weighted Clusters, and 3) KLIEP-weighted [25]. The percentage improvement in recall over the unweighted method is shown. The plot for precision is omitted because it shows similar results. The figure shows that our weighting scheme can improve performance over the unweighted method. On the other hand,

the KLIEP weighting method always results in worse performance on the tasks at Hospital 1. At Hospital 2, it results in worse performance than our methods in 5 of the 6 tasks. We computed the statistical significance of these results using McNemar’s test. These results are shown in Tables 4 and 5.

The results from the KLIEP weighting method are significantly worse in two cases (prolonged LOS at Hospital 1 for isolated AVRs and isolated MVRs) at a significance level of 0.05. The weighting schemes we propose lead to improved performance compared to using all of the data without weights, but these results are not significant.

From Figure 3, the improvements of our weighting schemes over the unweighted method appear more consistent across the tasks at Hospital 2 compared to the tasks at Hospital 1. At both Hospital 1 and Hospital 2, our weighting schemes clearly outperform the KLIEP weighting method. At Hospital 1, our proposed weighting schemes led to improved performance over Target + Source (Unweighted) for three tasks: operative mortality for Isolated AVR and post-operative LOS for both Isolated MV Repair and Isolated MVR. At Hospital 2, our proposed weighting schemes led to improved performance over Target + Source (Unweighted) for 5 of the 6 tasks.

We also compared the weighting scheme that assumed some structure about the source data (Weighted Clusters) with the method that did not (Weighted Instances). In 8 out of the 12 cases, this clustering scheme improved performance over the Weighted Instances method. The majority of these cases were for tasks at Hospital 2. In 5 out of the 6 tasks from Hospital 2, clustering the source data before computing the instance-weights improved performance over the Weighted Instances method. One possible reason for the improved performance in these cases is that Hospital 2 has a larger cardiac surgical volume (and therefore more of the source data surgeries are from Hospital 2).

There are two main reasons why the clustering approach helped in these cases. First, it allows the weighting scheme

Table 5: Prolonged Length of Stay (LOS > 14 days): Comparing performance of weighted methods versus Target + Source (Unweighted). Estimated precision, recall, and AUC are shown for Isolated AVR, Isolated MVR, and Isolated MV Repair procedures at each hospital. The best precision, recall, and AUC are in bold.

Procedure	Method	Hospital 1				Hospital 2			
		Precision	Recall	AUC	McNemar's	Precision	Recall	AUC	McNemar's
Isolated AVR	Target + Source (Unweighted)	0.2404	0.6471	0.7790	—	0.2375	0.5000	0.7591	—
	KLIEP-weighted	0.1803	0.4853	0.7533	0.05*	0.2125	0.4474	0.7296	0.26
	Weighted Instances	0.2295	0.6176	0.7792	0.49	0.2469	0.5197	0.7614	0.33
	Weighted Clusters, K = 1	0.2350	0.6324	0.7741	0.84	0.2344	0.4934	0.7631	0.73
	Weighted Clusters, K = 2	0.2131	0.5735	0.7725	0.09	0.2500	0.5263	0.7659	0.26
	Weighted Clusters, K = 3	0.2131	0.5735	0.7770	0.06	0.2500	0.5263	0.7603	0.31
Isolated MVR	Target + Source (Unweighted)	0.5789	0.5366	0.7324	—	0.4167	0.5000	0.7666	—
	KLIEP-weighted	0.4474	0.4146	0.6790	0.05*	0.3333	0.4000	0.6938	0.17
	Weighted Instances	0.5789	0.5366	0.7386	1	0.3958	0.4750	0.7612	0.69
	Weighted Clusters, K = 1	0.5789	0.5366	0.7368	1	0.3958	0.4750	0.7722	0.77
	Weighted Clusters, K = 2	0.6053	0.5610	0.7440	0.63	0.3958	0.4750	0.7686	0.75
	Weighted Clusters, K = 3	0.5526	0.5122	0.7394	0.50	0.4375	0.5250	0.7730	0.73
Isolated MV Repair	Target + Source (Unweighted)	0.0952	0.6667	0.8656	—	0.1486	0.5306	0.7787	—
	KLIEP-weighted	0.0476	0.3333	0.6444	0.41	0.1314	0.4694	0.7329	0.51
	Weighted Instances	0.1111	0.7778	0.8372	0.80	0.1429	0.5102	0.7836	0.75
	Weighted Clusters, K = 1	0.0952	0.6667	0.8627	1	0.1429	0.5102	0.7762	0.77
	Weighted Clusters, K = 2	0.1111	0.7778	0.8893	0.83	0.1486	0.5306	0.7810	1
	Weighted Clusters, K = 3	0.0952	0.6667	0.8707	1	0.1543	0.5510	0.7760	0.78

Figure 3: Comparison of performance when target and source data are unweighted versus when different weighting schemes are used at Hospital 1 (left) and Hospital 2 (right). Percentage improvement in recall over the unweighted approach is shown. Operative mortality is referred to by its variable short name ‘mtopd,’ and prolonged LOS is referred to as ‘plos.’

to leverage structure in the data. Second, it reduces overfitting in the sense that each example is not assigned a weight based on its own distance from the target training data mean. Instead, its influence on its own weight is moderated by nearby examples. This prevents the instance-weighting scheme from unduly giving any example too much or too little weight.

However, assuming a structure as we did in the Weighted Clusters methods can also contribute to overfitting. For example, there may be no underlying structure in the data that can be accurately captured using K-means, and we

have forced a framework that led to less informative weights. These factors have a larger impact when the data are smaller (e.g., at Hospital 1).

Finally, we compared the results from the weighting schemes with the STS model using McNemar’s Test. For the 12 tasks we considered, the performance of our proposed weighting methods was not significantly different from the STS performance. However, the KLIEP performance was significantly worse for post-operative LOS for isolated MVR at Hospital 1 at a significance level of 0.01 (p-value = 0.006).

6. DISCUSSION & CONCLUSION

In this paper, we proposed a simple instance-weighting scheme based on the mean of the target training data and the distance of the farthest target example from the mean. We used this method to help address the problem of small data with high dimensionality in developing institution-specific, procedure-specific risk models for cardiac surgery. We compared our method against the Target-only, Source-only, and the Target + Source (Unweighted) baselines. We also compared our method against another instance-weighting approach from the literature (KLIEP) and against an established risk model in the cardiac surgery literature developed by the Society for Thoracic Surgeons.

Our results show that making appropriate use of source data in conjunction with target training data can improve performance over using only target data. In addition, using our instance-weighting methods can result in improvements relative to unweighted instance transfer. We also demonstrated that our instance-weighting methods outperforms the KLIEP instance-weighting method proposed in [25] in most of the tasks we considered. Lastly, our weighting scheme led to results comparable to those of the STS Isolated Valve model, which was developed and tested on over one hundred thousand surgeries from hundreds of institutions [17].

We showed that an instance-weighting approach that assumes structure about the data can result in improved performance over a method that does not make any assumptions. This method improved performance more for the target tasks at Hospital 2, the hospital with the larger cardiac surgical volume.

Although the instance-weighting methods we investigated sometimes performed better than the baseline of using all of the available data without instance weighting, these differences were not significant. In future work, we hope to test these methods on larger multi-center datasets. We also plan to further investigate how robust the instance-weighting methods are, and how this depends on the amount of available target data.

Our results indicate that our methods work even when there are few target examples available. Because our methods leverage the available data even when they are not directly related to the task of interest, they can be used to construct models for surgeries that currently do not have established risk models, providing a new source of information for clinical decision-makers.

7. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 1122374, the National Science Foundation under Grant No. IIS-1065079, and Quanta Computer Inc.

8. REFERENCES

- [1] S. G. Baker and D. J. Sargent. Designing a randomized clinical trial to evaluate personalized medicine: a new approach based on risk prediction. *Journal of the National Cancer Institute*, 2010.
- [2] R. B. D’Agostino, R. S. Vasan, M. J. Pencina, P. A. Wolf, M. Cobain, J. M. Massaro, and W. B. Kannel. General cardiovascular risk profile for use in primary care the framingham heart study. *Circulation*, 117(6):743–753, 2008.
- [3] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning*, pages 193–200. ACM, 2007.
- [4] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.
- [5] F. Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, 2004.
- [6] J. Gao, W. Fan, J. Jiang, and J. Han. Knowledge transfer via multiple model local structure mapping. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 283–291. ACM, 2008.
- [7] L. Ge, J. Gao, H. Ngo, K. Li, and A. Zhang. On handling negative transfer and imbalanced distributions in multiple source transfer learning. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 2014.
- [8] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schoelkopf. Dataset shift in machine learning. In J. Quiñero-Candela, M. Sugiyama, A. Schwaighofer, and N. Lawrence, editors, *Covariate Shift and Local Learning by Distribution Matching*, pages 131–160. MIT Press, 2008.
- [9] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.
- [10] J. Huang, A. Gretton, K. M. Borgwardt, B. Schölkopf, and A. J. Smola. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems*, pages 601–608, 2006.
- [11] J. Ivanov, J. V. Tu, and C. D. Naylor. Ready-made, recalibrated, or remodeled? issues in the use of risk indexes for assessing mortality after coronary artery bypass graft surgery. *Circulation*, 99(16):2098–2104, 1999.
- [12] J. Jiang and C. Zhai. Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 264–271. ACL, 2007.
- [13] G. Lee, I. Rubinfeld, and Z. Syed. Adapting surgical models to individual hospitals using transfer learning. In *2012 IEEE 12th International Conference on Data Mining, Workshop on Biological Data Mining and its Applications in Healthcare (BioDM)*, pages 57–63. IEEE, 2012.
- [14] X. Liao, Y. Xue, and L. Carin. Logistic regression with an auxiliary data source. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 505–512. ACM, 2005.
- [15] P. Luo, F. Zhuang, H. Xiong, Y. Xiong, and Q. He. Transfer learning from multiple source domains via consensus regularization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pages 103–112. ACM, 2008.

- [16] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.
- [17] S. M. O’Brien, D. M. Shahian, G. Filardo, V. A. Ferraris, C. K. Haan, J. B. Rich, S.-L. T. Normand, E. R. DeLong, C. M. Shewan, R. S. Dokholyan, et al. The Society of Thoracic Surgeons 2008 cardiac surgery risk models: part 2 – isolated valve surgery. *The Annals of Thoracic Surgery*, 88(1):S23–S42, 2009.
- [18] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.
- [19] O. Pitkänen, M. Niskanen, S. Rehnberg, M. Hippeläinen, and M. Hynynen. Intra-institutional prediction of outcome after cardiac surgery: comparison between a locally derived model and the EuroSCORE. *European Journal of Cardio-Thoracic Surgery*, 18(6):703–710, 2000.
- [20] M. Rahman, R. K. Simmons, A.-H. Harding, N. J. Wareham, and S. J. Griffin. A simple risk score identifies individuals at high risk of developing type 2 diabetes: a prospective cohort study. *Family Practice*, 25(3):191–196, 2008.
- [21] M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich. To transfer or not to transfer. In *Proceedings of NIPS 2005 Workshop on Inductive Transfer: 10 Years Later*, 2005.
- [22] D. M. Shahian, S. M. O’Brien, G. Filardo, V. A. Ferraris, C. K. Haan, J. B. Rich, S.-L. T. Normand, E. R. DeLong, C. M. Shewan, R. S. Dokholyan, et al. The Society of Thoracic Surgeons 2008 cardiac surgery risk models: part 1– coronary artery bypass grafting surgery. *The Annals of Thoracic Surgery*, 88(1):S2–S22, 2009.
- [23] D. M. Shahian, S. M. O’Brien, G. Filardo, V. A. Ferraris, C. K. Haan, J. B. Rich, S.-L. T. Normand, E. R. DeLong, C. M. Shewan, R. S. Dokholyan, et al. The Society of Thoracic Surgeons 2008 cardiac surgery risk models: part 3 – valve plus coronary artery bypass grafting surgery. *The Annals of Thoracic Surgery*, 88(1):S43–S62, 2009.
- [24] H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.
- [25] M. Sugiyama, S. Nakajima, H. Kashima, P. V. Buenau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems*, pages 1433–1440, 2008.
- [26] J. Wiens, J. Gutttag, and E. Horvitz. A study in transfer learning: leveraging data from multiple hospitals to enhance hospital-specific predictions. *Journal of the American Medical Informatics Association*, 0:1–8, 2014.
- [27] P. Wu and T. G. Dietterich. Improving SVM accuracy by training on auxiliary data sources. In *Proceedings of the 21st International Conference on Machine Learning*, pages 110–117. ACM, 2004.
- [28] R. Xia, X. Hu, J. Lu, J. Yang, and C. Zong. Instance selection and instance weighting for cross-domain sentiment classification via PU learning. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 2176–2182. AAAI Press, 2013.
- [29] R. Xia, J. Yu, F. Xu, and S. Wang. Instance-based domain adaptation in nlp via in-target-domain logistic approximation. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pages 1600–1606. AAAI, 2014.
- [30] Y. Yao and G. Doretto. Boosting for transfer learning with multiple sources. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1855–1862. IEEE, 2010.
- [31] B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21st International Conference on Machine Learning*, pages 114–121. ACM, 2004.