Reframing for Non-Linear Dataset Shift

Md Shadman Rafid, Mohammad Mazedul Islam,
Md Naimul Hoque, Chowdhury Farhan Ahmed

Department of CSE, University of Dhaka, Bangladesh
shadmanrafiddeep@gmail.com
mazidmailbox@gmail.com
naimul.et@easternuni.edu.bd
farhan@du.ac.bd

Abstract. Discriminative classification models assume that both training and deployment data have same distributions of data attributes. These models give significantly varied performances when they are deployed under varied circumstances with different data distributions. This phenomenon is called Dataset Shift. In this paper we have provided a method which first determines whether there is a significant shift in the distributions of attributes between the training and deployment datasets. If there exists a shift in the data the proposed method then uses a Hill climbing approach to map this shift irrespective of its nature i.e. (linear or non-linear) to the equation for quadratic transformation. Experimental results on three real life datasets show strong performance gains achieved by the proposed method over previously established methods such as retraining and linear reframing.

1 Introduction

The main concerns of supervised machine learning is to learn a model for classification, regression or any other function using a set of training data and then applying this new learned model to deployment data. While deploying a particular data model it is implicitly assumed that both the training and test data will follow the same distributions. But in real life scenarios it is natural for the distributions of data attributes and decision functions to change especially when the training data is collected in one context while the deployment data is used in a different context e.g.(the training data is collected the summer season while the model is deployed on the data for the season of autumn). Such "Dataset Shift" [Han et al. (2012)] if not compensated for can greatly reduce the efficiency of the results provided by the learned model. One solution is to retrain the entire model on the deployment data. But it is not a feasible option often as collection and labelling of data in deployment may become costly. Another recent approach is reframing the data so as to make the learned model compensate for the shift in deployment data in real time and provide efficient results. We mainly focus on the shifts of continuous attributes of the data from test to deployment dataset. For example the daily food consumption of the residents of a city in North America may vary greatly from that of the residents of a similar city in South America.

In order to deal with dataset shift several research works [Lachiche and Flach (2003)], [Charnay et al. (2013)], [Zhao et al. (2011)], [Hernandez-Orallo (2013)] have been published proposing several methods of using the learnt methods in different deployment environments by adjusting the output values. Reframing is a method of transforming the input values to the learnt model in real time during deployment and thus compensating for the shift.

Let us consider a real life scenario where we consider two cities A and B. It is seen that in City A most people buy a car when they are at least 35 years old and have a minimum income of $5000 while in City B most residents buy a car once they reach the age of 40 and have an income of at least $4500. Let us consider that we use the data collected from City A as the training data to create the model and deploy the created model on the data of City B to determine which residents of City B are likely to buy cars. As the data of the two cities do not follow the same distributions a simple rescaling of the age and income parameters of the data will allow the existing model to correctly classify the data of City B without requiring any retraining or adjustments of the output. This is a very simple example of reframing which is needed whenever the decision function in case of deployment is different from the decision functions in case of training. It is also very effective when only a few labelled data is available in the deployment conditions.

This type of Dataset Shift where the input values are shifted is known as Covariate Shift [Moreno-Torres et al. (2012)], [Shimodaira (2000)]. Covariate Shift is the situation when the training and test data follow different distributions while the conditional probability distribution remains the same [Moreno-Torres et al. (2012)]. There are also other types of Dataset Shifts such as Concept Drift where the data distribution remains the same but the decision function changes [Moreno-Torres et al. (2012)]. Different kinds of approaches [Sugiyama et al. (2007)], [Bickel et al. (2009)], [Gretton et al. (2009)] have been proposed to deal with covariate shift. But most of these approaches require conspicuous retraining of the created classification models and as a result also require a large amount of labelled data to be available in the deployment context. These approaches are not very applicable in real life scenarios where the availability of labelled data is very low and very limited time is present for classifying the data and as such the aforementioned approaches are not suitable fro reframing. The *Reframing with Stochastic Hill Climbing(RSHC)* [Ahmed et al. (2014)] approach does compensate for covariate shift with real time reframing of the deployment data but it can only compensate for linear shifts in data and cannot differentiate between the linear and non-linear shifts in data.

Our proposed approach can properly address the problem of reframing the input data in deployment dataset for both linear and non linear dataset shifts. Our approach does this in a two step process whereby it first detects whether there is any shift in the data from the test to the deployment dataset. If there exists shift in the data, it then maps the shift in the data to the equation for quadratic transformation which compensates for non-linear shifts in data but can also compensate for linear shift in the data by eliminating the non linear part.

The remainder of this paper is organized in the following manner. In section 2 we discuss the related work. In section 3 we provide our proposed approach for reframing in case of dataset shift and in section 4 experimental results are provided for multiple real life datasets. Finally conclusions are drawn in section 5.

2 Background Study

Different types of score based methods have been proposed to handle classification problems in different deployment scenarios. For example, the binary classification algorithm of [Lachiche and Flach (2003)] generates scores of being positive vs. negative and define one threshold to divide these scores to predict the boundary between two classes. Depending on the deployment environment, typically a matrix of misclassification costs and the prior probability of the classes, this threshold can be tuned and the model is adapted for class prediction . Research has also been done to handle this problem for multi-class classification.

2.1 Dataset Shift

Dataset shift is the phenomenon when the data items in the deployment(testing) dataset undergo a change in the distribution of a single attribute or multiple attributes or a change in the class defining boundaries resulting in the deployment data exhibiting different behaviour than that of the training dataset.

Training Dataset: The training dataset is the set of data items with well defined class labels and attribute values from which the classification model which is being built can easily learn the range of different attributes for each class boundary.

Deployment Dataset: The deployment dataset is the dataset where the data items have no class labels but have the same attributes as the data items of the training dataset. The classification model is used on this dataset to make predictions of the classes of the data items. The classification model studies the attribute values of each data item and uses the knowledge it gained from the training dataset to make predictions about the class of the data items.

These contexts are often different in some non-trivial way. For instance, a model may be built using training data collected in a certain period of time and in a particular country, and deployed to data in a future time and/or in a different country.

2.1.1 Types of Dataset Shift

— **Covariate Shift:** Covariate shift [Shimodaira (2000)] refers to changes in the distribution of the input variables i.e. changes in the attributes of the input data items. Covariate shift is the most studied type of dataset shift. It is also known as "Population Drift" [Hand (2006)], [Kelly and Adams (1999)].

— **Prior Probability Shift :** Changes to the distributions of the class variable i.e. class distributions [Webb and Ting (2005)] is known as Prior Probability Shift.

— **Concept Shift :** It is mainly known as "Concept Drift". It is also defined as "Changes to the definitions of the classes" [Hand (2006)].

In this paper we shall mainly focus on the detection of Covariate Shift and the reframing of input variables in the deployment context in case of Covariate Shift.

2.2 Versatile Decision Tree

The Versatile Decision Tree (VD)T algorithm proposes different approaches to build Decision Trees in the presence of covariate observation shifts [Al-Otaibi et al. (2015)] i.e. this algorithm is only applicable in those cases in which covariate shifts occur in the deployment

data. This algorithm makes two major contributions. The first contribution is that it proposes a new and unforeseen approach to build DTs using percentiles.

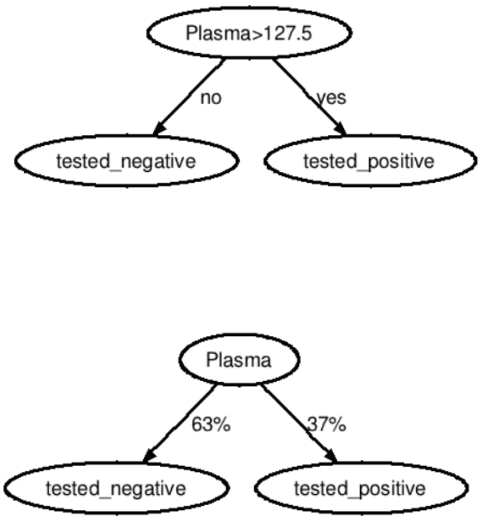

FIG. 1 – *Two types of models; the upper figure is the model using a fixed threshold while the lower figure is the model using percentiles. For each deployment context, the decision tree is deployed in such a way that the deployment instances are split to the leaves in the same percentile amounts of 63 percent and 37 percent.*

The main idea proposed in this algorithm is to learn a conventional DT and then to replace the internal decision thresholds with percentiles which can then deal with monotonic shifts in the deployment data. The second contribution of this algorithm is that it proposes a more general Versatile Model (VM) that deploys different strategies (including percentiles) to update the DT threshold for each deployment context i.e. for each different deployment dataset according to the shifts observed in the data. The shifts are identified by applying a non-parametric statistical test.

2.3 Reframing with Stochastic Hill Climbing

The Reframing with Stochastic Hill Climbing (RSHC) algorithm maps any shift in the data instances from the training to the deployment dataset to the equation for linear transformation:

$$y = \alpha x + \beta$$

By mapping the shift in the data to linear transformation the RSHC algorithm tries to compensate for the shift that occurs in the deployment instance by transforming the distribution of the attributes of the deployment data to that which is shown by the training data.

This algorithm presents the reframing concept to tackle the shift for continuous input attributes of the data items in the deployment dataset and proposes an efficient and simple process of learning the optimal parameter values of the shifted attributes for classification. The input attributes of the deployment data items will be transformed by these parameter values of linear transformation and then will be applied to the originally built classification model with the training database to provide accurate result i.e. list of accurately classified data items of the deployment dataset [Ahmed et al. (2014)].

3 Our Proposed Approach

3.1 RNLDS: Our Proposed Approach

We are proposing a new Algorithm Reframing for Non linear Dataset Shift (RNLDS) which will detect shift in dataset and will map it to a chosen non linear transformation. Our proposed algorithm is an improvement of the Reframing with Stochastic Hill Climbing (RSHC) algorithm proposed in [Ahmed et al. (2014)]. The RSHC Algorithm takes a Hill Climbing approach to determine the most appropriate values of the parameters of the equation for linear transformation. In our algorithm we used the Hill Climbing approach to determine the most appropriate values of the parameters of the equation for non linear transformation which we use. In our proposed algorithm we have used the quadratic equation to map non linear shift in the deployment data i.e. we have used the equation:

$$y = \alpha x^2 + \beta x + \gamma$$

But in real life deployment any of the equations for non linear transformations can be used to map dataset shift. In case of using a different equation the number of Hill Climbing steps in the algorithm have to be changed to the number of parameter values of the equation.

3.2 Example Scenario

Let us assume that we have 3 datasets. Each of which contains the data about the students of a school i.e. the three datasets contains information about the students of three different schools. We assume that each of these schools remain separated from each other i.e they are at different geographical locations and the information about the students of each of these schools were collected at different intervals of time. Let the datasets contain 3 types of data about each student of each of the schools, their attendance, test scores for a certain term and their extra curricular performance expressed in numerical values between 1 and 100. Let we have to classify the students into four non numerical classes of: Bad, Moderate, Good and Excellent according to their scores. Let the three datasets are as follows.

The student information of the students of the School 1 remain labelled with the class labels of the class to which each student belongs according to their test scores. The following information about the students of School 2 and School 3 also remain labelled but it will be up to the classifier to determine their classes and check with the original classification. The following datasets have less information than the first one.

Std. ID	Attendance	Test Score	Extracurricular Performance	Class Label
1	50	23	80	Bad
2	62	30	50	Good
3	57	27	90	Moderate
4	67	40	67	Good
5	70	48	75	Excellent
6	66	56	70	Good
7	20	13	33	Bad
8	70	60	87	Good
9	80	77	40	Excellent
10	60	39	72	Good
11	50	42	60	Moderate
12	72	63	85	Excellent
13	47	29	63	Bad
14	70	37	50	Excellent
15	75	47	92	Excellent

TAB. 1 – *Student performance of a particular term in School 1.*

Std ID	Attendance	Test Score	Extracurricular Performance	Class Label
1	90	88	22	Excellent
2	75	67	80	Good
3	62	52	50	Moderate
4	77	59	46	Moderate
5	57	38	88	Bad
6	60	69	62	Moderate
7	83	72	65	Good
8	97	92	45	Excellent
9	65	52	75	Bad
10	40	32	69	Bad

TAB. 2 – *Student performance of a particular term in School 2.*

We see that though there remains more or less the same percentage of students with bad, moderate, good and excellent academic performances, their performances are not at a similar level. So there is a shift of the data of the 2nd and 3rd datasets from the 1st dataset. Now if we try to map the shift of the data of the 2nd and 3rd datasets to the equation of quadratic transformation i.e.

$$y = \alpha x^2 + \beta x + \gamma$$

it will be seen that for the values of the parameters of the equation i.e. $\alpha = 0.015, \beta = 0.1$ and $\gamma = 7$ we can compensate for the shift in the data of the 2nd set and make it follow the same distribution as the data in the 1st dataset.

Std ID	Attendance	Test Score	Extracurricular Performance	Class Label
1	45	37	62	Bad
2	63	62	50	Good
3	67	70	33	Excellent
4	40	23	80	Bad
5	70	47	25	Moderate
6	23	39	90	Bad
7	90	77	75	Excellent
8	75	52	52	Moderate
9	31	21	77	Bad
10	80	59	47	Good

TAB. 3 – *Student performance of a particular term in School 3.*

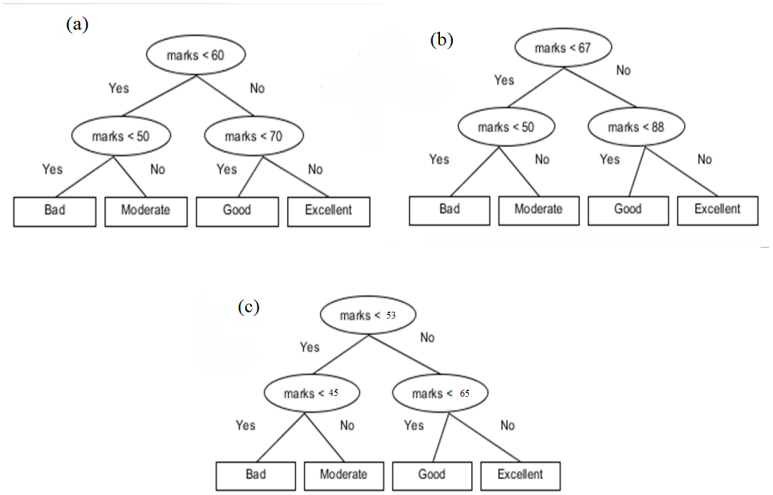

FIG. 2 – *Decision function for three different schools.*

While calculating the accurate parameter values for the quadratic equation to compensate for the shift in datasets 2 and 3 from dataset 1 we use the above given algorithm. The determination of the parameter values is a very complex task as the shift for the different data tuples might be different and we are to determine a single value for each parameter of the quadratic equation which would give the best possible result i.e. accuracy while compensating for the shift in the data during the classification of the data of the deployment dataset using the classification model.

We use the RNLDS algorithm given in Figure 3 to perform the complex task of determining the parameter values of the quadratic equation to compensate for the dataset shift in any type real life data.

Input: C, means of base model; Few deployment data T_d, Test data T_{tst}; Precision of adjustment $p > 0$
Output: Accuracy of the classifier on test datadset.

$param \leftarrow [1, 1, 0]$
for $i \leftarrow 0$ **to** 3 **do**
    $param \leftarrow chooseAlphaBetaGama(param, i, p)$
end for

Classify T_{tst} by applying new shift parameters ($param$) and **return** Accuracy.

procedure CHOOSEALPHABETAGAMMA($param, index, p$)
    for $i \leftarrow 1$ **to** 2 **do**
        $param_t \leftarrow param$
        while true **do**
            $newAccr \leftarrow classifierAccr(T_d, param_t)$
            if $shouldContinue(newAccr) = true$ **then**
                $param_t[index] \leftarrow param_t[index] + p[i]$;
            else
                break;
            end if
        end while
        $saveParam[i] \leftarrow param_t$
        $saveAccr[i] \leftarrow newAccr$
    end for
    $param \leftarrow saveParam[(\ saveAccr[0] \leq saveAccr[1]\ ?\ 0 : 1)]))$
    Return $param$
end procedure

FIG. 3 – *Choosing α, β and γ using Hill Climbing*

4 Experimental Results

We have performed several experiments on synthetic and real-life datasets to show the efficiency and effectiveness of our approach.In this section we shall compare the accuracy of the classification of data given by our approach while classifying the data of a particular dataset against the accuracy of the classification provided by the other approaches while classifying the data of the same dataset as before. We shall provide graphical comparison of the accuracies of the different approaches while classifying the data of a particular dataset for ease of comparison. In our approach we used the Naive Bayesian Classifier in order classify the dataset we wish to use in the deployment data. We have compared our results with the results provided by the base classifier, retraining approach and the linear approach i.e. Reframing with Stochastic Hill Climbing.

4.1 Dataset 1 : Chronic Kidney Disease

We at first run our algorithm on the chronic kidney disease dataset available in the UCI Machine Learning Data Repository. This dataset contains the complete information of patients who were admitted to Apollo hospital in Karaikudi, Tamil Nadu over a period of two months. It can be used to predict which group of people in a certain area can be affected by chronic kidney disease at a certain period of time.We split up the data using the age value i.e. we separated the patients of different age groups into different datasets. Of the different age groups the data of patients belonging to the age group of 0-30 is considered as the training data and the data of patients belonging to age age group of 70+ is considered as the test data.

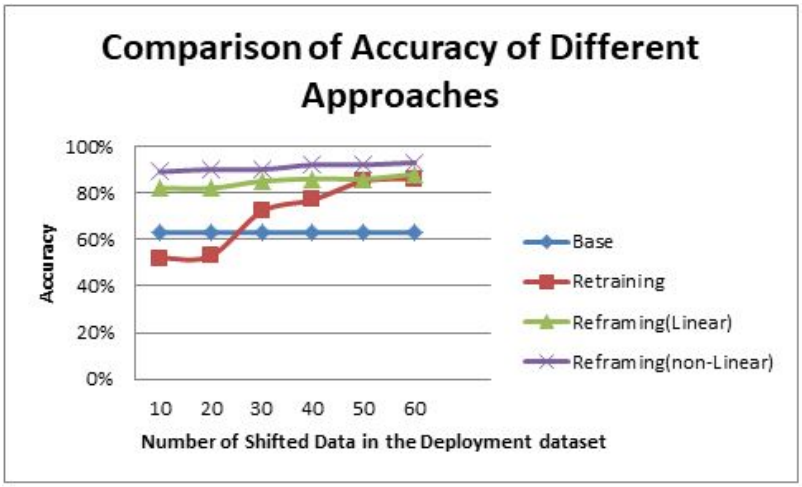

FIG. 4 – *Learning Curve for Reframing for Non Linear Transformation, Reframing for Linear Transformation, Retraining and Base Model.*

4.2 Dataset 2 : Census Income

In our second experiment we run our algorithm on the Census Income dataset available in the UCI Machine Learning Data Repository. This dataset contains the census information of a large number of people of several countries. It contains information on their sex, family, education, occupation etc. We split the dataset up according to country i.e. we consider the information of the people of U.S.A as the training dataset and the information on the people of Jamaica as the deployment dataset. Now, the classification model built using the training data can predict depending on their information if a person living in Jamaica earns a salary of less or more than 50,000.

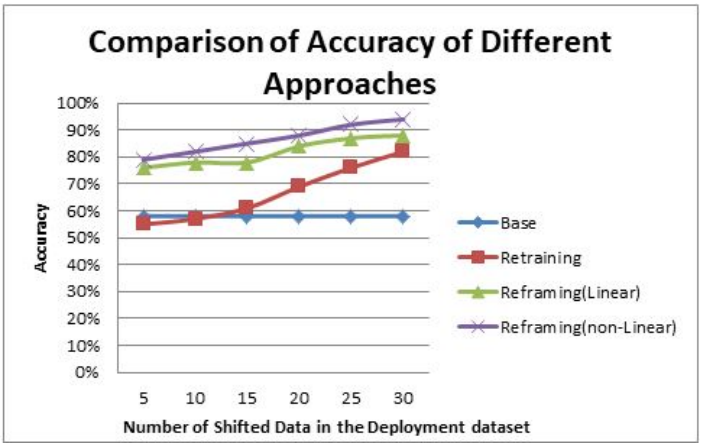

FIG. 5 – *Learning Curve for Reframing for Non Linear Transformation, Reframing for Linear Transformation, Retraining and Base Model.*

4.3 Dataset 3: ILP (Indian Liver Patient)

We now run our algorithm on the ILP (Indian Liver Patient) dataset available in the UCI Machine Learning Data Repository. This dataset contains the complete information of a collection of people of whom a lot were affected with liver diseases and others were not. It was collected from the north east of Andhra Pradesh state of India. It can be used to predict which group of people in a certain area can be affected by kidney disease at a certain period of time. We split up the data according to the sex of the people i.e. we separated the people into two different groups males and females. The dataset of males is considered as the training dataset while that of the females is considered the test dataset. Experimental results in these real-life datasets demonstrate that our approach is quite capable of learning the optimal shift parameter values for the equation for quadratic transformation using almost no labeled data at deployment in a real-life environment where the nature of a shift is unknown from source to deployment. These results also reveal the applicability of non linear shift in real-life domains by clearly expressing its strength to tackle these unknown dataset shifts between one training and different deployment contexts.

5 Conclusion

We have proven with our research that non linear shifts occur in real life datasets.In this paper, we have proposed a new approach of reframing the values of continuous input attributes. Moreover, we have designed an efficient algorithm to learn the optimal parameter values for the shifted continuous input attributes in case of classification of unlabelled data. Our algorithm has the ability to adapt to different types of changes to the distributions of data attributes and thus making the existing model usable in different deployment environments. Even with no labelled data available in the deployment scenario, it can deliver the required optimal parameter

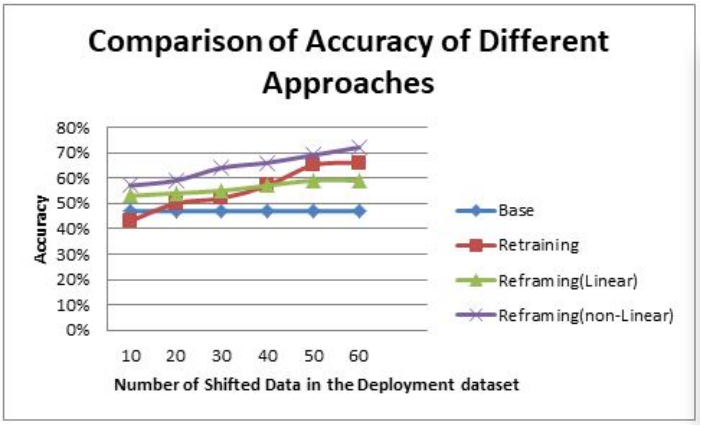

FIG. 6 – *Learning Curve for Reframing for Non Linear Transformation, Reframing for Linear Transformation, Retraining and Base Model.*

values in order to adapt to the actual dataset shift. With extensive experimental results we have shown that our approach is better than the existing reframing based approach using only linear shift in data attributes with respect to accuracy. In the worst case, the results of our approach is equivalent to the results provided by the reframing for linear shift approach which is only possible if the dataset has only linear shift in the deployment scenario. Furthermore, it is quite suitable for those environments where retraining is not applicable to learn the decision functions. Finally, we have presented the existence of dataset shift in three real-life datasets by considering differences of age, geographical area and sex respectively as the difference between the datasets. We have demonstrated the capability of our approach to approximate these unknown real-life dataset shifts accurately.

References

Ahmed, C. H., N. Lachiche, C. Charnay, and A. Braud (2014). Reframing continuous input attributes. *IEEE ICTAI*, 31–38.

Al-Otaibi, R., R. B. Prudencio, M. Kull, and P. Flach (2015). Versatile decision trees for learning over multiple contexts. *Machine Learning and Knowledge Discovery in Databases*, 184–199.

Bickel, S., M. Bruckner, and T. Scheffer (2009). Discriminative learning under covariate shift. *Journal of Machine Learning Research 10*, 2137–2155.

Charnay, C., N. Lachiche, and A. Braud (2013). Pairwise optimization of bayesian classifiers for multi-class cost-sensitive learning. *IEEE ICTAI*, 499–505.

Gretton, A., A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Scholkopf (2009). Covariate shift by kernel mean matching. *Dataset Shift in Machine Learning*, 131–160.

Han, J., M. Kamper, and J. Pei (2012). *Data Mining Concepts and Techniques*. 225 Wyman Street, Waltham, MA 02451, USA: Morgan Kaufmann Publishers.

Hand, D. J. (2006). Classifier technology and the illusion of progress. *Statistical Science 21(1)*, 30–34.

Hernandez-Orallo, J. (2013). Roc curves for regression. *Pattern Recognition 46(12)*, 3395–3411.

Kelly, M.G., D. H. and N. Adams (1999). The impact of changing populations on classifier performance. *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 58(1)*, 367–371.

Lachiche, N. and P. A. Flach (2003). Improving accuracy and cost of two-class and multi-class probabilistic classifiers using roc curves. *ICML*, 416–423.

Moreno-Torres, J. G., T. Raeder, R. Alaiz-Rodriguez, N. V. Chawla, and F. Herrera (2012). A unifying view on dataset shift in classification. *Pattern Recognition 45*, 521–530.

Shimodaira (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference 90(2)*, 227–244.

Sugiyama, M., M. Krauledat, and K.-R. Muller (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research 8*, 985–1005.

Webb, G. and K. Ting (2005). On the application of roc analysis to predict classification performance under varying class distributions. *Machine Learning 58(1)*, 25–32.

Zhao, H., A. P. Sinha, and G. Bansal (2011). An extended tuning method for cost-sensitive regression and forecasting. *Decision Support Systems 51(3)*, 372–383.

Résumé

Les modèles de classification discriminante supposent que les données de formation et de déploiement ont les mêmes distributions d'attributs de données. Ces modèles donnent des performances très variées lorsqu'ils sont déployés dans des conditions variées avec différentes distributions de données. Ce phénomène est appelé Dataset Shift. Dans cet article, nous avons fourni une méthode qui détermine d'abord s'il y a un changement significatif dans les distributions d'attributs entre les ensembles de données d'apprentissage et de déploiement. S'il existe un changement dans les données, la méthode proposée utilise ensuite une approche de Hill climbing pour cartographier ce décalage, quelle que soit sa nature, c'est-à-dire (linéaire ou non linéaire) à l'équation pour la transformation quadratique. Les résultats expérimentaux sur trois jeux de données réels montrent de forts gains de performance obtenus par la méthode proposée par rapport aux méthodes précédemment établies telles que le reconditionnement et le recadrage linéaire.