# Dataset Shift in a Real-Life Dataset

Chowdhury Farhan Ahmed, Nicolas Lachiche, Clément Charnay, and Agnès Braud

ICube Laboratory, University of Strasbourg, France
{cfahmed, nicolas.lachiche, charnay, agnes.braud}@unistra.fr

**Abstract.** Dataset shift refers to the problem where training and testing datasets follow different distributions. Since this problem may occur in many real-life scenarios, detecting and tackling dataset shift becomes an important research issue in machine learning. Even though it is natural to observe shift in several real-life datasets, unfortunately, existence of clear dataset shift is rarely found in the publicly available real-life datasets. In this paper, we present the existence of significant dataset shift in a real-life dataset. We experimentally analyze how to split this dataset to achieve remarkable shifts in both input and output variables. Moreover, future research directions are discussed where this dataset can effectively be used as a real-life benchmark.

**Keywords:** Machine Learning, Dataset Shift, Real-Life Datasets.

## 1 Introduction

Machine learning techniques build a model in a training environment and deploy that model in different environments. It is natural in the real-life scenarios that data distributions and decision functions may change from one location to another location. This phenomenon is called Dataset Shift [11, 16]. However, if a trained model is deployed at the destination without detecting and adapting the occurred dataset shift, a large amount of classification errors may take place. Therefore, detecting and tackling dataset shift is a crucial problem in machine learning and data mining.

Consider a real-life scenario described in Fig. 1 where we are building a model using training data taken from City 1. We discovered a knowledge from the training data that once the temperature is greater than $25°C$, most of the people buy an ice-cream. Hence, we build a classifier based on the knowledge of the continuous attribute temperature. Afterwards, we move to City 2 and observe the same event happens when the temperature is greater than $18°C$. Now if we rescale this data by adding a numeric value of 7, we can easily use our trained model for City 1 without any modification and be able to predict whether a given person in City 2 will buy an ice-cream or not. This is a very simple example of detecting and tackling dataset shift.

Several methods [20, 17, 4, 7, 13] have been proposed to use the learnt model at different deployment environments by adjusting the output/input values.


**Fig. 1.** An example of dataset shift.

However, most of the existing dataset shift related papers have done their experiments using synthetic datasets or real-life datasets with artificial shifts. For example, Moreno Torres [13, 12] created some benchmark datasets for dataset shift[1]. This dataset repository contains some real-life binary classification datasets in original (the original datasets are also available at [1] and UCI Machine Learning Repository [2]) and shifted format while artificial shifts are generated randomly in one arbitrary attribute. Few existing methods used real-life datasets, but those datasets are restricted and not publicly available, for example, Brain-Computer Interface dataset of IWCV [17].

The importance of dataset shift problem in machine learning as well as the non-availability of public real-life datasets motivated us to find shifts in a real-life dataset. In this paper, we show the existence of dataset shift in the real-life Bike Sharing dataset [6, 2]. The original use of this dataset was to detect some events in a city and [6] did not mention anything about dataset shift. Appropriate splits exhibit dataset shifts on both input and output variables. The presence and absence of dataset shifts in different splits will be shown experimentally. Moreover, future research directions are given to exploit this dataset as a real-life benchmark.

The remainder of this paper is organized as follows. Section 2 discusses related work. In Section 3, we describe the real-life Bike Sharing dataset. In Section 4, our proposed splits to achieve dataset shifts in this dataset are presented and analyzed. Section 5 presents the suitability of Kaggle split on this dataset for dataset shift. Finally, conclusions are drawn in Section 6 with some future research directions.

## 2 Related Work

Some score-based methods have been proposed to handle classification problem at different deployment scenarios. For example, the binary (2-class) classification algorithm of [10] generates scores of being positive vs. negative and defines one threshold to divide these scores to predict the boundary between two classes. According to the deployment environment, typically a matrix of misclassification

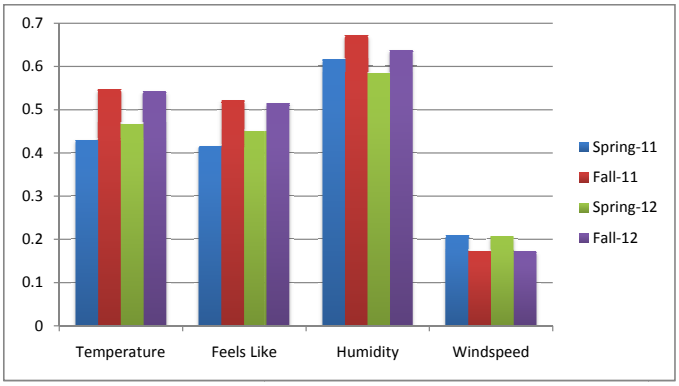
[1] http://sci2s.ugr.es/dataset-shift/

costs and the prior probability of the classes, this threshold can be tuned and the model is reframed for class prediction. Research has also been done in this area to handle multi-class classification problems [10, 19, 14, 5, 15].

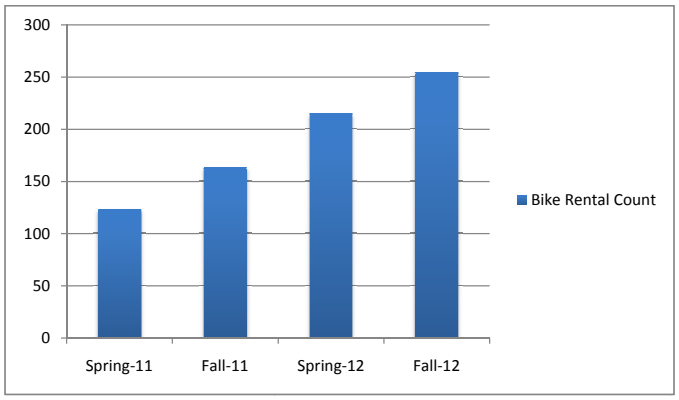
It is also possible to adapt regression outputs when the cost function changes. A tuning method has been proposed to adjust the average misprediction cost of a cost-sensitive regression model [3] by assuming the cost function to be convex. This method adds/subtracts a constant shift to all the predicted values of the original regression model. Later, the same authors, extended their idea to any polynomial transformation [20]. A model of ROC analysis for regression is developed by [9]. Two axes have been marked with over and under estimations of predicted values. The ROC curve is drawn by considering several shift values with the original predictions. Reframing relies on adding a shift to the original regression model. This shift is chosen according to the deployment conditions, typically the loss function and the prior distribution of instances in the deployment condition. However, the above mentioned methods perform the adjustments on the model output.

In the data mining and machine learning literature, input variable shift is most often known as covariate shift [11]. The situation when the training and test data follow different distributions while the conditional probability distribution remains same, is called the covariate shift. Besides covariate shift, there exists some other important cases of dataset shifts such as concept drift where data distribution remains the same but the decision function changes [11]. Some research works have been done to tackle covariate shift. A new variant of cross validation, called Importance Weighted Cross Validation (IWCV), is proposed by assuming that the ratio of the test and training input densities is known at training time [17]. According to this ratio, it applies different weights to each input during cross validation to handle covariate shift. Another method, called Integrated Optimization Problem (IOP), derives a discriminative model for learning under different training and test distributions. The contribution of each training instance to the optimization problem ideally needs to be weighted with its test-to-training density ratio [4]. It characterizes what the possibility of an instance to occur in the test set is rather than in the training set. Instantiating the general optimization problem leads to a kernel logistic regression and an exponential model classifier for covariate shift [4]. In order to cope with covariate shift, Kernel Mean Matching (KMM) approach [7] reweighs the training data such that the means of the training and test data in a reproducing kernel Hilbert space (i.e., in a high dimensional feature space) are close.

Recently, an input transformation based approach, called GP-RFD (Genetic Programming-based feature extraction for the Repairing of Fractures between Data), has been proposed which can handle general dataset shift [13]. At first, it creates a classifier $C$ for a training dataset $A$ and considers $B$ as a test dataset with different distribution. To discover the optimal transformation, it applies several genetic operators (e.g., selection, crossover, mutation) on $B$ and creates dataset $S$. Subsequently, it applies $C$ on dataset $S$ and calculates the accuracy in order to determine the best transformation. This approach is computationally




(a)




(b)

**Fig. 2.** Distributions of the average values of (a) input and (b) output attributes in the semester splits.

huge and needs a large amount of data of $S$ to be labelled and can almost be referred to as retraining. Moreover, its accuracy may fall below the base model's accuracy (the classifier trained with the training data and used directly at the deployment) if it cannot learn the optimal parameter values with its random genetic programming trials [12].

## 3 Description of the Bike Sharing Dataset

The Bike Sharing dataset [6, 2] contains usage logs of a bike sharing system called Capital Bike Sharing (CBS) at Washington, D.C., USA for two years (2011 and 2012). The dataset was prepared by Fanaee-T and Gama [6], and is publicly available in UCI Machine Learning Repository [2]. It contains bike rental counts hourly and daily based on the environmental and seasonal settings. The *hour.csv*

**Table 1.** Distributions (in detail) of the input and output attributes in the semester splits.

| Attribute | Measure | Spring-11 | Fall-11 | Spring-12 | Fall-12 |
|---|---|---|---|---|---|
| Temperature | min | 0.02 | 0.14 | 0.02 | 0.14 |
| | max | 0.94 | 0.96 | 0.98 | 1 |
| | avg | 0.43 | 0.546 | 0.467 | 0.543 |
| | std | 0.199 | 0.179 | 0.176 | 0.189 |
| Feels Like | min | 0 | 0.152 | 0.015 | 0.152 |
| | max | 0.879 | 1 | 0.924 | 0.909 |
| | avg | 0.416 | 0.521 | 0.45 | 0.515 |
| | std | 0.18 | 0.157 | 0.159 | 0.168 |
| Humidity | min | 0 | 0.16 | 0.16 | 0.16 |
| | max | 1 | 1 | 1 | 1 |
| | avg | 0.615 | 0.671 | 0.584 | 0.638 |
| | std | 0.208 | 0.18 | 0.201 | 0.171 |
| Windspeed | min | 0 | 0 | 0 | 0 |
| | max | 0.806 | 0.851 | 0.806 | 0.657 |
| | avg | 0.21 | 0.173 | 0.206 | 0.172 |
| | std | 0.126 | 0.118 | 0.126 | 0.115 |
| Bike Rental Count | min | 1 | 1 | 1 | 1 |
| | max | 638 | 651 | 957 | 977 |
| | avg | 123.448 | 163.47 | 215.162 | 254.091 |
| | std | 123.951 | 139.902 | 195.559 | 219.717 |

file [2] contains 17,379 records where the values of bike sharing counts have been aggregated on hourly basis. Similarly the *day.csv* file [2] aggregates the bike sharing counts on daily basis and contains 731 records. The input variables contain day, hour, season, workday/holiday and some weather information such as temperature, feels like temperature, humidity and wind speed.

The original objective of the creators of this dataset was event and anomaly detection. It is explained in [6] that count of rented bikes are correlated to some events in the town which easily are traceable via search engines. For instance, query like "2012-10-30 Washington D.C." in Google returns related results to Hurricane Sandy. The method proposed in [6] showed how monitoring the residuals of predicted and actual bike rental counts help to detect event together with a search engine and some background knowledge.

However, we have found significant shifts inside this dataset. In the next section, we will discuss some useful splits to observe these shifts along with the distribution analysis of input and output variables.

**Table 2.** Performance of Base Model Spring-11 in other semesters.

| Source | Measure | Deployment | | | |
|---|---|---|---|---|---|
| | | Spring-11 | Fall-11 | Spring-12 | Fall-12 |
| Spring-11 | MAE | 71.789 | 102.293 | 132.226 | 158.884 |
| | RMSE | 99.058 | 135.427 | 186.459 | 224.307 |

## 4 Occurrences of Dataset Shift

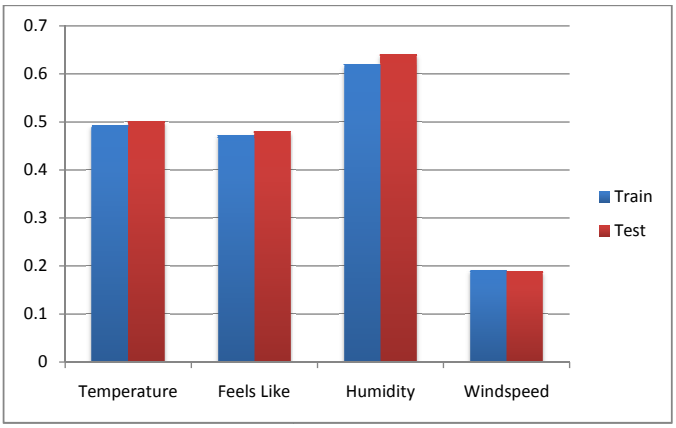
In real-life, weather changes according to the change of seasons. Moreover, the renting behaviour of people may also change according to time. To observe these changes in this real-life dataset, we have splitted this dataset into four parts according to six months of sequential time. Thus, these four parts can also be referred to as four semesters of these two years and labelled as Spring-11, Fall-11, Spring-12, Fall-12 (Spring: January-June, Fall: July-December).

Fig. 2(a) shows distributions of four most influential input attributes of this dataset representing weather information. They are temperature, feels like temperature, humidity and windspeed. The values of these attributes have been normalized as follows

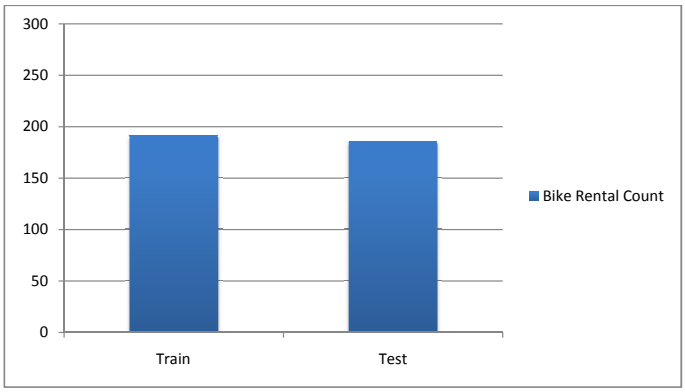
- Temperature: The values (Celsius) are normalized by dividing by 41 (max).
- Feels Like Temperature: The values (Celsius) are normalized by dividing by 50 (max).
- Humidity: The values are normalized by dividing by 100 (max).
- Windspeed: The values are normalized by dividing by 67 (max).

However, we have shown the average value of each attribute in an hour for that particular semester in Fig. 2(a). Shifts in these attributes from one semester to another can easily be observed in this figure. For example, the average temperature of Spring-11 is 0.43, while the average temperature of Fall-11 is 0.546. On the other hand, Fig. 2(b) represents the increasing bike renting behaviour of customers with respect to time. It represents the average bike rental counts in an hour in those specified semesters. The shift is remarkable when we observe it from one semester to another. For instance, the average bike rental counts of Fall-11 and Spring-12 are 163.47 and 215.162, respectively. In addition, we have reported the detailed distributions of these attributes using the minimum, maximum, average and standard deviation measures in Table 1 for better understanding their characteristics.

Now, we report an experimental study in Table 2 to validate the presence of dataset shift in these splits. We have trained a regression model in Spring-11 and tested in the other three semesters. A regression tree, called REPTree, from Weka [8, 18] is used to build the regression model, and two measures, MAE (mean absolute error) and RMSE (root mean square error), are used to show




(a)




(b)

**Fig. 3.** Distributions of the average values of (a) input and (b) output attributes in the Kaggle split.

the performance. The third column indicates the performance of 10-fold cross-validation in Spring-11. Columns 4, 5 and 6 indicate the performance of Spring-11 regression model in Fall-11, Spring-12 and Fall-12, respectively. The effect of dataset shift is quite obvious here. Moreover, the prediction errors are increasing according to time which reflects the distributional changes of attributes in Fig. 2. Therefore, we observe dataset shift in this real-life dataset using these splits. It would be similar with some other shifts with respect to months, terms and years.

**Table 3.** Distributions (in detail) of the input and output attributes in the Kaggle split.

| Attribute | Measure | Train | Test |
|---|---|---|---|
| Temperature | min | 0.02 | 0.02 |
| | max | 1 | 0.98 |
| | avg | 0.493 | 0.501 |
| | std | 0.19 | 0.197 |
| Feels Like | min | 0.015 | 0 |
| | max | 0.909 | 1 |
| | avg | 0.473 | 0.48 |
| | std | 0.17 | 0.176 |
| Humidity | min | 0 | 0.16 |
| | max | 1 | 1 |
| | avg | 0.619 | 0.641 |
| | std | 0.192 | 0.193 |
| Windspeed | min | 0 | 0 |
| | max | 0.851 | 0.836 |
| | avg | 0.191 | 0.189 |
| | std | 0.122 | 0.123 |
| Bike Rental Count | min | 1 | 1 |
| | max | 977 | 976 |
| | avg | 191.574 | 185.924 |
| | std | 181.144 | 181.753 |

## 5 The Split of Kaggle

Recently, Kaggle[2] has provided a problem on Bike Sharing dataset. The original dataset has been divided into two parts called train and test. The train part contains data of each month from day 1 to 19, and test part contains data from day 20 to the end of a month. The problem is to build a regression model with the train dataset and predict the bike rental counts in the test dataset. In this section, we have analyzed whether there is a dataset shift in this split.

Fig. 3(a) and Fig. 3(b) show the distributions (average) of input and output attributes, respectively as like Section 4. It can be noticed that there is almost no shift either in the input or output variables. For better understanding, Table 3 reports the detailed distributions of these attributes.

Afterwards, we have presented an experimental study in Table 4. It also indicates the non-existence of dataset shift between the train and test datasets. Actually, here the train and test datasets have been prepared by mixing data of every month. This is the main reason behind the absence of dataset shift in this Kaggle split.

[2] https://www.kaggle.com/c/bike-sharing-demand

**Table 4.** Performance of Base Model Train in Test for the Kaggle split.

| Source | Measure | Deployment | |
|---|---|---|---|
| | | Train | Test |
| Train | MAE | 117.595 | 117.17 |
| | RMSE | 158.607 | 157.517 |

## 6 Conclusions and Open Questions

It is difficult to get a publicly available real-life dataset with dataset shift. The main contribution of this paper is to detect and present remarkable dataset shifts inside a publicly available real-life dataset even though the objective of the original authors when the created this dataset was quite different. Moreover, we have analyzed the characteristics of input and output variables of this dataset and experimentally show some splits that exhibit dataset shift and some that do not.

This paper proposes benchmark to compare transfer learning, reframing and domain adaptation approaches. A learning model can be trained in one scenario and tested in different scenarios where the data distributions and decision functions are quite different. It will also be very useful if the learning task is incremental or online. For instance, train a model in month 1 and test in month 2. When we have the output labels of month 2, update the model for first two months and predict the output labels for month 3, and so on. Moreover, availability of data in both hour and day formats makes it eligible for multi-level learning, e.g, learning for first 15 days and try to predict the output of the different hours in day 16.

## Acknowledgments

This work was supported by the REFRAME project granted by the European Coordinated Research on Long-term Challenges in Information and Communication Sciences & Technologies ERA-Net (CHIST-ERA).

## References

1. Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S.: Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. Multiple-Valued Logic and Soft Computing 17(2-3), 255–287 (2011)
2. Bache, K., Lichman, M.: UCI machine learning repository. [http://archive.ics.uci.edu/ml] (2013)

3. Bansal, G., Sinha, A.P., Zhao, H.: Tuning data mining methods for cost-sensitive regression: A study in loan charge-off forecasting. J. of Management Information Systems 25(3), 315–336 (2009)
4. Bickel, S., Brückner, M., Scheffer, T.: Discriminative learning under covariate shift. Journal of Machine Learning Research 10, 2137–2155 (2009)
5. Charnay, C., Lachiche, N., Braud, A.: Pairwise optimization of bayesian classifiers for multi-class cost-sensitive learning. In: IEEE International Conference on Tools with Artificial Intelligence (ICTAI). pp. 499–505 (2013)
6. Fanaee-T, H., Gama, J.: Event labeling combining ensemble detectors and background knowledge. Progress in Artificial Intelligence 2(2-3), 113–127 (2014)
7. Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., Schölkopf, B.: Covariate shift by kernel mean matching. In: Dataset Shift in Machine Learning, pp. 131–160. The MIT Press (2009)
8. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: An update. SIGKDD Explor. Newsl. 11(1), 10–18 (Nov 2009)
9. Hernández-Orallo, J.: ROC curves for regression. Pattern Recognition 46(12), 3395–3411 (2013)
10. Lachiche, N., Flach, P.A.: Improving accuracy and cost of two-class and multi-class probabilistic classifiers using ROC curves. In: International Conference on Machine Learning (ICML). pp. 416–423 (2003)
11. Moreno-Torres, J.G., Raeder, T., Alaíz-Rodríguez, R., Chawla, N.V., Herrera, F.: A unifying view on dataset shift in classification. Pattern Recognition 45(1), 521–530 (2012)
12. Moreno-Torres, J.G.: Dataset shift in classification: Terminology, benchmarks and methods. PhD Thesis, Available Online: http://sci2s.ugr.es/publications/ficheros/Thesis_JGMorenoTorres_2013.pdf (2013)
13. Moreno-Torres, J.G., Llorà, X., Goldberg, D.E., Bhargava, R.: Repairing fractures between data using genetic programming-based feature extraction: A case study in cancer diagnosis. Information Sciences 222, 805–823 (2013)
14. O'Brien, D.B., Gupta, M.R., Gray, R.M.: Cost-sensitive multi-class classification from probability estimates. In: International Conference on Machine Learning (ICML). pp. 712–719 (2008)
15. Pires, B.A., Szepesvári, C., Ghavamzadeh, M.: Cost-sensitive multiclass classification risk bounds. In: International Conference on Machine Learning (ICML) (3). pp. 1391–1399 (2013)
16. Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N.D.: Dataset Shift in Machine Learning. The MIT Press (2009)
17. Sugiyama, M., Krauledat, M., Müller, K.R.: Covariate shift adaptation by importance weighted cross validation. Journal of Machine Learning Research 8, 985–1005 (2007)
18. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers Inc., Amsterdam, 3 edn. (2011)
19. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD). pp. 694–699 (2002)
20. Zhao, H., Sinha, A.P., Bansal, G.: An extended tuning method for cost-sensitive regression and forecasting. Decision Support Systems 51(3), 372–383 (2011)