PREDICTING TARGETS WITH DATA FROM NON-CONFORMING SOURCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine learning applications to real-world settings are often tasked with making predictions on data generated by multiple sources. There are many methods for understanding when data is Out-Of-Distribution (OOD). A less explored area of importance is where OOD data can be considered In-Distribution (ID) when conditioned by its generating data source. Within this preliminary research, we focus on this issue and propose methods for building classification models capable of making predictions on data in which labels can depend on their source.

1 INTRODUCTION

Within applications of machine learning to real-world settings, distinct data sources appear naturally. Within Federated Learning, there is the concept of personalised models (Tan et al., 2022), and there is extensive literature on the detection of Out-Of-Distribution (OOD) data (Yang et al., 2021) and learning from noisy labels (Song et al., 2022), focused on the rejection of data points that might lead to reduction on performance on In-Distribution (ID) data.

However, here we focus on the setting in which data sources generate similarly distributed data points, with different label assignments that we want to learn. This context was discussed in Wiens et al. (2019) and Oh et al. (2018) in which variables associated with some risk in one hospital were protective in another. Given a dataset, D, we consider that D is composed of several sources, S where $D = \bigcup_{i \in |S|} S_i$. Each data source, S_i, contains its own label assignment, such that a target for a data point depends on its source. Within this preliminary work, we consider all input data to be drawn from the same distribution, with only the label assignments depending on the source.

2 PROPOSED SOLUTIONS

To learn multiple label assignments, dependent on a point's underlying data source, we evaluated multiple methods [1]. **Input Append:** We include the source information in the input. **Classifier Append:** We append the source information to the output of a predictive model and apply another fully connected layer before the output. **Predict Mapping:** We use the source information to predict a matrix that maps the outputted layer from the predictive model to a source dependent output (Figure 1a). **Separate Classifier:** A separate classification layer is built per source (Figure 1b). **Calculate Mapping:** A binary mapping matrix is calculated per source, to allow for the mapping of the output from a predictive model to the source assigned labels, motivated by a change in basis of the label-spaces (Figure 1b). See A.2 for more information.

Predicting or calculating a matrix to perform a change of basis on the label-space is motivated by the following: consider a predictive model assigning a class, $c \in C$ to a data point. A source dependent label-space is equivalent to the space spanned by C, under a change of basis. If the source dependent spaces contain the same number of dimensions, calculating the mapping between them can be done by calculating a change of basis matrix, i.e: for all labels in C_{S_i}, labels in C_{S_j} can be calculated using $C_{S_i} M_{S_j}^{S_i} = C_{S_j}$. Since C_{S_i} can be written as the identity matrix, $M_{S_j}^{S_i} = C_{S_j}$. Therefore, to calculate the labels in S_j based on S_i, C_{S_j} should be ordered such that C_{S_i} forms the identity matrix. However, since we do not know the label of a single data point under two different sources, we are

[1] Experiment code is made available in the supplementary material.

unable to calculate $M_{S_j}^{S_i}$ directly. Instead, we attempt to calculate the change of basis between the predicted label from the final layer of the predictive model, and the source dependent labels.

This insight is used in "Predict Mapping" and "Calculated Mapping", in which the former uses source information and gradient descent to learn the mapping, and the latter calculates the mapping between the predicted label and the expectation of the true labels on each batch of data, for each source. "Separate Classifier" (a fully connected layer for each source) is closely related, except that the mapping contains a bias and a continuous valued matrix.

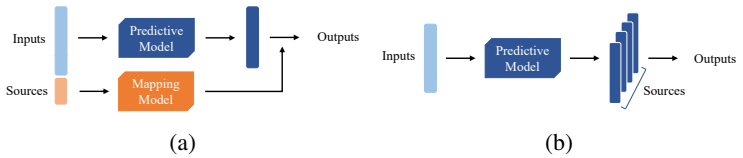


Figure 1: (a)"Predict Mapping" , (b) "Separate Classifier" and "Calculated Mapping".

3 Results

To evaluate how these methods performed at predicting source dependent labels, we employed two public datasets. **MNIST** (LeCun et al., 2010); a collection of images of handwritten digits, and **PTB-XL** (Wagner et al., 2020; A et al., 2000); a collection of ECG measurements and a label corresponding to normal or abnormal heart behaviour (Section A.1). A Multi-Layer Perceptron was used for MNIST, and a 1D ResNet (He et al., 2016) was used for PTB-XL (Section A.2).

Table 1: Accuracy (standard deviation) % on MNIST with an MLP model.

Model	Number of Sources				
	5	10	50	100	500
Input Append	94.2 (0.4)	93.1 (0.4)	86.4 (0.6)	68.7 (6.8)	11.3 (0.1)
Classifier Append	94.1 (1.0)	79.0 (5.3)	26.4 (0.9)	17.7 (3.9)	11.0 (1.0)
Predict Mapping	94.4 (1.2)	75.9 (5.6)	30.9 (3.4)	21.2 (1.1)	13.2 (0.5)
Separate Classifier	**96.5 (0.2)**	**96.3 (0.3)**	**95.1 (0.8)**	**94.9 (0.3)**	**90.5 (0.9)**
Calculate Mapping	93.5 (3.2)	93.2 (3.3)	70.3 (5.4)	75.4 (10.0)	63.4 (5.1)

Table 2: Accuracy (standard deviation) % on PTB-XL with a ResNet model.

Model	Number of Sources			
	5	10	50	100
Classifier Append	60.0 (11.9)	60.4 (5.7)	51.3 (1.5)	50.5 (1.1)
Predict Mapping	77.2 (7.0)	64.8 (5.6)	56.8 (5.0)	52.2 (1.8)
Separate Classifier	84.1 (1.0)	77.9 (11.9)	79.8 (13.3)	78.2 (12.7)
Calculate Mapping	**84.3 (2.0)**	**84.1 (0.9)**	**83.4 (1.8)**	**83.0 (1.5)**

The "Separate Classifier" performed the best on MNIST (Table 1), with the least drop in performance as the number of sources increased. The performance of "Calculate Mapping" did not scale well with the number of sources in this experiment. On PTB-XL (Table 2), "Calculate Mapping" performed best, and scaled well when the number of sources was increased.

4 Concluding Remarks

Models capable of performing consistently and reliably across different data sources are important as they can enable more fair and robust predictions on real-world data. In this preliminary, proof-of-concept study, we suggest multiple methods for adapting predictive models to work on datasets in which label assignments are dependent on the source generating a data point. We found that creating separate classifier heads per source, and calculating a mapping from predictive model output to source predictions both performed well on different datasets. Future work will focus on developing methods further and researching how mappings can be learned directly from the source embedding.

URM STATEMENT

The authors acknowledge that at least one key author of this work meets the URM criteria of ICLR 2023 Tiny Papers Track.

REFERENCES

Goldberger A, Amaral L, Glass L, Hausdorff J M, Ivanov PC, Mark R, Mietus J, Moody G, Peng C, and Stanley H. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000. Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2016. doi: 10.1109/cvpr.2016.90. URL https://doi.org/10.1109/cvpr.2016.90.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL https://arxiv.org/abs/1412.6980.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Jeeheh Oh, Maggie Makar, Christopher Fusco, Robert McCaffrey, Krishna Rao, Erin E. Ryan, Laraine Washer, Lauren R. West, Vincent B. Young, John Guttag, David C. Hooper, Erica S. Shenoy, and Jenna Wiens. A generalizable, data-driven approach to predict daily risk of clostridium difficile infection at two large academic health centers. *Infection control and hospital epidemiology*, 39(4):425–433, March 2018. doi: 10.1017/ice.2018.16. URL https://doi.org/10.1017/ice.2018.16.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–19, 2022. doi: 10.1109/TNNLS.2022.3152527.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–17, 2022. doi: 10.1109/TNNLS.2022.3160699.

Patrick Wagner, Nils Strodthoff, Ralf-Dieter Bousseljot, Dieter Kreiseler, Fatima I. Lunze, Wojciech Samek, and Tobias Schaeffter. PTB-XL, a large publicly available electrocardiography dataset. *Scientific Data*, 7(1), May 2020. doi: 10.1038/s41597-020-0495-6. URL https://doi.org/10.1038/s41597-020-0495-6.

Jenna Wiens, Suchi Saria, Mark Sendak, Marzyeh Ghassemi, Vincent X. Liu, Finale Doshi-Velez, Kenneth Jung, Katherine Heller, David Kale, Mohammed Saeed, Pilar N. Ossorio, Sonoo Thadaney-Israni, and Anna Goldenberg. Do no harm: a roadmap for responsible machine learning for health care. *Nature Medicine*, 25(9):1337–1340, Sep 2019. ISSN 1078-8956, 1546-170X. doi: 10.1038/s41591-019-0548-6.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey, 2021. URL https://arxiv.org/abs/2110.11334.

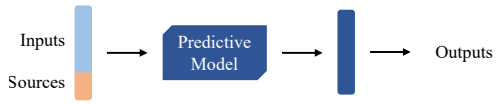
A APPENDIX

A.1 THE DATASETS

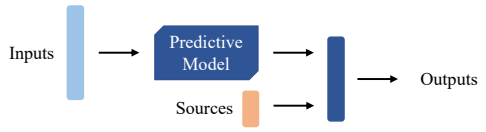
MNIST (LeCun et al., 2010): A collection of 70,000 images of handwritten digits of resolution 28×28.

PTB-XL (Wagner et al., 2020; A et al., 2000): A collection of 21837 ECG measurements using 12 leads sampled at 100Hz on 18885 patients, and a label corresponding to normal or abnormal heart behaviour. Each data point represented 10 seconds of measurements.

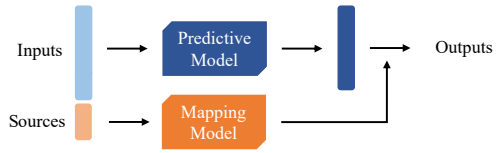
A.2 MODEL ARCHITECTURES AND HYPER-PARAMETERS



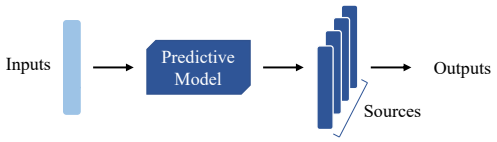
(a) **Input Append:** We include the source information in the input.



(b) **Classifier Append:** We append the source information to the output of a predictive model and apply another fully connected layer before the output.



(c) **Predict Mapping:** We use the source information to predict a matrix that maps the outputted layer from the predictive model to a source dependent output.



(d) **Separate Classifier:** A separate classification layer is built per source. **Calculate Mapping:** A mapping matrix is calculated per source, to allow for the mapping of the output from a predictive model to the source assigned labels.

Figure 2

MNIST: The predictive model shown in Figure 2 is a Multi-Layer Perceptron (MLP) with hidden layers of sizes [100, 100, 10]. The entire model was trained for 100 epochs, with an early stopping criterion on the validation loss with a patience of 5 and a tolerance of 0.0001. It was optimised using Adam (Kingma & Ba, 2014) with a learning rate of 0.001, $(\beta_1, \beta_2) = (0.9, 0.999)$, and a batch size of 512. When calculating the mapping per batch, we used a weighted sum in which the newly calculated mapping was weighted 5:1 compared to the previous mapping. These parameters were chosen using a grid search.

PTBXL: The predictive model shown in Figure 2 is a 1 dimensional ResNet (He et al., 2016) with 4 blocks, each increasing the number of features by 12 and decreasing the number of output time steps by a factor of 4. Since the input had 12 features and 1000 time steps, the final output from the ResNet blocks had 60 features and 3 time steps, which when flattened, was passed to a hidden layer that outputted 2 features, for the binary prediction task. The entire model was trained for 100 epochs, with an early stopping criterion on the validation loss with a patience of 5 and a tolerance of 0.0001. It was optimised using Adam (Kingma & Ba, 2014) with a learning rate of 0.001, an L2 parameter of 0.0001, $(\beta_1, \beta_2) = (0.99, 0.999)$, and a batch size of 1024. When calculating the mapping per batch, we used a weighted sum in which the newly calculated mapping was weighted 1:1 compared to the previous mapping. These parameters were chosen using a grid search.

All experiments were run on random permutations of the dataset, and with new random model parameter initialisations 5 times to ensure reliability.