Doubly Robust Joint Learning for Factorization Machine on Data Missing Not at Random

Xing Hong[1,*], Mingyu Li[1], Dongming Li[2], Ziyu Liu[3], Hantian Zhang[4]

[1] University of California, San Diego, 92093, United States
[2] Boston University, Boston, 02215, United States
[3] Shanghai Pinghe School, Shanghai, 201206, China
[4] Xiamen University Malaysia, Sepang, 43900, Malaysia
*Corresponding author's e-mail: Tim02020@protonmail.com

Abstract. When a recommendation system is preferred to use for predicting the ratings, it is common that many users-to-items ratings cannot be found. And what makes the prediction even trickier is that these missing ratings are missing not at random (MNAR), meaning that there are other factors contributing to why some ratings cannot be found. In order to make better predictions, common approaches to reducing the prediction errors for those missing ratings are using imputed errors and weighting the ratings that are observed with the observed propensities, but each of them is not perfect enough (since they still show bad performance with bias or heavily influenced by the variance of propensities in estimation). To generate a better recommendation system algorithm that could produce a satisfactory outcome, a new estimator combining the imputed errors and propensities in a doubly-robust way is designed to perform an unbiased estimation and meanwhile reduce the influence caused by the propensity variance. In addition, joint learning with error imputation and rating prediction based on the estimator is adopted to acquire a more reliable performance, which shows a better result than the dataset without doubly-robust joint learning.

Keywords- EIB (error-imputation-based), IPS (inverse-propensity-score), Doubly Robust (DR), Logistic Regression, Joint Learning

1. Introduction

A recommendation system is an algorithm predicting users' behaviors. The recommendation list includes the features of users, the features of items, and the user-item past interactions [1]. The goal of the recommendation system is to induce the relationship between these features and the behavior of the users.

However, there are some critical issues that need to be considered. A recommendation system often suffers from a high proportion of missing data. For those missing data, they are missing not at random (MNAR). In specific, when a person does not vote for a particular item, his view on that item will not be recorded. There could be all kinds of reasons that he does not make his ratings, such as his mood, his accessibility to rating, or his inclination to be a part of any rating. Thus, data missing at random (MAR) approach does not give us an unbiased result. An example can demonstrate why that is the case based on the idea that people tend to vote for their favorite items, as shown in Figure 1. The big number in the middle stands for the rating of a particular group of people. Those with green/red squares stand for those who have rated (not necessarily their true ratings), and the rest is merely a prediction. In the first column: Horror (movies). In the following example, 11 horror movie lovers voted five and 3 romance movie lovers voted one. Clearly, the outcome is severely biased by simply taking the mean of ratings.

Figure 1. Observed losses are misleading due to selection bias.

Those MNAR data make the prediction of rating much more complicated, and it is difficult to make sure the estimated performance is good enough [2]. The performance of a prediction model is shown in its prediction inaccuracy, which means the prediction errors on average like mean square error (MSE) for all ratings [3]. When estimating the prediction rating with MNAR ratings, only consider the observed ratings, like the one just shown in graph, are severely biased, then the prediction model can greatly overestimate or underestimate the prediction inaccuracy, which can be also called as the bias (Little & Rubin, 2014).

There are two approaches to deal with the MNAR problem recently: (1) The error-imputation-based (EIB) approach calculates the imputed error like prediction error estimated by each MNAR rating, but there is always a great bias because of its imputation inaccuracy, for example, deviations between the prediction errors and the imputed errors [4], which can also influence the training session and enlarge the prediction inaccuracy [5]. (2) The inverse-propensity-scoring (IPS) approach weights each observed rating's prediction error with its propensity in an inverse way [6], but it is always influenced by the high variance of the propensities [7], and once the propensities get inversed, the training losses will be unstable, and the ability of generalization will be poor.

Rather than those recent approaches, a doubly-robust estimator is proposed, which could combine the advantages of those recent approaches. That is, it calculates the imputed errors and inversely weighs each prediction error with its propensity to reduce the bias of prediction inaccuracy. To minimize bias, the joint learning of a prediction model and an imputation model is proposed. The imputation model trains to obtain two hyper-parameter values that minimize imputation error after the prediction model makes its prediction, while the prediction model learns to decrease the prediction errors according to the imputation model using the imputation error that was just generated. In this way, the prediction model and imputation model gradually regularize each other to decrease prediction inaccuracy and imputation inaccuracy [8].

This paper aims to discuss three methods to deal with MNAR, which include error-imputation-based model, inverse-propensity-score model, and Doubly Robust Method. After comparing experiments using these methods, it will be clear that the Doubly Robust method produces an outcome with less bias compared to other recent approaches.

2. Preliminaries

2.1. Imputation
The way the error-imputation-based reduce bias is to minimize the imputed errors. Here, the imputed error is the deviation of the imputation from the actual value.
Imputed error could be calculated by

$$\left(\omega\left|r_{u,i}-\gamma_{u,i}\right)\right|(AbsoluteError)\right) \quad (1)$$

or

$$\omega\left(r_{u,i}-\gamma_{u,i}\right)^{2} \; (SquaredError) \quad (2)$$

Where $r_{u,i}$ is the true rating of ω and γ are hyper-parameters? The specific equation for calculation could be expressed as the following.

2.2. IPS-Propensity Model
The other common method to reduce bias is Inverse-propensity-method. This method aims to obtain the propensity.

$$p_{u,i}=P\left(O_{u,i}=1\middle|X,X_{hid},Y\right) \quad (3)$$

Propensity represents the probability of observing the interaction between a user and an object. This method inversely weights the obtained propensity so that the objects with higher propensity will have lower prediction inaccuracy. The detailed equation is:

$$\varepsilon_{IPS}\left(\hat{R},R^{O}\right)=\frac{1}{|D|}\sum_{u,i\in D}\frac{o_{u,i}e_{u,i}}{\hat{p}_{u,i}} \quad (4)$$

To determine the propensity, there are two methods.

2.2.1. Naïve Bayes. The first way is Naïve Bayes, which uses simple probability calculation to obtain the propensity. It is assumed that the dependency between covariates X, X_{hid}, Y could be ignored since the data is MNAR. According to Naïve Bayes Theory:
Notice that $P(Y=r \mid O=1)$ and $P(O=1)$ could be observed, while the value of $P(Y=r)$ requires MAR data.

2.2.2. Logistic Regression. The second way is Logistic Regression, which aims to determine model parameter so that

$$P\left(O_{u,i}\left|X,\,X_{hid},Y\right)=P\left(O_{u,i}\right|X,\,\phi\right) \quad (5)$$

The model assumes the existence of $\phi=(\omega,\beta,\gamma)$ such that

$$P_{u,i}=\sigma\left(\omega^{T}X_{u,i}+\beta_{i}+\gamma_{u}\right) \quad (6)$$

Where σ represents a sigmoid function, and (β_i,γ_u) represents per-item and per-user offsets.
In the later experiment, we decided to use the logistic regression method as it is more intuitive and convenient to compute the propensities.

3. Joint Learning
One of the problems encountered in the training of the recommendation system model using the doubly robust predictor is that the imputation model often forms a large error deviation, which deteriorates the training of the prediction model. To prevent this problem, a method called joint learning is used to train the imputation model and the prediction model.
The main algorithm of this method is to train the imputation model and the prediction model respectively in the same loop. First, the prediction model needs to formulate an initial hyperparameter θ to predict an initial prediction rating $\hat{r}_{u,i}$ and use it to get the initial predict error $e_{u,i}$ as a constant. Then use initial $\hat{r}_{u,i}$ and $e_{u,i}$ in imputation model and training the model with loss function:

$$L_{e}(\theta,\varphi)=\sum_{u,i\in O}\frac{(\hat{e}_{u,i}-e_{u,i})^{2}}{\hat{p}_{u,i}}+v||\varphi||^{2}{}_{F} \quad (7)$$

to update hyperparameter φ by taking the derivative. So that the imputed error $\hat{e}_{u,i}$ will be updated as well. Then use the updated imputed error $\hat{e}_{u,i}$ in prediction model as a constant and training the model with loss function:

$$L_r(\theta,\varphi)=\sum_{u,i\in D}\hat{e}_{u,i}+\frac{o_{u,i}(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}+v||\theta||^2{}_F \tag{8}$$

to update the hyperparameter θ, so that the predict rating $\hat{r}_{u,i}$ and predict error $e_{u,i}$ will be updated as well. Then put this updated information into $L_e(\theta,\varphi)$ and repeat the above steps until optimal loss is achieved.

The doubly robust method ensures the unbiased prediction mathematically. The formula of doubly robust loss function is:

$$L_{DR}(\theta,\varphi)=\frac{1}{D}\sum_{u,i\in D}[\hat{e}_{u,i}+\frac{o_{u,i}(L_r(\theta)-\hat{e}_{u,i})}{\hat{p}_{u,i}}] \tag{9}$$

$$E[L_{DR}(\theta,\varphi)]=E[L_r(\theta)+\frac{(o_{u,i}-\hat{p}_{u,i})(L_r-\hat{e}_{u,i})}{\hat{p}_{u,i}}$$
$$=E[L_r(\theta)+\frac{(p_{u,i}-\hat{p}_{u,i})(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}] \tag{10}$$

If true propensity equals predicted propensity or true predicted error equals imputed error:

$$\frac{(p_{u,i}-\hat{p}_{u,i})(e_{u,i}-\hat{e}_{u,i})}{\hat{p}_{u,i}}=0 \tag{11}$$

In short, the expected value of this doubly robust method will be equal to the expected value of the prediction model when either ($p_{u,i}$ = $\hat{p}_{u,i}$ or $e_{u,i}$ = $\hat{e}_{u,i}$). In other words, this doubly robust method will be unbiased if either the imputation model or the propensity model is unbiased.

4. Experiment

4.1. Data Introduction
Unbiased estimation of our prediction inaccuracy needs MAR ratings, as we decided to adapt real world dataset as follows.

4.2. Coat Shopping Dataset. [6]
The dataset simulating MNAR data of customers shopping for a coat in an online store including 6,960 MNAR and 4,640 MAR ratings of 290 users to 300 coats, therefore, 87,000 possible user-items pairs. In this case, MNAR ratings will be adapted for training and MAR ratings for testing.

4.3. Experiment Setup
For feature engineering, Auto-Encoder will be conducted to reduce dimensions to improve the generalization ability of the model [9]. Given that the feature for coats contains gender (2), type (16), color (13), promotes (2), 33 columns in one-hot encoded form, while features for user contains 14 columns of features. In this case, we use Auto-encoder as an approach to avoid the problem of having a sparse feature matrix. As a result, reducing coat features of type to 10 and coat features of colors to 7 is possible, and concatenated to form a feature matrix for user item pairs with 30 features.

4.4. Logistic Regression on Propensities
A standard regularized logistic regression [10] was trained using all pairs of user and item covariates as features and cross-validated to optimize log-likelihood of the self-selected observations. As a result, a matrix of propensities for all user item pairs with a score of 0.92 can be constructed.

4.5. Prediction Model
Factorization Machines (FM) is a flexible and powerful modeling framework for collaborative filtering recommendation [11]. To be more specific, Factorization Machines (FM) is a generic supervised learning model that map arbitrary real-valued features into a low-dimensional latent factor space and can be applied naturally to a wide variety of prediction tasks including regression, classification, and ranking. In our case, FM will be used as a parametric model using combined features of user-item pairs to predict ratings.

4.6. Imputation Model

As above, the Inverse-Propensity-Scoring model is an alternative of EIB model by computing the propensity that the probability of observation is the true rating. In this project, the approach of alternating between training the prediction and imputation models via stochastic gradient descent [12] was being implemented. Given the property of Doubly Robust Estimator, the computation of the prediction error by the difference (instead of the absolute or squared difference) was being carry out, so the imputation model can learn to distinguish whether a predicted rating is larger or smaller than the true rating. As mentioned in the joint learning section, the loss function for stochastic gradient descent is given by prediction error, observed error and learned propensities.

4.7. Joint Learning

In this project, the dataset of MNAR ratings for training and MAR for testing on COAT is being used.

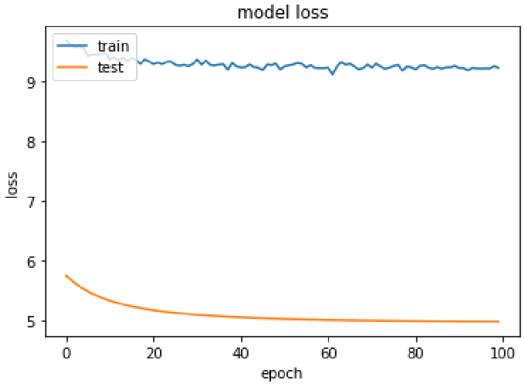

Figure 2. model loss of Doubly Robust Factorization Machine, which is under the definition of formula (9)

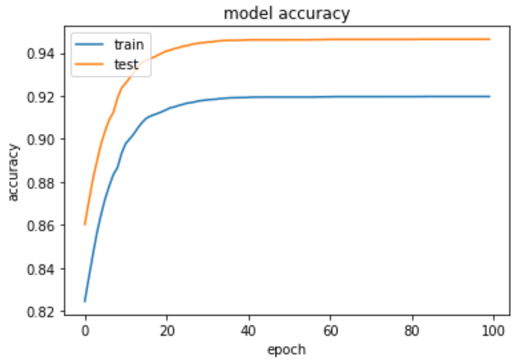

Figure 3. Model accuracy of Doubly Robust Factorization Machine

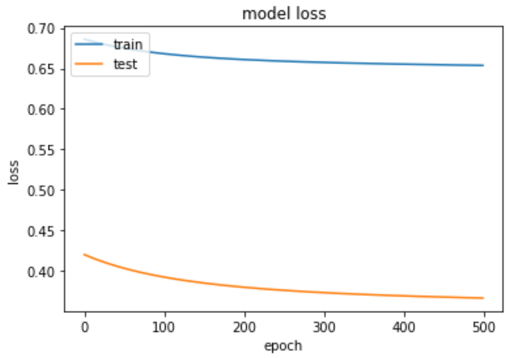

Figure 4. model loss of Naïve Factorization Machine

Table 1. Loss Value for different models

	MSE	MAE
MNAR Naïve FM	3.645	1.513
MNAR DR	3.183	1.432
Test Naïve FM	1.547	1.191
Test DR	**1.244**	**1.038**

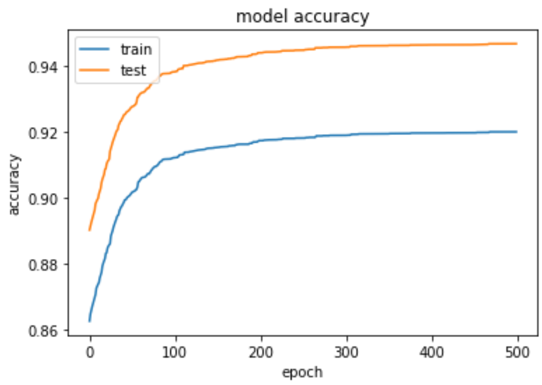

Figure 5. model accuracy of Naïve Factorization Machine

The Testing set has higher accuracy and lower loss than the Training set (as shown in Figure 2, Figure 3, Figure 4, Figure 5) because the Testing set is MAR

The reason why the doubly robust method has a higher nominal loss than that of Naïve FM (not reflected clearly in Figure 2 and Figure 4, be shown in Table 1) is that the work established a unique loss function for each model. The effectiveness of the model is reflected by the speed and performance of the model's accuracy and the convergence of its loss. It is obvious that the Doubly Robust Factorization Machine outperforms the Naïve Factorization Machine model.

5. Conclusion

In this work, an effective and robust approach to figure out the rating of missing not at random data for recommendation system is being designed. First, a doubly-robust estimator was proposed, in which way to estimate the prediction inaccuracy by using imputed errors and propensities. Then, a joint learning approach based on the doubly-robust estimator was being constructed, it learns rating prediction and error imputation in a joint learning way to make sure a low prediction inaccuracy. Additionally, compared experiment according to the real-world dataset was being completed in this work, and the result shows that our approach performs better than those commonly used ones since the estimator just designed significantly reduces the bias of estimating the prediction inaccuracy.

References

[1] Zhang, S., Yao, L., Sun, A. (2017) Deep learning based recommender system: A survey and new perspectives. arXiv preprint arXiv:1707.07435.

[2] Steck, H. (2011) Item popularity and recommendation accuracy. In: RecSys. pp. 125-132.

[3] Salakhutdinov, R., Mnih, A., and Hinton, G. (2007) Restricted boltzmann machines for collaborative filtering. In: ICML. pp. 791-798.

[4] Dudík, M., Langford, J., and Li, L.(2011) Doubly robust policy evaluation and learning. In ICML, Bellevue, Washington.

[5] Steck, H.(2013) Evaluation of recommendations: rating-prediction and ranking. In: RecSys, Los Gatos, California.
[6] Schnabel, T., Swaminathan, A., Singh, A., Chandak, N., and Joachims, T. (2016) Recommendations as treatments: Debiasing learning and evaluation. In: ICML, New York City.
[7] Thomas, P. and Brunskill, E. (2016) Data-efficient off-policy policy evaluation for reinforcement learning. In ICML. pp. 2139-2148.
[8] Wang, X., Zhang, R., Sun, Y., & Qi, J. (2019, May). Doubly robust joint learning for recommendation on data missing not at random. In ICML. pp. 6638-6647.
[9] Hinton, G., & Salakhutdinov, R. (2006). Reducing the Dimensionality of Data with Neural Networks. Science, 313(5786): 504-507.
[10] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cour- napeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011) Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12: 2825–2830.
[11] S. Rendle. (2010) Factorization Machines. IEEE International Conference on Data Mining, 995-1000.
[12] Bottou, Léon (1998). Online Algorithms and Stochastic Approximations. Online Learning, 5: 6-3.