Imputation of Missing Financial Data

Name: Xueye Ping
SUNet ID: xueye
Department of MS&E
Stanford University
xueye@stanford.edu

Name: Satita Vittayaareekul
SUNet ID: satita97
Department of SCPD
Stanford University
satita97@stanford.edu

Name: Zhengdan Li
SUNet ID: zhengdan
Department of Statistics
Stanford University
zhengdan@stanford.edu

1 Introduction

Missing data is a common practical problem often encountered in data collection. It is prevalent in virtually any area and the finance industry is no exception. Actually, in asset pricing studies, missing covariates like firm characteristics are frequently observed. However, little attention has been paid to this problem and most studies simply use some naive methods to deal with this problem. One common approach is to exclude a whole record if one or more covariates are missing. This would significantly reduces the amount of data that can be used. For example, in this project, the original dataset we are using contains over 3 million records. But approximately 2 million of them have more than five characteristics missing and only 0.6 million records have complete data of all the 45 characteristics. Furthermore, some of the dropped observations may contain crucial information related to the asset return. Another straightforward way is to use the cross-sectional mean or median for imputation. This is proved to be highly biased due to the inconsistency in estimates and the inaccuracy in standard deviations [1]. As an increasing number of researchers start to use these covariates to predict future returns cross-sectionally, it becomes imperative to handle this problem more properly.

In this project, we aim to find a better approach for imputing financial data via machine learning techniques, so that more complete data can be used for further analysis. To be more specific, we are going to use firm characteristics as our imputation target. Some important characteristics include book-to-market (B2M), operating profitability (OP), investment (INV), and leverage (LEV). We mainly try four different algorithms to impute missing data in these firm characteristics: Generative Adversarial Imputation Nets (GAIN), Variational Autoencoder (VAE), KNN impute algorithm, Random Forest Imputation (MissForest).

2 Related Work

There is vast literature on missing data imputation from a general perspective. Not many paper address it explicitly in finance area. The most widely used approaches in the related work are the two mentioned in the introduction: 1) cross-sectional mean or median [2][3], 2) only using complete data [4] [5] [6]. Although cross-sectional mean or median is easy to compute, the methodology is relatively simple and usually does not lead to a good performance in the downstream prediction or classification task, especially when the proportion of missing data is large in the full dataset. The method of using only the fully observed cases often affects the sample size and also the performance of future tasks. Moreover, some important predictors such as size might be abandoned during this process [1].

3 Dataset and Features

The data of the 45 firm characteristics we use are from the missing financial data paper [7], which are derived from fundamental firm-level information in the Compustat database and the Center for Research in Security Prices (CRSP) database. The definitions of these 45 characteristics can be found in the appendix of that paper [7]. Besides, we also extract the data of corresponding stock return from that dataset. The remaining three variables: date, company code and update frequency are ignored as they are not significantly relevant here. The dataset contains data from January 1967 to December 2020. The size of the original dataset is about 3.6 million. Before training the model, we performed data preprocessing on the dataset. Only those complete records are kept since we need the real data to test the model's performance. Thus, we dropped all rows containing null values in any column. The size of the dataset we finally use is 0.6 million.

4 Methodology

4.1 Generative Adversarial Imputation Nets (GAIN)

GAIN is a variation of the well-known Generative Adversarial Nets (GAN), where convolutional neural networks are used in the discriminator and generator[8]. Different from the concepts of GAN, the generator G of GAIN aims to accurately impute missing entries in the dataset, while the discriminator D attempts to distinguish between observed and imputed data. A generative model and an adversary are trained in parallel using backpropagation with a minimax criterion. In other words, the discriminator is trained to minimize the classification error and the generator is trained to maximize the discriminator's classification error. The mask M is a binary matrix where ones indicate observed data and zeros indicate missing data. In addition, GAIN introduces a hint mechanism H on top of the GAN framework, where the hint H reveals partial information about the missingness M to the discriminator to ensure G generates according to the true distribution. Suppose X and $\hat{X}$ denote the full data and the imputed values, respectively. Then, the objective function of GAIN is defined as

$$\text{min}_G\text{max}_D V(D,G)$$

, where

$$V(D,G) = \mathbb{E}_{\hat{X},M,H}[M^T logD(\hat{X},H) + (1-M)^T log(1-D(\hat{X},H))]$$

4.2 Variational Autoencoder (VAE)

Variational Autoencoder (VAE) is another generative model we used to analyze the accuracy of imputing missing data [9]. It is a type of likelihood-based generative model. VAE takes the input data and encodes into μ and σ, which is used to help decoder to distinguish between similar data. Then μ and σ combine to create a latent space, and uses reconstruction loss and KL-Divergence to train better μ and σ. Further, its decoder reconstructs its latent space back to the input data. VAE, as a generative model, generates new data by sampling from its continuous latent space and feeds it to the decoder. Its ability to generate analogous new data from the input training data is all attributed to its mapped distribution of latent space.

4.3 KNN impute algorithm

The KNN-based imputation algorithm imputes the missing values according to their K nearest neighbors, which was developed by Troyanskaya et al [10]. To be more specific, if we consider record 1 has one missing value in feature A. Then this method would find K records from the remaining data, which have a value present in feature A and share similarities with record 1 in the other N-1 features (N denotes the total number of features). Finally, the imputed value of feature A in record 1 is taken as the weighted average of the values for feature A in these K records, where the weights depend on their similarities with record 1. In addition, Euclidean distance is used here as the measure for similarity based on [10].

4.4 Random Forest Imputation (MissForest)

MissForest is an iterative imputation method based on Random Forest [11]. At first, an initial guess is made for the missing values using some basic imputation methods such as mean imputation. Then, the features are sorted according to the number of missing values in ascending order. Iteratively, each feature is fitted with a Random Forest model and then the missing values of the corresponding feature are predicted using the fitted model. The stopping criterion is based on the difference of the imputed data matrices over two consecutive iterations.

For example, if feature A has the least number of missing values, then the algorithm would start from the column corresponding to feature A. For now, all columns except the column of feature A have complete data since we initially set the missing values of other columns to be the column mean. A Random Forest model is then fitted with feature A as the response and the other N-1 features as the predictors. The missing values of feature A is predicted using the trained model afterwards. Then the same procedure starts again on the feature that has the second least number of missing values.

5 Experiments

5.1 Experimental Setup

5.1.1 Data Processing and Performance Metrics

After we removed all missing data rows from the original dataset, we randomly masked out data fields by the percentage of missing rate we chose, and later used the Root Mean Squared Error (RMSE) metrics to measure the difference between our predicted data and the actual financial data. We chose to use RMSE because it is more straightforward compared to Mean Squared Error (MSE) and penalizes large errors better. RMSE is defined as

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\Big(\frac{d_i - f_i}{n}\Big)^2}$$

where n = number of non-missing data points, d_i = expectation, and f_i = prediction.

5.1.2 Hyperparameter Tuning

- In GAIN, we used grid search and chose hint rate to be 0.9, batch size to be 128, and iterations to be 10000.
- In VAE, we chose hint rate to be 0.9, batch size to 10, latent size to 20, and iterations to 100 due to limited computational resources.
- In KNN, n_neighbors (the number of neighboring samples to use for imputation) is chosen as default value 5.
- In MissForest, due to computational cost, we set n_estimators (the number of trees) to be 10.

5.2 Main Experiments

We experimented on different sample sizes, different missing rates, and used LASSO to predict on imputed data for all four models we compared with.

5.2.1 Varying Missing Rates

Little and Rubin (2002) suggested that a missing rate less than 20% is acceptable [12]. And Rubin, D. B. (1999) suggested that a missing rate of less than 5% should be ignored since it won't make a significant effect in terms of analyzing [13]. Bennett DA. (2001) concluded that when the missing rate of a dataset is more than 10%, a bias is more likely to occur in the estimates [14]. Thus, we set 10% as our default missing rate, and experimented with values ranging from 20% to 80%.

The result shows that the smaller the missing rate, the more accurate our prediction. For VAE, we noticed that it is impossible to impute data for more than 25% of missing rate, the Root Mean Squared Error is either too high or unable to predict due to all rows having at least a field of missing data.

From all four models we compared, MissForest has the lowest Root Mean Squared Error from all missing rate we tested. The increase in the error of MissForest followed a clear pattern.

5.2.2 Varying Sample Sizes

Our pre-processed data has 630k rows total, we reduce by 100k per sample size experiment and default the missing rate to be 10% for each model. Overall, the result met our original expectation that the larger the dataset is, the better the imputation performance is. It is observable from Figure 1 that the performances of VAE and KNN were insensitive to the change in the number of samples. On the contrary, the RMSE of MissForest decremented steadily following an obvious trend as the sample size increases. In addition, MissForest outperformed the rest models with RMSE smaller than 0.15. However, the performance of the GAIN model was not as good as that in the original paper where GAIN performs dominantly well compared to each benchmark, potentially because financial data belongs to a category different from the datasets used in the original paper. The sample size used in the original paper was also smaller than the size of our financial dataset.

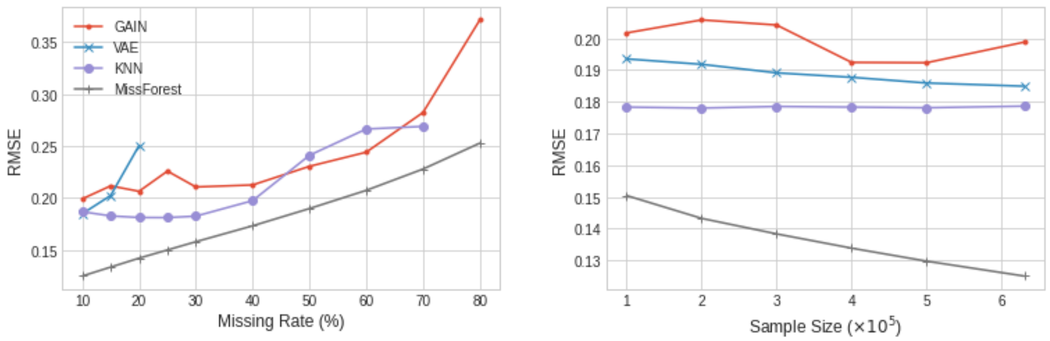

Figure 1: Root mean squared error for four imputation methods in different experiment settings: (a) various missing rates, (b) various sample sizes

5.3 Prediction Performance

Furthermore, we experimented with the imputational results from the four models to investigate how the imputed data perform for the following prediction task. LASSO was used to build regression models to predict the stock return in the dataset, which was left out during the imputation process above to preserve its true values. The remaining characteristics in our dataset were used as explanatory variables. In addition to the four sets of data obtained using various imputation methods, a LASSO regression model was also built based on our original dataset (true values) as a baseline. LASSO is formulated from linear regression with a regularization on the weight. Similar to the imputation task, RMSE was used as a loss function to measure the model performance. We utilized the LASSO from the scikit-learn package and cross-validation with 5 folds. To choose the value of the hyperparameter α that controls the weight of penalty to the loss function, grid search was applied to find an α among $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 5, 10\}$. For all five LASSO models, $\alpha = 0.001$ was selected as the regularization parameter that yielded an optimal error value.

As shown in Figure 2, the five models including the one built from original data all had root mean squared errors around 0.11 with small discrepancies. The smallest RMSE was obtained from the original data with a value of 0.1038. In terms of the feature importance, we can observe from Figure 3 that, although some models selected extra features, all LASSO models selected at least four variables: **R2_1** (short-term reversal), **IdioVol** (Idiosyncratic volatility), **SUV** (standard unexplained volume), and **VAR** (variance of daily returns in the past two months). It is also noticeable that **R2_1** is dominant among the four features. Data imputed using MissForest yielded important features most similar to that from the original dataset. Therefore, the four imputation methods all achieved a relatively good performance when applied to a downstream prediction task.

6 Conclusion

To address the problem of missing data in the financial area, we trained models using four imputation methods, GAIN, VAE, KNN, and MissForest, aiming to accurately impute missing entries. Root mean squared error between the original observed data and imputed data was used to quantitatively compare the different methods. Among the models, MissForest achieved the best performance in both sets of experiments varying missing rates and sample sizes, respectively. We further trained LASSO regression models from the result datasets to simulate how the imputed data can be used in a downstream prediction task. While all imputed data led to a good prediction error rate, the features selected from LASSO model trained using MissForest-imputed data are most close to those

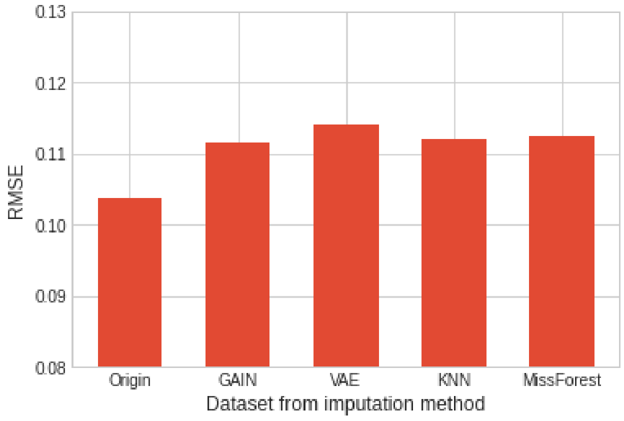

Figure 2: LASSO prediction using datasets from different imputation methods

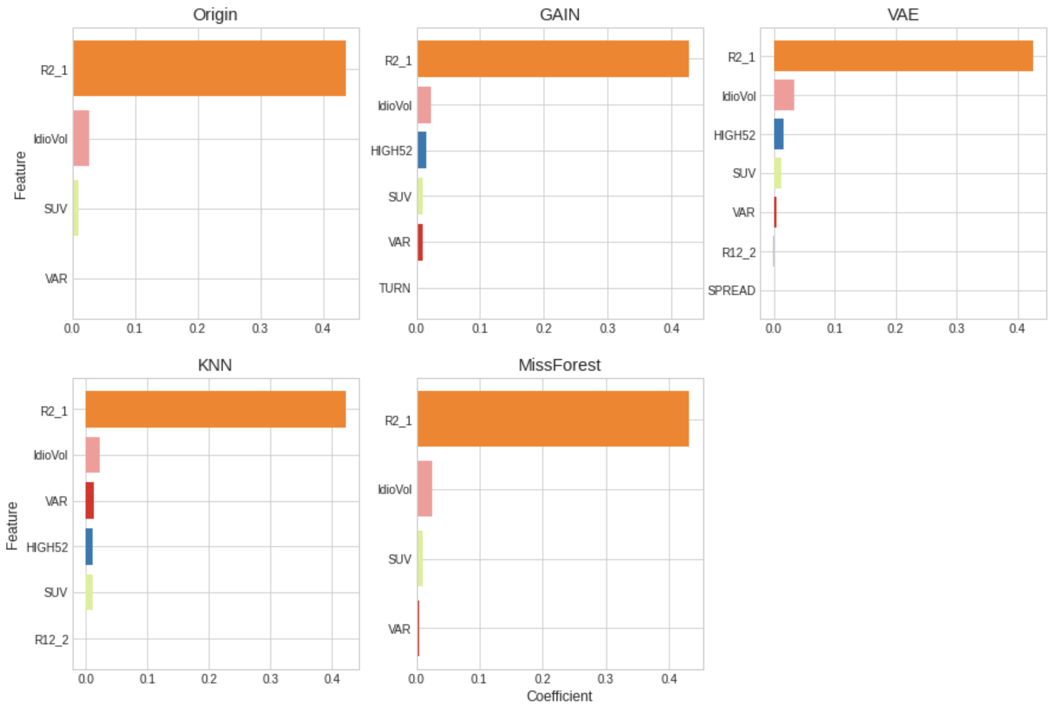

Figure 3: Feature importance of LASSO models from different imputed datasets

from the observed true data. If more computational resources were available, we could try larger n_estimators for MissForest and this may bring even better performance. It could be useful as well to place less wight on imputed observations in the prediction part. Another potential line of future work is to include the time information in the dataset considering that most financial data exhibit strong auto-correlation. For example, we can incorporate some auto-regressive structures into the current model.

7 Contribution

Xueye Ping contributed to the implementation of MissForest and KNN impute algorithm. Satita Vittayaareek contributed to the coding part of VAE. Zhengdan Li contributed to the implementation of the GAIN model. We all contributed to data processing and final report writing.

References

[1] Joachim Freyberger, Björn Höppner, Andreas Neuhierl, and Michael Weber. Missing data in asset pricing panels. *Available at SSRN*, 2021.

[2] Nathaniel Light, Denys Maslov, and Oleg Rytchkov. Aggregation of information about the cross section of stock returns: A latent variable approach. *The Review of Financial Studies*, 30(4):1339–1381, 2017.

[3] Serhiy Kozak, Stefan Nagel, and Shrihari Santosh. Shrinking the cross-section. *Journal of Financial Economics*, 135(2):271–292, 2020.

[4] Joachim Freyberger, Andreas Neuhierl, and Michael Weber. Dissecting characteristics nonparametrically. *The Review of Financial Studies*, 33(5):2326–2377, 2020.

[5] Bryan T Kelly, Seth Pruitt, and Yinan Su. Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics*, 134(3):501–524, 2019.

[6] Soohun Kim, Robert A Korajczyk, and Andreas Neuhierl. Arbitrage portfolios. *The Review of Financial Studies*, 34(6):2813–2856, 2021.

[7] Svetlana Bryzgalova, Sven Lerner, Martin Lettau, and Markus Pelger. Missing financial data. 2022.

[8] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. Gain: Missing data imputation using generative adversarial nets, 2018.

[9] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[10] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

[11] Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

[12] Roderick JA Little and Donald B Rubin. Bayes and multiple imputation. *Statistical analysis with missing data*, pages 200–220, 2002.

[13] DB Rubin. Multiple imputation: a primer. *Statistical Methods in Medical Research*, 8(1):3–15, 1999.

[14] Derrick A Bennett. How can i deal with missing data in my study? *Australian and New Zealand journal of public health*, 25(5):464–469, 2001.

Code Reference

GAIN: https://github.com/jsyoon0823/GAIN
VAE: https://github.com/gevaertlab/BetaVAEImputation
missingpy: https://github.com/epsilon-machine/missingpy