Group Number: 20

Group Member Name: Aniket Chougule A20552758

Sahil Bhaware A20552865

Vandan Vaishya A20552904

Research Report:

"Efficient and Effective Data Imputation with Influence Functions"

Abstract

The increasing volume of incomplete datasets poses a considerable challenge in training data imputation models due to the computational impracticality in real-world scenarios. The study introduces EDIT, an innovative data imputation system employing influence functions, which expedites training of parametric imputation models with representative samples while ensuring imputation accuracy. EDIT is composed of two core modules: Imputation Influence Evaluation (IIE) and Representative Sample Selection (RSS). IIE calculates the effect of complete and incomplete samples on the imputation model's predictions, while RSS compiles a minimal set of influential samples that meet a predefined imputation accuracy level. Additionally, a weighted loss function is integrated to concentrate the imputation model's efforts on influential samples. EDIT's efficiency is highlighted through extensive testing, where it is shown to require only about 5% of the total samples to increase training speed by an average of 4 times and improve accuracy by over 11% when compared to ten leading imputation methods.

Introduction

The escalating volumes of incomplete datasets have necessitated the development of efficient and effective imputation models to fill in missing values, a task that is critical across numerous data-driven domains. Traditional imputation methods, however, struggle to cope with the sheer size of modern datasets, resulting in a computational bottleneck. This report delves into the EDIT system, which tackles this challenge through a novel application of influence functions, optimizing both the efficiency and efficacy of the imputation process. By focusing on representative samples, EDIT drastically reduces the computational overhead involved in model training without sacrificing accuracy, paving the way for practical real-world data imputation.

Methodology

Imputation Influence Evaluation (IIE) Module

Objective: Estimate the influence of each data sample on the prediction result of imputation models.

Approach: Utilizes influence functions to quantify the effect of each (in)complete sample on the model's predictions. The process involves computing gradients and Hessian matrices, which can be resource-intensive but critical for determining the "influence power" of samples.

Representative Sample Selection (RSS) Module

Objective: Build a minimal set of high-impact samples to satisfy user-specified imputation accuracy.

Approach: Selects samples with the highest influence power scores from the IIE module. This set is then used to train the imputation model, significantly reducing the number of samples needed without compromising accuracy.

Weighted Loss Function

Purpose: To emphasize the importance of high-impact samples during the training process of the imputation model.

Implementation: Introduces a weighting scheme within the loss function during model optimization, where more influential samples are given higher weights.

Algorithmic Steps

Initialization: Set initial parameters, including the learning rate, number of iterations, and initial sample size based on dataset requirements.

Influence Power Calculation: For each sample in the dataset, compute the influence power using the IIE module.

Sample Selection: Using the RSS module, select a subset of samples with the highest influence scores.

Model Training: Train the imputation model on the selected subset using the weighted loss function to prioritize learning from samples with higher influence power.

Performance Evaluation: Assess the imputation model's accuracy using the selected subset and compare it with the required user-specified accuracy.

Iteration: If the desired accuracy is not met, adjust the sample selection, weights, and retrain the model.

Experimental Setup

Datasets Used: Experiments were conducted on various datasets like Power, Gas, HIGGS, and Criteo to validate the model's effectiveness.

Benchmarking: Compared EDIT's performance against ten state-of-the-art imputation methods, both traditional (e.g., MICE) and deep learning-based (e.g., GAIN, GINN).

Hyperparameters: The learning rate, dropout rate, training epochs, and batch sizes were set specific to the nature of the dataset and the imputation method employed.

Experiments and Results

The experimental evaluation involved four real-world datasets: Power, Gas, HIGGS, and Criteo, using metrics like training time, RMSE, and training sample rate. EDIT

significantly outperformed ten benchmark imputation methods in all metrics. Notably, EDIT required only about 5% of samples for model training, sped up the process by an average of 4x, and improved accuracy by more than 11%.

Table 2 in the paper showcases a comprehensive comparison of imputation performance on the Power dataset, clearly demonstrating EDIT's superior performance over the original methods across various metrics, such as RMSE and training time. Figure 4 illustrates the robustness of EDIT under different missing rates, indicating its consistent outperformance in terms of RMSE and training efficiency compared to traditional methods.

Conclusion

The findings from this study indicate a potential paradigm shift in data imputation, particularly for scenarios with vast volumes of missing data. EDIT's framework for rapidly training parametric imputation models under accuracy guarantees, with minimal sample requirements, presents a promising solution for real-world applications where computational efficiency and accuracy are paramount.

Acknowledgments

The research received support from several grants, including the Zhejiang Provincial Natural Science Foundation, the National Natural Science Foundation of China, and others, indicating the importance and collaborative nature of this research work.

For an exhaustive understanding, readers are encouraged to refer to the paper directly for detailed results, methodologies, and discussions.

Figures and Tables

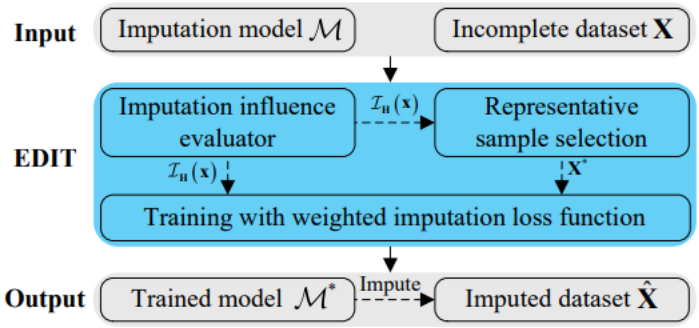

Figure 1: The architecture of EDIT

Table 2: Imputation performance comparison on *Power*

Method		RMSE (Bias)	Time (s)	R_t (%)
MissF	Original	–	–	–
	EDIT	0.240 (± 0.060)	34,362	15.63
Baran	Original	–	–	–
	EDIT	0.212 (± 0.022)	96,261	8.98
MICE	Original	–	–	–
	EDIT	0.233 (± 0.029)	28,046	14.90
DataWig	Original	–	–	–
	EDIT	0.182 (± 0.021)	24,231	6.87
RRSI	Original	–	–	–
	EDIT	0.158 (± 0.014)	18,123	**2.12**
MIDAE	Original	–	–	–
	EDIT	0.218 (± 0.041)	21,012	10.26
VAEI	Original	–	–	–
	EDIT	0.233 (± 0.015)	12,682	**2.12**
HIVAE	Original	0.119 (± 0.056)	3,504	100
	EDIT	**0.116 (± 0.024)**	**1,573**	3.88
GINN	Original	–	–	–
	EDIT	0.220 (± 0.098)	22,621	5.34
GAIN	Original	0.183 (± 0.038)	2,909	100
	EDIT	0.163 (± 0.042)	1,794	8.26

Table 3: Imputation performance comparison on *Gas*, *HIGGS*, and *Criteo*

Method		*Gas*			*HIGGS*			*Criteo*		
		RMSE (Bias)	Time (s)	R_t (%)	RMSE (Bias)	Time (s)	R_t (%)	RMSE (Bias)	Time (s)	R_t (%)
HIVAE	Original	0.182 (± 0.089)	32,291	100	**0.404 (± 0.031)**	54,721	100	–	–	–
	EDIT	**0.169 (± 0.067)**	6,872	2.81	0.412 (± 0.063)	35,621	4.22	**0.324** (± 0.063)	94,291	**3.96**
GAIN	Original	0.241 (± 0.037)	17,548	100	0.448 (± 0.092)	44,044	100	–	–	–
	EDIT	0.213 (± 0.036)	**6,199**	**2.43**	0.423 (± 0.125)	**29,443**	**2.91**	0.366 (± 0.045)	**85,683**	4.22

Table 4: Imputation performance comparison (RMSE) of Random, noW-EDIT−, and noW-EDIT

Method		*Power*	*Gas*	*HIGGS*	*Criteo*
HIVAE	Random	0.141 (± 0.034)	0.187 (± 0.058)	0.433 (± 0.064)	0.345 (± 0.070)
	noW-EDIT-	0.149 (± 0.042)	0.194 (± 0.067)	0.444 (± 0.058)	0.359 (± 0.061)
	noW-EDIT	**0.120 (± 0.024)**	**0.175 (± 0.060)**	**0.420 (± 0.063)**	**0.329 (± 0.063)**
GAIN	Random	0.184 (± 0.038)	0.243 (± 0.045)	0.452 (± 0.094)	0.386 (± 0.032)
	noW-EDIT-	0.192 (± 0.043)	0.252 (± 0.032)	0.460 (± 0.099)	0.397 (± 0.038)
	noW-EDIT	**0.173 (± 0.042)**	**0.221 (± 0.036)**	**0.425 (± 0.102)**	**0.375 (± 0.045)**

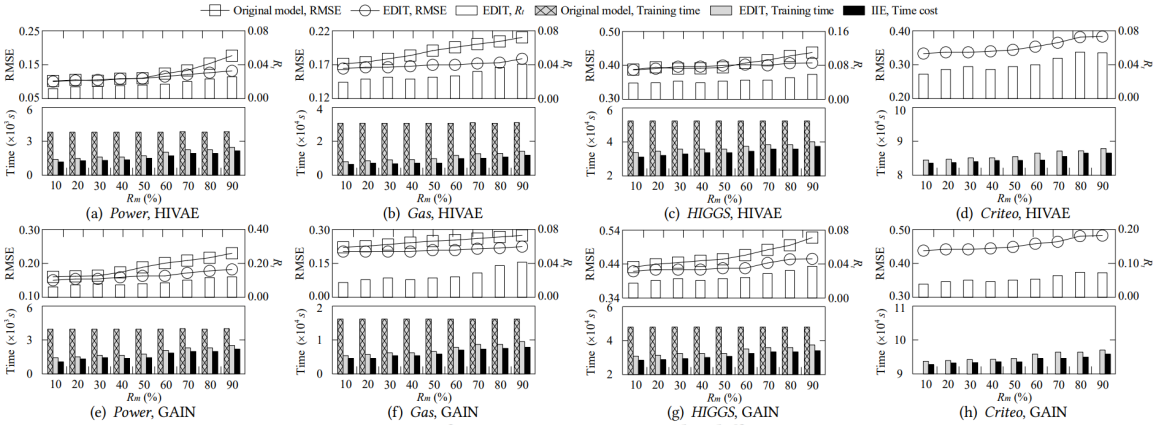

Figure 4: Imputation performance comparison under different missing rates

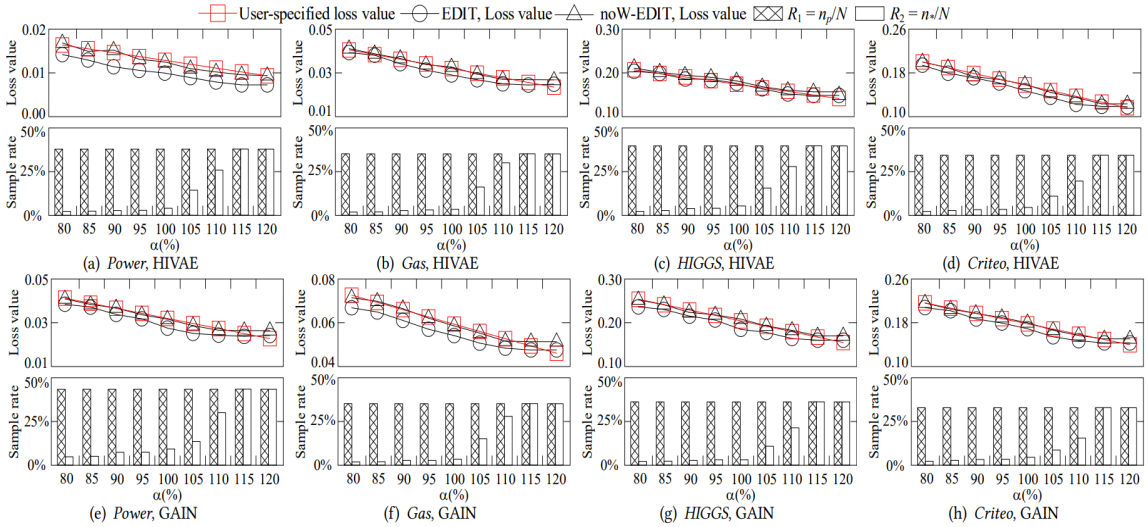

Figure 5: Imputation performance comparison under different values of α

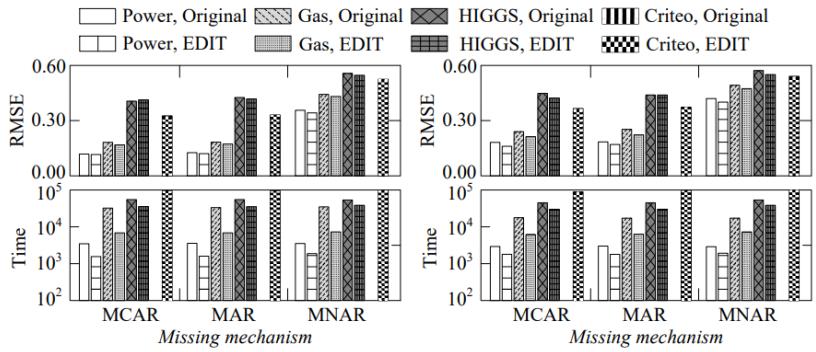

Figure 6: Comparison under different missing mechanisms