ENHANCED MULTI-TASK LEARNING OF IMPUTATION AND PREDICTION VIA FEATURE RELATIONSHIP GRAPH LEARNING

Anonymous authors

Paper under double-blind review

ABSTRACT

Missing values present significant challenges in machine learning, often degrading predictive performance. Traditional and deep learning imputation methods often overlook the relationships between features and their connections to downstream tasks. To address these gaps, we propose PIG (multi-task learning of Prediction and Imputation via feature-relationship Graph learning), a model that integrates imputation and prediction by leveraging feature interdependencies. PIG utilizes a graph-based approach to capture intricate feature relationships, thereby enhancing the accuracy of both imputation and downstream tasks. Our strategic training process begins with pre-training for both tasks, ensuring the model learns effective representations. This is followed by fine-tuning the entire model to further optimize imputation and downstream tasks simultaneously. We evaluated our method using nine benchmark datasets, three for regression and six for classification. Our method showed superior imputation and prediction performance across nine datasets, achieving an average rank of 1.33 for both imputation and regression tasks and 1.83 for imputation and 1.17 for classification tasks. Additionally, in sensitivity analysis with respect to missing rates, our method demonstrated its robustness, especially in predictive performance, compared to other methods that showed significant degradation.

029

006

008 009 010

011 012 013

014

015

016

017

018

019

021

025

026

027

028

031 032

033

1 INTRODUCTION

034 Missing values arise from various factors across diverse fields (Kim et al., 2018; Yoon et al., 2016; Li et al., 2021; Koren et al., 2009), underscoring the importance of effective imputation. Tradi-035 tionally, basic statistical methods, such as mean, median, or mode, have been used for imputation; however, these approaches often yield inaccurate results. Matrix completion methods, commonly 037 employed in recommendation systems, have demonstrated significant improvements in missing data imputation (MDI) (Koren et al., 2009; Hu et al., 2008; Rendle et al., 2014; Mazumder et al., 2010), yet they have limitations in handling categorical features. Recently, deep learning methods have 040 been applied to MDI using AutoEncoders (AEs) and their variants (Gondara & Wang, 2018; Vin-041 cent et al., 2008), enabling flexible processing of mixed-type data that includes both numeric and 042 categorical features. Furthermore, some studies have utilized generative models, such as Variational 043 AutoEncoders (VAEs) and Generative Adversarial Networks (GANs), to capture the underlying data 044 distribution and thereby generate plausible values for missing data (Nazabal et al., 2020; Yoon et al., 2018).

Despite the advancements offered by the aforementioned methods, they exhibit notable limitations.
First, all these approaches focus solely on the imputation task, neglecting the downstream tasks,
which are the ultimate goal of the analysis. This oversight can result in sub-optimal performance in
downstream tasks. Second, while deep learning methods have demonstrated competitive accuracy
in imputation, they are often sensitive to irrelevant features that are commonly present in real-world
datasets (Grinsztajn et al., 2022). This sensitivity can lead to overfitting to noisy patterns of irrelevant
features, hindering the generalization of imputation performance. Consequently, these limitations
highlight a critical gap in both imputation and downstream tasks, necessitating the development of

More recent methods, such as GRAPE (You et al., 2020), have demonstrated the potential of Multi-Task Learning (MTL) to address both imputation and downstream tasks. Additionally, by transforming tabular data into a bipartite graph structure and using Graph Neural Networks (GNNs), GRAPE
implicitly captures feature relationships through edge-level prediction tasks for imputation, while
downstream tasks are framed as node-level predictions. However, despite its strong performance,
GRAPE still has limitations, such as an indirect representation of feature relationships through partially observed data and challenges related to stability during MTL (Liu et al., 2021; Yu et al., 2020; Chen et al., 2018).

062 To this end, we propose PIG, multi-task learning of Prediction and Imputation via feature-063 relationship Graph learning. PIG incorporates a Graph Learning Layer (GLL) that explicitly learns 064 feature relationships, which are then fully utilized in a Graph Residual Imputation Layer (GRIL) to consider only relevant features in imputation and thereby enhance its performance. Addition-065 ally, PIG enhances the stability of MTL through a carefully designed training strategy. The training 066 process begins with the pre-training of the GLL and a Prediction Layer (PL), which is designed to 067 perform downstream tasks. This is followed by fine-tuning the entire model to simultaneously opti-068 mize both imputation and downstream tasks. This approach, as demonstrated to be effective in recent 069 research (Qu et al., 2024), mitigates the complexities associated with multiple objectives, resulting in stable performance improvements, particularly in downstream tasks. The main contributions of 071 this work are as follows. 072

- We present PIG, an MTL model that integrates imputation and prediction (classification or regression) for mixed-type tabular data, leveraging their interdependencies to enhance both tasks.
 - We propose a GLL which identifies feature relationships, and a GRIL, which utilizes these relationships to optimize both imputation and downstream tasks.
- We propose a training strategy to optimize PIG for both imputation and prediction tasks, which begins with the pre-training of a GLL and a PL, followed by fine-tuning the entire model.

Through extensive experiments across diverse datasets that encompass both regression and classification downstream tasks, we rigorously evaluated the effectiveness of the proposed method. Several ablation studies further explains the significant impact of individual components within our model, thereby showing comprehensive and deliberate design of the proposed method.

2 RELATED WORK

073

074

075

076

077

078

079

081

087

880

MDI methods have evolved significantly over time, transitioning from simple statistical methods to more advanced machine learning and deep learning methods.

Statistical Methods. Early MDI methods included simple statistical methods such as mean, median,
 and mode imputation. While these methods are easy to implement, they often overlook relationships
 between features, leading to poor performance.

Machine Learning Methods. More sophisticated methods have emerged, such as multiple impu-096 tation by chained equations (MICE) (Van Buuren & Groothuis-Oudshoorn, 2011), k-nearest neigh-097 bors (KNN) (Troyanskaya et al., 2001), and matrix completion methods. MICE iteratively predicts 098 missing values using estimators like Bayesian ridge or random forests, while KNN imputes missing 099 values based on similar data points. Matrix completion methods, commonly employed in recommendation systems (Koren et al., 2009), have demonstrated strong performance in MDI by identi-100 fying and leveraging latent structures among samples and features. However, both MICE and KNN 101 can face scalability issues and may struggle with large, complex datasets, and matrix completion 102 methods are generally limited to numeric data, posing challenges with categorical variables. 103

Deep Learning Methods. Recently, deep learning methods have emerged as powerful alternatives for MDI. AEs (Gondara & Wang, 2018) and VAEs (Nazabal et al., 2020) project incomplete data into a latent space to reconstruct missing values. GAN-based methods, such as generative adversarial imputation networks (GAIN) (Yoon et al., 2018), utilize adversarial learning to generate plausible imputed data, reporting strong performance across various settings. Building on



Table 1: Summary of previous studies.

Figure 1: Overall process of PIG. The input data matrix X, with missing entries indicated by the missing indicator M, undergoes initial imputation to generate the complement matrix X. The Graph Learning Layer (GLL) refines an initial feature relationship A^{prior} into a learned relationship matrix
A. The Graph Residual Imputation Layer (GRIL) leverages A and X to produce the imputed matrix
X. Finally, the Prediction Layer (PL) uses X for downstream tasks, yielding predictions Y.

143

108

144 these advancements, GRAPE (You et al., 2020) leverages MTL to simultaneously tackle imputa-145 tion and downstream tasks. Also, by applying GNNs to the tabular data represented as a bipartite 146 graph, GRAPE implicitly captures meaningful feature relationships, enhancing imputation perfor-147 mance and facilitates improved outcomes in downstream tasks. MIRACLE (Kyono et al., 2021) is 148 proposed to preserve the causal structure of data for more accurate imputation. It refines initial im-149 putation from traditional methods by using neural networks while maintaining data dependencies. However, due to the specific purpose for data dependencies, MIRACLE suffers from performance 150 degradation in inappropriate scenarios. At last, HyperImpute (Jarrett et al., 2022), a generalized 151 iterative imputation framework, combines iterative imputation and recent powerful models; for each 152 feature, it adaptively and automatically selects proper parameters and models, ranging from simple 153 ML-based models to boosting trees and neural networks. 154

155 156

157

3 Methods

158 3.1 OVERVIEW

159

This section briefly introduces the overall process of PIG from the original data matrix with missing values to the imputed data matrix and predicted values (Figure 1). Suppose an incomplete input data matrix $X \in \mathbb{R}^{N \times d}$ and a missing indicator matrix $M \in \{0, 1\}^{N \times d}$ where N is the number of observations and d is the number of features. Each element M_{ij} equals 1 if X_{ij} is observed, and 0 if it is missing, for i = 1, ..., N and j = 1, ..., d.

Initially, missing values in X should be filled using any simple imputation method. This initial imputation usually employs statistical and ML-based methods, yielding the complement matrix $\tilde{X} \in \mathbb{R}^{N \times d}$ that contains no missing values. However, the imputed values can be further improved by utilizing informative feature relationships and considering both imputation and downstream task together.

170 Specifically, PIG uses residual learning (He et al., 2016) and feature relationships via graph con-171 volution (Kipf & Welling, 2016) to correct \tilde{X} . Then, PIG outputs the final imputation matrix 172 $\hat{X} \in \mathbb{R}^{N \times d}$, which is derived from the combination of observed values in X and imputed val-173 use for missing ones.

This final imputation matrix \hat{X} is fed to PL for downstream tasks. For simplicity, assume one downstream task, either a classification or regression. Then we can denote the target feature as $Y \in \mathbb{R}^{N \times c}$ and its predicted values as $\hat{Y} \in \mathbb{R}^{N \times c}$, where *c* is the number of classes in the classification task and 1 in the regression task.

178 179

180 3.2 THE DETAILED MODEL ARCHITECTURE

PIG has three main layers (Figure 1): GLL, GRIL, and PL. The GLL learns the feature relationships, represented as a matrix $A \in \mathbb{R}^{d \times d}$, that help both imputation and prediction. The GRIL takes \tilde{X} and A to correct the imputation residuals, then returns \hat{X} by filling in missing values of X. Then, PL takes \hat{X} and predicts targets Y in downstream tasks. Each main layer will be described in the following subsections.

186 187

188

199 200

201

202

209 210

3.2.1 GLL: GRAPH LEARNING LAYER

Motivated by graph construction in various studies (Shi et al., 2019; Li et al., 2017; Wu et al., 2020; Yan et al., 2018), we devised GLL to adaptively infer feature relationships, which are typically unknown in real-world datasets. GLL utilizes independent representations: the destination-embedding matrix $E_1 \in \mathbb{R}^{d \times d_E}$ and the source-embedding matrix $E_2 \in \mathbb{R}^{d \times d_E}$ where d_E is the dimensionality of feature embedding.

First, each embedding matrix is projected using $P_1 \in \mathbb{R}^{d_E \times d_E}$ and $P_2 \in \mathbb{R}^{d_E \times d_E}$ and activated with a scaled hyperbolic tangent function as $Z_1 = \alpha \tanh(E_1P_1)$ and $Z_2 = \alpha \tanh(E_2P_2)$ where α is a hyperparameter that controls sensitivity. Then, the feature relationship matrix A is computed by applying the sigmoid function to the inner product of the projected embeddings:

$$\boldsymbol{A} = sigmoid\left(\boldsymbol{Z}_1 \boldsymbol{Z}_2^T\right),\tag{1}$$

where each A_{ij} indicates the informativeness of feature *j* for imputing feature *i*. This relationship matrix *A* enables PIG to exploit feature dependencies for accurate imputation and prediction.

203 3.2.2 GRIL: GRAPH RESIDUAL IMPUTATION LAYER 204

GRIL takes the relationship matrix A and the complement matrix \tilde{X} as inputs and returns the final imputation matrix \hat{X} . Initially, GRIL normalizes A as $\tilde{A} = D^{-1}A$, where D is a diagonal matrix such that $D_{ii} = \sum_{j=1}^{d} A_{ij}$, $i \in \{1, ..., d\}$, and then uses \tilde{A} to aggregate relevant features for every feature. This aggregation step, called relationship graph convolution (RGC), is

$$\boldsymbol{H} = \tilde{\boldsymbol{X}}\tilde{\boldsymbol{A}}\boldsymbol{W}_1 + \boldsymbol{b}_1, \tag{2}$$

where $W_1 \in \mathbb{R}^{d \times d}$ and $b_1 \in \mathbb{R}^d$ are trainable parameters. Unlike usual graph convolution (Veličković et al., 2017; Hamilton et al., 2017), parameters of the features are directly aggregated. Then, residual connection (He et al., 2016) adds \tilde{X} to $H \in \mathbb{R}^{N \times d}$ as $\bar{X} = \sigma \left(\tilde{X} + H \right)$ where σ can be *tanh* for numeric features and *sigmoid* for categorical features. With the use of residual learning at this layer, H serves as corrections that addresses the errors remaining in \tilde{X} .



Figure 2: An example of random masking strategy. Observed values (white boxes) in the original mini-batch are randomly masked with probability p_{mask} .

At last, GRIL calculates the final imputation matrix \hat{X} by maintaining observed values in X and replacing missing values in X with imputed values in \bar{X} , as

$$\hat{\boldsymbol{X}} = \boldsymbol{X} \odot \boldsymbol{M} + \bar{\boldsymbol{X}} \odot (1 - \boldsymbol{M}), \qquad (3)$$

where \odot is an element-wise multiplication.

3.2.3 PL: PREDICTION LAYER

After GRIL, PL uses the final imputation matrix \hat{X} to conduct downstream tasks. Depending on the downstream task, the prediction is $\hat{Y} = softmax(\hat{X}W_2 + b_2)$ if the downstream task is classification, and $\hat{Y} = \hat{X}W_2 + b_2$ if it is regression, where $W_2 \in \mathbb{R}^{d \times c}$ and $b_2 \in \mathbb{R}^c$ are trainable parameters. Here, PL uses a one-layer fully-connected network (FCN) for the prediction, but the structure can be extended to increase complexity.

241 242

268

269

224

225 226 227

228

233 234

235

242 3.2.4 OBJECTIVE FUNCTION

The core idea of PIG is to use appropriate feature relationships to improve the solution for both
imputation and prediction. The objective function is constructed over the three main layers (Figure 8
in Appendix E).

First, GLL computes the regularization loss \mathcal{L}_{reg} to ensure that the inferred relationship matrix Aalign with prior knowledge A^{prior} about feature relationships. Specifically, it uses binary crossentropy to assess how much A deviates from A^{prior} . This loss encourages the model to deviate from A^{prior} if deviations significantly enhance both imputation and prediction.

Second, GRIL computes the two types of losses: reconstruction loss \mathcal{L}_{rec} and imputation loss \mathcal{L}_{imp} . The reconstruction loss ensures that the model accurately restores the actual values of the observed ones, whereas the imputation loss ensures that the model accurately guess true values of the missing ones. The imputation loss cannot be calculated directly because true values for unobserved ones are unknown. To solve this problem, observed values are randomly masked with probability p_{mask} to simulate missing values. Then, the imputation loss is computed using these artificially masked values and the reconstruction loss is computed using only observed and unmasked values (Figure 2).

Specifically, GRIL distinguishes between numeric and categorical features by applying proper
 losses: squared differences for numeric features and cross-entropy loss for categorical features.
 This approach enables PIG to flexibly and accurately process mixed-type datasets.

Lastly, PL computes the downstream task loss \mathcal{L}_{dt} . Depending on whether the task is classification or regression, the last loss is calculated differently. It uses cross-entropy loss between the predicted labels \hat{Y} and the true labels Y for classification task, whereas it utilizes mean squared error (MSE) between the predicted values \hat{Y} and the actual values Y for regression task.

Finally, the total loss \mathcal{L} of PIG is 267

$$\mathcal{L} = \lambda \mathcal{L}_{reg} + \mathcal{L}_{rec} + \mathcal{L}_{imp} + \mathcal{L}_{dt}, \tag{4}$$

where λ is a term that adjusts the regularization penalty.

270 271 272

273 274 275

276 277

278



Figure 3: Two-step training of PIG. Orange boxes represent pre-training and red boxes represent the completion of training.

279 280 281

282

297 298 299

301

302

303 304

305

3.3 HEURISTIC ALGORITHM TO FIND PRIOR FEATURE RELATIONSHIPS

The feature relationships are unknown in many real-world data, so we propose a heuristic algorithm to estimate the prior feature relationships A^{prior} (Algorithm 1). Imputation can be considered as a supervised learning task in which each feature is predicted by the others. If the *i*-th feature $X_{.i}$ helps to predict another feature $X_{.j}$, then a reasonable assumption is that $X_{.i}$ is relevant in imputing $X_{.j}$.

This relevance can be decently inferred using the feature importance of the decision tree algorithm (Hastie et al., 2009). Thus, we propose to use d decision trees, each of which is fitted to predict each feature in data X. In this process, only complete data instances that have no missing values are used. Specifically, each decision tree is trained to predict $X_{\cdot i}^c$ using $X_{\cdot -i}^c$, which is obtained by excluding the *i*-th feature $X_{\cdot i}^c$. Here, the superscript *c* indicates the subset of X in which all features are observed.

In this case, the feature importance of *j*-th feature for predicting *i*-th feature, denoted as FI_{ij} , is the mean decrease in impurity of X_{i} . After the feature importance of every pair is computed, A^{prior} is constructed as

$$A_{ij}^{prior} = \begin{cases} 1, \text{ if } FI_{ij} \ge \text{Average}(\mathbf{FI}_{i.}) \\ 0, \text{ otherwise} \end{cases}$$
(5)

where Average(FI_{i}) = $\frac{1}{d-1} \sum_{j \neq i} FI_{ij}$. All diagonal elements A_{ii} are set to 0.

These prior relationships may be a strong assumption on data, so A will be updated during the training process only if a deviation from the prior is advantageous for reconstruction, imputation, or downstream tasks. The detailed steps of the algorithm are provided in Appendix D.

3.4 TRAINING STRATEGY

If parameter values were inappropriately initialized, then gradients of tightly related losses would have different directions. We carefully designed the two-step optimization strategy for PIG (Figure 3) to avoid this 'conflicting gradients' problem via pre-training. In step 1, the strategy pre-trains GLL and PL separately, and finds prior feature relationships A^{prior} . In this process, only complete data instances are used, so the pre-trained parameters may be sub-optimal. In step 2, GLL and PL are initialized with the pre-trained parameters and then all the parameters in PIG are jointly optimized using the total loss \mathcal{L} . This strategy succesfully guides PIG to an improved solution (see Table 3).

313 314

315

4 EXPERIMENTS

316 317 4.1 DATASETS AND EXPERIMENTAL SETTINGS

To evaluate the performance of PIG, we conducted various experiments using nine benchmark datasets from the UCI Machine Learning Repository (Lichman, 2013), as in previous studies on MDI for tabular datasets (Table 1); Among them, three datasets involve the regression downstream task and six involve the classification downstream task (Table 4). In a few datasets, we removed data instances with missing values to make these datasets evaluable. We then randomly generated missing values in the datasets to simulate data that have missing values. Finally, we pre-processed all the numeric features to be scaled to [-1, 1], and categorical features to be one-hot encoded. 324 We first evaluated imputation and prediction accuracy. For this experiment, we randomly removed 325 20% of elements and used them as originally-missing values, and then calculated the performance. 326

For the imputation task, we measured MSE for numeric features and CE for categorical features. 327 For the regression downstream task, we assessed mean absolute error (MAE) and MSE, while for 328 the classification downstream task, we assessed accuracy, precision, recall, and F1 score.

330 We conducted an ablation study to test the effect of GLL, the initial imputation strategy for X, and the pre-training strategy. At last, We also conducted the sensitivity analysis to test how PIG is 331 332 sensitive to missing rates, adjusting the ratio from 5% to 50% (see Appendix K).

333 We repeated every experiment 10 times and reported the average accuracy on the test set. For each 334 repetition, We first randomly divided the datasets into 70% for training, 20% for validation, and 10% 335 for testing. Then, we introduced random missing values into the training dataset.

336 We set hyperparameters of PIG as follows. The initial imputation method is Soft-Impute (SI), as 337 described by Mazumder et al. (2010), unless otherwise stated. We set the dimensionality d_E of 338 feature embeddings to |d/2| differently according to datasets, where $|\cdot|$ is the floor function. The 339 scaling parameter α that controls the sensitivity in GLL was set to 5 and the regularization parameter 340 λ (Equation 4) that controls the penalty of the regularization loss to 0.01. We trained PIG for 60 341 epochs using Adam optimizer (Kingma & Ba, 2014) with learning rate 0.01, and applied early 342 stopping of 10-epoch-patience. We set a random masking probability p_{mask} to 20% for all datasets.

343 We compared PIG with some existing imputation methods and their MTL versions (if extendable) 344 to see whether the improvement of our method is attributed to only MTL. MTL versions of existing 345 models are implemented by modifying each method to learn imputation and prediction simultane-346 ously as in PIG. In addition, a model without the MTL structure first learns imputation with a com-347 plete matrix and then trains PL by using imputed data, where the prediction layer is one-layer FCN 348 as in PIG. As baseline models, we used SI (Mazumder et al., 2010), AE, VAE, GRAPE (You et al., 2020), MIRACLE (Kyono et al., 2021), and HyperImpute (Jarrett et al., 2022). A brief description 349 of these baseline models is in Appendix C. 350

4.2 IMPUTATION AND PREDICTION ACCURACY

We conducted experiments on nine benchmark datasets and measured the accuracies of the imputation and downstream tasks. Three datasets (i.e., "Energy", "Abalone", and "Diabetes") have a regression downstream task and the rest (i.e., "Spam", "Mobile", "Pulsar", "Breast Cancer", "Faults", and "Wine") have a classification downstream task.



Figure 4: The imputation performance across three regression datasets. Each chart displays the results for a specific dataset. To reduce complexity, only numerical imputations are visualized.

PIG showed the best imputation and regression accuracy for three regression datasets (Figure 4 and 376 5). PIG consistently showed best performance, except for a second-place on the "Diabetes" dataset. 377 In this dataset, GAIN outperformed our methods in both imputation and regression performance;



373

374 375

351 352

353 354

355

356

357

359

360



Figure 5: The regression performance for three datasets. Each chart displays the results for a specific dataset.

however, it was unstable in that its imputation performance was worse than that of traditional methods like SI and MissForest in other datasets. All experimental results are detailed in Appendix I.



Figure 6: The imputation performance for six classification datasets. Each chart displays the results for a specific dataset. To reduce complexity, only numerical imputations are visualized.

In addition, PIG showed superior performance for the six classification datasets (Figure 6 and 7).
PIG achieved first place in all classification datasets, with the exception of the "Wine" dataset, where
PIG ranked second in terms of classification performance despite the best imputation performance.
All experimental results are detailed in Appendix J.

In both imputation and prediction tasks, regardless of regression or classification, PIG demonstrated
superior performance while achieving the top average ranking among all baseline models (Table 5
in Appendix F). Especially, the gap with the runner-up method is fairly large in prediction tasks
compared to imputation tasks. Additionally, methods with MTL demonstrated similar or slightly
degraded imputation performance compared to other methods, but exhibited superior predictive performance in both regression and classification. This implies that considering downstream tasks is



Figure 7: The classification performance for six datasets. Each chart displays the results for a specific dataset.

crucial when imputation is necessary due to missing data. Among the MTL methods, PIG outperformed the other methods, which means that considering feature relationships is crucial for addressing tabular data with missing values. In both regression and classification tasks, PIG demonstrated significantly improved imputation performance compared to SI, which is used to generate initial imputation matrix in PIG. This means that PIG greatly contributed to the enhancement of imputation performance by utilizing graph convolution with adaptive feature relationships.

465 4.3 ABLATION STUDY

We conducted a few ablation studies to evaluate the individual effect of the GLL, initial imputation strategies, and pre-training strategies. We explored their effects on both types of downstream tasks, regression and classification, using "Diabetes" and "Mobile" datasets, respectively. Due to space constraints, the ablation study on initial imputation strategies is provided in Appendix H, and the experimental settings beyond the focus of each ablation study are detailed in Section 4.2.

4.3.1 THE EFFECT OF GLL

We compared PIG with and without GLL and showed that the feature relationships changed after training PIG ended. We visualized the difference in feature relationships between A and A^{prior} (Figure 9 in Appendix G). The prevalence of consistent relationships (black arrows, Figure 9) high-lights the stability of our heuristic approach (Section 3.3). Furthermore, the presence of red and blue arrows implies that GLL facilitates adaptive learning of feature relationships. These changes also led to improvement of performance in both imputation and downstream tasks (Table 2), demonstrat-ing that GLL contributed the improvement of imputation and downstream tasks by adjusting feature relationships in a data-driven manner.

482 4.3.2 THE EFFECT OF PRE-TRAINING STRATEGY483

Finally, we assessed the impact of our pre-training strategy by comparing the performance of imputation and prediction tasks with and without pre-training (Table 3). It appears that the pre-training strategy is not as effective for imputation tasks, as PIG with pre-training showed more accurate per-

486	Table 2: Ablation results to measure the impact of GLL for two datasets: Mobile (classification) and
487	Diabetes (regression).

Detect	Madala	Imputation Task MSE		Regression Task				
Dataset	Models			M	SE	MAE		
Diabatas	PIG	$0.034\pm$ 0.009		$0.122 \pm$ 0.027		0.280 ± 0.035		
Diabetes	PIG w/o GLL	$0.041\pm$ 0.006		$0.125\pm$ 0.024		$0.288\pm$ 0.037		
Detecat	Models	Imputation Task		Classification Task				
Dataset	Widdels	MSE	CE	Accuracy	Recall	Precision	F1-score	
Mobile	PIG	$0.084 \pm \scriptscriptstyle 0.007$	$0.222 \pm \scriptscriptstyle 0.042$	$0.952 \pm \scriptscriptstyle 0.017$	$0.951 \pm \scriptscriptstyle 0.018$	$0.952 \pm \scriptscriptstyle 0.017$	0.950 ± 0.018	
Mobile	PIG w/o GLL	0.101 ± 0.005	$0.344\pm$ 0.016	$0.938\pm \scriptscriptstyle 0.014$	$0.940\pm$ 0.014	$0.938\pm \scriptscriptstyle 0.014$	$0.938\pm \scriptscriptstyle 0.014$	

T 11 0	A 1 1 . *	1		. 1	•	C	
Table 3.	Ablation	roculte to	maggiira	the	imnact	ot	nro fraining
Table 5.	AUIALIOII	icsuits u	Jincasure	unc	mudact	UI.	Dic-u anning

Dete	Modala	Imputat	ion Task	Regression Task			
Data	wodels	MSE	CE	MSE		MAE	
Enoral	PIG	0.046 ± 0.005	-	0.031	± 0.006	0.126	\pm 0.015
Lifergy	PIG w/o pre-train	0.043 ± 0.007	-	0.080	\pm 0.065	0.194	± 0.084
Abalana	PIG	$0.004 \pm \scriptscriptstyle 0.002$	0.100 ± 0.013	0.027	± 0.005	0.118	± 0.009
Abalone	PIG w/o pre-train	$0.007\pm \scriptscriptstyle 0.007$	$0.124\pm$ 0.013	0.029	± 0.007	0.121	± 0.009
Diabatas	PIG	0.034 ± 0.009	-	0.122	± 0.027	0.280	± 0.035
Diabetes	PIG w/o pre-train	$0.028\pm$ 0.006	-	0.136 ± 0.026		36 ± 0.026 0.299 ± 0.03	
Data	M. 1.1.	Imputation Task		Classifica		ation Task	
Data	Models	MSE	CE	Accuracy	Recall	Precision	F1-score
Casar	PIG	0.006 ± 0.001	-	0.899 ± 0.020	$0.823 \pm {\scriptstyle 0.022}$	0.901 ± 0.020	0.892 ± 0.022
Span	PIG w/o pre-train	0.005 ± 0.001	-	$0.849\pm$ 0.017	$0.823 \pm \scriptscriptstyle 0.022$	$0.859\pm \scriptscriptstyle 0.017$	$0.828\pm ext{0.021}$
Mahila	PIG	$0.084 \pm$ 0.007	$0.222 \pm \scriptscriptstyle 0.042$	$0.952 \pm ext{0.017}$	$0.951 \pm ext{0.018}$	$0.952 \pm ext{0.017}$	0.950 ± 0.018
Mobile	PIG w/o pre-train	0.061 ± 0.006	$0.393 \pm \scriptscriptstyle 0.011$	$0.712 \pm \scriptstyle 0.034$	$0.719 \pm \scriptscriptstyle 0.034$	$0.712\pm ext{0.037}$	0.691 ± 0.036
Dulcor	PIG	$0.046\pm$ 0.002	-	$0.923 \pm ext{0.018}$	0.923 ± 0.018	0.926 ± 0.017	0.922 ± 0.018
Fulsai	PIG w/o pre-train	$0.014 \pm \scriptscriptstyle 0.002$	-	$0.909 \pm \scriptscriptstyle 0.027$	$0.909\pm \scriptscriptstyle 0.025$	$0.913 \pm \scriptscriptstyle 0.029$	0.906 ± 0.029
Proact Concor	PIG	$0.010\pm$ 0.006	-	0.962 ± 0.029	0.956 ± 0.033	0.966 ± 0.025	0.960 ± 0.030
Breast Cancer	PIG w/o pre-train	$0.020\pm$ 0.006	-	$0.714 \pm ext{0.138}$	$0.536\pm$ 0.157	$0.567 \pm \scriptscriptstyle 0.264$	$0.473\pm ext{0.182}$
Foulto	PIG	0.085 ± 0.134	-	$0.692\pm extrm{0.026}$	$0.714 \pm ext{0.048}$	$0.690\pm$ 0.042	0.686 ± 0.042
Faults	PIG w/o pre-train	$0.153\pm$ 0.219	-	$0.693 \pm {\scriptstyle 0.042}$	$0.705\pm ext{0.055}$	0.755 ± 0.051	0.668 ± 0.064
Wine	PIG	0.013 ± 0.003	-	0.561 ± 0.020	$0.405\pm$ 0.078	$0.775\pm \scriptscriptstyle 0.037$	0.397 ± 0.080
wille	PIG w/o pre-train	$0.009 \pm \scriptscriptstyle 0.002$	-	0.569 ± 0.056	0.546 ± 0.059	0.810 ± 0.035	0.517 ± 0.068

formance in only four out of nine datasets. However, the pre-training strategy significantly enhanced regression or classification performance in almost all datasets, with the exception of the "Wine" dataset. This implies that pre-training effectively stabilizes the two different objectives, imputation and prediction, by properly setting the initial parameters in GLL and PL.

5 CONCLUSION

This paper has presented PIG, a framework for MTL of imputation and prediction as a downstream task. We also proposed a heuristic method to easily find feature relationships that may be helpful to imputation and downstream tasks. The framework can exploit these relationships in graph convolution to correct the initial imputation, then updates the relationships if necessary to improve the accuracy of both tasks. To achieve the mentioned purposes, the objective function of PIG has several loss terms, which are difficult to optimize simultaneously; thus, we suggested first pre-training GLL and PL and then jointly optimizing the whole PIG. This training strategy successfully balances several tasks. Empirically, PIG achieved the best performance in imputation and prediction tasks across nine datasets, leading to the average rank of 1.78 for imputation task and 1.22 for downstream tasks. Consequently, PIG can be the robust and versatile approach in handling tabular data with missing values with specific downstream tasks.

6 Reproducibility

The hyperparameters, data preprocessing steps, and experiment settings are provided in Section 4.1
of the main paper, while the details of the datasets are summarized in Table 4 in the appendix. For reproducibility, we will make the source code publicly available.

540 REFERENCES 541

553

554

563

- Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient 542 normalization for adaptive loss balancing in deep multitask networks. In International conference 543 on machine learning, pp. 794–803. PMLR, 2018. 544
- Lovedeep Gondara and Ke Wang. Mida: Multiple imputation using denoising autoencoders. In 546 Pacific-Asia conference on knowledge discovery and data mining, pp. 260–272. Springer, 2018. 547
- 548 Leo Grinsztajn, Edouard Oyallon, and Gael Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, 549 K. Cho, and A. Oh (eds.), Advances in Neural Information Processing Systems, volume 35, pp. 550 507–520. Curran Associates, Inc., 2022. 551
- 552 Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. Advances in neural information processing systems, 30, 2017.
- 555 Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. The elements of statistical learning: data mining, inference, and prediction, volume 2. Springer, 2009. 556
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recog-558 nition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 559 770–778, 2016. 560
- 561 Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. 562 In 2008 Eighth IEEE international conference on data mining, pp. 263–272. Ieee, 2008.
- Daniel Jarrett, Bogdan C Cebere, Tennison Liu, Alicia Curth, and Mihaela van der Schaar. HyperIm-564 pute: Generalized iterative imputation with automatic model selection. In Kamalika Chaudhuri, 565 Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), Proceedings of 566 the 39th International Conference on Machine Learning, volume 162 of Proceedings of Machine 567 Learning Research, pp. 9916–9937. PMLR, 17–23 Jul 2022. 568
- 569 Kyung-Jun Kim, Kyu-Jin Kim, Chi-Hyuck Jun, Il-Gyo Chong, and Geun-Young Song. Variable 570 selection under missing values and unlabeled data in semiconductor processes. *IEEE Transactions* 571 on Semiconductor Manufacturing, 32(1):121–128, 2018.
- 572 Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint 573 arXiv:1412.6980, 2014. 574
- 575 Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional net-576 works. arXiv preprint arXiv:1609.02907, 2016.
- Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender 578 systems. Computer, 42(8):30-37, 2009. 579
- 580 Trent Kyono, Yao Zhang, Alexis Bellot, and Mihaela van der Schaar. Miracle: Causally-aware 581 imputation via learning missing data mechanisms. In M. Ranzato, A. Beygelzimer, Y. Dauphin, 582 P.S. Liang, and J. Wortman Vaughan (eds.), Advances in Neural Information Processing Systems, 583 volume 34, pp. 23806–23817. Curran Associates, Inc., 2021. 584
- Jiang Li, Xiaowei S Yan, Durgesh Chaudhary, Venkatesh Avula, Satish Mudiganti, Hannah Husby, 585 Shima Shahjouei, Ardavan Afshar, Walter F Stewart, Mohammed Yeasin, et al. Imputation of 586 missing values for electronic health record laboratory data. NPJ digital medicine, 4(1):1-14, 587 2021. 588
- 589 Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural net-590 work: Data-driven traffic forecasting. arXiv preprint arXiv:1707.01926, 2017. 591
- M Lichman. Uci machine learning repository [http://archive. ics. uci. edu/ml]. irvine, ca: University 592 of california, school of information and computer science. URL: http://archive. ics. uci. edu/ml, 2013.

- 594 Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent 595 for multi-task learning. Advances in Neural Information Processing Systems, 34:18878–18890, 596 2021. 597 Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learn-598 ing large incomplete matrices. The Journal of Machine Learning Research, 11:2287–2322, 2010. 600 Alfredo Nazabal, Pablo M Olmos, Zoubin Ghahramani, and Isabel Valera. Handling incomplete 601 heterogeneous data using vaes. Pattern Recognition, 107:107501, 2020. 602 Yansong Qu, Zhenlong Li, Xiaohua Zhao, and Jushang Ou. Towards real-world traffic prediction 603 and data imputation: A multi-task pretraining and fine-tuning approach. *Information Sciences*, 604 657:119972, 2024. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2023.119972. 605 606 Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars BPR Schmidt-Thieme. Bayesian 607 personalized ranking from implicit feedback. In Proc. of Uncertainty in Artificial Intelligence, 608 pp. 452-461, 2014. 609 Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional 610 networks for skeleton-based action recognition. In Proceedings of the IEEE/CVF conference on 611 computer vision and pattern recognition, pp. 12026–12035, 2019. 612 613 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. 614 Bioinformatics, 17(6):520–525, 2001. 615 616 Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equa-617 tions in r. Journal of statistical software, 45:1-67, 2011. 618 Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua 619 Bengio. Graph attention networks. arXiv preprint arXiv:1710.10903, 2017. 620 621 Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and 622 composing robust features with denoising autoencoders. In Proceedings of the 25th international 623 conference on Machine learning, pp. 1096–1103, 2008. 624 Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Con-625 necting the dots: Multivariate time series forecasting with graph neural networks. In Proceedings 626 of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, pp. 627 753–763, 2020. 628 629 Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for 630 skeleton-based action recognition. Proceedings of the AAAI Conference on Artificial Intelligence, 631 32(1), Apr. 2018. doi: 10.1609/aaai.v32i1.12328. 632 Jinsung Yoon, Camelia Davtyan, and Mihaela van der Schaar. Discovery and clinical decision 633 support for personalized healthcare. IEEE journal of biomedical and health informatics, 21(4): 634 1133-1145, 2016. 635 636 Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In International conference on machine learning, pp. 5689–5698. PMLR, 2018. 637 638 Jiaxuan You, Xiaobai Ma, Yi Ding, Mykel J Kochenderfer, and Jure Leskovec. Handling missing 639 data with graph representation learning. Advances in Neural Information Processing Systems, 33: 640 19075-19087, 2020. 641 Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 642 Gradient surgery for multi-task learning. Advances in Neural Information Processing Systems, 643 33:5824-5836, 2020. 644 645 646
- 647

648 A TYPES OF MISSINGNESS

650 Missing data are of three types: Missing At Random (MAR), Missing Not At Random (MNAR), 651 and Missing Completely At Random (MCAR). MAR refers to the case that missingness depends 652 on the observed features, MNAR refers to the case that missingness depends on both observed and 653 unobserved data. In MCAR case, missingness is independent of both observed and unobserved data. For example, consider two observed input features x_1 and x_2 , and an unobserved feature z. If the 654 missingness of x_1 depends only on x_2 or vice versa, the data is MAR. If the missingness of x_1 or 655 x_2 additionally depends on z, the data is MNAR. Finally, if missingness is independent of other 656 features, it is MCAR. 657

Among the three types, MCAR represents the simplest type of missing data and thus enables the evaluation of imputation models without the introduction of biases unlike MAR and MNAR. Due to this simplicity, most existing studies on MDI are based on the MCAR assumption as a stepping stone towards more complex scenarios of MAR and MNAR. Therefore, this study also focuses on the MCAR case to make our results be directly comparable with those studies.

B DETAILS OF DATASETS.

663

665 666

667 668

669

670

671

672

673

674

675

676

677

678 679 680

681 682

683

684

685

686 687

688 689

690

691

692

693

696

697

features Task Dataset # instances Numeric Categorical 1,296 28 0 Energy 4,177 7 Regression Abalone 1 442 10 0 Diabetes Spam 4.601 57 0 Mobile 2,000 14 6 Pulsar 12,528 9 0 Classification Breast Cancer 30 0 569 Faults 1,941 33 0 Wine 1,143 12 0

Table 4: Details of datasets.

C BASELINE MODEL DESCRIPTION

- Soft-Impute (SI) (Mazumder et al., 2010): a matrix completion method that uses the nuclear norm as a regularization loss.
- Auto-Encoder (AE): A traditional DL method that focuses on the reconstruction loss to impute missing values.
- Variational Auto-Encoder (VAE): A deep generative method that has a similar model structure to AE but trains the model weights by maximizing the evidence lower bound
- GRAPE (You et al., 2020): A GNN-based representation learning method to impute missing values. This method constructs a bipartite graph of N data instance nodes and d feature nodes. This method can be itself an MTL approach.
- MIRACLE (Kyono et al., 2021): A neural network-based imputation method that refines imputation results by incorporating data dependencies.
- HyperImpute (Jarrett et al., 2022): A generalized iterative imputation framework that adaptively selects the most suitable model, ranging from traditional to recent models, for each feature.

698 MTL versions of AE, VAE, and GRAPE are denoted as AE (MTL), VAE (MTL), and GRAPE 699 (MTL), respectively. Both PIG and MTL versions of baseline models used a one-layer FCN for fair 700 comparisons. AE and VAE require a complete matrix as an input, so SI first initializes an incomplete 701 matrix X for those two methods. Hyperparameters were set to as suggested by their authors (You et al., 2020) or determined experimentally.

D HEURISTIC ALGORITHM FOR PRIOR RELATIONSHIPS

Algorithm 1 The proposed heuristic algorithm for prior relationships

Inputs:

11: return A^{prior}

702

703 704

705 706

707

708

721 722 723

724

 X^c : a subset of X that contains only data instances with no missing values.

1: Initialize prior feature relationship matrix $A^{prior} \in \mathbb{R}^{d \times d}$ 709 710 2: for i = 1, ..., d do $oldsymbol{X}^{c}_{\cdot i} \leftarrow i$ -th column of $oldsymbol{X}^{c}$ 711 3: $X_{\cdot i}^c \leftarrow$ a subset of X^c derived by excluding *i*-th column $X_{\cdot i}^c$ 4: 712 5: Fit a decision tree DT_i that predicts $X_{\cdot i}^c$ from $X_{\cdot -i}^c$ 713 6: FI_i . \leftarrow feature importance derived from DT_i $\triangleright FI_{ii}$ is set to zero. 714 7: for $j = 1, \ldots, d$ do 715 $A_{ij}^{prior} \leftarrow \begin{cases} 1, \text{ if } FI_{ij} \ge \text{Average}(\boldsymbol{FI}_{i}) \\ 0, \text{ otherwise} \end{cases}$ 716 8: 717 end for 9: 718 10: end for 719 720

E OBJECTIVE FUNCTION DESIGN OF PIG





F AVERAGE RANKINGS OF IMPUTATION AND PREDICTION

760								
761	T1-	M - 411	Impu	tation	Regre	ession	Classifi	cation
762	Task	Method	MSE	CE	MSE	MAE	Accuracy	F1 score
763		SI	5.00	7.00	5.00	5.33	-	-
764		MICE	7.33	10.00	6.67	6.67	-	-
765		MissForest	3.33	11.00	6.33	5.67	-	-
766		AE	7.67	2.00	7.67	8.67	-	-
767		VAE	8.33	4.00	7.67	8.67	-	-
768		AE (MTL)	7.00	5.00	6.00	7.00	-	-
769	Regression	VAE (MTL)	10.33	6.00	7.00	8.00	-	-
770		GRAPE (MTL)	9.33	3.00	4.33	5.33	-	-
771		GRAPE	12.00	8.00	11.00	7.67	-	-
772		MIRACLE	13.00	13.00	13.00	13.00	-	-
773		HyperImpute	2.33	12.00	8.00	9.00	-	-
774		GAIN	3.67	9.00	5.33	4.00	-	-
774		PIG	1.33	1.00	1.33	1.33	-	-
776		SI	3.67	3.00	-	-	5.17	4.67
777		MICE	8.00	8.00	-	-	9.00	9.83
770		MissForest	2.83	9.00	-	-	7.33	8.67
778		AE	7.67	3.00	-	-	6.67	6.00
779		VAE	7.67	2.00	-	-	5.33	5.33
780		AE (MTL)	5.83	6.00	-	-	3.67	4.50
781	Classification	VAE (MTL)	7.83	12.00	-	-	5.83	5.17
782		GRAPE (MTL)	10.33	7.00	-	-	5.00	5.00
783		GRAPE	11.83	12.00	-	-	10.00	8.83
784		MIRACLE	13.00	5.00	-	-	13.00	12.33
785		HyperImpute	2.83	10.00	-	-	9.00	9.67
786		GAIN	6.67	11.00	-	-	8.33	9.17
787		PIG	1.83	1.00	-	-	1.17	1.17

Table 5: Average rankings of imputation and prediction.



Figure 9: The visualization of changes in feature relationships by GLL (Upper section: "Mobile" dataset, lower section: "Diabetes" dataset). Circles represent features in each dataset, and arrows indicate the asymmetric relationships between these features, where the features at the tail of arrows significant affect the features at the head. Black arrows represent feature relationships that are consistent in both A^{prior} and A. The blue arrows indicate feature relationships that were present in A^{prior} but deleted in A, while red arrows represent feature relationships that were not in A^{prior} but have been newly added to A. We created the feature relationship graph by thresholding A at a value of 0.5.

- 852 853
- 854
- 855 856
- 857
- 858
- 859
- 860
- 861
- 862
- 863

ABLATION STUDY: THE EFFECT OF THE INITIAL IMPUTATION METHOD Η

We tested the performance of PIG using different initial imputation methods (Table 6). We compared four simple imputation strategies: SI, MEAN, MEDIAN, and CONSTANT. MEAN and MEDIAN use each feature's mean or median value to fill in the missing values, and CONSTANT replaces missing values with zeros. We first measured the imputation accuracy of these four imputation methods and tested downstream task accuracy using the data imputed by these methods (the prediction model is trained in a separate step). Then we trained four PIG models that used different initial imputation methods and measured their imputation and downstream task accuracy. Each model is denoted as PIG (imputation method), such as PIG (MEAN).

The results of Mobile dataset show that initial imputation methods can make meaningful differences in both classification performance and imputation accuracy of categorical features. In the results of Diabetes dataset, imputation accuracy of numerical features varied according to initial imputation methods. However, PIG increased the performance of both imputation and downstream tasks regardless of initial imputation methods, compared to two-phase training methods that use four sim-ple imputation strategies.

We also compared PIG with four simple imputation methods to other baseline models. PIG achieved accuracy much higher than the baseline models regardless of the initial imputation method. This means that PIG greatly contributed to the enhancement of imputation performance from the initial imputation matrix by utilizing graph convolution with adaptive feature relationship.

Table 6: Comparison of PIG models with different initial imputation methods for two datasets: Diabetes (regression) and Mobile (classification).

			T					
888	Data	Models	Imputation Task		м	Regress	ION TASK	
889		DIC (CONSTANT)	M	<u>SE</u>	0.127	SE	MAE	
890		PIG (CONSTANT)	0.047	± 0.007	0.127	± 0.017	0.289 ± 0.021	
004		PIG (MEDIAN)	0.044	± 0.010	0.130	+ 0.021	0.289 ± 0.030	
891		PIG (SI)	0.034	+ 0.003	0.122	+ 0.027	0.280 ± 0.027	
892		CONSTANT	0.251	+ 0.031	0.122	+ 0.024	0.324	+ 0.031
893		MEAN	0.230	± 0.050	0.135	± 0.021	0.302	± 0.030
004	Diabetes	MEDIAN	0.305	± 0.067	0.119	\pm 0.022	0.280	\pm 0.027
094		SI	0.207	\pm 0.029	0.146	± 0.018	0.315	± 0.020
895		AE	0.214	± 0.026	0.144 ± 0.033		$0.309 \pm {}_{0.042}$	
896		VAE	0.216	\pm 0.035	0.155 ± 0.026		0.325 ± 0.031	
807		AE (MTL)	$0.220\pm$ 0.032		$0.140\pm$ 0.027		$0.308 \pm {}_{0.028}$	
0.57		VAE (MTL)	0.227 ± 0.034		0.141 ± 0.029		$0.302\pm$ 0.033	
898		GRAPE (MTL)	0.226	± 0.019	0.123	± 0.022	0.285	± 0.028
899	Data	Models	Imputat	ion Task		Classifica	ation Task	
900	Data	Widdels	MSE	CE	Accuracy	Recall	Precision	F1-score
004		PIG (CONSTANT)	0.081 ± 0.004	0.278 ± 0.009	0.922 ± 0.022	0.924 ± 0.021	0.921 ± 0.023	0.921 ± 0.022
901		PIG (MEAN)	0.078 ± 0.004	0.202 ± 0.023	0.922 ± 0.019	0.923 ± 0.017	0.920 ± 0.023	0.920 ± 0.021
902		PIG (MEDIAN)	0.079 ± 0.004	0.319 ± 0.021	0.911 ± 0.016	0.912 ± 0.020	0.912 ± 0.015	0.909 ± 0.018
903		PIG (SI)	0.084 ± 0.007	0.222 ± 0.042	0.952 ± 0.017		0.952 ± 0.017	0.950 ± 0.018
000		CONSTANT	0.382 ± 0.014	0.693 ± 0.000	0.598 ± 0.031	0.601 ± 0.032	0.594 ± 0.033	0.592 ± 0.035
904	Mahila	MEDIAN	0.340 ± 0.014	0.710 ± 0.019	0.594 ± 0.038	0.590 ± 0.036	0.591 ± 0.034	0.580 ± 0.035
905	Mobile	NIEDIAN	0.340 ± 0.011	0.723 ± 0.032 0.676 ± 0.032	0.399 ± 0.065 0.500 ± 0.005	0.397 ± 0.063	0.001 ± 0.064	0.392 ± 0.065 0.582 ± 0.065
906			0.334 ± 0.015	0.070 ± 0.012	0.390 ± 0.041	0.394 ± 0.038	0.388 ± 0.040	0.382 ± 0.041
007		VAE	0.346 ± 0.013 0.336 ± 0.013	0.670 ± 0.008 0.665 ± 0.008	0.009 ± 0.047 0.616 ± 0.077	0.010 ± 0.048 0.611 ± 0.070	0.005 ± 0.033 0.606 ± 0.076	0.601 ± 0.032 0.603 ± 0.077
907		AE (MTL)	0.336 ± 0.009	0.605 ± 0.005	0.010 ± 0.017 0.758 ± 0.036	0.011 ± 0.010 0.760 ± 0.034	0.000 ± 0.000 0.791 ± 0.030	0.003 ± 0.077 0.763 ± 0.032
908		VAE (MTL)	0.343 ± 0.005	0.727 ± 0.034	0.750 ± 0.021	0.750 ± 0.000	0.777 ± 0.022	0.754 ± 0.022
909		GRAPE (MTL)	0.371 ± 0.006	0.697 ± 0.001	0.757 ± 0.033	0.756 ± 0.030	0.785 ± 0.034	0.758 ± 0.030
010		. /						

I EXPERIMENTAL RESULTS FOR THREE REGRESSION DATASETS

Table 7: The regression and imputation performance for three datasets. MAE stands for mean absolute error. Bold: best score. Underline: runner-up.

923			D	· 70 1	Imputation Teals		
924	Data	Models	Kegress	ION LASK	MSE CE		
925		~~	MSE	MAE	MSE	CE	
026		SI	0.045 ± 0.011	0.152 ± 0.016	0.213 ± 0.029	-	
920		MICE	$0.0/4 \pm 0.004$	0.185 ± 0.005	0.443 ± 0.018	-	
927		MISSFORESL	$\frac{0.041 \pm 0.002}{0.082 \pm 0.018}$	$\frac{0.145 \pm 0.001}{0.214 \pm 0.000}$	0.200 ± 0.025 0.502 ± 0.021	-	
928	Energy	VAE	0.082 ± 0.018 0.082 ± 0.018	0.214 ± 0.029 0.214 ± 0.029	0.302 ± 0.021 0.502 ± 0.021	-	
929		GRAPE	0.002 ± 0.018 0.278 ± 0.044	0.214 ± 0.029 0.145 ± 0.053	0.902 ± 0.021 0.915 ± 0.384	-	
930		GAIN	0.044 ± 0.011	0.153 ± 0.006	0.297 ± 0.002	-	
931		MIRACLE	0.953 ± 0.025	0.706 ± 0.019	14.963 ± 0.142	-	
932		HyperImpute	0.205 ± 0.015	$0.376\pm ext{0.013}$	0.136 ± 0.029	-	
933		ĂE (MTL)	$0.062\pm$ 0.017	0.184 ± 0.025	$\overline{0.388\pm \scriptscriptstyle 0.071}$	-	
03/		VAE (MTL)	$0.081\pm$ 0.014	$0.213 \pm \textbf{0.016}$	$0.516\pm ext{0.044}$	-	
934		GRAPE (MTL)	$0.042\pm$ 0.006	$0.163\pm$ 0.016	$0.468 \pm ext{0.043}$	-	
935		PIG	$0.031\pm$ 0.006	0.126 ± 0.015	0.046 ± 0.005	-	
936		SI	$0.033\pm$ 0.007	0.132 ± 0.015	$0.022\pm$ 0.028	$0.699\pm$ 0.010	
937		MICE	0.033 ± 0.001	0.124 ± 0.000	0.009 ± 0.0000	0.825 ± 0.026	
938		MissForest	0.035 ± 0.004	0.125 ± 0.008	0.007 ± 0.000	0.831 ± 0.002	
939	Abalone	AE	0.045 ± 0.008	0.154 ± 0.011	0.083 ± 0.017	$\frac{0.606 \pm 0.033}{0.640 \pm 0.033}$	
940		VAE	0.040 ± 0.003	0.151 ± 0.007	0.090 ± 0.009	0.648 ± 0.017	
0/1		GRAPE	0.428 ± 0.197	0.422 ± 0.326	1.062 ± 0.651	0.807 ± 0.102	
941		GAIN MIDACLE	0.050 ± 0.050	0.147 ± 0.007	0.015 ± 0.001	0.824 ± 0.001	
942		HuperImpute	0.939 ± 0.123	0.743 ± 0.057 0.126 ± 0.057	11.370 ± 0.174	1.083 ± 0.011	
943		AF (MTL)	$\frac{0.032 \pm 0.002}{0.045 \pm 0.002}$	0.120 ± 0.002 0.152 ± 0.002	$\frac{0.003 \pm 0.000}{0.072 \pm 0.012}$	0.830 ± 0.000 0.680 ± 0.012	
944		$V\Delta F (MTL)$	0.045 ± 0.005	0.152 ± 0.009 0.158 \pm 0.010	0.072 ± 0.013 0.098 ± 0.008	0.089 ± 0.013 0.693 ± 0.000	
945		GRAPE (MTL)	0.043 ± 0.007 0.044 ± 0.011	0.150 ± 0.010 0.150 ± 0.024	0.050 ± 0.008 0.157 ± 0.058	0.000 ± 0.000 0.617 ± 0.065	
946		PIG	0.027 ± 0.005	0.118 ± 0.009	0.004 ± 0.002	0.100 ± 0.013	
947		SI	0.146 ± 0.018	0.315 ± 0.02	0.207 ± 0.029	-	
948		MICE	0.205 ± 0.008	0.364 ± 0.008	$0.250\pm ext{0.032}$	-	
949		MissForest	0.214 ± 0.006	$0.383 \pm \scriptscriptstyle 0.001$	$0.173\pm$ 0.014	-	
949	Diabetes	AE	0.144 ± 0.033	$0.309 \pm \scriptscriptstyle 0.042$	$0.214\pm$ 0.026	-	
950	Diabetes	VAE	$0.155\pm extrm{0.026}$	$0.325\pm \scriptstyle 0.031$	$0.216\pm$ 0.035	-	
951		GRAPE	0.185 ± 0.055	0.332 ± 0.050	0.605 ± 0.408	-	
952		GAIN	0.047 ± 0.006	0.146 ± 0.008	0.014 ± 0.001	-	
953		MIRACLE	1.071 ± 0.128	0.792 ± 0.053	13.457 ± 0.135	-	
954		HyperImpute	0.206 ± 0.013	$0.3/5 \pm 0.017$	0.132 ± 0.016	-	
955		AE (MIL)	0.140 ± 0.027 0.141 \pm 0.027	0.308 ± 0.028 0.202 ± 0.028	0.220 ± 0.032	-	
956		CPADE (MTL)	0.141 ± 0.029 0.122 ± 0.029	0.302 ± 0.033	0.227 ± 0.034	-	
957		PIG	0.123 ± 0.022 0.122 ± 0.027	0.280 ± 0.028 0.280 ± 0.025	0.220 ± 0.019 0.034 ± 0.009	-	
050		110	0.122 ± 0.027	0.200 ± 0.035	0.034 ± 0.009		
300							
959							
960							
961							
962							
963							
064							

J EXPERIMENTAL RESULTS FOR SIX CLASSIFICATION DATASETS

Table 8: The classification and imputation performance for six datasets. Bold: best score. Underline: runner-up.

977	Data	Model		Classifica	ation Task		Imputatio	on Task
978		moder	Accuracy	Recall	Precision	F1 score	MSE	CE
070		SI	0.831 ± 0.025	0.805 ± 0.031	0.838 ± 0.022	0.814 ± 0.029	0.016 ± 0.003	-
979		MICE	0.803 ± 0.007	0.783 ± 0.013 0.820 \pm 0.010	0.796 ± 0.006 0.856 \pm 0.001	0.787 ± 0.012	0.043 ± 0.003	-
980		AE	0.830 ± 0.007 0.828 ± 0.020	0.329 ± 0.010 0.795 ± 0.026	0.830 ± 0.001 0.848 ± 0.016	0.807 ± 0.009 0.807 ± 0.025	$\frac{0.014 \pm 0.002}{0.020 \pm 0.003}$	-
981		VAE	0.818 ± 0.032	0.783 ± 0.043	0.846 ± 0.021	0.793 ± 0.044	0.018 ± 0.003	-
000		GRAPE	0.816 ± 0.063	0.805 ± 0.060	0.810 ± 0.061	0.807 ± 0.061	0.032 ± 0.015	-
982	Spam	GAIN MIRACLE	0.858 ± 0.017 0.596 ± 0.001	0.828 ± 0.020 0.581 ± 0.000	0.878 ± 0.018 0.579 ± 0.001	0.842 ± 0.020 0.580 \pm 0.001	0.019 ± 0.002 7 029 ± 0.037	-
983		HyperImpute	0.390 ± 0.001 0.850 ± 0.007	0.331 ± 0.000 0.830 ± 0.009	0.379 ± 0.001 0.852 ± 0.006	0.380 ± 0.001 0.838 ± 0.008	0.014 ± 0.001	-
984		AE (MTL)	0.883 ± 0.011	0.866 ± 0.013	0.890 ± 0.010	0.874 ± 0.012	$\overline{0.017 \pm 0.007}$	-
00-		VAE (MTL)	0.883 ± 0.019	0.869 ± 0.024	0.887 ± 0.015	0.875 ± 0.021	0.017 ± 0.002	-
985		GRAPE (MTL)	0.843 ± 0.025	0.814 ± 0.029 0.887 \pm 0.022	0.858 ± 0.022	0.825 ± 0.027	0.021 ± 0.003	-
986		SI	0.590 ± 0.020	0.594 ± 0.022	0.588 ± 0.040	0.582 ± 0.022	0.334 ± 0.015	- 0.676 \pm 0.012
987		MICE	0.505 ± 0.015	0.514 ± 0.018	0.520 ± 0.016	0.495 ± 0.015	0.616 ± 0.002	0.699 ± 0.003
507		MissForest	0.612 ± 0.025	0.620 ± 0.028	0.622 ± 0.030	0.601 ± 0.024	0.339 ± 0.002	0.707 ± 0.000
988		AE	0.609 ± 0.047 0.616 ± 0.077	0.610 ± 0.048 0.611 ± 0.070	0.605 ± 0.053 0.606 ± 0.076	0.601 ± 0.052 0.603 ± 0.077	0.348 ± 0.015 0.336 ± 0.013	0.676 ± 0.008 0.665 ± 0.009
989		GRAPE	0.653 ± 0.034	0.658 ± 0.030	0.000 ± 0.070 0.724 ± 0.038	0.649 ± 0.036	1.431 ± 0.231	$\frac{0.003 \pm 0.009}{0.727 \pm 0.041}$
000	Mobile	GAIN	0.581 ± 0.025	0.579 ± 0.020	0.573 ± 0.016	0.566 ± 0.026	0.397 ± 0.016	0.717 ± 0.001
550		MIRACLE	0.367 ± 0.006	0.374 ± 0.001	0.468 ± 0.003	0.316 ± 0.002	14.271 ± 0.545	0.690 ± 0.000
991		HyperImpute	0.582 ± 0.007 0.758 \pm 0.026	0.589 ± 0.010 0.760 \pm 0.024	0.584 ± 0.004 0.701 \pm 0.020	0.577 ± 0.011 0.762 ± 0.032	0.343 ± 0.004 0.336 ± 0.000	0.714 ± 0.009 0.605 ± 0.026
992		VAE (MTL)	$\frac{0.750 \pm 0.030}{0.750 \pm 0.021}$	$\frac{0.760 \pm 0.034}{0.750 \pm 0.020}$	$\frac{0.771 \pm 0.030}{0.777 \pm 0.022}$	$\frac{0.765 \pm 0.052}{0.754 \pm 0.022}$	0.343 ± 0.009	0.093 ± 0.020 0.727 ± 0.034
002		GRAPE (MTL)	0.757 ± 0.033	0.756 ± 0.030	0.785 ± 0.034	0.758 ± 0.030	0.371 ± 0.006	0.697 ± 0.001
993		PIG	0.952 ± 0.017	0.951 ± 0.018	0.952 ± 0.017	0.95 ± 0.018	0.084 ± 0.007	$\textbf{0.222} \pm \textbf{0.042}$
994		SI	0.906 ± 0.026 0.903 ± 0.007	0.904 ± 0.026 0.807 \pm 0.000	0.909 ± 0.026	0.905 ± 0.026	0.043 ± 0.006	-
995		MissForest	0.903 ± 0.007 0.891 ± 0.003	0.897 ± 0.009 0.885 ± 0.006	0.913 ± 0.003 0.902 ± 0.000	0.901 ± 0.003 0.888 ± 0.004	0.049 ± 0.000 0.020 ± 0.004	-
000		AE	0.899 ± 0.022	0.899 ± 0.023	0.901 ± 0.021	0.899 ± 0.022	0.148 ± 0.027	-
990		VAE	0.883 ± 0.026	0.885 ± 0.025	0.885 ± 0.025	0.882 ± 0.026	0.157 ± 0.017	-
997	Pulsar	GRAPE	0.827 ± 0.048 0.863 ± 0.009	0.829 ± 0.046 0.861 \pm 0.009	0.831 ± 0.047 0.880 \pm 0.005	0.827 ± 0.048 0.861 \pm 0.009	0.281 ± 0.084 0.053 ± 0.004	-
998		MIRACLE	0.626 ± 0.009	0.635 ± 0.024	0.642 ± 0.003	0.601 ± 0.009 0.623 ± 0.028	9.823 ± 0.136	-
000		HyperImpute	0.849 ± 0.012	0.828 ± 0.015	0.852 ± 0.007	0.836 ± 0.014	$\textbf{0.014} \pm \textbf{0.001}$	-
999		AE (MTL)	0.910 ± 0.026	0.909 ± 0.025	0.911 ± 0.025	0.910 ± 0.026	0.114 ± 0.017	-
1000		GRAPE (MIL)	0.880 ± 0.012 0.912 ± 0.023	0.880 ± 0.013 0.912 ± 0.023	0.881 ± 0.011 0.916 ± 0.022	0.879 ± 0.012 0.911 ± 0.024	0.160 ± 0.016 0.164 ± 0.011	-
1001		PIG	$\frac{0.912 \pm 0.023}{0.923 \pm 0.018}$	$\frac{0.912 \pm 0.023}{0.923 \pm 0.018}$	$\frac{0.910 \pm 0.022}{0.926 \pm 0.017}$	$\frac{0.911 \pm 0.021}{0.922 \pm 0.018}$	0.046 ± 0.002	-
1000	-	SI	0.962 ± 0.033	0.949 ± 0.047	0.970 ± 0.026	0.957 ± 0.040	0.021 ± 0.005	-
1002		MICE	0.771 ± 0.029	0.486 ± 0.046	0.453 ± 0.158	0.455 ± 0.068	0.071 ± 0.007	-
1003		AE	0.762 ± 0.000 0.943 ± 0.029	0.474 ± 0.009 0.923 ± 0.041	0.448 ± 0.144 0.959 ± 0.020	0.432 ± 0.000 0.936 ± 0.034	0.001 ± 0.009 0.086 \pm 0.019	-
1004		VAE	0.945 ± 0.024	0.928 ± 0.035	0.959 ± 0.016	0.939 ± 0.029	0.088 ± 0.020	-
1001		GRAPE	0.888 ± 0.066	0.882 ± 0.057	0.899 ± 0.059	0.880 ± 0.065	0.501 ± 0.522	-
1005	Breast Cancer	GAIN MIRACLE	0.800 ± 0.029 0.752 ± 0.029	0.500 ± 0.000 0.474 ± 0.009	0.900 ± 0.014 0.443 ± 0.130	0.444 ± 0.009 0.429 ± 0.010	0.286 ± 0.046 0.870 ± 0.233	-
1006		HyperImpute	0.752 ± 0.029 0.771 ± 0.029	0.474 ± 0.009 0.474 ± 0.009	0.453 ± 0.150 0.453 ± 0.159	0.435 ± 0.009	0.055 ± 0.019	-
1007		AE (MTL)	0.947 ± 0.021	0.932 ± 0.029	0.953 ± 0.020	0.939 ± 0.024	0.077 ± 0.011	-
4000		VAE (MTL)	0.943 ± 0.019	0.932 ± 0.025	0.946 ± 0.025	0.937 ± 0.022	0.102 ± 0.016	-
1008		PIG	0.940 ± 0.036 0.962 ± 0.029	0.912 ± 0.048 0.956 ± 0.033	0.959 ± 0.024 0.966 ± 0.025	0.928 ± 0.042 0.960 + 0.030	0.136 ± 0.027 0.010 ± 0.006	-
1009		SI	0.631 ± 0.022	0.609 ± 0.088	0.672 ± 0.040	0.591 ± 0.080	0.094 ± 0.012	-
1010		MICE	0.583 ± 0.009	0.457 ± 0.009	0.561 ± 0.017	0.465 ± 0.010	0.302 ± 0.010	-
1010		MissForest	0.617 ± 0.022	0.472 ± 0.022	$\frac{0.721 \pm 0.005}{0.005}$	0.475 ± 0.023	0.079 ± 0.002	-
1011		VAE	0.017 ± 0.032 0.635 ± 0.035	0.334 ± 0.038 0.558 + 0.061	0.029 ± 0.093 0.693 ± 0.062	0.535 ± 0.002 0.576 ± 0.052	0.241 ± 0.034 0.290 ± 0.104	-
1012		GRAPE	$\frac{0.033 \pm 0.033}{0.437 \pm 0.087}$	0.435 ± 0.090	0.415 ± 0.098	0.378 ± 0.081	1.174 ± 0.625	-
1013	Faults	GAIN	0.612 ± 0.005	0.467 ± 0.006	0.616 ± 0.006	0.469 ± 0.000	$\textbf{0.079} \pm \textbf{0.004}$	-
1010		MIRACLE	0.354 ± 0.000	0.287 ± 0.007 0.463 ± 0.016	0.287 ± 0.006 0.723 \pm 0.005	0.262 ± 0.002 0.468 \pm 0.017	9.029 ± 0.193	-
1014		AE (MTL)	0.525 ± 0.002	0.462 ± 0.010 0.462 ± 0.050	0.723 ± 0.003 0.550 ± 0.105	0.408 ± 0.017 0.470 ± 0.055	0.085 ± 0.010 0.200 ± 0.015	-
1015		VAE (MTL)	0.544 ± 0.033	0.465 ± 0.068	0.576 ± 0.087	0.477 ± 0.069	0.263 ± 0.010	-
1016		GRAPE (MTL)	0.565 ± 0.027	0.546 ± 0.055	0.547 ± 0.050	0.505 ± 0.044	0.513 ± 0.148	-
4047		PIG SI	0.692 ± 0.026 0.554 ± 0.048	0.714 ± 0.048 0.393 ± 0.101	0.690 ± 0.042 0.759 ± 0.071	0.086 ± 0.042 0.383 ± 0.105	0.085 ± 0.134 0.067 ± 0.011	-
1017		MICE	0.543 ± 0.016	0.230 ± 0.045	0.697 ± 0.045	0.309 ± 0.109 0.210 ± 0.045	0.061 ± 0.004	-
1018		MissForest	0.539 ± 0.026	0.213 ± 0.014	0.828 ± 0.040	0.192 ± 0.012	0.047 ± 0.006	-
1019		AE	0.541 ± 0.030	0.395 ± 0.108	0.800 ± 0.055	0.381 ± 0.109	0.080 ± 0.018	-
4000		GRAPE	$\frac{0.301 \pm 0.046}{0.511 \pm 0.046}$	0.420 ± 0.150 0.375 ± 0.111	0.797 ± 0.069 0.606 ± 0.151	0.415 ± 0.152 0.336 ± 0.123	0.080 ± 0.019 0.286 ± 0.085	-
1020	Wine	GAIN	0.537 ± 0.008	0.233 ± 0.045	0.847 ± 0.007	0.207 ± 0.046	0.050 ± 0.001	-
1021		MIRACLE	0.491 ± 0.013	0.243 ± 0.043	0.807 ± 0.043	0.253 ± 0.043	9.623 ± 0.012	-
1022		HyperImpute	0.534 ± 0.010 0.563 \pm 0.042	0.227 ± 0.055 0.290 ± 0.087	$\frac{0.843 \pm 0.005}{0.716 \pm 0.084}$	0.207 ± 0.055 0.282 ± 0.002	$\frac{0.043 \pm 0.002}{0.083 \pm 0.002}$	-
1000		VAE (MTL)	0.558 ± 0.042	0.374 ± 0.111	0.758 ± 0.084	0.352 ± 0.093	0.083 ± 0.008 0.084 ± 0.013	-
1023		GRAPE (MTL)	0.561 ± 0.029	0.358 ± 0.118	0.814 ± 0.071	0.337 ± 0.121	0.138 ± 0.050	-
1024		PIG	0.561 ± 0.020	0.405 ± 0.078	0.775 ± 0.037	0.397 ± 0.080	$\textbf{0.013} \pm \textbf{0.003}$	-

Κ **EXPERIMENTAL RESULTS FOR SENSITIVITY ANALYSIS**

We measured how sensitive each model is to missing rates. We varied the missing rates from 5% to 50% and reported the results. The sensitivity analysis was implemented using the "Mobile" dataset.

PIG showed remarkable robustness to various missing rates compared to baseline models. Here, we present the results of a sensitivity analysis for some representative methods, SI, GAIN, GRAPE, and PIG (Figure 10). While PIG also exhibits performance degradation as missing rates increase, the other methods quickly saturate to lower performance. Refer to Table 9 for more detail.



Figure 10: Sensitivity analysis on different missing rates. Left: classification performance (F1score). Center: imputation performance (MSE) for numeric features. Right: imputation perfor-mance (CE) for categorical features.

1081 Classification Imputation 1082 Missing rates Model MSE CE Accuracy Recall Precision F1-score 1083 SI 0.319 ± 0.024 0.680 ± 0.027 0.679 ± 0.041 0.675 ± 0.041 0.670 ± 0.042 0.678 ± 0.038 1084 MICE 0.622 ± 0.019 0.683 ± 0.012 0.532 ± 0.011 0.540 ± 0.013 0.548 ± 0.019 0.518 ± 0.011 MissForest 0.319 ± 0.017 0.686 ± 0.015 0.559 ± 0.003 0.567 ± 0.005 0.564 ± 0.005 0.545 ± 0.004 1085 0.344 ± 0.019 0.679 ± 0.019 0.658 ± 0.050 AE 0.656 ± 0.050 0.662 ± 0.048 0.651 ± 0.050 0.678 ± 0.044 0.667 ± 0.044 VAE 0.328 ± 0.019 0.668 ± 0.019 0.677 ± 0.041 0.674 ± 0.044 1086 AE (MTL) 0.339 ± 0.026 0.708 ± 0.050 0.892 ± 0.019 0.893 ± 0.018 0.892 ± 0.019 0.891 ± 0.019 0.05 0.349 ± 0.023 0.754 ± 0.043 0.892 ± 0.024 0.892 ± 0.024 VAE (MTL) 0.893 ± 0.025 0.896 ± 0.023 1087 GRAPE (MTL) 0.367 ± 0.005 0.697 ± 0.005 0.897 ± 0.027 0.897 ± 0.030 0.902 ± 0.025 0.894 ± 0.030 GRAPE 1.272 ± 0.386 0.696 ± 0.087 0.893 ± 0.020 $\overline{0.891\pm0.022}$ 0.902 ± 0.017 0.892 ± 0.021 MIRACLE 13.732 ± 0.030 0.691 ± 0.001 0.417 ± 0.024 0.424 ± 0.023 0.479 ± 0.027 0.394 ± 0.026 1089 0.340 ± 0.005 0.686 ± 0.012 0.568 ± 0.007 0.577 ± 0.008 0.568 ± 0.008 0.556 ± 0.007 HyperImpute GAIN 0.407 ± 0.008 0.407 ± 0.009 0.587 ± 0.005 0.588 ± 0.001 0.584 ± 0.001 0.581 ± 0.001 1090 0.040 ± 0.001 0.917 ± 0.018 0.918 ± 0.016 0.923 ± 0.019 0.917 ± 0.018 PIG 0.169 ± 0.056 1091 0.673 ± 0.023 0.671 ± 0.040 0.658 ± 0.043 0.318 ± 0.019 0.671 ± 0.042 0.669 ± 0.040 0.571 ± 0.006 MICE 0.679 ± 0.006 0.564 ± 0.012 0.578 ± 0.003 0.551 ± 0.012 0.613 ± 0.022 MissForest 0.343 ± 0.003 0.680 ± 0.001 0.578 ± 0.007 0.585 ± 0.009 0.584 ± 0.010 0.566 ± 0.005 AE 0.344 ± 0.021 0.672 ± 0.019 0.642 ± 0.064 0.642 ± 0.064 0.637 ± 0.062 0.633 ± 0.064 1093 VAE 0.342 ± 0.016 0.672 ± 0.017 0.644 ± 0.050 0.649 ± 0.044 0.642 ± 0.052 0.637 ± 0.052 1094 AE (MTL) VAE (MTL) 0.340 ± 0.013 0.682 ± 0.042 $\frac{0.853 \pm 0.026}{0.831 \pm 0.024}$ $\frac{0.852 \pm 0.026}{0.831 \pm 0.027}$ $\frac{0.860 \pm 0.024}{0.840 \pm 0.022}$ $\frac{0.852 \pm 0.026}{0.831 \pm 0.025}$ 0.10 0.339 ± 0.017 0.738 ± 0.036 1095 0.367 ± 0.005 0.840 ± 0.025 GRAPE (MTL) 0.698 ± 0.001 0.841 ± 0.024 0.838 ± 0.025 0.853 ± 0.026 GRAPE 1.085 ± 0.418 0.714 ± 0.028 0.802 ± 0.037 0.803 ± 0.038 0.815 ± 0.034 0.799 ± 0.039 1096 0.382 ± 0.008 0.391 ± 0.012 MIRACLE 12.720 ± 0.521 0.691 ± 0.001 0.441 ± 0.017 0.335 ± 0.014 0.319 ± 0.001 0.676 ± 0.015 0.575 ± 0.002 0.565 ± 0.004 HyperImpute 0.581 ± 0.006 0.578 ± 0.008 1097 $\begin{array}{c} 0.605 \pm 0.020 \\ \textbf{0.915} \pm \textbf{0.018} \end{array}$ 0.598 ± 0.019 GAIN 0.374 ± 0.002 0.700 ± 0.010 0.604 ± 0.019 0.603 ± 0.020 1098 0.041 ± 0.002 0.171 ± 0.047 0.915 ± 0.019 PIG 0.916 ± 0.018 0.921 ± 0.019 0.339 ± 0.020 0.670 ± 0.014 0.631 ± 0.043 0.633 ± 0.045 0.630 ± 0.046 0.623 ± 0.045 1099 $\begin{array}{c} 0.690 \pm 0.016 \\ 0.680 \pm 0.006 \end{array}$ $\begin{array}{c} 0.515 \pm 0.001 \\ 0.581 \pm 0.004 \end{array}$ MICE 0.737 ± 0.030 0.506 ± 0.005 0.504 ± 0.004 0.493 ± 0.008 MissForest 0.329 ± 0.001 0.576 ± 0.002 0.584 ± 0.005 0.568 ± 0.000 1100 0.336 ± 0.020 0.679 ± 0.007 0.629 ± 0.062 0.630 ± 0.056 0.625 ± 0.060 0.618 ± 0.061 AE 1101 $\begin{array}{c} 0.637 \pm 0.050 \\ 0.795 \pm 0.022 \end{array}$ $\begin{array}{c} 0.622 \pm 0.051 \\ 0.798 \pm 0.022 \end{array}$ VAE 0.344 ± 0.016 0.673 ± 0.012 0.628 ± 0.052 0.629 ± 0.052 AE (MTL) 0.342 ± 0.012 0.696 ± 0.020 0.796 ± 0.026 0.809 ± 0.023 1102 0.15 VAE (MTL 0.347 ± 0.014 0.744 ± 0.040 0.799 ± 0.026 0.797 ± 0.026 0.818 ± 0.025 0.800 ± 0.026 GRAPE (MTL) 0.366 ± 0.007 0.695 ± 0.010 0.779 ± 0.022 0.780 ± 0.022 0.797 ± 0.021 0.780 ± 0.021 1103 1.380 ± 0.250 0.757 ± 0.049 0.755 ± 0.055 0.739 ± 0.049 0.757 ± 0.052 0.791 ± 0.045 GRAPE MIRACLE 13.192 ± 0.363 0.692 ± 0.001 0.444 ± 0.003 0.447 ± 0.007 0.519 ± 0.007 0.406 ± 0.001 1104 0.312 ± 0.011 $\frac{0.312\pm0.000}{0.691\pm0.007}$ 0.545 ± 0.008 HyperImpute 0.552 ± 0.007 0.560 ± 0.008 0.565 ± 0.002 1105 0.428 ± 0.002 0.553 ± 0.008 0.544 ± 0.008 GAIN 0.548 ± 0.010 0.550 ± 0.011 0.043 ± 0.003 0.204 ± 0.047 0.910 ± 0.018 0.910 ± 0.018 0.912 ± 0.019 $\textbf{0.909} \pm \textbf{\underline{0.018}}$ PIG 1106 0.334 ± 0.015 0.676 ± 0.012 0.590 ± 0.041 0.594 ± 0.038 0.588 ± 0.040 0.582 ± 0.041 SI MICE 0.561 ± 0.017 0.682 ± 0.005 0.567 ± 0.010 0.575 ± 0.011 0.569 ± 0.009 0.558 ± 0.011 1107 MissForest 0.352 ± 0.013 0.683 ± 0.009 0.558 ± 0.008 0.566 ± 0.006 0.567 ± 0.010 0.547 ± 0.004 1108 AE VAE 0.348 ± 0.015 $\begin{array}{c} 0.676 \pm 0.008 \\ 0.665 \pm 0.009 \end{array}$ 0.609 ± 0.047 $\begin{array}{c} 0.610 \pm 0.048 \\ 0.611 \pm 0.070 \end{array}$ $\begin{array}{c} 0.605 \pm 0.053 \\ 0.606 \pm 0.076 \end{array}$ $\begin{array}{c} 0.601 \pm 0.052 \\ 0.603 \pm 0.077 \end{array}$ 0.336 ± 0.013 0.616 ± 0.077 1109 0.336 ± 0.009 AE (MTL) 0.695 ± 0.026 0.758 ± 0.036 0.760 ± 0.034 0.791 ± 0.030 0.763 ± 0.032 0.20 $\begin{array}{c} 0.727 \pm 0.034 \\ 0.697 \pm 0.001 \end{array}$ VAE (MTL) 0.343 ± 0.015 0.750 ± 0.021 0.777 ± 0.022 0.754 ± 0.022 1110 GRAPE (MTL) 0.371 ± 0.006 0.757 ± 0.033 0.758 ± 0.030 0.785 ± 0.034 0.724 ± 0.038 1111 1.431 ± 0.231 0.727 ± 0.041 0.649 ± 0.036 GRAPE 0.653 ± 0.034 0.658 ± 0.030 12.429 ± 0.107 0.340 ± 0.004 MIRACLE 0.691 ± 0.000 0.373 ± 0.006 0.377 ± 0.010 0.407 ± 0.004 0.327 ± 0.005 1112 0.578 ± 0.016 0.677 ± 0.009 0.572 ± 0.009 0.579 ± 0.019 0.562 ± 0.015 HyperImpute GAIN 0.445 ± 0.004 0.692 ± 0.002 0.530 ± 0.032 0.523 ± 0.035 0.521 ± 0.036 0.518 ± 0.033 1113 $\textbf{0.046} \pm \textbf{0.002}$ PIG 0.222 ± 0.042 0.923 ± 0.018 0.923 ± 0.018 $\textbf{0.926} \pm \textbf{0.017}$ 0.922 ± 0.018 0.343 ± 0.011 0.682 ± 0.011 0.562 ± 0.048 0.556 ± 0.052 1114 SI 0.565 ± 0.052 0.565 ± 0.048 MICE 0.630 ± 0.020 0.674 ± 0.005 0.521 ± 0.018 0.528 ± 0.015 0.518 ± 0.021 0.513 ± 0.018 1115 MissForest 0.330 ± 0.006 0.687 ± 0.006 0.526 ± 0.018 0.535 ± 0.015 0.513 ± 0.021 0.506 ± 0.023 0.335 ± 0.014 0.675 ± 0.009 0.567 ± 0.028 0.561 ± 0.026 0.570 ± 0.028 0.565 ± 0.024 AE 1116 VAE 0.570 ± 0.038 0.340 ± 0.010 0.677 ± 0.006 0.576 ± 0.036 0.575 ± 0.039 0.571 ± 0.043 AE (MTL) VAE (MTL) $\begin{array}{c} 0.339 \pm 0.004 \\ 0.345 \pm 0.010 \end{array}$ $\begin{array}{c} 0.677 \pm 0.028 \\ 0.669 \pm 0.022 \end{array}$ 0.705 ± 0.023 0.679 ± 0.026 0.731 ± 0.024 0.685 ± 0.026 1117 0.716 ± 0.034 0.667 ± 0.024 0.677 ± 0.022 0.30 0.734 ± 0.025 1118 0.367 ± 0.005 0.698 ± 0.001 0.745 ± 0.022 0.698 ± 0.027 GRAPE (MTL) 0.694 ± 0.026 0.695 ± 0.024 $\begin{array}{c} 1.434 \pm 0.269 \\ 14.066 \pm 0.116 \end{array}$ $\begin{array}{c} 0.741 \pm 0.040 \\ 0.692 \pm 0.000 \end{array}$ $\frac{0.562 \pm 0.049}{0.416 \pm 0.003}$ $\frac{0.668 \pm 0.065}{0.473 \pm 0.008}$ $\frac{0.552 \pm 0.056}{0.391 \pm 0.003}$ GRAPE 0.562 ± 0.047 1119 0.409 ± 0.001 MIRACLE 0.336 ± 0.006 0.680 ± 0.009 0.560 ± 0.002 0.568 ± 0.005 0.550 ± 0.002 0.540 ± 0.002 HyperImpute 1120 $\begin{array}{c} 0.376 \pm 0.001 \\ \textbf{0.052} \pm \textbf{0.003} \end{array}$ $\begin{array}{c} 0.697 \pm 0.008 \\ \textbf{0.295} \pm \textbf{0.023} \end{array}$ $\begin{array}{c} 0.504 \pm 0.003 \\ \textbf{0.897} \pm \textbf{0.018} \end{array}$ $\begin{array}{c} 0.507 \pm 0.001 \\ \textbf{0.897} \pm \textbf{0.018} \end{array}$ $\begin{array}{c} 0.493 \pm 0.000 \\ \textbf{0.899} \pm \textbf{0.017} \end{array}$ $\begin{array}{c} 0.494 \pm 0.001 \\ \textbf{0.896} \pm \textbf{0.018} \end{array}$ GAIN 1121 PIG 0.347 ± 0.009 0.678 ± 0.009 0.536 ± 0.041 0.528 ± 0.039 0.534 ± 0.036 0.531 ± 0.036 1122 MICE MissForest ${ 0.611 \pm 0.001 \atop 0.363 \pm 0.009 }$ $\begin{array}{c} 0.682 \pm 0.004 \\ 0.687 \pm 0.011 \end{array}$ 0.549 ± 0.002 $\begin{array}{c} 0.558 \pm 0.002 \\ 0.573 \pm 0.008 \end{array}$ $0.546 \pm 0.004 \\ 0.571 \pm 0.004$ $\begin{array}{c} 0.540 \pm 0.003 \\ 0.552 \pm 0.004 \end{array}$ 0.564 ± 0.004 1123 0.337 ± 0.009 0.674 ± 0.005 0.522 ± 0.044 0.530 ± 0.049 0.522 ± 0.045 AE 0.520 ± 0.046 1124 VAE 0.339 ± 0.008 0.672 ± 0.006 0.522 ± 0.042 0.522 ± 0.037 0.523 ± 0.044 0.518 ± 0.042 0.712 ± 0.015 0.616 ± 0.033 $\frac{0.627 \pm 0.033}{0.615 \pm 0.036}$ AE (MTL) 0.337 ± 0.014 0.618 ± 0.034 0.699 ± 0.024 1125 0.40 VAE (MTL) 0.344 ± 0.012 0.734 ± 0.036 0.680 ± 0.036 0.606 ± 0.037 0.606 ± 0.039 $\begin{array}{c} 0.608 \pm 0.025 \\ 0.487 \pm 0.081 \end{array}$ GRAPE (MTL) 0.366 ± 0.007 0.697 ± 0.001 0.600 ± 0.024 0.599 ± 0.022 0.690 ± 0.040 1126 1.220 ± 0.342 0.722 ± 0.033 0.507 ± 0.070 0.517 ± 0.066 0.617 ± 0.060 GRAPE 0.386 ± 0.003 0.633 ± 0.021 1127 MIRACLE 13.896 ± 0.318 0.692 ± 0.000 0.383 ± 0.009 0.479 ± 0.007 0.361 ± 0.002 HyperImpute 0.336 ± 0.008 0.336 ± 0.007 0.627 ± 0.009 0.641 ± 0.023 0.614 ± 0.017 1128 0.585 ± 0.000 0.585 ± 0.000 GAIN 0.363 ± 0.019 0.688 ± 0.002 0.586 ± 0.003 0.581 ± 0.002 0.065 ± 0.003 0.392 ± 0.040 $\textbf{0.894} \pm \textbf{0.022}$ $\textbf{0.892} \pm \textbf{0.025}$ 0.895 ± 0.022 $\textbf{0.892} \pm \textbf{0.024}$ PIG 1129 $\begin{array}{c} 0.355 \pm 0.006 \\ 0.594 \pm 0.008 \end{array}$ 0.476 ± 0.033 0.542 ± 0.014 0.683 ± 0.005 0.473 ± 0.032 0.481 ± 0.041 0.468 ± 0.037 MICE 0.528 ± 0.016 1130 0.683 ± 0.009 0.536 ± 0.018 0.548 ± 0.012 0.356 ± 0.006 0.686 ± 0.013 0.515 ± 0.015 0.522 ± 0.011 0.517 ± 0.010 0.500 ± 0.015 MissForest 1131 AE VAE 0.343 ± 0.005 0.670 ± 0.005 0.470 ± 0.026 0.471 ± 0.026 0.495 ± 0.036 0.472 ± 0.028 0.465 ± 0.030 0.336 ± 0.006 0.469 ± 0.028 0.468 ± 0.027 0.483 ± 0.034 0.670 ± 0.008 1132 AE (MTL) 0.337 ± 0.007 0.714 ± 0.024 0.534 ± 0.028 0.533 ± 0.025 0.649 ± 0.032 0.547 ± 0.027

1080 Table 9: Sensitivity analysis to different missing rates. Bold: best score. Underline: runner-up.

 0.533 ± 0.024

 0.521 ± 0.038

 0.528 ± 0.027

 0.525 ± 0.031

 0.639 ± 0.024

 0.639 ± 0.023

 0.536 ± 0.033

 0.528 ± 0.034

 0.720 ± 0.021

 0.694 ± 0.010

1133

0.50

VAE (MTL

GRAPE (MTL)

 0.342 ± 0.007

 0.368 ± 0.008

Missing rates	Model	Impu	CE	Accuracy	Recall	Precision	F1-score
	GRAPE	1.011 ± 0.467 13.425 ± 0.087	0.705 ± 0.031 0.691 ± 0.001	0.474 ± 0.061 0.361 ± 0.018	0.478 ± 0.049 0.366 ± 0.012	0.598 ± 0.045 0.412 ± 0.015	0.453 ± 0.071
	HyperImpute	$\frac{0.323 \pm 0.001}{0.323 \pm 0.001}$	0.691 ± 0.001 0.684 ± 0.015	0.549 ± 0.002	0.300 ± 0.012 0.557 ± 0.006	0.412 ± 0.013 0.555 ± 0.008	0.530 ± 0.012 0.539 ± 0.004
	GAIN PIG	0.432 ± 0.015 0.083 ± 0.007	0.696 ± 0.007 0.494 ± 0.012	$\frac{0.599 \pm 0.014}{0.863 \pm 0.021}$	$\frac{0.600 \pm 0.014}{0.862 \pm 0.022}$	0.598 ± 0.018 0.864 ± 0.020	$\frac{0.582 \pm 0.015}{0.861 \pm 0.021}$
	-						