Generating MNAR Missingness in Image Data, with Additional Evaluation of MisGAN

Natasha T.J. van den Berg*, Bram O. Broekgaarden*, Dionysia P.A. Mahieu*, Jolijn G.M.J. Martens*, Jonas M. Niederle*, Rianne M. Schouten(✉), and Wouter Duivesteijn

Eindhoven University of Technology, the Netherlands
{n.t.j.v.d.berg,b.o.broekgaarden,d.p.a.mahieu,j.g.m.j.martens,j.m.niederle}@student.tue.nl
{r.m.schouten,w.duivesteijn}@tue.nl

Abstract. To the question of missing values in image data, MisGAN [16] provides an answer based on a Generative Adversarial Network (GAN). Traditionally, in the study of missing data, three main underlying mechanisms are distinguished: MCAR, MAR and MNAR. The MisGAN paper assumes that the missingness follows the MCAR mechanism and empirically shows that MisGAN performs well on image data generated with MCAR missingness, leaving MAR and MNAR as future work. In this paper, we propose a method for generating MNAR missingness in the MNIST dataset: we let higher gray-scale pixel values have a higher probability of being missing. In addition, we vary the extent to which these missingness probabilities depend on the pixel values and investigate the effect of simultaneously occurring mechanisms. Indeed, we find that MisGAN is not working quite as well on MNAR data as it is on MCAR data. In addition, we make auxiliary comments about result evaluation using the Fréchet Inception Distance, and discuss the difficulty of defining a pixel-level MAR missingness mechanism in image data.

Keywords: Missing Data in Images · Data Amputation · MNAR · Missingness Mechanisms · Generative Adversarial Networks

1 Introduction

Many data mining methods simply assume that we have a large, complete dataset available for analysis. In real life, one often comes across missingness in the data due to failure of data collection or lost records. If the missingness is not handled well, there is a large probability that the incomplete data leads to incorrect or unrealistic results. A number of studies have been conducted on missing data imputation, to impute the missing values with plausible data. Various techniques dealing with machine learning and deep learning have been studied [15,30]. Among them, the most frequently presented models for image data are those based on Generative Adversarial Networks (GANs) [11]. Each of the GAN-based models focuses on improving some specific aspect of missing data imputation [15]. One

* These authors have contributed equally to this paper.

of these models is *MisGAN* [16], a GAN-based framework for learning from complex, high-dimensional, incomplete data. MisGAN performs quite well on data in which the missingness is independent from observed or unobserved information, better known as the *Missing Completely at Random* (MCAR) mechanism. Although the future work section of [16] suggests that the framework could also work for missingness that is dependent on the missing or non-missing values, no results are shown. As the underlying mathematical model does not necessarily hold when allowing missingness to be dependent, it remains an open research question to see whether MisGAN can indeed be successfully applied under missingness mechanisms other than MCAR. In this paper, we empirically explore MisGAN's performance on data where the missingness is generated through the *Missing Not At Random* (MNAR) mechanism.

2 Background and Related Work

Consider dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{m}_i)\}_{i=1,2,...,N}$ to be a collection of N partially observed samples. Each sample consists of a partially observed data vector $\mathbf{x} \in \mathbb{R}^n$ and a missing data indicator $\mathbf{m} \in \{0,1\}^n$ indicating which entries in $\mathbf{x}$ are observed; $m_d = 1$ if x_d is observed and $m_d = 0$ if x_d is missing, for $d = 1, 2, ..., n$. It is thus possible to split the data into $\mathbf{x}_{\text{obs}}$ and $\mathbf{x}_{\text{mis}}$, representing the observed and missing values, respectively.

2.1 Missing Data Mechanisms

In the study of missing data, we call the process that governs the missingness probabilities the *missing data model* or *missing data mechanism*: $\mathbf{m} \sim p_\phi(\mathbf{m}|\mathbf{x}) = p_\phi(\mathbf{m}|\mathbf{x}_{\text{obs}}, \mathbf{x}_{\text{mis}})$ [3,17,22,23]. Distinguishing between missing data models is important for understanding underlying reasons for the prevalence of missingness in the data, as well as for determining which missing data methods are applicable to and valid for the data at hand.

Traditionally, we distinguish three types of missingness mechanisms: MCAR, MAR, and MNAR [22,23]. Data is said to be *Missing Completely At Random* (MCAR) if the probability of being missing depends on some fixed parameters, and is unrelated to the observed and missing data distribution:

$$p_\phi(\mathbf{m}|\mathbf{x}) = p_\phi(\mathbf{m})$$

The consequence is that the missing values are not different from observed values. Data is *Missing At Random* (MAR) if the probability to be missing depends on observed data:

$$p_\phi(\mathbf{m}|\mathbf{x}) = p_\phi(\mathbf{m}|\mathbf{x}_{\text{obs}})$$

Even though MAR missingness may create severely biased data, the information about the missing values is available in the dataset and can be used to obtain valid statistical inference. This concept is known as *ignorability* [22]. Finally,

data is *Missing Not At Random* (MNAR) if the information about the missing values is missing from the data. Specifically, we write

$$p_\phi(\mathbf{m}|\mathbf{x}) = p_\phi(\mathbf{m}|\mathbf{x}_{\text{mis}}, \mathbf{x}_{\text{obs}})$$

to indicate that (at least some of) the information about the missingness probabilities depends on the missing values themselves.

In practice, these mechanisms are not as distinct as the theoretical definitions imply. For instance, we say missing data is *partially* or *latently* MAR (instead of MNAR) when it is possible to model the missingness using information in the missing data indicator $\mathbf{m}$ [12]. Furthermore, for bivariate, numerical data, strictly distinct missingness mechanisms may yield equivalent statistical inferences [28]. Particularly, MAR missingness becomes indistinguishable from MCAR missingness when data correlations are low; when data relations are strong, it may be unnecessary to assume MNAR since the missing data can be described well using observed data information [28]. In addition, missingness mechanisms may occur concurrently. When a numerical feature contains a mixture of multiple missingness mechanisms, the effect on the distribution of that feature is an additive combination of the effects of the separate mechanisms [27].

To evaluate the effectiveness of missing data methods, some form of ground truth must be established. To that end, we distinguish design-based simulation, where complete sub-samples from real-world datasets are used, from model-based simulation, where simulation data is drawn from a known probability distribution [20]. For both approaches, we require a methodology for artificially generating missing values in complete data: the amputation methodology. An amputation procedure exists for multivariate, numerical data [26]. There, MAR and MNAR missingness probabilities are conditioned on linear combinations of observed and unobserved features, respectively, and the effect on the distribution of an incomplete feature is manipulated by applying one of four variations of the logistic function (the LEFT, RIGHT, MID, and TAIL type) [26].

For image data, such an amputation framework does not really exist. One could create an MCAR mechanism: the probability that a pixel value is missing is fixed; it is similar for every pixel value. MisGAN [16] assumes this type of missingness mechanism. In case of an MNAR mechanism, we propose an approach where the probability that a pixel value is missing depends on the pixel value itself. For instance, all red pixels are missing. In this paper, we let lighter pixel values have a higher probability to be missing than darker pixels: our amputation approach would impact the quality of MNIST digits.

2.2 Data Imputation Methods

The act of replacing a missing value by an actual value ("filling in the blanks", colloquially speaking) is known as data *imputation*. The simplest imputation methods are single, univariate imputation methods, where each missing value is imputed once using a fixed value; examples include zero imputation, where a zero is imputed each time a data entry is missing, and mean imputation, where

each missing entry is imputed by the mean value of the observed entries for that column [35], or by creating a 'prediction' model that predicts the missing values using observed data information (cf. [3] for a collection of examples).

Multivariate imputation methods assume the missing data has a monotone structure [23], can be modeled using a joint probability distribution [25], or use an iterative approach such as Fully Conditional Specification [4] (also known as MICE: Multivariate Imputation by Chained Equations). When missing values are replaced more than once, we speak of *multiple imputation* [5].

Deep learning methods for missing data imputation include Denoising AutoEncoders (DAEs) [10,33] and methods based on Generative Adversarial Networks (GAN) [11] such as MisGAN [16], developed for image data, and GAIN [34], developed for numerical data. Most currently existing deep learning imputation methods are based on the MCAR assumption (with not-MIWAE [14] as a notable counterexample).

2.3 Generative Adversarial Networks and MisGAN

A Generative Adversarial Network (GAN) [11] is a framework that estimates generative models via an adversarial process. In general, GANs are characterized by training a pair of networks (a generator and a discriminator) that are in competition with each other [11]. The generator G tries to fool the discriminator D by generating false data that mimics the real data distribution as closely as possible. The discriminator tries to detect the generator's actions, by classifying the data as being real or created by the generator. The GAN framework corresponds to a two-player minimax game with value function $V(D, G)$ where discriminator D is trained to maximize the probability of assigning the correct label to both training examples and generated samples, and generator G is trained to minimize $\log(1 - D(G(\mathbf{z})))$ (i.e., maximize the probability of D making a mistake) [11]:

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))].$$

Here, $p_z(\mathbf{z})$ is a prior on input noise variables.

GANs are useful for image, video, and voice generation. One of their core capabilities is image synthesis: creating new images from some form of image description [6]. GANs can also be used for super resolution: the process of generating high-resolution images from lower-resolution images, and for artistic style transfer, which renders natural images in the style of artists [36].

MisGAN [16] is a GAN-based framework for learning high-dimensional incomplete data that can be used for imputation (cf. Figure 1). Denoting the incomplete data as a pair of a partially-observed data vector $\mathbf{x}$ and a corresponding mask $\mathbf{m}$ (cf. start of Section 2), MisGAN starts by defining a masking operator f_τ, which fills the missing entries with a constant value τ. MisGAN then employs two distinct GANs: the missing data process is explicitly modeled using the mask generator G_m, and the complete data generator G_x is trained

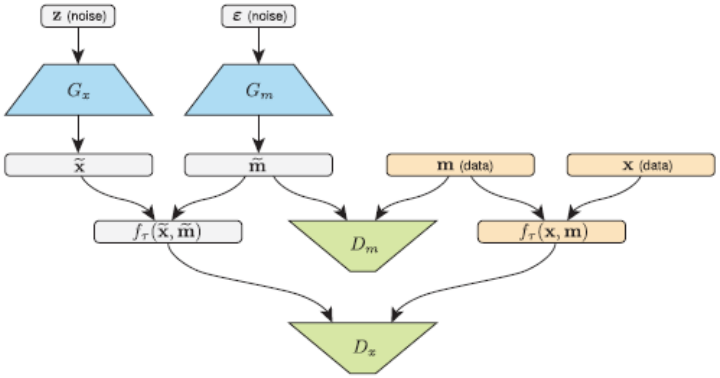


Fig. 1. Learning from incomplete data with MisGAN. Image taken from [16].

by comparing the real incomplete data with the generated incomplete data. The following are the loss functions of the mask and data GAN, respectively:

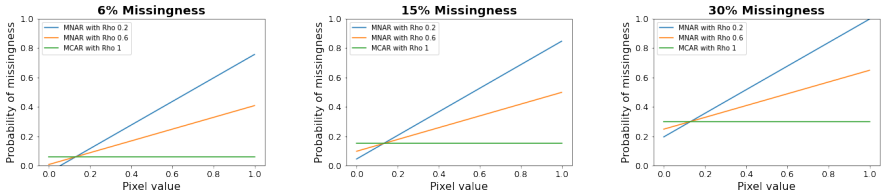
$$\begin{aligned}\mathcal{L}_m\left(D_m, G_m\right) &= \mathbb{E}_{(\mathbf{x},\mathbf{m})\sim p_{\mathcal{D}}}\left[D_m(\mathbf{m})\right] - \mathbb{E}_{\boldsymbol{\varepsilon}\sim p_\varepsilon}\left[D_m\left(G_m(\varepsilon)\right)\right] \\ \mathcal{L}_x\left(D_x, G_x, G_m\right) &= \mathbb{E}_{(\mathbf{x},\mathbf{m})\sim p_{\mathcal{D}}}\left[D_x\left(f_\tau(\mathbf{x},\mathbf{m})\right)\right] \\ &\quad - \mathbb{E}_{\boldsymbol{\varepsilon}\sim p_\varepsilon, \mathbf{z}\sim p_z}\left[D_x\left(f_\tau\left(G_x(\mathbf{z}), G_m(\varepsilon)\right)\right)\right].\end{aligned}$$

In addition to learning from incomplete data, MisGAN can be used as an imputer. Then, an additional GAN is created (G_i, D_i) where the generator outputs the completed samples with the imputed entries. For an image of the structure of the framework, see [16, Figure 2].

Note that the two generators are mutually independent, both with their own noise distributions p_z and p_ϵ. This is where the assumption of MCAR missingness occurs: the missingness of the data does not depend on any values in the data. If this MCAR missingness assumption is violated, the proofs of the theoretical results fortifying MisGAN no longer hold [16, cf. Section 3 and Appendix A]. This implies that the training objective for MisGAN is no longer theoretically justified for the missing data problem. Of course, that does not necessarily guarantee that MisGAN will fail when the missingness is not MCAR. In this paper we evaluate the degree to which MisGAN still works when the missingness is MNAR.

3 Generating MNAR Missingness in Image Data

We evaluate MisGAN on MNAR generated data using the Modified National Institute of Standards and Technology (MNIST) dataset [9]. It consists of $N = 60000$ training examples of handwritten digits images of size 28 x 28 pixels. We use the provided aligned and cropped images and re-scale the pixel values to $[0, 1]$ (black to white). The average pixel value over the entire dataset is 0.131.



(a) avg. missingness: 6% (b) avg. missingness: 15% (c) avg. missingness: 30%

Fig. 2. Effect of pixel value on the missingness probability for various ρ and c values.

3.1 Generating MNAR Missingness in MNIST

We generate missing data in the MNIST dataset that adheres to the MNAR missingness principles: the probability to be missing depends on the missing value itself. We thus let the probability to be missing depend on the grayscale values of the pixels in the images. In contrast, generating MCAR missingness would be done by applying a fixed probability to all pixels.

In addition, our approach contains a scale parameter ρ that allows us to vary the extent to which the information about the missingness is missing. In other words, we do not merely generate a pure MCAR or MNAR mechanism, but rather create a mixture of the two mechanisms. Note that, although in practice this implies that not all information about the missing values is lost, theoretically a little MNAR is still MNAR (cf. Section 2.1). It is merely the presentation in the data that is affected. We employ the following amputation procedure:

$$P_i = (\mu_i \cdot \mathrm{J} - X_i)\rho + X_i + c \cdot \mathrm{J} \tag{1}$$

Here, we overload the generic notation[1] and denote a complete sample image as a matrix $X_i \in [0,1]^{28\times 28}$. The mean pixel value of image i is denoted with μ_i, J is the all-ones matrix, c is a constant value and ρ is our *scale* parameter in the range $[0,1]$. For every image i, we then obtain a matrix $P_i \in [0,1]^{28\times 28}$ that contains the missingness probabilities per pixel (we post process the probabilities to ensure that they are in the interval $[0,1]$ by setting all sub-zero values to zero, and all values above one to one). Lastly, we sample from a Bernoulli distribution using these p-values to determine whether or not a pixel is missing:

$$\text{a pixel is} \begin{cases} \text{missing} & \text{with probability } p \\ \text{not missing} & \text{with probability } 1-p \end{cases} \tag{2}$$

[1] Sample X_i differs from $\mathbf{x}_i$ in Section 2 in two ways: 1) X_i is a matrix rather than a vector, 2) X_i is *complete*, whereas $\mathbf{x}_i$ contains missing values. Only after we use the Bernoulli distribution to translate the probability matrix P_i to a mask matrix ($\mathbf{M}_i$), we know which values in X_i are missing. This incomplete sample image $\mathbf{X}_i$ is the actual input of MisGAN. The imputed image that MisGAN outputs is then denoted by $\hat{\mathbf{X}}_i$.

Table 1. Experimental setup (amputation and simulation conditions) and MSE scores (summary statistics; lower is better) over nine test settings.

Test case	Amputation			Mean Squared Error		
	ρ	c	avg. miss.	Mean	Std	Median
MCAR_06	1	-0.071	6%	0.515	0.076	0.518
MCAR_15	1	0.019	15%	0.515	0.077	0.518
MCAR_30	1	0.169	30%	0.515	0.076	0.518
MNAR_06_0.6	0.6	-0.071	6%	0.545	0.066	0.544
MNAR_15_0.6	0.6	0.019	15%	0.549	0.065	0.548
MNAR_30_0.6	0.6	0.169	30%	0.558	0.063	0.557
MNAR_06_0.2	0.2	-0.071	6%	0.587	0.061	0.586
MNAR_15_0.2	0.2	0.019	15%	0.599	0.060	0.598
MNAR_30_0.2	0.2	0.169	30%	0.626	0.063	0.625

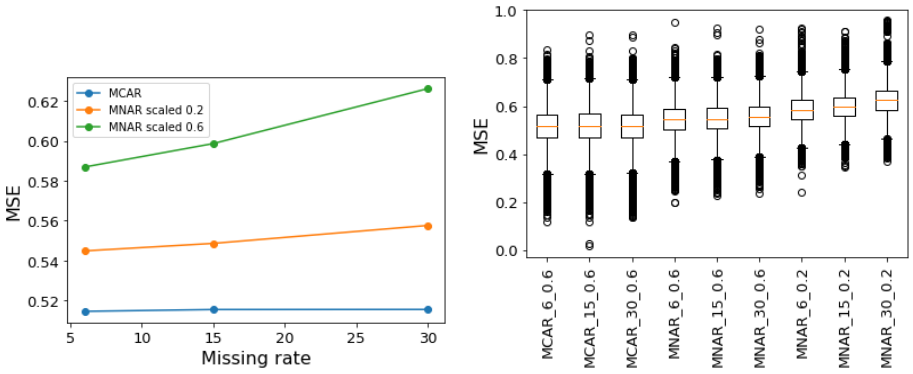
In Equation (1), parameter ρ determines the extent to which the missingness depends on the missing pixel value; the amount of MCAR missingness mixed with MNAR. If ρ equals 1, we generate a pure MCAR missingness mechanism. Then, the probability of missingness for each pixel is exactly similar for all pixels: close to $\mu_i + c$ (in Section 2.1, this fixed probability is denoted by ϕ). Figure 2 demonstrates the generation of MCAR missingness with the horizontal (green) lines; there is no relation between pixel values and probability of being missing. The authors from MisGAN refer to this type of missingness as *dropout* [16]. When scale parameter ρ decreases, the missingness will increasingly depend on the pixel values and become MNAR. This can be seen in Figure 2 by comparing the blue lines for $\rho = 0.2$ with the orange lines for $\rho = 0.6$: the smaller ρ, the higher the missingness probability for high-grayscale pixel values. Then, more information from the MNIST image will be lost (in contrast to black pixels: if those are masked, no substantial information is lost).

Constant c controls the percentage of missing values in the entire dataset. Since the pixel values are re-scaled to $[0, 1]$, the mean value μ_i represents a probability as well. Therefore, the expected percentage of missing values over all images is the sum of the overall pixel mean (0.131), and c. In Figure 2, this effect can be seen by the vertical shift of the lines (i.e., compare the y-axis values).

Note that we subtract the pixel values in M_i from μ_i and thus let higher grayscale values (i.e., white pixels) obtain a higher probability of being missing than lower grayscale values (i.e., black pixels). This corresponds to the RIGHT missingness type as defined in the multivariate amputation approach of [26].

4 Experimental Setup

We choose to run our experiments with $\rho \in \{0.2, 0.6, 1\}$. We generate average missingness percentages of 6%, 15%, and 30%, which means that we let $c \in \{-0.071, 0.019, 0.169\}$ (see Table 1). We chose to use these percentages after visually inspecting the missingness; since a large part of each MNIST image is



(a) Mean estimates. (b) Distributions.

Fig. 3. MSE score metadata over all images for all 9 test cases.

black background, containing no actual information on the hand-written digit (we will refer to these black pixels as "uninformative" while calling the non-black pixels "informative"), MNAR missingness with too high missingness percentages will leave no data for MisGAN to train on. For the same reason, the lowest scale parameter value that we use is 0.2; applying a pure MNAR mechanism with $\rho = 0$ will also remove too many informative pixel values.

We evaluate the performance of MisGAN by computing the Mean Squared Error (MSE) of the imputed informative pixels:

$$\text{MSE}(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_i) = \frac{1}{k_i} \sum_{j=0}^{k_i - 1} \left((\boldsymbol{y}_i)_j - (\hat{\boldsymbol{y}}_i)_j \right)^2 \tag{3}$$

Here, $\boldsymbol{y}_i$ is the vector of all k_i informative pixels of image i and $\hat{\boldsymbol{y}}_i$ is the vector of imputed values for those same pixels.[2] Naturally, there is no error when informative pixels were not masked. The lower the MSE score, the better the performance of the imputation. For reproducibility purposes, all code can be accessed at our Github page.[3]

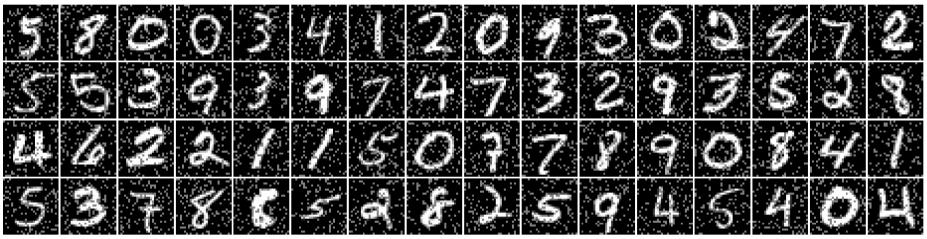
5 Experimental Results

Estimates of the mean, standard deviation and median of the MSE scores over all images are shown in Table 1. The estimates of the mean are displayed in Figure 3a; Figure 3b shows the distribution of the MSE scores.

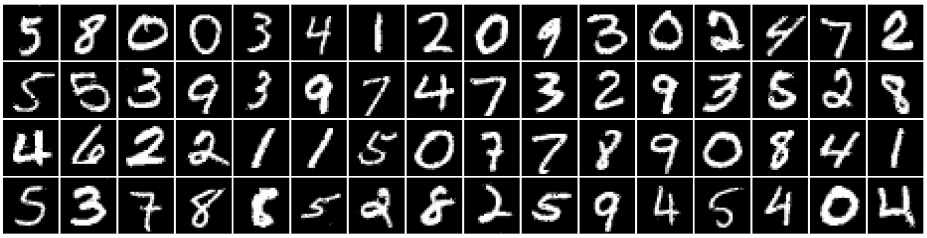
We find that MSE scores increase when data is amputed with an MNAR missingness mechanism, compared to an MCAR mechanism. The stronger the

[2] To be precise, $\boldsymbol{y}_i = \{X_i[d] \mid X_i[d] > 0\}$ and $\hat{\boldsymbol{y}}_i = \{\hat{X}_i[d] \mid X_i[d] > 0\}$ for all $d \in \{1, 2, ..., 784\}$.

[3] https://github.com/RianneSchouten/misgan_mnar/



(a) Masked images.



(b) Imputed images.

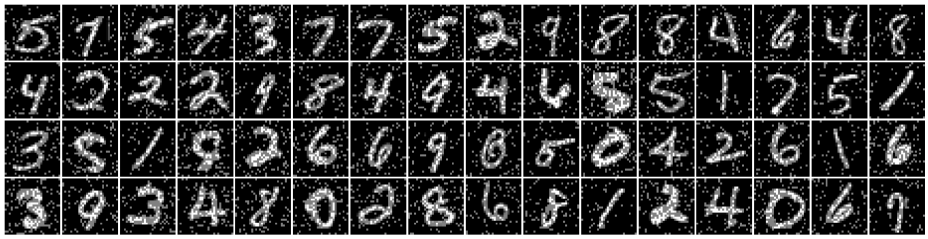
Fig. 4. Masked (top) and imputed (bottom) MNIST images under 15% missingness for MCAR. Masking is shown in gray.

MNAR aspect of the MCAR-MNAR mixture, the higher the MSE scores (cf. Figure 3a, orange and green lines). Furthermore, we find that MSE increases when missingness percentages increase (e.g., MSE increases from 0.545 via 0.549 to 0.558 when missingness percentage increases from 6% via 15% to 30% and $\rho = 0.6$; see Table 1). With a pure MCAR mechanism, all mean estimates of the MSE scores are 0.515: the missingness percentage does not influence the accuracy of the imputation procedure (cf. Figure 3a, horizontal blue line).

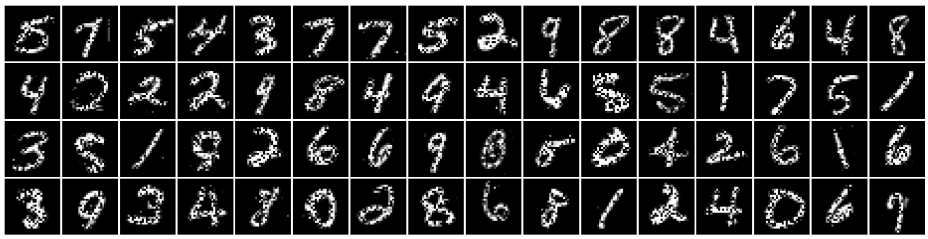
Figures 4b, 5b and 6b demonstrate the imputation procedure for 15% missingness in a few example images. Although it is clear that a stronger MNAR mechanism results in more diffused imputed digits, the extent to which the original digits are visible and recognizable by the human eye can be considered surprising (e.g., see Figure 6b), especially since MisGAN will have had little information to train on (note that the digits in Figure 6a are more clearly visible since the amputed pixel values are indicated as masked by a gray color).

6 Discussion

This paper investigates the effectiveness of MisGAN under the MNAR assumption. We propose a method for generating MNAR missingness based on the grayscale values; we apply some function to them, and use Bernoulli probabilities to determine missingness in the images, subsequently testing MisGAN on this type of missingness (cf. Equation (1)). This is just one approach to model MNAR missingness. One could also decide to let the missingness depend on the



(a) Masked images.

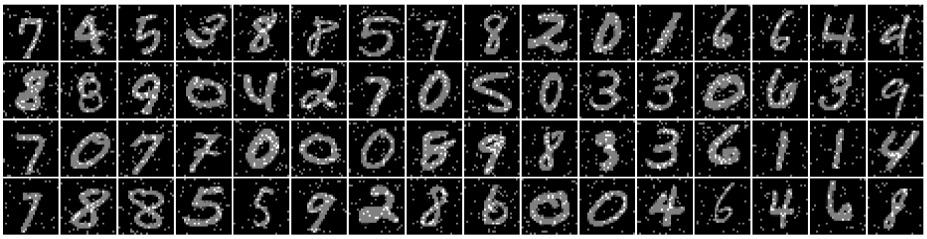


(b) Imputed images.

Fig. 5. Masked (top) and imputed (bottom) MNIST images under 15% missingness for MNAR with $\rho = 0.6$. Masking is shown in gray.

target variable or certain shapes in the image; there are varying ways to simulate MNAR. Even for the MNAR missingness based on grayscale value, there are countless ways to simulate MNAR missingness. Consider Figure 7. Suppose that missingness can only depend on the grayscale value of a pixel. In the left graph one can see MCAR missingness, since for all grayscale values, there is an equal probability of missingness. The graph in the middle and on the right are both showing MNAR missingness as the probability depends on the grayscale value. The middle graph, however, looks much more like the MCAR case and will therefore be easier to impute than the one on the right, based on the results of the MSE scores and visual inspection in Section 5. It makes sense that the performance of MisGAN in the MNAR case not only depends on the extent of the missingness, but also on the degree to which the MNAR missingness resembles MCAR missingness [28].

It is hard to say whether or not MisGAN could still be employed under these circumstances. In this paper, we evaluated by calculating the Mean Squared Error over the informative pixels. This is an imputation-accuracy based evaluation metric which has been criticized ("imputation is not prediction" [3, Section 2.6], the distribution of imputed values is multimodal [16]). At the same time, the goal of our paper is to demonstrate how to effectively generate MNAR missingness in image data. Since our MNAR amputation approach strongly connects to individual pixel values, it seems appropriate to directly evaluate the imputed values of those individual pixels. At least, there is precedent for using MSE to evaluate imputation methods on MNIST [7,8]. In addition, the approach of us-



(a) Masked images.

(b) Imputed images.

Fig. 6. Masked (top) and imputed (bottom) MNIST images under 15% missingness for MNAR with $\rho = 0.2$. Masking is shown in gray.

ing an inference-based evaluation metric as suggested by [3] is not applicable to image data. The approach of using a classification-accuracy-based evaluation metric seems more appropriate, but requires the context of a train-test split [31] which is beyond scope for this paper.

We investigated one more approach: the Fréchet Inception Distance (FID), suggested by [13] as a way of evaluating the performance of a GAN in such a way that it reflects the visual quality of the generated images. The FID is a distance measure over the mean and covariance structure of the activation scores of a batch of images loaded into a pre-trained image classification model (in the case of [13], this is Inception [32] trained on ImageNet [24]). The idea is that the deepest layers of the network are close enough to the output layer to reflect classification accuracy, but far enough from the raw input pixel values to prevent issues such as high modality.

Although we recognize the potential value of this approach, we currently believe it is not sufficiently standardized for generic use. On the one hand, at the moment, there is no consensus on which image classification network ought to serve as a ground-truth model. The typically used Inception [32] network is trained on ImageNet [24], which may be a decent ground truth for generic image classification problems, but a more tailored ground truth is likely more appropriate for specific data spaces such the black and white MNIST image space. At the same time, the alternative of a LeNet model as used by [16, p.6] seems at best an arbitrary choice. On the other hand, FID is proposed as an evaluation measure of *generated images* [13]; its appropriateness as a measure

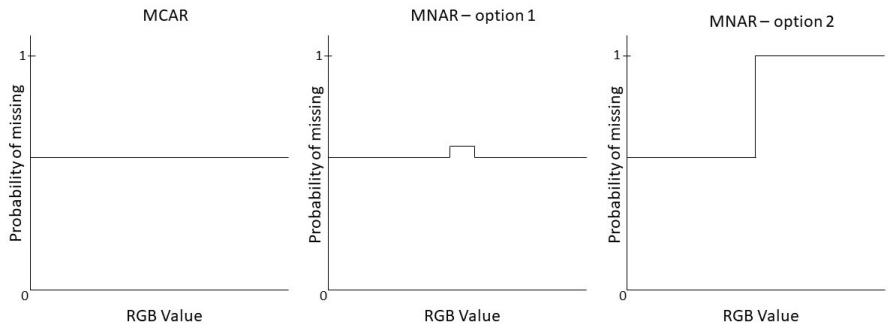


Fig. 7. Illustration of ways to simulate MNAR missingness

of evaluating *imputation quality* is not yet clear. Possible issues with Inception-based scores that result to unreasonably low FID scores, such as images that are too similar or not realistic enough [2,29], may be particularly vexing in the context of imputation and disturb the simulation process.

As possible alternatives for evaluating the performance of GANs, two measures exist akin to recall and precision [29] proposed, and a three-dimensional metric exists that captures the fidelity, diversity, and generalization quality of the generated images [1]. It is worth investigating to what extent these metrics would be useful for evaluating the *imputation* performance of a GAN-based imputer such as MisGAN.

7 Conclusion

We determine the degree to which MisGAN [16] is capable of handling image data with Missing Not At Random (MNAR) missingness. We propose a method for generating MNAR missingness in the MNIST dataset and let higher grayscale pixel values have a higher probability of being missing. In addition, we vary the extent to which these missingness probabilities depend on the pixel values and investigate the effect of simultaneously occurring mechanisms. Indeed, we find that MisGAN is not working quite as well on MNAR data as it is on MCAR data; the stronger the MNAR effect, the higher the drop in performance.

7.1 Future Work

This research extended the research of [16], which presented a GAN-based framework for learning from complex, high-dimensional incomplete data where data is assumed to be MCAR. Our contribution to this research is that we assume data to be MNAR. However, we do not investigate the underlying mathematical theorem for the MNAR approach. In order to have a deeper understanding on MisGAN in the case of MNAR, a theoretical analysis is required. Furthermore, it would be interesting to perform similar investigations for MisGAN-adjacent methods, such as GAIN [34], MCFlow [21], EMFlow [18], and MIWAE [19].

Finally, one last elephant remains in the room: this whole paper rather ignores the existence of the MAR missingness mechanism. For traditional, flat-table data, having three distinct missingness mechanisms makes sense: MCAR makes the missingness of data depend on neither the observed nor the missing data, MAR makes the missingness of data depend on observed but not the missing data, and MNAR makes missingness of data depend on the missing data itself. So, in flat-table data, MCAR missingness is typically uniformly random, MAR missingness means that information on missing values can be derived from values in other columns of the dataset, and MNAR missingness hides the information on the missing value in the missing value itself. These three mechanisms do not necessarily translate to pixel-level missingness mechanisms in image data. Generating MCAR missingness in image data is trivial: just mask randomly selected pixels. We introduced one specific mechanism to generate MNAR missingness in image data in Section 3.1. However, it is not immediately apparent how to generate MAR missingness in image data, without accidentally generating MNAR missingness. The MAR concept is that the missingness of a pixel would depend on the values in other pixels, but not on the value of the pixel itself. However, suppose that we find ourselves in a white pixel that lies on the diagonal stripe of a seven. We can make its missingness depend on the value of another pixel, which might also lie on that same diagonal stripe. In that case, the value of the other pixel is strongly correlated with the value of the pixel whose missingness we are trying to determine, and in trying to generate MAR missingness we have in fact generated MNAR missingness. A solution to this problem can probably be found by not trying to generate MAR missingness in image data *on the pixel level*, but instead trying to generate such missingness on a higher, conceptual level. This implies that MAR missingness in image data might perhaps be generated *in a convolutional layer of a neural network*. Properly defining such a mechanism and making it work in practice would encompass an exciting future research direction.

References

1. Alaa, A., Van Breugel, B., Saveliev, E.S., van der Schaar, M.: How faithful is your synthetic data? Sample-level metrics for evaluating and auditing generative models. In: Proc. ICML. pp. 290–306 (2022)
2. Barratt, S., Sharma, R.: A note on the inception score (2018), https://arxiv.org/abs/1801.01973
3. van Buuren, S.: Flexible imputation of missing data. CRC press (2018)
4. van Buuren, S., Brand, J.P., Groothuis-Oudshoorn, K., Rubin, D.B.: Fully conditional specification in multivariate imputation. Journal of Statistical Computation and Simulation **76**(12), 1049–1064 (2006)
5. van Buuren, S., Groothuis-Oudshoorn, K.: mice: Multivariate imputation by chained equations in R. Journal of Statistical Software **45**, 1–67 (2011)
6. Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A.: Generative Adversarial Networks: An overview. IEEE Signal Processing Magazine **35**(1), 53–65 (2018)

7. Dalca, A.V., Guttag, J., Sabuncu, M.R.: Unsupervised data imputation via variational inference of deep subspaces. arXiv preprint arXiv:1903.03503 (2019)
8. Danel, T., Śmieja, M., Struski, L., Spurek, P., Maziarka, L.: Processing of incomplete images by (Graph) Convolutional Neural Networks. In: Proc. NeurIPS. pp. 512–523 (2020)
9. Deng, L.: The MNIST database of handwritten digit images for machine learning research [best of the web]. IEEE Signal Processing Magazine **29**(6), 141–142 (2012)
10. Gondara, L., Wang, K.: MIDA: Multiple imputation using Denoising Autoencoders. In: Proc. PAKDD. pp. 260–272 (2018)
11. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Proc. NeurIPS. pp. 2672—-2680 (2014)
12. Harel, O., Schafer, J.L.: Partial and latent ignorability in missing-data problems. Biometrika **96**(1), 37–50 (2009)
13. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs Trained by a two time-scale update rule converge to a local Nash equilibrium. In: Proc. NeurIPS. pp. 6627–6638 (2017)
14. Ipsen, N.B., Mattei, P.A., Frellsen, J.: not-MIWAE: Deep generative modelling with missing not at random data. In: Proc. International Conference on Learning Representations (2021)
15. Kim, J., Tae, D., Seok, J.: A survey of missing data imputation using Generative Adversarial Networks. In: Proc. ICAIIC. pp. 454–456 (2020)
16. Li, S.C., Jiang, B., Marlin, B.M.: MisGAN: Learning from incomplete data with generative adversarial networks. In: Proc. ICLR (2019)
17. Little, R.J., Rubin, D.B.: Statistical analysis with missing data. John Wiley & Sons (2019)
18. Ma, Q., Ghosh, S.K.: EMFlow: Data Imputation in Latent Space via EM and Deep Flow Models (2022), https://arxiv.org/abs/2106.04804
19. Mattei, P., Frellsen, J.: MIWAE: Deep generative modelling and imputation of incomplete data sets. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the International Conference on Machine Learning. vol. 97, pp. 4413–4423 (2019)
20. Oberman, H.I., Vink, G.: Toward a standardized evaluation of imputation methodology. Biometrical Journal p. 2200107 (2023)
21. Richardson, T.W., Wu, W., Lin, L., Xu, B., Bernal, E.A.: McFlow: Monte Carlo Flow Models for Data Imputation. In: Proceedings of the Conference on Computer Vision and Pattern Recognition. pp. 14193–14202 (2020)
22. Rubin, D.B.: Inference and missing data. Biometrika **63**(3), 581–592 (1976)
23. Rubin, D.B.: Multiple imputation for nonresponse in surveys. John Wiley & Sons (2004)
24. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. Proc. NeurIPS **29** (2016)
25. Schafer, J.L.: Analysis of incomplete multivariate data. CRC press (1997)
26. Schouten, R.M., Lugtig, P., Vink, G.: Generating missing values for simulation purposes: A multivariate amputation procedure. Journal of Statistical Computation and Simulation **88**(15), 2909–2930 (2018)
27. Schouten, R.M., Taşcău, V., Ziegler, G.G., Casano, D., Ardizzone, M., Erotokritou, M.A.: Dropping incomplete records is (not so) straightforward. In: Proc. IDA. pp. 379–391 (2023)
28. Schouten, R.M., Vink, G.: The dance of the mechanisms: how observed information influences the validity of missingness assumptions. Sociological Methods & Research **50**(3), 1243–1258 (2021)

29. Shmelkov, K., Schmid, C., Alahari, K.: How good is my GAN? In: Proc. ECCV. pp. 213–229 (2018)
30. Smieja, M., Struski, L., Tabor, J., Zieliński, B., Spurek, P.: Processing of missing data by Neural Networks. Proc. NeurIPS pp. 2719–2729 (2018)
31. Sperrin, M., Martin, G.P., Sisk, R., Peek, N.: Missing data should be handled differently for prediction than for description or causal explanation. Journal of clinical epidemiology **125**, 183–187 (2020)
32. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proc. CVPR. pp. 1–9 (2015)
33. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proc. ICML. pp. 1096–1103 (2008)
34. Yoon, J., Jordon, J., van der Schaar, M.: GAIN: Missing data imputation using Generative Adversarial Nets. In: Dy, J.G., Krause, A. (eds.) Proceedings of the International Conference on Machine Learning. vol. 80, pp. 5675–5684 (2018)
35. Zhang, Z.: Missing data imputation: Focusing on single imputation. Annals of Translational Medicine **4**(1), 9 (2016)
36. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proc. ICCV. pp. 2223–2232 (2017)