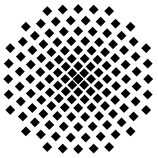


University of Stuttgart
Institute for Signal Processing and System Theory
Professor Dr.-Ing. B. Yang

Forschungsarbeit S1455

Avoiding Shortcut-Learning Through Mutual Information Minimization for Datasets with Multiple Confounding Variables

Author: Gautham Mohan

Date of work begin: 17/10/2023
Date of submission: 16/04/2024

Supervisor: Louisa Fay

Keywords: Causality, deep learning

Abstract Deep learning methodologies rely on the ability to leverage relationships between features within datasets to extract meaningful representations. However, confounding variables can introduce spurious or meaningless correlations into the dataset, leading to biases in deep learning models. The mutual information minimization model successfully learns causal relationships from datasets with a single confounding variable. This thesis proposes an extension of the concept for datasets with multiple confounding variables. A detailed study of the mutual information neural estimation and an exploration of the causal structure of datasets with multiple confounding variables have been done for this. The proposed model has been tested on two datasets - a benchmark Morpho-MNIST dataset and a medical CheXpert dataset. The experiments performed validate the success of the model in learning true causal relationships from datasets with multiple confounding variables.

Contents

1. Introduction

The development of convolutional neural networks (CNNs) has catalysed a shift in computer vision, moving away from expert systems towards deep learning [1]. Although gradual, the medical imaging community has also undergone this change, employing deep learning for various applications such as segmentation, classification, and detection across a range of data types including MRI, microscopy, CT scans, and X-rays. Notably, leading participants in various challenges in computer-aided diagnostics (CAD), such as those on Kaggle and in DREAM challenges, consistently rely on deep learning techniques [2].

Learning algorithm often assume that features in training and test dataset are independent and identically distributed [3].But this assumption does not hold in real-world, particularly in medical datasets. Distribution of features such as the sex and age of patients, as well as the hospital environment maybe skewed [4] [5] [6]. Such deviations from the ideal distribution can lead to the emergence of spurious non-predictive correlations[7] [8].

Deep learning, which relies on learning associations between variables in the data, encounters challenges in distinguishing between causal relationships and mere spurious correlations [5] [8]. While causal relationships are invariant to distribution shifts, spurious correlations are often tied to the distribution and can mislead the models to wrong results when the distribution changes, such as shifts in demography or hospital environments from which the data is collected [9] [10] [5]. These distribution shifts can have significant implications for the reliability and generalizability of predictive models.

The primary feature of a model that learns the true causal relationships in data is counterfactual invariance, i.e. the model should be able to predict the correct output label for a given input even if the spuriously correlated variable in the input changes [11]. The process of bringing such invariance to the deep learning model requires an understanding of the causal structure of the original training data. To bring forth counterfactual invariance it is important to determine the degree of spurious correlation between two variables. Mutual information (MI) can be used as a measure for this purpose [12]. A mutual information neural estimation (MINE) technique which relies on the Donsker-Varadhan representation of Kullback-Leibler (KL) divergence has been proposed by Belghazi et al. [13].

Fay et. al. [12] proposed a novel mutual information minimization model (MIMM) for datasets with a single spurious correlation variable. The framework requires a feature encoder combined with a MINE model to estimate MI and use it as a regularisation term in the loss function for classification tasks. This thesis is an extension of the concept for datasets with multiple spurious correlation variables.

2. Literature Review

The literature review was conducted to explore the basic foundations of causal inference and counterfactual invariance. MINE was studied as a method to quantify spurious correlation between variables, while also emphasizing essential considerations crucial for its application. Furthermore, a comprehensive examination of MIMM, as introduced by Fay et al., is provided, along with its intricate training process. The insights gained from this review are presented below in subsequent sections.

2.1. Causal Background

The basics to understanding causality is to understand that the existence of a correlation between two variables does not mean that one variable causes the other[14][15]. The idea behind this statement is that correlation is not a necessary or sufficient condition for causal relationships. Such correlations which has a tendency to imply a causal relationship between correlated event but actually does not mean anything of the same nature are called spurious correlations.

Reichenbach's common cause principle suggests a mechanism by which spurious correlations can arise [14]. The statement suggests that given two variables Z and Y are statistically dependent i.e. $Z \not\perp\!\!\!\perp Y$,then there exists a third variable U causally influencing both of them. In such cases, the Z and Y are independent of each other given we know the confounding variable U, i.e. $Z \perp\!\!\!\perp Y \mid U$.

This is illustrated in 2.1 through an example, Z="type of scanner" and Y="type of neurological disease" and U="Doctor" [12]. Now assume that "Doctor A" is a specialist in Alzheimer's disease (AD) and works mostly with "Scanner A" and "Doctor B" attends mostly to general neurological disorders while using "Scanner B" for his procedures. This results in a data-generating process which shows a spurious correlation between AD and "Scanner A". However, this correlation is not causal; it is solely influenced by the doctor, i.e. U treating the patients, i.e. we can independently determine the "type of scanner" and "type of neurological disease" from the knowledge of which "Doctor" is treating the patient.

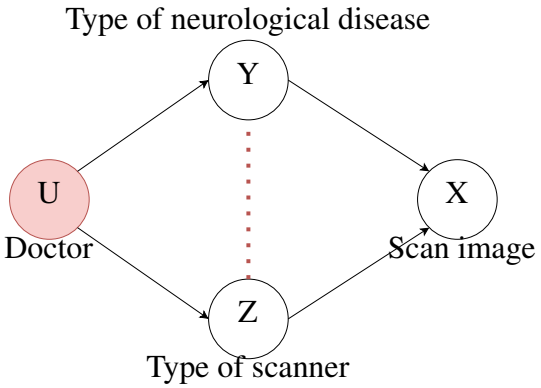


Figure 2.1.: Spurious correlation through confounding. The confounding variable is marked in red. The arrow marks the direction of causation and the spurious correlation is shown through the red dotted line.

The other common mechanism for spurious correlation which is often discussed in the medical domain in selection bias. This occurs when two independent variables Y and Z jointly cause an observation S [16]. Here although the Y and Z are marginally independent, when conditioned on S they become dependent, i.e. $Z \not\perp\!\!\!\perp Y \mid S$.

The selection bias is illustrated in 2.2 through an example [16]. Suppose a particular graduate school admits students either on the criteria that they have good grades or on the criteria they are exceptionally musically talented, then the grades and musical talent across the population of students admitted to the school will appear to be negatively correlated. Here, although the general population shows no particular correlation between musical talent and grades, if we select a single student admitted to the school and if the student has low grades then the student will obviously need to have higher musical talent to be admitted to the school.

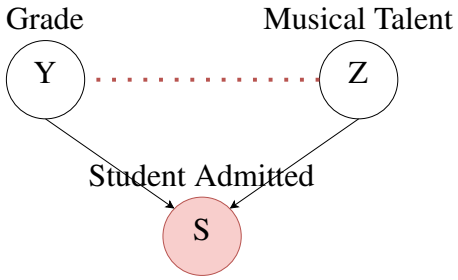


Figure 2.2.: Spurious correlation through selection. The selection variable is marked in red. The arrow marks the direction of causation and the spurious correlation is shown through the red dotted line.

Machine learning is essentially learning associations between input samples X and their labels Y, this is done in the training process by adjusting the parameters of a model f to map X to Y [5] [11].This mapping from X to Y is said to be in the causal direction if X is the cause of Y, an example would be humidity, temperature and wind speed (X) being used to predict the chance for rain (Y). The same mapping is said to be in the anti-causal direction if Y is the cause for X, an example can be drawn from the MNIST dataset where the idea of the number "4" that is the label "4" (Y) is the cause for the shape of the number in its image input (X) [12].

The concepts described above can be well summarized using the example of mophoMNIST image classification as provided by Fay et. al. [12] illustrated in 2.3. In this example a writer "A" uses thick nib pens and writes mostly high-valued numbers like "9" while another writer "B" uses thin nib pens to write mostly low-valued numbers like "2". This end up in a process that creates spurious correlation between the numbers ranging from 0-4 ("low") and thin profile, and the numbers in the range 5-9 ("high") and thick profile, the profile being the spurious correlation variable Z and the writer U being the confounding variable. The label Y is the primary task to be predicted by the model f from the input image X, this is indicated by green thick arrow. The value taken by label, i.e. "high" or "low" and the profile chosen, i.e. "thick" or "thin" determines the kind of image that we get, hence the direction of causation along the black arrows as indicated.

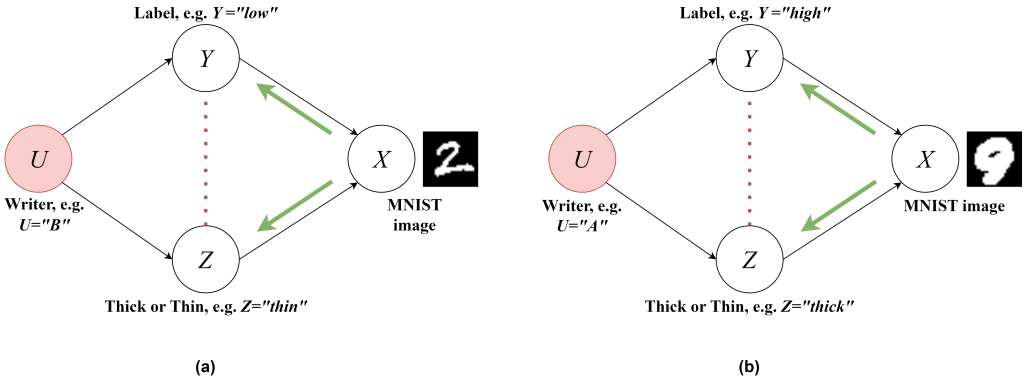


Figure 2.3.: Predictions in anti-causal directions in a confounded Morpho-MNIST dataset.

2.2. Counterfactual Invariance

In Figure 2.4 we see that X is causally influenced by Y and Z [12]. This means that in the image X of the number of "4", there is a representation of the label Y = ”4” as the shape of the number and the thickness Z of the number Z. Now, if we fix the value of Y = ”4” and then vary the value of Z between "thick" and "thin" we get a counterfactual pair of X. We can formalise such counterfactuals as $X(y, z)$ and $X(y, z')$, in the example z and z' correspond to "thick" and "thin" and y is fixed which is "4" here [11].

A neural network is a function f which takes X as input to make a prediction about the label Y. The variable Y here refers to the ground truth and the actual prediction is represented by $f(X)$. The predictions of such a model is said to be counterfactually invariant if it satisfies the following condition-

$$f(X(y, z)) = f(X(y, z')) \tag{2.1}$$

This means that the prediction $f(X)$ remains the same even if Z changes ("independent of Z"), as long as the value of $Y = y$ remains fixed ("given Y"). Hence the above condition can be re-written as below-

$$f(X) \perp\!\!\!\perp Z \mid Y \tag{2.2}$$

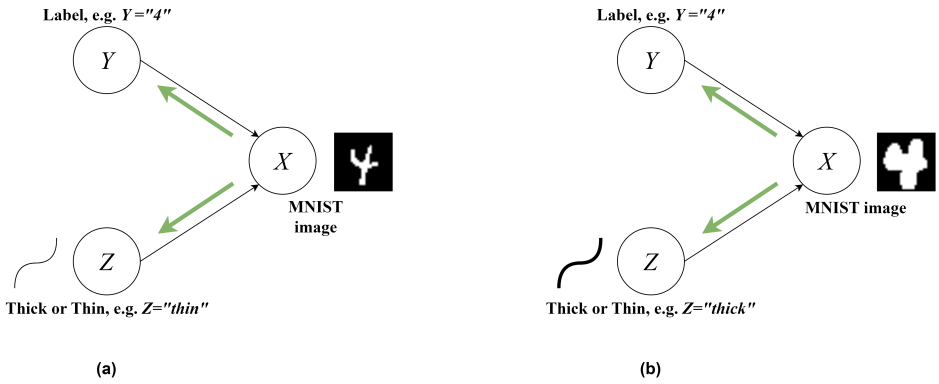


Figure 2.4.: Counterfactuals generated by varying the thickness of lines pen used to write the number 4.

This implies that f learns the representation of Y embedded in X and ignores the representatio of Z in it. For the example in Figure 2.4 this would mean that the neural network learns the shape of the number by ignoring the information about thickness embedded in the image. However, the neural network cannot learn representations of the label Y from X in isolation, this is because of the presence of confounding variables that create spurious correlation between Y and Z, confusing the network into using Z to predict Y [17] [11].

In the example illustrated in Figure 2.3, here all the samples that are "thick" are also "high" and all the samples that are "thin" are also "low". This information is embedded in the input image X which could result in the model relying on the information that the image X has a thick profile to predict that the number belongs to "high" class [12]. Let there be an input $x = X(Y = "small", Z = "thick")$, this is an out-of-domain image since there is no image similar to the given x present in the original training set. Since f learned the spurious correlation between thickness and the label, it is probable that the model will now predict x to belong to class "high" even though it actually does not belong to it [17]. To avert this and enhance the model's out-of-domain generalization, we must sever the spuriously correlated connection between X and Z [12].

2.3. Mutual Information Neural Estimation

The previous sections explain how spurious correlation between some features in the data and the label can reduce the ability of machine learning models to generalise well to the data. To prevent the occurrence of spurious correlation, it is necessary to able quantify it.

While linear correlation measures can be used to gauge spurious correlation, it has the limitation that it only captures linear relationships. It merely indicates whether one variable increases or decreases as the other increases but often the relationships between features in a dataset is much more complex and a simple linear relationship might not be able to capture this. MI serves as a robust measure that can be utilized in this scenario. [18] [12].

MI is a measure based on shannon entropy as summarised by the expression 2.3 [13]. This can be interpreted as the information gained about Y from the knowledge of Z, which also can be interpreted as the dependence of Y on Z.

$$I(Y;Z) = H(Y) - H(Y|Z) \tag{2.3}$$

It can also be represented using KL divergence as 2.4 [13]. The higher the divergence between the joint distribution and product of the marginal distributions of X and Z the higher the MI between them, this also corresponds to the independence criteria where Y and Z are independent if $\mathbb{P}_{YZ} = \mathbb{P}_Y \otimes \mathbb{P}_Z$.

$$I(Y;Z) = D_{\mathrm{KL}}(\mathbb{P}_{YZ} \| \mathbb{P}_Y \otimes \mathbb{P}_Z) \tag{2.4}$$

The expression 2.4 cannot be used directly and a more tractable form of the equation is available through the Donsker-Varadhan representation of KL-divergence as given in expression 2.5 [13]. Here we are taking supremum over all functions T that maps from the sample space Ω to a real number $\mathbb{R}$.

$$D_{\mathrm{KL}}(\mathbb{P} \| \mathbb{Q}) = \sup_{T:\Omega \to \mathbb{R}} \left(\mathbb{E}_{\mathbb{P}}[T] - \log(\mathbb{E}_{\mathbb{Q}}[e^T]) \right) \tag{2.5}$$

MINE uses a neural network to model T and it now belongs to a class of functions $\mathcal{F}$, this network is called the statistics network. Since the range of functions that can be represented by the statistics network is limited, it now represents a tight lower bound for KL divergence. The expression 2.5 can now be adapted for KL divergence given in expression 2.4 with $T : Y \times Z \to \mathbb{R}$ parameterized by θ of the network [13].

$$D_{\mathrm{KL}}(\mathbb{P}_{YZ} \| \mathbb{P}_Y \otimes \mathbb{P}_Z) = \sup_{\theta \in \Theta} \left(\mathbb{E}_{\mathbb{P}_{YZ}}[T_\theta(Y,Z)] - \log(\mathbb{E}_{\mathbb{P}_Y \otimes \mathbb{P}_Z}[e^{T_\theta(Y',Z')}]) \right) \tag{2.6}$$

The samples from the independent distribution $(Y', Z') \sim \mathbb{P}_Y \otimes \mathbb{P}_Z$ can be produced by shuffling the samples from the joint distribution $(Y, Z) \sim \mathbb{P}_{YZ}$. In order to find the supremum the term in expression 2.6 is maximised using stochastic gradient descent (SGD). The gradient of 2.6 with respect to θ yields us expression 2.7, here B represents a mini-batch [13]. MINE is strongly consistent given that we have a sufficient batch size and a statistics network T_θ with sufficient representational power.

$$\widehat{G}_B = \mathbb{E}_B\left[\nabla_\theta T_\theta\right] - \frac{\mathbb{E}_B\left[\nabla_\theta T_\theta e^{T_\theta}\right]}{\mathbb{E}_B\left[e^{T_\theta}\right]}, \tag{2.7}$$

$$\frac{\mathbb{E}_B\left[\nabla_\theta T_\theta e^{T_\theta}\right]}{\mathbb{E}_B\left[e^{T_\theta}\right]} \neq \frac{\nabla_\theta T_\theta e^{T_\theta}}{e^{T_\theta}} \neq \nabla_\theta T_\theta. \tag{2.8}$$

The second term in expression 2.7 leads to a biased estimate of the gradient for smaller mini-batch sizes. This issue can be understood by considering the extreme case in which the batch size is just a single sample as illustrated in expression 2.8 [19]. Here the expression is wrongly evaluated because of simplification of numerator and denominator through e^{T_θ}. This problem can be overcome by replacing the denominator in the second term using a moving average to estimate $\mathbb{E}_B\left[e^{T_\theta}\right]$.

In experiments where MINE output was used as a regularisation term, it is suggested to adaptively scale the the MI term [13]. The gradients from the MI term, i.e. g_{MI}, might be large compared to the gradients from the loss function of the main objective, e.g. classification task i.e. g_{class}. This might cause the learning algorithm to concentrate more on the MI term and ignore the main objective. It is proposed to scale g_{MI} by limiting the Frobenius norm of the gradient from the MI term to that of the gradient from the main objective as given by expression 2.9 [13].

$$g_a = \min\left(\|g_{class}\|, \|g_{MI}\|\right) \frac{g_{MI}}{\|g_{MI}\|} \tag{2.9}$$

2.4. Mutual Information Minimization Model

MIMM model was proposed to reduce the influence of spuriously correlation on predictions of deep learning models [12]. MIMM uses a feature encoder to encode the primary task and the spurious correlation as two feature vectors and then the MINE model is used to estimate the MI between the two feature vectors which is then used as a regularisation term in the loss function.

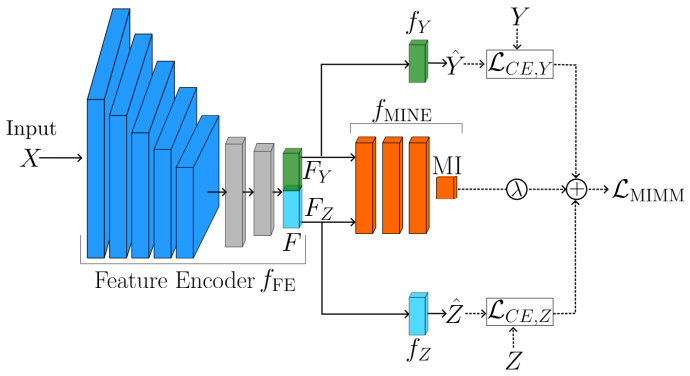


Figure 2.5.: The architecture of MIMM model [12]

The feature encoder is fed with the input image X from which the feature vector F is extracted. The feature vector F is then split into two parts F_Y and F_Z one for the prediction of the primary task and the other for the prediction of the spurious correlation task respectively. Here, f_{MINE} represents the statistics network T_θ along with the computations necessary for MI, while f_Y and f_Z represents the classification heads for primary and spurious correlation tasks. The MI computation is now given by expression 2.10 in which we replace the primary task and spurious correlation variable by their respective feature vectors. The samples from independent distribution (F'_Y, F'_Z) can be obtained by shuffling the joint distribution (F_Y, F_Z) obtained from the feature encoder along the batch axis [13].

$$I(F_Y, F_Z) = \sup_{\theta \in \Theta} \left(\mathbb{E}_{\mathbb{P}_{F_Y F_Z}}[T_\theta(F_Y, F_Z)] - \log(\mathbb{E}_{\mathbb{P}_{F_Y} \otimes \mathbb{P}_{F_Z}}[e^{T_\theta(F'_Y, F'_Z)}]) \right) \tag{2.10}$$

The idea here is to make the feature encoder learn representations of PT and SC, i.e. F_Y and F_Z, such that they share minimum information between them. This would reduce the

influence of spurious correlation variable Z on the primary task variable Y.For this, the cross-entropy classification loss $\mathcal{L}_{CE,Y}$ and $\mathcal{L}_{CE,Z}$ are combined through a scaling factor λ with the MI regularisation term in expression 2.11 [12]. This is then minimised to update the parameters of the feature encoder. This ensures that the feature encoder learns the representation of primary task and spurious correlation task while keeping a check on the MI shared between them.

$$\mathcal{L}_{MIMM} = \mathcal{L}_{CE,Y}(X, Y) + \mathcal{L}_{CE,Z}(X, Z) + \lambda \cdot MI(X) \tag{2.11}$$

$$\mathcal{L}_{CE,Y} = -Y^T \log f_Y(F_Y) \tag{2.12}$$

$$\mathcal{L}_{CE,Z} = -Z^T \log f_Z(F_Z) \tag{2.13}$$

The training of the MINE model is essentially to find the supremum and since the input to the model comes from the feature encoder it is necessary to maximise the output of MINE model after each update of the feature encoder. This is done by minimising the loss function $\mathcal{L}_{MIMM}$ to update the feature encoder using 1 batch followed by N_B - 1 batches of maximisation of the output of MINE model where N_B is a hyperparameter. This ensures that the feature encoder learns representations with a low value of MI, however, this value will never be zero since the MI term in $\mathcal{L}_{MIMM}$ is a soft constraint.

The paper tested the model on two benchmark datasets Morpho-MNIST and FashnionMNIST and three medical databases; German National Cohort, UK Biobank, and ADNI. Spurious correlations were created in the datasets through labels distribution, for example the training and test datasets used for the UKB/NAKO experiment are as illustrated in Figure 2.6. It can be seen here that in the training data, female and lower age group has a high correlation as does male and higher age group. This distribution is flipped in case of the test dataset and in case of balanced test set there is equal representation of sex in both the age groups.

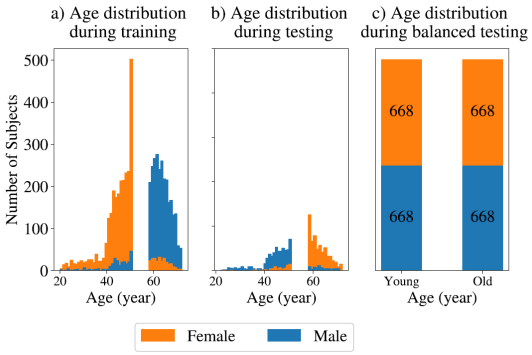


Figure 2.6.: UKB/NAKO age group and sex label distribution [12]

The evaluation was performed by measuring the accuracy on the test and balanced test for five different models including a baseline model and MIMM. To test the independence of the feature vectors, F_Y was used to predict the label Z and F_Z was used to predict Y. The

accuracy in this case was expected to be less than the random chance, i.e. 50% in case of binary classification tasks, for this the balanced test dataset is used. Further analysis was performed by plotting the t-SNE plot of F_Y and then marking it with the labels of Z as illustrated in fig 2.7.

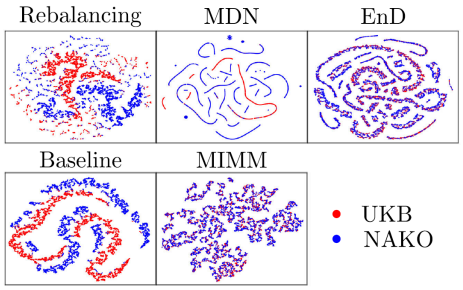


Figure 2.7.: UKB/NAKO the feature vector F_Y coloured by the labels of spurious correlation task Z [12]

MIMM was observed to perform well in all the test cases including the t-SNE analysis. It can be seen from the t-SNE that the labels of the spuriously correlated variable have been equally distributed across the feature vector of the primary task, whereas a separation can be made for all the other models.

3. Materials and Methods

3.1. Datasets

The experiments where conducted using two datasets. The hypotheses was tested on a non-medical benchmark Morpho-MNIST [20] dataset and a medical CheXpert Small dataset [21]. The experiments were conducted such that a neural network is trained to perform image classification as the primary task Y. The confounding of the datasets were done with two spuriously correlated variables Z_0 and Z_1 using the ratios in table 3.1. The details of the dataset are discussed in the following sections.

Task	**Train Data**		**Validation Data**		**Test Data**		**Balanced Test Data**	
Z / **Y**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
0	90%	10%	90%	10%	10%	90%	50%	50%
1	10%	90%	10%	90%	90%	10%	50%	50%

Table 3.1.: Confounding Ratios, $Z = Z_0$ or Z_1

3.1.1. Morpho-MNIST Dataset

The Morpho-MNIST dataset is derived from the benchmark MNIST dataset. The dataset was created by introducing a set of morphological perturbations which includes global changes like thinning and thickening and also local changes like swelling, and fractures of the digits in the MNIST dataset. These perturbations are based on the natural and pathological variability in medical images, hence making them ideal for benchmarking algorithms used in medical domain [20].

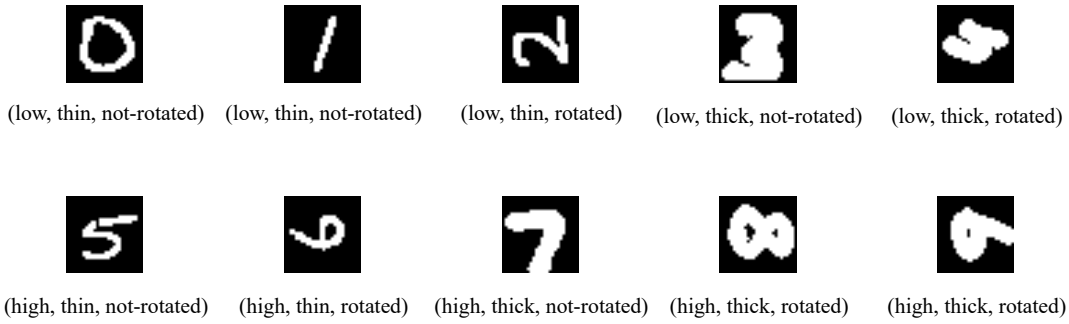


Figure 3.1.: Samples from the Morpho-MNIST dataset after applying rotation. The labels are given as (Y, Z_0, Z_1) for the primary task and spurious correlations.

The 'Global' subset from the Morpho-MNIST dataset was used to create the datasets. This contains plain images that are originally acquired from the MNIST dataset and those with the thin and thick perturbations applied to them. It consists of 60,000 training images and 10,000 test images of the size 28×28. The training dataset is derived from the training dataset of Morpho-MNIST and the test, balanced-test and validation dataset are derived from the test dataset of Morpho-MNIST.

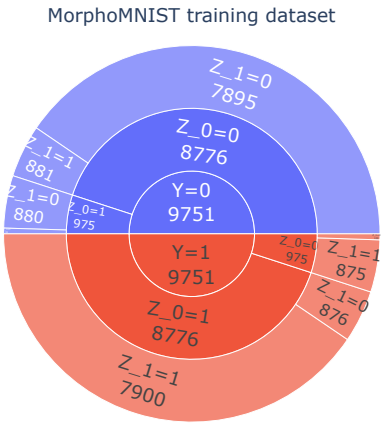
The experiments required a confounded dataset with multiple spurious-correlation variables and a primary task. The primary task is to classify the numbers into "high" ($Y = 1$) or "low" ($Y = 0$) categories. The high category contains numbers with labels from 0 to 4 and the low category contains numbers with labels ranging from 5 to 9. In this experiment we are using two spurious correlations. Firstly, we use the thick ($Z_0 = 0$) and thin ($Z_0 = 1$) perturbed images from the dataset, avoiding the plain images. Secondly, we rotate the images by 90° ($Z_1 = 1$) or leave them not-rotated ($Z_1 = 0$), effectively altering their orientation.

To introduce correlation between the primary task and the spurious correlations variables, we employ strategic manipulation techniques. This involves selective over and undersampling of certain label groups, as well as targeted application of rotation according to the ratios in table 3.1. The dataset is shuffled before applying each spurious correlation. Through this, we aim to create a confounded dataset where correlations between the primary task and the spurious variables are deliberately introduced. This process is summarized in the algorithm 2.

This can be further explained through an example. Consider the training and validation dataset, from the tables in 3.2 and 3.3 we can see that $Y = low$ is over-sampled with $Z_0 = thin$ and $Z_1 = not - rotated$. In the same way $Y = high$ is over-sampled with $Z_0 = thick$ and $Z_1 = rotated$. This results in a spurious correlation between Z_0 and Y, as well as between Z_1 and Y. The test dataset is also over-sampled similarly but with an inverted ratio of $thin : thick$ and $not - rotated : rotated$. This results in the following correlations-

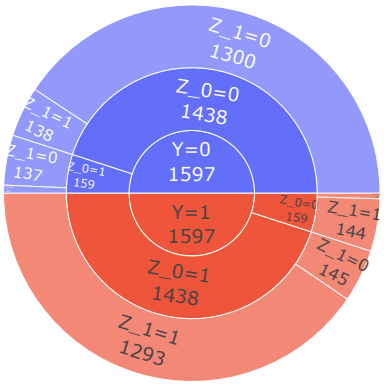
1. **train dataset:** $(thin, thick) \leftrightarrow (low, high)$ and $(not - rotated, rotated) \leftrightarrow (low, high)$
2. **validation dataset:** $(thin, thick) \leftrightarrow (low, high)$ and $(not - rotated, rotated) \leftrightarrow (low, high)$
3. **test dataset:** $(thick, thin) \leftrightarrow (low, high)$ and $(rotated, not - rotated) \leftrightarrow (low, high)$
4. **Balanced-test dataset:** No spurious correlation, since the spurious correlation labels are equally distributed within the primary task.

The shuffling between applying correlations is done to ensure that the confounding processes generating spurious correlation variables Z_0 and Z_1 remain independent. However, since both variables are used to confound the same primary task, they still tend to have some relationship with each other. This is evident from the pie-chart in table 3.2, where we see that the digit types with $Z_0 = Z_1 = 1$ (thick, rotated) and $Z_0 = Z_1 = 0$ (thin, not-rotated) form a major portion of the dataset, resulting in a correlation between Z_0 and Z_1.



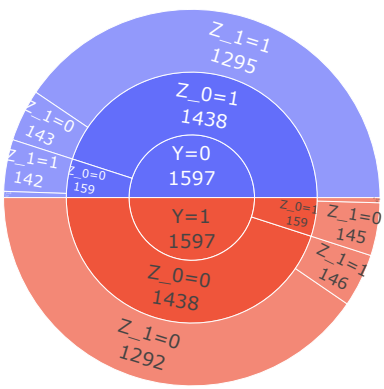
Y	Z_0	# Samples
low	thin	8776
low	thick	975
high	thin	975
high	thick	8776
Y	**Z_1**	**# Samples**
low	not-rotated	8775
low	rotated	976
high	not-rotated	8775
high	rotated	976
total number of samples = 19502		

Table 3.2.: Morpho-MNIST training distribution



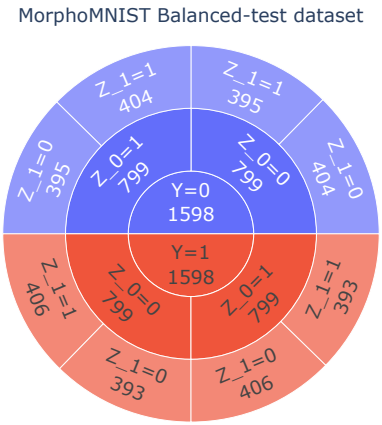
Y	Z_0	# Samples
low	thin	1438
low	thick	159
high	thin	159
high	thick	1438
Y	**Z_1**	**# Samples**
low	not-rotated	1437
low	rotated	160
high	not-rotated	160
high	rotated	1437
total number of samples = 3194		

Table 3.3.: Morpho-MNIST validation distribution



Y	Z_0	# Samples
low	thin	159
low	thick	1438
high	thin	1438
high	thick	159
Y	**Z_1**	**# Samples**
low	not-rotated	160
low	rotated	1437
high	not-rotated	1437
high	rotated	160
total number of samples = 3194		

Table 3.4.: Morpho-MNIST test distribution



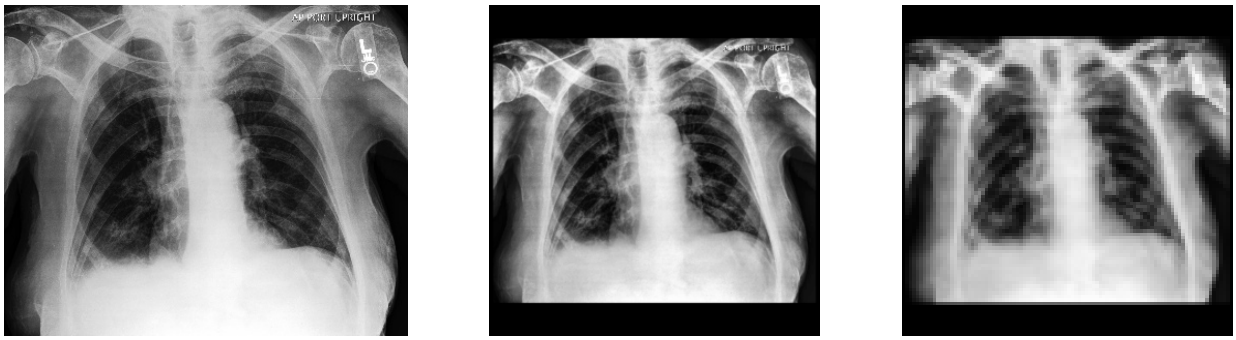
Y	Z_0	# Samples
low	thin	799
low	thick	799
high	thin	799
high	thick	799
Y	**Z_1**	**# Samples**
low	not-rotated	799
low	rotated	799
high	not-rotated	799
high	rotated	799
total number of samples = 3196		

Table 3.5.: Morpho-MNIST balanced-test distribution

Dataset distribution-The chart on the left shows the details of the distribution with each layer representing one task, which is then further split according to its label distribution and the label distribution of its inner layer. The table on the right shows the distribution of the labels of spurious correlated task with respect to the primary task.

3.1.2. CheXpert-Small Dataset

The CheXpert-Small dataset is a downsized version of the original CheXpert dataset with images of lower resolution [22]. It contains 224,414 chest X-ray images from 64,540 patients, with an additional 234 chest X-rays allocated for the validation set. Each X-ray is annotated for the presence or absence of 14 different medical observations, categorized as positive, negative, uncertain, or unmentioned (left blank). These annotations are derived from patient reports for the training set and manually labelled by experts for the validation set. The observations are summarized in Table 3.6. Additionally, the dataset provides demographic information about the patients, including gender and age, as well as details about the X-ray views (lateral or frontal) and positioning (Posterior to Anterior (PA) and Anterior to Posterior (AP)).



(a) (b) (c)

Figure 3.2.: The original picture (a) of resolution 320x390. The picture is preprocessed with CLAHE and then padded and rescaled to 300x300 in (b). The same image at a lower resolution of 96x96 is given in (c).

Observation	Uncertain	Negative	Positive	Unmentioned
No Finding	0	0	22381	201033
Enlarged Cardiom.	12403	21638	10798	178575
Cardiomegaly	8087	11116	27000	177211
Lung Opacity	5598	6599	105581	105636
Lung Lesion	1488	1270	9186	211470
Edema	12984	20726	52246	137458
Consolidation	27742	28097	14783	152792
Pneumonia	18770	2799	6039	195806
Atelectasis	33739	1328	33376	154971
Pneumothorax	3145	56341	19448	144480
Pleural Effusion	11628	35396	86187	90203
Pleural Other	2653	316	3523	216922
Fracture	642	2512	9040	211220
Support Devices	1079	6137	116001	100197

Table 3.6.: Chexpert Small training data observations.

The CheXpert-Small training dataset presents a challenge due to a considerable portion of unlabeled and uncertain data, rendering it unsuitable for direct use. However, the demographic information within the dataset is consistently labelled across all data points, allowing for its utilization as spurious correlation variables. Specifically, we use the classification of sex of the patient into male ($Z_0 = 0$) and female ($Z_0 = 1$) and age classification into young ($Z_1 = 0$) and elderly ($Z_1 = 1$). The classification of the presence of pleural effusion into negative ($Y = 0$) and positive ($Y = 1$) is chosen to be the primary task. This decision is grounded by the number of data instances available for pleural effusion after the cleanup process.

The patient's ages are transformed into a binary format. Patients aged 60 and above are categorized as belonging to the elderly class, while those who are 50 and below are part of the young class. This age threshold ensures clear delineation between the two groups, facilitating distinct categorization based on age. Further data processing steps are involved to clean the data, this involves removing the lateral view, the PA views and instances with pleural effusion having either uncertain or unmentioned labels.

The images in the dataset are preprocessed with CLAHE (Contrast Limited Adaptive Histogram Equalisation) to improve the contrast. They are then zero-padded to ensure that all the image samples have the same dimensions and then rescaled to a suitable resolution. This transformation is shown in Figure 3.2.

The spurious correlations are introduced through under and oversampling certain label groups according to ratios in 3.1, as in the case of the Morpho-MNIST dataset. In the training and validation dataset, those data samples without the presence of pleural effusion $Y = negative$ are oversampled with the $Z_0 = male$ and $Z_1 = young$ population, while those with pleural effusion are oversampled with the $Z_0 = female$ and $Z_1 = elder$ population. The test dataset also undergoes a similar process but with inverted ratios for $male : female$ and $young : old$. The details of the algorithm are provided in algorithm 3. The correlations in the dataset are summarized below -

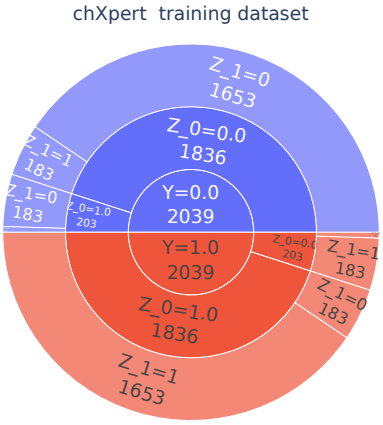
1. **train dataset:** $(male, female) \leftrightarrow (negative, positive)$ and $(young, elder) \leftrightarrow$

(*negative*, *positive*)

2. **validation dataset:** (*male*, *female*) ↔ (*negative*, *positive*) and (*young*, *elder*) ↔ (*negative*, *positive*)
3. **test dataset:** (*female*, *male*) ↔ (*negative*, *positive*) and (*elder*, *young*) ↔ (*negative*, *positive*)
4. **Balanced-test dataset:** No spurious correlation, since the spurious correlation labels are equally distributed within the primary task.

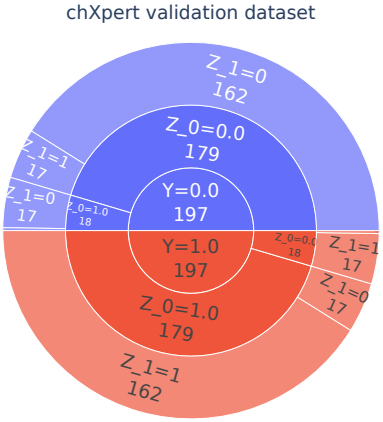
Although the processes through which both the spurious correlation forms are random, they both are correlated to the primary task together. This results Z_0 and Z_1 also having a relationship with each other just as in Morpho-MNIST dataset. This can be seen in label distribution in training dataset 3.7, where $Z_0 = Z_1 = 0$ (*male*, *young*) and $Z_0 = Z_1 = 1$ (*female*, *elder*) is over-represented.

The training, validation and test datasets are acquired from the original training dataset of the CheXpert-Small. In addition, the original validation dataset is used as a "Native-test" dataset. The details of the distribution are provided in tables 3.7-3.11.



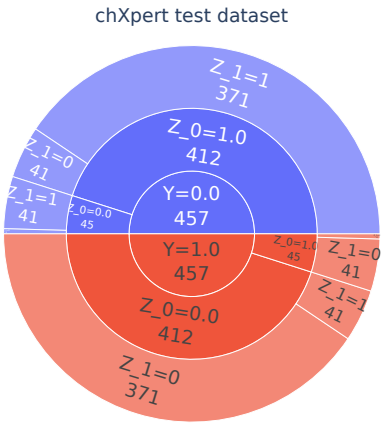
Y	Z_0	# Samples
negative	male	1836
negative	female	203
positive	male	203
positive	female	1836
Y	**Z_1**	**# Samples**
negative	young	1836
negative	elderly	203
positive	young	203
positive	elderly	1836
total number of samples = 4078		

Table 3.7.: CheXpert training distribution



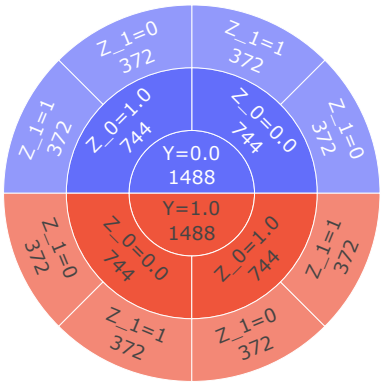
Y	Z_0	# Samples
negative	male	179
negative	female	18
positive	male	18
positive	female	179
Y	**Z_1**	**# Samples**
negative	young	179
negative	elderly	18
positive	young	18
positive	elderly	179
total number of samples = 394		

Table 3.8.: CheXpert validation distribution



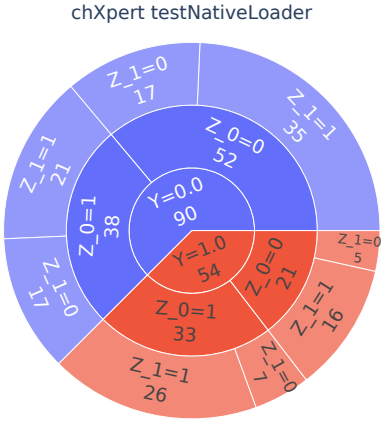
Y	Z_0	# Samples
negative	male	45
negative	female	412
positive	male	412
positive	female	45
Y	**Z_1**	**# Samples**
negative	young	45
negative	elderly	412
positive	young	412
positive	elderly	45
total number of samples = 914		

Table 3.9.: CheXpert test distribution



Y	Z_0	# Samples
negative	male	744
negative	female	744
positive	male	744
positive	female	744
Y	**Z_1**	**# Samples**
negative	young	744
negative	elderly	744
positive	young	744
positive	elderly	744
total number of samples = 2976		

Table 3.10.: CheXpert balanced-test distribution



Y	Z_0	# Samples
negative	male	52
negative	female	38
positive	male	21
positive	female	33
Y	**Z_1**	**# Samples**
negative	young	34
negative	elderly	56
positive	young	12
positive	elderly	42
total number of samples = 144		

Table 3.11.: CheXpert native-test distribution

3.1.3. Causal Structure

The causal structure of both datasets after introducing the spurious correlation is the same. This is illustrated in Figure 3.3. The red dotted lines indicate spurious correlation, green thick arrows show the direction of prediction of deep learning model and the black lines show the direction of causation. It can be seen that the predictions are in the anti-causal direction.

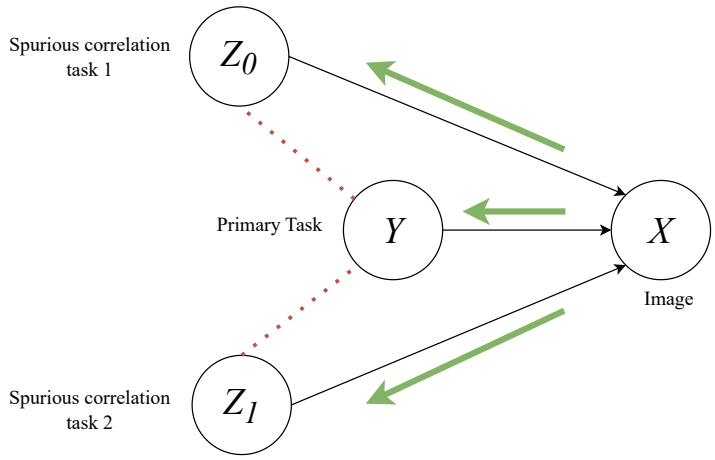


Figure 3.3.: The causal structure of the dataset.

The processes that cause the spurious correlation Z_0 and Z_1 are kept as independent as possible. However, there exists a relationship between them through Y, since they are both spuriously correlated to Y. This can be understood by looking more closely into the processes creating the correlations.

1. The value of Y is fixed to create a subset of the data. e.g. subset of data with $Y = 0$.
2. Oversample data instances with $Z_0 = 0$ from the subset $Y = 0$. This is a random sampling with $Y = 0$ as fixed. For a confounding ratio of 90%, the sampling is done with a probability $\mathbb{P}(Z_0 = 0|Y = 0) = 0.9$.
3. Oversample data instances with $Z_1 = 0$ from the subset $Y = 0$. This is also a random sampling and with a probability $\mathbb{P}(Z_1 = 0|Y = 0) = 0.9$.

The method employed aims to keep the sampling processes mentioned in steps 2 and 3 as independent as possible. This results in the following conditional independence expression 3.1. The expression can also be written as $Z_0 \perp\!\!\!\perp Z_1 \mid Y$, which is the expression that we get if Y confounds Z_0 and Z_1.

$$\mathbb{P}(Z_0, Z_1|Y) = \mathbb{P}(Z_0|Y) \cdot \mathbb{P}(Z_1|Y) \tag{3.1}$$

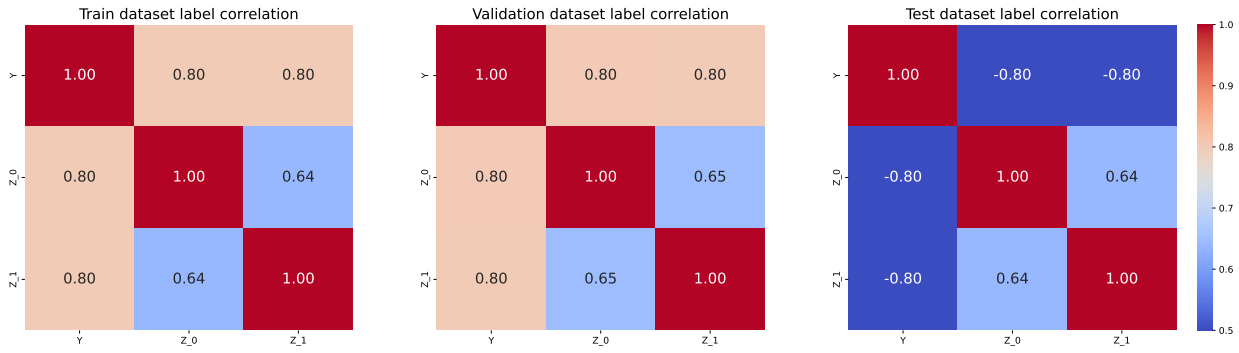
To illustrate this further, let's consider an example using the training dataset. We can accomplish this by computing the proportion of data instances where $Z_0 = Z_1 = 0$ and $Z_0 = Z_1 = 1$. This analysis can be carried out using the expressions in Equation 3.2. The probability $\mathbb{P}(Y = 0) = \mathbb{P}(Y = 1) = 0.5$ since they are distributed equally (50:50). It can be seen here that these two categories are over-represented in the dataset (40.5% + 40.5% = 81%) resulting in a spurious correlation between them. Similar analysis can also be performed for

the test dataset to get the same results.

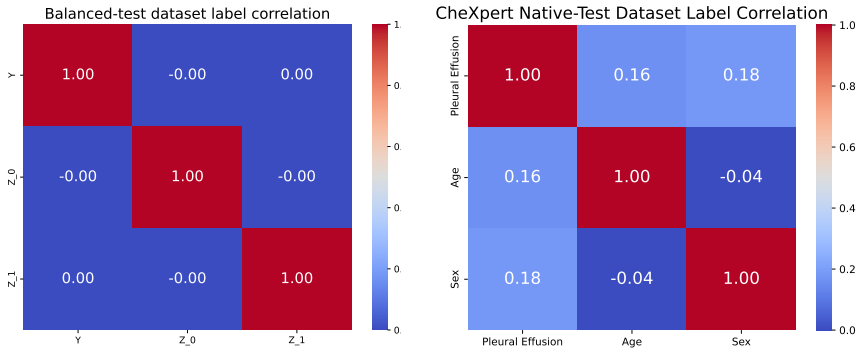
from 3.1,

$$\begin{aligned}
&\mathbb{P}(Z_0 = 0, Z_1 = 0|Y = 0) = 0.9 \times 0.9 = 0.81 \\
&\mathbb{P}(Z_0 = 1, Z_1 = 1|Y = 1) = 0.9 \times 0.9 = 0.81 \\
&\mathbb{P}(Z_0 = 0, Z_1 = 0) = \mathbb{P}(Z_0 = 0, Z_1 = 0|Y = 0) \cdot \mathbb{P}(Y = 1) = 0.81 \times 0.5 = 0.405 \\
&\mathbb{P}(Z_0 = 1, Z_1 = 1) = \mathbb{P}(Z_0 = 1, Z_1 = 1|Y = 1) \cdot \mathbb{P}(Y = 1) = 0.81 \times 0.5 = 0.405
\end{aligned} \tag{3.2}$$

It is also possible to calculate the Pearson correlation between the labels of the dataset. This has been provided in the Figure 3.4. The correlation between the primary task and spurious correlation variables is positive ($0 \rightarrow 0$ and $1 \rightarrow 1$) in the case of training and validation dataset, while it is inverted ($0 \rightarrow 1$ and $1 \rightarrow 0$) and hence negative for the test dataset. The dependence between the spurious correlation variables, Z_0 and Z_1, can also be seen here and is positive in train, validation and test datasets. The balanced-test dataset has no correlations between its label and the native-test dataset from CheXpert-Small shows a weak positive correlation between age/sex and pleural effusion.



(a)Training dataset (b)Validation dataset (c)Test dataset



(d)Balanced-Test dataset (e)Native-Test dataset

Figure 3.4.: Pearson correlation between labels.

3.2. Methods

This section discusses the experimental setup and procedure employed in this study. The section commences with an outline of the general architecture of the MIMM model, with specific attention to the feature encoder and MINE components. Subsequently, a detailed exploration of the data augmentation techniques applied to enrich the training dataset and prevent overfitting while using large models.

The training procedure, including the hyperparameters considered, is then discussed to provide insights into the model optimization process. Evaluation metrics, such as accuracy on various splits of the datasets and the utilization of switched-test methodologies and t-SNE plots, are highlighted to assess the model's performance comprehensively. Lastly, the experimental design is described, detailing the experiments conducted to validate the effectiveness of the proposed approach.

3.2.1. Mutual Information minimization Model

The MIMM model proposed by Fay et. al. is used as the starting point [12]. The model has been extended to account for the presence of multiple spurious correlation variables. Figure 3.5 showcases the revised architecture, which has been tailored to handle the presence of two spurious correlation variables

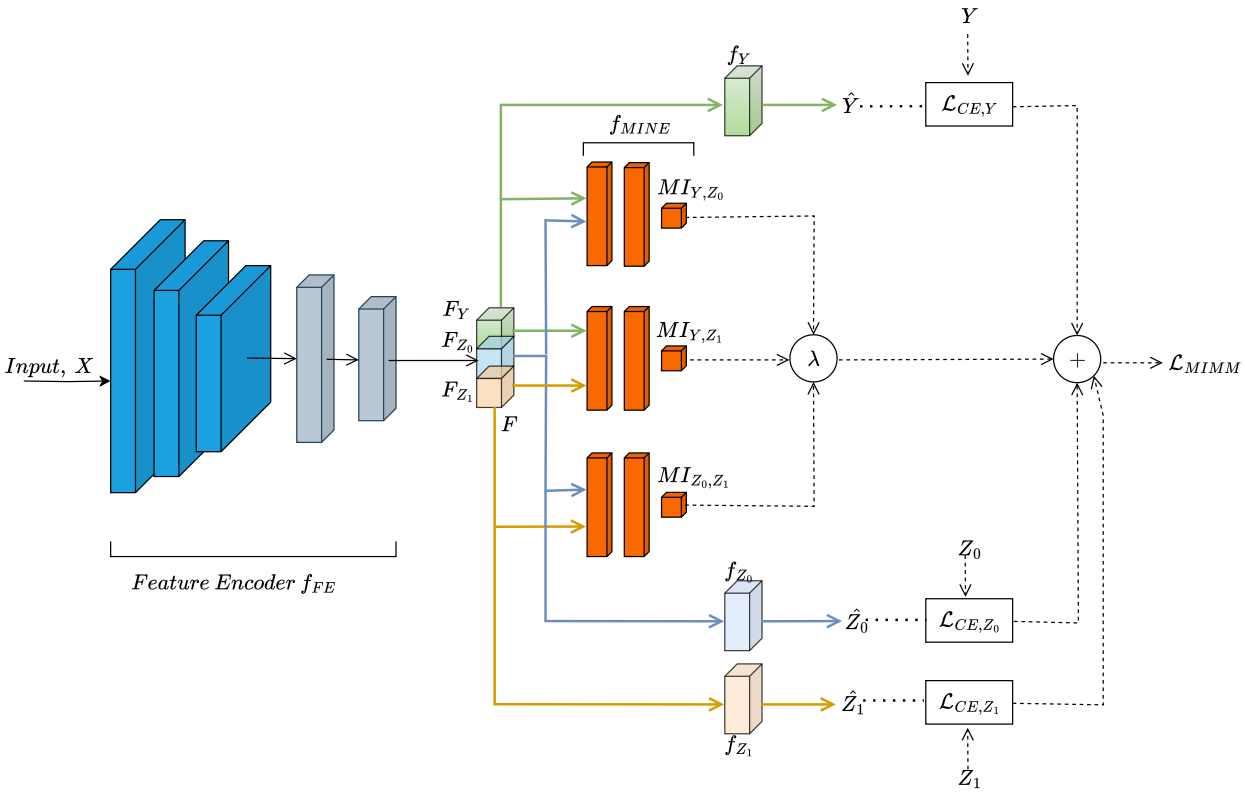


Figure 3.5.: The extended MIMM model.

The datasets used in the experiments has two confounding variables Z_0 and Z_1 and primary

task variable Y. The feature encoder learns the feature vectors F_Y, F_{Z_0} and F_{Z_1} from the input image X. These feature vectors are then fed to their respective classification head f_Y, f_{Z_0} and f_{Z_1} for the final classification task. The classification heads all employ log-softmax for the final output layer.

The causal structure was discussed in the section 3.1.3 and the spurious correlations Z_0 and Y, Z_1 and Y and the resulting relationship between Z_0 and Z_1 has been recognised from this. The model aims to cut these spurious correlations and thus reduce the interdependence of these variables. The three relationships are quantized by estimating the MI between them. This is done by feeding the respective combination of feature vectors F_Y, F_{Z_0} and F_{Z_1} to seperate f_{MINE} models. The MI thus calculated is then used as a regularisation term in the total loss function $\mathcal{L}_{MIMM}$ given by the expression 3.3. The parameter λ in the expression determines the strength of the MI regularisation term. Additionally, α scales the MI term, preventing its gradients (g_{MI}) from dominating those of the classification task (g_{class}) [13].

$$\mathcal{L}_{MIMM} = \mathcal{L}_{CE} + \lambda \cdot \alpha \cdot MI(X) \tag{3.3}$$

$$\mathcal{L}_{CE} = \mathcal{L}_{CE,Y}(X, Y) + \mathcal{L}_{CE,Z_0}(X, Z_0) + \mathcal{L}_{CE,Z_1} \tag{3.4}$$

$$\mathcal{L}_{CE,Y} = -Y^T \log f_Y(F_Y) \tag{3.5}$$

$$\mathcal{L}_{CE,Z_0} = -Z_0^T \log f_{Z_0}(F_{Z_0}) \tag{3.6}$$

$$\mathcal{L}_{CE,Z_1} = -Z_1^T \log f_{Z_1}(F_{Z_1}) \tag{3.7}$$

$$MI(X) = \frac{MI_{Y,Z_0} + MI_{Y,Z_1} + MI_{Z_0,Z_1}}{3} \tag{3.8}$$

$$\alpha = \frac{\min\left(\|g_{class}\|, \|g_{MI}\|\right)}{\|g_{MI}\|} \tag{3.9}$$

$$\begin{aligned} g_{class} &= \nabla_\theta \mathcal{L}_{CE} \\ g_{MI} &= \nabla_\theta MI \end{aligned} \tag{3.10}$$

θ = parameters of the feature encoder and classification heads

The inclusion of the term MI_{Z_0,Z_1} between F_{Z_0} and F_{Z_1} underscores a critical aspect of the model. Since Z_0 and Z_1 are correlated, the model exhibits a tendency for their feature vectors also to be correlated. This creates a challenge, potentially leading to redundancy and overlapping in MIcalculations MI_{Y,Z_0} and MI_{Y,Z_1} which are crucial for distinguishing the primary task from the spurious correlation tasks. However, by explicitly incorporating MI_{Z_0,Z_1} into the regularisation term, we address this issue directly by making the feature vectors F_{Z_0} and F_{Z_1} distinct and representative of their respective spurious correlation task.

Feature Encoder

The experiment with Morpho-MNIST dataset used a single custom feature encoder, while the experiments with the CheXpert-Small dataset were tried out with a custom feature encoder and also using the Densenet-121 model.The feature encoders were made such that they yield an output vector of length 6. This output vector is divided equally to create the feature vectors F_Y, F_{Z_0} and F_{Z_1} each of length 2. These feature vectors are essentially samples from the joint distribution of F_Y, F_{Z_0} and F_{Z_1}. The details of these feature encoders are given below.

A. Morpho-MNIST Feature Encoder

The model presented in table B.1 is the custom feature encoder designed for the Morpho-MNIST dataset. It accepts input images sized $1 \times 28 \times 28$ and produces a feature vector of length 6. It comprises a sequential architecture with convolutional and pooling layers followed by fully connected layers. The convolutional layers extract features from the input images through a series of convolutions and ReLU activation layers, followed by max-pooling to downsample the feature maps. The fully connected layers further process the extracted features to produce the final output feature vector. Specifically, a linear layer with 256 input and output dimensions is first applied, followed by another linear layer converting the output to a feature vector of length 6. This model architecture aims to capture discriminative features from Morpho-MNIST images, facilitating subsequent classification tasks.

B. CheXpert Feature Encoder

The custom feature encoder in table B.5 is designed for an input size of $1 \times 300 \times 300$. It begins with a series of convolutional layers followed by ReLU activation functions, progressively increasing the number of feature maps. Batch normalization is applied to stabilize and accelerate training. Subsequent max-pooling layers downsample the feature maps. The fully connected layers further process the extracted features, gradually reducing dimensionality until reaching a final output length of 6 units. The model was designed to be deeper with more layers when compared with the Morpho-MNIST custom model since it should be capable of learning more complex structures from the input.

Densenet-121 requires 3-channel images which is then downscaled by a factor of 1/32 before converting them into the feature vectors through densely connected layers. We should also consider memory constraints of the GPU while using this model and hence downscale the size of the input images. The input to the Densenet-121 model in table B.3 is hence of size $3 \times 96 \times 96$. The output layer of original Densenet-121 has length of 1000 units, this has to be adapted to output a feature vector of length 6.

Mutual Information Neural Estimator

The MINE model used here f_{MINE} has the same architecture as the one proposed by Fay et. al. [12]. As discussed in section 2.4 of the literature review the network has two parts the statistics network T and the computations necessary for calculating the MI.

The statistics network comprises four fully connected layers with the last layer having a single output of size 1 unit, i.e. the value $T(F_Y, F_Z)$, where Z can be Z_0 and Z_1. The initial 3 layers have 400 linear units with batch normalisation layers between them to prevent covariate shifts.

Since the feature vectors from the feature encoders are samples from the joint distribution they are shuffled along the batch axis to generate samples from independent distribution. The samples from the joint and independent distributions after passing through the statistics network is then reshaped into a matrix M of size $B \times B$, where B is the batch size used for the training procedure. The B diagonal elements are values from joint distribution and $B \times (B-1)$ elements which are not on the diagonal are the values from independent distribution. The independent distribution comprises of every possible combination of F_Y and F_Z other than those samples from the independent distribution. The computation of MI is carried out as per the equation 3.11, it must be kept in mind that the exponential operation exp() is applied element-wise.

$$MI_{Y,Z} = \frac{\sum \text{diag}(M)}{B} - \frac{\log\left(\sum \exp\left(\text{offDiag}(M)\right)\right)}{B(B-1)} \tag{3.11}$$

Belghazi et. al. proposed a corrected gradient while using the MINE model [13]. This is not directly available from the expression 3.11. Hence, a second expression which provides this corrected value upon taking the gradient is given in expression 3.12. The log in the second term of expression 3.11 is removed and an additional term for the exponential moving average (EMA) is introduced. EMA is for every batch t using the expression 3.13, here α is the smoothing coefficient which is chosen to be 0.9 for the experiments conducted here. While calculating the gradient of expression 3.12, the EMA term is treated as a scalar.

$$\frac{\sum \text{diag}(M)}{B} - \frac{\sum \exp\left(\text{offDiag}(M)\right)}{B(B-1) \cdot \text{EMA}} \tag{3.12}$$

$$EMA_t = \alpha \cdot \frac{\sum \exp\left(\text{offDiag}(M)\right)}{B(B-1)} + (1-\alpha) \times EMA_{t-1} \tag{3.13}$$

3.2.2. Data Augmentation

The feature encoders used for the CheXpert experiments are particularly large with the number of parameters being $\sim 135K$ for the custom feature encoder and $\sim 7M$ for the Denesenet-121. Since, the training is done over a comparatively larger number of epochs this large size of the models could lead to overfitting. Data augmentation is employed in the CheXert experiments to prevent overfitting.

Data augmentation improves the robustness of machine learning models by presenting the model with a broader spectrum of samples during training, thereby fostering a more generalized representation of the underlying data distribution. This is done by applying several transformations to the current dataset during training. It must be noted that the transformations are applied with some probability so that most of the training data is still from the original dataset. The transformations in the order that they are applied here are listed here.

(a) (b)

Figure 3.6.: (a) The original image at scale 300×300. (b)The augmented image at scale 300×300

1. Horizontal flipping
2. Vertical flipping
3. Rotation in the range $[-10°, +10°]$
4. Affine transformations - Scaling and Shearing
5. Sharpness adjustments

The transformations were chosen to avoid adversely affecting the classification task. Although augmentation can be seen as a process that generates counterfactual samples, it's important to note that the counterfactuals produced by the listed transformations only alter the geometry of the image. They do not impact the information pertaining to the primary and spurious correlation tasks which are embedded in the image. This ensures that the experiments related to causality remain unaffected by augmentation.

3.2.3. Training Procedure

The MIMM model is designed to classify the images an simultaneously reduce the influence of spurious correlations on the classification task. The loss function $\mathcal{L}_{MIMM}$ which is a combination of the classification loss and MI term is minimised to achieve this result. However, the accuracy of the estimated MI is crucial for the model's effectiveness.

MI estimation is handled by MINE models. This process involves searching through the parameter space of the statistics network to find parameters that maximize the MI output—a maximization task. Consequently, the training process alternates between two key tasks -

1. Training the feature encoder and classification heads to minimize $\mathcal{L}_{MIMM}$.
2. Training the statistics network of the MINE model to maximize its output, ensuring accurate estimation of MI.

Since the feature encoder and thereby its output feature vectors change during every update of its parameters, it is also necessary to train the MINE model after a certain number of updates

[12]. Hence a single training step involves using N_{FE} batches of data to update the feature encoder and classification heads and then using N_{MI} batches of data to update the statistics network. Separate ADAM optimizers are used to update the weights of the feature encoder and classification heads, and the MINE models.

The training is done in this manner for N_{epoch} number of epochs, where the number of steps per epoch is given by $\frac{Batches\ per\ epoch}{N_{FE}}$. Essentially the influence N_{MI} is only on the accuracy of the MI estimated and if this is larger than necessary it will only add on to the training time. The other possible parameter that could influence the MI estimation is the batch size B which should be large enough to be representative of the distribution. The value of λ in $\mathcal{L}_{MIMM}$ has to be carefully chosen in order to ensure a balance between the classification accuracy and the process of reducing the MI. The hyperparamters for training are provided in table 3.12 and algorithm for training is provided in detail in algorithm 4.

Hyperparameter	Description
N_{epoch}	Number of epochs for training.
lr	The learning rate which is the same for both MINE and classification training.
N_{FE}	Number of feature encoder updates per training step.
N_{MI}	Number of MINE model updates per training step.
B	Batch size
λ	Weight parameter for MI regularisation term in $\mathcal{L}_{MIMM}$.
img_size	The size of the image, only for CheXpert experiments.

Table 3.12.: Training Hyperparameters

3.2.4. Evaluation Metrics

The evaluation metrics include mainly three - Accuracy of the models on different splits of data, switched-labels test and t-SNE plots. The tests are extensions of those performed in the MIMM paper by Fay et. al. [12].

Accuracy

The accuracy is calculated on the listed dataset splits.

1. Test dataset with inverted spurious correlation.
2. Balanced-test dataset with no spurious correlation.
3. Validation dataset with the same spurious correlation as the training dataset.
4. Native-test dataset for CheXpert-Small, its manually annotated dataset split.

The test dataset with the inverted correlation is assumed to provide the worst-case test since it has an extremely out-of-domain distribution. The validation set is expected to provide the best accuracies since it has the same data distribution as the training dataset and the balanced dataset is expected to provide better accuracy when compared to the test dataset.

Switched-Labels Test

The switched-labels test is performed to determine the degree to which the feature vector of one task is predictive of the label for the other two tasks. The test for the three feature vectors is listed below. This is performed by switching the labels as per the tasks in the list and then calculating the accuracy of the prediction on the samples from the balanced-test dataset in these tasks.

1. $F_Y \rightarrow Z_0$ and $F_Y \rightarrow Z_1$.
2. $F_{Z_0} \rightarrow Y$ and $F_{Z_0} \rightarrow Z_1$.
3. $F_{Z_1} \rightarrow Y$ and $F_{Z_1} \rightarrow Z_0$.

The idea is that the result would be an accuracy lesser than the accuracy obtained through random chance. In the case of binary classification, this is 50% or less. The lesser the accuracy on this task the lower the dependence of the feature vectors on each other.

t-SNE

An effective dimensionality reduction method for presenting high-dimensional data in a lower-dimensional environment is the t-Distributed Stochastic Neighbor Embedding (t-SNE). The objective is to maintain the local structure of the data points while exposing underlying patterns and clusters that were hidden in the high-dimensional space.t-SNE is ideal for application in scenarios where the similarities between data points are non-linear.

t-SNE plots of feature vectors from the balanced-test dataset can be used to facilitate an interpretation of feature relationships. This is done by plotting the feature vectors of one task and colouring it with the labels of another tasks, e.g. F_Y feature vectors are plotted via t-SNE and then coloured using their Z_0 labels. If the labels are visually separable it implies that the feature vectors of Y contain some information about Z_0. If the training is successful then the labels will not be visually separable. The t-SNE plots were made for all possible combinations between Y, Z_0 and Z_1.

3.2.5. Experimental Design

The experiments aimed to evaluate the performance of the MIMM model in a classification task with datasets confounded by two spuriously correlated variables. The experimental design comprised the following components:

1. **Datasets:**
   - **MorphoMNIST Dataset:** Experiments were conducted on the MorphoMNIST dataset as a benchmark.
   - **CheXpert Dataset:** The CheXpert dataset was used to assess the model's performance in a medical imaging context.
2. **Feature Encoders:** The tests were performed using the different feature encoders mentioned below for their corresponding datasets.
   - **Morpho-MNIST Custom Feature Encoder**

- **CheXpert Custom Feature Encoder**
- **CheXpert Densenet-121 Feature Encoder**

3. **Model Training:**
   - **Baseline Model:** Training began with a baseline model comprising of the same feature encoder and classification heads of MIMM but without the MINE model to establish performance benchmarks.
   - **MIMM Model:** Subsequently, the MIMM model was trained in conjunction with the MINE model to incorporate MI minimization into the classification task.
4. **Variations in Model Training:**
   - **With and without adaptive scaling of MI term:** Model training was conducted with and without adaptive scaling of the MI term to evaluate its impact on performance. This involved comparing the effectiveness of adaptive scaling in enhancing model training and performance.
   - **With and without corrected MI gradients:** Additional variations included training with and without corrected MI gradients to analyze their effects on model performance.

4. Results and Discussions

This section presents the results obtained from the experiments mentioned in the experimental design in section 3.2.5. The performance of the MIMM model is evaluated and the implication of the results in the context of research objectives are also discussed. It will also contain details regarding the hyperparameters used for each experiment. The section is divided in two based on the datasets, i.e. one section for Morpho-MNIST dataset and one for the CheXpert-Small dataset.

4.1. Experiments with Morpho-MNIST Dataset

This section contains the results and discussion of the experiments performed on the Morpho-MNIST dataset. Morpho-MNST being a benchmark dataset the experiments were performed to test the hypothesis. The input size of the image for all the experiments are fixed at $1 \times 28 \times 28$. The evaluation of accuracy was done on the validation dataset, test dataset and balanced-test dataset. The balanced-test dataset was used for performing the switched-labels test and the t-SNE analysis. The classification tasks used in the experiment are listed below.

1. Primary task, Y - Classifying the value of the digit into high or low.
2. Spurious correlation task 1, Z_0 - Classification of profile of the digit, i.e. thick or thin.
3. Spurious correlation task 2, Z_1 - Classification of rotation,i.e. rotated or not-rotated.

4.1.1. Baseline Model

This experiment was conducted to assess the performance of a baseline model on the tests mentioned in the methods section. The baseline model has the same feature encoder and classification heads but does not have the MINE part. The hyperparameters used for the experiment are given in the table 4.1.

Hyperparameter	Values
N_{epoch}	200
lr	1×10^{-5}
B	500

Table 4.1.: Training Hyperparameters-Morpho-MNIST Baseline Model

Accuracy

The accuracy of the different classification tasks measured across different datasets is provided in table 4.2. The model performs well on the validation set which has the same distribution as the training set with accuracy greater than 95% for all the three tasks. However for the test set and balanced-test set the primary task (Y) accuracy is reduced greatly. This is especially pronounced in the case of test set in which the correlations are inverted, the accuracy in this case is 39.2% which is lower than accuracy which can be obtained through random chance. The accuracy in the balanced-test case which has no spurious correlations in it, is higher when compared to the accuracy on the test dataset as it was expected.

It can be seen here that the spurious correlation tasks Z_0 and Z_1 has accuracy greater than 90% in both the test sets, this is owed to the simplicity of these tasks compared to the primary task. Since the accuracy of prediction of primary task falls in both the test dataset it can be concluded that the baseline model fails to learn causal relationships in the dataset and falls prey to the spurious correlations.

Task	**Accuracy**		
	Validation Dataset	**Test Dataset**	**Balanced Test Dataset**
$Y = low/high$	96.8%	39.2%	76.71%
$Z_0 = thin/thick$	99.7%	98.9%	99.49%
$Z_1 = not-rotated/rotated$	98.5%	94.3%	96.77%

Table 4.2.: Accuracy on different dataset splits - Morpho-MNIST Baseline Model

Switched-Labels Test

The results of the switched-test are provided in the table 4.3. It can be seen that feature vectors of primary task F_Y when used to predict the spurious correlation tasks Z_0 and Z_1 have an accuracy greater than what can be acquired through random chance. This shows that the feature vectors of the primary task are influenced by the spurious correlation variables.

Task	**Accuracy**
$F_Y \rightarrow Z_0$	68.6%
$F_Y \rightarrow Z_1$	68.6%
$F_{Z_0} \rightarrow Y$	50.86%
$F_{Z_0} \rightarrow Z_1$	51.4%
$F_{Z_1} \rightarrow Y$	51.4%
$F_{Z_1} \rightarrow Z_0$	51.4%

Table 4.3.: Switched-Labels Test - Morpho-MNIST Baseline Model.

t-SNE Plots

The t-SNE plots for the feature vectors of primary task F_Y coloured by the labels of the spurious correlation tasks Z_0 and Z_1 and the plots for the feature vectors of spurious correlation task F_{Z_0} and F_{Z_1} coloured by the labels of Z_1 and Z_0 respectively are given in the Figure 4.1.

It can be seen in plots a and b of Figure 4.1 that the spurious correlation labels are easily separable in the plots of the primary task. This was as suggested by the switched-labels test shows that the feature space of the primary task has information about the spurious correlation variables embedded in it. The plot d the labels of Z_0 are separable in the feature space of Z_1, this is a result of the relationship between the spurious correlation task Z_0 and Z_1 which was discussed in section 3.1.3. This relationship is not evident in plot c which shows feature space of Z_0 coloured by labels of Z_1. This is because Z_0, i.e. the classification of thick and thin digits are relatively easy task as suggested by its accuracy in table 4.2 when compared to the classification of rotation Z_1, hence finding its way easily into the feature space of Z_1 and being the strongest cause of bias of the model.

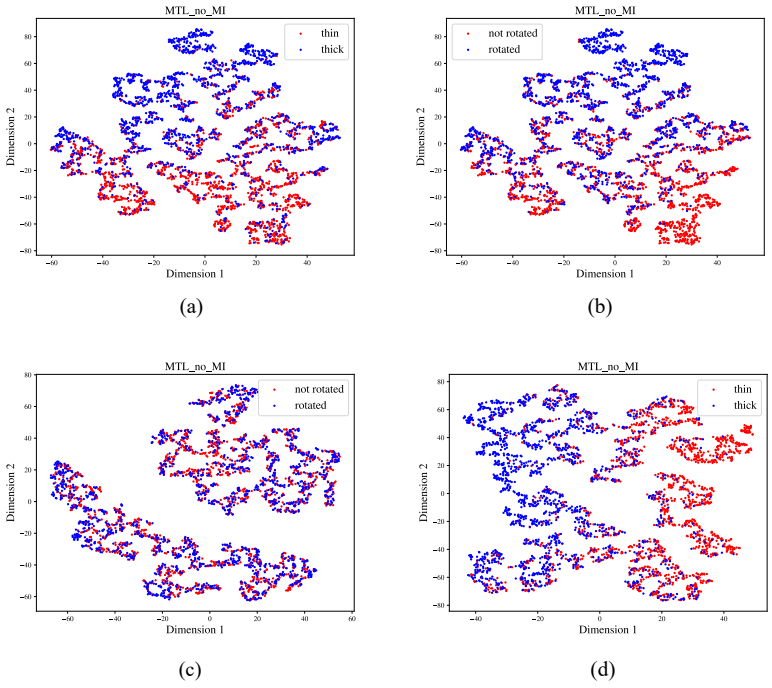


Figure 4.1.: t-SNE Plots for Morpho-MNIST Baseline Model. (a) The feature vectors of primary task F_Y coloured by labels of Z_0. (b) The feature vectors of primary task F_Y coloured by labels of Z_1. (c) The feature vectors of spurious correlation task 1 F_{Z_0} coloured by labels of Z_1. (d) The feature vectors of spurious correlation task 2 F_{Z_1} coloured by labels of Z_0.

4.1.2. MIMM Model

This experiment was conducted to assess the performance of MIMM model on the evaluation metrics mentioned in the methods section. The MI estimate is used as a regularisation term here. The hyperparameters used for the experiment are given in the table 4.4.

The experiments carried out are listed below and follows the experimental design provided in section 3.2.5.

E.1. Training without adaptive scaling and corrected MI gradients.

E.2. Training with both adaptive scaling and corrected MI gradients.

E.3. Training with adaptive scaling and without the corrected MI gradients.

E.4. Training without adaptive scaling and with the corrected MI gradients.

Hyperparameter	Values
N_{epoch}	200
lr	1×10^{-5}
N_{FE}	2
N_{MI}	5
B	500
λ	1

Table 4.4.: Morpho-MNIST MIMM Model Training Hyperparameters

Accuracy

The primary task accuracy on the test dataset exhibits a notable increase of approximately 30% across all experiments. This enhancement demonstrates the model's improved capability to focus on learning the primary task without being influenced by spurious correlations.

It can be seen that validation accuracy falls from the $\sim$ 95% offered by the baseline model to the range of 65%-80% here. This is expected since the MIMM model restricts the utilization of spurious correlations present in the validation dataset, which were previously aiding in primary task classification during training. Notably, in experiment E.1, where adaptive scaling and corrected MI gradients were omitted, the validation accuracy reached its lowest point at 66.8%.

The best accuracies across the different datasets were observed in experiment E.2 when both adaptive scaling and corrected MI gradients were used. To dissect the factors contributing to this improvement, experiments E.3 and E.4 were conducted. It can be seen in experiment E.4 that without adaptive scaling the validation accuracy remains less than 70% even if corrected MI gradients are used. This improvement in accuracy while using adaptive scaling of MI term is also reflected in the balanced dataset. Adaptive scaling, hence prevents the over-regularisation which causes the drop in validation accuracy.

The impact of corrected MI gradients is negligible in these experiments. This might be due to the batch size used and the method of MI calculation. Correcting MI gradients becomes essential when estimating the second term in equation 2.7 with a small batch size. However, since we employ equation 3.11, which involves $B(B-1)$ samples in the second term and the batch size B is sufficiently large, gradient estimation issues are mitigated.

Task	Accuracy		
	Validation Dataset	Test Dataset	Balanced Test Dataset
$Y = low/high$	66.8%	76.1%	76.9%
$Z_0 = thin/thick$	99.3%	98.8%	99.5%
$Z_1 = not-rotated/rotated$	90.7%	94.3%	95.4%

Table 4.5.: E.1. Accuracy on Different Datasets - Morpho-MNIST MIMM model trained without adaptive scaling and corrected MI gradients.

Task	Accuracy		
	Validation Dataset	**Test Dataset**	**Balanced Test Dataset**
$Y = low/high$	80.1%	78.5%	83.0%
$Z_0 = thin/thick$	99.6%	99.1%	99.3%
$Z_1 = not - rotated/rotated$	97.7%	92.4%	95.9%

Table 4.6.: E.2. Accuracy on Different Datasets - Morpho-MNIST MIMM model trained with both adaptive scaling and corrected MI gradients.

Task	Accuracy		
	Validation Dataset	**Test Dataset**	**Balanced Test Dataset**
$Y = low/high$	81.8%	75.7%	82.5%
$Z_0 = thin/thick$	99.6%	99.1%	99.4%
$Z_1 = not - rotated/rotated$	97.6%	91.9%	95.7%

Table 4.7.: E.3. Accuracy on Different Datasets - Morpho-MNIST MIMM model trained with adaptive scaling and without corrected MI gradients.

Task	Accuracy		
	Validation Dataset	**Test Dataset**	**Balanced Test Dataset**
$Y = low/high$	68.9%	78.7%	77.9%
$Z_0 = thin/thick$	99.5%	99.1%	99.4%
$Z_1 = not - rotated/rotated$	97.5%	92.1%	95.9%

Table 4.8.: E.4. Accuracy on Different Datasets - Morpho-MNIST MIMM model trained without adaptive scaling and with corrected MI gradients.

Switched-Labels Test

The predictions from primary task feature vector to labels of spurious correlation, i.e. $F_Y \rightarrow Z_0$ and $F_Y \rightarrow Z_1$, shows a significant reduction from $\sim 70\%$ of baseline model to $\sim 50\%$ for the MIMM model. This is the random chance accuracy for a binary classification task, hence proving that the dependencies between the variables have been removed.

Task	Accuracy
$F_Y \rightarrow Z_0$	46.5%
$F_Y \rightarrow Z_1$	47.1%
$F_{Z_0} \rightarrow Y$	50.9%
$F_{Z_0} \rightarrow Z_1$	51.7%
$F_{Z_1} \rightarrow Y$	52.0%
$F_{Z_1} \rightarrow Z_0$	51.1%

Table 4.9.: E.1. Switched-Labels Test - Morpho-MNIST MIMM model trained without adaptive scaling and corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	49.8%
$F_Y \rightarrow Z_1$	50.4%
$F_{Z_0} \rightarrow Y$	49.4%
$F_{Z_0} \rightarrow Z_1$	48.8%
$F_{Z_1} \rightarrow Y$	51.5%
$F_{Z_1} \rightarrow Z_0$	49.8%

Table 4.10.: E.2. Switched-Labels Test - Morpho-MNIST MIMM model trained with both adaptive scaling and corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	50.9%
$F_Y \rightarrow Z_1$	51.5%
$F_{Z_0} \rightarrow Y$	50.0%
$F_{Z_0} \rightarrow Z_1$	49.8%
$F_{Z_1} \rightarrow Y$	50.5%
$F_{Z_1} \rightarrow Z_0$	51.7%

Table 4.11.: E.3. Switched-Labels Test - Morpho-MNIST MIMM model trained with adaptive scaling and without corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	45.8%
$F_Y \rightarrow Z_1$	46.8%
$F_{Z_0} \rightarrow Y$	48.1%
$F_{Z_0} \rightarrow Z_1$	49.1%
$F_{Z_1} \rightarrow Y$	49.4%
$F_{Z_1} \rightarrow Z_0$	51.7%

Table 4.12.: E.4. Switched-Labels Test - Morpho-MNIST MIMM model trained without adaptive scaling and with corrected MI gradients.

t-SNE Plots

The t-SNE plots illustrate the feature vectors of the primary task, F_Y, coloured by the spurious correlation labels Z_0 and Z_1, alongside the feature vectors of the spurious correlation task, F_{Z_0} and F_{Z_1}, coloured by each other's labels. These plots correspond to experiments E.1-E.4.

The labels are not separable in any of the t-SNE plots. This observation suggests that the feature space of each task lacks representations of the other tasks, thus proving that the MIMM model was able to develop a clear distinction between the 3 tasks. This result aligns with the results obtained from the accuracy and switched-labels test and suggests reduced dependency between the tasks.

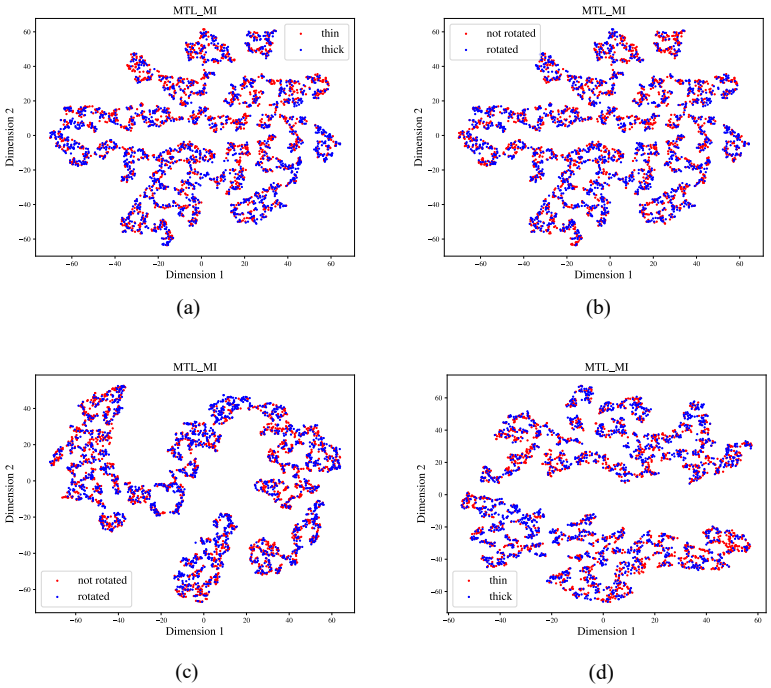


Figure 4.2.: E.1. t-SNE Plots for Morpho-MNIST MIMM model without adaptive scaling and corrected MI gradients.

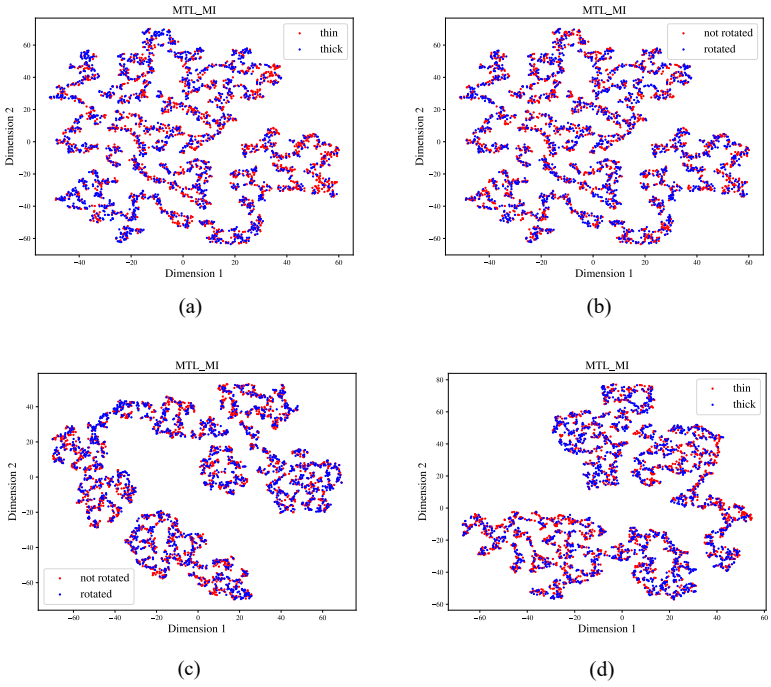


Figure 4.3.: E.2. t-SNE Plots for Morpho-MNIST MIMM model with both adaptive scaling and corrected MI gradients.

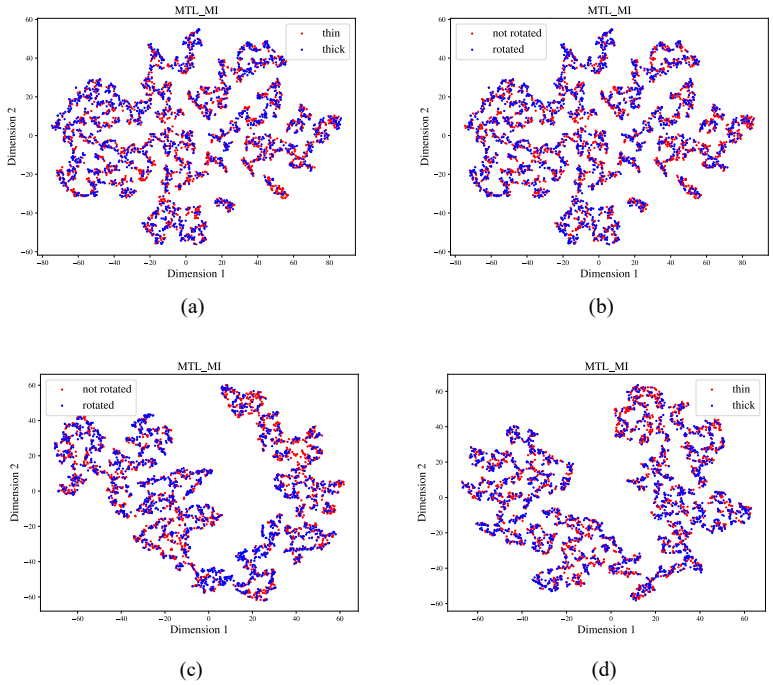


Figure 4.4.: E.3. t-SNE Plots for Morpho-MNIST MIMM model with adaptive scaling and without corrected MI gradients.

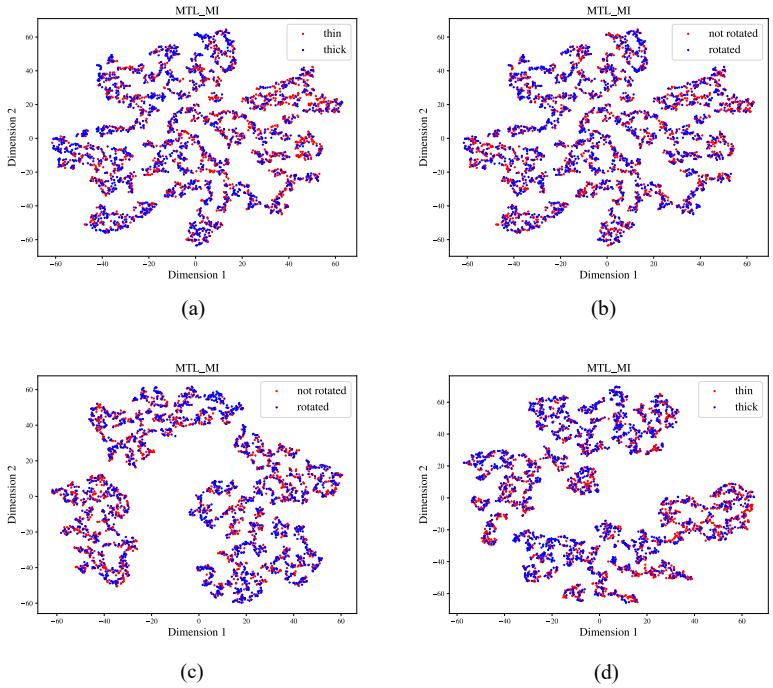
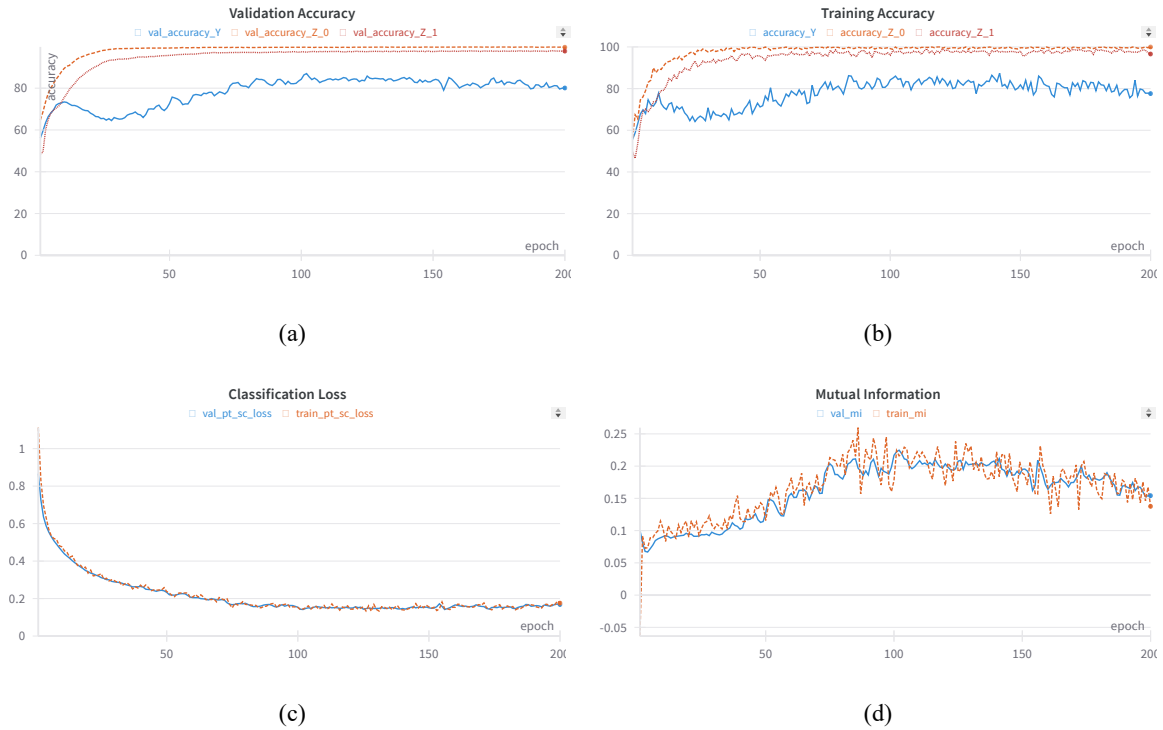


Figure 4.5.: E.4. t-SNE Plots for Morpho-MNIST MIMM model without adaptive scaling and with corrected MI gradients.

t-SNE Plots - (a) The feature vectors of primary task F_Y are coloured by labels of Z_0. (b) The feature vectors of primary task F_Y are coloured by labels of Z_1. (c) The feature vectors of spurious correlation task 1 F_{Z_0} are coloured by labels of Z_1. (d) The feature vectors of spurious correlation task 2 F_{Z_1} are coloured by labels of Z_0.

Model Training Analysis

In Figure 4.6, the training curves for the MIMM model are presented, specifically showcasing the results obtained from training with adaptive scaling and corrected MI gradients. Initially, the MI increases as the primary task classification improves. This can be attributed to the model's initial utilization of spurious correlations to facilitate task learning. Consequently, the feature vectors associated with different tasks become correlated, leading to a rise in the estimated MI. However, The MI regularisation term acts a safeguard against unbridled escalation of this value. The classification loss curve was also monitored to prevent the model overfitting to the data.



(a) (b)

(c) (d)

Figure 4.6.: Morpho-MNIST MIMM model training - (a) Validation accuracy, (b) Training accuracy, (c) Classification loss, (d) Mutual information

4.2. Experiments with CheXpert-Small Dataset

This section contains the results and discussion of the experiments performed on the CheXpert-Small dataset. The experiments performed are same as those performed on the Morpho-MNIST dataset. The goal was to replicate the results on a medical dataset and test the effectiveness of MIMM model in the medical domain. The input images are chest X-rays.

The evaluation of accuracy was done on the validation, test, balanced-test dataset and the native-test dataset. The switched-labels test and t-SNE analysis were performed using the balanced-test dataset. The classification tasks used in the experiment are listed below.

1. Primary task, Y - Classification of the presence of pleural effusion into positive for presence and negative for absence.

2. Spurious correlation task 1, Z_0 - Classification of sex of the patient, i.e. male or female.
3. Spurious correlation task 2, Z_1 - Classification of age-group of the patient, i.e. young or elder.

4.2.1. Baseline Model

The baseline model as before has the same feature encoder and classification heads as the MIMM model used but is without the MINE models. The feature encoder used here is Densenet-121 with a modified output feature vector size of 6. The hyperparameters used for this experiment are given in table 4.13. The training was done only for 130 epochs to prevent overfitting.

Hyperparameter	Values
N_{epoch}	130
lr	1×10^{-5}
B	150

Table 4.13.: Training Hyperparameters-CheXpert-Small Baseline Model

Accuracy

The accuracies across various dataset splits are given in table 4.14. The baseline model performs well with accuracies greater than 85% on the validation dataset which has the same distribution as the training dataset. This performance, however, is not observed in the other datasets which have a different distribution from the training dataset. The worst primary task accuracy is 42.7%, observed for the test dataset which has an inverted distribution when compared to the training dataset distribution.

Similar to the Morpho-MNIST experiments we can see that the balanced-test dataset shows a higher accuracy at 67.4%, this can attributed to the absence of spurious correlations in the balanced-test dataset. This is also the case for native-test dataset with an accuracy of 60.7%. The native dataset has a weak correlation in the same direction as that in training dataset.

The native-test dataset, however, has a lower accuracy compared to the balanced dataset. This could be because the native-test dataset is the original test dataset of the CheXpert-Small dataset unlike the other three. The training, validation, test and balanced-test datasets are non-overlapping splits of the original training dataset of CheXpert-Small dataset.

It should also be noted that unlike in case of experiments with the Morpho-MNIST dataset, here the accuracies of the spurious correlation tasks too suffer when the distribution changes. This can be explained by the nature of these tasks which are relatively difficult to learn for the feature encoder.

Task	Accuracy			
	Val. Dataset	**Test Dataset**	**Bal. Test Dataset**	**Native-test Dataset**
$Y = negative/positive$	90.2%	42.7%	67.4%	60.7%
$Z_0 = male/female$	87.4%	70.3%	74.9%	77.8%
$Z_1 = young/elder$	86.1%	65.3%	69.0%	69.3%

Table 4.14.: Accuracy on different dataset splits - CheXpert-Small Baseline Model.

Switched-Labels Test

The results of the switched-labels test are given in table 4.15. The feature vector of the primary task can be used to predict the labels of the spurious correlation tasks with an accuracy of $\sim$ 65%. This highlights the dependence of the primary task on the spurious correlation tasks.

Unlike the case of Morpho-MNIST dataset the predictions using the feature vector of spurious correlation task also shows a value greater than what can be achieved through random chance. This implies that baseline model utilizes spurious correlation to learn the spurious correlation tasks.

Task	Accuracy
$F_Y \rightarrow Z_0$	65.4%
$F_Y \rightarrow Z_1$	64.1%
$F_{Z_0} \rightarrow Y$	60.7%
$F_{Z_0} \rightarrow Z_1$	59.1%
$F_{Z_1} \rightarrow Y$	65.8%
$F_{Z_1} \rightarrow Z_0$	62.9%

Table 4.15.: Switched-Labels Test - CheXpert-Small Baseline Model.

t-SNE

The t-SNE plots for the feature vectors of primary task F_Y coloured by the labels of the spurious correlation tasks Z_0 and Z_1 and the plots for the feature vectors of spurious correlation task F_{Z_0} and F_{Z_1} coloured by the labels of Z_1 and Z_0 respectively are given in the Figure 4.7. It can be seen that a fuzzy separation can be made for the labels in the different feature spaces illustrated. This reflects the dependencies between the classification tasks.

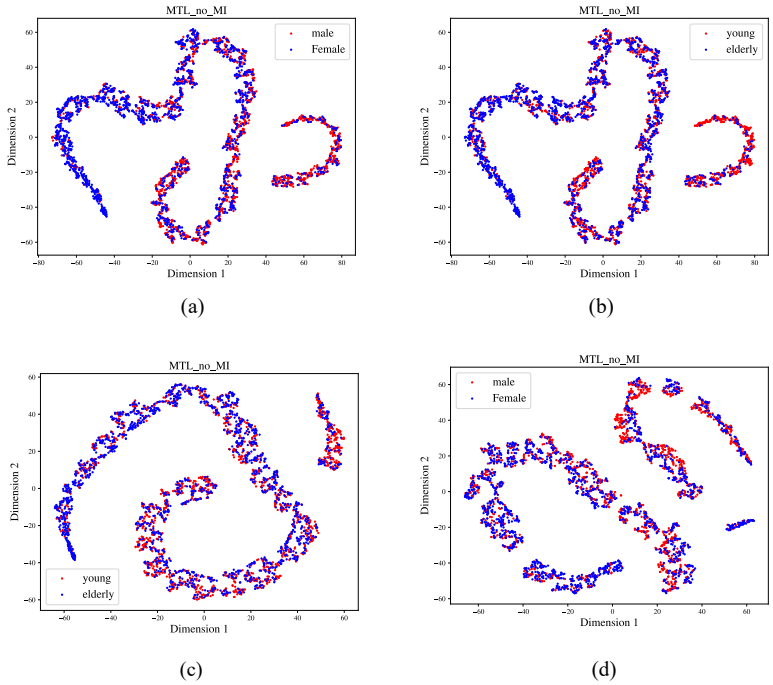


Figure 4.7.: t-SNE Plots for CheXpert-Small Baseline Model. (a) The feature vectors of primary task F_Y coloured by labels of Z_0. (b) The feature vectors of primary task F_Y coloured by labels of Z_1. (c) The feature vectors of spurious correlation task 1 F_{Z_0} coloured by labels of Z_1. (d) The feature vectors of spurious correlation task 2 F_{Z_1} coloured by labels of Z_0.

4.2.2. MIMM Model with Custom Feature Encoder

The MIMM model with a custom encoder underwent training using the hyperparameters outlined in Table 4.16. A higher learning rate of 1×10^{-3} was employed compared to that used for Morpho-MNIST. This adjustment was necessary to achieve satisfactory accuracy with the custom model. Without it, the model training progresses too slowly, and the accuracy plateaus at a relatively low value. However, the introduction of the MI regularization term complicates matters. The elevated learning rate hampers learning when combined with the MI regularization term.

The challenge arises from the fact that the large learning rate corresponds to a large increment of the parameters. Also during the initial training steps, the MI estimates may not accurately guide the model towards the true minima. These two factors might inadvertently steer the model away from the true minima. The curves for the training process can be seen in Figure 4.8, it can be seen that the loss does not decrease and therefore the accuracy never increases from 50% of the random guess.

Hyperparameter	Values
N_{epoch}	300
lr	1×10^{-3}
N_{FE}	1
N_{MI}	5
B	200
img_size	$1 \times 300 \times 300$
λ	1.5

Table 4.16.: CheXpert-Small MIMM Model with Custom Feature Encoder Training Hyperparameters

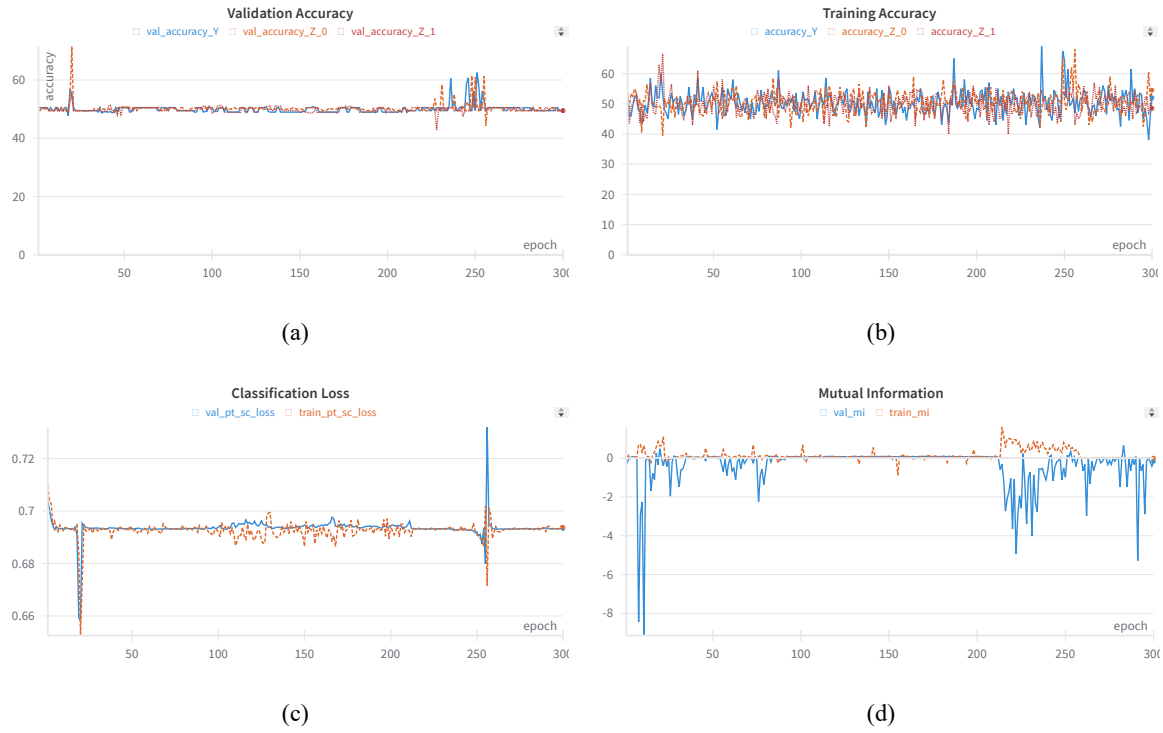


Figure 4.8.: CheXpert-Small MIMM model with Custom Feature Encoder training - (a) Validation accuracy, (b) Training accuracy, (c) Classification loss, (d) Mutual information

Task	Accuracy			
	Val. Dataset	Test Dataset	Bal. Test Dataset	Native-test Dataset
$Y = negative/positive$	49.5%	49.8%	49.9%	37.8%
$Z_0 = male/female$	49.5%	49.8%	49.9%	50.0%
$Z_1 = young/elder$	49.5%	49.8%	49.9%	67.8%

Table 4.17.: Accuracy on different dataset splits - CheXpert-Small MIMM Model with custom feature encoder.

4.2.3. MIMM Model with Densenet-121 Feature Encoder

This MIMM model used for the experiments here employs the Densenet-121 architecture as the feature encoder. The MI estimate is used as a regularisation term and hyperparameters used for the experiment are given in the table 4.18. An additional hyperparameter for image size is also provided to rescale the input images. The images are downscaled due to memory constraints. The experiments are the same as those performed for the Morpho-MNIST dataset and are listed below.

E.1. Training without adaptive scaling and corrected MI gradients.

E.2. Training with both adaptive scaling and corrected MI gradients.

E.3. Training with adaptive scaling and without the corrected MI gradients.

E.4. Training without adaptive scaling and with the corrected MI gradients.

Hyperparameter	Values
N_{epoch}	300
lr	1×10^{-5}
N_{FE}	1
N_{MI}	5
B	150
img_size	$3 \times 96 \times 96$
λ	1.5

Table 4.18.: CheXpert-Small MIMM model with Densenet-121 Feature Encoder training Hyperparameters

Accuracy

The tables below showcase the accuracy obtained for experiments E.1-E.4. The accuracy of the primacy task for the test dataset has improved to a value greater than 65% from the 42.7% of the baseline model. This improvement can also be seen in balanced-test and native-test datasets. The primary task accuracy on the test dataset is greater than 70% when adaptive scaling is used during training, hence illustrating its ability to prevent over-regularisation and thus helping the model to balance between classification loss and MI loss.

The validation accuracy, as expected from the Morpho-MNIST experiments, falls compared to that offered by the baseline model. This is because the model no longer utilizes the spurious correlation from the training dataset for predicting against similar correlations in the validation dataset. However, this accuracy still remains comparable to the accuracy on test dataset.

Task	Accuracy			
	Val. Dataset	**Test Dataset**	**Bal. Test Dataset**	**Native-test Dataset**
$Y = negative/positive$	81.0%	65.9%	75.2%	67.8%
$Z_0 = male/female$	77.2%	78.2%	75.8%	74.3%
$Z_1 = young/elder$	69.7%	67.9%	70.6%	66.4%

Table 4.19.: E.1. Accuracy on Different Datasets - CheXpert-Small MIMM model trained without adaptive scaling and corrected MI gradients.

Task	Accuracy			
	Val. Dataset	**Test Dataset**	**Bal. Test Dataset**	**Native-test Dataset**
$Y = negative/positive$	76.4%	71.9%	72.3%	66.4%
$Z_0 = male/female$	82.0%	76.7%	81.0%	79.3%
$Z_1 = young/elder$	79.5%	69.4%	74.8%	73.6%

Table 4.20.: E.2. Accuracy on Different Datasets - CheXpert-Small MIMM model trained with both adaptive scaling and corrected MI gradients.

Task	Accuracy			
	Val. Dataset	**Test Dataset**	**Bal. Test Dataset**	**Native-test Dataset**
$Y = negative/positive$	72.8%	72.1%	73.3%	65.7%
$Z_0 = male/female$	79.2%	81.3%	81.5%	85.0%
$Z_1 = young/elder$	73.6%	69.9%	74.6%	69.3%

Table 4.21.: E.3. Accuracy on Different Datasets - CheXpert-Small MIMM model trained with adaptive scaling and without corrected MI gradients.

Task	Accuracy			
	Val. Dataset	**Test Dataset**	**Bal. Test Dataset**	**Native-test Dataset**
$Y = negative/positive$	78.9%	68.1%	73.8%	66.4%
$Z_0 = male/female$	79.5%	77.2%	78.1%	82.1%
$Z_1 = young/elder$	69.5%	70.9%	70.3%	75.0%

Table 4.22.: E.4. Accuracy on Different Datasets - CheXpert-Small MIMM model trained without adaptive scaling and with corrected MI gradients.

Switched-Labels Test

The results for the switched test is laid out in the tables 4.23-4.26. The MIMM model reduces the accuracy of predicting the switched labels from $\sim$ 65% in the baseline model to $\sim$ 50%. Since the accuracies for the switched-label tasks are near to that which can be acquired through random chance, it can be said that MIMM model has been successful at learning the tasks without relying on the spurious correlation.

Task	Accuracy
$F_Y \rightarrow Z_0$	55.3%
$F_Y \rightarrow Z_1$	56.2%
$F_{Z_0} \rightarrow Y$	48.7%
$F_{Z_0} \rightarrow Z_1$	49.6%
$F_{Z_1} \rightarrow Y$	48.7%
$F_{Z_1} \rightarrow Z_0$	48.7%

Table 4.23.: E.1. Switched-Labels Test - CheXpert-Small MIMM model trained without adaptive scaling and corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	47.7%
$F_Y \rightarrow Z_1$	49.5%
$F_{Z_0} \rightarrow Y$	50.9%
$F_{Z_0} \rightarrow Z_1$	49.4%
$F_{Z_1} \rightarrow Y$	54.5%
$F_{Z_1} \rightarrow Z_0$	51.5%

Table 4.24.: E.2. Switched-Labels Test - CheXpert-Small MIMM model trained with both adaptive scaling and corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	49.5%
$F_Y \rightarrow Z_1$	49.9%
$F_{Z_0} \rightarrow Y$	50.6%
$F_{Z_0} \rightarrow Z_1$	48.9%
$F_{Z_1} \rightarrow Y$	54.3%
$F_{Z_1} \rightarrow Z_0$	50.6%

Table 4.25.: E.3. Switched-Labels Test - CheXpert-Small MIMM model trained with adaptive scaling and without corrected MI gradients.

Task	Accuracy
$F_Y \rightarrow Z_0$	51.2%
$F_Y \rightarrow Z_1$	53.4%
$F_{Z_0} \rightarrow Y$	50.2%
$F_{Z_0} \rightarrow Z_1$	50.1%
$F_{Z_1} \rightarrow Y$	50.6%
$F_{Z_1} \rightarrow Z_0$	48.8%

Table 4.26.: E.4. Switched-Labels Test - CheXpert-Small MIMM model trained without adaptive scaling and with corrected MI gradients.

t-SNE Plots

The t-SNE plots for experiments E.1-E.4 are given in the figures in the next page. It can be seen that the labels are inseparable in t-SNE plots. The inseparability of Z_0 and Z_1 labels from the plots of $F_Y \rightarrow Z_0$ and $F_Y \rightarrow Z_1$, i.e. the plots (a) and (b), show that the influence of demographics sex and age on the detection of pleural effusion has been removed. It can also be seen that the proposed model was also able to remove the relationship between sex and age, i.e. the spurious correlation variables.

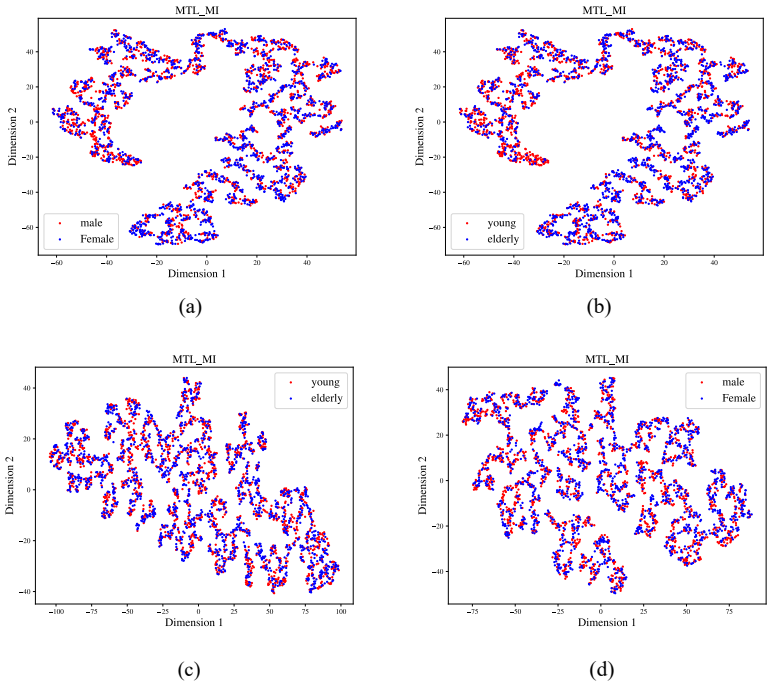


Figure 4.9.: E.1. t-SNE Plots for CheXpert-Small MIMM model without adaptive scaling and corrected MI gradients.

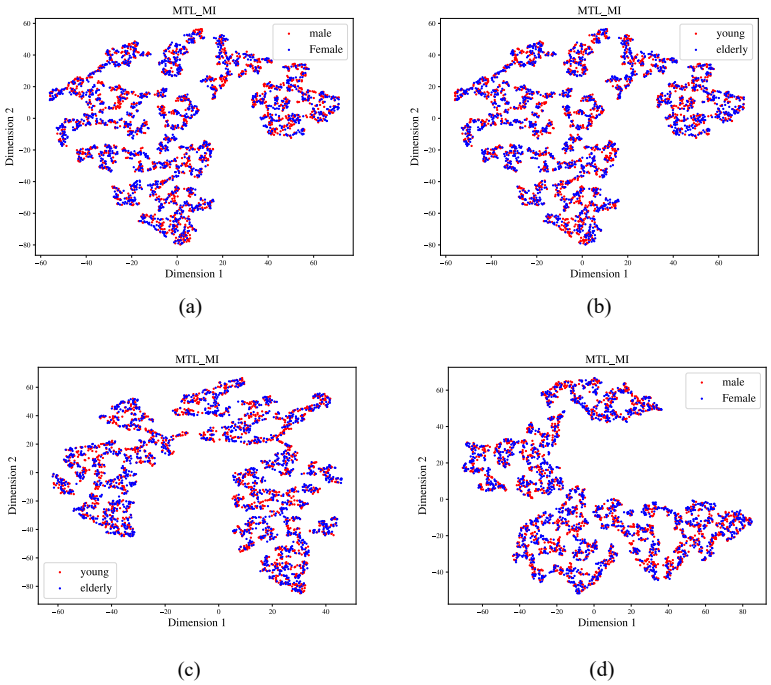


Figure 4.10.: E.2. t-SNE Plots for CheXpert-Small MIMM model with both adaptive scaling and corrected MI gradients.

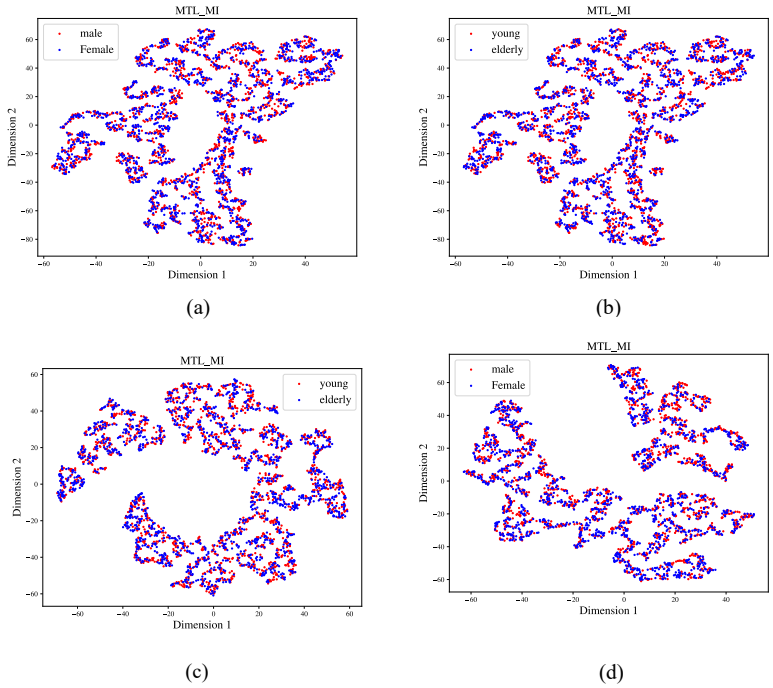


Figure 4.11.: E.3. t-SNE Plots for CheXpert-Small MIMM model with adaptive scaling and without corrected MI gradients.

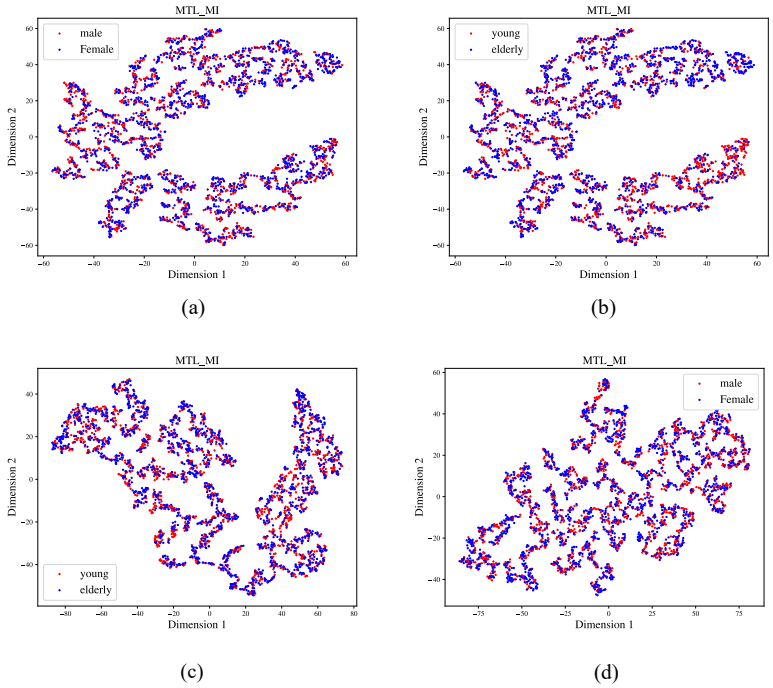
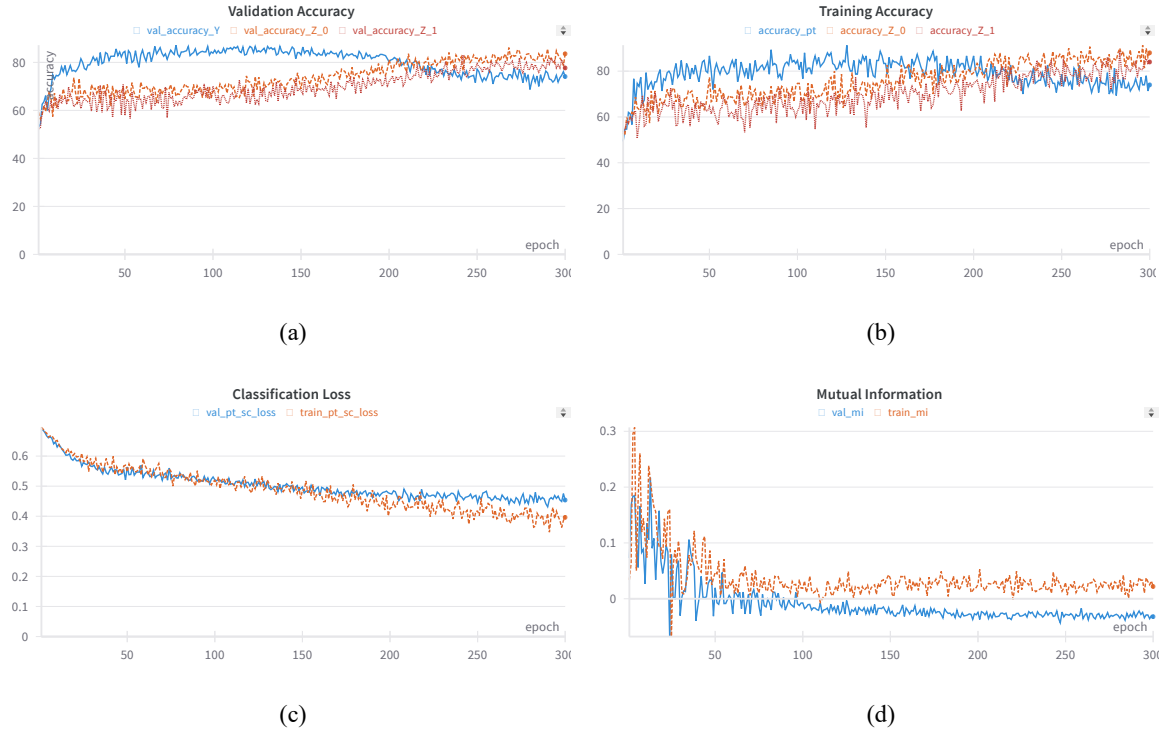


Figure 4.12.: E.4. t-SNE Plots for CheXpert-Small MIMM model without adaptive scaling and with corrected MI gradients.

t-SNE Plots - (a) The feature vectors of primary task F_Y coloured by labels of Z_0. (b) The feature vectors of primary task F_Y coloured by labels of Z_1. (c) The feature vectors of spurious correlation task 1 F_{Z_0} coloured by labels of Z_1. (d) The feature vectors of spurious correlation task 2 F_{Z_1} coloured by labels of Z_0.

Model Training Analysis

The training curves are presented in Figure 4.13. The classification loss curve is monitored and the training was stopped when overfitting was noticed. The MI as in case of Morpho-MNIST dataset spikes first and then is controlled through the regularisation term. It can be observed from the accuracy curves that the spurious correlation tasks, i.e. sex and age, were relatively more difficult to learn compared to the primary task of pleural effusion. Although this was the case the spuriously correlated tasks still had a significant influence on the primary task while training with the baseline model while the MIMM model evaded this issue.



(a) (b)

(c) (d)

Figure 4.13.: CheXpert-Small MIMM model with Densenet-121 Feature Encoder training - (a) Validation accuracy, (b) Training accuracy, (c) Classification loss, (d) Mutual information

5. Conclusion

The research thesis delved into learning causal relationships from data in the presence of multiple spurious correlations. A thorough examination of causal inference and counterfactual invariance was conducted to establish a foundation for understanding the complexities of causal relationships and the limitations of relying solely on correlations for inference. The causal structure of two datasets, Morpho-MNIST and CheXpert-Small, was analyzed and the MIMM, originally proposed by Fay et al., was extended to address the challenge posed by multiple correlations.

The proposed model successfully learned the causal structure of the data while circumventing the spurious correlations inherent within. An exploration of the MINE model was conducted to understand its weaknesses, prompting the implementation of adaptive scaling within the MIMM loss function's MI regularization term. This adjustment proved to be useful in preventing the model from fixating on the regularization term, thus striking a balance between enhancing classification accuracy and mitigating the influence of spurious correlations. Additionally, the effects of utilizing corrected gradients for MI were investigated through experiments. It was observed that the impact of employing this technique was negligible. A plausible explanation, contingent upon how MI was calculated, was also provided.

The proposed methodology although successful has areas of improvement where future work is possible. The implementation of adaptive scaling requires additional gradient calculation which increases the computational cost. Moreover, the current approach necessitates the calculation of MI between all possible combinations of spurious correlation tasks, which is also computationally demanding. Additionally, the model assumes prior knowledge of the spuriously correlated tasks, highlighting the need for further research in the direction of causal discovery algorithms [12].

A. Algorithms

A.1. Algorithm for mutual information neural estimation

Algorithm 1 MINE - Algorithm for mutual information neural estimation [13]

$\theta \leftarrow$ initialize network parameters

repeat

Draw B minibatch samples from the joint distribution: $(y^{(1)}, z^{(1)}), \ldots, (y^{(b)}, z^{(b)}) \sim \mathbb{P}_{YZ}$

Generate $B \times (B-1)$ shuffled samples for the independent distribution $(y', z') \sim \mathbb{P}_Y \otimes \mathbb{P}_Z$.

Evaluate the lower-bound:

$$\mathcal{V}(\theta) \leftarrow \frac{1}{B} \sum_{i=1}^{B} T_\theta(y^{(i)}, z^{(i)}) - \log\left(\frac{1}{B\times(B-1)} \sum_{i=1}^{B\times(B-1)} e^{T_\theta(y', z')}\right)$$

Evaluate the gradients:

$$\widehat{G}(\theta) \leftarrow \widehat{\nabla}_\theta \mathcal{V}(\theta)$$

Update the statistics network parameters:

$$\theta \leftarrow \theta + \widehat{G}(\theta)$$

until convergence

A.2. Steps for creating confounded Morpho-MNIST dataset

Algorithm 2 Steps for creating confounding in the Morpho-MNIST dataset

1: Load the 'Global' Morpho-MNIST dataset
2: Load the 'Global' training dataset for creating training dataset or the test dataset for validation, test and balanced-test.
3: Set the primary task labels as low (0) and high (1):
    Primary task (Y): 0-4 $\rightarrow$ 0 5-9 $\rightarrow$ 1
4: Remove the plain images
5: Set the perturbation (thin/thick) labels:
    Spurious correlation task-1 (Z_0): thin $\rightarrow$ 0 thick $\rightarrow$ 1
6: Create the spurious correlation through sampling according to specified ratios in table 3.1.
7: Shuffle the dataset
8: Create the spurious correlation Z_1 through selective rotation by 90° according to specified ratios in table 3.1:
    spurious correlation task-2 (Z_1): not-rotated $\rightarrow$ 0 rotated $\rightarrow$ 1

A.3. Steps for creating confounded CheXpert-Small dataset

Algorithm 3 Steps for creating confounded CheXpert-Small dataset

1: Load the CheXpert-Small Dataset
2: Split the original training dataset in the ratio 74.7:8.3:17 for training:validation:test dataset.
3: Map the sex of the patients to 0 and 1:
    Spurious correlation task-1 (Z_0): Male $\rightarrow$ 0 Female $\rightarrow$ 1
4: Binarize the age of the patients to 0 and 1:
    Spurious correlation task-2 (Z_1): <= 50 $\rightarrow$ 0 >= 60 $\rightarrow$ 1
5: Remove lateral views.
6: Remove PA views.
7: Remove data instances where pleural effusion is not labelled.
8: Remove data instances where pleural effusion label is uncertain.
9: Calculate the ratio of each combination of labels in the dataset using table 3.1. Step omitted for native-test dataset.
10: Sample from the dataset according to this ratio to create correlation in the dataset.Step omitted for native-test dataset.

A.4. Algorithm for training MIMM model

Algorithm 4 MIMM Training

1: **Input:** Training data loader *trainLoader* and Validation data loader *valLoader*.
2: Initialize ADAM optimizers for classification models and MINE model
3: **for** $epoch$ **in** 1 **to** N_{epochs} **do**:
4: **for** $step$ **in** $num\_steps\_per\_epoch$ **do**
5: **for** $batches$ **in** N_{FE} **do**
6: Freeze the parameters of Mutual Information Models
7: Estimate MI using the current state of MINE models and the classification loss using the current state if feature encoder.
8: Train Feature Encoder and Primary Task Models by minimizing $\mathcal{L}_{MIMM}$ and backpropagating through both of them.
9: **end for**
10: **for** $batches$ **in** N_{MI} **do**
11: Freeze the parameters of Feature Encoder and Primary Task Models
12: Train Statistics network by maximizing the MI output of MINE model and backpropagating through the MINE models.
13: **end for**
14: **end for**
15: Evaluate models on validation set
16: **end for**
17: Save the Models for testing and t-SNE plots.

B. Models

B.1. Morpho-MNIST Custom Feature Encoder

Table B.1.: Model Summary - Morpho-MNIST Custom Model

Layer	Output Shape	Param #
FeatureEncoderNetwork	[1, 6]	–
Sequential	[1, 16, 4, 4]	–
Conv2d	[1, 6, 26, 26]	60
ReLU	[1, 6, 26, 26]	–
MaxPool2d	[1, 6, 13, 13]	–
Conv2d	[1, 16, 11, 11]	880
ReLU	[1, 16, 11, 11]	–
Conv2d	[1, 16, 9, 9]	2,320
ReLU	[1, 16, 9, 9]	–
MaxPool2d	[1, 16, 4, 4]	–
Linear	[1, 256]	65,792
Linear	[1, 6]	1,542

Table B.2.: Model Details

Total params	70,594
Trainable params	70,594
Non-trainable params	0
Total mult-adds (M)	0.40

B.2. Densenet-121 feature encoder

Table B.3.: Model Summary - Densenet-121

Layer	Output Shape	Param #
chXception	[1, 6]	–
DenseNet	[1, 6]	–
Sequential	[1, 1024, 3, 3]	–
Conv2d	[1, 64, 48, 48]	9,408
BatchNorm2d	[1, 64, 48, 48]	128
ReLU	[1, 64, 48, 48]	–
MaxPool2d	[1, 64, 24, 24]	–
DenseBlock	[1, 256, 24, 24]	335,040
Transition	[1, 128, 12, 12]	33,280
DenseBlock	[1, 512, 12, 12]	919,680
Transition	[1, 256, 6, 6]	132,096
DenseBlock	[1, 1024, 6, 6]	2,837,760
Transition	[1, 512, 3, 3]	526,336
DenseBlock	[1, 1024, 3, 3]	2,158,080
BatchNorm2d	[1, 1024, 3, 3]	2,048
Linear	[1, 6]	6,150

Table B.4.: Model Details - Densenet-121

Total params	6,960,006
Trainable params	6,960,006
Non-trainable params	0
Total mult-adds (M)	520.46

B.3. CheXpert Custom Feature Encoder

Table B.6.: Model Details - CheXpert Custom Model

Total params	135,760
Trainable params	135,760
Non-trainable params	0
Total mult-adds (M)	188.60

Table B.5.: Model Summary - CheXpert Custom Model

Layer	Output Shape	Param #
MedicalFeatureEncoder	[1, 6]	–
Sequential	[1, 128, 1, 1]	–
Conv2d	[1, 3, 298, 298]	30
ReLU	[1, 3, 298, 298]	–
Conv2d	[1, 6, 296, 296]	168
ReLU	[1, 6, 296, 296]	–
Conv2d	[1, 8, 294, 294]	440
ReLU	[1, 8, 294, 294]	–
Conv2d	[1, 8, 292, 292]	584
ReLU	[1, 8, 292, 292]	–
MaxPool2d	[1, 8, 97, 97]	–
BatchNorm2d	[1, 8, 97, 97]	16
Conv2d	[1, 16, 95, 95]	1,168
ReLU	[1, 16, 95, 95]	–
Conv2d	[1, 16, 93, 93]	2,320
ReLU	[1, 16, 93, 93]	–
Conv2d	[1, 16, 91, 91]	2,320
ReLU	[1, 16, 91, 91]	–
Conv2d	[1, 16, 89, 89]	2,320
ReLU	[1, 16, 89, 89]	–
MaxPool2d	[1, 16, 29, 29]	–
BatchNorm2d	[1, 16, 29, 29]	32
Conv2d	[1, 32, 27, 27]	4,640
ReLU	[1, 32, 27, 27]	–
Conv2d	[1, 32, 25, 25]	9,248
ReLU	[1, 32, 25, 25]	–
Conv2d	[1, 32, 23, 23]	9,248
ReLU	[1, 32, 23, 23]	–
MaxPool2d	[1, 32, 7, 7]	–
BatchNorm2d	[1, 32, 7, 7]	64
Conv2d	[1, 64, 5, 5]	18,496
ReLU	[1, 64, 5, 5]	–
Conv2d	[1, 128, 3, 3]	73,856
ReLU	[1, 128, 3, 3]	–
AvgPool2d	[1, 128, 1, 1]	–
Sequential	[1, 6]	–
Linear	[1, 64]	8,256
ReLU	[1, 64]	–
Linear	[1, 32]	2,080
ReLU	[1, 32]	–
Linear	[1, 12]	396
ReLU	[1, 12]	–
Linear	[1, 6]	78

List of Figures

List of Tables

Bibliography

[1] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 12 2017.

[2] M. Tsuneki, "Deep learning models in medical image analysis," *Journal of Oral Biosciences*, vol. 64, 03 2022.

[3] V. Vapnik, "An overview of statistical learning theory," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 988–999, 1999.

[4] A. J. Larrazabal, N. Nieto, V. Peterson, D. H. Milone and E. Ferrante, "Gender imbalance in medical imaging datasets produces biased classifiers for computer-aided diagnosis," *Proceedings of the National Academy of Sciences*, vol. 117, p. 12592–12594, 06 2020. [Online]. Available: https://www.pnas.org/content/117/23/12592

[5] P. Sanchez, J. P. Voisey, T. Xia, H. I. Watson, A. Q. O'Neil and S. A. Tsaftaris, "Causal machine learning for healthcare and precision medicine," *Royal Society Open Science*, vol. 9, 08 2022.

[6] N. Díaz-Rodríguez, R. Binkytė, W. Bakkali, S. Bookseller, P. Tubaro, A. Bacevičius, S. Zhioua and R. Chatila, "Gender and sex bias in covid-19 epidemiological data through the lens of causality," *HAL (Le Centre pour la Communication Scientifique Directe)*, vol. 60, pp. 103 276–103 276, 05 2023.

[7] A. Lynch, G. J.-S. Dovonon, J. Kaddour and R. Silva, "Spawrious: A benchmark for fine control of spurious correlation biases," 2023.

[8] R. Geirhos, J. Jacobsen, C. Michaelis, R. S. Zemel, W. Brendel, M. Bethge and F. A. Wichmann, "Shortcut learning in deep neural networks," *CoRR*, vol. abs/2004.07780, 2020. [Online]. Available: https://arxiv.org/abs/2004.07780

[9] S. Beery, G. V. Horn and P. Perona, "Recognition in terra incognita," *CoRR*, vol. abs/1807.04975, 2018. [Online]. Available: http://arxiv.org/abs/1807.04975

[10] S. Sagawa, P. W. Koh, T. B. Hashimoto and P. Liang, "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization," *CoRR*, vol. abs/1911.08731, 2019. [Online]. Available: http://arxiv.org/abs/1911.08731

[11] V. Veitch, A. D'Amour, S. Yadlowsky and J. Eisenstein, "Counterfactual invariance to spurious correlations: Why and how to pass stress tests," *CoRR*, vol. abs/2106.00545, 2021. [Online]. Available: https://arxiv.org/abs/2106.00545

[12] L. Fay, E. Cobos, B. Yang, S. Gatidis and T. Küstner, "Avoiding shortcut-learning by mutual information minimization in deep learning-based image processing," *IEEE Access*, vol. 11, pp. 64 070–64 086, 2023.

[13] I. Belghazi, S. Rajeswar, A. Baratin, R. D. Hjelm and A. C. Courville, "MINE: mutual information neural estimation," *CoRR*, vol. abs/1801.04062, 2018. [Online]. Available: http://arxiv.org/abs/1801.04062

[14] J. Peters, D. Janzing and B. Scholkopf, *Elements of causal inference : foundations and learning algorithms*. Mass, 2017.

[15] A. Ward, ""spurious correlations and causal inferences"," *Erkenntnis*, vol. 78, pp. 699–712, 11 2012.

[16] J. Pearl, *Causality*, 2nd ed. Cambridge, UK: Cambridge University Press, 2009.

[17] Y. Ming, H. Yin and Y. Li, "On the impact of spurious correlation for out-of-distribution detection," *CoRR*, vol. abs/2109.05642, 2021. [Online]. Available: https://arxiv.org/abs/2109.05642

[18] W. Li, "Mutual information functions versus correlation functions," *Journal of Statistical Physics*, vol. 60, no. 5, pp. 823–837, Sep 1990. [Online]. Available: https://doi.org/10.1007/BF01025996

[19] "Explanation of Mutual Information Neural Estimation — ruihongqiu.github.io," https://ruihongqiu.github.io/posts/2020/07/mine/, [Accessed 30-03-2024].

[20] D. C. Castro, J. Tan, B. Kainz, E. Konukoglu and B. Glocker, "Morpho-MNIST: Quantitative assessment and diagnostics for representation learning," *Journal of Machine Learning Research*, vol. 20, no. 178, 2019.

[21] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. L. Ball, K. S. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren and A. Y. Ng, "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," *CoRR*, vol. abs/1901.07031, 2019. [Online]. Available: http://arxiv.org/abs/1901.07031

[22] S. Ihler, F. Kuhnke and S. Spindeldreier, "A comprehensive study of modern architectures and regularization approaches on chexpert5000," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, L. Wang, Q. Dou, P. T. Fletcher, S. Speidel and S. Li, Eds. Cham: Springer Nature Switzerland, 2022, pp. 654–663.

Declaration

Herewith, I declare that I have developed and written the enclosed thesis entirely by myself and that I have not used sources or means except those declared.

This thesis has not been submitted to any other authority to achieve an academic grading and has not been published elsewhere.

Stuttgart, 16/04/2024 Gautham Mohan