CLASS-CONDITIONAL AND CLASS-GENERIC DISENTANGLED REPRESENTATIONS

REPORT

Sairam Satwik Kondamudi
Department of Computer Science
IIT-H
satwik@iith.ac.in

Vineeth N Balasubramanian
Department of Computer Science
IIT-H
vineethnb@iith.ac.in

December 24, 2019

ABSTRACT

We present a framework that can learn both class-conditional and some class generic factors of variation given a dataset. We do this by making some changes to the evidence lower bound and by introducing a new objective to learn these representations which includes a max-min objective to channel the class information from one set of variables into another set. We prove our point by showing few qualitative results that we have obtained on benchmark datasets. We also provide few quantitative results to validate the ability of our framework to (a)learn a representation capable of performing well in transfer learning(b) channel the class-information from one set of variables into another.

1 Introduction

Learning a low dimensional and interpretable representation of data has been the long standing aim of the machine learning community. It is strongly believed that such representations can have serious advantages in several deep learning tasks like transfer learning and zero-shot learning(Lake eta al [1]). There have been works by Hinton et al. [2] which is one of the earliest efforts in successfully learning a low-dimensional latent representation of data using a neural network, but the representations learned by the conventional autoencoder architecture lacked a proper structure and are not amenable to human understanding. By structure we mean that- by introducing minor perturbations in a latent representation, we want to qualify exactly what are the specific changes that these perturbations have caused by reconstructing the perturbed representation. There have been multiple efforts in this space where we are interested in learning factors of variation of data called disentangle representations. There's no formal definition available for disentangled representations but the research community follows the definition proposed by Benjio et al. [3]: a representation where a change in one dimension corresponds to a change in one factor of variation, while being relatively invariant to changes in other factors.

With the success of the deep generative models like VAE by Kingma et al. [4] there have been a flurry of works which got proposed in this space by making changes to the VAE's objective which is optimizing a variational lower bound in every training step. The most recent works on representation learning mostly(Higgins et al. [5],Hsu et al. [6], Kim et al. [7], Chen et al [8],Esmaeili et al. [9], Dupont et al. [10]) focus on modeling disentangled latent representation in an unsupervised setting.

While all these methods focus on learning the factors that are common to all the classes in the dataset, in this paper we propose a model which is capable of learning both class-conditional and class generic factors by using a modified VAE-objective. We show our results by depicting various factors for each class that our model has learnt and also give a few results of our experiments to validate the capability of our objective function to separate out the class specific information from the class-generic information.

2 Related Work

As mentioned earlier, Higgins et al [5] proposed to learn disentangled representations by putting an additional weight term(a hyper parameter) on the KL divergence term in VAE's objective. Kim et al [7]) and Chen et al [8] learn a factorial latent representation by minimizing TC(Total Correlation) of the latent dimensions. These works learn latent representations in which each dimension in the representation is responsible for one particular factor of variation in the data. And, Works by Esmaeili et al. [9] and Dupont et al. [10] learn both discrete and continuous factors in an unsupervised setting by utilizing the gumbel-softmax trick to help with the back-propagation. The difference being Esmaeili et al. [9] considered a hierarchical latent-space model and Dupont et al. [10] considered a parallel latent space model. All these works learn a generalized set of factors which are common to all the classes. With these frameworks, to attain class-specific factors we would require to train separate models for every class which is quite cumbersome to handle. Esmaeili et al.'s [9] work will behave like a class-conditional model but only for simple datasets like MNIST [11] because of learning discrete factors in it's hierarchical framework.

Works like Klyss et al's [12] proposed learning of latent sub-spaces using full supervision from the labels in the form of presence/absence of specific attributes like spectacles, facial hair in a data sample(whose subspace is meant to be learnt) but we only consider a high-level supervision in the form of class-label. There were also some prior works(Creswell et al [13], Kingma et al [14], Sohn et al [15]) which took a different definition for learning disentangled representations wherein, they model in a latent space with two sets of variables- one set exclusively for learning the class information(generally discrete) and the other set for learning the non-class information(generally called style) of the data. Of these works, [13] especially learns the class corresponding representation using a min-max objective to make sure that the class information mostly resides in one variable. Most of the above works focus on learning generalized factors of variation for all the datapoints in the dataset. But the representations learnt by these works don't comply with the definition coined by Benjio et al [3], hence these are not exactly solving the problem we are interested in.

A generalized representation is not going to give a complete understanding of a dataset. Modeling latent space based on class-dependent factors and class-independent factors of variation is another space which is yet to be explored. We achieve this objective by minimizing the dependence of one set of variables on the class-information using a classifier along with the condition which forces the representation to be disentangled in one-step and correcting the classifier by making it predict the correct class label in another step. We would like to add that using a classifier in an adversarial manner like this is not done for the first time([13],[12]), but we differ from other works in terms of various reasons like the framework, the level of supervision we're using to achieve and the nature of the random variable we've defined to solve our objective.

3 Methodology

We assume that we have a dataset $\mathbb{D}$ of samples (x, y) with n classes where x is our observation and y is it's one-hot encoded class label belonging to $\{y \in \{0,1\}^n | \sum_i (y_i) = 1\}$. We consider a VAE kind of setup for our experiments and problem solving. Since we are interested in learning disentangled representations which take the class information into consideration, the obvious choice is to explicitly condition the encoder with the class label for learning representations with cues from the class label. Using this trick we describe 2 possible models to achieve our objective.

3.1 Naive Model

We consider the set of latent variables to be denoted by $\mathbf{z}$ and we define a posterior $q_\phi(z|x, y)$, a prior $p(z|y)$ and likelihood $p_\theta(x|z, y)$ This model's framework is similar to a fully supervised conditional VAE setup, where we try to optimize the variational lower bound

$$p(x|y) \geq E_{q_\theta(z|x,y)}[p(x|z, y)] - KL(q_\theta(z|x, y)||p(z)) \tag{1}$$

But in our case since we want to learn disentangled representations, like in $\beta-$VAE([5]), we have an additional weight on the KL-divergence term to ensure disentanglement of the variables. The problem with this objective is quite clear, similar to that of $\beta-$VAE's the additional weight that we put on KL-divergence term is adding on to minimizing the mutual information between z and the joint variable (x, y) w.r.t the variational joint distribution as well. Hence, even though the z gets disentangled, the latent code's knowledge about the data is minimized in every step. This can be

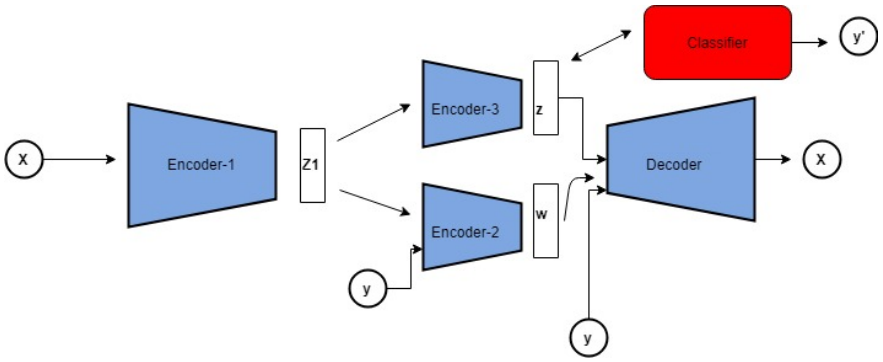

Figure 1: Proposed model framework. The blocks in blue are trained in one step and the block in red is trained in another step

easily shown following the Makhzani et al.'s([16]) and Kim et al's proof([7]).

$$
\begin{aligned}
E_{(x,y)\sim D(x,y)}[KL(q_\theta(z|x,y)||p(z))] &= E_{(x,y)\sim D(x,y)}[E_{q_\theta(z|x,y)}[log[\frac{q_\theta(z|x,y)}{p(z)}]]] \\
&= E_{(x,y)\sim D(x,y)}[E_{q_\theta(z|x,y)}[log[\frac{q_\theta(z|x,y)}{p(z)}\frac{q(z)}{q(z)}]]] \\
&= E_{(x,y)\sim D(x,y)}[E_{q_\theta(z|x,y)}[log[\frac{q_\theta(z|x,y)}{q(z)}\frac{q(z)}{p(z)}]]] \\
&= E_{(x,y)\sim D(x,y)}[E_{q_\theta(z|x,y)}[log[\frac{q_\theta(z|x,y)}{q(z)}] + log[\frac{q(z)}{p(z)}]]] \\
&= I(z;(x,y)) + KL(q(z)||p(z))
\end{aligned}
$$

We generally take standard gaussian as p(z), so the representation q(z) gets disentangled because of the weight term. To restrict the problem aroused by I(z;(x,y)), we can put a cap on the KL term in this objective with a gradually increasing positive term C_z and modify our objective similar to that of burgess et al's.([17]).
Using an approach like this simply narrows down our intent for learning meaningful representations. We expect to learn factors both which are and which are not affected by class-information. To solve this problem, we propose the following model, the details of which are explained in the next subsection.

3.2 Proposed Model

We propose a model which is a modification to the previous model's framework which allows us to model both class-dependent and class-independent factors. Let **z**,**w** denote the set of class-dependent and class-independent variables respectively. We define a joint posterior $q_\phi(z,w|x,y)$, a prior $p(z,w|y)$ and likelihood $p_\theta(x|z,w,y)$. With this premise, the conventional β-VAE kind of objective is,

$$L(\theta,\phi) = E_{q_\phi(z,w|x,y)}[p_\theta(x|z,w,y)] - \beta.KL(q_\phi(z,w|x,y)||p(z,w|y)) \tag{2}$$

Since we have assumed that $z \perp y$ and $z \perp w$, we rewrite this objective as

$$L(\theta,\phi) = E_{q_\phi(z,w|x,y)}[p_\theta(x|z,w,y)] - \beta.KL(q_\phi(z|x)||p(z|y)) - \gamma.KL(q_\phi(w|x,y)||p(w|y)) \tag{3}$$

Taking inspiration from burgess et al.([17]), we put a cap on the KL-divergence terms so that there is not much information about the data lost. With this our modified objective is

$$L_{VAE} = E_{q_\phi(z,w|x,y)}[p_\theta(x|z,w,y)] - \beta|KL(q_\phi(z|x)||p(z|y)) - C_z| - \gamma|KL(q_\phi(w|x,y)||p(w|y)) - C_w|$$

3.3 Minimizing class dependence

We have assumed that $z \perp y$ and we wanted most information related to y to dwell in w, so we explicitly minimize the mutual information between z and y. We do this by making use of a classifier network. If $z \perp y$, the classifier misclassifies a given z. For this purpose, we define an augmented loss to be maximized by the network using a binary-cross-entropy.

$$L_{class}(z) = y.log(\sigma(h_\psi(z))) + (1-y).log(1-\sigma(h_\psi(z))) \tag{5}$$

Since the classifier must also be capable of predicting the correct class label given z, the classifier's parameters are also to be trained to achieve the below defined objective-

$$\min_{\psi} \quad L_{class} \tag{6a}$$

So, our overall objective is to solve the below multi-step optimization problem where we train the parameters of encoder and decoder by solving the maximization problem in the first-step and train the parameters of the classifier in the next step while solving the minimization problem-

$$\max_{\theta,\phi} \quad L(\theta,\phi) + L_{class}$$
$$\min_{\psi} \quad L_{class}$$

We noticed that using w instead of z in the second step of our objective turned out to be very helpful. So, we used w in step 2 of all our experiments.

4 Experiments and Results

We have used Gaussian MLPs with diagonal covariance for our experiments and chose $p(w|y)$ and $p(z)$ as standard gaussians for our implementation. We have run our experiments using carefully chosen set of hyperparameters. We noticed that, when we set higher value for C_z, the model tends to ignore z and encode most part of the information in w which is undesirable. So we set the maximum value of C_z decisively such that it's value is not much higher than the maximum value of C_w for obvious reasons- Higher value of C_z gives enough opportunity for the model to lose most information about the data. For our experiments, we chose the dimensionality of both z and w to be 10. We couldn't compare with any of the existing works learning disentangled representations is because none of these works do specifically learn class-conditional disentangled representation. For these methods to learn factors for every class, separate models have to be trained for each class which itself is a major drawback and hence is ruled out. For this purpose, to validate our work, we defined our own experiments and the details of them are presented in the coming sectons.

4.1 Implementation Details

We conducted experiments using PYTORCH framework([18]) on the publicly available **celebA**([19]), **MNIST**([11]) and **Fashion MNIST**([20]) datasets. In celebA, we have chosen two classes namely male and female for our experiments on **celebA**([19]) and the regular class labels as class labels in our experiments on MNIST([11]) and FashionMNIST([20]). In line with our intent, our model was able to learn separate sets of disentangled representations for each class that we have defined and one common latent factor for each of MNIST and Fashion MNIST datasets. To verify the factors that the model has learnt, we have taken a set of test images(untrained images) and visualized the changes in each of z and $w|y$ by reconstructing after every perturbation in the range $[-5, 5]$ across every dimension individually. By doing so, we find the factors of variation and the dimensions in the latent that are responsible for those factor across various input images for a given label.

While traversing between -5 and 5 of z sampled from the posterior of MNIST and FMNIST images, our model was

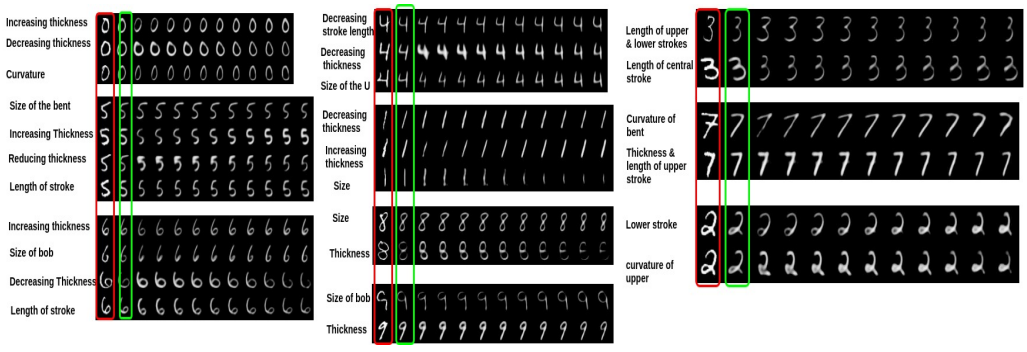

Figure 2: Reconstructions for positive traversal from -5 to 5 from left to right in $w|y$ for the original image in the red-box of the actual reconstruction from the green-box

able to learn the factors "thickness" and "lighting" in MNIST and Fashion MNIST respectively which are common to

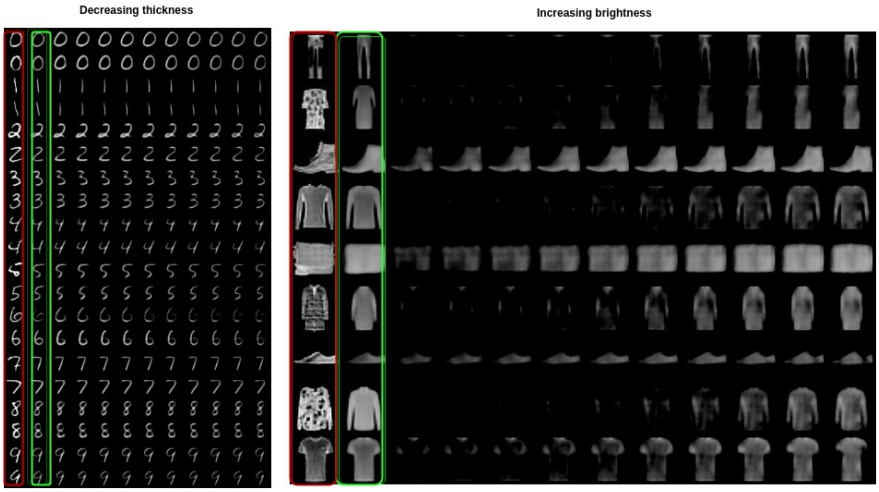

Figure 3: Reconstructions for positive traversal from -5 to 5 from left to right in z for the original image in the red-box of the actual reconstruction from the green-box

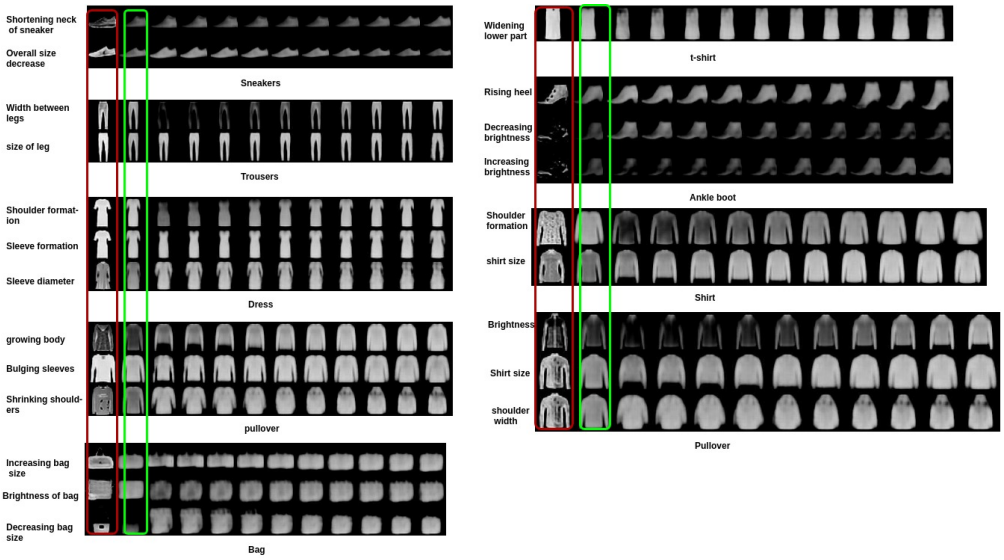

Figure 4: Reconstructions for positive traversal from -5 to 5 from left to right in $w|y$ for the original image in the red-box of the actual reconstruction from the green-box

all of their respective classes. This can be observed in figure 3. From figure 2 we can see the various factors that were disentangled for each of the digits in MNIST. It is also to be noticed that most of these factors were not particularly disentangled by any of the existing methods(Kim et al [7], Chen et al [8], Hsu et al [6],Esmaeili et al [9]). The factors disentangled for each of the classes in $w|y$ in our experiments on Fashion mnist can be found in figure 4 and figure 5 shows the results of our experiments on celebA. Like most methods in learning disentangled representations, our method too suffers from the trade-off between reconstruction quality and disentangling extent. So, we couldn't notice anything in particular in the sandal class because of the relatively poor reconstruction quality. We didn't find any factor being disentangled in z in our experiments on celebA dataset but we did find some interesting factors in male and female classes in our experiments.

Figure 5: Reconstructions for positive traversal from -5 to 5 from left to right in $w|y$ the actual reconstruction from the green-box

4.2 Quantitative Evaluation

4.2.1 Checking the separation of class information between z and w

To check the extent to which our objective has helped in channeling the class-information from z to w, we have trained separate classifiers(with same capacity) on samples from z and $w|y$ respectively to predict their respective class labels with and without using L_{class} on the datasets MNIST,Fashion-MNIST and CelebA. The accuracies we are mentioning here are the average accuracies of 3 runs. We took this average to avoid misjudgment caused due to randomization. It can be clearly observed that in all the 3 datasets that we have tested upon, the case when using L_{class} is helping w in achieving higher accuracy in predicting the true label. This gives enough validation to our intent for the class-specific information to reside in w and non-class specific information to be in z. For all our experiments, we chose equal values

Dataset	Method	Accuracy on z	Accuracy on w
Fmnist	without L_{class}	0.337	0.47
	with L_{class}	0.32	**0.68**
Mnist	without L_{class}	0.37	0.31
	with L_{class}	0.30	**0.45**
celebA	without L_{class}	0.63	0.68
	with L_{class}	0.685	**0.745**

Table 1: Average Classification accuracies for class prediction on w and z

for γ and β.

4.2.2 Checking the ability of γ, β, C_z and C_W to help learn good representation

To check the extent to which γ and β are influencing the disentangling, we set $\gamma = \beta = 1$, set $C_z = C_w = 0$ and remove the absolute value function from the objective 4 i.e, we train our model without any condition forcing disentanglement. We tried to visualize the reconstructions while traversing in z. From figure 7 we can notice the reconstructions while traversing from -5 to 5 of particular dimension of z where each row corresponds to the reconstructions of the traversals on that particular digit. We can see that for different digits there are different factors being changed which is not

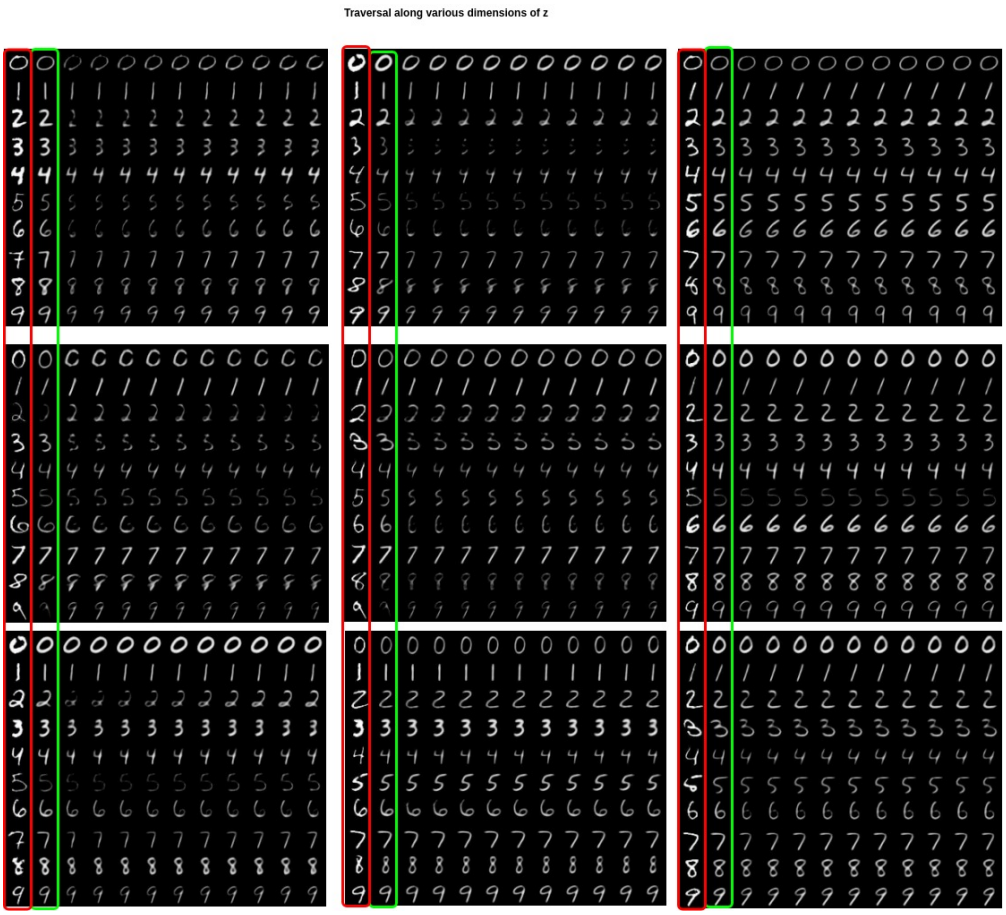

Figure 6: Reconstructions for positive traversal from -5 to 5 from left to right in z the actual reconstruction from the green-box

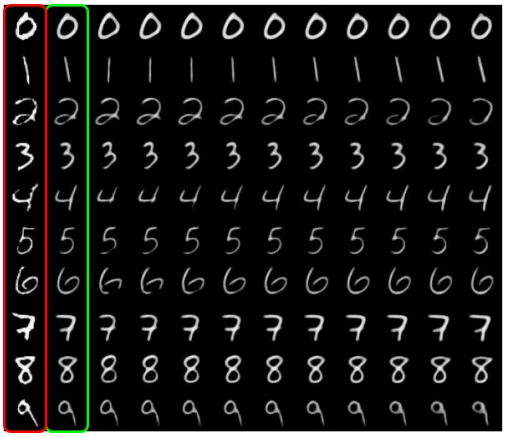

Figure 7: Reconstructions for positive traversal from -5 to 5 from left to right in z the actual reconstruction from the green-box when $\gamma = 1$

expected from z(we want z to learn class specific factors). But that is not the case of our proposed model when using a higher γ and β i.e, $\gamma = \beta = 100.0$. and $C_z(max) = 15.0$ and $C_w(max) = 25.0$. This can be observed from figures 6 and 3, where each sub-image in figure 6 corresponds to the traversal of all the 10 digits for 9 dimensions in which we didn't notice any disentanglement but we noticed a constant factor that is being changed for all input images in figure 3 which corresponds to the dimension in the latent representation which successfully learnt a factor of variation.

We have also conducted a simple experiment where we transfer the learned $w|y$ to a classification network to predict the class label. This can be treated as a test for the ability of the latent representation to help in transfer learning. The experimental setup used here is same as that in section 4.2.1. But in here, our intent is to check how good the representation is. Higher the performance of the downstream task using the latent, better is the latent's representation. From Lake et al [1] a disentangled representation can boost the performance of the downstream tasks.

Dataset	γ	β	**Accuracy**
F-MNIST	1	1	0.58
	100	100	**0.68**
MNIST	1	1	0.44
	100	100	**0.45**

Table 2: Average Classification accuracies for transfer learning when predicting class label

Here $\gamma = \beta = 1$ is the case where we don't force the network to disentangle and thus set $C_z = C_w = 0$ and remove the absolute value function from the objective 4. These results confirm that our model was indeed learning good representations.

5 Conclusions & Future Work

In this paper, we have introduced a novel framework to learn both class conditional and class-generic disentangled representations using a cross-entropy loss function to separate these factors based on class-information. Through our experiments on MNIST, Fashion-MNIST and CelebA datasets, we shown qualitatively the range of factors that our model can learn. We have also provided quantitative results to prove the capability of our model to separate the factors based on class-information.
Some of the main problems in our network are that it is prone to mode collapse and the poor reconstruction quality because of the disentanglement vs reconstruction trade-off. For addressing this trade-off by using Jacobian-supervision like in Lezama et al's [21] work. For our future work, we would like to pursue in this direction for solving the above two problems.

References

[1] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

[2] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[3] Yoshua Bengio, Aaron C Courville, and Pascal Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR, abs/1206.5538*, 1:2012, 2012.

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

[5] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

[6] Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Advances in neural information processing systems*, pages 1878–1889, 2017.

[7] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. *arXiv preprint arXiv:1802.05983*, 2018.

[8] Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in vaes. NIPS, 2018.

[9] Babak Esmaeili, Hao Wu, Sarthak Jain, Alican Bozkurt, N Siddharth, Brooks Paige, Dana H Brooks, Jennifer Dy, and Jan-Willem van de Meent. Structured disentangled representations. *stat*, 1050:12, 2018.

[10] Emilien Dupont. Joint-vae: Learning disentangled joint continuous and discrete representations. *arXiv preprint arXiv:1804.00104*, 2018.

[11] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[12] Jack Klys, Jake Snell, and Richard Zemel. Learning latent subspaces in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 6445–6455, 2018.

[13] Antonia Creswell, Yumnah Mohamied, Biswa Sengupta, and Anil A Bharath. Adversarial information factorization. *arXiv preprint arXiv:1711.05175*, 2017.

[14] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589, 2014.

[15] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015.

[16] Alireza Makhzani and Brendan J Frey. Pixelgan autoencoders. In *Advances in Neural Information Processing Systems*, pages 1975–1985, 2017.

[17] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in backslashbeta-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[18] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[19] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[20] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

[21] José Lezama. Overcoming the disentanglement vs reconstruction trade-off via jacobian supervision. In *International Conference on Learning Representations*, 2019.