Conditional Variational Autoencoders with Fuzzy Inference

Yury Gurov[1][0000−0002−7033−9996] and Danil Khilkov[1][0000−0001−9284−6924]

NIIAS Institute of Informatization, Automation and Communication in Railway Transport, Russia, Moscow 109029 Nizhegorodskaya str., 27 bldg. 1 info@vniias.ru
www.vniias.ru/

Abstract. We present an approach to constructing Conditional Variational Autoencoders (C-VAE) models with fuzzy inference during classification. This preserves disentangling capabilities of VAE and at the same time performs latent space clusterization. Fuzzy C-VAE model provides useful features for anomaly detection, utilizing partially labeled datasets and controlled generation of new samples.

Keywords: fuzzy logic · deep learning · fuzzy inference · Conditional Variational Autoencoders · fuzzy cvae · neuro-fuzzy

1 Introduction

Hybrid neuro-fuzzy systems has a long time history and is still an active research area [3]. Main feature that attract attemtion to neuro-fuzzy systems is possibility to combine the power of neural network with advantages of fuzzy logic, such a human-like reasoning. At this work we propose an approach to constructing Conditional Variational Autoencoders (C-VAE) [7, 6, 10] models with fuzzy inference during classification phase. This approach may preserve disentangling capabilities of VAE and at the same time performs latent space clusterization. Such fuzzy C-VAE model provides useful features for anomaly detection, utilization of partially labeled datasets and controlled generation of new samples.

The source code is available at GitHub repository[1].

2 Related work

In [1] attempt to apply fuzzy logic to the latent space of VAE was made with fuzzy c-mean clustering. Main drawback of of this approach is that it requires a priori knowledge about the number of clusters and their interpretation

In [9] conditional VAE was modified in a way to process partially observed datasets. Authors proposed method that augments the conditional VAEs with a prior distribution for the missing covariates and estimates their posterior using amortised variational inference. At first sigh this approach has nothing to do with fuzzy logic, but it provides insigh into the problem of latent space clustering.

[1] https://github.com/kenoma/pytorch-fuzzy

3 Methods

3.1 Variational Autoencoders

Variational inference is used to approximate a posterior distribution of a directed graphical model whose latent variables and parameters are intractable. The Variational Auto-Encoder (VAE) combines this approach with an autoencoder framework to learn the prior distribution of a latent space, $p_\theta(z)$, with parameters θ. The idea is that the prior distribution can then be sampled to produce a latent code, z, which is passed as input to the decoder to produce a sample output, $\tilde{x}$. VAEs consists of two NN for the probabilistic encoding and decoding process (see Figure 1a). As the true underlying distribution of the posterior is intractable and complex, a simple parametric surrogate distribution, $q_\Phi(z|x)$ (such as a Gaussian), with parameters Φ, is assumed to approximate the distribution and is optimized for best fit. The encoder network implicitly models the surrogate distribution, by mapping the distribution parameters, Φ, during the training process. The resulting model, $q_\Phi(z|x)$, is referred to as the recognition model. The optimization process of the recognition model revolves around minimizing the Kullback-Leibler (KL) divergence between the posterior and surrogate distributions. Once the latent prior distribution is learned, z can be sampled via the reparameterization trick. The (probabilistic) decoder network performs a mapping of the latent code to a structured sample output for each sample, thus producing a distribution of outputs, $p_\theta(x|z)$.

3.2 Conditional VAE

The C-VAE expands upon the framework of the VAE, by combining variational inference with a conditional directed graphical model. In the case of C-VAE, the objective is to learn a prior distribution of the latent space that is conditioned on an input variable y such that $p_\theta(z|y)$ (see Fig. 1b). The conditioning of the distributions results in a prior that is modulated, by the input variable, creating a method to control modality of the output.

3.3 Fuzzy C-VAE

We propose C-VAE architecture where additional conditions are applied only to $/mu$ component in order to reorganize the latent space structure (see Fig1c). Reorganization achieved by using fuzzy term functions, where each term associated with sole condition i.e. label. Multidimentional Gaussian function is used to represent the fuzzy term function:

$$\nu(z, A_i) = e^{||[A_i . \tilde{z}]_{1 \cdots m}||^2},$$

where m is a size of latent space, i term number, $\tilde{z} = [z_1, z_2, \ldots, z_m, 1]$ and A_i is transformation matrix in form

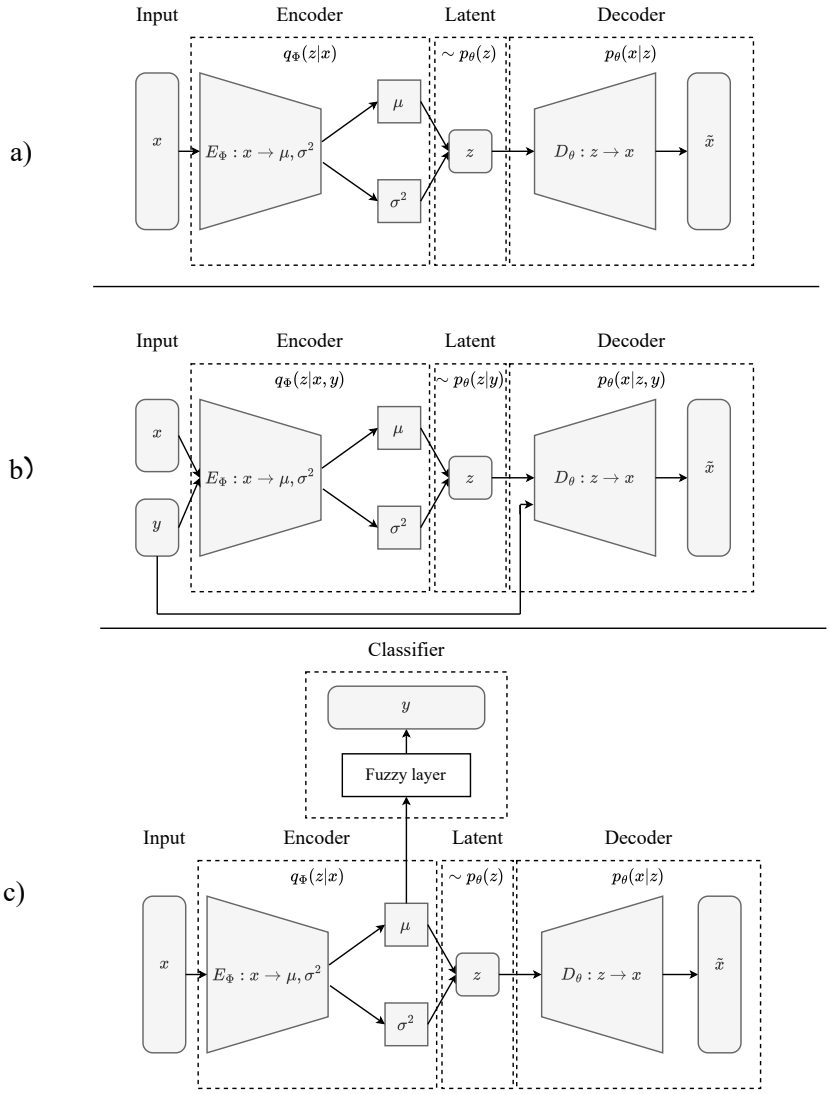

Fig. 1. Overview of the *a*) VAE *b*) CVAE by [10] and *c*) proposed fuzzy C-VAE model.

$$A_{(m+1)\times(m+1)} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} & c_1 \\ a_{2,1} & a_{2,1} & \cdots & a_{2,m} & c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,m} & c_m \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix},$$

with $c_{1\cdots m}$ centroid position and $a_{k,l}$ is a matrix responsible for scaling and aligment of Gaussian in m-dimensional space. Such representation of multidimentional Gaussian term function easyly can be adopted for use in any modern machine learning framework. Set of $\nu(z, A_i)$ we call a fuzzy layer. Intuition behind fuzzy layer is that during training procedure every input vector z will be forced to group closer near centroid of correspoinding term function. Disentangling features of VAE combined with clustering possibilities of fuzzy layer provides a way to learn suprvised latent space which can be useful for anomaly detection and other tasks we discuss further.

3.4 Learning Fuzzy C-VAE

To train fuzzy C-VAE we use the same loss function as in standard VAE with addition of fuzzy layer loss:

$$Loss = MSE(\tilde{x}, x) + KL(\mu, \log\sigma^2) + FZ(\tilde{y}, y),$$

where $MSE(\tilde{x}, x)$ is reconstruction loss, $KL(\mu, \log\sigma^2)$ is the KL-divergence (for more details see [7]) and $FZ(\tilde{y}, y)$ represents the mean squared error between the output and target conditional verctor.

Main drawback of fuzzy layer is that in hight dimensional cases it's hard to find good initial values for centroids and scaling factors mainly due to vanishing gradients. Is such a case it is possible to pass to fuzzy layer subsection of vector $/mu$ leaving remained part to be trained by VAE without any conditional restrictions.

4 Experiments

In this paper we would like to demonstrate ideas of fuzzy C-VAE on playground MNIST dataset [4] in comparison to vanilla VAE. For demonstation purposes we will use 2 dimensional latent space. Network topology for VAE and fuzzy C-VAE are identical in common encoding and decoding part (see Figure 1) and differs only in fuzzy layer. Both models are trained using the Adam optimizer [5].

4.1 Latent space clusterization

On Figure 2 is depicted latent space structure resulted by pure VAE without any conditional restrictions. Cluster structure is not very clear and for some clusters it is not possible to separate. Without application of prior knowledge

about number labels task of extracting corresponding clusters is very challenging. Passing label information directly to fuzzy C-VAE during training leads to more fine-grained latent space structure.

Reconstruction losses for VAE and fuzzy C-VAE during our experiments were almost the same while KL-loss for fuzzy C-VAE was slightly highter all the time.

Classification accuracy of fuzzy C-VAE on 2d case is fairly poor 97% but better results can be achieved with larger latern vectors sizes.

Figure 3 shows how fuzzy C-VAE is able to classify numbers.

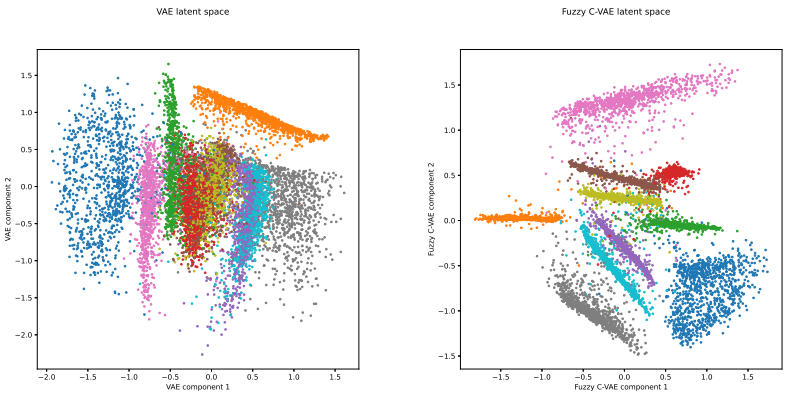

Fig. 2. Latent space granulation for vanila VAE (left) and Fuzzy C-VAE (right)

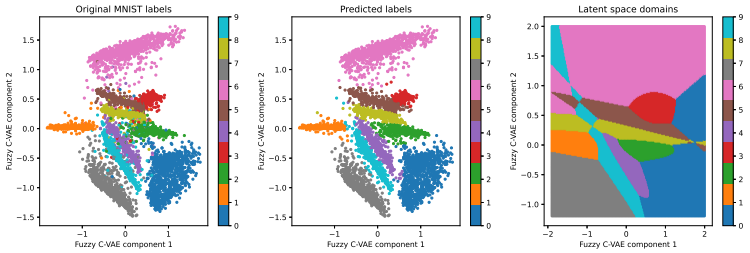

Fig. 3. Fuzzy C-VAE latent space colored by true number labels (left) predicted number labels (center) and number domais at latent space directed by fuzzy layer (right)

4.2 Controlled samples generation

After training procedure cluster properties such as mean and variance can be obtained. This allows us to get an idea of the relationships between clusters

and to plan the generation of new samples with predefined properties. Figure 4 demonstrates example how the digit 7 is purposely made into the digit 9.

Not all transitions from one digit to another are possible without crossing clusters of other digits. However, interpreted topology of the latent space gives more possibilities for generating synthetic samples.

Fig. 4. Fuzzy C-VAE latent space colored by true number labels (left) predicted number labels (center) and predicted outline class (right)

4.3 Anomaly detection

For anomaly detections, fuzzy C-VAE model provides a number of possibilities. Here we present a straightforward approach and leave more complex scenarios for future work. The idea is that clusters in the latent space after model training are represented by localized groups of points with not very complex structure. This make possible to apply standart anomaly detection routine to every individual cluster. As anomalous samples we used the EMNIST [2] dataset which has alphabetic characters never seen by trained models.

Isolated forest [8] classifier was used for anomaly detection to both the fuzzy C-VAE and VAE representations. Results are summarized in Table 1. Fuzzy C-VAE model with isolation forest demonstrate quite interpretable results, in which the anomaly detection rate are worse for those symbols whose outlines look more like digits. At the same time for VAE straightforward approach does not work well and it is clear that more sophisticated approach is required.

4.4 Learning on partially labeled dataset

Structure of fuzzy C-VAE model allows to pass unlabeled data sample to update only VAE weight. This feature can be useful with training on dataset with limited expert knowledge in order to make VAE part of model more reliable. To achieve this loss function has to be modified

$$Loss = MSE(\tilde{x}, x) + KL(\mu, \log \sigma^2) + \gamma * L(x, \tilde{y}, y),$$

where $\gamma > 1$ is a hyperparameter that controls the influence of fuzzy part of model and

$$L(x, \tilde{y}, y) = \begin{cases} 0, & x \text{ is unlabeled} \\ FZ(\tilde{y}, y), & x \text{ is labeled} \end{cases}.$$

Symbols	Fuzzy CVAE	VAE
0123456789	0.18	0.32
Oo	0.36	0.91
Ww	0.39	0.01
Nn	0.61	0.24
Mm	0.62	0.01
Ii	0.63	0.24
Ff	0.65	0.08
Vv	0.66	0.15
Uu	0.69	0.14
Ll	0.70	0.33
Aa	0.71	0.47
Pp	0.71	0.11
Tt	0.73	0.16
Dd	0.75	0.64
Gg	0.76	0.54
Qq	0.76	0.61
Yy	0.78	0.08
Bb	0.79	0.61
Kk	0.79	0.29
Cc	0.80	0.83
Hh	0.81	0.23
Ee	0.82	0.55
Ss	0.82	0.63
Rr	0.83	0.13
Jj	0.85	0.42
Zz	0.86	0.79
Xx	0.89	0.20

Table 1. Anomaly detection rates for symbols from MNIST and EMNIST datasets

On Fugure 5 we provide model performance on MNIST dataset with different unlabeled rate.

With this results we can conclude that it's reasonable to pass unlabeled samples during trainig to maintain better reconstruction quality. On the other side presented technique may be useful during markup phase to determine while dataset sufficiently labeled and further markup is not useful

5 Discussion

In this paper, we introduced a novel fuzzy inference layer to improve the performance of conditional VAEs. We achieve this by making trainable multidimentional representation of fuzzy term. The method that we proposed is applicable to a variety of conditional VAE models. The efficacy of our proposed method was demonstrated on MNIST dataset. Whereas fuzzy conditions influence on VAE should be discussed more deeply we leave it for future work.

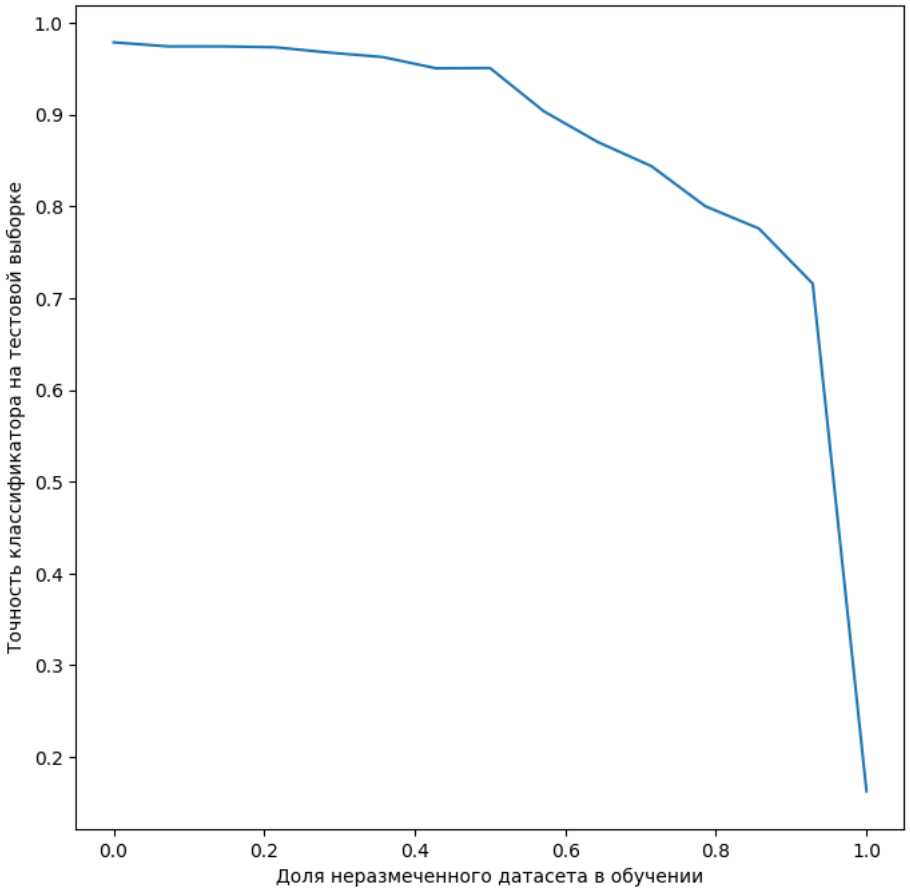

Fig. 5. Model test set accuracy on MNIST dataset with different unlabeled rate.

References

1. Bölat, K., Kumbasar, T.: Interpreting variational autoencoders with fuzzy logic: A step towards interpretable deep learning based fuzzy classifiers. In: 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). pp. 1–7 (July 2020). https://doi.org/10.1109/FUZZ48607.2020.9177631
2. Cohen, G., Afshar, S., Tapson, J., Schaik, A.V.: Emnist: Extending mnist to handwritten letters. 2017 International Joint Conference on Neural Networks (IJCNN) (2017). https://doi.org/10.1109/ijcnn.2017.7966217
3. de Campos Souza, P.V.: Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature. Applied Soft Computing **92**, 106275 (2020). https://doi.org/https://doi.org/10.1016/j.asoc.2020.106275, https://www.sciencedirect.com/science/article/pii/S1568494620302155

4. Deng, L.: The mnist database of handwritten digit images for machine learning research [best of the web]. IEEE Signal Processing Magazine **29**(6), 141–142 (2012). https://doi.org/10.1109/MSP.2012.2211477
5. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017). https://doi.org/10.48550/arXiv.1412.6980
6. Kingma, D.P., Welling, M.: An introduction to variational autoencoders. Foundations and Trends® in Machine Learning **12**(4), 307–392 (2019). https://doi.org/10.1561/2200000056, http://dx.doi.org/10.1561/2200000056
7. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2022). https://doi.org/https://doi.org/10.48550/arXiv.1312.6114
8. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: 2008 Eighth IEEE International Conference on Data Mining. pp. 413–422 (2008). https://doi.org/10.1109/ICDM.2008.17
9. Ramchandran, S., Tikhonov, G., Lönnroth, O., Tiikkainen, P., Lähdesmäki, H.: Learning conditional variational autoencoders with missing covariates. Pattern Recognition **147**, 110113 (2024). https://doi.org/https://doi.org/10.1016/j.patcog.2023.110113, https://www.sciencedirect.com/science/article/pii/S0031320323008105
10. Sohn, K., Yan, X., Lee, H.: Learning structured output representation using deep conditional generative models. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2. p. 3483–3491. NIPS'15, MIT Press, Cambridge, MA, USA (2015). https://doi.org/10.5555/2969442.2969628