Semi-supervised Gaussian Mixture Variational Autoencoder for Pulse Shape Discrimination

SEMI-SUPERVISED GAUSSIAN MIXTURE VARIATIONAL AUTOENCODER FOR PULSE SHAPE DISCRIMINATION

Abdullah Abdulaziz[(1)], *Jianxin Zhou*[(2)], *Angela Di Fulvio*[(2)], *Yoann Altmann*[(1)], *Stephen McLaughlin*[(1)]

[(1)]School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, U.K.
[(2)]Department of Nuclear, Plasma, and Radiological Engineering,
University of Illinois at Urbana-Champaign, Urbana, U.S.A.

ABSTRACT

We address the problem of pulse shape discrimination (PSD) for radiation sources characterization by leveraging a Gaussian mixture variational autoencoder (GMVAE). When using PSD to characterize radiation sources, the number of emission sources and types of pulses to be classified is usually known. Yet, the creation of labeled data can be challenging for some classes as it requires expensive expert annotation. In this context, GMVAE can learn the distinct features of pulses from only unlabeled data. We show that classification accuracy can be further enhanced by adopting a semi-supervised GMVAE with auxiliary loss functions when labeled data are available. The preliminary results on two datasets with different number of classes suggest superior performance of GMVAE compared to other classifiers such as Gaussian mixture model (GMM) for unsupervised and semi-supervised learning and random forest for supervised learning.

Index Terms— Semi-supervised classification, Gaussian mixture variational autoencoder, pulse shape discrimination.

1. INTRODUCTION

In many real world classification applications, there are often a variety of unlabeled data, however obtaining class labels for entire datasets is expensive or sometimes impossible. In this work, we address the task of classifying different types of pulses for radiation sources. In this context, organic scintillation detectors are capable of detecting both gamma rays and neutrons and are also able to discriminate between them in a process known as pulse shape discrimination (PSD) thanks to the different shape of light pulses produced by gamma-ray and neutron interactions with the detectors. Organic scintillation detectors are widely used in a variety of applications, including characterization of nuclear physics interactions and safeguards of nuclear materials [1, 2]. In these scenarios, a few neutron pulses are often the signature of interest to be detected in the presence of a high-intensity gamma-ray background. Traditional PSD methods are based on gated charge integration (CI) of the detected pulses and achieve good classification accuracy [3]. However, the shapes of neutron and gamma-ray pulses are increasingly similar as the deposited energy decreases, hence their discrimination is particularly challenging at low energies. Therefore, classification methods that utilize both unlabeled data jointly with trusted labeled data are needed. Semi-supervised learning methods handle the classification problem when only a small subset of the observations are labeled. These methods rely on the properties of unlabeled data to improve the decision boundaries and increase the classification accuracy in comparison with classifiers depending on labeled data alone.

In this paper, we introduce a semi-supervised variational autoencoder with Gaussian mixture latent space (GMVAE) for solving the PSD problem. GMVAE [4, 5, 6, 7, 8] is a powerful tool for clustering that allows learning a low-dimensional space in which the different types of pulses are characterized by a Gaussian mixture model. The adopted GMVAE optimizes two auxiliary loss functions in addition to maximizing the evidence lower-bound (ELBO) [5, 9]. The auxiliary losses, namely the labeling loss and the triplet embedding loss, allow for semi-supervised learning when labeled pulses are available. The resulting semi-supervised GMVAE can learn discriminant features from both labeled and unlabeled pulses, hence improving the classification accuracy. To demonstrate its robustness, we validate our method on two different datasets.

The remainder of the paper is structured as follows. In Section 2, we introduce the variational autoencoder and its Gaussian mixture variant. The semi-supervised GMVAE adopted here and its training procedure are presented in Section 3. Experiments and results are presented in Section 4. Conclusions are finally reported in Section 5.

2. VARIATIONAL AUTOENCODER

Variational autoencoders (VAEs) [10, 11] introduce the use of neural networks to perform variational Bayesian inference

This work was supported by the Royal Academy of Engineering under the Research Fellowship scheme RF201617/16/31 and by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/T00097X/1.

over a set of training data points. VAEs generally consist of two models: the inference model and the generative model. The essence of VAEs is to learn a low-dimensional latent space representation that can accurately capture the important features of the high-dimensional dataset. The Gaussian mixture VAE (GMVAE) [4, 5, 6, 7, 8] is one of the variants of VAE which can be used for joint nonlinear dimensionality reduction and clustering or semi-supervised learning. In contrast to the standard VAE which contains a single continuous latent vector $\boldsymbol{z}$ in latent space, the GMVAE also contains a discrete latent variable $\boldsymbol{y}$ representing the data class or cluster. The observed data $\boldsymbol{x}$ are assumed to follow a generative process:

$$\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}) = \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\ \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{y})\ \mathrm{p}(\boldsymbol{y}), \tag{1}$$

$$\boldsymbol{y} \sim \mathrm{Cat}(\boldsymbol{y}|\ 1/k), \tag{2}$$

$$\boldsymbol{z}|\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{z}|\ \boldsymbol{\mu}_{\boldsymbol{z}_{\boldsymbol{\theta}}}(\boldsymbol{y}), \boldsymbol{\sigma}^2_{\boldsymbol{z}_{\boldsymbol{\theta}}}(\boldsymbol{y})), \tag{3}$$

$$\boldsymbol{x}|\boldsymbol{z} \sim \mathcal{B}(\boldsymbol{x}|\ \boldsymbol{\mu}_{\boldsymbol{x}_{\boldsymbol{\theta}}}(\boldsymbol{z})), \tag{4}$$

where k represents the number of classes and $\boldsymbol{\mu}_{\boldsymbol{z}_{\boldsymbol{\theta}}}$, $\boldsymbol{\sigma}^2_{\boldsymbol{z}_{\boldsymbol{\theta}}}$ and $\boldsymbol{\mu}_{\boldsymbol{x}_{\boldsymbol{\theta}}}$ are given by the GMVAE generative network with parameters $\boldsymbol{\theta}$. The objective of GMVAE is to approximate the posterior distribution $\mathrm{p}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x})$ which is intractable and typically approximated by a factorized posterior, namely the inference model:

$$\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x}) = \mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})\ \mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}), \tag{5}$$

$$\boldsymbol{y}|\boldsymbol{x} \sim \mathrm{Cat}(\boldsymbol{y}|\ \pi_{\boldsymbol{\phi}}(\boldsymbol{x})), \tag{6}$$

$$\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{z}|\ \boldsymbol{\mu}_{\boldsymbol{z}_{\boldsymbol{\phi}}}(\boldsymbol{x},\boldsymbol{y}), \boldsymbol{\sigma}^2_{\boldsymbol{z}_{\boldsymbol{\phi}}}(\boldsymbol{x},\boldsymbol{y})), \tag{7}$$

where $\boldsymbol{\phi}$ are the parameters of the inference network. The categorical distribution $\mathrm{Cat}(\boldsymbol{y}|\ \pi_{\boldsymbol{\phi}}(\boldsymbol{x}))$ is approximated using the Gumbel-Softmax (Concrete) distribution [12, 13]. Fig. 1 shows a schematic of the GMVAE. The parameters of the inference and generative networks, $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$, are learnt by maximizing the evidence lower-bound (ELBO) [5, 6], that is a lower bound on the log probability of the observations:

$$\begin{aligned}\mathrm{ELBO} = &-\alpha\mathrm{KL}(\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})||\ \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{y})) \\ &- \beta\mathrm{KL}(\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x})||\ \mathrm{p}(\boldsymbol{y})) + \mathrm{E}_{\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}[\log \mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})],\end{aligned} \tag{8}$$

where $\alpha, \beta > 0$ are hyper-parameters controlling the weights of the different priors. KL stands for the Kullback-Leibler divergence and the last term is given by the reconstruction loss.

3. SEMI-SUPERVISED GMVAE

Eq. (8) corresponds to training the GMVAE in an unsupervised fashion. To increase the classification accuracy, prior knowledge about labeled data can be incorporated into the model in the form of auxiliary loss functions [5, 9].

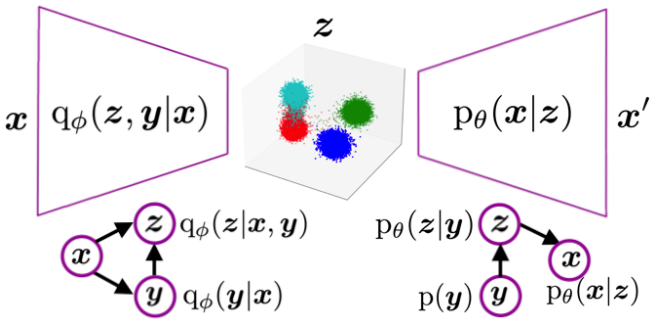


Fig. 1. Schematic of Gaussian mixture variational autoencoder.

We start by assigning labels to unlabeled data based on the k-nearest neighbors (k-NN) rule [14]. Given the fact that data points belonging to the same class typically share similar features in the latent space, each unlabeled pulse is classified by the majority label of its k-NN in the latent space. Doing so, the first auxiliary loss function, namely the labeling loss, is a cross entropy loss between the assigned labels and the unsupervised predicted labels.

Motivated by the fact that k-NN classification is dependant on the way that distances are computed between different pulses, k-NN classification can be greatly enhanced by learning an appropriate distance metric from the assigned labels. Therefore, we adopt the triplet embedding loss as a second auxiliary loss function to regularize the latent space [15, 16]. The triplet loss penalizes large distances between features belonging to the same class while penalizing small distances between pulses with non-matching labels. The final objective for the semi-supervised GMVAE takes the form:

$$\mathcal{L}_{\mathrm{GMVAE}} = \mathrm{ELBO} + \omega\mathcal{L}_{\mathrm{label}} + \gamma\mathcal{L}_{\mathrm{triplet}}, \tag{9}$$

where $\mathcal{L}_{\mathrm{label}}$ is the labeling loss, $\mathcal{L}_{\mathrm{triplet}}$ is the triplet embedding loss and $\omega, \gamma > 0$ are the hyper-parameters controlling the weights of the auxiliary loss functions.

3.1. GMVAE training

We adopt dense layers for the architecture of the inference and generative networks. All layers, except the output layer of the generative network, are followed by rectified linear unit (ReLU) as activation function. The output layer is followed by logistic (sigmoid) activation function.

The inference network starts with three consecutive layers of the respective sizes (L, 200, 200) to learn the features of the input pulse $\boldsymbol{x} \in \mathbb{R}^L$, dubbed $\boldsymbol{x}_f \in \mathbb{R}^{200}$. The probability distribution $\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x})$ is parameterized by a layer of size k to get the logits where its input is $\boldsymbol{x}_f$. The output categories are given by the Gumbel-Softmax (Concrete) distribution [12, 13] of the logits. The probability distribution $\mathrm{q}_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})$ is parameterized by two layers of size 20 that take the concatenation of $\boldsymbol{x}_f$ and the categories as an input to produce $\boldsymbol{\mu}_{\boldsymbol{z}_{\boldsymbol{\phi}}}$ and

$\boldsymbol{\sigma}^2_{\boldsymbol{z}_\phi}$, respectively, and the output vector is given by the reparameterization trick of the Gaussian distribution. For the generative network, $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{y})$ is represented by two layers of size 20 to produce $\boldsymbol{\mu}_{\boldsymbol{z}_\theta}$ and $\boldsymbol{\sigma}^2_{\boldsymbol{z}_\theta}$, respectively. Finally, the distribution $\mathrm{p}_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ is parameterized by three consecutive layers of the respective sizes (200, 200, L) to produce the output vector $\boldsymbol{\mu}_{\boldsymbol{x}_\theta}$.

GMVAE is trained over 500 epochs using the Adam optimizer [17] with learning rate $\mathrm{lr} = 10^{-4}$. For the labeling loss $\mathcal{L}_{\text{label}}$, we considered only the nearest neighbor (1-NN). The batch size was set to 100 and the hyper-parameters α, β, ω and γ setting the trade-off between the different priors are all set to 1.

4. EXPERIMENTS AND RESULTS

In this section, we showcase the performance of GMVAE utilizing two PSD datasets with normalized pulses and $L = 296$ (the size of each pulse). The first dataset, denoted by 3-class dataset, encompasses three groups of pulses, classified as alpha, neutron, and gamma-ray pulses. The second dataset, denoted by 4-class dataset, includes pulses belonging to four classes: alpha, deuterium, proton, and electron. The first classification refers to the radiation interacting with the detector material, while the second to the particle depositing energy inside the detector. All data are measured leveraging a recently developed deuterated trans-stilbene organic scintillator (d_{12}-stilbene) [18]. The scintillator is coupled with HAMAMATSU R6091 head-on photomultiplier tube and H6559 photomultiplier assembly. The detector pulses are collected by a CAEN DT5730 14-bit 500 Ms/s desktop digitizer. The digitizer was used in the "waveform" mode, which samples the signal every 2 ns. The total record length for each pulse is 592 ns. Alpha particles are directly-ionizing charged particles, which can directly deposit energy through ionization in the crystal and hence generate a detectable light pulse. Conversely, gamma rays and neutrons are neutral particles, and they transfer their energy to the organic scintillator material through charged produced secondary ions, i.e., Compton electrons and protons or deuterium ions, respectively.

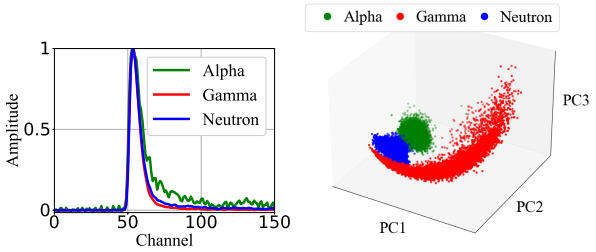
The data labelling, for pulses deemed reliable has been performed using a standard method which consists of the total integral of each pulse and the integral of the tail of each pulse, to compute the so-called tail-to-total ratio (TTR) [19]. The 2D representation using the total integral and TTR allows the discrimination of neutrons and gamma rays (using a quadratic boundary) and this method is referred to as the charge integration method [20].

For the 3-class dataset, we train using 27000 pulses (9000 per class) and test on 3000 pulses (1000 per class). Fig. 2, (a) shows one sample of each class of the training dataset and Fig. 2, (b) presents the training dataset in 3D space projected by PCA. For the 4-class dataset, we train using 36000 pulses (9000 per class) and test on 4000 samples (1000 per class). Similarly, Fig. 3, (a) shows one sample of each class of the training dataset and Fig. 3, (b) presents the training dataset in 3D space projected by PCA. In both cases, the pulses present similar shapes, with subtle differences.

We evaluate the performance of GMVAE in terms of classification accuracy by training with different numbers of labeled data points (from 0 to 100% of the size of the training dataset). The accuracy of GMVAE is compared to a semi-supervised version of the Gaussian mixture model (GMM) [21] implemented using the expectation-maximization (EM) algorithm. The accuracy with labeled data is also compared to different classifiers for supervised learning. These are: k-NN [14], logistic regression [22], support vector machine (SVM) [23], linear discriminant analysis (LDA), quadratic discriminant analysis (QDA) [24], decision tree [25], random forest [26] and AdaBoost [27]. Note that, in contrast to GMVAE and GMM which can be trained on both labeled and unlabeled data, the supervised classifiers are trained only on the subset of labeled data.

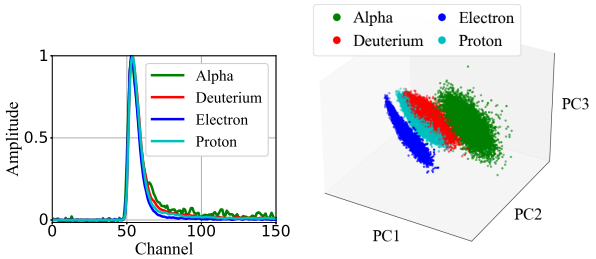
Fig. 4 displays the classification accuracy obtained by GMVAE, GMM and the three best supervised classifiers for the 3-class dataset. This dataset exhibits a relatively easy problem where most supervised classifiers achieved high accuracy (around 99.7%) for all tests with labeled data $\geq 5\%$ of the size of the training dataset (equivalently ≥ 1350 labels). Interestingly, GMVAE maintains the same high accuracy for all tests with labeled data $\geq 0.1\%$. When compared to unsupervised GMM which achieves 90.1% accuracy, unsupervised GMVAE reaches 98.6%. This confirms the ability of GMVAE to learn the discriminative features of the different classes from only unlabeled data. Fig. 5 presents the 3D latent space of GMVAE with 1% labels projected by PCA. We can see the clear separation between the different classes for both the training and testing data points.

Similarly, Fig. 6 shows the classification accuracy obtained by GMVAE, GMM and the three best supervised classifiers for the 4-class dataset. This dataset represents a more challenging problem where two of the classes, namely the deuteriums and protons, are very similar. The figure confirms the superiority of GMVAE in comparison with GMM in learning meaningful features from both labeled and unlabeled data, suggesting that non-linearities contribute to better classification results. Furthermore, supervised GMVAE reaches a peak of 97.2% compared to 92.4% for random forest that gives the best results among all other supervised classifiers. The 3D latent space of GMVAE with 1% labels projected by PCA is shown in Fig. 7. The figure illustrates the high similarity between the pulses associated with deuteriums and protons. We notice that the classification accuracy of GMVAE linearly increases by increasing the number of labeled data points.



(a) (b)

Fig. 2. (a) Samples of the 3-class dataset (the samples have 296 channels but we show only the first 150 channels for better visualization of the samples' tail). (b) The 3-class training dataset in 3D space projected by PCA.



(a) (b)

Fig. 3. (a) Samples of the 4-class dataset (the samples have 296 channels but we show only the first 150 channels for better visualization of the samples' tail). (b) The 4-class training dataset in 3D space projected by PCA.

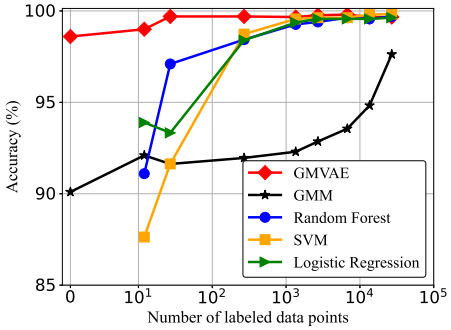
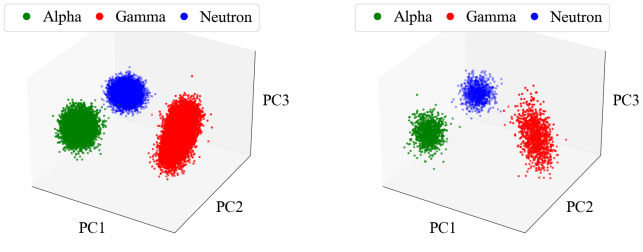


Fig. 4. Classification accuracy of different classifiers plotted as a function of the number of labeled data points for the 3-class dataset.

5. CONCLUSIONS

We presented a semi-supervised classification approach for solving the PSD problem. Our approach leveraged a GMVAE which can be trained in an unsupervised fashion. To increase the classification accuracy and take advantage of the usually available but scarce labeled data, we introduced a semi-supervised GMVAE with auxiliary loss functions: the labeling loss which is a cross entropy loss between the assigned labels based on the k-NN rule and the unsupervised predicted labels, and the triplet embedding loss that regularizes the latent space to enhance the k-NN assignment. We studied the performance of GMVAE utilizing two datasets with different number of classes. The results confirmed the superior performance of GMVAE compared to other classifiers such as GMM for unsupervised and semi-supervised learning and random forest for supervised learning. Future work includes the use of GMVAE for the classification with piled up pulses, i.e., outliers.



(a) (b)

Fig. 5. GMVAE latent space of the 3-class dataset with 1% labels. The latent vectors from (a) training and (b) testing data points are project by PCA onto the 3D space.

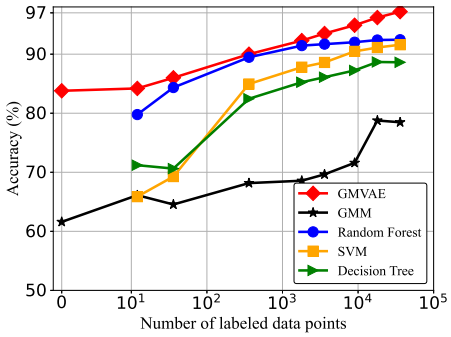
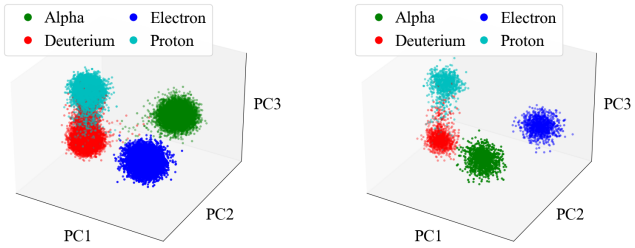


Fig. 6. Classification accuracy of different classifiers plotted as a function of the number of labeled data points for the 4-class dataset.



(a) (b)

Fig. 7. GMVAE latent space of the 4-class dataset with 1% labels. The latent vectors from (a) training and (b) testing data points are project by PCA onto the 3D space.

6. REFERENCES

[1] A. Di Fulvio, T.H. Shin, T. Jordan, C. Sosa, M.L. Ruch, S.D. Clarke, D.L. Chichester, and S.A. Pozzi, "Passive assay of plutonium metal plates using a fast-neutron multiplicity counter," *NIMPR, Section A*, vol. 855, pp. 92–101, 2017.

[2] A. Di Fulvio et al., "Fast-neutron multiplicity counter for active measurements of uranium oxide certified material," *NIMPR, Section A*, vol. 907, pp. 248–257, 2018.

[3] M.M. Bourne, S.D. Clarke, M. Paff, A. Di Fulvio, M. Norsworthy, and S.A. Pozzi, "Digital pile-up rejection for plutonium experiments with solution-grown stilbene," *NIMPR, Section A*, vol. 842, pp. 1–6, 2017.

[4] R. Shu, "Gaussian mixture vae: Lessons in variational inference, generative models, and deep nets," .

[5] J. A. Figueroa, "Semi-supervised learning using deep generative models and auxiliary tasks," in *NeurIPS*, 2019.

[6] M. Collier and H. Urdiales, "Scalable deep unsupervised clustering with concrete gmvaes," *arXiv preprint arXiv:1909.08994*, 2019.

[7] R. Charakorn, Y. Thawornwattana, S. Itthipuripat, N. Pawlowski, P. Manoonpong, and N. Dilokthanakul, "An explicit local and global representation disentanglement framework with applications in deep clustering and unsupervised object detection," *arXiv preprint arXiv:2001.08957*, 2020.

[8] Y. B. Varolgüneş, T. Bereau, and J. F. Rudzinski, "Interpretable embeddings from molecular simulations using gaussian mixture variational autoencoders," *Machine Learning: Science and Technology*, vol. 1, no. 1, pp. 015012, 2020.

[9] J. A. Figueroa and A. R. Rivera, "Learning to cluster with auxiliary tasks: a semi-supervised approach," in *IEEE SIBGRAPI*, 2017, pp. 141–148.

[10] D.P. Kingma and M. Welling, "Auto-encoding variational bayes," in *ICLR*, 2014.

[11] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[12] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.

[13] C. Maddison, A. Mnih, and Y. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *ICLR*, 2017.

[14] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[15] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Adv. Neural Inf. Process. Syst.*, 2006, pp. 1473–1480.

[16] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE CVPR*, 2015, pp. 815–823.

[17] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic gradient descent," in *ICLR*, 2015, pp. 1–15.

[18] N. Gaughan, J. Zhou, F.D. Becchetti, R.O. Torres-Isea, M. Febbraro, N. Zaitseva, Y. Altmann, and A. Di Fulvio, "Characterization of stilbene/d12 for neutron spectroscopy without time of flight," *NIMPR, Section A*, 2021.

[19] J.K. Polack, M. Flaska, A. Enqvist, C.S. Sosa, C.C. Lawrence, and S.A. Pozzi, "An algorithm for charge-integration, pulse-shape discrimination and estimation of neutron/photon misclassification in organic scintillators," *NIMPR, Section A*, vol. 795, pp. 253–267, 2015, cited By 44.

[20] F. D. Brooks, "A scintillation counter with neutron and gamma-ray discriminators," *Nucl. Instrum. Methods*, vol. 4, no. 3, pp. 151–163, 1959.

[21] L. Xu, "Unsupervised learning by em algorithm based on finite mixture of gaussians," in *World Congress on Neural Networks (Portland, OR)*, 1993, vol. 2, pp. 431–434.

[22] M. Svensén and C. M. Bishop, "Pattern recognition and machine learning," 2007.

[23] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM TIST*, vol. 2, no. 3, pp. 1–27, 2011.

[24] T. Hastie, r. Tibshirani, and J. Friedman, "The elements of statistical learning," 2008.

[25] W.-Y. Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[26] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[27] T. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class adaboost," *Statistics and its Interface*, vol. 2, no. 3, pp. 349–360, 2009.