Proceedings of the 2nd International Conference on Computing Innovation and Applied Physics (CONF-CIAP 2023)
DOI: 10.54254/2753-8818/5/20230405

Study of MLP-based classification of multi-cluster TPC signals

Naiyuan Liu[1*], Dehao Shao[2], Jiaqi Tan[3], Qianqian Wan[4], Tianyu Zhou[5]

[1]College of Art and Science, New York University, New York, 10003, US,
[2]Milton International School, Qingdao, 266000, China
[3]Newchannel International School, Beijing, 100071, China
[4] Hefei No.8 Highschool, Anhui, 230000, China
[5]Henry Samueli School of Engineering, University of California, Irvine, 92617, US

nl1829@nyu.edu

Abstract. This study focuses on the classification of multi-cluster events based on a parameterization of data from a time projection chamber using machine learning. Samples containing a mixture of single and overlapping two-cluster events, both in one and two dimensions, were studied using multi-layer perceptrons and other MVA algorithms provided in the Scikit-learn package. The classification was based on various sets of features and classification accuracies of up to 97% for 1D clusters and 97% for 2D clusters were obtained. This study demonstrates that the efficient classification of signals for further processing through machine learning is feasible and efficient.

Keywords: Machine learning, Scikit-learn, Classification.

1. Introduction

Classification of image and sensor data is a prototypical use-case for multivariate analysis (MVA) techniques in many areas of science, and technology, including wide-ranging commercial applications. In this study, we apply MVA techniques based on machine learning (ML) to parameterized data representing electronic signals from particle detectors used in fundamental physics, materials analysis, and medical applications. Efficient and accurate processing of such data, which in some applications amounts to 100's of petabyte per year, presents a significant computing challenge, as the most general analysis methods through, e.g., multi-dimensional fits may be prohibitively resource-expensive. We investigate how a fast classification through ML algorithms may be used to accurately select a subset of the signal events for further processing, reducing the overall computing cost. Various MVA algorithms provided in the scikit-learn Python library are investigated [1,2].

The specific signal distributions we study are based on time projection chamber (TPC) signals [3]. TPCs are detectors used to accurately measure the 3-D trajectories of charged particles traversing a large gas volume. We applied our techniques both to 1-D signals (clusters) corresponding to a single electronic readout channel, as well as 2-D clusters representing the combined signal of a planar arrangement of readout channels. For both cases, the ML algorithms are used as classifiers to discriminate between events where a single cluster is observed (allowing determination of the signal

DOI: 10.54254/2753-8818/5/20230405

location through a simple weighted average) and events where two signals overlap, requiring a more computing extensive deconvolution procedure to accurately determine the signal positions.

The performance of the algorithms is characterized by their classification accuracy averaged over the respective data sets, as well as through the dependence of the accuracy on the distance between the centroids of overlapping signals. In Section 2 of the paper, we discuss the methods and results for the 1-D signal case. The 2-D signal case is described in Section 3 and Section 4 presents a summary and discussion of our findings. Extensive graphical representations of the decision boundaries for various approaches are collected in the Appendix [4,5].

2. Studies of 1-D Cluster Signals

2.1. Data Samples

The Dataset used in this analysis consists of 100,000 one-dimensional samples. Each sample is either a one-cluster event or a two-cluster event. Furthermore, for each sample, the data points of one or two clusters are scattered along a 1 x 48 grid, with the total number of data points within the one or two clusters defined to be the height of the cluster. For the convenience of visualization, the number of data points residing in the one-unit length bin is used in the analysis. Figure 1 demonstrates four possible situations of cluster distributions.

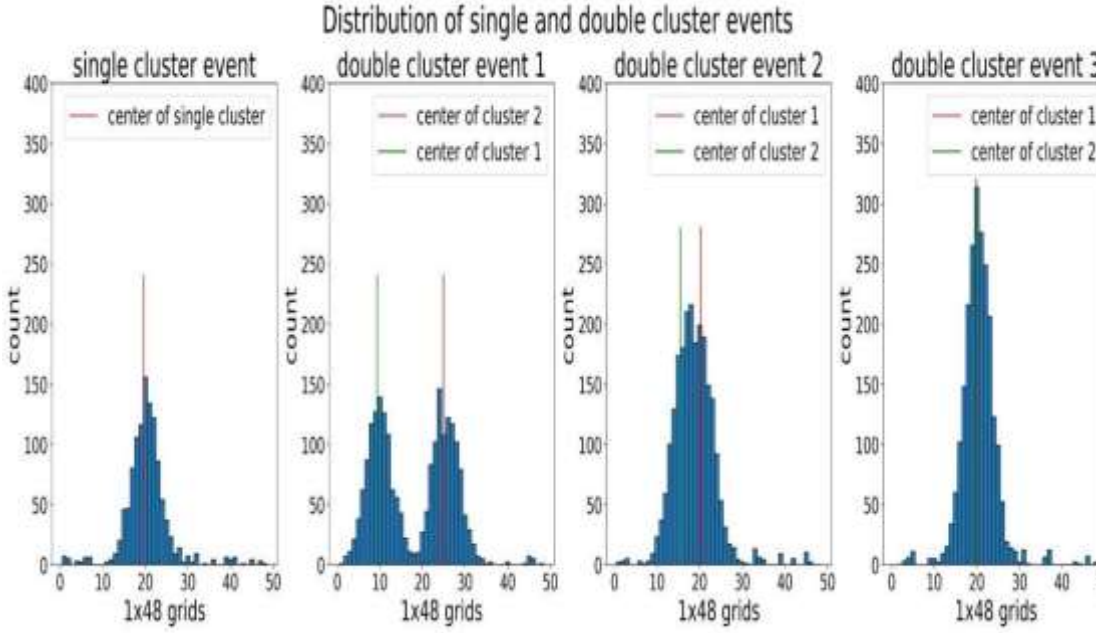

Figure 1(a). single cluster event, (b)two-cluster event clearly separated, (c)two-cluster event relatively close, (d)two-cluster event stacking up.

Figure 1(a) shows the situation of a single cluster event with the vertical line showing its respective center. Figure 1(b)shows the situation of a two-cluster event with their respective center clearly separated, which is also the most obvious type of two-cluster event that could be distinguished from a single-cluster event. Figure 1(c) shows the situation of a two-cluster event with their respective center relatively close to each other. It is already challenging to distinguish the two-cluster event from the single-cluster event as the shape becomes similar to a single-cluster event. Figure 1(d) shows the situation of a two-cluster event with two nearly identical cluster centers. With the distribution of two clusters stacked upon each other, it is extremely difficult to tell whether this is a two-cluster event or a single-cluster event that has a large height by sheer visual speculation. In the following section, two sets of features are extracted from the Dataset and utilized as parameters to distinguish single-cluster

DOI: 10.54254/2753-8818/5/20230405

events from two-cluster events using the multi-layer perceptron provided by the Scikit-learn package [2].

2.2. Classification Using Moments

In this section, the feature that will be used as parameters for the MLP is the moments of the cluster distribution. The second (Variance), third (Skewness), and forth (Kurtosis) moment of each event contained in the Dataset are extracted[4]. To do so, the initial calculation involves the weighted mean W (1)

$$W = \sum_{i=1}^{n} w_i X_i \tag{1}$$

where n is the number of bins of each event, in the case of this Dataset-48. X_i is the total number of data points the

i th bin contains, by definition it is just the height of each bin. And w_i is the proportion of the height of the i th bin compared to the height of the whole event.

After the weighted mean of each event is calculated, it is used to calculate the variance σ^2 (2), skewness $\hat{\mu}_3$ (3), and kurtosis $\hat{\mu}_4$ (4) of each event. The distributions of which are shown in Figure 2, as we can see they show different patterns for single and double-cluster events.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(X_i - W)}{n-1} \tag{2}$$

$$\hat{M}_3 = \frac{\sum_{i=1}^{n}(X_i - W)^3}{(n-1)\sigma^3} \tag{3}$$

$$\hat{\mu}_4 = \frac{\sum_{i=1}^{n}(X_i - W)^4}{(n-1)\sigma^4} \tag{4}$$

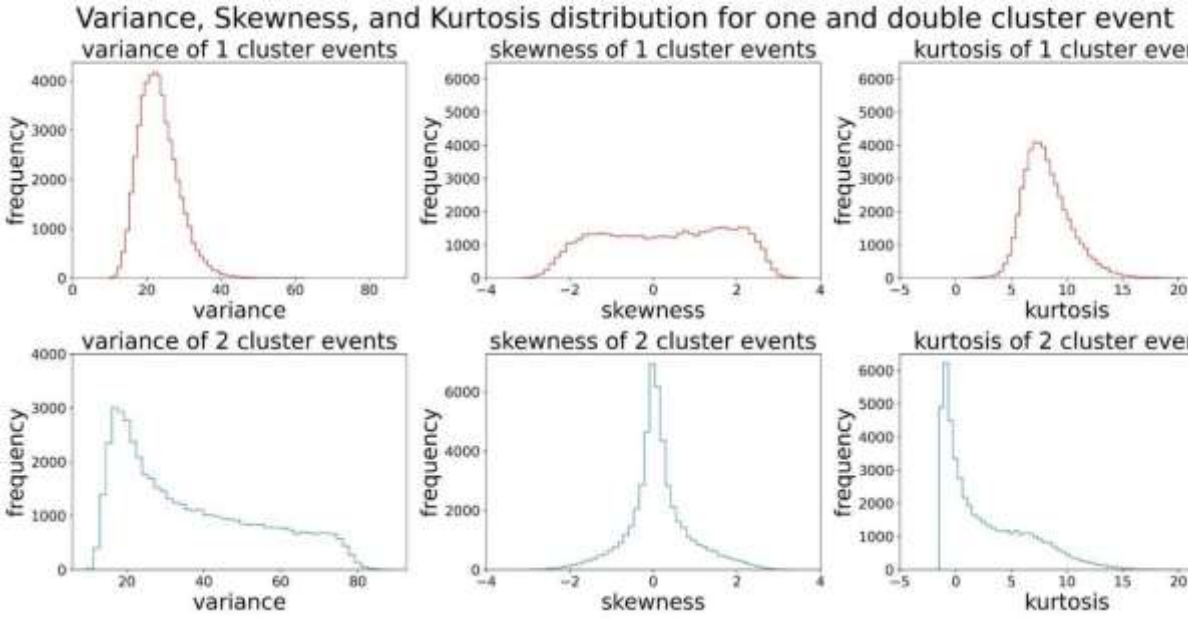

Figure 2. The comparison of three moments between single and two-cluster event, in which a clear difference between the two types of events could be seen.

DOI: 10.54254/2753-8818/5/20230405

After the extraction of the moment information from each event, the entire Dataset are divided and 75% of which are used as the training sample and 25% of which are used as the testing sample. Then the data are imported directly into the MLP classifier. The classifier used in this experiment is set up as follows: hidden_layer_sizes=(100,100,100), max_iter=500, alpha=0.0001, solver='adam', verbose=10, random_state=21,tol=0.000000001. After about 155 iterations, the model converges and yields an accuracy of 83%. You can find visualizations of the decision boundary formed with the MLP classifier and some other models used in the study for the purpose of comparison in Figure 3,4,5,6 [5-7].

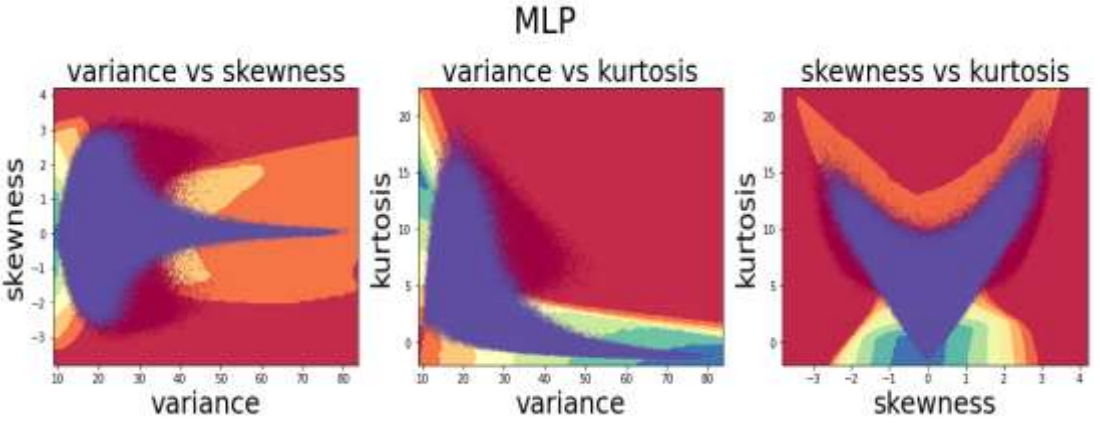

Figure 3. Decision boundary of MLP classifier with an Accuracy of 83% [5].

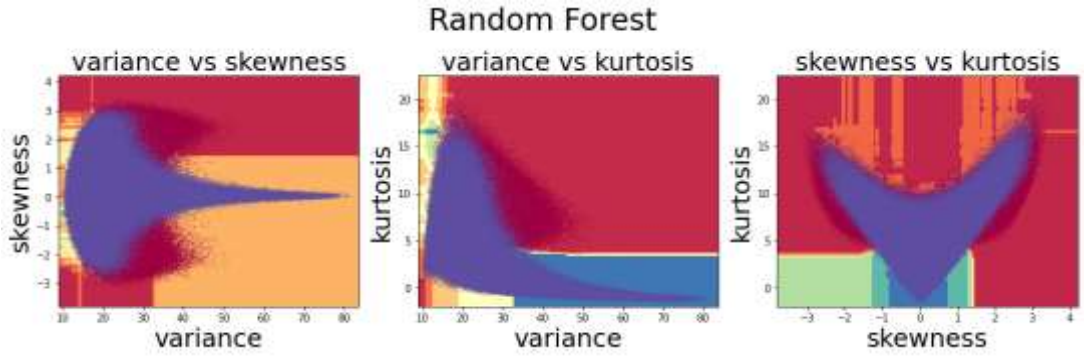

Figure 4. Decision boundary of Random Forest with an Accuracy of 82% [8].

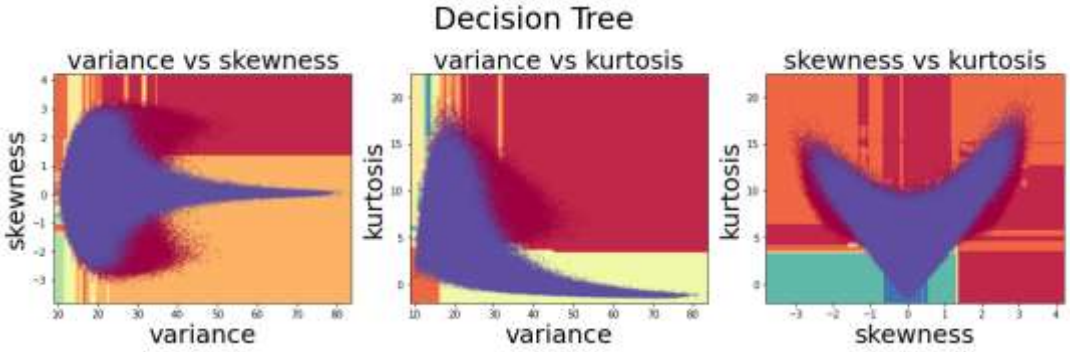

Figure 5. Decision boundary of Decision Tree with an Accuracy of 77% [9].

DOI: 10.54254/2753-8818/5/20230405

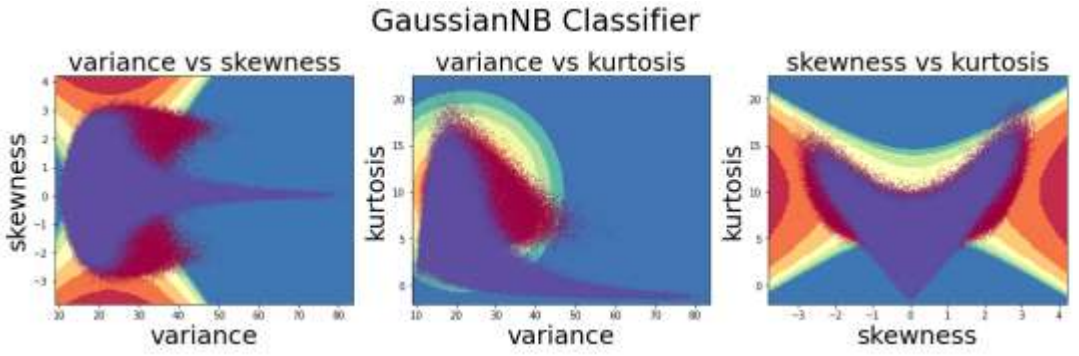

Figure 6. Decision boundary of Gaussian Naive Bayes Classifier with an Accuracy of 78%[10].

2.2.1. Dependence of Accuracy on Two-Cluster Distance. In retrospect from Figure 1, as the distance between two clusters increases, it becomes more effortless to tell the two clusters apart. As a result, upon the examination of the relationship between the distance of two cluster centers and the accuracy of the model, a monotonically increasing function should be the expectation. The Figure 7 below shows that function, and its manifestation is within the assumption in the interval (4,16). As the cluster center's distance increases, the accuracy for the unit length bin also increases gradually toward 100% [11]. However, the MLP classifier also performs quite well when the distance between two cluster centers is pretty close, namely between the interval (0.2), compared to when the distance is relatively close interval (2,4) [5].

This could be explained by Figure 8, where the red dots represent the correctly identified two-cluster events and the blue dots represents the incorrectly identified two-cluster events by the MLP classifier [5]. The green dots represent the single-cluster events. Since single-cluster events do not have a cluster center distance, they are randomly scattered along the axis. In the variance case, it is made obvious that during the questioned interval, when two clusters stack upon each other closely, their variance becomes lower than the typical single-cluster events, thus explaining the abnormality. It could also be inferred that the skewness does not play a crucial role in MLP's identification of two cluster events since all the dots are scattered evenly compared to the variance and kurtosis, where a rather clear separation between the correctly and incorrectly identified two-cluster events could be seen.

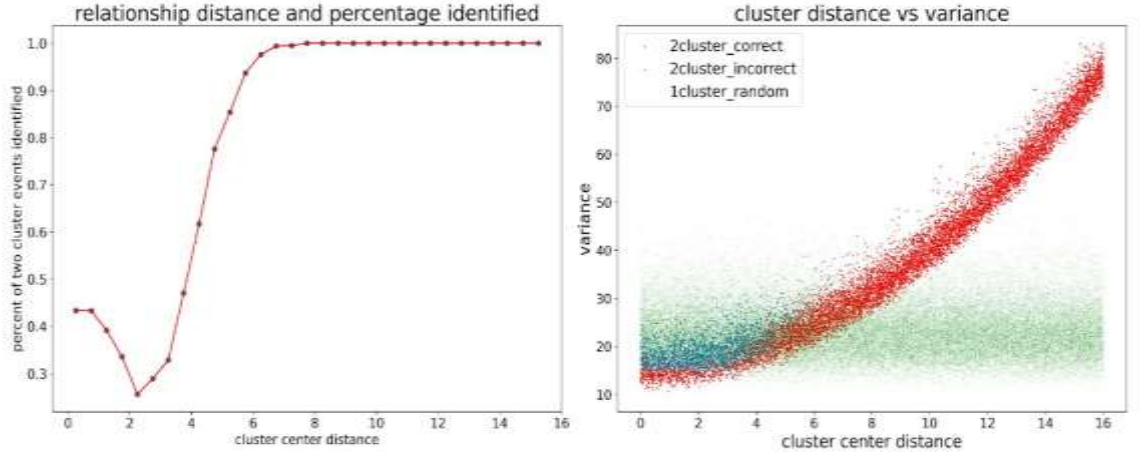

Figure 7. A projection line representing the relationship between the cluster center distance and the percentage of two-cluster event correctly identified (left), A scatter plot of the cluster center distance vs the variance of the respective sample(right).

DOI: 10.54254/2753-8818/5/20230405

Instead of a monotonically increasing curve, an unexpected dip is observed on the interval (0,2). A clear separation of correctly identified two-cluster event (red dots) and the single-cluster event (green dots) is observed, which could explain the abnormality observed on the interval (0,2) from the left figure.

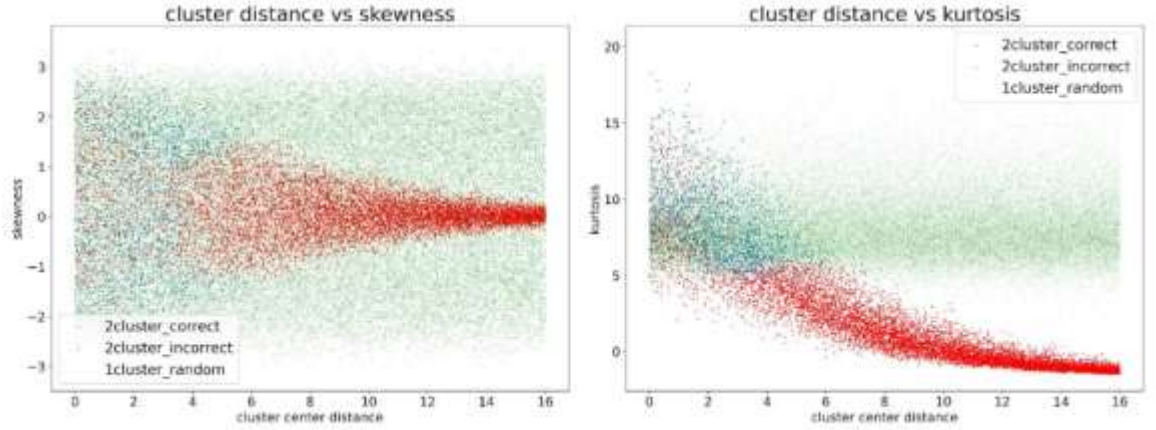

Figure 8. A scatter plot of the cluster center distance vs the skewness of the respective sample(left), A scatter plot of the cluster center distance vs the kurtosis of the respective sample(right).

No clear separation of correctly identified two-cluster event (red dots) and the single-cluster event (green dots) is observed, which means that skewness does not play a crucial row in identifying the two types of events. A clear separation of correctly identified two-cluster event (red dots) and the single-cluster event (green dots) is observed, which means kurtosis also helps with the identification of the two types of events.

2.3. Classification Using Normalized Signal Distribution

The second set of features that are used to train the model is the distribution of the cluster events [12]. Similar to the visualization of the cluster event, a count of the number of data points residing in each 1-unit length bin is conducted, resulting in 48 heights that will be used as the new input of the MLP classifier [5]. To avoid the classifier from learning undesired patter from the sum of these 48 heights and solely focusing on the distribution pattern, normalization is applied first. The 48 heights are summed up first and each height is divided by the total sum, transforming them into the proportion of the total height. Furthermore, since some bins only contain a small number of data points, the proportion calculated from the previous step is multiplied by 2000 to further distinguish them from 0.

After the normalization, the 48 new features extracted from each event are now ready to be imported into the MLP classifier [5]. The Dataset is again divided into 75% training samples and 25% testing samples. The parameter for the MLP is exactly the same as the one used in the moment case. To reiterate: hidden_layer_sizes=(100,100,100), max_iter=500, alpha=0.0001, solver='adam', verbose=10, random_state=21,tol=0.000000001. After 70 iterations, the model converges and yields an accuracy of 97%.

2.4. Discussion

Table 1. Accuracy of Five Types of Classifiers.

Model	MLP with Moments	MLP with Distributions	Random Forest	Decision Tree	GaussianNB
Accuracy	83%	97%	82%	77%	78%

DOI: 10.54254/2753-8818/5/20230405

A short discussion about the discoveries in the one-dimensional case is presented in this section. From the Table 1, the most accurate model that could distinguish single and double-cluster events is the MLP classifier with 48 normalized bin heights as the parameters. All the other methods in the experiments resulted in relatively low accuracies. However, it does not render other methods useless. For one thing, this method takes 48 inputs as parameters, which requires more time and computing power to process and learn from the training sets. On the other hand, if only the three moments are used as parameters, the model would run much faster. Furthermore, as discussed in section 2.2.1, the MLP classifier with three moments as inputs could better distinguish the two-cluster events whose centers are very close to each other through the second moment(variance) [5].

3. Study of 2-D Cluster Signals

3.1. Data Sample

A more complex but at the same time similar two-dimensional case is studies in Section 3. Similar to the data used in the one-dimensional analysis, each sample is still either a one-cluster event or a two-cluster event. The data points of one or two clusters are scattered along a 2d 48x48 grid, instead of the 1d 1x48 grid. The total number of data points within one or two clusters is defined to be the height of the cluster. A count of the number of data points residing in the one-unit length by one-unit length bin is conducted. Below (Figure 9) 2-d histograms displaying the same four types of situations mentioned in the previous 1d section could be found.

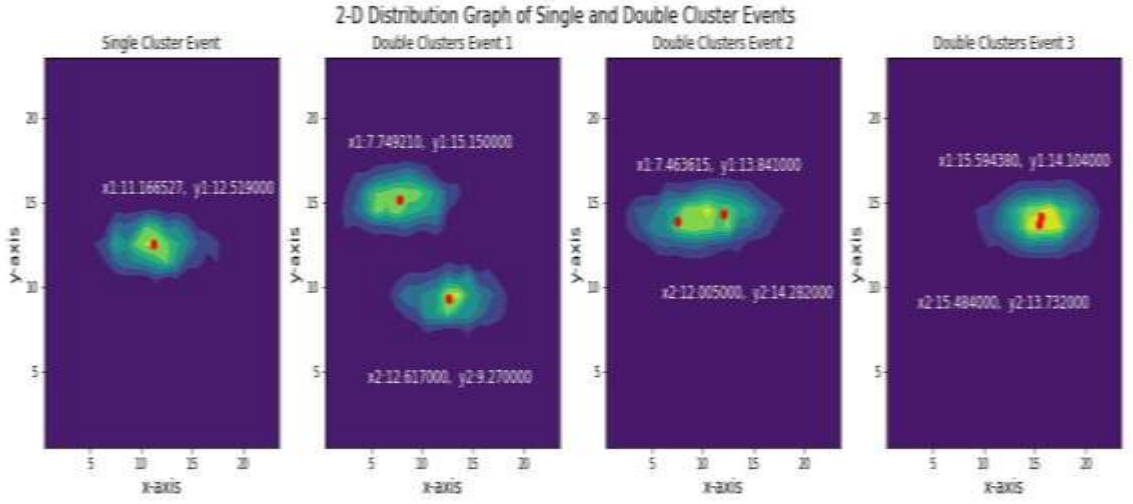

Figure 9. (a). single cluster event, (b) two-cluster event clearly separated, (c)two-cluster event relatively close, (d) two cluster events stacking up.

3.2. Classification Using Moments

The first set of features extracted is still the moments of each event [4]. To replicate the same procedure used in the one-dimensional space, the 2d clusters is projected to the x-axis and y-axis separately, i.e., sum up the height of each column and row, and acquire two 1x24 projections. The same formula (Formula1,1 through 1,4) is applied again to calculate their relative moments. In this case, it yields six parameters (variance x projection, skewness x projection, kurtosis x projection, variance y projection, skewness y projection, kurtosis y projection) compared to the one-dimensional cluster case. After the extraction, the moment information are imported into the MLP classifier [5], the parameters used is as follow: hidden_layer_sizes=(100,100,100), max_iter=1000, alpha=0.0001, solver='adam', verbose=10, random_state=21, tol=0.000000001. After about 52 iterations, the model converges and yields an accuracy of 97.5%. Visualizations of the decision boundary formed with the

DOI: 10.54254/2753-8818/5/20230405

MLP classifier is presented in Figure 10. Since two projections are done before inputting them into the model, two sets of decision boundaries are acquired [5,6].

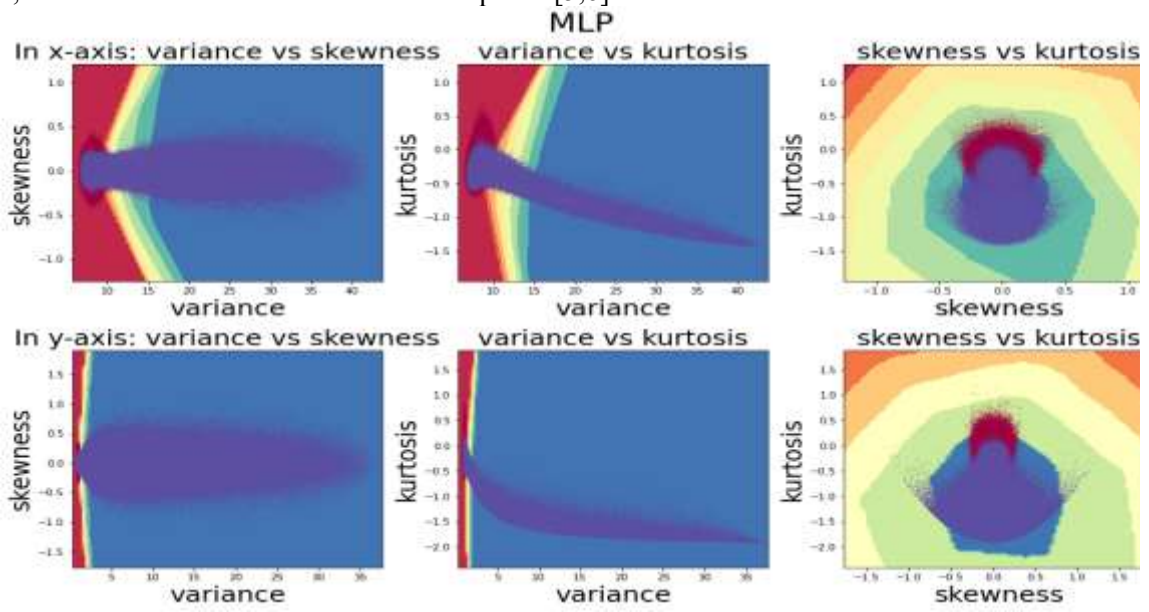

Figure 10. Decision boundary of MLP classifier with an accuracy of 83 in x-axis and an accuracy of 93% in y-axis [5].

3.3. Using MLP Regressor as Basis for Classification

In this part, the true positions of two clusters and differentiate between single-cluster or two-clusters is studied. Ideally, double clusters have two pairs of x-y positions and a significant distance between two clusters while single clusters only have one x-y position. The original 24x24 data set, which has 576 numbers, will make the computation too complicated, so to simplify it into 1d problems and compute positions in x and y separately, numpy.max is used to project the 2d grid onto the x and y axis and generate two 24-bins lists [13].

Before using MLP Regression, a mathematical method (method 1) is tried to find the x and y position [14]. The “center of mass” of two clusters might be located between the centers of two clusters, so an attempt to split two clusters by the “center” is conducted. After splitting, the local maximum positions of each side should be the positions of two clusters. Considering 2/3 mass is near the center of the normal distribution cluster, finding the weighted mean of several bins near the local maximum could compute the position more accurately. This study finds the positions of two clusters in both x and y projections. In a single cluster case, the distance of two found positions should be zero or very tiny. 9 bins near the local maximum are used to compute positions and distances less than square root 2 are considered to be single clusters, in which 82.13% accuracy is yielded in predicting single or double clusters.

For MLP Regression (method 2), the same simplification is used again to turn 2d problems into 1d problems and compute x and y separately [14]. The MLPRegression method from sklearn.neural_network is used to make the train and prediction. Therefore, 24 bins of projection will be the input and 2 positions will be the output. Since the true positions of the second cluster in the single cluster case are zero, the second cluster positions are set to be the same as the first ones. According to the graphs, the clusters are horizontal ellipses, which mean the normal distribution in x and y are different. Two model will be used to train and predict x and y positions separately with parameters

(alpha=0.000001, hidden_layer_sizes=(100,100,100),random_state=27, batch_size = 100,max_iter=300,solver='adam',verbose=10,tol=0.001).

DOI: 10.54254/2753-8818/5/20230405

70% of 1 million data are used as the train set and 30% are used to evaluate the accuracy. In general, x-axis predictions get 0.99360 in R2 score while y get 0.99896. After predicting x and y positions using MLP Regression[14], cases with cluster distance less than square root 0.1 are considered to be a single cluster. At last, this method has 95.2754% accuracy in predicting cluster numbers.

Figure 11. Three heat maps representing the local maximum method(left), 0.1 distance method(middle), and the distribution method(right).

3.4. Classification Using Normalized Signal Distribution

Based on the figure double clusters event 3, there are cases having two clusters located at about the same position. The previous method using Regression to find the position first and determine by distance may not distinguish the two clusters. There are some other features that could be extracted from the original 24x24 grid data in addition to cluster positions, such as three moments. As the neural network is doing well in finding profound relationships, it could predict cluster numbers directly using the original 24x24 grid of Normalized Signal Distribution as input [12]. Rather than 576 inputs, we decide to use numpy.max to get x and y projection and use 24 + 24 numbers as input, which will make training more efficient. While using the cluster numbers (1 or 2) directly as output of training (method 3) The model use the MLPClassifier from sklearn.neural_network with parameters. (alpha=0.000001, hidden_layer_sizes=(100,100,100), random_state=27, batch_size = 100, max_iter=300, solver='adam', verbose=10, tol=0.001) In the same way, we use 70% of one million data as training data and 30% as test data. Generally, this method has 97.632% accuracy on test data.

4. Conclusion

In the overall study, neural network models have better accuracy than decision tree models. Simultaneously, accuracy of models trained from Distribution inputs are relatively higher than models trained from three moments. Neural network is likely to outcrop some potential relation between data and is able to handle more columns of input. MLP Regression focuses on the center locations of distribution. This method could predict locations well and have extremely high R2 scores. However, when two clusters are located too close to each other within almost the same location, MLP Regression will fail to distinguish between one and two clusters [14]. The models trained on three moments are focused on 'shape' of distribution. The accuracy increases as the distance between clusters decreases when the distance is small, which means three moments could help on the classifier if two clusters are close [11]. If combining these two methods together, higher accuracy might be achieved. This study achieved the highest 97% accuracy in classifier single and double clusters using

DOI: 10.54254/2753-8818/5/20230405

the MLP Classifier model directly trained from distributions [5]. MLP Classification is supposed not to focus on location or shape, but the neural network will automatically explore the potential features from distribution to help classification. In addition to accuracy, efficiency also needs to be considered when choosing the model. MLP models take voluminous time to fit as the number of input columns increase. Decision Tree models trained on extracted features like three moments are a better choice if clusters to classify become more complex [9].

In the further research, deep learning using PyTorch or Tensorflow is worth trying to build up more complex neural network models. With the help of a huge number of neurons, the models will be able to solve more complex problems and have higher accuracy. Meanwhile, further study could also try extract more features from distribution such as local maximum for decision tree models, which may improve the accuracy while maintaining high efficiency. After this study on classifying single and double clusters, further studies could consider to explore the resolutions to cases with more cluster numbers and train on data from realistic occasions involving various interferences.

References

[1] Tucci, Salvatore, and Charles H. Sauer. "The tree MVA algorithm." Performance Evaluation 5.3 (1985): 187-196.https://www.sciencedirect.com/science/article/abs/pii/0166531685900124

[2] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." the Journal of machine Learning research 12 (2011): 2825-2830. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf?ref=https://githubhelp.com

[3] Musa, L., et al. "The ALICE TPC front end electronics." 2003 IEEE Nuclear Science Symposium. Conference Record (IEEE Cat. No. 03CH37515). Vol. 5. IEEE, 2003. https://ieeexplore.ieee.org/abstract/document/1352697

[4] Lai, Kin Keung, Lean Yu, and Shouyang Wang. "Mean-variance-skewness-kurtosis-based portfolio optimization." First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS'06). Vol. 2. IEEE, 2006. https://ieeexplore.ieee.org/abstract/document/4673719

[5] Windeatt, Terry. "Accuracy/diversity and ensemble MLP classifier design." IEEE Transactions on Neural Networks 17.5 (2006): 1194-1211. https://ieeexplore.ieee.org/abstract/document/1687930

[6] Suchismita Sahu (2021) “Decision boundary for classifiers: An introduction”https://medium.com/analytics-vidhya/decision-boundary-for-classifiers-an-introduction-cc67c6d3da0e

[7] M. A. Migut, Marcel Worring, Cor J. Veenman (2013) “Visualizing multi-dimensional decision boundaries in 2D”https://www.researchgate.net/publication/271658381_Visualizing_multidimensional_decision_boundaries_in_2D

[8] Biau, Gérard, and Erwan Scornet. "A random forest guided tour." Test 25.2 (2016): 197-227. https://link.springer.com/article/10.1007/s11749-016-0481-7

[9] Myles, Anthony J., et al. "An introduction to decision tree modeling." Journal of Chemometrics: A Journal of the Chemometrics Society 18.6 (2004): 275-285. https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/cem.873

[10] Ontivero-Ortega, Marlis, et al. "Fast Gaussian Naïve Bayes for searchlight classification analysis." Neuroimage 163 (2017): 471-479.https://www.sciencedirect.com/science/article/abs/pii/S1053811917307371

[11] Foody, G. M., and M. K. Arora. "An evaluation of some factors affecting the accuracy of classification by an artificial neural network." International Journal of Remote Sensing 18.4 (1997): 799-810. https://www.tandfonline.com/doi/abs/10.1080/014311697218764

[12] Testud, Jacques, et al. "The concept of ‘normalized’ distribution to describe raindrop spectra: A tool for cloud physics and cloud remote sensing." Journal of Applied Meteorology 40.6

DOI: 10.54254/2753-8818/5/20230405

(2001): 1118-1140. https://journals.ametsoc.org/view/journals/apme/40/6/1520-0450_2001_040_1118_tcondt_2.0.co_2.xml

[13] Shimanaka, Hiroki, Tomoyuki Kajiwara, and Mamoru Komachi. "Ruse: Regressor using sentence embeddings for automatic machine translation evaluation." Proceedings of the Third Conference on Machine Translation: Shared Task Papers. 2018.https://aclanthology.org/W18-6456/

[14] Yilmaz, Işık, and Oguz Kaynar. "Multiple regression, ANN (RBF, MLP) and ANFIS models for prediction of swell potential of clayey soils." Expert systems with applications 38.5 (2011): 5958-5966. https://www.sciencedirect.com/science/article/abs/pii/S0957417410012649