Binning high-dimensional classifier output for HEP analyses through a clustering algorithm

Svenja Diekmann[1,*], *Niclas* Eich[1,**], and *Martin* Erdmann[1]
on behalf of the CMS Collaboration

[1]RWTH Aachen University

Abstract. The usage of Deep Neural Networks (DNNs) as multi-classifiers is widespread in modern HEP analyses. In standard categorisation methods, the high-dimensional output of the DNN is often reduced to a one-dimensional distribution by exclusively passing the information about the highest class score to the statistical inference method. Correlations to other classes are hereby omitted. Moreover, in common statistical inference tools, the classification values need to be binned, which relies on the researcher's expertise and is often non-trivial. To overcome the challenge of binning multiple dimensions and preserving the correlations of the event-related classification information, we perform K-means clustering on the high-dimensional DNN output to create bins without marginalising any axes. We evaluate our method in the context of a simulated cross section measurement at the CMS experiment, showing an increased expected sensitivity over the standard binning approach.

1 Introduction

At the Large Hadron Collider (LHC) near Geneva, the standard model of particle physics is tested by measuring physics processes such as the production of the Higgs boson in different final states and production modes. A major challenge in these searches is separating the vast number of background processes from the desired signal process, especially for processes with a low cross section. To extract the signal successfully, multivariate methods like Boosted Decision Trees and Deep Neural Networks (DNNs) are used, which have become standard tools within the high energy physics community. However, the utilization of DNNs is not straightforward, since many analyses rely on binned likelihood fits that cannot be inferred from a raw network output. For this reason, the most common strategy is using the maximum score of the DNN prediction as the fit-variable, while discarding the remaining values of the outputs nodes. This variable is then binned into a summary statistic with a strategy chosen by the analyst. We present a novel approach to utilize the network's output in a more general approach by performing an unsupervised K-Means-clustering [2][3][4] on an multi-dimensional classifier output. With this simple approach, we determine bins in a high-dimensional vector space, without suffering under the curse of dimensionality or marginalizing over any specific axes. We present the resulting distribution with a visualisation of the 8-dimensional cluster center coordinates as well as an improvement of the 95% Confidence Level (C.L.) limit for a

*e-mail: svenja.diekmann@cern.ch
**e-mail: niclas.steve.eich@cern.ch

Higgsstrahlung analysis at the Compact Muon Solenoid [1] (CMS) experiment. All studies are performed with Monte Carlo (MC) simulation for the 2017 data taking period.

2 Physics Process Classification and Assignment Strategy

A common signal-search analysis approach is to first perform a basic phase-space selection by specifying desired physics objects such as electrons or b-tagged jets and then applying kinematic requirements on transverse momentum, angles between objects or more complex connections. On this reduced number of samples a DNN based classifier is trained to differentiate between the signal process and multiple background processes, being trained on MC-simulation. The network's output is a probability for its confidence of each event belonging to one of the process categories. To perform a likelihood fit on this distribution, each event can only be once in the summary statistics to avoid double counting. This is most commonly done by taking the maximum score of the network output values and aggregating the events in N different histograms. This assignment is straight forward but has the disadvantage that only one dimension of the networks output is utilized and all information of the remaining axes is neglected. While this is not as important for high confidence predictions, especially events that are predicted with equally high probabilities between several processes lose a lot of information in this method.

In this publication, all figures are taken from a vector boson associated Higgs production analysis (VH) performed at the CMS experiment in the $H \to bb$ and $Z \to e^+e^-/\mu^+\mu^-$, $W \to e\nu_e/\mu\nu_\mu$ final states. In the standard approach, the analysis was further split into subcategories, corresponding to the combinations of number of leptons and b-tagged jets. Figure 1 shows the distributions, resulting from the described event assignment in the 1μ and 1 b-tagged jet category in the W + jets and VH process categories. In total, eight classifier categories are used, consisting of the signal (VH), $t\bar{t}$+jets (TT), Drell-Yan (DY), multibosonic (VV(V)), QCD, W+jets and single top (ST) processes.

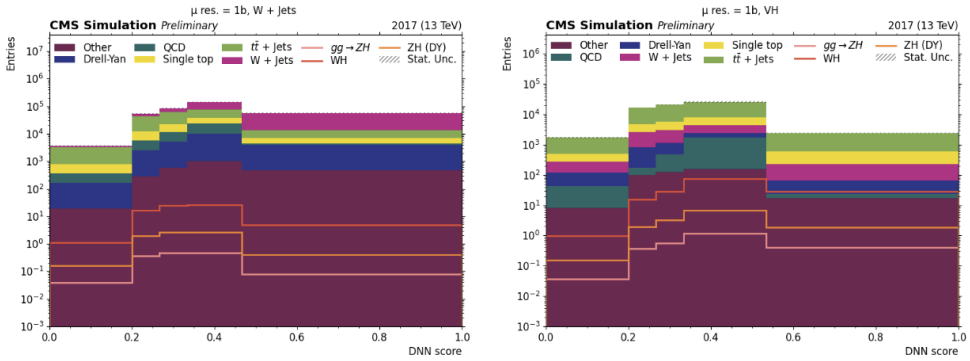

Figure 1. DNN scores for the W+jets and VH categories in the 1 μ + 1 b-tagged jet (resolved, i.e. AK4) category. Each event is assigned to the category of its highest DNN prediction and the resulting histograms are then binned separately.

3 Event clustering as binning algorithm

3.1 Concept

We present a new approach to exploit as much information of the DNN's predictions as possible. Instead of taking the maximum score, we infer the bins directly in the 8-dimensional vector space. Standard grid-like binning approaches in multiple dimensions are not capable of doing that since they suffer under the curse of dimensionality and bin yields would diminish because of low densities. Instead, we deploy a K-Means-clustering algorithm on the classifier output to determine k clusters. These clusters are then utilized to derive a summary statistic for a likelihood fit. K-Means fits this applicaion well, since it runs completely unsupervised and has only the cluster number K as a tuneable hyperparameter.

3.2 Clustering

The K-Means-algorithm is an unsupervised algorithm that is guaranteed to converge to a solution. The algorithm is explained in pseudocode in Algorithm 1 and starts by initializing a set of k starting centers c_j in the given N-dimensions. Next, every point x_i is assigned to its closest cluster center c_j. Then, new centers are determined by taking all the points for each center c_j and calculating their mean positions. This procedure is done for as many iterations until the cluster centers only move below a certain small ϵ or until a set maximum of iterations $m_{\max}$ is reached. In our studies a maximum number of 100 iterations was sufficient.

Data: n events x_i
Result: k bins c_j with n_j events

initialize k cluster centers c_j randomly
for m_{max} *iterations* **do**
  for *all* x_i **do**
    Assign datapoint x_i to closest center c_j by L_2 norm
    $c_j \leftarrow x_i$ with $j = \text{argmin}(\|x_i - c_j\|, \forall j \in k)$
  end
  for *all* c_j **do**
    Compute new cluster centers
    $c_j = \text{mean}(x_i, \forall x_i \in c_j)$
  end
end
All c_j are the final bins with the events $x_i \in c_j$ belonging to them.

Algorithm 1: KMeans clustering as binning

We use the K-Means implementation by scikit-learn [5] in the mini-batch variant. This speeds up the iterative process by using randomly shuffled batches of data-points to determine the centers. In our studies, the batch-variant with a batch-size of 10k points ended up producing comparable results to the full variant but saving on computation time.

4 Results

This section presents the results of applying the clustering algorithm as a binning in the VH-analysis. We further visualise the 8-dimensional coordinates of the high-signal cluster centers and check if a bias is introduced by the MC-dataset. Last, we probe the influence of the analysis sensitivity by computing the upper 95% C.L. limits on the signal strength modifier.

4.1 Distributions

Figure 2 show the result of performing a K-Means clustering on the DNN-predictions with $k = 200$ and $k = 500$ clusters respectively. The cluster index is arbitrary and only depends on the random initialization. Thus, the bins are sorted by absolute signal (VH) yield to improve visual interpretability. The distribution shows several interesting attributes. The bin yields fluctuate by approximately one order of magnitude up and down. There are no bins with fewer than 10^3 expected events for the 200 clusters case and most bins contain between 10^4 and 10^5 events. This is an important property, since bins with very few events can cause numerical problems in performing a likelihood fit. In addition, the signal-curve in red shows a steady rise to the higher cluster numbers with a sharper rise in the top 20 clusters that is also visible in the signal over background ratio $S/\sqrt{B}$. Here, an increased signal to background ratio is observed, gaining sensitivity to the VH signal. The dominating background is the $t\bar{t}$ process, shown in the green color but there exist bins that are not dominated by the $t\bar{t}$ background but contain for example more W + jets or Drell-Yan events. This is an important property, as these bins may help to control these individual backgrounds during a likelihood fit.

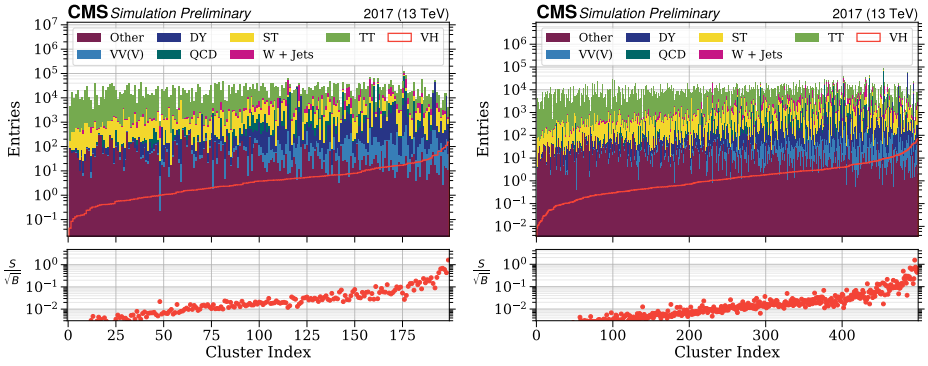

Figure 2. The distributions for the $k = 200$ (left) and $k = 500$ (right) clustering are shown with their signal (S) to square root of background (B) ratio. Both distributions are sorted by total signal yield in the bins for convenience in the visualisation. Both plots show a similar general distribution with the higher cluster count having a higher resolution.

Further, the distributions of $k = 200$ and $k = 500$ show very similar shapes, just with a different resolution. The clustering is not dependent on the random initialization and shows the same behaviour across different clustering numbers k.

4.2 Cluster visualisation

For the common maximum-score approach, the representation of the bins is straightforward. It is known which axis from the 8d output is used and whether the event lies in a high-

(score→ 1) or low-confidence (score→ 0) region. With the clustering approach, the bins are harder to interpet because of their 8d location. As stated, the cluster number is an arbitrary value and the sorting provides some visual convenience. Especially the very high signal yield bins are of interest since these have a major contribution to the analysis sensitivity. The top 25 clusters are displayed in figure 3 for the $k = 200$ and $k = 500$ cases. Despite using more than twice the number of clusters, very similar characteristics are visible, such as a high signal to background ratio bin at the fourth position from the right.

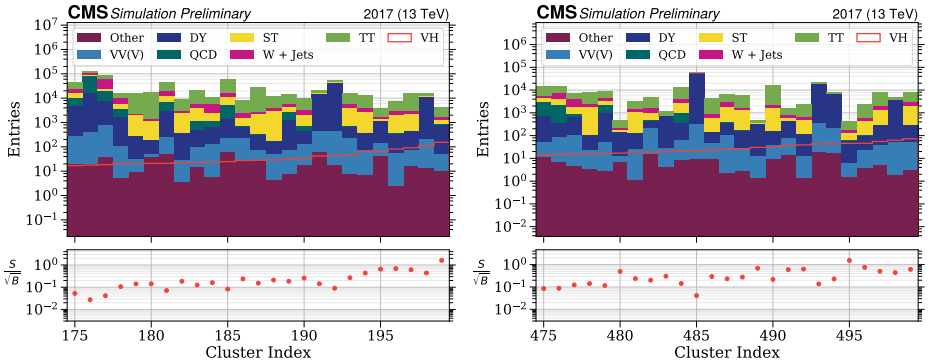

Figure 3. The 25 bins (clusters) with the highest signal yield are shown for the $k = 200$ (left) and $k = 500$ (right) clustering. Similar features are visible in both histograms like a bin with especially high $S/\sqrt{B}$ at fourth position from the right side.

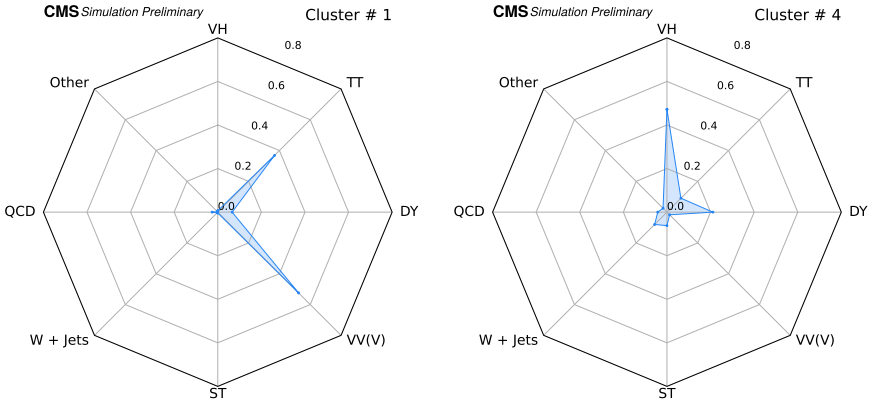

Figure 4. The radar plots are a visualisation of the cluster centers position with the highest (left) and fourth highest (right) signal yield for $k = 200$. The lines from the origin to the outer bound represents an axis in the multidimensional space. Each point then indicates the value of the point on the respective axis. The #1 cluster is located in the 8d space close to the background predictions for the $t\bar{t}$ and multi-boson (VV(V)) backgrounds, whereas the #4 cluster lies in the high confidence signal prediction region.

In order to further understand the clustering, we visualise the cluster centers in the radar plot in figures 4,5 for the $k = 200$ clustering. Here, the eight dimensions correspond to the

eight axes, with the process label at the outer end. In figure 4, the #1 cluster and #4 cluster from the right are shown. The value on the specific axes shows the location in the 8d space of the cluster center. The #1 cluster shows a different composition than the #4 cluster. The #1 cluster has the largest contribution of the $t\bar{t}$ and multi-boson (VV(V)) backgrounds. This can be contradictory to the expectation that the most signal should be in a high confidence DNN-prediction. This cluster represents the events that cannot be classified correctly but are mistaken as $t\bar{t}$ or VV(V). In contrast to this, the #4 cluster center has the highest score in the VH-signal axis with DY being the largest signal contribution. It has the highest $S/\sqrt{B}$ ratio amongst all bins, while having a smaller total event yield than the others. These events correspond to a high signal confidence.

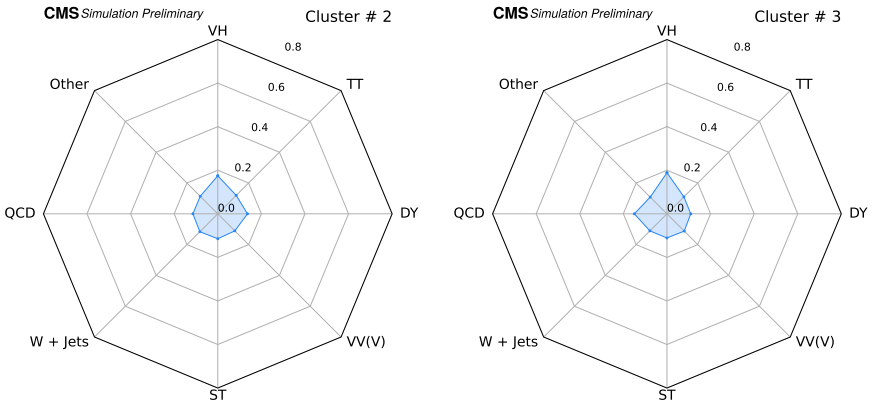

Figure 5. The radar plots are a visualisation of the cluster centers position with the second (left) and third (right) highest signal yield. These cluster centers do not have an as strong confidence prediction as the centers of figure 4 but have a peak in the signal axis as well as the DY/QCD axis.

The #2 and #3 clusters, depicted in figure 5, show a much smaller surface in the radar plot. Only the output scores of the DNN are normalized to 1; this does not need to be the case for the cluster centers, hence the visual difference to the other radar plots. The cluster centers show a peak in the signal VH-process and in the DY/QCD backgrounds axes. This corresponds to events that have a low confidence in the classification and thus small scores for each process.

4.3 Bias Test

Because the binning of the final summary statistic is determined on the used simulated events themselves, one has to check if a bias is introduced by the methodology. To test this, the MC-events are split into a training and test set with each 50% of the data. The clustering is determined on the training set and then evaluated on the training and test set separately. This is benchmarked by computing the expected 95% C.L. limit for the signal strength of the signal process. As uncertainties, the background normalization as well as statistical uncertainties from the event yield and MC-statistics uncertainties are considered. The limits are normalized to one for each training set, since the absolute sensitivity of the analysis is not of interest for benchmarking the methodology.

Figure 6 shows a comparison of the 95% C.L. limits for $k = 100$, $k = 250$ and $k = 500$ clusters. It is apparent that the value of the test set barely deviates from the biased training

set. Especially within the margin of the error bars, one can claim that this method is robust towards being biased to the training set and safe to be used on the MC-simulation.

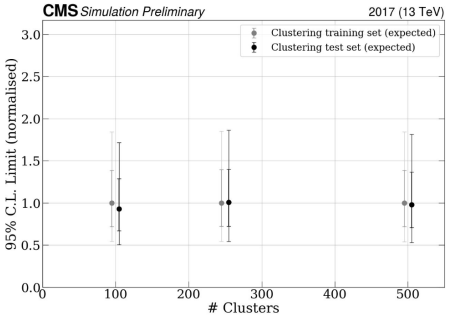

Figure 6. A bias test is performed by splitting the MC events into a training and test set. The clustering is performed on the training set solely and then evaluated on the training and test set separately by computing the upper 95% C.L. limit on the signal strength. The markers show the resulting limit for the two sets with their 1σ and 2σ confidence interval respectively. The test shows no bias towards the training set and thus is not strongly dependent on it.

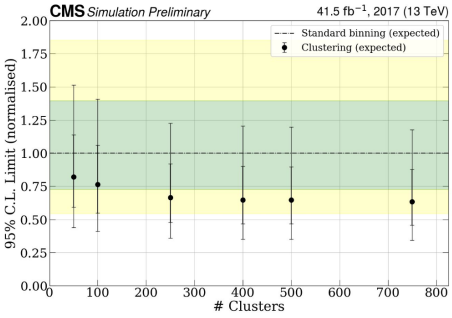

Figure 7. The sensitivity improvement of the clustering is tested by computing the upper 95% C.L. limit for different number of clusters. The results are compared to the standard approach, shown in the dashed line with the 1σ (2σ) confidence level in the green (yellow) band. The clustering improves on the standard binning with an improved limit with increasing clustering number. The improvement saturates around $k = 250$ clusters.

4.4 Limit improvement

An important test for a new algorithm within an analysis is probing its impact on the sensitivity of the analysis. As for the bias test, this is done by computing the expected upper 95% C.L. limits on the signal strength modifier. The limits are computed for different values of the number of clusters k. The baseline is the standard maximum score approach. This standard binning could be improved itself by adjusting the bin sizes but the clustering approach includes no tuneable hyperparameters besides the cluster number k. Thus, to give a basic insight into its capabilities, this serves as a sufficient first comparison. Figure 7 shows the 95% C.L. limits for the different cluster numbers with the 1σ (2σ) intervals respectively, as well as a comparison with the standard approach. It demonstrates that the clustering approach yields a better limit than the standard approach with a decreasing limit up to $\sim$ 250 clusters. The value for the limit then saturates.

5 Conclusion

The use of a Deep Neural Network (DNN) based multi-process classification has become standard for many LHC analyses. However, using the maximum score of the DNN predictions to create a summary statistic does not utilize all available information. In our approach, we utilize the K-Means clustering algorithm to directly determine bins in the high-dimensional DNN output space, omitting the marginalization over any axes. The clustering shows to be robust in a bias-test performed on MC events by performing a training and test split. The visualisation by radar plots gives an insight into the clusters with very high signal yield and helps developing more understanding for this method. Last, the upper 95% C.L.

limits show an improvement of $\sim 60\%$ with respect to the standard binning approach, where the limit saturates at $k = 250$ clusters.

Acknowledgements

This research was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant 400140256 - GRK 2497: The physics of the heaviest particles at the LHC. This work is also supported by the Ministry of Innovation, Science, and Research of the State of North Rhine-Westphalia.

References

[1] The CMS collaboration, 2008 The CMS Experiment at the CERN LHC, *JINST*, **volume 3**, S08004

[2] MacQueen, J. 1967 Some methods for classification and analysis of multivariate observations *Proc. of the 5th Berkeley Symp. on Mathematical Statistics and Probability, Davis, Ca, USA, June 21 – July 18, 1965 and Dec. 27, 1965 – Jan. 7, 1966 University of California Press*, **volume 5.1**, 281–97

[3] Lloyd, S. 1982 Least squares quantization in PCM, *IEEE Transactions on Information Theory*, **volume 28**, 129–37

[4] Sirunyan, A.M. *et al.* (CMS collaboration) 2018 Evidence for associated production of a Higgs boson with a top quark pair in final states with electrons, muons, and hadronically decaying τ leptons at $\sqrt{s} = 13$ TeV, *J. High Energy Phys.* JHEP08(2018)066

[5] Pedregosa *et. al.*, E., Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*, **volume 12**, 2011

[6] Svenja Diekmann, Niclas Eich, Martin Erdmann, 2022 Binning high-dimensional classifier output for HEP analyses through a clustering algorithm, *J. Phys.: Conf. Ser.*, (submitted for publication)

[7] The CMS collaboration, 2023 Binning high-dimensional classifier output for HEP analyses through a clustering algorithm, *Detector Performance Summary*, DP-2023/074