# Event classification of black hole and sphaleron events with machine learning

Jonas Halvorsen, Håvard Haug
University of Bergen

## Abstract

The task of recognizing event types from the output of Compact Muon Solenoid detector [1,2] at the Large Hadron Collider [3] requires intense computation, as the span of event types is large and the objects that belong to different events can be quite similar. To investigate how machine learning can be used in aiding event recognition, a data pipeline which transforms labeled event data into images was built, with parameters deduced by performing analysis of multiple simulated event datasets. The output of this data pipeline was then used to train a convolutional neural network in the task of classification between the events containing: Sphalerons [4], Microscopic Black Holes [5], and Standard Model processes. Using transfer learning on a pretrained Resnet model [6], several new models were built based on various hypotheses for what could produce the most accurate result on real events from the CMS detector. For the labeled input data a large set of events were generated using simulator software Pythia8 [7] and Delphes [8], with parameters which are expected to yield the same results in the detector as when the real event occurs. Confusion matrices were created for the trained models, and a classification accuracy of (86.8±0.6)% was achieved in one of the models which was trained to recognize a plethora of different Black Hole types. The code, models, datasets and output images produced during this project have been uploaded and made publicly available for further use, and could form a good foundation for exploring a general classification pipeline for Compact Muon Solenoid detector data in LHC Olympics [9,10] format.

# 1 Introduction

Particle collider experiments such as those performed in the Large Hadron Collider, yield incredible amounts of exciting data, which are then processed and analyzed to aid in our understanding of the laws of physics. ATLAS [11,12] and Compact Muon Solenoid (CMS) are multipurpose detectors at the Large Hadron Collider designed to see a wide spectrum of possible new phenomena, as well as to perform precise measurements of Standard Model physics. To use all this data, different approaches are taken, and an approach recently gaining in popularity is to apply machine learning methods on the data to help identify if the experimental work matches the theoretical.

As part of the project "**Understanding the Early Universe: interplay of theory and collider experiments**" [13] which aims to expand our knowledge of the early Universe using knowledge gained from collider experiments, and to explore physics beyond the Standard Model, the following task has been defined: "**Investigating the sphaleron and mini-black hole production at the LHC and its dependence on the mechanism of the electroweak phase transition**". This task is further split into three subtasks, where the first task is to determine what the characteristics of Sphaleron and Black Hole events are, and use the knowledge gained from this to create sensible parameters for processing event data and training a machine learning model. This can further be used to estimate what collider energies and luminosities are necessary to classify Sphaleron and black-hole events. This project is centered around the first of these subtasks, where the end goal is to create a trained multiclass machine learning model with as high an accuracy as possible for classifying events into the classes Sphaleron, Black Hole, ttbar, and ttbar with large transverse momentum jets.

# 2 Problem/Background

The Standard Model describes well three of the fundamental interactions, although it leaves some questions. Gravity is a much weaker force than the rest of the fundamental interactions. The large difference in forces is what is called the hierarchy problem. The Standard Model does not explain the abundance of matter as opposed to antimatter in the Universe, known as the matter antimatter asymmetry. Many solutions to these problems among others have been proposed, also known as Beyond Standard Models.

Attempts at solving the hierarchy problem have been made, some by suggesting the presence of extra dimensions at small scales, in which only gravity propagates [14]. The models suggest that the gravitational force is fundamentally stronger, but the extra space it acts in, diminishes its apparent strength, and results in the apparent strength we experience. A consequence of the extra dimensions the theory suggests, are microscopic Black Holes. Therefore a discovery of microscopic Black Holes could help explain the hierarchy problem.

The matter-antimatter asymmetry in the Universe, suggests the existence of processes breaking the conservation laws of baryon and lepton number. In electroweak baryogenesis, a possible nonperturbative quantum process called a Sphaleron, could change the baryon and lepton number, while keeping $\Delta(B - L) = 0$, breaking the symmetry. [15]. There have been discussions whether the Sphalerons could be produced in high energy colliders for decades[16].

Both Sphalerons and microscopic Black Holes are expected to decay to a final state of high multiplicity. Specifically, multi-jet final states are characteristic for a "wide class of beyond Standard Models." [13, 15]. Distinguishing between them could be difficult. Their signatures are similar, but their origins are different, as Black Holes will evaporate and create Standard Model particles according to the thermal blackbody radiation [15]. If these differences are significant, they could be differentiated by machine learning, which could prove to be a valuable tool in analysing high energy collisions.

# 3 Equipment/Software

To perform this project, the programming language Python [17] was chosen as it offers many readily available libraries for machine learning and data analysis with a low barrier of entry to start using these. As coding was performed as a team in python, Google Colab [18] was used to be able to easily perform prototyping and collaborative coding, without the initial need for setup of a dedicated machine to run the code. As the code reached a more mature stage, a dedicated machine with a modern graphics card (Nvidia RTX 2080 ti) was taken into use as a backend for the Google Colab document. Google colab integrates with jupyter backends [19], so a jupyter server was set up on a machine owned by the ATLAS project at the Department of Physics and Technology, and through an ssh tunnel was made available to Colab for running the training of the models with deeper nets and more epochs. Although Google Colab with hosted runtime in the free tier had GPUs available, they were limited in performance, and the availability of these GPUs was unpredictable, so therefore a local runtime was used.

In the start of the project, fastai was used for prototyping a machine learning pipeline, but due to certain limitations in flexibility of the api, PyTorch was settled on as the framework to use for performing machine learning tasks. As PyTorch offered pretrained models which could be used for transfer learning, it was possible to just download exactly the network structure with some default weights that was desired. To determine how the input data should be processed for usage with the machine learning model, numpy and matplotlib were used for performing data analysis and plotting.

# 4 Procedure

## 4.1 First look at the data

Table 1. Black Hole datasets. Center-of-mass energy 14 [TeV], Planck scale at 3910 [GeV]. Number of extra dimensions and the minimum mass of Black Holes. Data generated by Rafal Maselek

| Nr | Dataset | Extra dimensions | Minimum mass of Black Hole [TeV] |
|---|---|---|---|
| 1 | BlackMaxOutputFirstRun.lhco | | |
| 2 | BlackMaxOutputFirstRun1.lhco | | |
| 3 | MH_n4_M8.lhco | 4 | 8 |
| 4 | MH_n4_M9.lhco | 4 | 9 |
| 5 | MH_n4_M10.lhco | 4 | 10 |
| 6 | MH_n4_M11.lhco | 4 | 11 |
| 7 | MH_n4_M12.lhco | 4 | 12 |
| 8 | MH_n5_M8.lhco | 5 | 8 |
| 9 | MH_n5_M9.lhco | 5 | 9 |
| 10 | MH_n5_M10.lhco | 5 | 10 |
| 11 | MH_n5_M11.lhco | 5 | 11 |
| 12 | MH_n5_M12.lhco | 5 | 12 |
| 13 | MH_n6_M8.lhco | 6 | 8 |
| 14 | MH_n6_M9.lhco | 6 | 9 |
| 15 | MH_n6_M10.lhco | 6 | 10 |
| 16 | MH_n6_M11.lhco | 6 | 11 |
| 17 | MH_n6_M12.lhco | 6 | 12 |

Table 2. Sphaleron datasets center-of-mass energy 13[GeV]. Dataset Nr. 1, had much higher multiplicities than the Black Hole datasets (fig 1). So the Sphaleron datasets 2 and 3 were generated. Sphaleron dataset 3 was decided as best suited for machine learning purposes, as the multiplicities better matched those of the Black Holes. Data generated by Kazuki Sakurai, Andreas Papaefstathiou.

| Nr | Dataset |
|---|---|
| 1 | PP13-Sphaleron-THR9-FRZ15-NB33-71-NSUBP5.lhco |
| 2 | PP13-Sphaleron-THR9-FRZ15-NB33-60-NSUBP50.lhco |
| 3 | PP13-Sphaleron-THR9-FRZ15-NB0-NSUBPALL.lhco |

Table 3. ttbar events. Nr 2, ttbar_largejets.lhco has at least one jet with $p_T \geq 200[GeV]$. Data generated by Nikolai Fomin.

| Nr | Dataset |
|---|---|
| 1 | ttbar.lhco |
| 2 | ttbar_largejets.lhco |

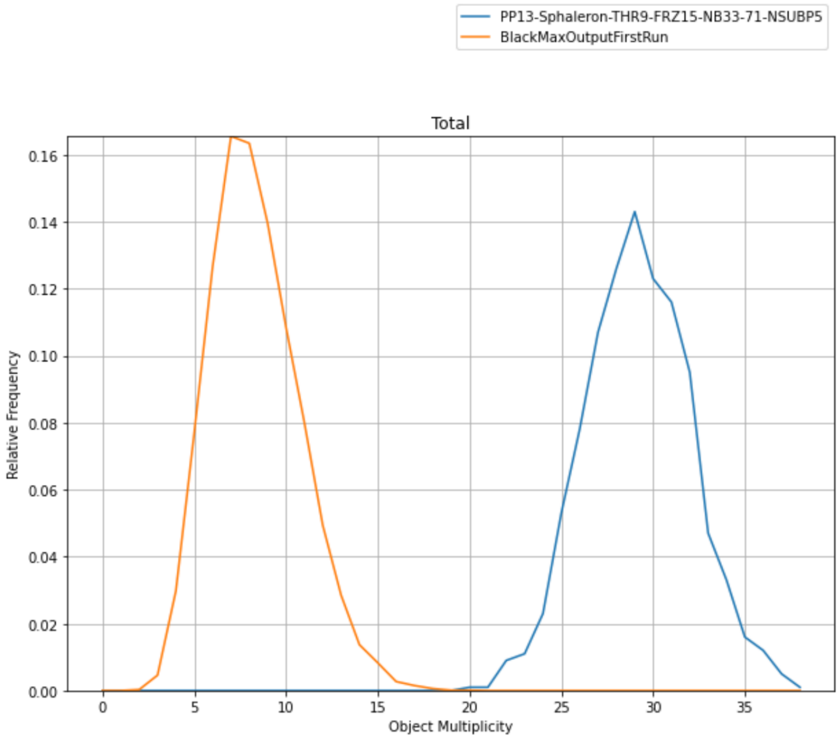


Figure 1. Figure shows the frequencies of multiplicities of all objects[1] in an event. The x-axis gives the total number of objects counted in an event, which can only be integers. The relative frequency is determined by the number of events in the file with the given multiplicity. The figure shows the large difference in the multiplicities of Black Hole file 1 (table 1) and Sphaleron file 1 (table 2).

[1] By objects we refer to the photons, electrons, muons, taus, jets, and missing transverse energy, which are represented by the numbers 0, 1, 2, 3, 4, and 6 respectively.

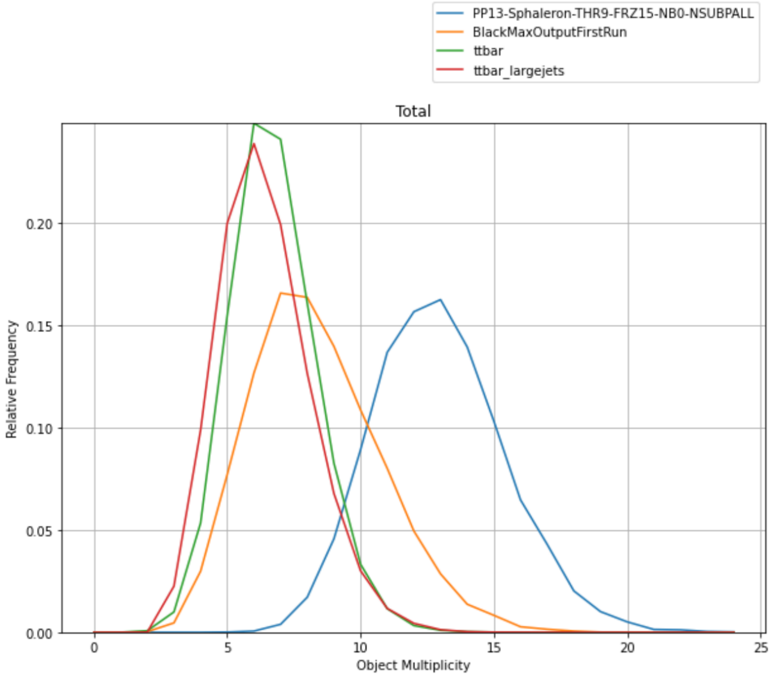


Figure 2. The relative frequency of object multiplicities. The overlap between the blue (Sphaleron model 3, used for machine learning), and orange BlackMaxOutputFirstRun, is greater than for. Green and red are background processes $t\bar{t}$, the latter with larger jet momentum.

The Black Hole files were generally similar with respect to object multiplicities, only a notable difference between the Black Hole files 1,2 and 3-17 (table 1), as seen in figure 3.

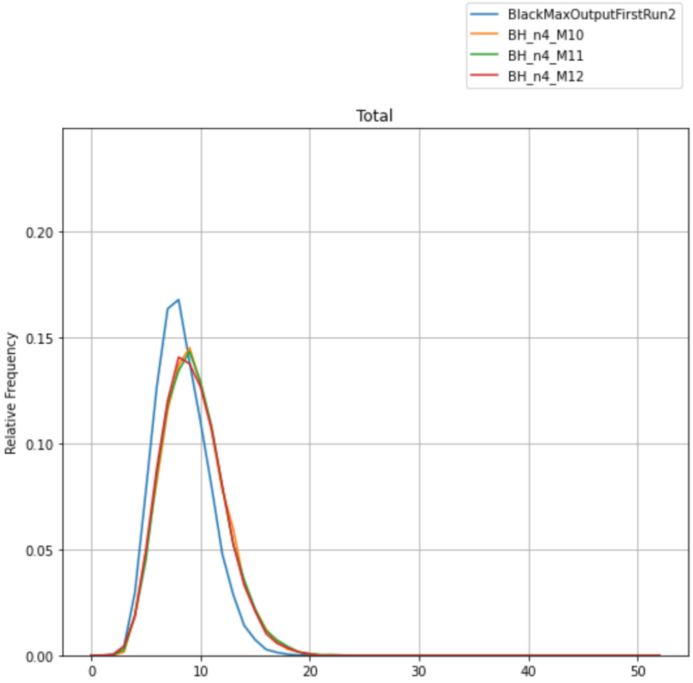


Figure 3. Object multiplicities of some Black Hole files. Blue shows one of the first two Black Hole files produced. The rest are among the files produced at a later date.

The scalar sum of transverse momentum from objects in an event are increasing with higher mass of the Black Hole. See figure 4. There is not any such visible trend with respect to increasing dimensions, see figure 5.

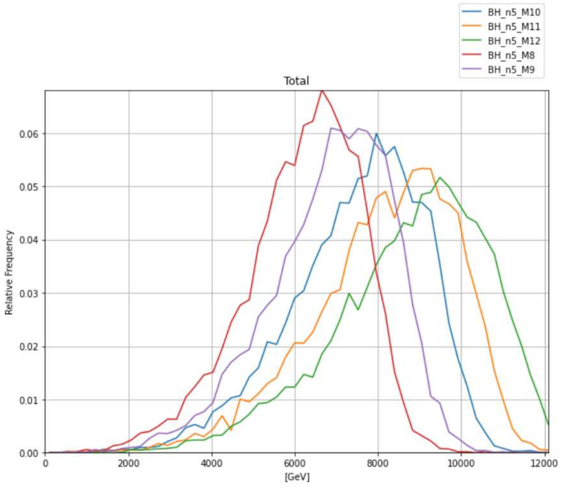


Figure 4. Relative frequency of total momentum. That is the scalar sum of transverse momentum of all objects in an event. The figure shows a trend towards higher momentum of events with higher Black Hole mass.

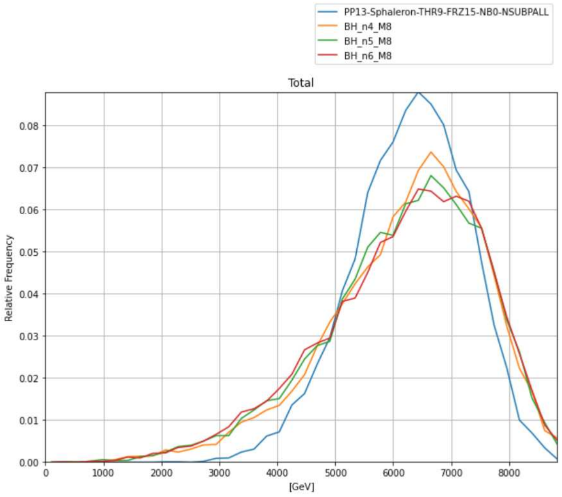


Figure 5. Relative frequency of total momentum, blue graph is from the Sphaleron dataset used in machine learning. The rest of the graphs are the lowest mass 8[TeV] Black Holes.

Figure 6 and 7 shows relative frequencies of transverse momentum of jets, i.e., the transverse momentum of any jet. There is little difference between the Black Hole files. The jets from Sphalerons have slightly less energy. The less energy of the Sphalerons is a recurring theme across all object types, which could make sense as there are generally more objects in the Sphalerons.

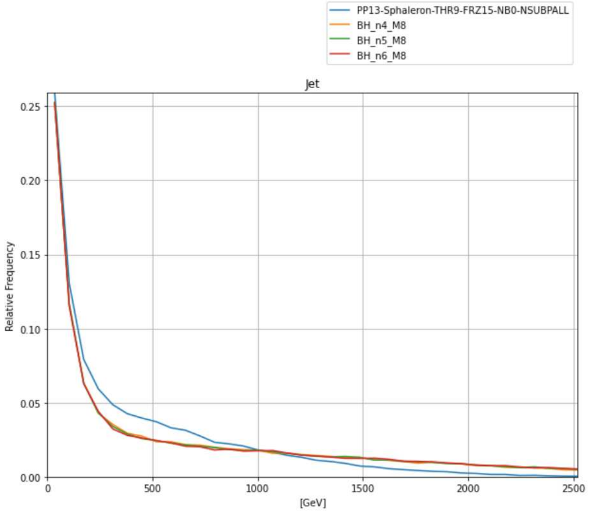


Figure 6. Relative frequency of $p_T$ of any jet in the Corresponding dataset. Blue is the Sphalerons used in machine learning. The rest are the Black Holes with the lowest mass 8[TeV].

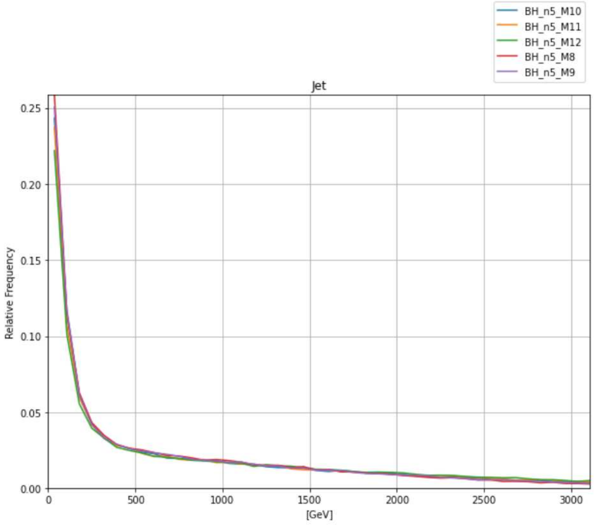


Figure 7. Relative frequency of $p_T$ of any jet in the corresponding dataset. The graphs are all Black Holes with n5 dimensions.

Figure 8 shows a higher missing transverse momentum (MTE) for Sphalerons than for the Black Holes. The MTE of the Black Holes (fig 9), does increase with increasing Black Hole mass.

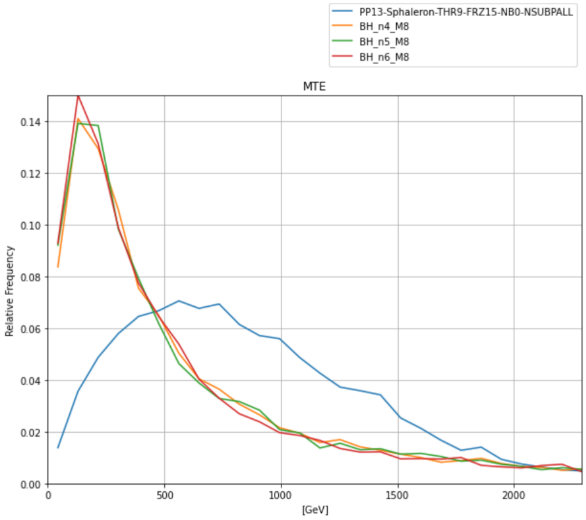


Figure 8. Relative frequency of MTE in the corresponding dataset. Blue is the Sphalerons used in machine learning. The rest are the Black Holes with the lowest mass 8[TeV].

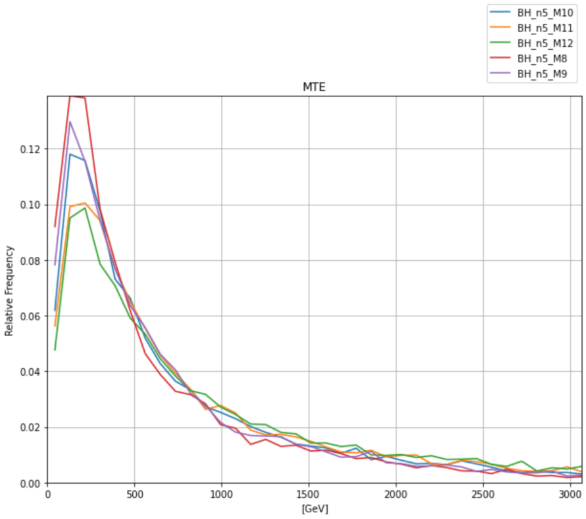


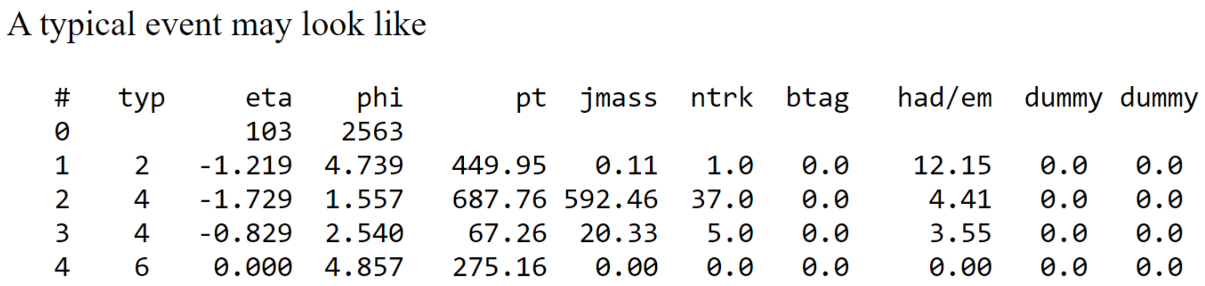
Figure 9. Relative frequency of MTE in the corresponding dataset. The graphs are all Black Holes with n5 dimensions.

The datasets were searched for Z bosons. The mass reconstruction of Z bosons showed decay to electrons as around twice the decay to muons. Z bosons decay as much to electrons as muons, which means mistakes in either the program for mass reconstruction, or the datasets. A reconstruction of momentum of the Z bosons were made, and no significant difference between electrons and muons were present. As such, the fault is most likely in the mass reconstruction.

## 4.2 Data pipeline

The data was in LHC Olympics (lhco) format[10], a reader for this format was developed with a simple class to store the data as it was read. The reader was partially inspired by the lhco_reader by Fowlie[20] but as that reader was made for python 2, and this project was made in python 3, a new one had to be developed. The reader that was made for this project has a much more limited scope than the general lhco_reader by Fowlie, as it was made in response to the issues that arose during the development of the data pipeline.

The lhco data format consists of a file with event data, where one file usually contains thousands of events. The structure of the lhco data format is described at [10] and as an example, the following typical event shown in figure 10.



A typical event may look like

```
#   typ     eta    phi       pt  jmass  ntrk  btag   had/em  dummy dummy
0           103   2563
1    2   -1.219  4.739   449.95   0.11   1.0   0.0    12.15   0.0   0.0
2    4   -1.729  1.557   687.76 592.46  37.0   0.0     4.41   0.0   0.0
3    4   -0.829  2.540    67.26  20.33   5.0   0.0     3.55   0.0   0.0
4    6    0.000  4.857   275.16   0.00   0.0   0.0     0.00   0.0   0.0
```

Figure 10: data with example objects for a single event.

Each event is the result from a proton-proton collision, with a list of observed objects and their properties, as reconstructed in the CMS detector The most relevant data for this project from each event was the "typ" column, which describes which type of object[2] is presented in the row. In addition the eta and phi columns which describe the pseudorapidity and azimuthal angle of the objects were crucial in the data pipeline. The momentum of objects is also described in this format.

The neural network of type ResNet was chosen for the recognition task, as it is readily available as a pretrained model, is fast to train and has been successfully used on CMS tasks by other teams[21]. To be able to use the pretrained ResNet model and apply transfer learning to adjust the weights of the model, the input data needs to be in the same format as the ResNet was originally trained with. This entails an expectation of an RGB image of a specific size. Therefore the output of the data pipeline needs to be an RGB image, and a sensible size has to be chosen so that it can be scaled to the size that the neural network expects.
Mapping the event data to an image was done with basis on some of the knowledge gained from[4] which lead to an algorithm which translates the eta and phi values of an object into a cartesian integer (x, y) coordinate system, and an energy calculation which adds the object's energy to the calculated (x, y)-coordinate. A subset of the object types were considered, as the mapping to RGB is performed through first creating a 2D image per object type in an event, and then designated the images to different colour channels. Muons, Electrons and Jets were chosen to be generated images for. As the number of objects per event was so low, a decision was made to use a low resolution image of 32 by 32 pixels, where each pixel has a value in the range [0, 255], which corresponds to a standard 8-bit colour channel. As the neural network has an expectation of minimum image size 224 x 224, a resizing step was used when training the network. The mapping from Eta and Phi to (x, y) coordinates was done by empirical knowledge gained from plotting the Eta and Phi distribution of objects, and understanding of how the CMS detector works. From figure 11 the possible values for eta can be seen, and the majority of objects are shown to be within [-2.5, 2.5], which was chosen to be the cutoff points for eta values used in image generation. This means that objects with eta outside of this value range are not included in training the neural network, or when converting future events into images for evaluation by the neural network. Looking at the probability distribution of eta, 2*$\sigma$ was found to cover the range of eta from appx [-2.1, 2.1], thus over 95% of all objects will be included in the final image with the chosen cutoff. Since the values of phi go from [-$\pi$, $\pi$] as seen in figure 12 a rounded value of [-3.14, 3.14] was used as the cutoff there.

[2] By objects we refer to the photons, electrons, muons, taus, jets, and missing transverse energy, which are represented by the numbers 0, 1, 2, 3, 4, and 6 respectively.

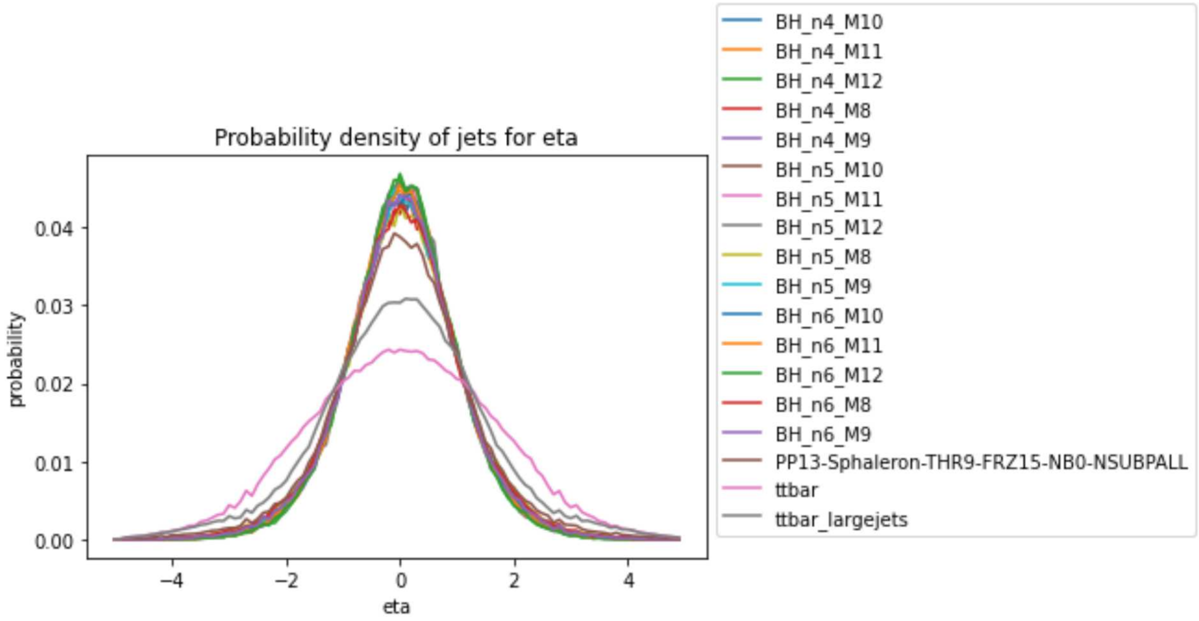


Figure 11: Eta values with distributions for the different data sets

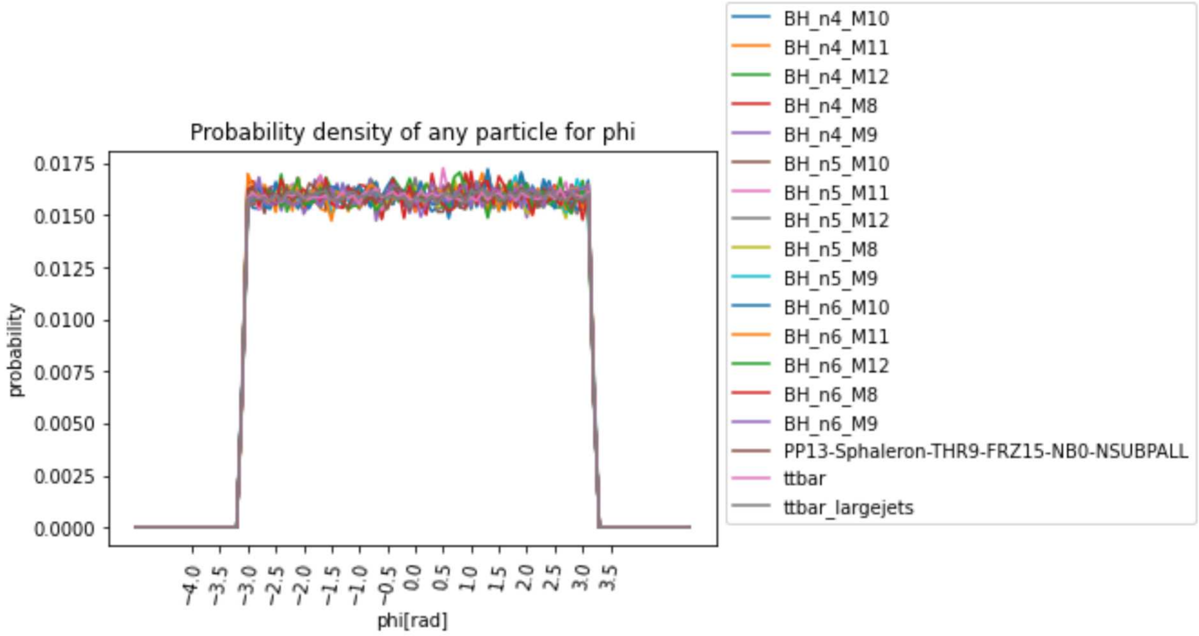


Figure 12: Phi values with distribution for the data sets

For a given object the energy calculation is done by formula 1.

$$\theta = 2 * atan(e^{-\eta})$$
$$P = p_T / sin(\theta)$$
$$E = \sqrt{P^2 + m^2} \quad (1)$$

Jet mass is taken from LHCO files

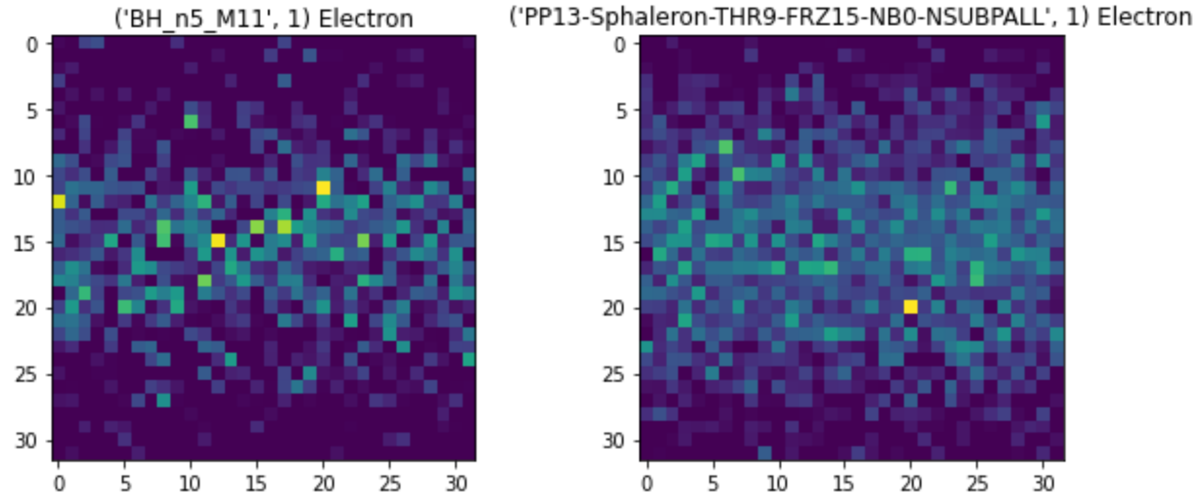
Where η is the pseudorapidity, $p_T$ is the absolute momentum, and m is the invariant mass. The actual calculations of the energy was mistakenly done with $m * 2$ in the root instead of $m ** 2$.

The final algorithm to convert events into images is described by the following pseudocode:

```
For each event:
    Create empty map to save pixel values for each event type
    For each object in event:
        Keep objects with eta within [-2.5, 2.5] and phi within [-3.14, 3.14]
        Linearly scale eta and phi to range [0, 31]
        Calculate the energy E of the object with formula 1
        Let map(scaledEta, scaledPhi) += E
```

The highest energy pixel for each object type in each dataset is tracked during the execution of the algorithm, and is then used to normalize the images so each pixel has a maximum value of 255.
In the created images, eta is mapped to the Y-axis, and phi to the X-axis.

To evaluate the energy distributions in the event images, a combined image summing all the energies for each object type in each dataset was created. A few examples can be seen in figure 13 where Sphalerons can be seen to generally have energies in a wider range of eta.


('BH_n5_M11', 1) Electron

('PP13-Sphaleron-THR9-FRZ15-NB0-NSUBPALL', 1) Electron

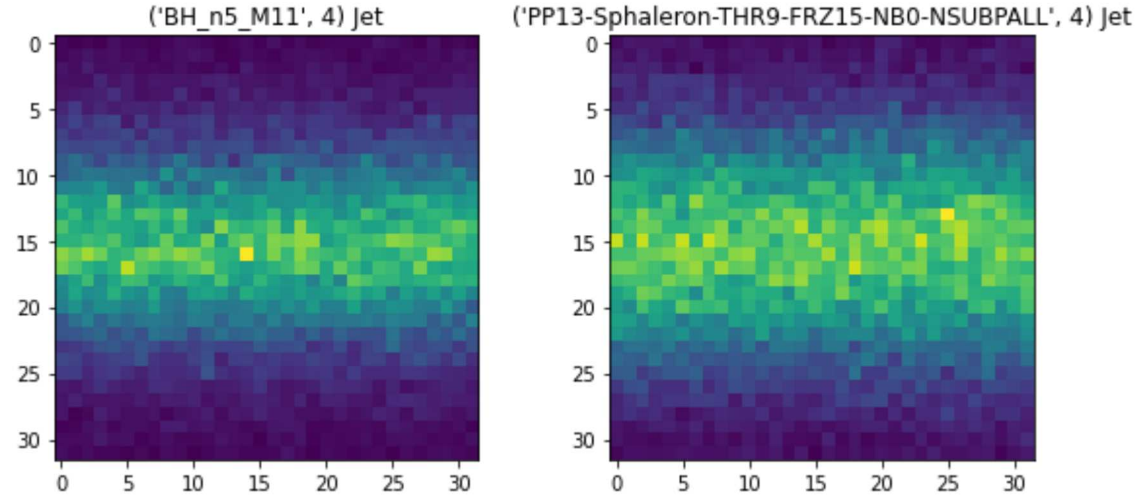


Figure 13: Energy distributions for electrons and jets in a specific Black Hole dataset and the Sphaleron dataset.

As a final step of the data pipeline, the pixel map representations of the Electron, Muon and Jets images are merged, and all pixel positions without a value receive a zero value. This format is then saved as a .png file using pythons Pillow library. To create balance in the data sets, a fixed value for how many images to output from each dataset is decided, and in this case was chosen to be 9000 randomly selected images. Subfolders for training data and validation data were created, and of the 9000 images, 20% were randomly selected to go into the validation dataset, and the remaining 80% as training data.

After outputting the images to the file system, another folder was created in test and training which was a randomly sampled subset of all the different Black Hole files with equal amounts of images as the other folders contain. This folder was used for training a general Black Hole recognizer.

## 4.3 Machine learning

As mentioned, a pretrained Resnet was used and the steps to training the model were set up based on pytorch's tutorial for transfer learning [22]. The images were loaded into a training and validation set, based on the folder structure in the file system. A custom image loader was built which allows for filtering on subdirectories, so that a subset of folders can be selected for training at a time, in opposition to the default Imagefolder loader, which creates a class for each subfolder found. The data transformations seen in Appendix A was applied to the images as they were loaded, this transforms the images to the appropriate size for the neural network and a predefined set of normalization values are applied which are required by the resnet.

The Resnet pretrained resnet model is then loaded and the number of classes that the network can output is set to the desired amount. The loss function used was simply selected to be cross entropy loss (formula 2) as this was the one used in the tutorial, and it was not clearly obvious what the advantages and disadvantages of the different loss functions would be on this dataset.

$$\text{Loss} = -\sum_{c=1}^{M} y_{o,c} \log(p_{o,c}) \qquad (2)$$

Where M is the number of classes, y is a binary indicator of whether class label c is the correct classification for observation o, and p is the predicted probability observation o is of class c.

The optimizer used is SGD(Stochastic Gradient Descent) which is guided towards the minima by the loss function results. Again this is used as no specific knowledge of any advantages and disadvantages of other optimizers on this data set are known.
The models were then trained with the different datasets. In total, 16 trained models were created. Every model was trained with Sphalerons, ttbar, and ttbar large jets as possible output classes, in addition one of the Black Hole datasets were added to each of the models. This was also done with the combined Black hole dataset. Thus 15 of the models were trained in recognizing a specific Black Hole type, while the last model was trained to recognize all types of Black Holes equally. The number of  The models were then stored, and confusion matrices were produced for all the models on both the corresponding validation sets for the classes they were trained on, and evaluated with the inclusion of all the other Black Hole datasets to see how well they would recognize the classes they were not trained on.

The accuracy of the values in the confusion matrices are calculated using $s(N) = \sqrt{N}$ where N is the number of events of that event type in the validation set.

# 5 Results

The resulting trained models gave varying accuracies when running on the validation data sets, but the general accuracy was far above a random selection. Two models stood out the most, the first was the one trained on the sampled Black Hole set, as it had an (86.8±0.6)% accuracy when running on the validation sets, as seen in figure 14 and (81.3±0.6)% accuracy when running on all the other classes of Black Holes seen in figure 15.

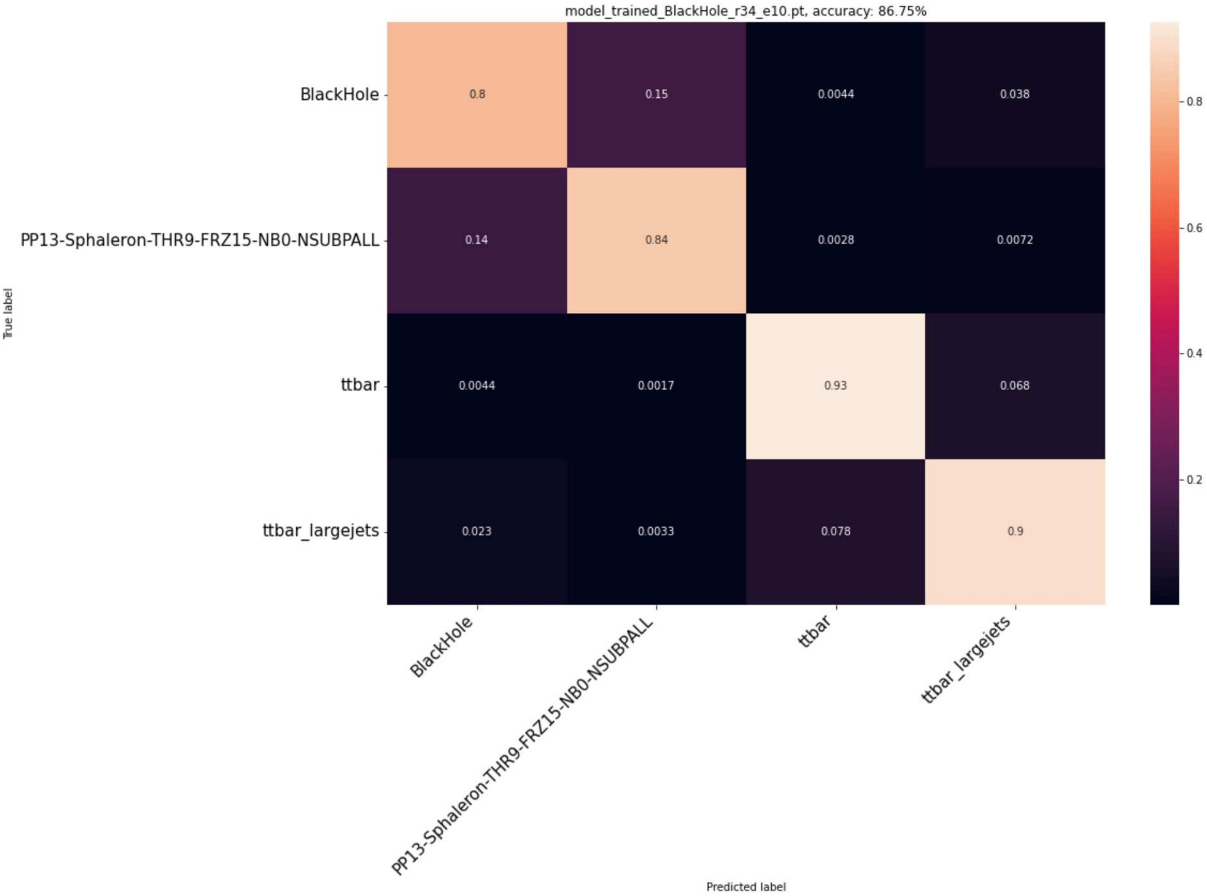


Figure 14. Confusion matrix of the model trained with all of the Black Hole types, run on the validation set made up of events of the same class. The y-axis gives the true labels, i.e., the actual labels of the event which is to be predicted. The x-axis gives the label predicted for the event.

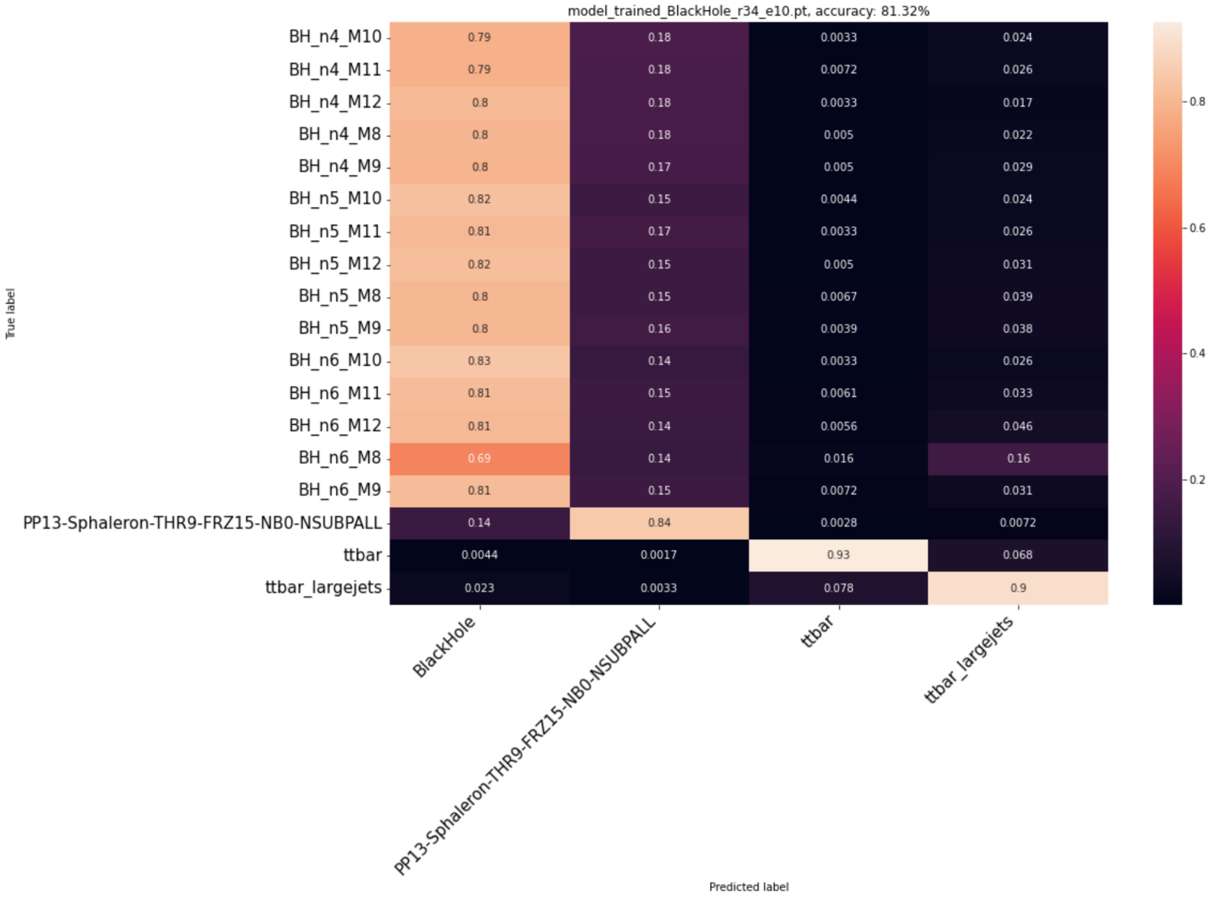


Figure 15. Confusion matrix of the model trained with all the Black Hole types, validated on the set created from all datasets.

The other was the model trained on BH_n6_M8 which had an accuracy of (90±1)% when running on the evaluation sets for itself and the three other datasets as seen in figure 16. It could clearly distinguish that class of Sphalerons from Black Holes, but when ran on the other Black Hole datasets which it was not trained on, it struggled to differentiate these from Sphalerons as can be seen in figure 17.

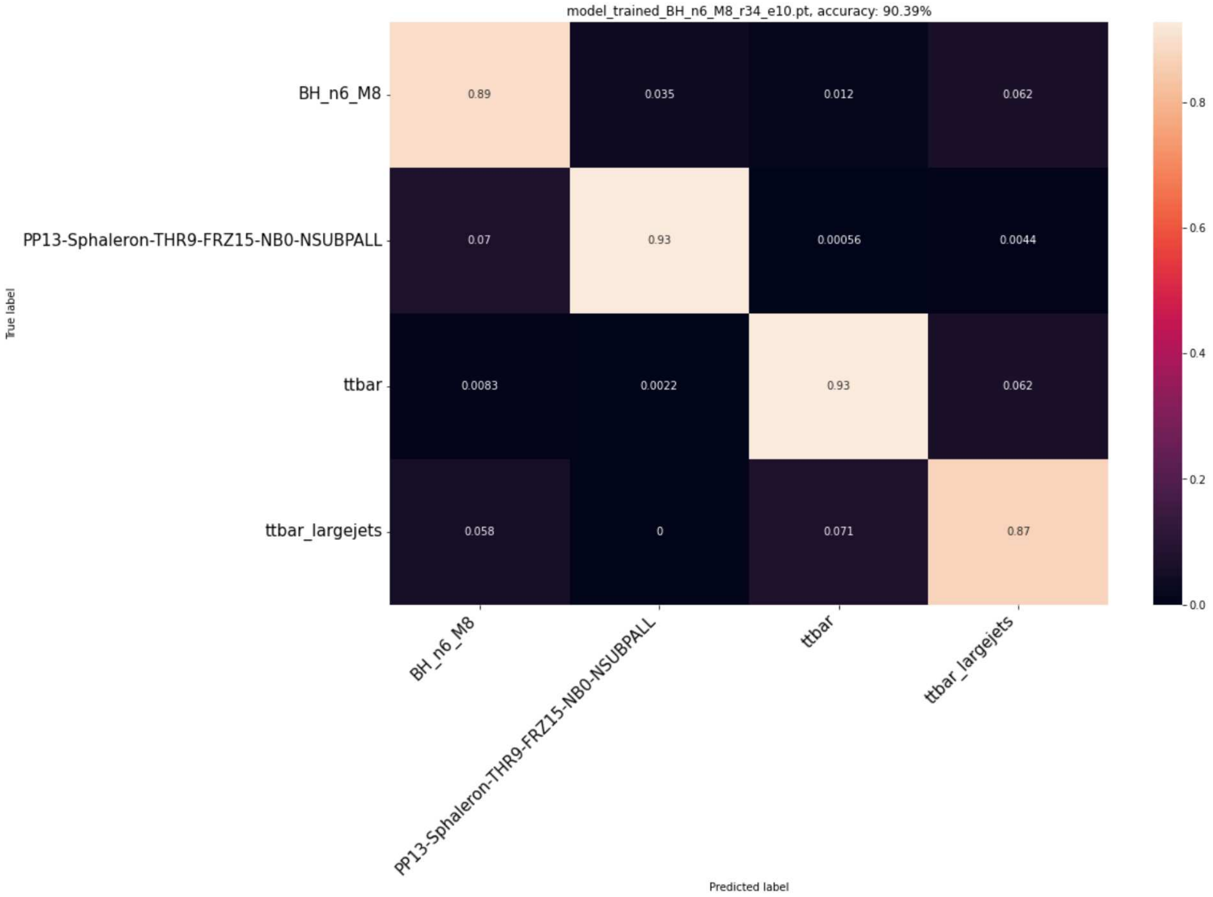


Figure 16. Confusion matrix of model trained with the BH_n6_M8 training set. Validated on its own validation set.

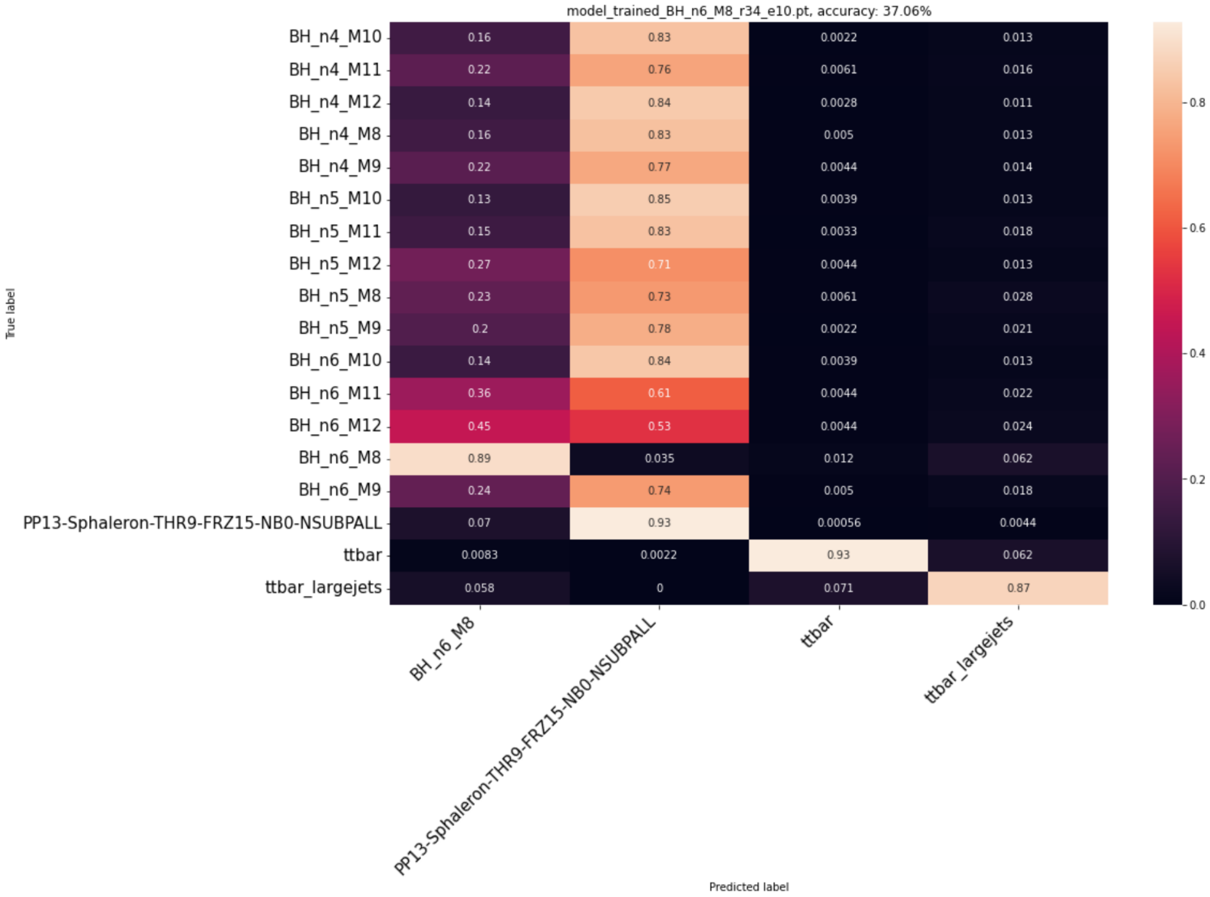


Figure 17. Confusion matrix of model trained with the BH_n6_M8 training set. Validated on the validation set sampled from all Black Hole datasets.

Of the models trained on a single Black Hole dataset, the best result when evaluating on all Black Hole types was achieved by the BH_n6_M9 class, where it could classify the other Black Holes, Sphalerons and ttbar sets with an (84±2)% accuracy as seen in figure 18.

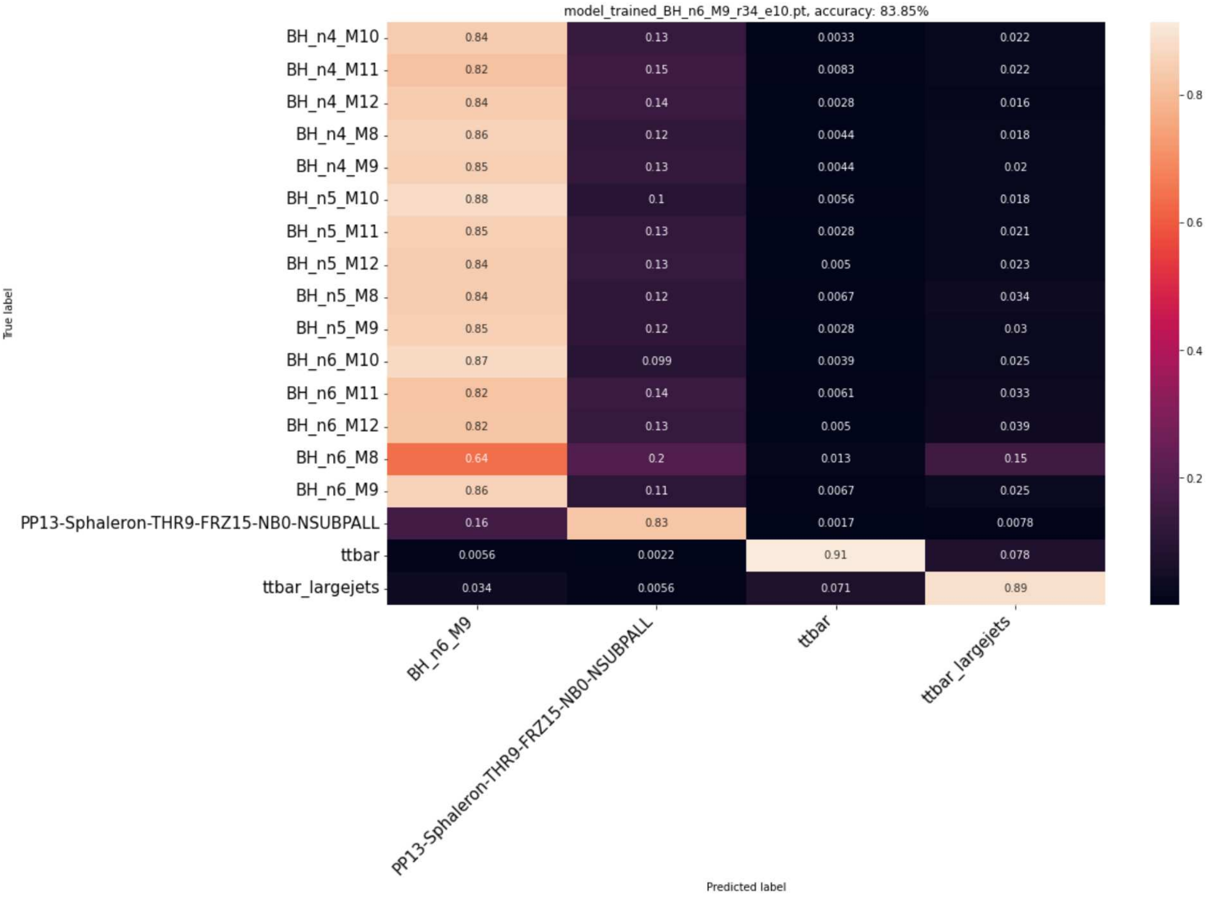


Figure 18. Confusion matrix for the model trained on the n6_m9 training set. Validated on the validation set sampled from all Black Hole datasets.

# 6 Discussion

The object multiplicities of the Black Hole files are generally smaller than that of the Sphalerons as shown in figure 2. The background processes ($T\overline{T}$) have the smallest multiplicities. One could expect by this basis, that the machine learning algorithms would have more troubles distinguishing between a Black Hole, and a Sphaleron or $T\overline{T}$ event, compared to distinguishing between Sphalerons and $T\overline{T}$ events. Figure 4 shows increasing $p_T$ with increasing Black Hole mass. The Black Hole datasets with the lowest masses has similar $p_T$ to that of the Sphalerons, as shown in figure 5. These could be expected to be more difficult to distinguish from the Sphalerons.

Figure 6 and 7 show that for Sphalerons, the jets have less $p_T$ than the jets from Black Holes, which is also the case for the other particles. Figure 8 shows that the MTE of Sphalerons are larger than for the Black Holes. These properties could provide a handle for the machine learning algorithms to distinguish the events.

Figure 14 shows the confusion matrix for the machine learning model trained on all Black Hole files. It provides a reasonable accuracy of (86.8±0.6)% when run on its own validation set. In figure 15 this model is run on all classes of Black Holes, providing an accuracy of (81.3±0.6)%. A significant error rate is found when the model tries to validate the Black Holes with dimensions n6 and minimum mass 8[TeV]. When this label is presented, the model mistakenly predicts $T\overline{T}$ large_jets, (15±2)% of the time. It is the only label with an error rate comparable to the Sphalerons. The model trained on N6M8 dataset, was able to correctly predict its own validation set with (89±2)% accuracy, fig 16. When it was run across all the Black Hole files, it predicted labels from the Black Holes as Sphalerons, in the range of 50% to 84% of the time, fig 17. What caused the problems of the dataset is unknown. Figure 18 shows the confusion matrix of the model trained on the single Black Hole file that yielded the highest accuracy of these models. It's accuracy at (84±2)% is higher even than the model which included all Black Hole files in the training.

If the loss functions for the machine learning models were evaluated, potential accuracy increases from training the model for more epochs could be estimated. Training a single model for a high number of epochs could also determine if there is much to be gained. ~~An additional point of improvement is that the energy calculations for the images were off, and could be a factor in the accuracy of the machine learning models.~~

this sentence is obsolete

# 7 Conclusion

There was no time to compare the accuracy and confusion achieved in this work with neither previous work by the CMS Collaboration nor with other, more traditional event selection methods. Models trained were able to distinguish between the different event types, with accuracies up to 87%, providing some confidence that with further optimizations, machine learning could be a valuable tool for probing this area of physics.

see page 20 for an estimation of cross-section limits

# Appendix

## A. Functions

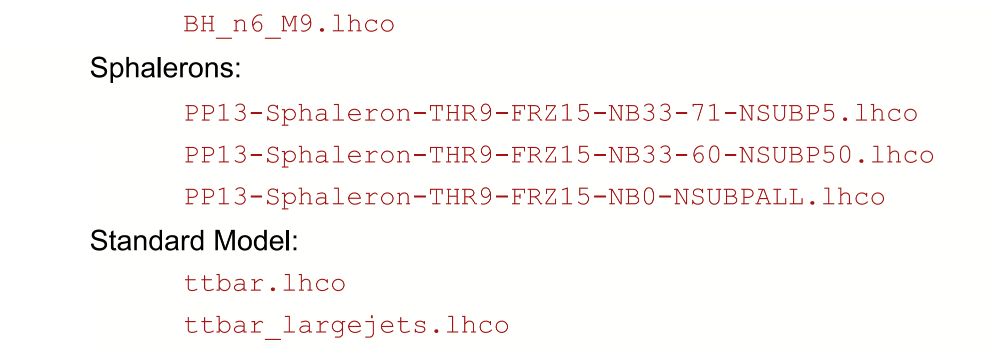
```
data_transforms = {
    'train': transforms.Compose([
        transforms.Resize(256),
        transforms.CenterCrop(224),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]),
    'val': transforms.Compose([
        transforms.Resize(256),
        transforms.CenterCrop(224),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]),
}
```

## B. Data and code

Datasets:

Black Holes:

BlackMaxOutputFirstRun.lhco
BlackMaxOutputFirstRun2.lhco
BH_n4_M10.lhco
BH_n4_M11.lhco
BH_n4_M12.lhco
BH_n4_M8.lhco
BH_n4_M9.lhco
BH_n5_M10.lhco
BH_n5_M11.lhco
BH_n5_M12.lhco
BH_n5_M8.lhco
BH_n5_M9.lhco
BH_n6_M10.lhco
BH_n6_M11.lhco
BH_n6_M12.lhco
BH_n6_M8.lhco



`BH_n6_M9.lhco`

Sphalerons:

`PP13-Sphaleron-THR9-FRZ15-NB33-71-NSUBP5.lhco`
`PP13-Sphaleron-THR9-FRZ15-NB33-60-NSUBP50.lhco`
`PP13-Sphaleron-THR9-FRZ15-NB0-NSUBPALL.lhco`

Standard Model:

`ttbar.lhco`
`ttbar_largejets.lhco`

https://github.com/mitresthen/phys117_2021

Example Limit on BH and SPH with numbers from Figure 18.

| Fig 18 | BH eff | ttbar_eff | ttbar_high_pt_eff |
|---|---|---|---|
| BHn4M8 | 0.86 | 0.0044 | 0.018 |
| BHn4M10 | 0.84 | 0.0033 | 0.022 |
| BHn6M10 | 0.87 | 0.0039 | 0.025 |

With the ttbar crossection 523000 fb and ttbar_high_pt crossection 1500 pb (that is contained in the ttbar crossection) we expect the following number of background events for the Run 2 luminosity of LUMI= 139/fb :

| | ttbar_eff | events from ttbar | ttbar_High_pt_eff | events from ttbar_high_pt | 2sigma (95%) xsection limit |
|---|---|---|---|---|---|
| BHn4M8 | 0.0044 | 319867+-566 | 0.018 | 3753+-61 | 9.5 fb |
| BHn4M10 | 0.0033 | 239900+-490 | 0.022 | 4587+-68 | 8.4 fb |
| BHn6M10 | 0.0039 | 283518+-532 | 0.025 | 5212+-72 | 8.8 fb |

The ttbar_high_pt events are expected to contained in ttbar events thus the limit on BH cross-section is calculated as 2sigma(ttbar)/(BH eff)/LUMI . These are limits in absence of any signal. Limits from the CMS paper are 0.1-10 fb depending on the BH mass (n=6). We still have a way to go, but it is not bad for the first trial, we need to spend more time on background rejection procedure.

Example limit on sphalerons:

| | eff | ttbar_eff | ttbar_high_pt_eff | events from ttbar | 2sigma (95%) xsec limit |
|---|---|---|---|---|---|
| THR9 | 0.83 | 0.0017 | 0.0078 | 123585+-351 | 6.1 fb |

--------------------------------------------------------------------------

After we improve the background estimation/rejection one can produce limits on BH assuming a presence of a given SPH signal and vice-versa.

# References

[1] Compact Muon Solenoid Detector, CERN https://home.cern/science/experiments/cms.

[2] Compact Muon Solenoid Detector, CERN https://cms.cern/

[3] the Large Hadron Collider, CERN https://home.cern/science/accelerators/large-hadron-collider

[4] F. R. Klinkhammer,  N. S. Manton, A saddle-point solution in the Weinberg-Salam theory. https://journals.aps.org/prd/abstract/10.1103/PhysRevD.30.2212

[5] Steven B Giddings, Scott Thomas, High Energy Colliders as Black Hole Factories: The End of Short Distance Physics. arXiv:hep-ph/0106219

[6] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90

[7] Pythia, https://pythia.org/

[8] DELPHES 3, A modular framework for fast simulation of a generic collider experiment. https://arxiv.org/abs/1307.6346

[9] The LHC Olympics 2020: A Community Challenge for Anomaly Detection in High Energy Physics. arXiv:2101.08320 **[hep-ph]**

[10] How to Read LHC Olympics Data Files (by Jesse Thaler, 19 Dic 2006). http://madgraph.phys.ucl.ac.be/Manual/lhco.html

[11] ATLAS, CERN https://home.cern/science/experiments/atlas

[12] ATLAS, CERN https://atlas.cern/

[13] PIs: Stefan Pokorski, Anna Lipniacka, grant no. 2019/34/H/ST2/00707 “Understanding the Early Universe: interplay of theory and collider experiments”  Accessed on Nov. 30, 2021. [Online]. Available: https://www.fuw.edu.pl/~ksakurai/grieg/index.html?lang=en

[14] Nima Arkani–Hamed, Savas Dimopoulos, Gia Dvali, The hierarchy problem and new dimensions at a millimeter, Physics Letters B, Volume 429, Issues 3–4, 1998, Pages 263-272, ISSN 0370-2693, https://doi.org/10.1016/S0370-2693(98)00466-3

[15] Sirunyan, A.M., Tumasyan, A., Adam, W., Ambrogi, F., Asilar, E., Bergauer, T., Brandstetter, J., Dragicevic, M., Erö, J., Del Valle, A.E. and Flechl, M., 2018. Search for black holes and sphalerons in high-multiplicity final states in proton-proton collisions at √s= 13 TeV. Journal of High Energy Physics, 2018(11), pp.1-49.

[16] Ringwald, A., 1990. High energy breakdown of perturbation theory in the electroweak instanton sector. Nuclear Physics B, 330(1), pp.1-18. https://doi.org/10.1016/0550-3213(90)90300-3

[17] Python, programming language, https://www.python.org/

[18] Google Colab, https://colab.research.google.com/

[19] Jupyter, https://jupyter.org/

[20] Fowlie, A., 2015. LHCO_reader: A new code for reading and analyzing detector-level events stored in LHCO format. *arXiv preprint arXiv:1510.07319*.

[21] Andrews, M., Paulini, M., Gleyzer, S. et al. End-to-End Physics Event Classification with CMS Open Data: Applying Image-Based Deep Learning to Detector Data for the Direct Classification of Collision Events at the LHC. Comput Softw Big Sci 4, 6 (2020). https://doi.org/10.1007/s41781-020-00038-8

[22] Sasank Chilamkurthy, "TRANSFER LEARNING FOR COMPUTER VISION TUTORIAL" Accessed on: Nov. 30, 2021. [Online]. Available: https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html