

1 **Using Machine Learning Techniques for Data Quality**
2 **Monitoring at CMS Experiment**

3 by

4 Guillermo A. Fidalgo Rodríguez

5 A thesis presented for the degree of

6 BACHELLOR'S OF SCIENCE??

7 in

8 Physics

9 UNIVERSITY OF PUERTO RICO
10 MAYAGÜEZ CAMPUS

11 2018

12 Approved by:

13 _____
14 Sudhir Malik, Ph.D.

15 President, Graduate Committee
Date

16 _____
17 Héctor Méndez, Ph.D.

18 Member, Graduate Committee
Date

19 _____
20 Samuel Santana Colón, Ph.D.
21 Member, Graduate Committee

Date

22 _____
23 Rafael A. Ramos, Ph.D.
24 Chairperson of the Department

Date

²⁵ Abstract

²⁶ The Data Quality Monitoring (DQM) of CMS is a key asset to deliver high-quality
²⁷ data for physics analysis and it is used both in the online and offline environment. The cur-
²⁸ rent paradigm of the quality assessment is labor intensive and it is based on the scrutiny of
²⁹ a large number of histograms by detector experts comparing them with a reference. This
³⁰ project aims at applying recent progress in Machine Learning techniques to the automa-
³¹ tion of the DQM scrutiny. In particular the use of convolutional neural networks to spot
³² problems in the acquired data is presented with particular attention to semi-supervised
³³ models (e.g. autoencoders) to define a classification strategy that doesn't assume previ-
³⁴ous knowledge of failure modes. Real data from the hadron calorimeter of CMS are used
³⁵ to demonstrate the effectiveness of the proposed approach.

³⁶ *Keywords:* [DQM, online, offline, Machine Learning]

³⁷ **Acknowledgments**

³⁸ I wish to thank United States State Department and University of Michigan for pro-
³⁹ viding the opportunity to work abroad at CERN during the 2018 Winter Semester. I also
⁴⁰ wish to thank CERN staff, CMS Experiment , Texas Tech University, and the University
⁴¹ of Puerto Rico at Mayagüez, with special thanks to Dr. Federico de Guio for his local
⁴² mentorship and Dr. Nural Akchurin, and Dr. Sudhir Malik for their guidance. A very
⁴³ special thanks to Dr. Jean Krisch for accepting me for this great research opportunity and
⁴⁴ Dr. Steven Goldfarb for being a wonderful host and overall local guidance at CERN.

⁴⁵ List of Figures

46	2.1 CMS Detector	4
47	2.2 The trajectory of a particle traveling through the layers of the detector leaving behind it's signature footprint	5
49	4.1 Occupancy maps with 5x5 affected regions	11
50	4.2 Weights and Biases	12
51	4.3 Gradient Descent algorithm	13
52	4.4 Loss Function surface	14
53	5.1 Occupancy Maps with 1x1 bad regions. A) Good image B) Dead image C) Hot image	15
54	5.2 Two Convolutional Layers Model	17
55	5.3	17
57	5.4 Jet multiplicity and the associated S_γ scale factor in the loose photon con- trol region before any corrections are applied.	19
59	5.5 N_{jet} distribution in the tight $\mu\mu$ control region after S_γ corrections.	20
60	5.6 p_T^γ (left) and $p_T^{\gamma(miss)}$ (right) distributions after applying the $S_\gamma(N_j)$ scale factor. Comparing to ??, an improvement in the agreement between data/MC can be observed.	20
63	5.8 H_T and m_{T2} distributions applying the $S_\gamma(N_j)$ scale factor.	21
64	5.7 N_b distribution before (left) and after (right) applying the $S_\gamma(N_j)$ scale factor.	21
66	5.9 Shown are data/MC comparisons for the p_T^{miss} (left) and H_T (right) distri- butions after applying both the N_j -dependent shape corrections (S_γ) and the global normalization scale factor (R_{norm}).	22
69	5.10 Shown are data/MC comparisons for the N_j (left) and N_b (right) distribu- tions after applying both the N_j -dependent shape corrections (S_γ) and the global normalization scale factor (R_{norm}).	23
72	5.11 Systematic uncertainty in the final prediction, as a function of the search bin, associated to the MC statistics.	24
74	5.12 $Z \rightarrow \nu\bar{\nu}$ background prediction for all search bins, including the break- down of the various uncertainties.	25

76 **Contents**

77	Abstract	i
78	Acknowledgments	ii
79	List of Figures	iii
80	1 Introduction	1
81	2 The CMS Experiment	3
82	3 Data Collection and Data Quality Monitoring	6
83	3.1 What is Data Collection for CMS?	6
84	3.2 What is Data Quality Monitoring?	7
85	4 What is Machine Learning?	9
86	4.1 Developing the Algorithm	10
87	4.2 Teaching the Algorithm	12
88	5 Results	15
89	5.1 SL Models for known anomalies in the HCAL data for DQM	16
90	5.1.1 Two Convolutional Layers for binary classification	16
91	5.1.2 Photon Purity and Fake Rate	18
92	5.1.3 Normalization Correction Using the tight $Z \rightarrow \mu^+ \mu^-$ Control Sample	21
93	5.2 Results	23
94	5.2.1 Systematics	23
95	5.2.2 $Z \rightarrow \nu \bar{\nu}$ Estimation for the Search Bins	25
97	6 References	26

₉₈ **Chapter 1**

₉₉ **Introduction**

₁₀₀ The work for this thesis was performed at CERN on CMS Experiment. CERN stands
₁₀₁ for European Organization for Nuclear Research. It was founded in 1954 and is located
₁₀₂ at the Franco-Swiss border near Geneva. At CERN, physicists and engineers are probing
₁₀₃ the fundamental structure of the universe. They use the world's largest and most complex
₁₀₄ scientific instruments to study the basic constituents of matter – the fundamental parti-
₁₀₅ cles. The instruments used at CERN are purpose-built particle accelerators and detectors.
₁₀₆ Accelerators boost beams of particles to high energies before the beams are made to col-
₁₀₇ lide with each other or with stationary targets. Detectors observe and record the results
₁₀₈ of these collisions. The accelerator at CERN is called the Large Hadron Collider (LHC),
₁₀₉ the largest machine ever built by humans and it collides particles (protons) at close to the
₁₁₀ speed of light. The process gives the physicists clues about how the particles interact, and
₁₁₁ provides insights into the fundamental laws of nature. Seven experiments at the LHC use
₁₁₂ detectors to analyze particles produced by proton-proton collisions. The biggest of these
₁₁₃ experiments, ATLAS and CMS, use general-purpose detectors designed to study the fun-
₁₁₄ damental nature of matter and fundamental forces and to look for new physics or evidence
₁₁₅ of particles that are beyond the Standard Model. Having two independently designed de-
₁₁₆ tectors is vital for cross-confirmation of any new discoveries made. The other two major
₁₁₇ detectors ALICE and LHCb, respectively, study a state of matter that was present just
₁₁₈ moments after the Big Bang and preponderance of matter than antimatter. Each experi-

119 ment does important research that is key to understanding the universe that surrounds and
120 makes us.

121

122 [Chapter 2](#) presents a basic description of the Large Hadron Collider and CMS Detector

123

124 ?? gives a brief motivation

125

126 ?? is dedicated to a study optimizing

127

128 ?? ptimated.

129

130 ?? details an improvarger production cross-section than Z+jets process used before.

131

132 The conclusions and results of each chapter are presented in the corresponding chap-
133 ter.

134

135 This thesis work has been presented at several internal meetings of the CMS Experi-
136 ment and at the following international meetings and conferences:

137 1. **Andrés Abreu** gave a talk “*Estimation of the Z Invisible Background for Searches*
138 *for Supersymmetry in the All-Hadronic Channel*” at “APS April 2018: American

139 Physical Society April Meeting 2018, 14-17 Apr 2018”, Columbus, OH

140 2. **Andrés Abreu** gave a talk “*Phase-2 Pixel upgrade simulations*” at the “USLUA

141 Annual meeting: 2017 US LHC Users Association Meeting, 1-3 Nov 2017”, Fer-
142 milab, Batavia, IL

¹⁴³ Chapter 2

¹⁴⁴ The CMS Experiment

¹⁴⁵ The Compact Muon Solenoid (CMS) detector is a general purpose particle detector
¹⁴⁶ designed to investigate various physical phenomena concerning the SM and beyond it,
¹⁴⁷ such as Supersymmetry, Extra Dimensions and Dark Matter. As its name implies, the
¹⁴⁸ detector is a solenoid which is constructed around a superconducting magnet capable of
¹⁴⁹ producing a magnetic field of 3.8 T. The magnetic coil is 13m long with an inner diameter
¹⁵⁰ of 6m, making it the largest superconducting magnet ever constructed. The CMS detector
¹⁵¹ itself is 21m long with a diameter of 15m and it has a weight of approximately 14,000
¹⁵² tons. The CMS experiment is one of the largest scientific collaborations in the history
¹⁵³ of mankind with over 4,000 participants from 42 countries and 182 institutions. CMS is
¹⁵⁴ located at one of these points and it essentially acts as a giant super highspeed camera
¹⁵⁵ that makes 3D images of the collisions that are produced at a rate of 40 MHz (40 million
¹⁵⁶ times per second). The detector has an onion-like structure to capture all the particles that
¹⁵⁷ are produced in these high energy collisions most of them being unstable and decaying
¹⁵⁸ further to stable particles that are detected. CMS detector was designed with the following
¹⁵⁹ features (as shown in [Figure 2.1](#)) :

- ¹⁶⁰ 1. A **magnet** with large bending power and high performance muon detector for good
¹⁶¹ muon identification and momentum resolution over a wide range of momenta and
¹⁶² angles.

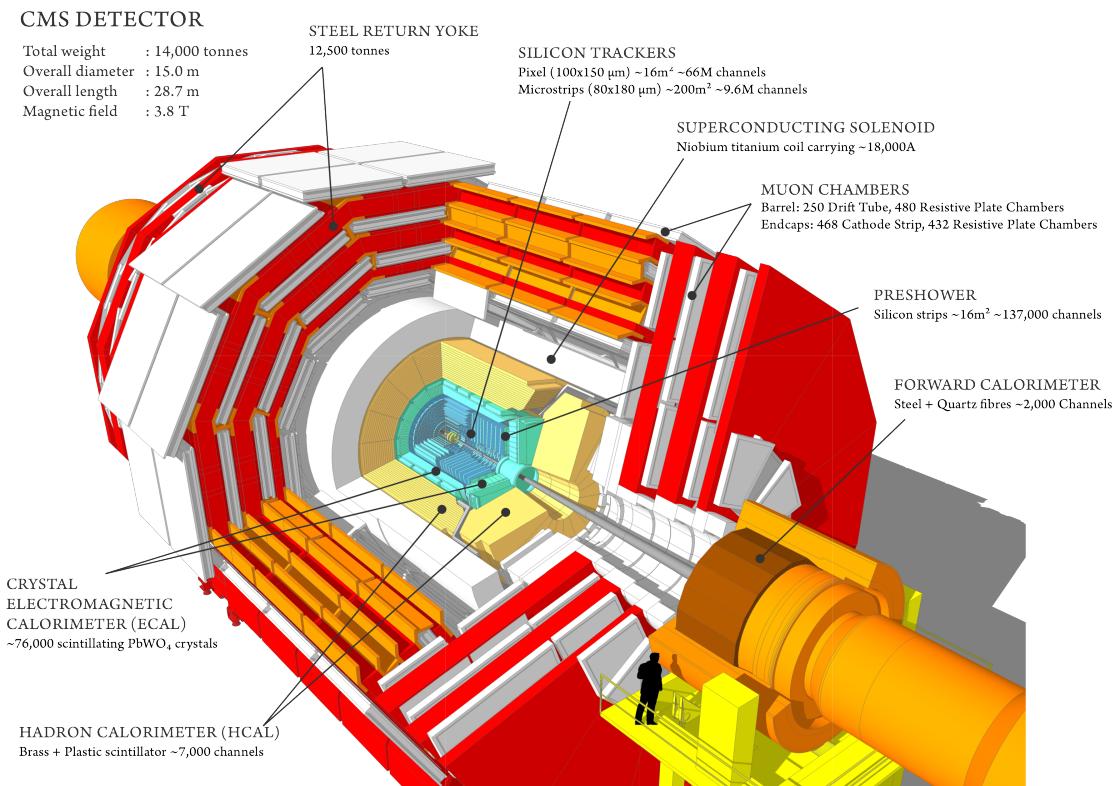


Figure 2.1: CMS Detector

- 163 2. An **inner tracking system** capable of high reconstruction efficiency and momen-
- 164 tum resolution requiring **pixel detectors** close to the interaction region.

- 165 3. An **electromagnetic calorimeter** able to provide good electromagnetic energy res-
- 166 olution and a high isolation efficiency for photons and leptons.

- 167 4. A **hadron calorimeter** capable of providing precise missing-transverse-energy and
- 168 dijet-mass resolution.

169 A property from these particles that is exploited is their charge. Normally, particles
 170 produced in collisions travel in a straight line, but in the presence of a magnetic field,
 171 their paths are skewed and curved. Except the muon system, the rest of the subdetectors
 172 lie inside a 3.8 Tesla magnetic field . Due to the magnetic field the trajectory of charged
 173 particle produced in the collisions gets curved (as shown in [Figure 2.2](#)) and one can
 174 calculate the particle's momentum and know the type of charge on the particle. The
 175 Tracking devices are responsible for drawing the trajectory of the particles by using a
 176 computer program that reconstructs the path by using electrical signals that are left by

177 the particle as they move. The Calorimeters measure the energy of particles that pass
 178 through them by absorbing their energy with the intent of stopping them. The particle
 179 identification detectors work by detecting radiation emitted by charged particles and using
 180 this information they can measure the speed, momentum, and mass of a particle. After the
 181 information is put together to make the “snapshot” of the collision one looks for results
 182 that do not fit the current theories or models in order to look for new physics.

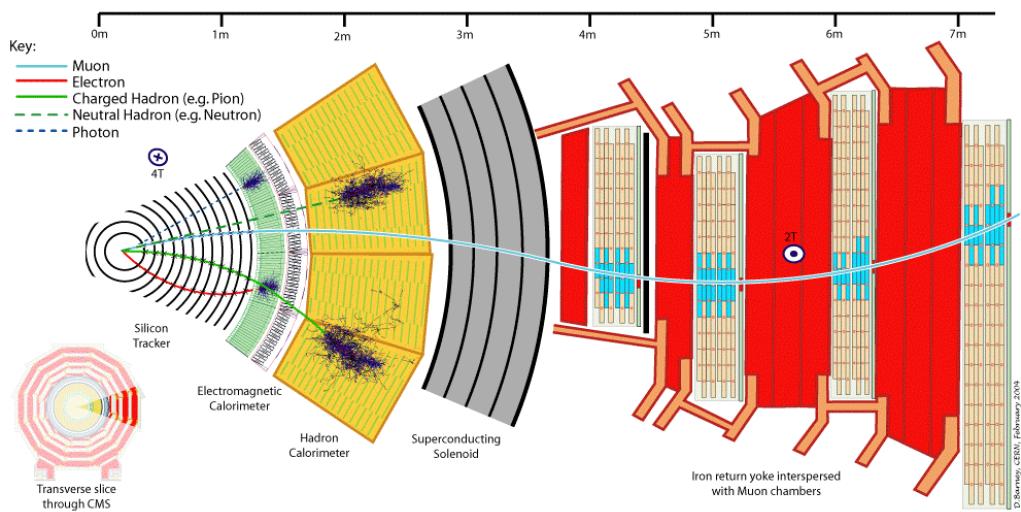


Figure 2.2: The trajectory of a particle traveling through the layers of the detector leaving behind its signature footprint

183 The project focusses specifically on data collected from one of the Calorimeters, - the
 184 Hadron Calorimeter (HCAL). The HCAL, as its name indicates, is designed to detect and
 185 measure the energy of hadrons or, particles that are composed of quarks and gluons, like
 186 protons and neutrons. Additionally, it provides an indirect measurement of the presence
 187 of non-interacting, uncharged particles such as neutrinos (missing energy) . Measuring
 188 these particles is important as they can tell us if new particles such as the Higgs boson or
 189 supersymmetric particles (much heavier versions of the standard particles we know) have
 190 been formed. The layers of the HCAL are structured in a staggered fashion to prevent any
 191 gaps that a particle might pass through undetected. There are two main parts: the barrel
 192 and the end caps. There are 36 barrel wedges that form the last layer of the detector inside
 193 the magnet coil, there is another layer outside this, and on the endcaps, there are another
 194 36 wedges to detect particles that come out at shallow angles with respect to the beam
 195 line.

₁₉₆ **Chapter 3**

₁₉₇ **Data Collection and Data Quality**
₁₉₈ **Monitoring**

₁₉₉ **3.1 What is Data Collection for CMS?**

₂₀₀ During data taking there are millions of collisions occurring in the center of the detec-
₂₀₁ tor every second. The data per event is around one million bytes (1 MB), that is produced
₂₀₂ at a rate of about 600 million events per second [1], that's about 600 MB/s. Keeping
₂₀₃ in mind that only certain events are considered "interesting" for analysis, the task of de-
₂₀₄ ciding what events to consider out of all the data collected is a two-stage process. First,
₂₀₅ the events are filtered down to 100 thousand events per second for digital reconstruction
₂₀₆ and then more specialized algorithms filter the data even more to around 100 200 events
₂₀₇ per second that are found interesting. For CMS there is a Data Acquisition System that
₂₀₈ records the raw data to what's called a High-Level Trigger farm which is a room full
₂₀₉ of servers that are dedicated to processing and classify this raw data quickly. The data
₂₁₀ then gets sent to what's known as the Tier-0 farm where the full processing and the first
₂₁₁ reconstruction of the data are done. [2]

212 3.2 What is Data Quality Monitoring?

213 To operate a sophisticated and complex apparatus as CMS, a quick online feedback on
214 the quality of the data recorded is needed to avoid taking low quality data and to guarantee
215 a good baseline for the offline analysis. Collecting a good data sets from the collisions
216 is an important step towards search for new physics as deluge of new data poses an extra
217 challenge of processing and storage. This all makes it all the more important to design
218 algorithms and special software to control the quality of the data. This is where the Data
219 Quality Monitoring (DQM) plays a critical in the maintainability of the experiment, the
220 operation efficiency and performs a reliable data certification. The high-level goal of
221 the system is to discover and pinpoint errors, problems occurring in detector hardware
222 or reconstruction software, early, with sufficient accuracy and clarity to maintain good
223 detector and operation efficiency. The DQM workflow consists of 2 types: **Online** and
224 **Offline**.

225 The **Online** DQM consists of receiving data taken from the event and trigger his-
226 tograms to produce results in the form of monitoring elements like histogram references
227 and quality reports. This live monitoring of each detector's status during data taking gives
228 the online crew the possibility to identify problems with extremely low latency, mini-
229 mizing the amount of data that would otherwise be unsuitable for physics analysis. The
230 scrutiny of the Online DQM is a 24/7 job that consists of people or shifters that work at the
231 CMS control center constantly monitoring the hundreds of different plots and histograms
232 produced by the DQM software. This consumes a lot of manpower and is strenuous work.

233 The **Offline** DQM is more focused on the full statistics over the entire run of the
234 experiment and works more on the data certification. In the offline environment, the
235 system is used to review the results of the final data reconstruction on a run-by-run basis,
236 serving as the basis for certified data used across the CMS collaboration in all physics
237 analyses. In addition, the DQM framework is an integral part of the prompt calibration
238 loop. This is a specialized workflow run before the data are reconstructed to compute and
239 validate the most up-to-date set of conditions and calibrations subsequently used during

²⁴⁰ the prompt reconstruction.

²⁴¹ This project aims to minimize the DQM scrutiny by eye and automate the process so
²⁴² that there is a more efficient process to monitor the detector and the quality of the data by
²⁴³ implementing Machine Learning techniques.

²⁴⁴ Chapter 4

²⁴⁵ What is Machine Learning?

²⁴⁶ Machine Learning (ML) can be defined as an application of Artificial Intelligence that
²⁴⁷ permits the computer system to learn without being told explicitly. In ML a computer
²⁴⁸ program is said to learn from experience E with respect to some class of tasks T and
²⁴⁹ performance measure P, if its performance at tasks in T, as measured by P, improves
²⁵⁰ with experience E [3]. ML has made tremendous strides in the past decades and has
²⁵¹ become very popular recently due to its multifaceted applications. It is being used on
²⁵² social media, marketing, and in the scientific community as well. Some examples of
²⁵³ ML applications are: the algorithms used on application in smartphones to detect human
²⁵⁴ faces, self-driving cars, computer games, stock prediction, and voice recognition. An
²⁵⁵ interesting characteristic of ML algorithms is that the more data one inputs the better is
²⁵⁶ the performance. The ML application has a very wide spectrum covering almost every
²⁵⁷ aspect of human endeavor that involves a lot of data. Scientific analysis today generates
²⁵⁸ enormous data and is hence a perfect use case to apply ML techniques. In this work
²⁵⁹ we use enhanced ML techniques based on progress in the recent past.

²⁶⁰ In general, there are two main categories to classify machine learning problems: **Su-**
²⁶¹ **pervised Learning** (SL) and **Unsupervised Learning** (UL). SL is the most used ML
²⁶² approach and has proven to be very effective for a wide variety of problems. Examples
²⁶³ of common SL problems are: spam filters, predicting housing prices, identifying a ma-
²⁶⁴ lignant or benign tumor, etc. These types of problems are characterized by providing a

265 “right answer” as a reference. For example, spam filter algorithms identify emails that
266 are spams by training on a dataset that has examples of such emails. In case of predicting
267 house prices, the algorithm is trained on a dataset of houses involving features like the
268 area of the house, number of rooms, and the selling price of the house.

269 UL algorithms are different in the sense that they do not have the “right answers”
270 given to the machine. Instead, UL algorithms are used for finding patterns and make
271 clusters from the given data. That is what also forms the basis of a search engine (e.g.
272 Google news). Clicking on a link to a news article, one gets many different stories of
273 different journals that have some correlation with the article searched. This happens be-
274 cause the ML algorithm is capable of learning features and shared patterns from a bunch
275 of data without being given any specifics. Another interesting UL problem is the so-called
276 “cocktail party” that involves distinguishing the voice of two people recording on two mi-
277 crophones located at different places. The ML algorithm is able to separate the sources of
278 the voices in the recordings by learning the voice features that correspond to each person,
279 showing the power of the UL algorithm.

280 In this study, I have focused on an SL approach and a variant of the UL approach,
281 called the **Semi-Supervised Learning** approach (SSL). The SSL is named so because
282 the data involves looking at images that are already known to be “Good” but one doesn’t
283 necessarily know every possible situation that produces a “Bad” image. The purpose is to
284 define a metric for a “good” image and subsequently decide if an image is “bad” in case
285 it deviates too much from an acceptable value.

286

4.1 Developing the Algorithm

287 To develop an ML algorithm the following are taken into consideration, what is the
288 task? and what is the method to approach the task? In our case, we are looking into images
289 that have information about the activity that the channels in the HCAL are detecting.
290 These images are called ”occupancy maps” and they are a visual way of monitoring the
291 health of the detector itself (see [Figure 4.1](#)). There are two common problems that can be

identified by viewing occupancy maps which are called "dead channels" and "hot towers". They are referred to as "**dead**" and "**hot**" respectively in the rest of this document. Dead channels mean that on a certain place in the occupancy map there is not any readout from the channels on the HCAL and hot channels mean that there are channels that are being triggered by noise or are damaged in a way that makes them readout too much activity.

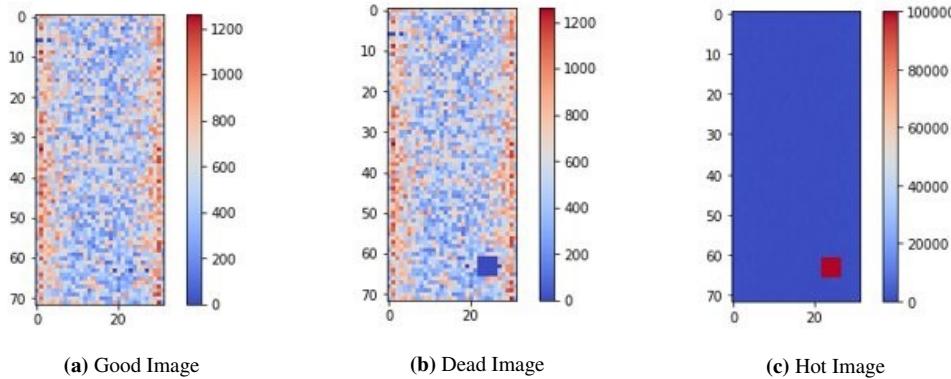


Figure 4.1: Occupancy maps with 5x5 affected regions

The problem is the following, to create a model that can detect and classify what type of scenario is occurring on each occupancy map. For this, we want to go with a SL approach which means that we will give the model the images as the input and it will train on these images by learning to identify patterns or features in the image and try to do a "fit" from the images to their corresponding labels. After the training, the algorithm will be given a testing set for us to evaluate the model's ability to correctly detect if there is a problem with the image and what type of problem is being detected. The output of the model will be the predicted class of the test image. The predictions are based on the labels and their corresponding images that were given to the model during training. This means that if the model was trained with 3 different types of images with their corresponding label the model will only work well for images that present similar patterns or characteristics to those presented in the training. For example, if we only train the model to distinguish between "good" and "hot" then when the model encounters images that aren't either of these two, like an image labeled "dead", then the model will not know what to do with this image and will give it an incorrect label. After the SL model has been tested the next step is trying an SSL model. The term semi-supervised

simply means that there isn't a ground truth label that is being given to the model during training because either there isn't necessarily a ground truth, or we don't know what the ground truth is. What we do know, is what is considered as a "good" image and what this approach hopes to accomplish is to use the error in the reconstruction of the input image and use that information to discriminate between the "good" vs the "bad" images.

4.2 Teaching the Algorithm

The way an ML algorithm learns is by an iterative process called an optimization algorithm in which the predicted output value of the model is compared to the desired output (See [Figure 4.2](#)) and the weights and biases of the model are adjusted such that the predicted output is closer to the desired output.

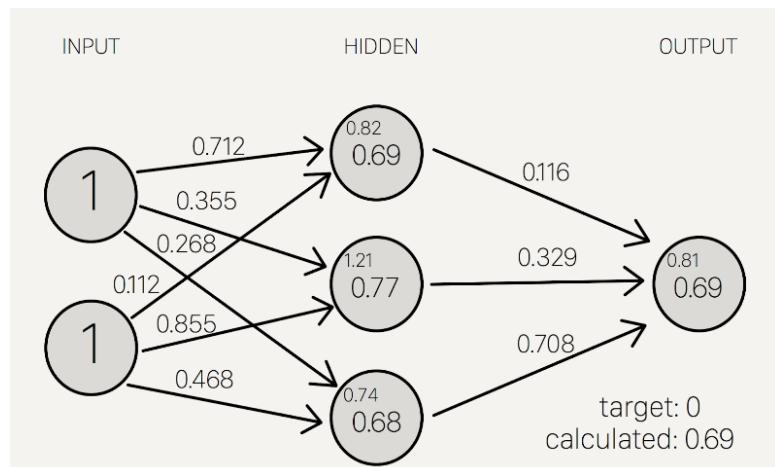


Figure 4.2: Weights and Biases

"Optimization algorithms helps us to **minimize** (or **maximize**) an **Objective function** (*another name for Error function*) $E(x)$ which is simply a mathematical function dependent on the Model's internal **learnable parameters** which are used in computing the target values(Y) from the set of *predictors*(X) used in the model. For example - we call the **Weights(W)** and the **Bias(b)** values of the neural network as its internal learnable *parameters* which are used in computing the output values and are learned and updated in the direction of optimal solution i.e. minimizing the **Loss** by the network's training process and also play a major role in the **training** process of the Neural Network Model." [4].

Gradient Descent

The “Learning” in Machine Learning.

Update the values of X (punish) it when it is wrong.

$$X = X - \eta \nabla(X)$$

X: weights or biases

η : Learning Rate (typically 0.01 to 0.001)

η :The rate at which our network learns. This can change over time with methods such as Adam, Adagrad etc.  (hyperparameter)

$\nabla(X)$: Gradient of X

We seek to update the weights and biases by a value indicating how “off” they were from their target.

Gradients naturally have increasing slope, so we put a negative in front of it to go downwards

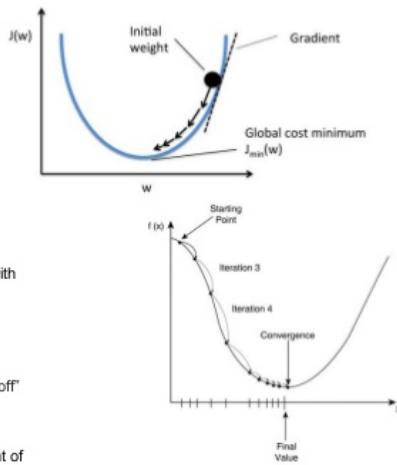


Figure 4.3: Gradient Descent algorithm

- 331 The most basic and probably the most used optimizer is called Gradient Descent (GD).
- 332 GD is based on the concept of using the gradient of a loss or cost function and moving
- 333 the weights and biases of the ML model so that the predicted value is taking a step in the
- 334 decreasing direction of this error function (See [Figure 4.3](#)). In general, the “terrain” of the
- 335 loss function is not a smooth bowl-shaped surface like the one present in the image. The
- 336 most general form of the surface is more similar to a rocky mountain (See [Figure 4.4](#)),
- 337 which presents a problem when using simple optimizers like GD.

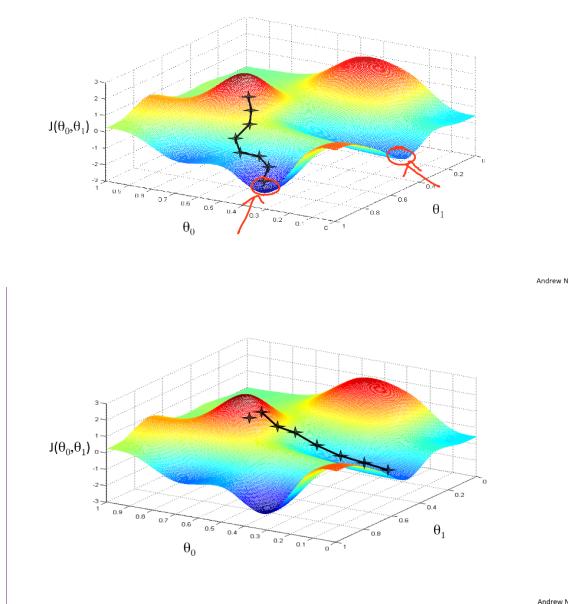


Figure 4.4: Loss Function surface

338

Chapter 5

339

Results

340 Here first the limitations of Scikit-learn predefined ML models - Logistic Regres-
341 sion(LR) and Multi-Layer-Perceptron(MLP), are described. The Logistic Regression
342 Model seems to work almost perfectly with all 3 classes when the bad region size is
343 5x5 (as in [Figure 4.1](#)) with either the same or randomized location. When the bad region
344 size is 1x1 like in [Figure 5.1](#) the LR Model performs poorly with an accuracy of approxi-
345 mately 20%. The MLP does not seem to work in any of the used cases that are studied as
346 it always performs poorly with an accuracy of $\approx 40\%$.

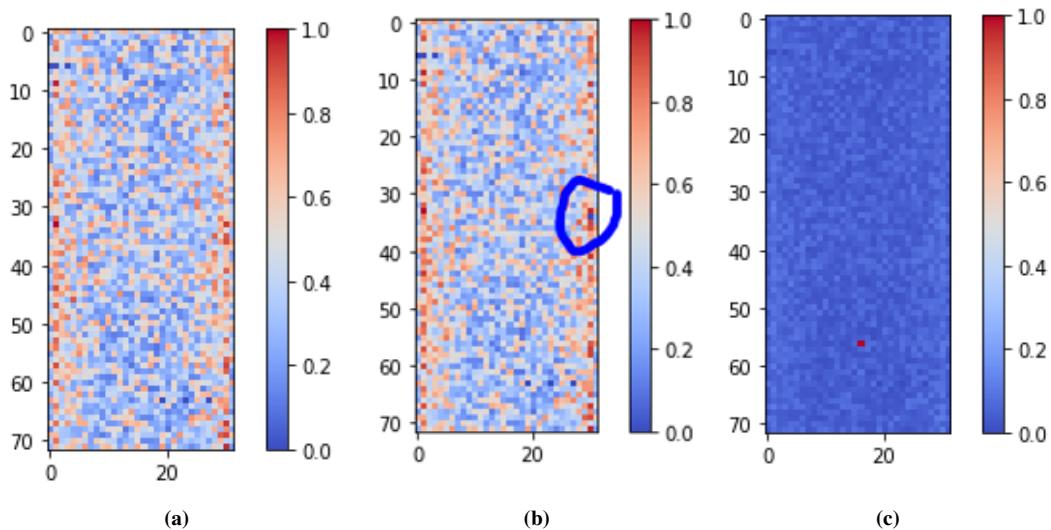


Figure 5.1: Occupancy Maps with 1x1 bad regions. A) Good image B) Dead image C) Hot image

347 Also, the use of Scikit-learn's library is limited in comparison with the Keras module
348 since one cannot customize the structure of the ML model with detail. Moreover, Keras is

349 an ML library designed for developing deep neural networks. Hence it was decided to use
 350 Keras primarily for the creation of the model. With the Keras library, numerous models
 351 were designed with both, SL method and SSL learning method. Using SL method, we are
 352 interested in detecting anomalies and classifying what type of anomaly is seen. With SSL
 353 method, we are interested in looking at the error of the reconstruction of an image to give
 354 an idea that the image given can be considered good or that it might have some unseen
 355 anomalies

356 5.1 SL Models for known anomalies in the HCAL data 357 for DQM

358 We considered three SL Models for classification of known anomalies in the HCAL
 359 data for DQM. These models are based on Convolutional Neural Networks and differ
 360 in the number of layers utilized, their ordering and number of units in each layer. The
 361 Models and the corresponding results are described below.

362 5.1.1 Two Convolutional Layers for binary classification

363 Several variations of the two Convolutional Layers Model were tested and optimized
 364 on the DQM data. This led to an optimal value of 8 units/neurons in the Convolutional
 365 layers. The detail of selecting the number of units per layer is of great importance to find
 366 a balance between efficiency and complexity of a model. More complex models (more
 367 layers and connections) are “heavy” to train in terms of computational cost, provide better
 368 results and are prone to “overfitting” to the training data. Simpler models (fewer layers
 369 and connections) are quicker to train, efficient and computationally economic. However,
 370 simpler models are more likely to “underfit” to the data. The [Figure 5.2](#) below shows a
 371 code snippet with this model.

372 [Figure 5.3](#) below shows the learning curve for this model trained with Good and Hot
 373 images for fixed 5x5 location and the corresponding Confusion Matrix.

```
model = Sequential([
    BatchNormalization(input_shape=input_shape),
    Conv2D(8, kernel_size=(3, 3), strides=(2, 2), activation='relu'),
    Conv2D(8, kernel_size=(3, 3), strides=(2, 2), activation='relu'),
    Dropout(0.25),
    Flatten(),
    Dense(2, activation='softmax')
])
```

Figure 5.2: Two Convolutional Layers Model

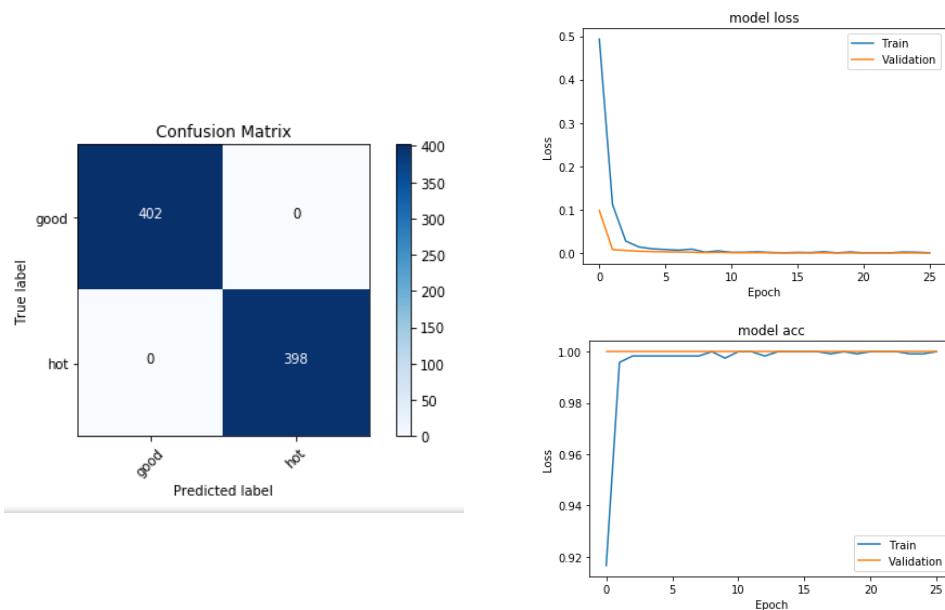


Figure 5.3

of fake photons. In addition to the cuts described, all of the simulation samples are subjected to weights that apply corrections to pileup as well as the b-tagging efficiency. Data, on the other hand, is obtained from a sample that contains events with at least one identified photon. Photons in this sample are also subjected to the high-level trigger LT_Photon175, which restricts the selection to photons that have a $p_T > 175$ GeV. Both simulation and data are subjected to the same selection criteria established in this section.

5.1.2 Photon Purity and Fake Rate

Three different types of photons make up the $\gamma+jets$ CS: prompt photons, produced either directly or through fragmentation, and fake photons. Prompt photons are defined as photons which are formed shortly after the proton-proton collision (i.e. before the produced quarks and gluons have had enough time to form hadrons). Two types of photons fit in this category. The first type, which we designate as direct photons, are photons that are produced directly from the proton-proton interaction [5]. A secondary type of prompt photon, that is virtually indistinguishable from the direct photons at the detector level, originates from the decay of π^0 mesons and are called fragmentation photons. The final type of photon found in the CS corresponds to fake (or non-prompt) photons. The fake photon contribution typically arises from leptons (mostly electrons) whose tracks are not properly reconstructed, yet leave energy measurements in the ECAL.

$$f = \frac{fake}{prompt+fake} ,$$

?? shows the purity and fakerate for photons that pass the loose ID/selection, have a $p_T > 200$ GeV and are within the ECAL acceptance range. A sample is obtained in which 77% of the photons are direct, 12% are fragmentation and 11% are fakes. This implies an average purity of $\sim 89\%$ for this sample, well within the value that is expected. ?? shows the same ratios for the loose $\gamma+jets$ control region described in ?? . Although the amount of statistics has decreased due to the additional cuts, a similar trend can be observed.

399

$$S_{\gamma}^i = \frac{\text{Data}^i - \text{MC}_{\text{other}}^i}{\text{MC}_{\gamma+\text{jets}}^i},$$

400 where i denotes any given bin in the N_j distribution. The shape correction factors S_{γ}^i
 401 are displayed graphically in Figure 5.4 (right) for each N_j bin. These factors correct for
 402 differences in the jet multiplicity shape, while the overall normalization is estimated from
 403 the tight $\mu\mu$ control region. Figure 5.5 shows the N_j distribution in the tight $\mu\mu$ control
 404 region after the calculated scale factors have been applied. The S_{γ} correction will be
 405 applied to the $Z \rightarrow \nu\bar{\nu}$ simulation final prediction for each of the analysis search bins. The
 406 uncertainty associated with the scale factor is estimated from the event yields in the loose
 407 photon control region. This uncertainty will form part of the total systematic uncertainty
 408 in the final prediction.

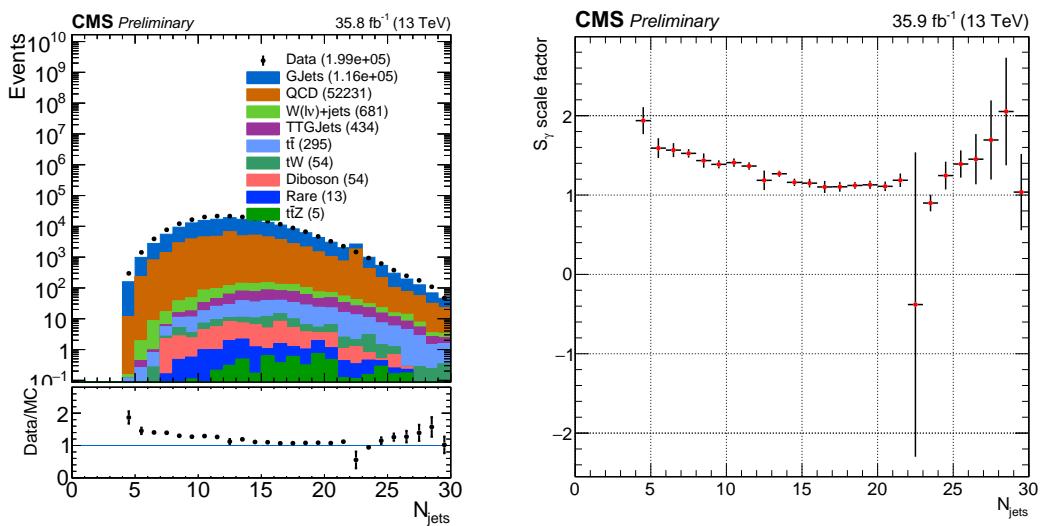


Figure 5.4: Jet multiplicity and the associated S_{γ} scale factor in the loose photon control region before any corrections are applied.

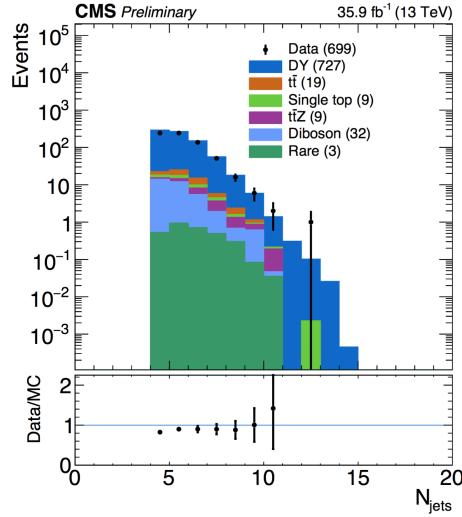


Figure 5.5: N_{jet} distribution in the tight $\mu\mu$ control region after S_γ corrections.

409 The effect of the $S_\gamma(N_j)$ scale factor is shown for various distributions. These results
 410 show that the overall agreement between data and simulation improves after applying the
 411 corresponding shape corrections.

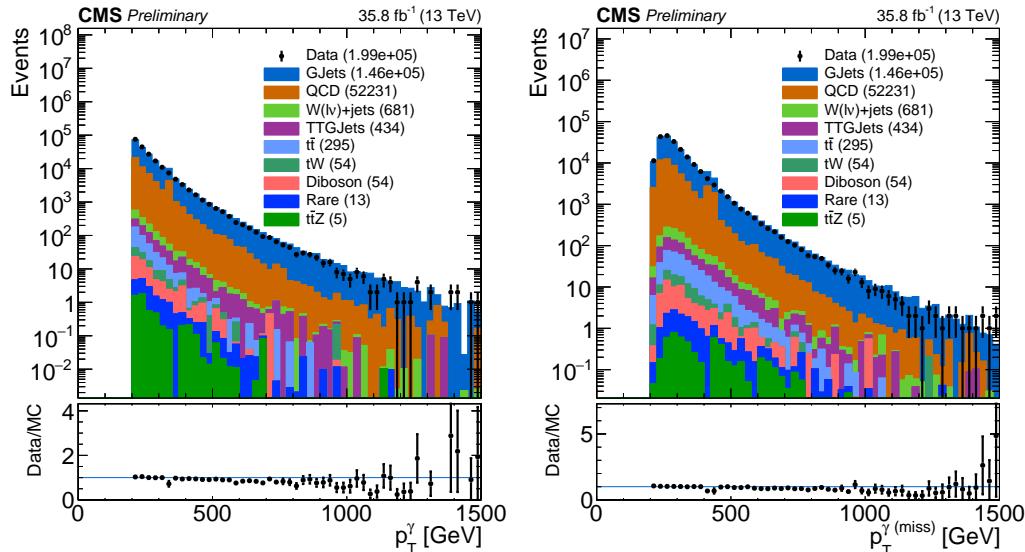


Figure 5.6: p_T^γ (left) and $p_T^{\gamma(\text{miss})}$ (right) distributions after applying the $S_\gamma(N_j)$ scale factor. Comparing to ??, an improvement in the agreement between data/MC can be observed.

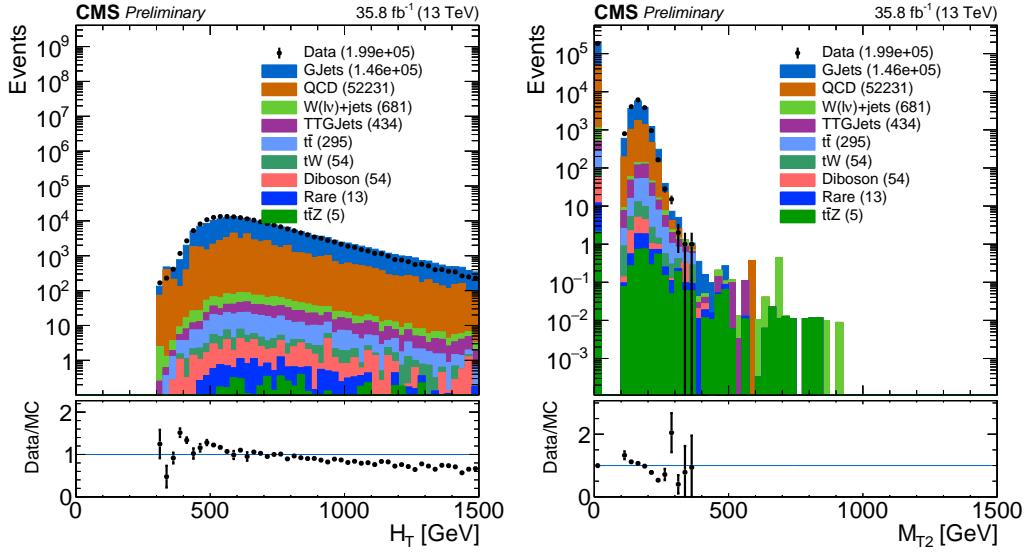


Figure 5.8: H_T and m_{T2} distributions applying the $S_\gamma(N_j)$ scale factor.

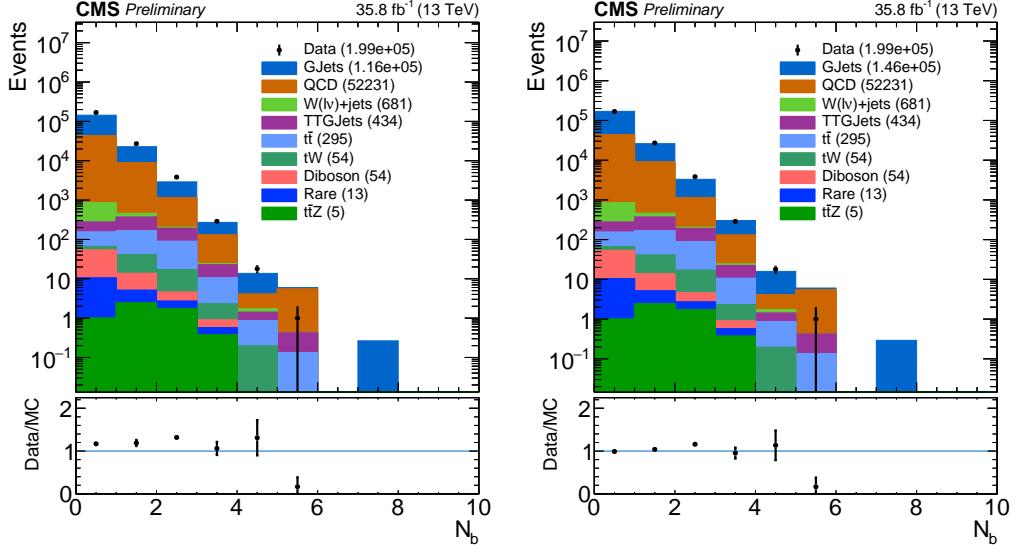


Figure 5.7: N_b distribution before (left) and after (right) applying the $S_\gamma(N_j)$ scale factor.

412 5.1.3 Normalization Correction Using the tight $Z \rightarrow \mu^+ \mu^-$ Control

413 Sample

414 In order to constrain the normalization of the $Z \rightarrow \nu \bar{\nu}$ simulation sample, a normal-
 415 ization correction factor R_{norm} is calculated from the tight $\mu\mu$ control region defined in
 416 subsection 5.1.3. Two categories are considered: the zero b-tagged jet category ($N_b = 0$),
 417 and the ≥ 1 b-tagged jet category ($N_b \geq 1$). Both of these categories are statistically
 418 consistent with each other but the inclusive region ($N_b \geq 0$) has a lower overall un-

certainty. The method used to calculate the normalization scale factor requires that the N_j -dependent shape correction factors already be applied. Then, the R_{norm} factor can be extracted from the ratio of the total event yield in data to that in the simulation. This factor is found to be:

$$R_{norm} = 1.070 \pm 0.085,$$

where the uncertainty includes only the associated statistical uncertainties on data and simulation. This uncertainty is found to be propagated to the final background prediction, see subsection 5.2.1.

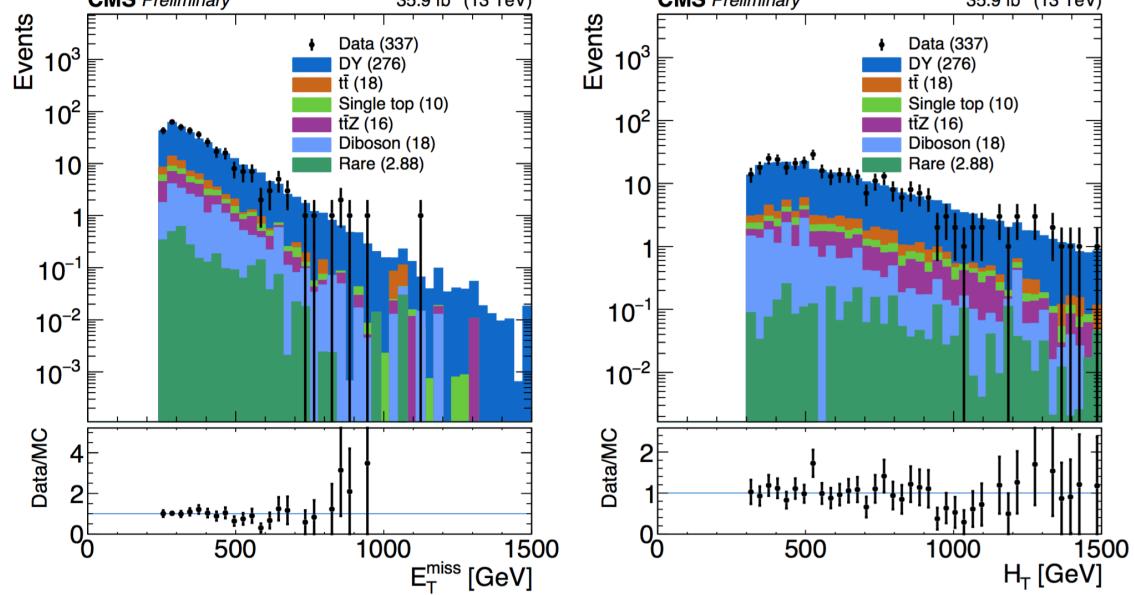


Figure 5.9: Shown are data/MC comparisons for the p_T^{miss} (left) and H_T (right) distributions after applying both the N_j -dependent shape corrections (S_γ) and the global normalization scale factor (R_{norm}).

Data/MC comparisons are shown in Figure 5.9 and Figure 5.10 after applying R_{norm} for several distributions in the study. With this final global scale factor all the required ingredients for the central value of the $Z \rightarrow \nu\bar{\nu}$ background prediction are obtained.

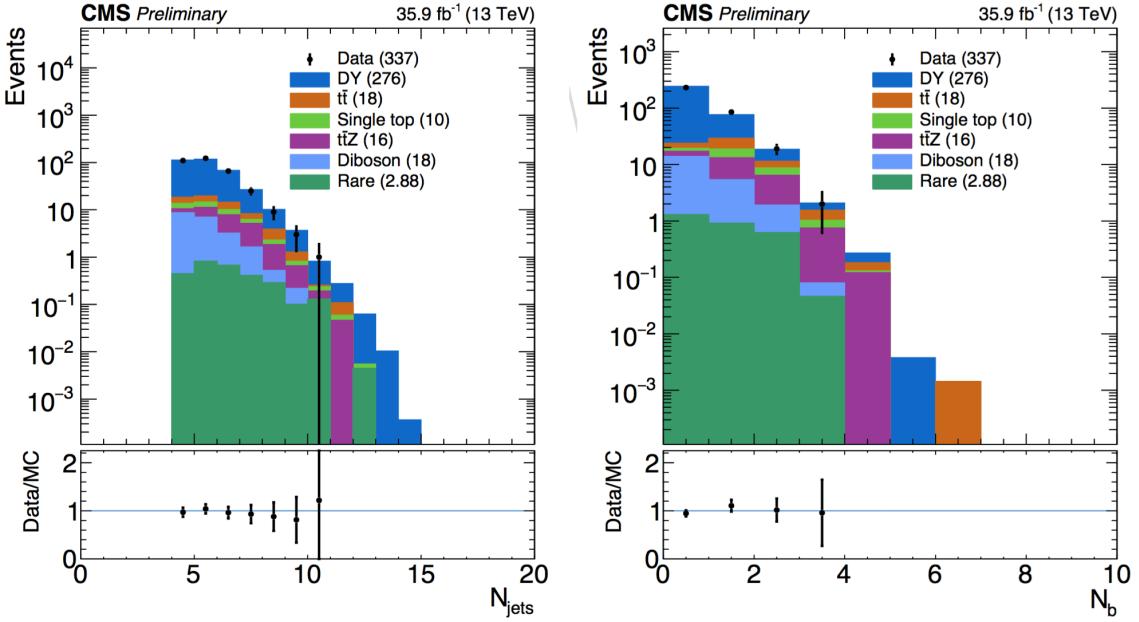


Figure 5.10: Shown are data/MC comparisons for the N_j (left) and N_b (right) distributions after applying both the N_j -dependent shape corrections (S_γ) and the global normalization scale factor (R_{norm}).

431 5.2 Results

432 In this section the results for the final estimation of the of the $Z \rightarrow \nu \bar{\nu}$ are presented.
 433 The current study includes preliminary results using only data obtained at the CMS detec-
 434 tor during 2016. The results for this study are intended to confirm the assumption that the
 435 additional $\gamma + \text{jets}$ control region introduced in this analysis reduce the overall uncertain-
 436 ties obtained in the 2016 analyses (described in ??). Furthermore, this study is intended
 437 as a benchmark for future analyses of the SUSY stop group based in Fermilab and will be
 438 the method used for the 2017 CMS data.

439 5.2.1 Systematics

440 Two categories of uncertainties for the $Z \rightarrow \nu \bar{\nu}$ prediction are considered: uncertain-
 441 ties that are associated to the use of MC simulation and the uncertainties specifically
 442 associated to the background prediction method. Several sources are acknowledged in the
 443 first category mentioned such as PDF and renormalization/factorization scale choices, jet
 444 and p_T^{miss} energy scale uncertainties b-tag scale factor uncertainties, and trigger efficiency
 445 uncertainties. Given that the simulation sample is normalized to data in the tight control

⁴⁴⁶ region, uncertainties associated with the luminosity and cross-section are excluded. In
⁴⁴⁷ addition, the overall $Z \rightarrow \nu\bar{\nu}$ statistical uncertainty from MC simulation is also taken into
⁴⁴⁸ account.

⁴⁴⁹

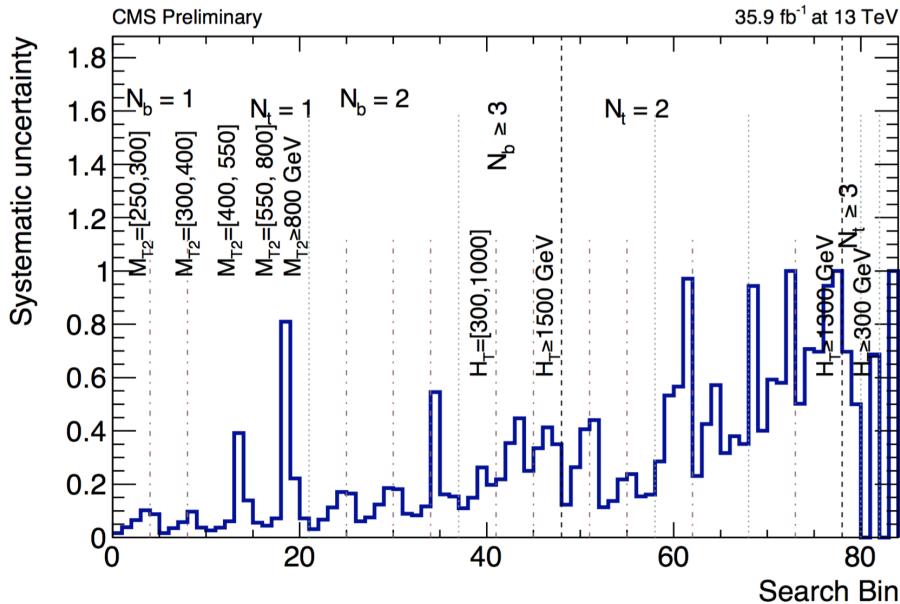


Figure 5.11: Systematic uncertainty in the final prediction, as a function of the search bin, associated to the MC statistics.

⁴⁵⁰ The statistical uncertainty associated with each bin in the MC is propagated as a sys-
⁴⁵¹ tematic uncertainty. The relative uncertainty per bin can be see in [Figure 5.11](#). It shows
⁴⁵² that the uncertainties for the MC vary from as low as 1% up to 81% and even 100% in
⁴⁵³ some regions. Since the final estimation is scaled using the global normalization factor
⁴⁵⁴ from the tight $\mu\mu$ control region (R_{norm}), the total uncertainty, due to limited amounts of
⁴⁵⁵ events in data, is propagated in the final prediction. This is also true for the $S_\gamma(N_j)$ scale
⁴⁵⁶ factor, in which the residual differences in search variables other than N_j are evaluated in
⁴⁵⁷ the loose photon control region. Both the uncertainty arising from the N_j re-weighting
⁴⁵⁸ as well as the residual differences are evaluated together. The uncertainty from R_{norm} is
⁴⁵⁹ propagated as a flat value of 7.9% uncertainty per each search bin.

460 5.2.2 $Z \rightarrow \nu\bar{\nu}$ Estimation for the Search Bins

461 The final estimation for the $Z \rightarrow \nu\bar{\nu}$ background calculated for all 84 search bins is
 462 shown in [Figure 5.12](#). The statistical uncertainty in bins that have zero events is treated
 463 as the average weight (the sum of the weights squared over the weight) times the poisson
 464 error on 0 which is 1.8. This average weight is calculated on the basis of a relaxed cut in
 465 which $N_b \geq 2$ is required. For comparison, a cut in which $N_t > 2$ where two tops are
 466 fake for the $Z \rightarrow \nu\bar{\nu}$ is used.

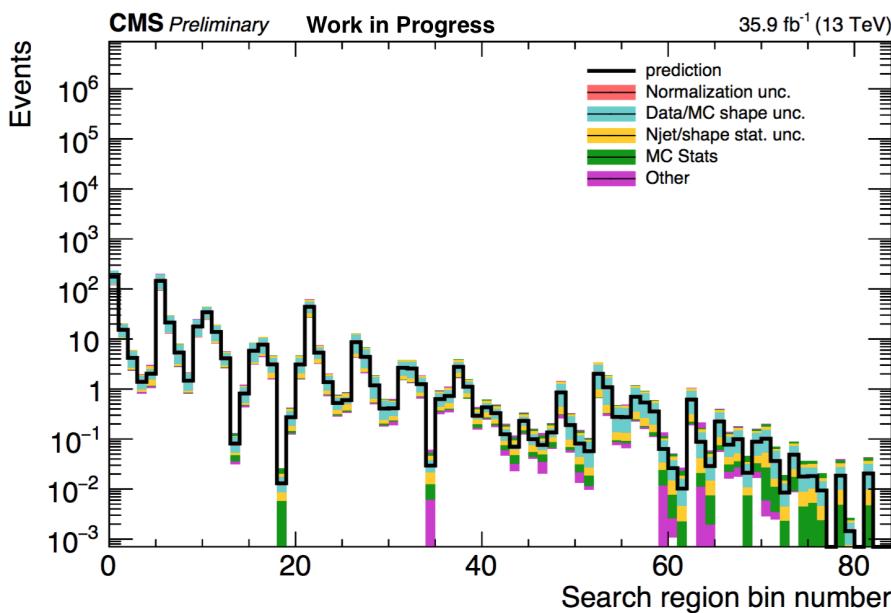


Figure 5.12: $Z \rightarrow \nu\bar{\nu}$ background prediction for all search bins, including the breakdown of the various uncertainties.

⁴⁶⁷ **Chapter 6**

⁴⁶⁸ **References**

⁴⁶⁹ [1] CERN, “Processing what to record?,” 2018.

⁴⁷⁰ [2] CMS, “The CMS Computing Project,” tech. rep., CERN, 2005.

⁴⁷¹ [3] Coursera, “Machine learning,” 2018.

⁴⁷² [4] A. S. Walia, “Types of optimization algorithms used in neural networks and ways to
⁴⁷³ optimize gradient descent,” 2018.

⁴⁷⁴ [5] M. Klasen, C. Klein-Bosing, and H. Poppenborg, “Prompt photon produc-
⁴⁷⁵ tion and photon-jet correlations at the LHC,” *JHEP*, vol. 03, p. 081, 2018.
⁴⁷⁶ [doi:10.1007/JHEP03\(2018\)081](https://doi.org/10.1007/JHEP03(2018)081).