Friedrich-Alexander-Universität
Faculty of Sciences

Bachelor's Thesis

Summer Semester 2024

Sebastian Weissbrod

Sensitivity of KM3NeT/ORCA6 to tau neutrino appearance with Graph Neural Network tau neutrino identification

Sensitivity of KM3NeT/ORCA6 to tau neutrino appearance with Graph Neural Network tau neutrino identification

Bachelor's Thesis in Physics

Presented by
Sebastian Weissbrod
27.11.2024

Erlangen Centre for Astroparticle Physics
Friedrich-Alexander-Universität Erlangen-Nürnberg

Supervisor: PD Dr. Thomas Eberl

Abstract

The observation of tau neutrino appearance in the flux of atmospheric electron and muon neutrinos by the KM3NeT/ORCA water Cherenkov detector will give insight into the standard three-flavor neutrino oscillation model. Graph Neural Networks have been shown to provide promising results in identifying tau neutrino interactions in KM3NeT/ORCA.
This Bachelor's thesis presents an implementation of a Graph Neural Network for the task of tau/nontau event classification that achieves a clear distinction between the event types on Monte Carlo simulation data of ORCA6, an early detector sub-array. This classification is used to replace the track/shower separation that is applied in the standard tau neutrino appearance analysis of ORCA6.
The scope of this thesis encompasses the full analysis pipeline, including preprocessing steps, training the network and compiling the data for the sensitivity study. Finally, first results of this sensitivity study and a comparison to the standard method are presented.

Zusammenfassung

Die Beobachtung von Tau-Neutrino-Appearance im Fluss der atmosphärischen Elektronen- und Myon-Neutrinos durch den KM3NeT/ORCA Wasser-Tscherenkov-Detektor wird Einsicht in das drei generationelle Neutrino-Oszillationsmodell geben. Es wurde gezeigt, dass Graph Neural Networks vielversprechende Resultate bei der Identifizierung von Tau-Neutrino-Wechselwirkungen in KM3NeT/ORCA erzielen.
Diese Bachelorarbeit präsentiert die Implementierung eines Graph Neural Networks für die Aufgaben von Tau/Nicht-Tau Ereignissklassifikation, welche eine klare Unterscheidung zwischen den Ereignisstypen auf Monte Carlo Simulationsdaten von ORCA6, einer frühen Detektor-Teilanordnung, erreicht. Diese Klassifikation wird genutzt, um die Spur/Schauer Unterteilung, die in der Standard Tau-Neutrino Appearance-Analyse von ORCA6 angewandt wird, zu ersetzten.
Der Rahmen dieser Arbeit umfasst die volle Analyse-Pipeline, inklusive Vorverarbeitung, Training des Netzwerks und dem Zusammenstellen der Daten für die Sensitivitätsstudie. Schließlich werden erste Resultate dieser Sensitivitätsstudie sowie ein Vergleich mit den standardmäßigen Methoden präsentiert.

Contents

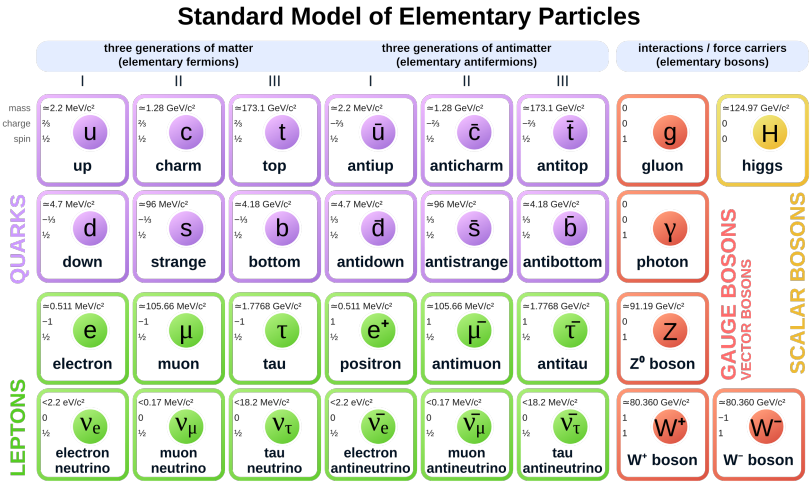


Figure 1: Representation of the Standard Model of particle physics. In the lower left corner are the leptons, including neutrinos and their antiparticles. Taken from [1].

1 Neutrino physics

Included in the Standard Model of particle physics (SM) as three generations of electrically neutral leptons, is the neutrino, denoted by 'ν'. The three generations consist of the electron neutrino ν_e, muon neutrino ν_μ, and tau neutrino ν_τ, see Figure 1.

Neutrinos are weakly interacting particles, carrying no charge, that were originally postulated to have zero mass. The latter came into scrutiny, when the famous Homestake-Experiment found discrepancies between the electron neutrino flux predicted by the solar model, and the actually observed electron neutrino flux in the experiment [2]. Eventually it was found [3] that neutrinos oscillate between different flavors, which requires non-zero neutrino masses. This prompted a multitude of further questions, that are still being investigated in today's experiments, like KM3NeT.

1.1 Neutrino Oscillations

Neutrino oscillation is the phenomenon of neutrinos changing their flavor while propagating through time and space.

Following the explanation of neutrino oscillations in vacuum in chapter 14 of [4], one can describe a given neutrino state either in a basis of three flavor $|\nu_\alpha\rangle$ or mass $|\nu_j\rangle$ eigenstates,

$$|\nu\rangle = \sum_{\alpha=e,\mu,\tau} c_\alpha |\nu_\alpha\rangle = \sum_{j=1,2,3} c_j |\nu_j\rangle \tag{1}$$

and finds, that the weakly interacting flavor eigenstates do not necessarily correspond to the mass eigenstates of the neutrinos. Hence a single flavor eigenstate can be expressed as a superposition of mass eigenstates:

$$|\nu_\alpha\rangle = \sum_j U^*_{\alpha j} \, |\nu_j\rangle \tag{2}$$

Or more generally, all states are expressed according to a neutrino mixing matrix U, called the Pontecorvo–Maki–Nakagawa–Sakata (PMNS) matrix:

$$\begin{bmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{bmatrix} = \begin{bmatrix} U_{e1} & U_{e2} & U_{e3} \\ U_{\mu 1} & U_{\mu 2} & U_{\mu 3} \\ U_{\tau 1} & U_{\tau 2} & U_{\tau 3} \end{bmatrix} \begin{bmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{bmatrix} \tag{3}$$

Since neutrinos weakly interact as flavor eigenstates, but propagate as mass eigenstates, one can examine the propagation of such a superposition

$$|\nu_\alpha(t)\rangle = \sum_j U^*_{\alpha j} \cdot e^{-i(E_j t)} \cdot |\nu_j(0)\rangle \tag{4}$$

with the energy E_j of the mass eigenstate being approximated as

$$E_j \simeq p + \frac{m_j^2}{2E}, \tag{5}$$

due to the relativistic nature of neutrinos.

Undergoing a Charged Current (CC) interaction, the probability of producing a lepton l_β will be:

$$\begin{aligned} P(\nu_\alpha \to \nu_\beta) = |\langle \nu_\beta | \nu_\alpha(t) \rangle|^2 = \delta_{\alpha\beta} - 4 \sum_{j<k}^{n} \mathrm{Re}\big\{ U_{\alpha j} U^*_{\beta j} U^*_{\alpha k} U_{\beta k} \big\} \sin^2 X_{jk} \\ + 2 \sum_{j<k}^{n} \mathrm{Im}\big\{ U_{\alpha j} U^*_{\beta j} U^*_{\alpha k} U_{\beta k} \big\} \sin 2X_{jk} \end{aligned} \tag{6}$$

where the argument $X_{jk} = \frac{\Delta m^2_{jk}}{4E} L$. It follows from the observation of neutrino oscillations that this probability is non-zero, which in turn means that neutrinos have different masses ($\Delta m^2_{jk} \neq 0$), i.e. a minimum of two neutrino mass eigenstates must have non-zero mass, and that mixing is taking place ($U_{\alpha j} U_{\beta j} \neq 0$).

Experimentally one can investigate appearance events $P(\alpha \to \beta)$ and survival events $P(\alpha \to \alpha)$, i.e. either the probability of a neutrino of flavor β to appear from neutrinos of flavor α, or the probability of a neutrino emitted as flavor α to still interact as flavor α, respectively.

1.2 Tau neutrinos

Of particular interest for this thesis is the appearance of tau neutrinos in KM3NeT/ORCA. The tau neutrino, with its existence suggested in 1975 and experimentally confirmed in 2000 [5], is the latest of the neutrinos to be discovered. The ν_τ CC cross-section being

notably suppressed in the few-GeV regime, as opposed to the cross-sections of the ν_e and the ν_μ, as well as the τ lepton itself imposing a relatively high production energy threshold of about 3.5 GeV, makes the ν_τ the most challenging neutrino to study [6]. Therefore properties like the ν_τ cross-section and the ν_τ oscillation parameters, i.e. the tau row of the PMNS matrix, remain significantly less constrained than those of the ν_e and ν_μ.

The ν_τ shows potential for new and interesting physics research, not only to further constrain these parameters, but also to investigate potential beyond Standard Model (BSM) physics, like sterile neutrinos and the matter asymmetry of the universe, dark matter neutrino coupling, as well as PMNS matrix non-unitarity. [7]

Any ν_τ detected by ORCA is expected to arise from neutrino oscillations. This is because ORCA targets atmospheric neutrinos in the few-GeV range (see section 2.1). These atmospheric neutrinos get produced in the decay of secondary particles formed by cosmic rays, interacting with air nuclei in the upper atmosphere. Due to the decay channels however, the atmospheric neutrino flux in the below TeV regime contains virtually no tau neutrinos [8]. So the only expected source of tau neutrinos in ORCA is from neutrino oscillations.

While the ν_μ disappearance channel is a good avenue for probing $\sin^2\theta_{23}$ and Δm^2_{31}, the ν_τ appearance, the specific identification of tau neutrino events, will provide a more direct insight into the interaction cross section of tau neutrinos on nucleons, as well as the potential non-unitarity of the mixing matrix [6]. The method by which the appearance of tau neutrinos is analyzed, will be discussed in section 4.

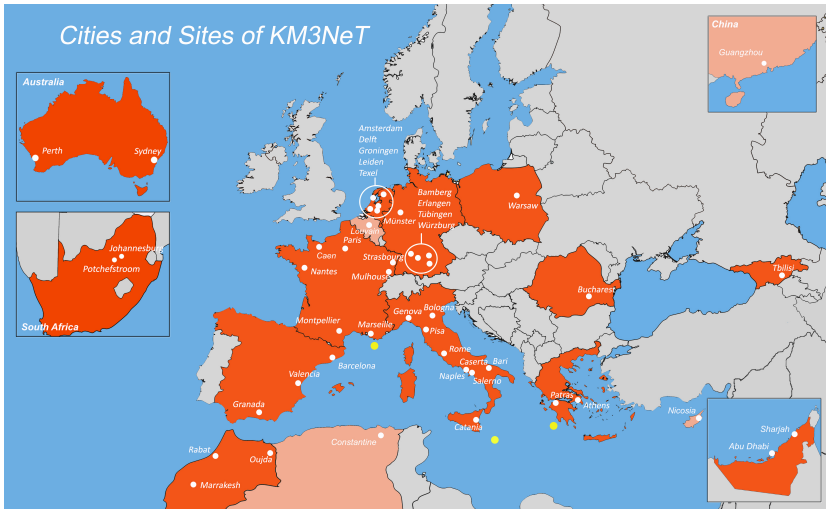


Figure 2: Cities with KM3NeT member institutes, white circles, and sites of the ARCA and ORCA detectors, yellow circles. [10]

2 KM3NeT/ORCA

The study of neutrinos is a complex task, that demands large and expensive infrastructure. This section focuses on the detector that is used in this thesis, discussing the research collaboration, responsible for its construction, the design of the detector and the physical principle used for detection. Finally there will be a discussion of the neutrino event topologies that can be expected inside the detector.

2.1 The KM3NeT collaboration and detectors

The Cubic Kilometer Neutrino Telescope (KM3NeT) experiment [9] is a collaborative research effort, building large volume water Cherenkov detectors in the Mediterranean Sea. Collaborators work at a multitude of international institutions, see Figure 2. The experiment consists of two detectors, ORCA and ARCA, currently being built at the KM3NeT-It and KM3NeT-Fr sites, respectively.

The main objective of the Astroparticle Research with Cosmics in the Abyss (ARCA) is the detection of high-energy neutrinos of cosmic origin. To this end the detector is configured sparsely and sensitive to TeV energies and above. ARCA will be able to explore the same signal as IceCube, but with a complementary field of view, including the galactic plane. ARCA is being built at the KM3NeT-It site.

The telescope relevant for this thesis is the more densely configured Oscillation Research with Cosmics in the Abyss (ORCA) detector. It is located at the KM3NeT-France site, 40 km offshore from Toulon, France, at a depth of 2450m. Targeting atmospheric neutrinos in the few-GeV regime, the main objective of ORCA is to measure atmospheric neutrino oscillations and to determine the neutrino mass hierarchy.

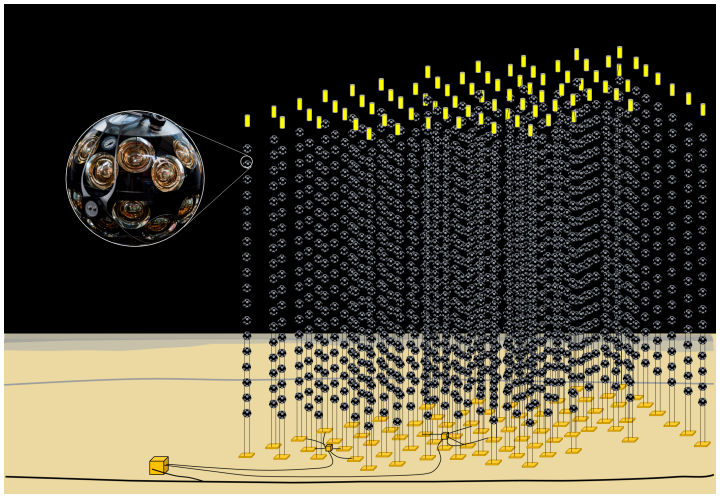


Figure 3: A KM3NeT Building Block. The Detection Units can be seen anchored to the sea floor and covering a volume with the Digital Optical Modules, one of which is displayed in more detail, in a magnification. Figure from [11].

2.2 Detector Design

Both ARCA and ORCA implement the same modular design, so called KM3NeT building blocks, which can be seen in Figure 3. These consist of 115 strings, known as Detection Units (DU), with one end anchored to the sea floor, the other end held straight and close to vertical by a buoy. Each DU comprises 18 Digital Optical Modules (DOM). A DOM is made up of 31 Photo Multiplier Tubes (PMT) and their readout electronics, all contained in a pressure resistant glass sphere.

The building blocks can be easily configured with different spacings between the DUs and DOMs to allow for the targeting of different neutrino energies. When fully realized, ORCA will consist of one, and ARCA of two building blocks.

The dense configuration of ORCA places strings with an average horizontal spacing of 20 m from each other. The strings are 200 m in height and the DOMs are spaced 9 m apart from each other. As of summer 2024 ORCA consists of 23 out of the planned 115 DUs [9].

2.3 Cherenkov Radiation

When an electrically charged particle moves faster than the local phase velocity of light, it will asymmetrically polarize the (dielectric) medium along its trajectory. This results in the conical emission of light under a characteristic Cherenkov angle θ_C:

$$\cos\theta_C = \frac{1}{\beta n} \tag{7}$$

The weak interactions from neutrinos in or near the detector create such charged particles. Their Cherenkov light can in turn be detected by the PMTs of the detector [12], illustrated in Figure 4. It is necessary to instrument a large volume, to get sufficient

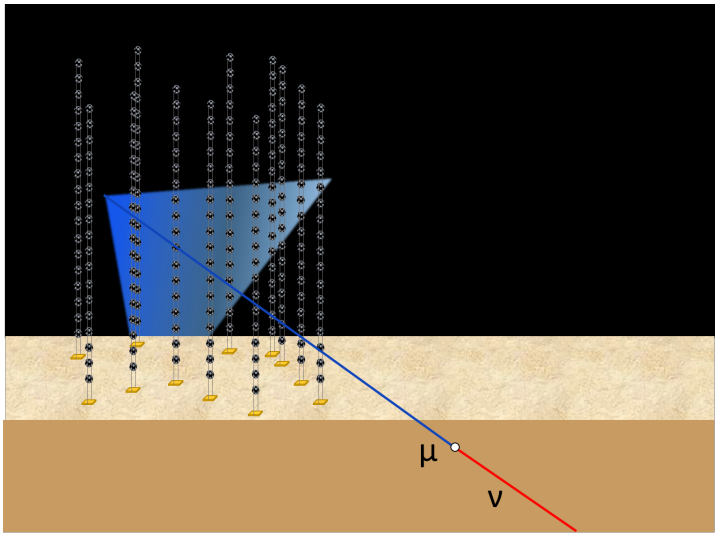


Figure 4: Depiction of a the Cherenkov radiation of a muon resulting from the interaction of a neutrino near the detector. The photons from the Cherenkov radiation reach the PMTs and can be detected. Figure from [13].

statistics for the neutrino interactions. The instrumented material also needs to be optically transparent. Therefore ice, like in the IceCube experiment, or sea water like in the ANTARES or KM3NeT experiments are a good choice, since they are already available in large quantities.

2.4 Neutrino event topologies

For the relevant GeV energies, a hadronic shower gets produced at every neutrino weak interaction vertex. Since these secondary hadronic particles have very short interaction lengths the emission is effectively point-like. Different flavors and decay channels show different event topologies. In the case of a Neutral Current (NC) interaction, just the hadronic shower is detectable and it is not possible to distinguish between the different flavors. In case of a Charged Current (CC) interaction, in addition to the hadronic shower, a lepton of the same flavor is produced at the interaction vertex, leading to differing topologies due to the associated lepton decay channels.

An $e^{\pm}$ will result in an electromagnetic shower of more $e^{\pm}$ and γ, giving ν_e^{CC} events a shower-like signature. In the below 100 GeV regime a $\mu^{\pm}$ already can propagate distances of up to 20-25 m in the detector, due to its near continuous, rather than instantaneous, energy loss. Therefore ν_μ^{CC} events, as well as atmospheric muon background events are track-like. The short lifetime of 2.9×10^{-13} s [4] of the $\tau^{\pm}$ makes its decay appear instantaneous to the detector. The main decay channels of the τ can be grouped into hadronic decays ($\approx 65\%$) and leptonic decays into either $e^{\pm}$ or $\mu^{\pm}$ ($\approx 17\%$ each). The hadronic decays result in a second hadronic shower and the decay into an electron again prompts an electromagnetic cascade. These channels hence have a shower-like event signature. The decay into a muon, on the other hand, results in a track-like signature, accordingly. [12]

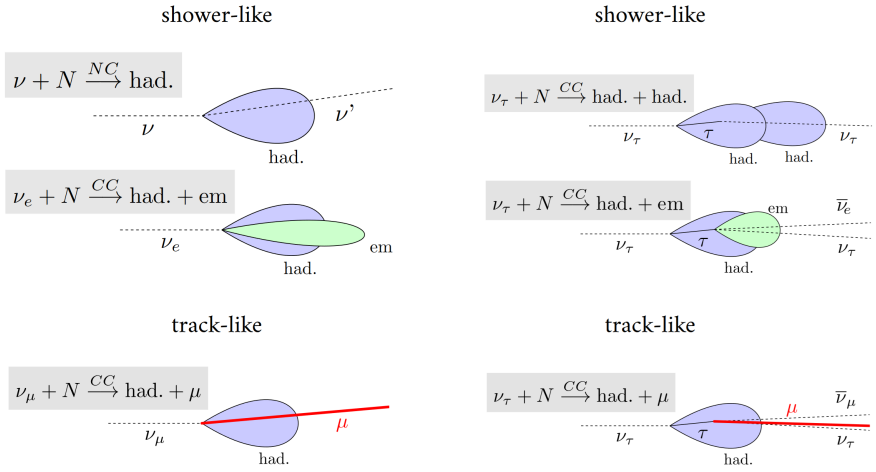


Figure 5: Neutrino interaction event topologies, separated by shower-like and track-like. On the left side are the "nontau events" and on the right side are the "tau events". Taken from [12].

The possible neutrino interactions and corresponding event topologies are presented in Figure 5. Topologies resulting from τ decays, linked to ν_τ are on the right side. These will be referred to as "tau events" in the following. Other events, including ν_τ^{NC}, since they can't be distinguished, will be called "nontau events". Considering the many similarities between tau and nontau event signatures the difficulty of tau event identification becomes apparent.

3 Particle Identification and Machine Learning

A large detector like ORCA produces huge amounts of data. Processing this data is often a complex, yet essential, step in obtaining results from the detector and gaining physical understanding. In the case of probing tau neutrino appearance, it is necessary to distinguish tau and nontau events. This task is known as Particle Identification (PID). Because this is such a complicated task one usually employs the help of Machine Learning (ML) algorithms. The following will be a short description of the classical reconstruction pipeline in KM3NeT/ORCA, used to perform this task. After which the theory and implementation of the method of GNNs will be discussed.

3.1 Classical Reconstruction Pipeline

A hit in the detector is determined by a PMT signal, that passes a preset threshold. A coincidence trigger of two or more of those hits, from different PMTs in the same optical module, within a small time frame, typically of the order of 10 ns, constitutes an event. Further filtering makes use of the orientation of the PMTs, as well as distinct optimized filters for different event topologies [9].

Energy and direction of the triggered events are reconstructed by maximum-likelihood based algorithms, for track-like and shower-like events respectively. The reconstructions determine event quantities like energy, direction and position from the PMT hits. For a more detailed explanation of the reconstruction algorithms see for example references [9, 14, 15].

To improve the quality of analysis not all triggered events are used. Instead some loose cuts are applied to the reconstructed events. These focus mainly on the position of event and subsequent Cherenkov light, to ensure good containment within the detector, as well as fit quality, to reject poorly reconstructed events. Also, events with a down-going direction are rejected, to reduce the amount of atmospheric muon background events. A more detailed explanation of the event preselection can be found in [12].

The PID is done by a random decision forest (RDF), that calculates track- and shower-probabilities for each event [16]. A RDF is a shallow machine learning technique, meaning it is trained on manually established high-level features [17].

But the hand selected features of shallow machine learning often don't represent the full feature space needed for an optimal solution. Therefore, with recent computational advances, it might be useful to examine different, more computationally complex, approaches, like deep neural networks. Although there exists no general definition, the main distinction between deep and shallow learning can be made not only on the number of layers, but also the way that features are learned [18].

3.2 Artificial Neural Networks

Machine Learning is the process of an algorithm improving its performance for a task, without being explicitly programmed on how to achieve said task. This encompasses a wide variety of techniques, ranging from simple Decision Trees to complex and deep ar-

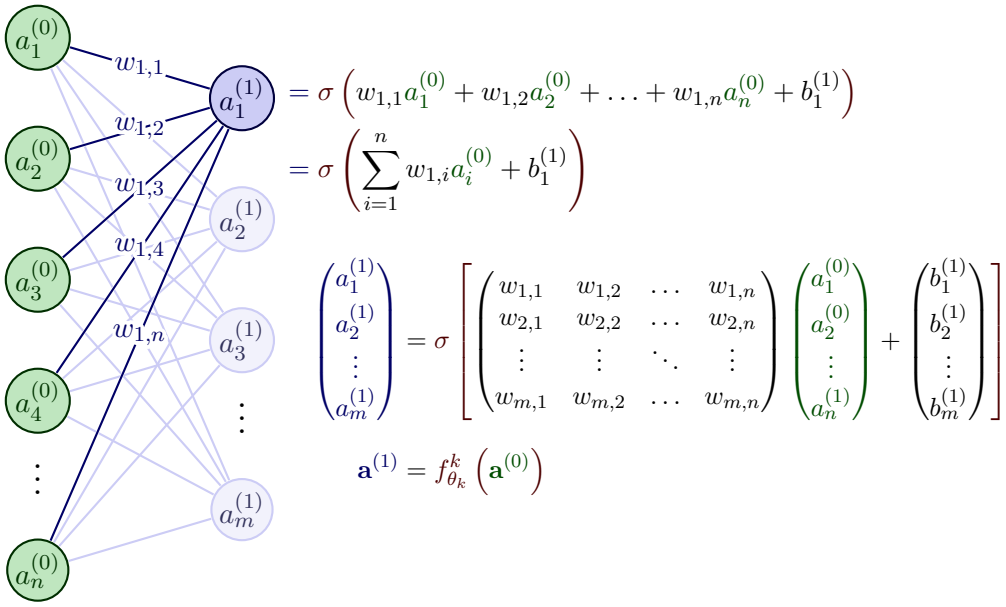


Figure 6: Schematic representation of the activation for the neuron $a_1^{(1)}$ in the first layer of a multi layer perceptron and the complete first layer respectively. Adapted from [20]

chitectures. The specific task of tau neutrino PID is a type of binary classification where the goal is to distinguish between two mutually exclusive labels *tau* and *nontau* [19].

A common ML technique is that of Artifical Neural Networks (ANN), where one models the flow of data through an artificial nervous system, comprised of neurons in multiple layers. In a feed-forward network information is passed from neurons of a previous layer to those of the next layer. This flow can be thought of as a complex, non-linear function $F(\mathbf{x})$, that maps values at the input layer, a real vector $\mathbf{x}$, to an output $\mathbf{y}$. A fully connected network, where every neuron in a layer is connected to every neuron in the next layer, is known as a multi layer perceptron (MLP). The layers between input and output are known as hidden layers. The activation $a_j^{(k)}$ of an arbitrary neuron j in the k-th layer can then be determined with an activation function σ:

$$a_j^{(k)} = \sigma \left(\sum_i^n w_{j,i} a_i^{(k-1)} + b_j^{(k)} \right) \tag{8}$$

This is schematically illustrated in Figure 6. The argument of the activation function is the sum over the activations $a_i^{(k-1)}$ of all n neurons in layer $(k-1)$, that are connected to j, scaled by a connection strength, the weight $w_{j,i}$. An additional term $b_j^{(k)}$, know as bias, may be associated with the neuron. All weights and biases of a layer will now be referred to as $\boldsymbol{\theta}_k$. The total activation of a layer k is then:

$$\mathbf{a}^{(k)} = f_{\boldsymbol{\theta}_k}^k (\mathbf{a}^{(k-1)}) \tag{9}$$

Where $f_{\boldsymbol{\theta}_k}^k$ represents the application of the activation function with its weights and

biases to every neuron. In order for the network to be able to learn complex features, the activation functions need to be non-linear. A common choice for its computational simplicity is the rectified linear unit (ReLU):

$$\sigma_{\mathrm{ReLU}}(x) = \max(0, x) \tag{10}$$

But there exist many different alternatives, used depending on the desired architecture, i.e. type and arrangement of layers and neurons [21]. Most complex architectures combine different activation functions across different layers.

Finally the output y in the last layer of a k layered network, with input at $k = 0$, is the composition of all previous activations:

$$\mathbf{y} = F(\mathbf{x}) = f^{k-1}\left(f^{k-2}\left(\ldots\left(f^{0}(\mathbf{x})\right)\right)\right) \tag{11}$$

To achieve good output performance then, the previous weights and biases need to be adjusted in such a way, that the outputs deviate as little as possible from desired values. This "learning" is done with an iterative processes called training. One can distinguish between supervised learning, where, like in this thesis, data is labeled with a true value, and unsupervised learning, where data is not labeled and the algorithm tries to find hidden patterns by itself. During training, $\boldsymbol{\theta}$ is changed in a way that minimizes the loss of the network. The loss is a quantity describing the difference between the current output $\hat{y}$ and the true value y for all N training samples. This of course depends on $\boldsymbol{\theta}$ since the output of a network is determined by its weights and biases. For binary classifications the loss is calculated by the binary cross entropy:

$$\mathcal{L}_{\mathrm{BCE}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i) \tag{12}$$

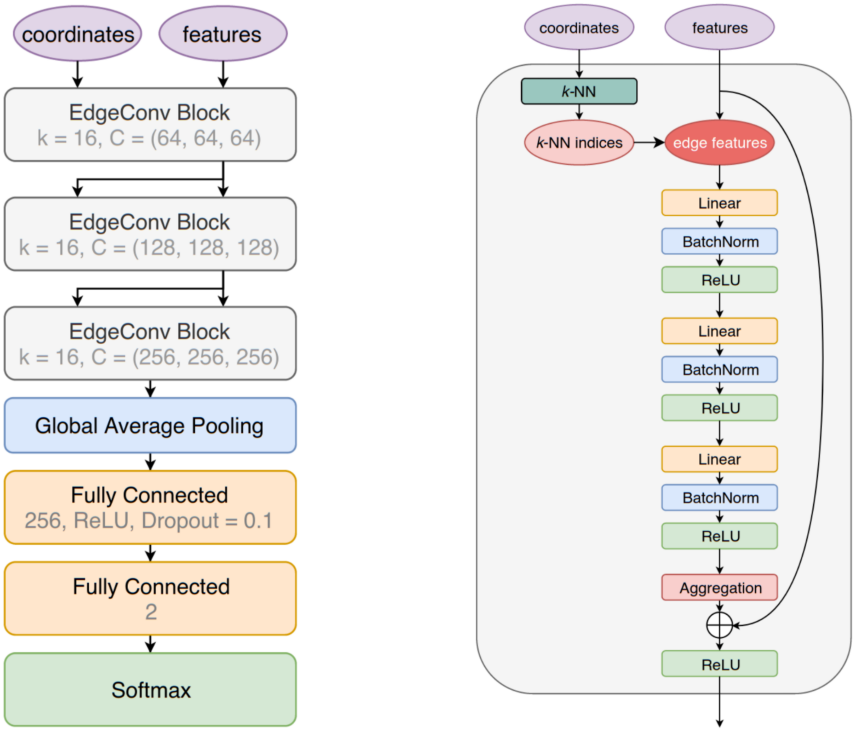
For a tau event, where the label $y_i = 1$, a value of $\log \hat{y}_i$ is added to the loss. In case of a good prediction $\hat{y}_i \to 1$ this term stays close to 0, and in case of a bad prediction $\hat{y}_i \to 0$ it approaches ∞. The same principle holds for a nontau event.

The loss function is minimized by a process called gradient descent, where in one time step t the weights and biases are shifted in the direction of the steepest descent, i.e. the negative gradient $-\nabla_{\boldsymbol{\theta}}$, of the loss function.

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \eta \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_{t-1}), \tag{13}$$

where η is the learning rate that controls how many iterations are needed to converge against a local minimum. If the learning rate is too large, there is a chance to shoot past a local minimum, but if it is too small, many computationally expensive iterations are needed to reach the minimum, which makes the learning process inefficient [19].

There exist further improvements to the concept of gradient descent, such as stochastic gradient descent, where average updates are determined from small batches of data, or the Adam optimizer [22], which adds a kind of momentum, based on the previous updates, to the gradient.



(a) Particle Net architecture. (b) EdgeConv block.

Figure 7: Schematic representation of the architecture used in the GNN to perform PID. (b) The Particle Net architecture, forming the basis for the GNN architecture used in this thesis. (a) The structure of the EdgeConv block, that performs a convolution on the edges of a graph. Graphics taken from [24]

3.3 Architecture for the Graph Neural Network

The approach to PID used in this thesis is that of a Graph Neural Network (GNN). In particular a specific network established in the Master's thesis of Lukas Hennig [19] has been re-implemented. Specifics about the necessity and process of re-implementation are discussed in section 5. The implementation is based on the *OrcaNet* framework in version *1.0.5* [23], with the added implementation of per event custom sample weights, and the Particle Net architecture (Figure 7a) proposed in [24]. This architecture implements the edge convolution (EdgeConv) operation that has been proposed in [25].

The individual hits that constitute a triggered event in ORCA, form a point cloud, that is characterized by their 3d position (x, y, z), the direction $(\hat{x}, \hat{y}, \hat{z})$ of the hit and the time t, that the hit occurred. A good representation for this kind of data are graphs.

Graphs are a collection of nodes, with a certain amount of node features F, and edges, that connect those nodes. Therefore each hit in an event can be identified with a node $\mathbf{n}_i \in \mathbb{R}^F$, with node features $(x, y, z, \hat{x}, \hat{y}, \hat{z}, t)$ and connected by edges to its k nearest neighbors. The graph representation of the event makes it a suitable input for a Graph Neural Network (GNN). The network may perform a convolution over the edges of the graph. For this purpose the EdgeConv block (Figure 7b) is used.

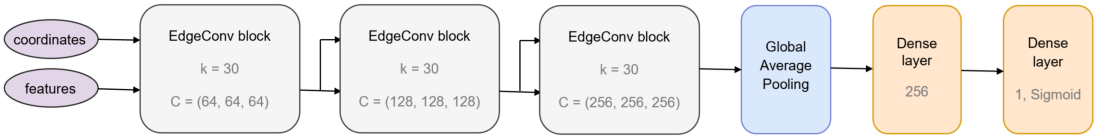


Figure 8: Architecture of the GNN used in A.2 of [19] and in this thesis.

First from the coordinates the k nearest neighbors of each node are calculated by the Euclidean distance: $\mathrm{d}s^2 = c^2\mathrm{d}t^2 + \mathrm{d}x^2 + \mathrm{d}y^2 + \mathrm{d}z^2$ [19].

Then the edge features are defined by an edge function $h_\theta\left(\mathbf{n}_i, \mathbf{n}_j - \mathbf{n}_i\right)$), where the neighbors are represented by their differences from the central point. h_θ is implemented as a three-layered MLP, the kernel network, with shared weights θ for all edges. This can be understood as a mapping $\mathbb{R}^F \times \mathbb{R}^F \to \mathbb{R}^{F'}$ from the original F node features to a new set of F' node features. The final update for a node $\mathbf{n}_i$ is then:

$$\mathbf{n}_i^* = \frac{1}{k}\sum_j^k h_\theta(\mathbf{n}_i, \mathbf{n}_j - \mathbf{n}_i) \tag{14}$$

where the mean of all neighbors is taken as the aggregation step [24].

Additionally the EdgeConv implements a shortcut for the original node features $\mathbf{n}_i$. They are fed through a single dense layer and batch normalized, to match the number of F' node features. Importantly they bypass the kernel network and are added element-wise to $\mathbf{n}_i^*$.

The architecture used in this thesis is the same architecture as detailed in Appendix A.2 of [19], which is closely inspired by the aforementioned ParticleNet architecture, but uses $k = 30$ nearest-neighbors and no dropout. The exact implementation is detailed in Figure 8.

4 Statistical methods

After the task of Particle Identification, the next step is to use this information to determine the sensitivity of KM3NeT/ORCA to the appearance of tau neutrinos. To this end, different hypotheses are tested by minimizing their log-likelihood ratios. To reduce the computational effort an Asimov dataset is employed. Both tools will be discussed in the following. Finally, the treatment of nuisance parameters and the software used for performing the minimizations are laid out. But first, a measure of the tau neutrino appearance needs to be introduced.

4.1 Tau normalization

To quantify the ν_τ appearance in KM3NeT/ORCA the tau normalization n_τ is used. n_τ is the ratio between the number of observed and expected tau neutrinos, $N_\tau^{(\mathrm{obs})}$ and $N_\tau^{(\mathrm{exp})}$:

$$n_\tau = \frac{N_\tau^{(\mathrm{obs})}}{N_\tau^{(\mathrm{exp})}} \tag{15}$$

It can be understood as a scaling factor of the total, in this case energy and zenith angle independent, number of tau neutrino events. That is, $n_\tau = 0$ for no ν_τ-appearance, $n_\tau = 1$ is expected from current standard models with PMNS matrix unitarity, and $n_\tau > 1$ would mean more ν_τ being observed, than conventionally expected [7].

The method used to estimate the statistical significance to the appearance of tau neutrinos achieved by ORCA, is based on studying the minimization of binned log-likelihood ratios of models with different tau normalizations with an Asimov dataset. A number of nuisance parameters that account for systematic uncertainties and their prior knowledge are also considered.

4.2 The Asimov dataset

A classical approach of studying the sensitivity to a binary hypothesis is to simulate a large ensemble of pseudo experiments and use them to determine the distribution of a test statistic, the likelihood ratio of the two hypotheses. Instead, an Asimov dataset is used, to estimate the median sensitivity. This approach was studied in [26]. The Asimov dataset is constructed to be the most representative outcome, i.e. the average experiment. That means, when fitting with the hypothesis used to construct the Asimov dataset, its values can be recovered exactly.

For good enough statistics on the likelihood ratio, often pseudo experiments in the order of several thousands are needed. The Asimov dataset approach has the distinct advantage of foregoing the computationally expensive production of those pseudo experiments. On top of that, one can only determine the sensitivity for a single value of n_τ, as a binary hypothesis, at a time. So in order to estimate the overall sensitivity of the detector, several values of $n_\tau \neq 1$ are tested, each of which would

require its own set of pseudo-experiments. In contrast, the Asimov dataset is constructed once and can then be used for all estimations.

4.3 Profile likelihood ratio

The function $\mathcal{L}_H(\boldsymbol{d} \,|\, \boldsymbol{\eta})$ denotes the likelihood of measuring data $\boldsymbol{d}$, under the assumption of a hypothesis H, and with nuisance parameters of values $\boldsymbol{\eta}$. In the case of the Asimov dataset, instead of fluctuated pseudo-data $\boldsymbol{d}$, the expectation values $\boldsymbol{\mu}_\mathrm{A}$ based on the average experiment are used. Evidently, this gives a likelihood of one, when the hypothesis corresponds exactly to the best-fit values. The likelihood can be split into the statistical likelihood $\mathcal{L}_\mathrm{stat}$ and the systematic likelihood $\mathcal{L}_\mathrm{syst}$, which accounts for the priors on the nuisance parameters.

$$\mathcal{L} = \mathcal{L}_\mathrm{stat} \times \mathcal{L}_\mathrm{syst} \tag{16}$$

As a test statistic the profile log-likelihood ratio $-2\Delta \log \mathcal{L}$ is defined:

$$-2\Delta \log \mathcal{L} = -2 \left[\frac{\min \left(\ln \mathcal{L}_H(\boldsymbol{\mu}_\mathrm{A} \,|\, \boldsymbol{\eta}) \right)}{\min \left(\ln \mathcal{L}_{H_0}(\boldsymbol{\mu}_\mathrm{A} \,|\, \boldsymbol{\eta}) \right)} \right] \tag{17}$$

for the Asimov dataset approach. Here $\mathcal{L}_H$ is evaluated for a fixed test value of n_τ and $\mathcal{L}_{H_0}$ is the likelihood evaluated at the best-fit value, where n_τ is left free. The log-likelihood ratio tests the incompatibility of this fixed value of n_τ with the Asimov dataset and can also be written as a difference of χ^2.

$$\Delta\chi^2 = -2\Delta \log \mathcal{L} \tag{18}$$

In the simplified case of one-dimensional Poisson distributed data in each bin, the statistical part of each of the χ^2_H is calculated in the following way:

$$\chi^2_{\mathrm{stat},H}(\boldsymbol{\mu}_\mathrm{A} \,|\, \boldsymbol{\eta}) = -2 \ln \mathcal{L}_\mathrm{stat} = \sum_{i=1}^{I} \left(N_{H,i} - N_{\mathrm{A},i} + N_{\mathrm{A},i} \ln \left(\frac{N_{\mathrm{A},i}}{N_{H,i}} \right) \right) \tag{19}$$

Where $I = I_E \times I_{\cos\theta} \times I_\mathrm{PID}$ is the total number of bins, which corresponds to the combination of the bins of all PID classes. For each class, the events are sorted into a two dimensional histogram of reconstructed energy and direction (represented by the cosines of the zenith angles). $N_{H,i}$ is the expected number of events in the i-th bin for the hypothesis H, based on the modeled neutrino interaction rate (including detector dependent and independent parts). And finally, $N_{\mathrm{A},i}$ is the number of events in the i-th bin in the Asimov dataset [18]. The $-2 \ln \mathcal{L}_H$ terms can be expressed as χ^2_H, since they are equal up to a term, that vanishes in the ratio [15].

Note that $N_{H,i}$ depends on the hypothesis that is being tested, so it will change for different values of n_τ, while $N_{\mathrm{A},i}$ stays consistent across all tests. The tau normalization is profiled for several fixed values in Equation 17. A large value indicates large incompatibility and the minimum is expected to be found as $-2\Delta \log \mathcal{L} = 0$ for $n_\tau = 1$, which matches the Asimov dataset exactly.

Additionally some of the systematic nuisance parameters η included in the fit, are constrained with Gaussian-like prior knowledge:

$$\chi^2_{\text{syst}}(\boldsymbol{\eta}) = \sum_{p \in \mathcal{P}_{\text{prior}}} \frac{(\eta_p - \mu_p)^2}{\sigma_p^2} \tag{20}$$

Each constrained parameter is associated with a Gaussian probability distribution of mean μ_p and standard deviation σ_p. The prior likelihood terms increase significantly, when values are far from the Gaussian mean, which represents the knowledge of the nuisance parameters from external experiments, acting as a penalty to the overall minimization. [15].

4.4 Description of nuisance parameters

Table 1 shows a list of parameters considered for the fit of the tau norm, their starting value, as well as the prior knowledge, when applicable.

Oscillation parameters are taken form NuFIT v5.0 [27]. Most of them are fixed, but Δm^2_{31} and θ_{23} are included in the fit, due to ORCAs sensitivity to them.

Uncertainties on the atmospheric neutrino flux are modeled by means of its shape, through the spectral index and the ratio of the number of horizontal and vertically up-going neutrinos ν_{hor}/ν_{ver}. These are small perturbations on the shape of the flux, acting as a tilt on the energy and cosine of the zenith angle, respectively [15].

Also considered are uncertainties of the flavor composition of the atmospheric neutrino flux. These constitute the ratio of muon and electron neutrinos ν_μ/ν_e, as well as the ratios of neutrinos to antineutrinos $\nu_\mu/\bar{\nu}_\mu$, and $\nu_e/\bar{\nu}_e$. The scale factor $\mathrm{S_{NC}}$ accounts for uncertainties of the all-flavor neutral current interaction. The parameter named "JSirene Norm" addresses potential differences that might appear due to different light generators being used for low- and high-energy simulations. The energy scale models more general uncertainties, like the PMT efficiency and water properties [6].

Finally, event numbers can be scaled globally with the overall Flux Norm, as well as in specific classes with the Track and Shower Norm, corresponding to said PID classes in the standard analysis. But since they are simple scaling factors, that are agnostic to the actual definitions of the classes, they can equally be used for the tau and nontau classes. Overall, the scaling factors are used to deal with any unforeseen uncertainties, allowing the best fit to achieve very good values. Naturally, since there can be no external knowledge about these factors, they are fitted without prior and can in principal reach any arbitrary value.

4.5 The Swim framework

The analysis detailed in this section is implemented in a software package called *Swim* [28]. *Swim* is a C++ framework, that was originally developed by Simon Bourret in his PhD-thesis [15] and then further maintained by the KM3NeT collaboration. It is used for neutrino oscillation analyses and sensitivity studies. The code is built on ROOT

Parameter	Starting value ± prior
$\Delta m^2_{31}[10^{-3}\,\mathrm{GeV}^2]$	2.517
θ_{23} [°]	49.2
Energy slope (Spectral index)	0.0 ± 0.30
Zenith slope (ν_{hor}/ν_{ver})	0.0 ± 0.02
$\nu_\mu/\bar{\nu}_\mu$	0.0 ± 0.05
$\nu_e/\bar{\nu}_e$	0.0 ± 0.07
ν_μ/ν_e	0.0 ± 0.02
S_{NC}	1.0 ± 0.20
Energy scale	1.0 ± 0.09
JSirene Norm	1.0 ± 0.5
Flux Norm	1.0
Track Norm	1.0
Shower Norm	1.0

Table 1: Parameters used for the fit of the tau norm. When applicable the prior is stated. In those cases the starting value corresponds to the central value of the prior knowledge.

[29] and uses *OscProb* to calculate the oscillation probabilities [30]. The definition of the tau and nontau classes will be discussed in section 5.5.

The events that are used for the analysis with *Swim* have to be provided to the program as a ROOT file. This file will be referred to as the "Swimfile" in the following.

5 Processing pipeline and sensitivity to tau neutrino appearance

The goal of this thesis was to develop a data analysis pipeline for studying the sensitivity of KM3NeT/ORCA6 to ν_τ appearance, when the standard PID, a track/shower classifier based on RDFs, is replaced by a GNN that is trained as a tau/nontau classifier. While earlier chapters provided the necessary theoretical framework, this chapter will focus on the specifics of the implementation of the GNN, the data preprocessing pipeline, and the subsequent sensitivity analysis based on the GNN classification results.

5.1 Event preprocessing

Data in KM3NeT is taken on a "run-by-run" basis, where each run represents a period of a few hours. For every run, general parameters describing the detector's status during that run are measured. These parameters consist of information about the detector geometry, calibration data and water properties, among others. The Monte Carlo (MC) simulations used for detector analyses are also based on runs. The measured parameters describing a run are used as inputs for the simulation chain. For this work, MC simulations for ORCA6, i.e. the sub-array of ORCA consisting of 6 DUs, were used, in accordance with the analysis in [6, 31] and the GNN trained in [19]. The dataset for the analysis, consists of more than 12 million simulated events in 2379 runs, mostly from version v7.2 MC simulations, but also including events from v7.1, where applicable. This dataset will now be referred to as the "rbr dataset".

Raw event data is saved as ROOT files. Before being used, these need to be preprocessed into a different format, specific for *OrcaNet*. This is done in a two step process. First, the ROOT files are converted to HDF5 files using the *h5extractf* tool from the *km3pipe* package [32]. Next, the events, still containing the raw hit information, are converted into a graph representation, required for the use with the GNN. This task is accomplished with the *OrcaSong* package [33], which has been extended to extract some additional meta information, that is used later on, to uniquely identify the events.

The data processing was implemented as a pipeline, using Snakemake [34], a workflow management system built on Python. The use of Snakemake allows a generalization of the workflow, where individual steps are automatically parallelized, when applicable. The pipeline and scripts are backed up on the KM3NeT GitLab website and can be found at [35].

To avoid overfitting, i.e. the network training on some features that are only inherent to the training dataset, but don't represent generic input data, one typically evaluates the performance of the network in a separate validation step after each epoch. The data used for the validation is not contained in the training dataset. And more generally, it was decided not to use any of the events from the rbr dataset, which is used for the sensitivity analysis, for either training or validation. Instead a different dataset, which had been produced by Lukas Hennig for his Master's thesis [19] was utilized. This dataset is referred to as the "flat dataset", due to its flat energy spectrum. It was originally intended to be used as an extension of the rbr dataset, to boost the statistics of tau events for the training. Since the flat dataset is based only on one run, it does

not properly reflect the run by run approach, which could lead to discrepancies in performance. Nevertheless, it was deemed sufficient enough to be used for the training.

The output of the simulation chain produces multiple files. The events from the flat dataset were split into a training and a validation dataset and the individual files contained therein, where concatenated into one singular file per dataset. To reduce the chance that the network would train to reproduce specific features inherent to the dataset, but not the data itself, e.g. the order of the events in the dataset or similar, the events were randomly shuffled. Both the concatenation and the shuffling were achieved with tools implemented in *OrcaSong*.

5.2 Sample weights

Custom sample weights were used to rescale the composition of the events, to approximate the expectation for measured data. Sample weights are calculated in accordance to the weights calculated by [19]. More specifically weights are calculated using the Honda weighting approach.

Honda weighting refers to an approach, where events are weighted by the ratio of the physically expected flux at the detector to the flux used for the generation of the event in the simulation. The name Honda weighting is explained by the model used to model the flux, which has been simulated by Honda et al. [36]. The specific flux table used for the sample weight calculation is the azimuth- and time-averaged flux $\phi^{\mathrm{H}}_{\alpha}(E, \cos\theta)$ at the Fréjus site, located close to KM3NeT/ORCA. Note the energy E and zenith angle $\cos\theta$ dependence, that is also present in the simulated dataset. α refers to the two flavors e, μ that get produced in the atmosphere.

To determine the total expected flux at the detector $\phi^{\mathrm{D}}_{\alpha}(E, \cos\theta)$, the effect of neutrino oscillations, as discussed in section 1.1, has to be taken into account. Suppose the probability of a neutrino oscillating from flavor ν_α into flavor ν_β is $P_{\alpha\to\beta}(E, \cos\theta)$, then $\phi^{\mathrm{D}}_{\alpha}(E, \cos\theta)$ can be calculated as:

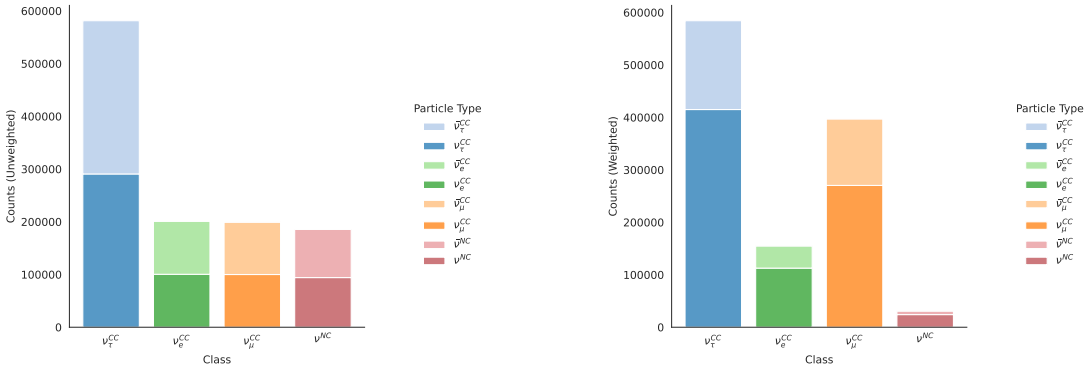
$$\phi^{\mathrm{D}}_{\alpha}(E, \cos\theta) = P_{e\to\alpha}(E, \cos\theta) \cdot \phi^{\mathrm{H}}_{e}(E, \cos\theta) + P_{\mu\to\alpha}(E, \cos\theta) \cdot \phi^{\mathrm{H}}_{\mu}(E, \cos\theta) \tag{21}$$

The oscillation probabilities are calculated with *OscProb* [30], which was accessed with *km3services* [37]. Finally, the Honda weight $w^{H}_{r,\alpha,k}$ for an event k in run r, and of flavor α is calculated as:

$$w^{H}_{r,\alpha,k} = \frac{\phi^{\mathrm{D}}_{\alpha}(E_{r,\alpha,k}, \cos\theta_{r,\alpha,k})}{\phi^{\mathrm{Gen}}_{r,\alpha}(E_{r,\alpha,k}, \cos\theta_{r,\alpha,k})} \tag{22}$$

Where $\phi^{\mathrm{Gen}}_{r,\alpha}$ is the flux used in the simulation of the event. A more thorough discussion of the way the sample weights were calculated, particularly with regards to ν^{NC} events, and the lifetime of the runs, can be found in [19].

This approach is also referred to as "physical weighting". For the training, the weights were also used to balance the tau $\mathcal{T}$ and nontau $\mathcal{T}'$ classes, opting to boost the statistics of tau events, as compared to the expected flux:



(a) Composition of the flat dataset, before applying the sample weights.

(b) Composition of the flat dataset, after applying the sample weights.

Figure 9: Impact of the sample weights on the flat dataset used for training. The individual rates of the particle types are scaled according to the expected physical rates at the detector. Since the classes are set to be balanced, the ratio of tau events to nontau events is kept to one.

$$w^{H,\beta}_{r,\alpha,k} = N \cdot w^{H}_{r,\alpha,k} \cdot \begin{cases} \left(\sum_{r,k} \sum_{\alpha \in \mathcal{T}} w^{H}_{r,\alpha,k}\right)^{-1} \cdot \beta, & \text{for } \alpha \in \mathcal{T} \\ \left(\sum_{r,k} \sum_{\alpha \in \mathcal{T}'} w^{H}_{r,\alpha,k}\right)^{-1} \cdot (1-\beta), & \text{for } \alpha \in \mathcal{T}' \end{cases} \tag{23}$$

With $\beta = 0.5$ to achieve the same number of tau and nontau events. The composition of the flat dataset and the effects of the sample weights can be seen in Figure 9.

5.3 Training the GNN

The network is trained using the architecture detailed in section 3.3. Since the training and validation datasets changed, compared to the initial training done by Lukas Hennig, the network had to be retrained. The progression of the training can be seen in Figure 10. The loss function for the training has extreme fluctuations, since the loss is plotted for each batch. A better measure for the epoch by epoch improvement of the network is the loss of the validation dataset, which can be seen steadily declining in the beginning of the training process until reaching a minimum after 15 epochs. Since the network didn't show any improvements on the validation dataset for a few epochs thereafter, this was chosen as a stopping point for further training. The loss of the training dataset seems to keep slightly decreasing, which can be interpreted as a sign of overfitting, giving another reason against further training. For the analysis, going forward, the model in epoch 15 will be used.

A further evaluation of the performance of this model is done by plotting a precision-recall curve. This concept is widely used as a metric for binary classifiers and is also known as purity and efficiency. Precision, or purity, refers to the ability of the classifier to only give those events a tau label, that are indeed actually tau events. It can therefore be understood as the ratio of true positives to all samples classified as positive.

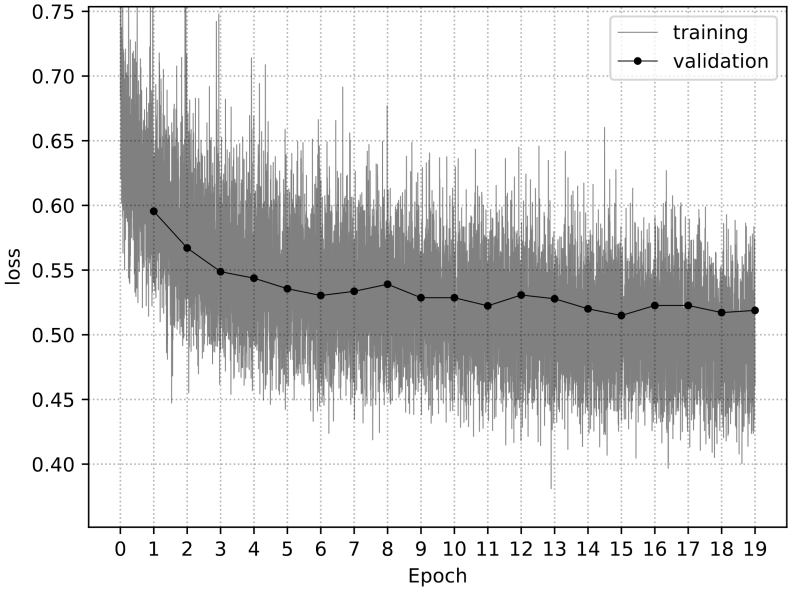


Figure 10: Progression of the loss function during the training.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{24}$$

Recall, or efficiency, refers to the proportion of tau events, that have been identified by the model. This can be expressed as the ratio of true positives to all samples that are positive.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{25}$$

Here TP, FP and FN refer to true positive, false positive and false negative, respectively.

The precision-recall curve for the best model can be seen in Figure 11. The higher fluctuations for small recall values can be explained by the naturally reduced statistics, which is also why recall values under 1% were completely omitted from the plot. The model clearly performs significantly better than just guessing randomly, confirming, that the GNN did indeed successfully learn to differentiate between tau and nontau events. The model achieves a precision of 22.3% at a recall of 30%. This performance is comparable, but slightly better, to the one achieved by the best (flat dataset only) model in Lukas Hennig's Master's thesis [19], which achieved a precision of 18.0% for 30% recall, with slightly different hyperparameters. While it is surprising, that the model trained for this thesis performs better, both it and the model it is compared to, were only trained with their respective configuration of hyperparameters one time, which leaves the evaluation vulnerable to random fluctuations, as was also pointed out in the Master's thesis.

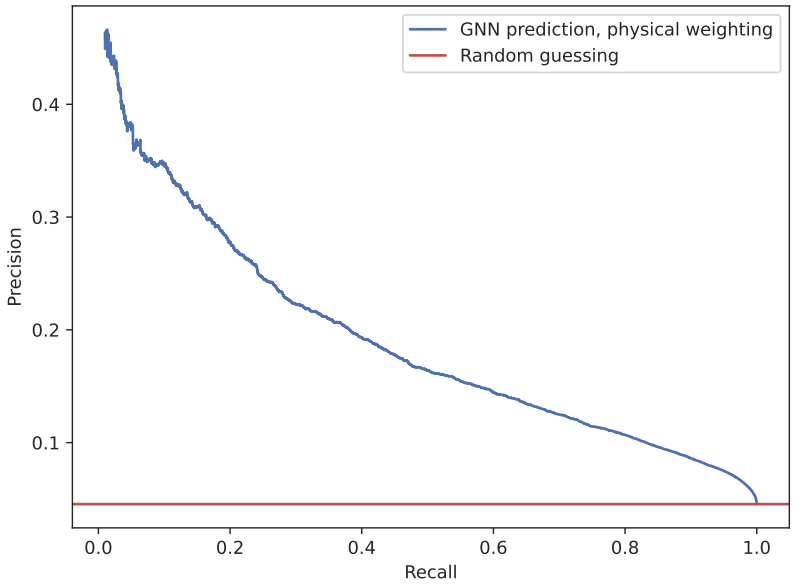


Figure 11: Precision-recall curve of the validation dataset, with physical weighting only.

5.4 Predictions on the rbr data

The rbr dataset was also fed through the same event preprocessing pipeline. The trained GNN was then used in its best performing epoch, to predict on the resulting files in the graph representation, giving each event a tau/nontau score, that represents the networks confidence about whether the event should be classified as tau or nontau.

A problem in this process arose, when trying to ensure maximum comparability to the earlier analyses [6, 31] with track/shower classification, and matching the events included in those analyses. Since the event preprocessing pipeline defined in section 5.1 was executed locally, but the long term storage of the MC simulation files is at a remote location, at CC-IN2P3 (Le Centre de Calcul de l'IN2P3), in Lyon, the files had to be transferred first, which was a time-intensive process. This meant that, due to the submission deadline, it was not possible to include all files used in the previous analyses in the analysis in this thesis. The available portion, that could be used, will now be referred to as the reduced rbr dataset. Its composition can be seen in Figure 12.

A histogram with the results from the predictions of the GNN can be seen in Figure 13a. The peak for the nontau events is close to a value of 0, i.e. a nontau prediction, but there is also quite a substantial amount of predictions between 0.2 and 0.8. For the tau events the peak is at around 0.8, which still shows, that the model was able to clearly distinguish tau events from nontau events, but is slightly further from the optimum value of 1. Predictions for tau events at lower values fall off faster. The considerable overlap of tau and nontau events at medium prediction values is a sign, that an optimal classification was not possible.

The prediction on the validation dataset in Figure 13b shows similar performance. Note that the width of the bins for this plot is doubled. This was done to mitigate the more prevalent statistical fluctuations, resulting from the relatively low number of

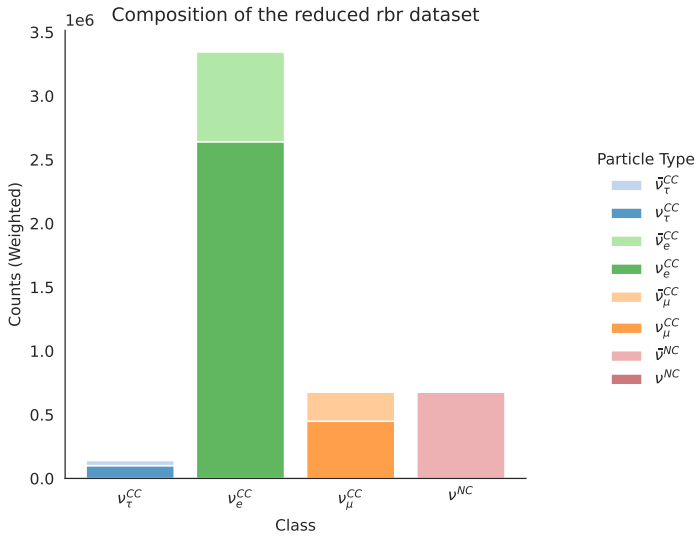


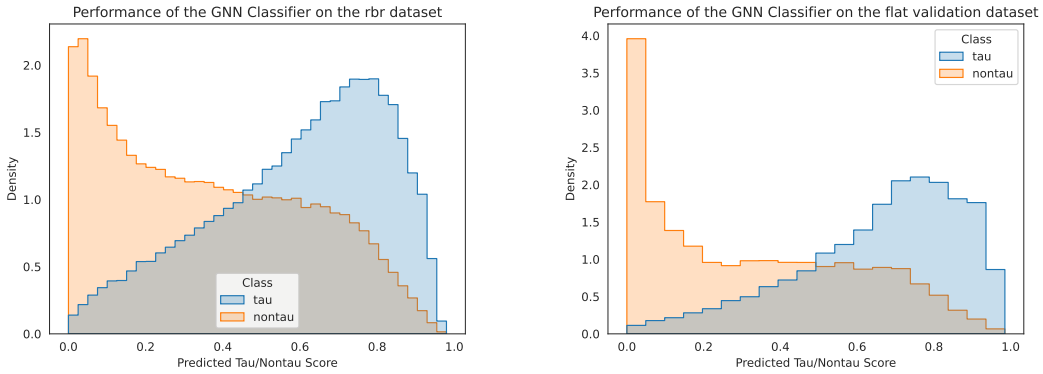
Figure 12: Composition of the reduced rbr dataset with physical weighting.

events in this dataset. Another notable difference are the steeper peak for the nontau class, as well as the reduced number of predictions in the central region for both classes. This might also be explained by the lower event numbers, but could as well be a hint to a potential difference in performance between the two simulation approaches, which also leads to different interaction type composition and energy and direction distribution of the events.

5.5 Analysis of the tau normalization

Having produced those predictions, the next step is to append them to the Swimfile. For this the information extracted with *OrcaSong* is used to uniquely identify the events and match each prediction to the correct event contained in the Swimfile. Then using *uproot* [38] all predictions can be appended as an additional branch to the information already contained in the file. As explained before, not all events were available. Most notably all atmospheric muon events were missing. For the events in the Swimfile without prediction a value of 0 was chosen, since no prediction by the GNN was exactly 0. Those events could then be easily disregarded by the general cuts for the analysis, applied by *Swim*. The execution of the fit was done within a singularity container to ensure equality to other analyses.

The configurations for the fit can be defined with *json* files, forgoing the need to modify the code itself. The handling of nuisance parameters was already explained in section 4.4. Another aspect addressed are the energy and zenith angle bins. These have already been optimized for previous analyses and are adopted from there. The cosine of the zenith angle $\cos\theta$ is divided into 10 equally spaced bins between values of $\cos\theta = -1$, i.e. vertically up-going, passing through the earth, and $\cos\theta = 0$, i.e. horizontal neutrinos. The reconstructed energy is binned between 2 GeV and 1 TeV, with 15 optimized bins on a logarithmic scale [6].



(a) Run-by-run dataset

(b) Validation dataset

Figure 13: The results of the tau/nontau classification on a) the reduced rbr dataset and b) the validation dataset. The area under the histograms integrates to 1. Also note the wider bins in the right plot. A value of 1.0 corresponds to an optimally tau-like prediction.

Finally, the classes are characterized by cuts that are being applied to the pool of events contained in the Swimfile. While previously for the track/shower classification, there were classes for showers, as well as low and high purity tracks, the initial analysis performed in this thesis only uses one class each for tau and nontau events. As a comparison the track/shower analysis was repeated with two classes, merging the low and high purity tracks. For the definition of the tau/nontau classes, the basic cuts on the reconstructed energy, zenith angle and muon background rejection were left untouched. Thus the only change was replacing the track score by the tau score. Refer to Appendix A for an example of a definition.

The analysis is based on the log-likelihood minimization explained in section 4.3. The Asimov dataset is constructed for the assumption that $n_\tau = 1$. To estimate the sensitivity of the detector and classification, one probes different values of n_τ to determine the confidence with which one can reject this hypothesis. In this analysis the tau normalization was profiled for 21 equally spaced points in the range $0 \leq n_\tau \leq 2$. Additionally, another fit is produced, where the value of n_τ is left free, allowing it to reach the theoretical best fit value. In the case of an Asimov dataset with MC data, the best fit is always expected to be at $n_\tau = 1$, but when analyzing real data, this step is significantly more crucial, as the best fit is generally not known.

The tau class was defined as all events reaching a prediction value over a certain threshold. It has to be noted again, that, within the time frame of this thesis, only a fraction of the events was available for the analysis. While the same restrictions that this imposed on the tau/nontau classes, were also applied to the track/shower classes, the missing events still diminish the significance of the analysis. Especially the full exclusion of the muon background could have a large impact.

To determine the optimal value for the class threshold, a comparison of different thresholds was conducted. The results of this comparison are displayed in the right plot of Figure 14. High thresholds, in particular at 0.9 and 0.8, performed the best, with the

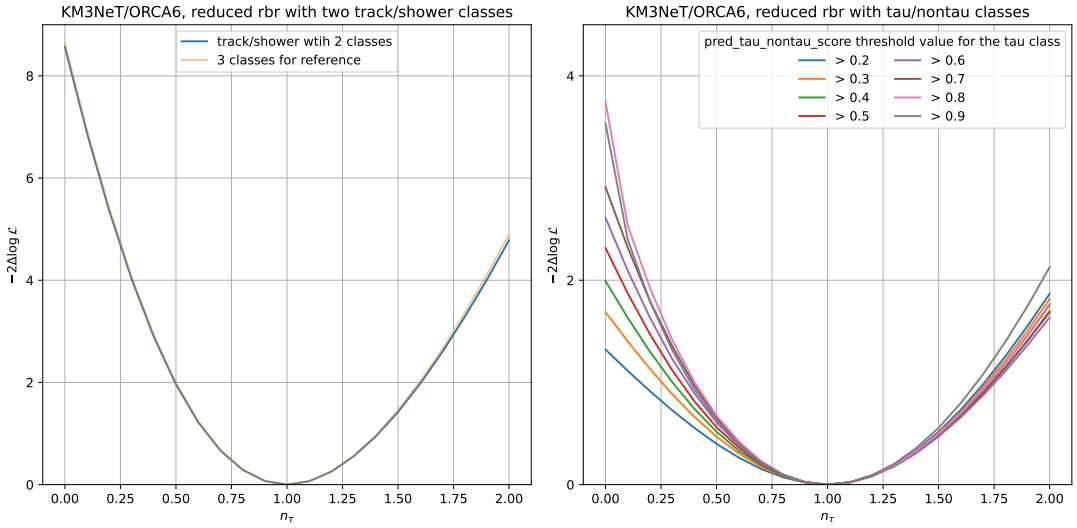


Figure 14: Comparison of the tau norm fits for track/shower classes (left) and tau/nontau classes (right). A multitude of cut values were chosen for the tau/nontau class definitions.

former achieving better performance for high values of n_τ. The left plot however shows a comparison to the track/shower classification with two and three classes, with the same reduced rbr dataset. Noting the different scaling of the y-axis, the track/shower classification performed significantly better. There are several potential reasons for this.

First, there are the aforementioned problems with the incomplete dataset. Additionally, the track/shower classes have been thoroughly optimized, while the tau/nontau classes adopted most of those class definitions, not being optimized with regards to reconstructed energy and zenith angle, as well as background rejection cuts, themselves. The large overlap of tau and nontau events for medium prediction values in Figure 13, suggests that the tau/nontau analysis might benefit from a third class for those middle prediction values, although ORCA6 doesn't have sufficient statistics for this [39]. Another potential issue arises from the flat dataset used for the training, that was also only simulated for one run, as discussed before. Furthermore, it was rather small, only comprising around 800,000 events, and could therefore have limited the performance, due to the reduced statistics.

It should also be considered as a possibility, that the GNN did not identify any distinctive features to differentiate tau event topologies from nontau event topologies, beyond those already utilized by the shallow classification techniques, outlined in section 3.1, to differentiate track and shower topologies. As a result, the tau/nontau classifier may in practice act as a track/shower classifier. In Figure 15 the performance of the GNN is further separated into the different particle and interaction types. Here, a clear difference between the performance for the track-like muons, which are predicted as nontau with very high certainty and the shower-like electron charged current ν_e^{CC} and neutral current ν^{NC} events, for which the prediction practically has no peak and is spread relatively evenly across a large range of values between 0 and 0.8, exists.

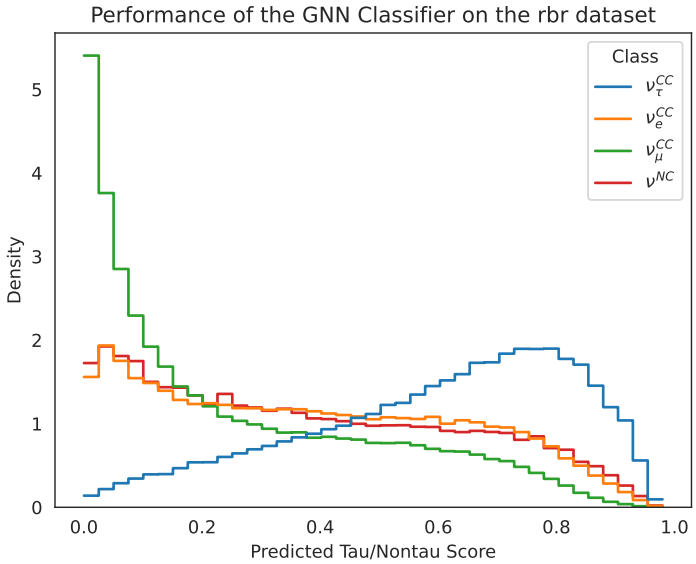


Figure 15: The result of the tau/nontau classification on the reduced rbr dataset, split by interaction types. Areas under the histograms integrate to 1. A value of 1.0 corresponds to an optimally tau-like prediction.

6 Summary and Outlook

This thesis details the development of a data analysis pipeline designed to investigate ν_τ appearance sensitivity with KM3NeT/ORCA6. Studying the appearance of tau neutrinos gives insight to its cross section on nucleons and into new physics phenomena such as potential non-unitarity of the PMNS matrix.

A central part of the pipeline is replacing the default track/shower classification with a Graph Neural Network, designed to perform tau/nontau PID on ORCA event data and initially developed in [19]. The GNN is retrained as part of this work, now excluding events of the rbr Monte Carlo dataset that were part of the original training dataset. The pipeline includes preprocessing the event data into a graph representation suitable for the GNN, for both the training data and the rbr data used in the analysis. The network is then employed to predict a tau/nontau score for the latter set of events.

Subsequently, these predictions are utilized in a profile likelihood ratio analysis of the tau normalization n_τ, to quantify the sensitivity of KM3NeT/ORCA6 to ν_τ appearance. This analysis was conducted using the *Swim* framework, demanding the definition of tau/nontau classes.

The GNN demonstrated clear capability in distinguishing between tau and nontau events, but the performance of the sensitivity to tau neutrino appearance lags behind that of the track/shower classification. Producing an optimal final analysis was however not possible within the constraints of this thesis. A key issue was the limitation in the availability of rbr events, at the time of analysis, allowing only a subset of the data to be used. To enhance the results, an immediate step would be to repeat the analysis with the full dataset. Furthermore, optimizing the class definitions and configurations used in *Swim* is another avenue of potential performance improvements that should be explored.

Additionally, further work could focus on building a more comprehensive dataset for training, not only improving its statistics, but also utilizing the run-by-run approach to ensure parity to the analysis data.

Concluding this thesis it can be said, that a final evaluation of the sensitivity of the tau/nontau classification, as compared to the track/shower classifier is yet to be achieved, while the necessary groundwork to accomplish this task has been laid out.

Acronyms

ANN Artifical Neural Networks. 9

ARCA Astroparticle Research with Cosmics in the Abyss. 4, 5

BSM beyond Standard Model. 3

CC Charged Current. 2, 6

DOM Digital Optical Modules. 5

DU Detection Units. 5, 17

EdgeConv edge convolution. 11, 12

GNN Graph Neural Network. 8, 11, 12, 17, 20, 21, 22, 24, 26

KM3NeT Cubic Kilometer Neutrino Telescope. 1, 2, 4, 5, 6, 8, 13, 15, 17, 18, 26

MC Monte Carlo. 17, 21, 23

ML Machine Learning. 8, 9

MLP multi layer perceptron. 9, 12

NC Neutral Current. 6

ORCA Oscillation Research with Cosmics in the Abyss. 2, 3, 4, 5, 8, 11, 13, 15, 17, 18, 24, 26

PID Particle Identification. 8, 9, 11, 13, 14, 15, 17, 26

PMNS Pontecorvo–Maki–Nakagawa–Sakata. 2, 3, 13, 26

PMT Photo Multiplier Tubes. 5, 6, 8, 15

RDF random decision forest. 8, 17

ReLU rectified linear unit. 10

SM Standard Model of particle physics. 1

A Definition of the tau and nontau classes

An example of a definition for the tau and nontau classes used for the fit with *Swim*. In this case every event with "pred_tau_nontau_scores" of more than 0.8 is placed in the tau class. Events with no tau/nontau prediction are disregarded. Note that the name, as well as the class norm, are agnostic to the actual definition of the classes and are left as "track" and "shower" for convenience.

```
{
  "classes": [
    { "name": "tracks",
      "general_cut": "pred_tau_nontau_scores != 0 &&
    cos_zenith_recoJGandalf<0 && pred_tau_nontau_scores<=0.8 &&
    rectype_JGandalf==4000 && energy_recoJEnergy>=2 && energy_recoJEnergy
    <=100 && antimu_proba_bkg<=0.0018",
      "muon_loose_cut": "pred_tau_nontau_scores != 0 &&
    cos_zenith_recoJGandalf<0 && pred_tau_nontau_scores<=0.8 &&
    antimu_proba_bkg<0.1  && rectype_JGandalf==4000 && energy_recoJEnergy
    >=2 && energy_recoJEnergy<=100",
      "reconstructions": {
        "E": "JEnergy",
        "cosT": "JGandalf",
        "By": "JGandalf"
      },
      "ClassNorm" : "TrackNorm"
    },
    { "name": "showers",
      "general_cut": "pred_tau_nontau_scores != 0 &&
    cos_zenith_recoJShower<0 && cos_zenith_recoJGandalf<0 &&
    pred_tau_nontau_scores>0.8 && rectype_JShower==4000 &&
    energy_recoJShower>=2 && energy_recoJShower<=1000 && antimu_proba_bkg
    <=0.0018",
      "muon_loose_cut": "pred_tau_nontau_scores != 0 &&
    cos_zenith_recoJShower<0 && cos_zenith_recoJGandalf<0 &&
    pred_tau_nontau_scores>0.8 && antimu_proba_bkg<0.1  &&
    rectype_JShower==4000 && energy_recoJShower>=2 && energy_recoJShower
    <=1000",
      "reconstructions": {
        "E": "JShower",
        "cosT": "JShower",
        "By": "JShower"
      },
      "ClassNorm" : "ShowerNorm"
    }
  ]
}
```

Bibliography

1. Cush. *Standard Model of Elementary Particles* `https://commons.wikimedia.org/wiki/File:Standard_Model_of_Elementary_Particles_Anti.svg`. [Online; last accessed 16.08.2024].

2. Cleveland, B. T., Daily, T., Davis Raymond, J., Distel, J. R., *et al.* Measurement of the Solar Electron Neutrino Flux with the Homestake Chlorine Detector. *Astrophys. J.* **496,** 505–526 (Mar. 1998).

3. Ahmad, Q. R., Allen, R. C., Andersen, T. C., Anglin, J. D., *et al.* Measurement of the Rate of $\nu_e + d \rightarrow p + p + e^-$ Interactions Produced by ^8B Solar Neutrinos at the Sudbury Neutrino Observatory. *Phys. Rev. Lett.* **87,** 071301. `https://link.aps.org/doi/10.1103/PhysRevLett.87.071301` (7 July 2001).

4. Workman, R. L., Burkert, V. D., Crede, V., Klempt, E., *et al.* Review of Particle Physics. *Progress of Theoretical and Experimental Physics* **2022,** 083C01 (Aug. 2022).

5. DONUT Collaboration, Kodama, K., Ushida, N., Andreopoulos, C., *et al.* Observation of tau neutrino interactions. *Physics Letters B* **504,** 218–224. arXiv: `hep-ex/0012035 [hep-ex]` (Apr. 2001).

6. KM3NeT Collaboration, Geißelbrecht, N., Cerisy, L., Lastoria, C., *et al.* Study of tau neutrinos with KM3NeT/ORCA. *to be published in J. High Energ. Phys.* (2024).

7. Abraham, R. M., Alvarez-Muñiz, J., Argüelles, C. A., Ariga, A., *et al.* Tau neutrinos in the next decade: from GeV to EeV. *Journal of Physics G Nuclear Physics* **49,** 110501. arXiv: `2203.05591 [hep-ph]` (Nov. 2022).

8. Fedynitch, A., Engel, R., Gaisser, T. K., Riehn, F., *et al. Calculation of conventional and prompt lepton fluxes at very high energy* in *European Physical Journal Web of Conferences* **99** (Aug. 2015), 08001. arXiv: `1503.00544 [hep-ph]`.

9. KM3NeT Collaboration, Adrián-Martínez, S., Ageron, M., Aharonian, F., *et al.* Letter of intent for KM3NeT 2.0. *Journal of Physics G Nuclear Physics* **43,** 084001. arXiv: `1601.07459 [astro-ph.IM]` (Aug. 2016).

10. KM3NeT. *KM3NeT Member Institues - Status 07/2024* `https://www.km3net.org/wp-content/uploads/2022/05/Cities-and-Sites-of-KM3NeT_220425-scaled.jpg`. [Online; last accessed 19.07.2024].

11. KM3NeT. *Artist's impression of the KM3NeT detector with the multi-PMT optical module* `https://www.km3net.org/wp-content/uploads/2021/12/artists-impression-km3net-detector-edw.png`. [Online; last accessed 12.08.2024].

12. Hallmann, S. *Sensitivity to atmospheric tau-neutrino appearance and all-flavor search for neutrinos from the Fermi Bubbles with the deep-sea telescopes KM3NeT/ORCA and ANTARES* PhD thesis (Friedrich-Alexander-Universität Erlangen-Nürnberg, 2021).

13. KM3NeT. *Artist's impression of a Cherenkov neutrino telescope* `https://www.km3net.org/wp-content/uploads/2021/12/Cherenkov-detector-edw.png`. [Online; last accessed 12.08.2024].

14. Domi, A. *Shower reconstruction and sterile neutrino analysis with KM3NeT/ORCA and ANTARES* PhD thesis (AIX-Marseille Université/Universitá degli studi di Genova, 2019).

15. Bourret, S. *Neutrino oscillations and Earth tomography with KM3NeT-ORCA* PhD thesis (Université Sorbonne Paris Cité, 2018).

16. Maderer, L. *Sensitivity to tau-neutrino apperance with the first seven strings of KM3NeT/ORCA* MA thesis (Friedrich-Alexander-Universität Erlangen-Nürnberg, 2019).

17. Breiman, L. Random Forests. *Machine Learning* **45,** 5–32 (Jan. 2001).

18. Moser, M. *Sensitivity studies on tau neutrino appearance with KM3NeT/ORCA using Deep Learning Techniques* PhD thesis (Friedrich-Alexander-Universität Erlangen-Nürnberg, 2020).

19. Hennig, L. *Tau neutrino identification with Graph Neural Networks in KM3NeT/ORCA* MA thesis (Friedrich-Alexander-Universität Erlangen-Nürnberg, 2023).

20. Neutelings, I. `https://tikz.net/neural_networks/`. [Online; last accessed 01.08.2024].

21. Guderian, D. *Development of detector calibration and graph neural network-based selection and reconstruction algorithms for the measurement of oscillation parameters with KM3NeT/ORCA* PhD thesis (Westfälische Wilhelms-Universität Münster, 2022).

22. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. *arXiv e-prints,* arXiv:1412.6980. arXiv: `1412.6980 [cs.LG]` (Dec. 2014).

23. Reck, S. & Moser, M. *OrcaNet* `https://ml.pages.km3net.de/OrcaNet/`. 2024.

24. Qu, H. & Gouskos, L. Jet tagging via particle clouds. *Phys. Rev. D* **101,** 056019. arXiv: `1902.08570 [hep-ph]` (Mar. 2020).

25. Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., *et al.* Dynamic Graph CNN for Learning on Point Clouds. *arXiv e-prints,* arXiv:1801.07829. arXiv: `1801.07829 [cs.CV]` (Jan. 2018).

26. Cowan, G., Cranmer, K., Gross, E. & Vitells, O. Asymptotic formulae for likelihood-based tests of new physics. *European Physical Journal C* **71,** 1554. arXiv: `1007.1727 [physics.data-an]` (Feb. 2011).

27. Esteban, I., Gonzalez-Garcia, M. C., Maltoni, M., Schwetz, T., *et al.* The fate of hints: updated global analysis of three-flavor neutrino oscillations. *Journal of High Energy Physics* **2020,** 178. arXiv: `2007.14792 [hep-ph]` (Sept. 2020).

28. *Swim* `https://git.km3net.de/oscillation/Swim`. 2024.

29. Rademakers, F., Brun, R., Canal, P., Naumann, A., *et al. ROOT - An Object-Oriented Data Analysis Framework. root-project/root: v6.10/04* version v6-10-04. Aug. 2017.

30. Coelho, J. *OscProb* `https://github.com/joaoabcoelho/OscProb`. 2024.

31. Geisselbrecht, N. *First measurement of tau appearance with KM3NeT/ORCA6* in *Proceedings of 38th International Cosmic Ray Conference — PoS(ICRC2023)* **444** (2023), 1107.

32. Gal, T. *km3pipe* `https://git.km3net.de/km3py/km3pipe`. 2024.

33. *OrcaSong* "`https://git.km3net.de/ml/OrcaSong`". 2024.

34. Mölder, F., Jablonski, K., Letcher, B., Hall, M., *et al.* Sustainable data analysis with Snakemake [version 2; peer review: 2 approved]. *F1000Research* **10** (2021).

35. Weissbrod, S. `https://git.km3net.de/sweissbrod/gnn_pipeline`. 2024.

36. Honda, M., Athar, M. S., Kajita, T., Kasahara, K., *et al.* Atmospheric neutrino flux calculation using the NRLMSISE-00 atmospheric model. *Phys. Rev. D* **92,** 023004. arXiv: `1502.03916 [astro-ph.HE]` (July 2015).

37. Gal, T. *km3services* `https://git.km3net.de/km3py/archive/km3services`. 2024.

38. *uproot* `https://pypi.org/project/uproot/`. 2024.

39. Geißelbrecht, N. personal correspondence. 2024.

Acknowledgments

I want to extend my deepest gratitude to everyone who supported me during the process of writing this thesis. Thank you to everyone at ECAP, and the KM3NeT group in particular, for providing me with such a welcoming work environment. Special thanks go to the following persons:

- PD Dr. Thomas Eberl, for sparking my interest in neutrinos, giving me the opportunity to work on such a fascinating project and guiding me through this thesis.
- Nicole Geißelbrecht, for helping me with Swim and always being available to answer my questions.
- Lukas Hennig, who originally constructed the GNN and helped me with Snakemake and OrcaNet.
- Dr. Rodrigo Gracia Ruiz ,for helping with the file transfers and additional support in the final stages of the thesis.
- My parents, who have been, and continue to be the best support I could hope for.

Finally, thank you very much, Thomas, Nicole, Rodri, Cedric and Kerstin for reviewing the thesis at various stages and providing me with feedback.

Eigenständigkeitserklärung

Hiermit versichere ich, Sebastian Weissbrod (22782636), die vorgelegte Arbeit selbstständig und ohne unzulässige Hilfe Dritter sowie ohne die Hinzuziehung nicht offengelegter und insbesondere nicht zugelassener Hilfsmittel angefertigt zu haben. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen und wurde auch von keiner anderen Prüfungsbehörde bereits als Teil einer Prüfung angenommen.

Die Stellen der Arbeit, die anderen Quellen im Wortlaut oder dem Sinn nach entnommen wurden, sind durch Angaben der Herkunft kenntlich gemacht. Dies gilt auch für Zeichnungen, Skizzen, bildliche Darstellungen, sowie für Quellen aus dem Internet.

Mir ist insbesondere bewusst, dass die Nutzung künstlicher Intelligenz verboten ist, sofern diese nicht ausdrücklich als Hilfsmittel von dem Prüfungsleiter bzw. der Prüfungsleiterin zugelassen wurde. Dies gilt insbesondere für Chatbots (insbesondere ChatGPT) bzw. allgemein solche Programme, die anstelle meiner Person die Aufgabenstellung der Prüfung, bzw. Teile derselben bearbeiten könnten.

Ort, Datum	Unterschrift