ECAP-2010-032

Diplomarbeit

Muon Energy Reconstruction for the Neutrino Telescope ANTARES using Neural Networks

Jutta Schnabel

Oktober 2010

SIGILLVM ACADEMIAE FRIDERICO ALEXANDRINAE

Erlangen Center for Astroparticle Physics
Friedrich-Alexander-Universität Erlangen-Nürnberg
Department für Physik

Abstract

High energy neutrino astrophysics is a is relatively new discipline at the intersection of particle physics and astronomy which promises to have a large potential for new discoveries. Neutrinos inducing charged particles with highly relativistic energies can be detected through the Cherenkov radiation of these charged particles when they traverse a suitable medium. Large volume Cherenkov detectors in water or ice such as the ANTARES detector in the Mediterranean Sea or Icecube at the South Pole use mainly neutrino induced muons to this end. Both detectors are complete or close to their completion and first data has already been analysed. For almost all analyses, an estimation of the muon energy is needed. Due to the statistical nature of muon energy loss mechanisms in a medium, probability density estimation lies at the core of muon energy reconstruction. One method to model probability density is derived from the basic functionality of our own brain cells, the neurons. An Artificial Neural Network (ANN) can derive an efficient probability density function from a suitable set of data through a training algorithm, if the set represents the phenomenology of the problem comprehensively and efficiently. In this work, two estimators are developed for muon energy reconstruction in the ANTARES detector using ANNs. The two estimators follow different objectives. The Simple Estimator aims to derive the muon energy from the most basic data without the need for any other prior analysis or reconstructions. The Cherenkov Estimator makes in addition use of an already existing track reconstruction method and is used to increase the energy resolution. In the progress of this work, possible input parameters for different data categories are derived from Monte Carlo simulation and suitable parameters are selected through application of various selection methods. The adaption of these parameters for ANN training are discussed and a Principle Component Analysis is performed to increase the efficiency of the feature representation in the data set. Finally, the performance of the estimators is evaluated and their applicability to ANTARES data is explored.

Contents

Chapter 1

High Energy Neutrino Physics

High-energy neutrino astrophysics means looking for the unknown with the unseen. The last century witnessed a tremendous development to probe ever deeper into the universe and increase the observable energy range from pure optical detection to radio and gamma astronomy. Optical methods are the broad foundation on which astrophysics is build, but are by far not the only signals we can receive. Other messengers were searched for and gave rise to the field of astroparticle physics. As early as in 1911 Hess showed in measurements with balloons up to an altitude of 5 km that an ionising radiation must exist that comes from above as its strength increases with altitude [12]. At this time, the only known particles were protons, neutrons and electrons an thus they were believed to be constituents of these cosmic rays. As the number of known particles grew, new possibilities opened up also for research with cosmic rays. The discovery of the neutrino by Cowan and Reines in 1956 gave access to the new branch of neutrino physics which proves also beneficiary to astrophysics. As "new" constituents of cosmic rays the small cross section of neutrinos proves to be a blessing as well as a curse. Especially high-energy neutrinos from distant sources might reach Earth where other particles at comparable energies would already have undergone scattering or absorption and traverse the atmosphere easily. Neutrinos are the most abundant cosmic rays at sea level [11], but on the other hand their probability of detection is small and large detector volumes and exposure times are needed to obtain significant statistics.

1.1 Neutrino Sources

The principle reaction which leads to the production of neutrinos is the decay of pions, kaons and muons [11]. For pions (similar to kaons) the reaction

$$\pi^{\pm} \longrightarrow \mu^{\pm} + \bar{\nu}_{\mu}(\nu_{\mu}) \tag{1.1}$$

leads to neutrinos and muons which can further decay to electrons, producing electron neutrinos. This reaction lies at the core of neutrino production in extragalactical processes as well as the formation of neutrinos in the Earth's atmosphere.

1.1.1 Cosmic Neutrinos

All astrophysical sources with non-thermal equilibrium are potential neutrino sources [16]. Charged particles emitted by these objects are accelerated e.g. by Fermi acceleration in shock fronts [12] and can thus gain ultra high energies (UHE). These highly energetic particles can either escape the acceleration regions and are detectable as cosmic rays or react in the acceleration regions to produce electrical neutral particles which are not bound

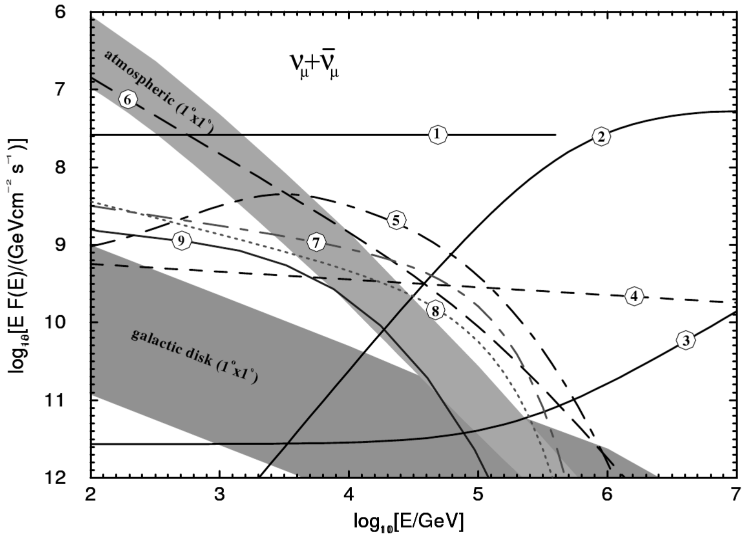


Figure 1.1: Summary of expected fluxes of UHE neutrinos for various accelerators. The lines indicate fluxes expected for single point sources, for details see text below and [16]

by electromagnetic fields of the source and propagate freely.
As an example neutrons might be produced in hadronic interactions such as $p + \gamma \rightarrow \pi^+(\pi^0) + n(p)$ and escape the source region, while the resulting charged or neutral pion decays to emit neutrinos or gamma rays (see equation 1.1). If the maximum efficiency of neutrino production is associated with each escaping nucleon [16], an upper limit for the neutrino flux can be estimated from measured cosmic rays and gamma spectra. Expected fluxes for various neutrino sources and models can be seen in Figure 1.1.

Extragalactic Sources The largest acceleration regions can be found in extragalactic objects which emit matter in large quantities through jets. They would be detectable as neutrino point sources and measuring UHE neutrino emission is the main aim of large volume detectors like ANTARES. Various models for acceleration and particle emission of these objects exists and detection of their neutrino fluxes will shed light on fundamental processes inherent to these objects. As protons have a limited range due to the Greisen-Zatsepin-Kuzmin cut-off [12], UHE neutrinos can sometimes be the only cosmic rays detectable from these sources in this energy regime.

Active Galactic Nuclei Supermassive black holes are believed to exist in the core of roughly 1 % of all bright galaxies [16]. This active nucleus emits the equivalent of the radiation power of more than our galaxy's stellar inventory from an area the size of a planetary system. In the jets protons can be accelerated, which react to cause neutrinos as described above. Neutrino emission might also originate from the host galaxy and surrounding matter. Predictions for the expected flux are highly model dependent. Three models for quasar 3C273 are indicated in Figure 1.1, lines (1)-(3).

Galaxy Clusters Although galaxies themselves do not significantly contribute to UHE cosmic rays, the interaction of cosmic rays from these galaxies with the intergalactic medium can give rise to neutrino emission. Therefore galaxy clusters are neutrino candidates, what can be seen from the prediction for the coma cluster in Figure 1.1, line (4).

Gamma-Ray Bursts Gamma-Ray Bursts (GRBs) are characterized by a flash of gamma rays from extragalactical sources, which have a duration between fractions of a second and hundreds of seconds [12]. Although their exact nature is still poorly understood, they are thought to be violent supernovae or the merger of two neutron stars, where in both cases a black hole is formed. Particles are believed to be accelerated in shock fronts, possibly to UHE. They might therefore emit UHE neutrinos as well.

Galactic Sources Within our galaxy, cosmic ray emitters are believed to contribute on a smaller scale to the expected UHE neutrino flux. They are generally surrounded by matter and photon fields which can serve as interaction regions for neutrino production.

Shell-Type Supernova Remnants Super Novae are violent star explosions which cause high neutrino fluxes, although of lower energy. This could be seen from supernova SN1987A, which is to date the only confirmed observation of cosmic neutrinos. Super Nova Remnants (SNRs) are blast waves driven into the interstellar medium by these explosions. Models predict detectable UHE neutrino fluxes for some SNRs in out Galaxy (see Figure 1.1, lines (7) - (9)).

Microquasars Microquasars are Galactic binary systems consisting of a compact object, e.g. a black hole, and a companion star. The compact object accretes mass from the companion star and produces jets, which serve as acceleration region with neutrino production. Accreting black holes mirror many properties of AGNs on a smaller scale and exhibit therefore UHE neutrinos.

Pulsars and Pulsar-Driven Nebulae Pulsars are rapidly spinning neutron stars which emit cosmic rays in the pulsar wind. Interaction of the wind with a surrounding synchrotron nebula could cause high energy neutrinos. The Crab Nebula is used as an example for this type of objects and its expected flux can be seen in Figure 1.1, line (5).

Neutrino Diffuse Emission Diffuse neutrino emission can originate from almost everywhere where there are cosmic rays and matter to interact with. The Galactic disc and non-resolved sources give also rise to diffuse emission of neutrinos. This diffuse flux from a $1^\circ \times 1^\circ$ regions indicated by the dark grey band in Figure 1.1. Cosmic rays hitting the sun lead to a neutrino flux expected from this direction (Fig. 1.1, line (6)) and extragalactical sources too faint to be identified as point sources also contribute to diffuse background. The flux of cosmic rays is closely connected to the expected neutrino flux through production processes and can be used to give an upper limit for the diffuse cosmic neutrino flux. The shape of the cosmic-ray spectrum for diffuse emission from the Galactic plane is given [22] as

$$\frac{dN}{dE} \propto E^{-2.7}. \tag{1.2}$$

This spectrum is of interest in two ways. Firstly because it indicates the expected neutrino spectrum, secondly because it determines the shape of the atmospheric neutrino spectrum caused by interaction of cosmic rays with the atmosphere.

1.1.2 Atmospheric Neutrinos

The largest part of cosmic rays is formed of protons, electrons and heavy nuclei (9% α particles, 1 % heavier nuclei [16]). When these particles hit Earth's atmosphere they cause air showers which lead to a multitude of different particles, including the reaction presented in 1.1. Neutrinos produced in these air showers can be used as a test beam for neutrino

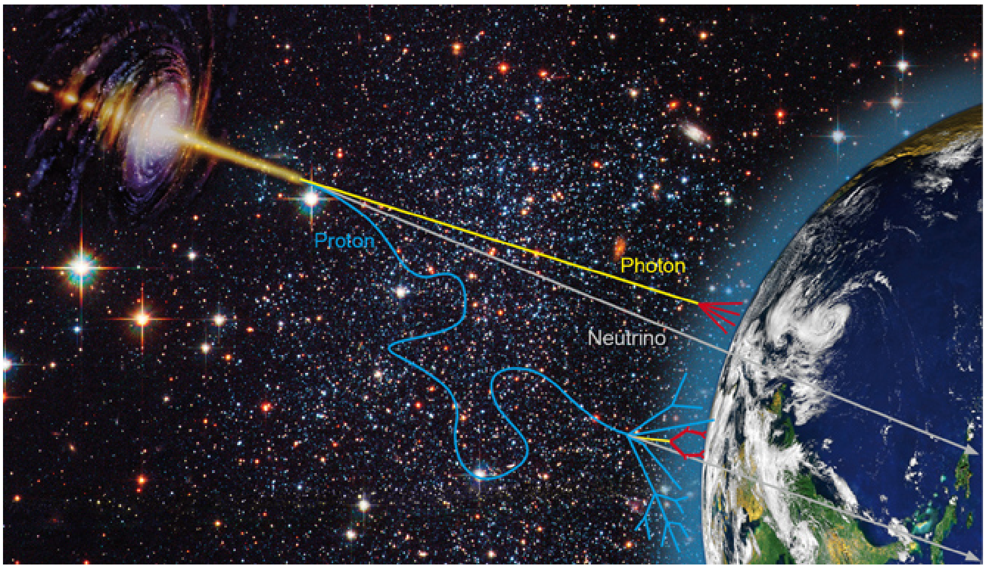


Figure 1.2: Artist's view of messengers from cosmic sources which can interact in the atmosphere and create various signatures used in astropartical physics [10]

telescopes, but together with the Galactic neutrino flux they form a constant irreducible background which complicates the search for point sources. Their expected spectrum is indicated by the light grey shading in Figure 1.1. An artist's view of the main sources for neutrinos at the Earth's surface, galactic emitters and atmospheric neutrinos, can be seen in Figure 1.1.

1.1.3 Other Associated Fields of Physics

Astrophysics is not the only relevant field that high -energy neutrino detectors can contribute to. As the neutrino mass is larger than zero, oscillation of neutrinos between their three flavours ν_e, ν_μ and ν_τ can be measured for neutrinos emitted at the source with a known flavour rate [16]. Furthermore, a number of dark matter theories predict a neutrino flux for annihilation of the dark matter particles, e.g. WIMPS (Weakly Interacting Massive Particles), in gravitational centres like the sun. The possibilities for applications of neutrino detectors is large and their implementation can therefore contribute widely to current research in various fields.

1.2 Detection Principle

One possible way to detect a neutrino is through a charged current interaction with neighbouring nucleons which leads to the production of a charged particle (see Figure 1.3). This particle, an electron, muon or tau, can then be detected. For detection of high energy neutrinos the Cherenkov radiation of the particles when traversing a medium is used. This is the basic principle with which current neutrino telescopes work and the contributing mechanisms will be discussed here briefly.

1.2.1 Neutrino Interactions

For the expected neutrino flux around 1 GeV, where the product of flux and cross section has a maximum, the interaction rate of atmospheric neutrinos per kiloton and year is of

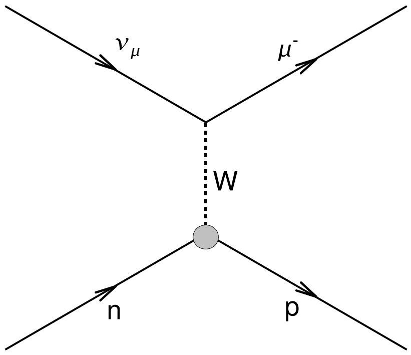


Figure 1.3: Muon production through weak interaction of a neutrino with a proton

the order of $100(\nu - \textbf{interactions})$ [11]. For energies above that, the detector size has to increase. Therefore detectors with volumes of the order of a megaton target material are required for effective studies. For this kind of detectors muons suggest themselves as most effectively detectable particle, as the typical track length of these ultra-relativistic muons is of the order of some meters to several kilometres, depending on their energy. For a nucleon N, the charged current process (Fig. 1.3) lying at the core of the neutrino detection is

$$\nu_\mu + N \longrightarrow \mu + \mathrm{X}. \tag{1.3}$$

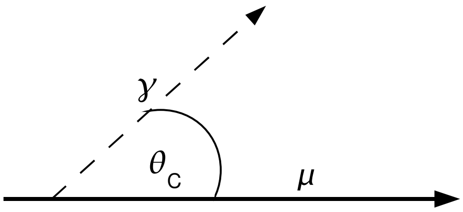
The larger the neutrino's energy, the smaller the deviation in direction between the secondary muon and the primary neutrino. The direction of the muon can therefore be used to calculate the origin of the neutrino. As target material water or ice is used, as it exists in abundance in nature and is transparent for the Cherenkov light.

1.2.2 Cherenkov Radiation

Charged particles which traverse a bipolar medium at a velocity exceeding the local phase velocity of light in this medium cause Cherenkov radiation. The passing charge induces local electromagnetic fields which relax by emitting photons. As the particle travels faster than the photons, the photons can interfere constructively and propagate. Cherenkov photons are radiated on a cone which is centred around the particle track with an opening angle θ_C, characteristic for the traversed medium and the particle velocity $\beta = v_\mu/c$.

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

with n being the refraction index of the medium and β the relativistic velocity of the charged particle. Charged particles will emit photons as long as $\beta > c/n$. Neutrino induced muons traversing water or ice as well as secondary charged particles produced in energy loss mechanisms of the muon (see Section 3.1) emit Cherenkov radiation. In water the Cherenkov angle is $\theta_C = 42^\circ$ for photons with $\lambda = 450\,\mathrm{nm}$ [6].



1.2.3 Photon Scattering and Absorption

Photons propagating through matter have a limited range through scattering and absorption. While absorption limits the effective range of photons, scattering alters their direction and makes it therefore more difficult to associate them with their origin. The absorption length can be measured by comparing the drop in intensity for a given source at distance d by

$$I(d) = I_0 e^{-d/(\lambda_{\text{abs}})} \tag{1.4}$$

While the absorption length λ_{abs} of Cherenkov photons at the ANTARES site lies between 50 - 70 m, the scattering length λ_{scat} is about 230 - 300 m [8]. Photon attenuation is therefore dominated by absorption. The effective attenuation length is given by

$$\frac{1}{\lambda_{\text{att}}} = \frac{1}{\lambda_{\text{abs}}} + \frac{1}{\lambda_{\text{scat}}}. \tag{1.5}$$

The effective attenuation length for the ANTARES site was measured to vary, depending on conditions, between 46 m and 60 m [8].

Chapter 2

The ANTARES Detector

2.1 Implementation of ANTARES in the Deep Sea

Neutrino cross section and detection principle presented in the last chapter call for an optical detector in a large volume of naturally abundant material like water or ice. The ANTARES neutrino telescope is the largest project to this end in the Northern hemisphere, situated in the Mediterranean Sea about 20 km off the French coast (Figure 2.1) at a depth of about 2400 m. Since May 2008, the ANTARES detector is completed and taking data, forming the largest undersea neutrino telescope deployed to date.

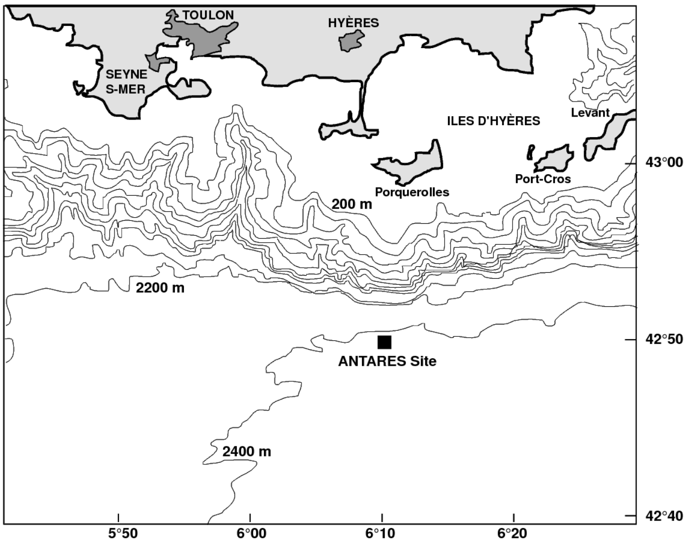


Figure 2.1: Location of the ANTARES detector off the French shore. Picture taken from [13]

2.1.1 Detector Layout

The ANTARES neutrino telescope consists of twelve strings, also called lines, which are attached to the sea bottom by a Bottom String Socket (BSS) and kept upright by a buoy at the top end. These strings have a length of about 460 m and each is equipped with 75 Optical Modules (OMs), see Figure 2.2, giving a total of 885 OMs. Three OMs are always grouped together around a Local Control Module (LCM) to form a storey. Therefore 25 storeys are located on each line, five of them being used for acoustical detection. All in all the ANTARES detector covers an area of 0.1 km^2 and is thus large enough for UHE neutrino detection.

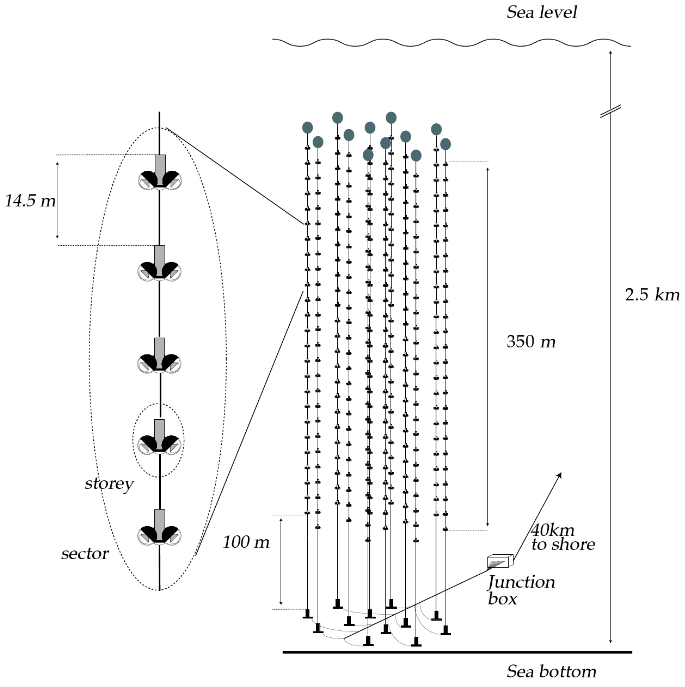


Figure 2.2: The ANTARES detector with storeys of three Optical Modules on each line. The Junction Box can be seen in the background. [5]

Within each OM the actual detecting units are situated, the photo-multiplier tubes (PMTs) [7]. An OM is a glass sphere with one PMT attached to the glass through an optical gel in one half of the sphere. They are sensitive to single photons with a varying efficiency depending on the incident angle of the photons. The main task of ANTARES is detection of neutrinos coming from the other side of the earth, as only thus their secondary muons are distinguishable against the background of atmospheric muons. Therefore the OMs are looking downwards at an angle of 45° to be especially sensitive for the detection of those “upgoing” muons.

2.1.2 Data Readout and Trigger

An incident photon in one of the PMTs forms the signal with a certain charge. If this charge exceeds a threshold of 0.3 p.e. it is passed on to shore as so-called “L0 hit”. The data is read out according to the “all data to shore”-concept, meaning that no filtering of the data is performed at the detector site. The readout system in the LCM digitizes the signal from the PMT and assigns it an arrival time from the clock system in the LCM. This information is then sent on along the lines and to the junction box bundling the readout of all lines. A deep sea cable connects the junction box to the shore station, where the filtering of the data takes place.

In order to decide about the passage of a muon, photon hits on the various OMs are searched

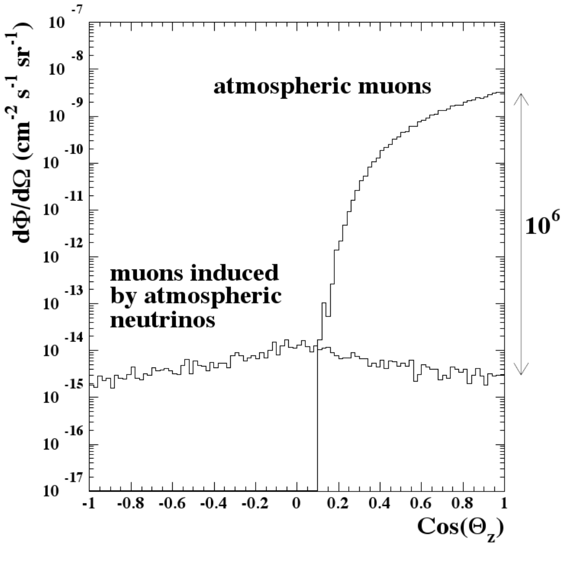


Figure 2.3: Expected flux of muons depending on the zenith angle of their track for the ANTARES site. [6]

for which could be causally connected. Various trigger algorithms exist to decide whether or not a muon has passed the detector, and only if a trigger criterion is met, the data for a time window around these triggered hits is written away as an event. In the course of this work the so called 3N-trigger is used as selection criterion for hits and therefore it will be briefly discussed here.

The 3N Trigger The trigger criterion is always applied on a subset of the L0 hit, the so called L1 hits. All hits with amplitudes larger than 2.5 p.e. are L1-hits. Of the remaining hits those are counted among the L1-hits that have at least one other hit within a coincidence window of 20 ns on the same LCM. The 3N trigger now looks for L1 hits which could be causally connected. If at least 5 causally connected hits are found on different storeys, the trigger identifies these as an event and all hits in a time window of $\pm 2.2\mu$s around the triggered hits are written away as triggered hits [9].

2.2 Event Classes and Background in the ANTARES Detector

In the deep sea not only photons from neutrino induced muons can cause hits. Apart from other relativistic particles emitting Cherenkov radiation, natural radioactivity in the sea water causes photons and the various fish and bacteria can contribute to a rather large background from which the signal has to be filtered.

Atmospheric and Neutrino Induced Muons The ANTARES detector was built deep under water in order to avoid a large background of atmospheric particles. In order to distinguish neutrino induced muons against the background of atmospheric particles, the Earth is used as a shield against atmospheric muons and only muons with tracks originating from below (“upgoing” muons) can truly be attributed to neutrino interactions. For muons coming from above (“downgoing”), it is not possible to distinguish between neutrino induced muons and atmospheric particles. At the ANTARES site, the number of downgoing atmospheric muons still exceeds that of upgoing muons by far (see Figure 2.3). For some physics analysis it might be of interest to analyse this downgoing muon flux. An energy estimation will therefore be developed for muons from all incident angles.

Other Neutrino Signatures Apart from muons, neutrinos of different flavour can induce electrons or tau leptons. Electrons generally do not propagate far in the detector but produce an electromagnetic shower, causing Cherenkov radiation through their secondary products. Tau neutrinos on the other hand induce a hadronic shower at their vertex, but charged tau leptons emit not as much photons along their path as muons due to their larger mass.

Background Photons Salt water contains radioactive K40 which emits photons through β-decay. This causes a constant background of about 40 kHz per OM. Apart from the fact that bacteria cause light emission on a small scale as well as localised “bursts” which illuminate a few OMs. While the bursts can be filtered out from the data, the random photon emission from bacteria adds to the noise in the detector. If conditions are good, the total background lies between $60 - 100$ kHz per OM and can rise much higher with a growing impact on data taking efficiency.

2.3 Monte Carlo

In order to determine the response of the ANTARES detector to passing particles, Monte Carlo simulations are performed. At the top of the atmosphere cosmic rays are simulated to produce showers and propagated to sea level using the CORSIKA program [17]. The successive propagation of these sea level particles through water was done by the MUSIC code. Of these particles only those matter which hit the sensitive volume around the detector, which is defined by the attenuation length of water at the ANTARES site. This volume is called the “can” and is a cylindrical volume around the detector’s center with a radius of $R = 238\,\mathrm{m}$ and a total height of 592 m. Particles at the can level are propagated further using the KM3 package [2], which also simulates the response of the ANTARES detector.
For upgoing neutrinos, the ANTARES Monte Carlo simulation is done with GENHEN [1], again using KM3 once the particles reach the can’s surface. More details about the ANTARES simulation chain can be found in [17].
If muons are produced by neutrinos already outside the simulation can, the Monte Carlo files list the muon’s energy at the can’s surface, in case of interaction within the can, at the muon vertex. This is also the simulated energy which will be reconstructed by the estimators presented in this work.

Chapter 3

The Principles of Muon Energy Reconstruction

Muons passing through matter lose energy and cause the emission of photons through various processes. While the Cherenkov radiation of these ultrarelativistic muons can be used for track reconstruction, a far greater amount of energy is lost through ionisation, bremsstrahlung and pair production. Due to the stochastical nature of these processes, the standard approach to reconstruction is modelling of a probability density function (PDF).

3.1 Muon Energy Loss Mechanisms

Relativistic muons passing through matter lose their energy either by single collisions which lead to ionization, atomic or collective excitation, or by radiative processes. As these effects are randomly distributed along the muon track, it is only possible to describe them statistically for a large number of muons over a given unit of length. For this purpose the mean rate of energy loss $-\left\langle \frac{dE}{dx} \right\rangle$ is introduced. As neutrino telescopes use primarily the Cherenkov emission of muons for detection, only muons with energies at least the order of the muon mass are of interest and will be considered in the following section.

3.1.1 Ionization

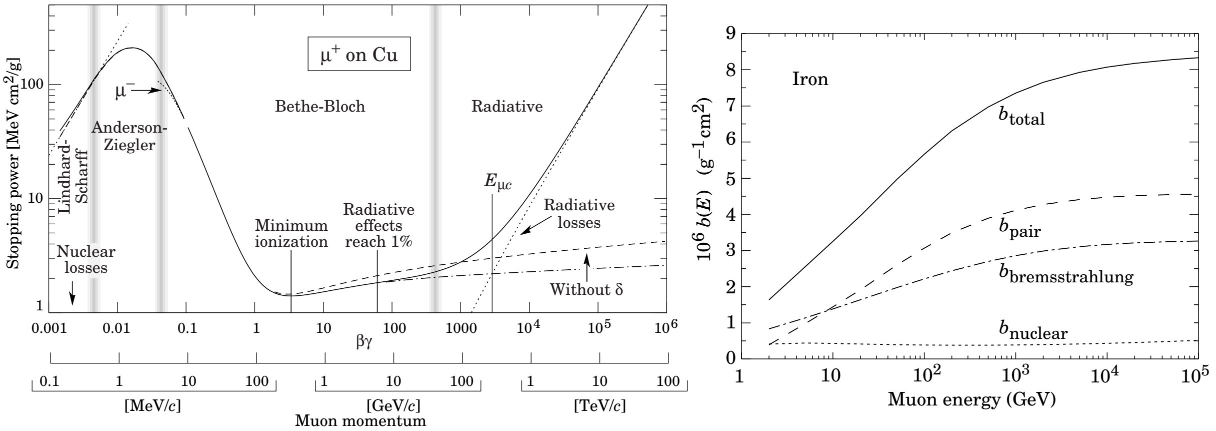
In the range of $0.1 \leq \beta\gamma \leq 1000$ ionization losses in a medium are given by the Bethe-Bloch equation

$$-\left\langle \frac{dE}{dx} \right\rangle = K \frac{Z}{A\beta^2} \left[\frac{1}{2} \ln \frac{2m_e c^2 \beta^2 \gamma^2 T_{max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right]. \tag{3.1}$$

such that dx is given in mass per unit area, e.g. g/cm^2. The absorber is described by atomic number Z and mass A, with $K = 4\pi N_A r_e^2 m_e c^2$, while $\delta(\beta\gamma)$ is the density effect correction, I the mean excitation energy and T_{max} the maximum kinetic energy which can be imparted to a free electron in a single collision. This formula exhibits a minimum at a few GeV. Muons at this energy are called "minimally ionizing particles". As can be seen from Figure 3.1a, ionization is dominant and applicable below the critical energy $E_{\mu c}$.

3.1.2 Radiative Effects

At muon energies above 1 TeV, radiative effects gradually gain importance. Those include (in oder of significance at high energies, see Figure 3.1b),



(a) mean energy loss

(b) radiative processes

Figure 3.1: Muon energy loss as function of the muon's energy: (a) The stopping power or mean rate of energy loss for muons versus their energy and muomentum (lower axis) and (b) fractional energy loss by muons in iron due to radiative processes in total and pair production, bremsstrahlung and photonuclear interaction seperately. From [21]

Pair Production In the vicinity of a nucleus a virtual photon is emitted by the muon which leads to the production of an electron-positron pair.

Bremsstrahlung Interaction of the muon with the electromagnetic field of an electron or nucleus leads to the emission of a photon.

Photonuclear Interaction Deep inelastic scattering between muons and nuclei causes a hadronic shower.

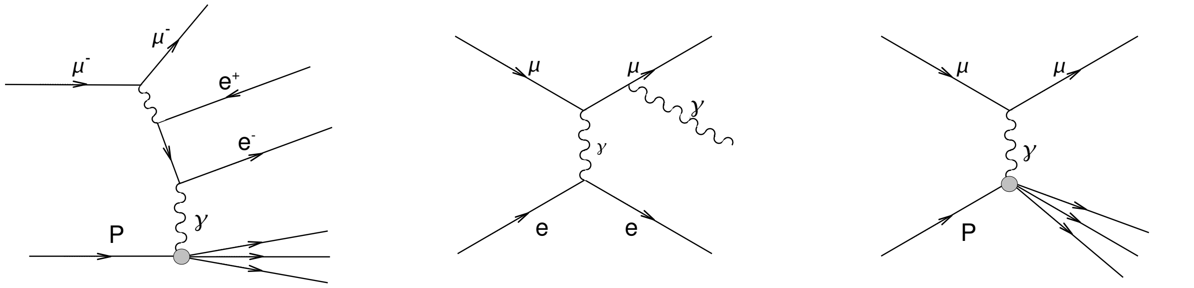
These processes have generally small cross sections and give rise to hard spectra and large energy fluctuations. As energy loss cannot be described as a continuous process, the mean energy loss is generally approximated by

$$-\frac{dE}{dx} = a(E) + b(E)E \tag{3.2}$$

with the ionization loss $a(E)$ as in (3.1) and the sum of the listed radiative processes parametrized by $b(E)$. The "muon critical energy" $E_{\mu c}$ is defined as the energy at which ionization losses equal radiation losses. In water $E_{\mu c}$ lies roughly at 780 GeV [19]. The incontinuous nature of the energy loss processes and the transition between ionization and radiative processes make the task of energy reconstruction difficult, as will be seen in the progress of this work.

3.1.3 Photon Emission through Showers

Pair production and Bremsstrahlung lead to the emission of very energetic photons which cause electromagnetic showers along the muon track by decaying into pairs of oppositely charged particles which themselves undergo radiative processes. As long as these secondary particles are faster than the speed of light in water, they also emit Cherenkov photons. The typical radiation length of electromagnetic showers is a few meters at the relevant energies and photons are generally emitted on a cone defined by the Cherenkov emission around the shower axis [13]. In comparison to electromagnetic showers, hadronic showers cause photon emission on almost the same length scale and comparable angular distribution, although induced by interaction between the muon and nucleons which lead to secondary hadrons



(a) Pair Production (b) Bremsstrahlung (c) Photonuclear Interaction

Figure 3.2: Exemplary Feynman diagrams of the radiative processes

and electromagnetic subshowers. They are relatively rare compared to electromagnetic showers. Both shower types typically carry a large part of the energy loss. Finally photons generally undergo scattering and absorption before reaching the OMs, which makes precise reconstructions of the photon's emission point relatively difficult. As will be seen in 6.1, it is however possible to distinguish roughly between direct Cherenkov photons from the muon and shower photons through their arrival time.

3.2 Probability Density Estimation

The stochastic nature of the energy loss mechanisms, the finite geometrical acceptrance of the detector and the presence of background photons makes a purely analytical approach to energy reconstruction impossible. A certain set of initial parameters x_i such as the measured amplitude per photomultiplier, arrival time of the photons and number of triggered optical modules which are accessible to the experiment can only with a certain probability be connected to a fixed muon energy E_μ. Therefore it is more convenient to use the probability density $p(\mathbf{x}, E_\mu)$ of a vector of the input parameters $\mathbf{x}$ to calculate an output energy E_μ. To estimate $p(\mathbf{x}, E_\mu)$ it is necessary to obtain a set of data input vectors $\mathbf{x}^n$ with $n \in [1, ..., N]$ which are derived from the phenomenology described by the PDF. There are three ways to approximate the probability density, referred to by [3] as parametric, semi-parametric and non-parametric method. While the non-parametric method models $p(\mathbf{x}, E_\mu)$ purely from a set of data $(\mathbf{x}^n, E_\mu)$ without assumption of any functional dependence between input and output, parametric methods are the complete opposite by first assuming a fixed functional dependence. They most widely used and will be introduced through the example of Maximum Likelihood Estimation. Neural Networks, which are used for this work, take the middle way and are therefore semi-parametric methods.

3.2.1 Maximum Likelihood

The parametric modelling in maximum likelihood procedures starts out with a functional form of $p(\mathbf{x}, E_\mu)$ which leads for a set of data $(\mathbf{x}^n, E_\mu)$ to the calculation of the likelihood $\mathcal{L}$ by using Bayes formula

$$\mathcal{L} = \prod_n p(\mathbf{x}^n, E_\mu) = \prod_n p(E_\mu|\mathbf{x}^n)p(\mathbf{x}^n) \tag{3.3}$$

where it is assumed that all data points are derived from the data independently. $p(E_\mu|\mathbf{x}^n)$ is the modelled probability of E_μ given a certain $\mathbf{x}$ and $p(\mathbf{x}^n)$ the initial probability of $\mathbf{x}$. To find the most probable solution for a given $\mathbf{x}$, it is necessary to maximize $\mathcal{L}$ to arrive at a concrete value of E_μ. Instead of maximizing $\mathcal{L}$ it is more convenient to minimize the error function $\mathcal{E}$ for the distribution, which is the negative likelihood scaled by the logarithm.

$$\mathcal{E} = -\ln \mathcal{L} = -\sum_n \ln p(E_\mu|\mathbf{x}^n) - \sum_n \ln p(\mathbf{x}^n) \tag{3.4}$$

As the last term only adds a constant to $\mathcal{E}$, the error function becomes

$$\mathcal{E} = -\sum_n \ln p(E_\mu|\mathbf{x}^n). \tag{3.5}$$

The search for the most probable output value E_μ for a given data point $\mathbf{x}$ has thus been reduced to minimizing the error for each data point presented to the probability density function $p(\mathbf{x}, E_\mu)$.

3.2.2 Semi-parametric Probability Density Estimation

As it is necessary to have a detailed and accurate theoretical model in order to obtain the PDF for maximum likelihood reconstruction, this prametric method can be difficult to apply to problems which are described by a very complex phenomenology. The non-parametric method, which would on the other hand assume no functional correlation between input und output data at all, completely neglects underlying functional dependence in the probability density estimation. The next chapter will introduce the third alternative, semi-parametric modelling through neural networks, which draws $p(\mathbf{x}^n, E_\mu)$ from data samples while allowing for a complex functional approximation between input and output data.

Chapter 4

Artificial Neural Networks

4.1 From Biology to Technology

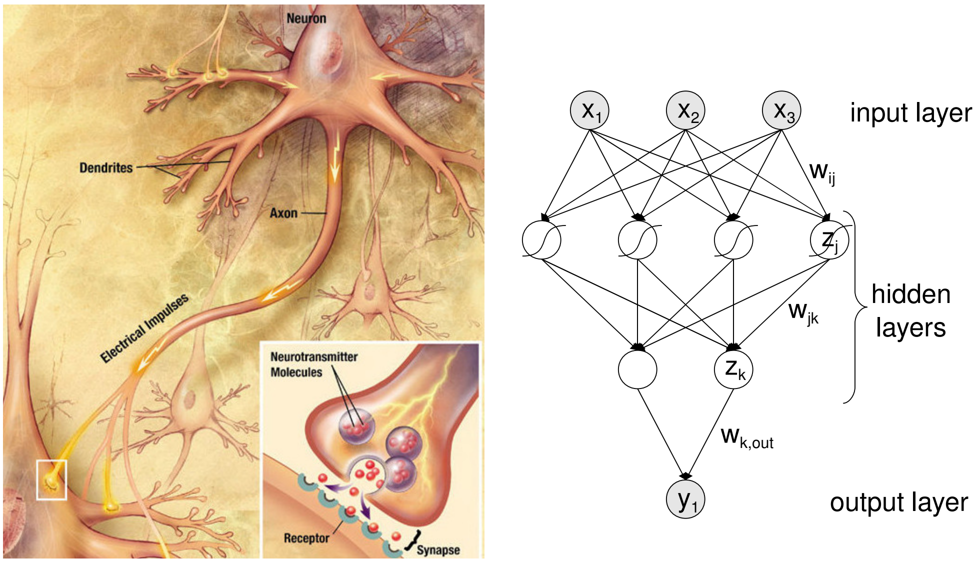
The idea of artificial neural networks (ANN) was derived from the basic function of biological neurons. Although the brain as a whole still seems to be only poorly understood, the functionality of simple networks of neurons has been explored to a large extend. Understanding neural networks has lead to the development of artificial perceptrons and ANNs.

4.1.1 The Neuron and Biological Neural Networks

A biological neuron represents a decision unit which processes and forwards a signal depending on the electrical signals received from other connected units. This signal is then reprocessed by other neurons or leads to a physical reaction of the organism. As can be seen in Figure 4.1a, the neuron receives electrical charge coming from the dendrites of other neurons and transmitted to its cell body at the synapses, whose efficacy can vary. If the summed up charge from all synapses exceeds a certain threshold, an action potential is triggered at the axon hillock and is transmitted through a single axon and dendrites towards connected neurons, which perform the same task. This system of simple no/yes decision units proves to be efficient enough to perform complex tasks which allow animals to respond to the multitude of stimuli presented by nature.

4.1.2 Artificial Neural Networks

The computational representation of such networks derive their functionality from the same principle as their biological counterparts. A number of input parameters x_i is presented to a network in the input layer (see Figure 4.1b). The value of these input parameters is transmitted to the nodes in the next layer, where an input x_i to a node j is assigned a *connection weight* w_{ij} which represents the synapses' varying efficacy of charge transmission. The summed up input to each node $\sum_i w_{ij} x_i$ leads to an output z_j which in the simplest case is given by a step function. The output of all these nodes are presented to the nodes in the next hidden layer or lead to final output values $\mathbf{y}_{\text{out}}$. The decision making is done in the hidden layers of the network, whose functionality is generally not accessible to the user, and in the output neuron. Therefore, the adaptivity to certain tasks derives from choosing the right connection weights w_{ij} which is achieved by a learning algorithm and will be discussed in Section 4.2.3. This type of network is called feed-forward neural network as neuron output is only presented to neurons closer towards the network output and no reconnections to nodes closer to the input are used. Only feed-forward networks will be considered in this work.



(a) biological neurons

(b) artificial neural net

Figure 4.1: Comparison of biological and artifical neural networks: (a) Artist's version of a neuron and charge transmission at the synapses; (b) ANN with input neurons x_i, hidden neurons z_j and output neurons y_{out}

4.1.3 The Perceptron

The perceptron represents a more advanced version of the simple neuron described above. The input of a perceptron can be extended by a bias parameter w_0 which is added to the input $\sum_{i=1}^{I} w_{ij}x_i + w_0$. To generate output, a generalized *activation function* $g(a)$ is introduced. The neuron output y_j is then given by

$$y_j = g\left(\sum_{i=1}^{I} w_{ij}x_i + w_0\right) \tag{4.1}$$

In the case of the "classical" ANN, this activation function was a step function with $g(a) = -1$ for $a < 0$ and else $g(a) = +1$. However, for many problems which deal with continuous variables as output rather than yes/no decisions it proved more useful to introduce continuous functions. A comparison of most commonly used activation functions will be done in Section 7.1.2. Generally, activation functions have a codomain between -1 an 1, which is also the range artificial neural nets use as input and output range. These functions usually contain a gradient which is called the activation steepness s. In the case of the widely used sigmoid symmetric activation function $g_{\text{sigsym}}(a)$ the *activation steepness* gives the gradient of the tanhfunction at $a = 0$

$$g_{\text{sigsym}}(a) = \tanh(s \cdot a) = \frac{2}{1 + \exp(-2s \cdot a)} - 1 \tag{4.2}$$

Large values for s approximate a step function and are therefore used for decision problems while smaller s might be more useful for problems with continuous output y. The right value for s can be adapted to suit the problem and will be explored for the purpose of energy reconstruction in Section 7.1.2.

4.2 Modelling Probability Density

The problem of energy reconstruction was introduced in the last chapter as the task to find the probability density $p(\mathbf{x}, E_\mu)$ connecting an input vector $\mathbf{x}$ and a resulting muon energy E_μ. Neural networks represent what is defined by [3] as a semi-parametric method for modelling $p(\mathbf{x}, E_\mu)$. Although no analytic function is used to describe the mapping between $\mathbf{x}$ and E_μ, approximation of the hidden functional dependence between input and output is possible through adaption of the connection weights w_{ij} and variation of the number of nodes in the neural network's hidden layers. These weights have to be derived from suitable data which is presented to the neural network during the training process and adjusted using the error backpropagation algorithm.

4.2.1 Multidimensional Functions

A feed-forward network which contains no loops can, according to [3], represent any continuous functional mapping $\mathbb{R}^n \rightarrow \mathbb{R}$ if it contains a minimum of two hidden layers and sigmoidal activation functions. The approximation of a function is generally improved as the number of hidden units is increased. Three hidden layers can be used to approximate any smooth mapping. Within the hidden layers, the network is capable to approximate so called hidden functions which only correlate a subset of the input variables. This way input parameters which are not directly correlated with the output can still have an influence on the output value by contributing to a hidden function. In the case of the energy estimator, one of these parameters is, for example, the zenith angle. The energy is not dependent on the zenith angle, but the detector acceptance which contributes to the PDF is. If the input variables contain all the information necessary to arrive at the output value, two layers should be sufficient to model the probability density $p(\mathbf{x}, E_\mu)$. In this work, networks with two and three hidden layers will be tested.

4.2.2 The Error Function

The likelihood $\mathcal{L}$ of a mapping was defined in Section 3.2.1 as $\prod_n p(E_\mu|\mathbf{x}^n)p(\mathbf{x}^n)$ for a set of n data samples $(\mathbf{x}^n, E_\mu)$. To point out the similarities between maximum likelihood and neural networks, let the muon energy, the target value of the mapping of an input value vector $\mathbf{x}$ be noted by E_μ and assume that the calculated output of the neural net $y(\mathbf{x}, \mathbf{w})$ is accurate apart from a Gaussian noise ϵ, such that

$$E_\mu = y(\mathbf{x}, \mathbf{w}) + \epsilon. \tag{4.3}$$

ϵ is Gaussian distributed, $p(\epsilon) = \alpha \exp(-\epsilon^2/2\sigma^2)$, and by inserting (4.3) into this distribution the probability density can be written as

$$p(E_\mu|\mathbf{x}) = \alpha \exp\left(-\frac{(E_\mu - y(\mathbf{x}, \mathbf{w}))^2}{2\sigma^2}\right). \tag{4.4}$$

If the negative logarithm is applied to the probability density $p(E_\mu|\mathbf{x})$ in the same way as to the likelihood in Section 3.2.1, a formula can be derived for the error of a neural network $\mathcal{E}$ as

$$\mathcal{E} = \frac{(y(\mathbf{x}, \mathbf{w}) - E_\mu)^2}{2\sigma^2} - \ln \alpha \tag{4.5}$$

where $\ln \alpha$ only contributes a constant value and can be neglected as well as the scalar $2\sigma^2$. If a data set with N data points is presented to the neural net, the error of this whole set is called *Mean Square Error* $\mathcal{E}^N_{MSE}$ of the data set, later reffered to as MSE, and is calculated as

$$\mathcal{E}^N_{MSE} = \sum_{n=1}^{N} (y(\mathbf{x}^{\mathrm{n}}, \mathbf{w}) - E^n_\mu)^2. \tag{4.6}$$

Minimizing $\mathcal{E}_{MSE}$ for a neural net can therefore be seen as equivalent to minimizing the error derived from a likelihood function without need of the parametric modelling of the likelihood method.

4.2.3 Backpropagation

The standard algorithm to minimize the error of a neural network is called error backpropagation and will be introduced here briefly, adapted from [3], as a successfull training process lies at the core of all neural network modelling.
Let the input values z_i of a node j with weights w_{ij} be summed up to $a_j = \sum_i w_{ij} z_i$. After applying the activation function $g(a_j)$, the output is given as $z_j = g(a_j)$. If any z_i is a parameter in the input layer of the ANN, it is be denoted by x_i. As ANN output value z_j is denoted by y_k (see Figure 4.1b). Assuming an error function $\mathcal{E}^N = \mathcal{E}^N(\mathbf{y})$ for the output vector $\mathbf{y}$ of N data points in a data set, the error depending on a certain weight w_{ij} is given by

$$\frac{\partial \mathcal{E}^N}{\partial w_{ij}} = \frac{\partial \mathcal{E}^N}{\partial a_j} \frac{\partial a_j}{\partial w_{ij}}. \tag{4.7}$$

Defining the "error" $\delta_j = \frac{\partial \mathcal{E}^N}{\partial a_j}$ and using the definition of a_j to replace $\frac{\partial a_j}{\partial w_{ij}} = z_i$ leads to

$$\frac{\partial \mathcal{E}^N}{\partial w_{ij}} = \delta_j z_i \tag{4.8}$$

This means that the derivative can be calculated by evaluating δ at the output end of the connection and simply multiplying it with the input z_i on the input end. In the output layers δ_k can be derived from its definition

$$\delta_k = \frac{\partial \mathcal{E}^N}{\partial a_k} = \sum_k g'(a_k) \frac{\partial \mathcal{E}^N}{\partial y_k} \tag{4.9}$$

For the hidden layers, applying the chain rule, using $a_k = \sum_j w_{kj} g(a_j)$ and inserting above formula, δ_j is found to be

$$\delta_j = \frac{\partial \mathcal{E}^N}{\partial a_j} = \sum_k \frac{\partial \mathcal{E}^N}{\partial a_k} \frac{\partial a_k}{\partial a_j} = g'(a_j) \sum_k w_{kj} \delta_k \tag{4.10}$$

This equation is called the back-propagation formula, as it allows for the calculation of all δ throughout the network starting from the output layer. After calculating the a_j for the network with input variabel vector $\mathbf{x}$, the δ are derived and the derivatives $\frac{\partial \mathcal{E}^N}{\partial w_{ij}}$ can be obtained. Summing up this error for the whole data set gives a deriative of the error with respect to a particular connection weight w_{ij}, which can now be adjusted to minimize the error function. This is done through a standard minimizer. An improved and faster version of the backpropagation algorithm used in the FANN library which will be used in this work. Details about the iRProp algorithm can be found in [15].

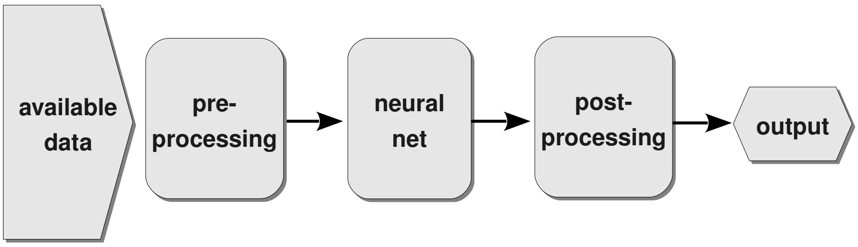


Figure 4.2: The main processing steps for neural nets contains preprocessing of input variables, processing through the neural net and postprocessing.

4.3 Data Processing Steps

Setting up an ANN model requires the selection of basic data and an ANN implementation for the neural network. The subsequent working steps follow closely the processing steps of the ANN (Fig. 4.2). The first and most time consuming task is formed by setting up the simulation samples for training, selecting suitable input parameters and preprocessing them to fit the ANN's requirements. The second step is the actual training of the ANN and the comparison of different network layouts, which then generates an output. For energy reconstruction, postprocessing simply consist of rescaling the energy back to physical units. A crosscheck of the actual performance of the network through various validation methods is the last step in the development process and discussed in chapter 8.

4.3.1 Selecting Monte Carlo Sets

For an effective training process, two data sets are necessary: the actual training set A with which the weights are adjusted and an independent test set B to monitor the performance and avoid specialization on the training set. The training set A should contain all relevant data categories to a certain extent, for the purpose of energy estimation this includes events over the available energy range from 10 GeV up to 10^8 GeV and all zenith angles of the muon. Therefore the different data sets were cut to an distribution over the logarithmic energy range which contains all energies to sufficient extend and evenly in the zenith angle (Fig. A.1, test sets B can be found in Appendix A). A subset of the test sets, which are generally larger than the training sets, is used during training as validation set V.

When developing an estimator for the ANTARES detector it is obviously recommendable to use Monte Carlo events which are optimized for a fully functional detector with moderate background. These conditions, which include a detector with 12 lines without broken OMs and a constant background rate of 60 kHz per OM, are in the context of this work called "ideal" conditions and denoted with number 1. Generally, the background is not as homogeneous as in this assumption and a certain number of not functioning OMs and a varying amount of lines has to be taken into account. To test the method against these "real" conditions, Monte Carlo samples were merged with the background of minimum bias events, which should contain only background, and the real OM conditions were taken from a data run. These samples are denoted by 2. In addition to that the method was tested for upgoing muons only (denoted by up) and muons from all zenith angles (denoted by all). Thus the set A_1^{up} is a training sample of upgoing muons with "ideal" detector conditions.

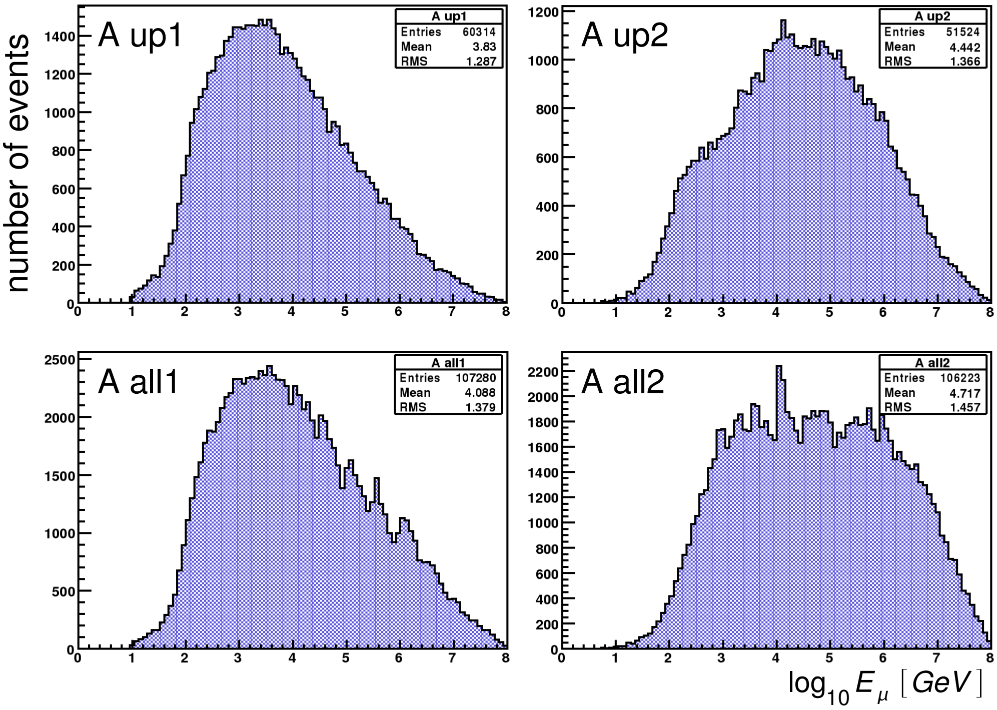


Figure 4.3: Energy distribution of the training samples

4.3.2 Network Training

For network implementation and training the FANN library [18] was used. It is a multilayer ANN library in C which includes various backpropagation training algorithms and allows control of all basic ANN parameters presented above. Preprocessing and postprocessing are completely left to the user and therefore an easy overview of all modelling steps can be obtained. Trained ANNs and data sets are stored as simple text files and allow direct access to all information, which will be of use for ANN analysis presented in the next Chapter 5. The topology of the neural network, which includes numbers of hidden layers, nodes in each hidden layer and, if we limit the discussion to feed-forward networks, the activation function in those nodes, can be chosen in various ways. The selection which was made for this work is presented in chapter 7.

For the follwing discussion of ANNs, a few conventions have to be introduced. The topology of a neural net will be given by the number of nodes in each layer as “input layer/first hidden layer/ second hidden layer/output layer”, i.e. the net 18/90/12/1 has 18 input parameters, 90 and 12 nodes in the two hidden layers, and 1 output.

The training process is monitored by evaluating the MSE of the training and validation set after adjusting the weights in each training step, which is called an *epoch*. If the MSE of the validation sample is constant or rises in five subsequent epochs, the neural net starts to adapt to peculiarities of the training set. Therefore the training is stopped when this *stop criterion* is met to guarantee generality of the neural net.

Chapter 5

Methods for Parameter Selection

Network training can only be as successful as the information of the input parameters is representative of the given problem. Although one could consider just expanding the number of parameters until all available information is included, a large number of parameters generally reduces the network's capabilities to extract the main features. Therefore methods are needed to distinguish parameters which add to the representation from those which do not contribute or just repeat features already contained in the input sample. Firstly, the covariance between parameters is used to display the degree of dependence between pairs of parameters. Secondly, once an ANN is trained, its connection weights can indicate the importance a certain parameter has for the calculation of the output value. Last but not least the response of the ANN output to variation of one input parameter displays the stability of the ANN model regarding this parameter.

5.1 Covariance

A standard stochastic mean to describe the distribution of a given set of values x_n with $i \in [1, \ldots, N]$ and mean value $\langle x \rangle$ is the *variance* which is given by

$$s^2(x) = \frac{1}{N}\sum_{n=1}^{N}(x_n - \langle x \rangle)^2.$$

If variance is viewed as a measure for the dependence of the values x_i on themselves, this concept can be expanded to quantify the dependence of two different parameters x, y for a set of data with respect to each other. Thus the *covariance* between two parameters is defined as

$$cov(x, y) = \frac{1}{N}\sum_{n=1}^{N}(x_n - \langle x \rangle)(y_n - \langle y \rangle). \tag{5.1}$$

Covariance can only be measured between two parameters. A compact way to display the mutual dependences between various parameters in a set of data is the *covariance matrix* **C**. To calculate **C** a sample of p parameters is viewed as a vector $\mathbf{x} = (x_1, \ldots, x_p)$ in its pattern space with Eigenvectors $\mathbf{e}_i$ such that $\mathbf{x} = \sum_{i=1}^{p} x_i \mathbf{e}_i$. The mean values of the input parameters are given by the mean vector $\langle \mathbf{x} \rangle = (\langle x_1 \rangle, \ldots, \langle x_p \rangle)$. With $\tilde{\mathbf{x}} = \mathbf{x} - \langle \mathbf{x} \rangle$ the covariance matrix can now be defined as $\mathbf{C} = \tilde{\mathbf{x}}\tilde{\mathbf{x}}^{\mathbf{T}}$. After normalization the individual entries of **C** equal the covariance between the respective parameters $\mathbf{C}_{ij} = cov(x_i, x_j)$. The covariance matrix of the input parameter set can now be used to display the extent to which parameters are dependent on each other and help avoiding redundancies. A large covariance between energy and input parameter on the other hand can be used as indicator to a high

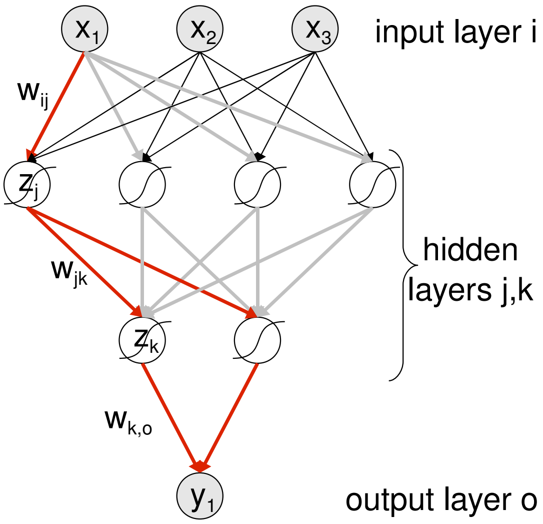


Figure 5.1: In the connection weight method, the connection weights w along a path between input x and output y are multiplied and summed up for all possible paths.

relevance of the parameter for energy reconstruction. Little covariance between energy and input parameter on the other hand can not serve as exclution criterion alone, as discussed in Section 4.2.1.

5.2 The Connection Weight Method

After training a neural net is stored by the FANN library as a simple collection of parameters, like number of nodes, their activation function, and the weights connecting the different nodes (see Section 4.3.2). The connection weights, which represent the weight each input contributes to the output of a node, can therefore be analysed easily.
Let the input layer of a net be denoted by i, the hidden layers by j and k and the output layer by o. The connection weight between two nodes is then given by w_{ij} between the input and first hidden layer. The *importance* of a parameters $\mathbf{Imp}(x_i)$ can then be defined as the connection weights between an input x_i and the output y_{out} along all possible paths connecting these two parameters [20]. Along a given path the connection weights are simply multiplied, neglecting the role of the activation function and the contribution of the other inputs to the node. In nets with two hidden layers, the importance is calculated as

$$\mathbf{Imp}_{CW}(x_i) = \sum_j \sum_k w_{ij} w_{jk} w_{ko}, \tag{5.2}$$

as can be seen in Figure 5.1. This method, far from being accurate, is useful to display relative importance between the various input parameters. For parameters which display no explicit energy dependence a small importance led to their discarding in the selection process resulting in the final set of input parameters presented in Chapter 6. Subsequent comparison of ANN training results with and without those discared parameters confirmed their negligibility. The connection weight method is therefore a useful tool to choose relevant parameters from a large number of candidates.

5.3 Stability of Network Performance: HIPR

Together with the connection weight method, the Holdback Input Randomization Method (HIPR) [20] can serve to determine the stability of the ANNS performance with regard to a certain parameter. The HIPR methods basically consists of altering one parameter while keeping all others at the same value and monitoring the deviation of the output parameter during this process.
Let the input parameter of an ANN be denoted by x_i and the minimum and maximum value of x_i by x_i^{min} and x_i^{max} respectively. The *response* of the network to the variation of x_i can be measured by altering the value of x_i in k steps between x_i^{min} and x_i^{max} while keeping all other input parameters at their original value and monitoring the change of the ANN output. Here $k = 10$ steps were used and a data sample with an original $\mathrm{MSE}(x_i^{\mathrm{true}}, y^{\mathrm{reco}})$ was compared to the altered sample with $\mathrm{MSE}(x_i^{\mathrm{min}} + k(x_i^{\mathrm{max}} - x_i^{\mathrm{min}})/10, y^{\mathrm{var}})$. For a set of N data samples $(\mathbf{x}, E_{\mu c})$ the response $\mathrm{Resp}(x_i)$ of the network for a single input parameter is then given by

$$\mathrm{Resp}(x_i) = \sum_{n=1}^{N} \sum_{k=1}^{10} |\mathrm{MSE}_n(x_i^{\mathrm{min}} + \frac{k}{10}(x_i^{\mathrm{max}} - x_i^{\mathrm{min}}), y^{\mathrm{var}}) - \mathrm{MSE}_n(x_i^{\mathrm{true}}, y^{\mathrm{reco}})| \qquad (5.3)$$

The HIPR method as well as the connection weight method has the drawback that it gives no definite information on the complete influence of an input parameter. The fluctuation of one parameter might within a certain sample not actually be possible the ways it is assumed within HIPR, e.g. assuming a number of hits within an event which is smaller than the number of OMs with hits is not possible but also added to the response in (5.3) when applied to the input selection in Chapter 6. On the other hand HIPR is useful to judge the stability of the ANN's energy estimation regarding parameters which have a varying representation in data compared to simulation, e.g. the number of hits with varying background condition.

5.4 Discarding Parameters

For better comparison, the covariance, importance and response for a set of input parameters are normalized to 100%. If a parameter showed comparatively low values for all three methods, or at least covariance and importance, it was discarded. An example for such a comparison can be found in Section 8.3. The same method was used for a large number of different input parameters, starting from those presented in [4]. They were finally discarded if their taking out showed no significance for the networks performance. In [4] azimuth of the track and x, y and z coordinates of the point of closest approach of the track were used as input parameters and discarded in this work for the above reason. The process of trying out new parameters and, if insignificant, discarding them, lead to the final list of parameters presented in chapter 6.

Chapter 6

Input Parameter Space and Preprocessing

A good selection of parameters derived from the data contains as much information about the event as possible and is on the other hand compact enough not to burden the neural network with irrelevant features which could impair its performance. Finding this selection is a long procedure which generally involves a lot trial and error. The parameters presented below are the result of this process in which the selection techniques presented in the last Chapter 5 were used.

The subsequent transformation of the parameters necessary to fit the requirements of optimal neural network training followed mainly two guidelines: the input parameter range for the neural network should be used to the full extent and the data should represent the features of the event in an effective way to make adaption for ANNs easier.

Therefore the preprocessing of the Monte Carlo events involved several steps which are depicted in Figure 6.1. In Section 6.1 the selection of input parameters for the neural net is discussed, which then have to be transformed to the input range $[-1, 1]$, suitable for neural nets. This transformation is shortly outlined in Section 6.2. Dependencies between the various input parameters can then be extracted using Principle Component Analysis (PCA) which is described in Section 6.3. The PCA output is used to transform the input parameters into feature space which is equivalent to finding a new linear independent basis. The least important parameters can then be tested and dropped if they have little relevance for the network performance. A final transformation of the input parameters in feature space to the interval $[-1, 1]$ forms the end of the preprocessing chain. A listing of all relevant parameters for the different data samples can be found in Appendix A.

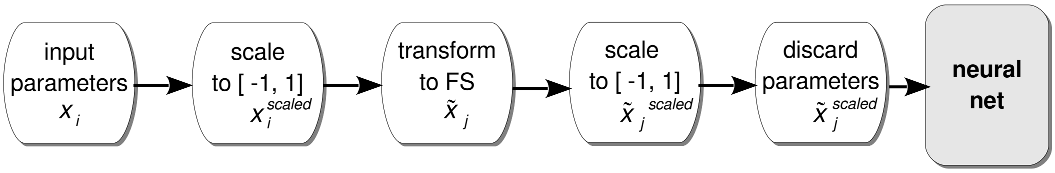

input parameters x_i
scale to [-1, 1] x_i^{scaled}
transform to FS $\tilde{x}_j$
scale to [-1, 1] $\tilde{x}_j^{scaled}$
discard parameters $\tilde{x}_j^{scaled}$
neural net

Figure 6.1: steps of the preprocessing chain

6.1 Input Parameter Selection

First it is investigated which data are actually available and relevant for the reconstruction. The basic data which is recorded by the data acquisition system of the detector is a simple series of photon hit times, positions and amplitude which was identified as an event by a trigger algorithm. For the purpose of some physics analysis such as dark matter search, it might be of use to get an energy estimation before any other processing of the data. After selecting input parameters which are generally useful for energy reconstruction in 6.1.1, the second subsection therefore deals with input parameters derived from this basic data which prove useful for a simple energy estimator (estimator S) without the need for prior track reconstruction.
The standard track reconstruction method for ANTARES is the so called Aart Strategy [14]. It contains several hit selections and a quality parameter. This strategy and its output form a prerequisite for the advanced energy estimation based on the expected Cherenkov radiation from the track (Cherenkov estimator C) and will therefore be discussed briefly in Section 6.1.3. The track reconstruction makes it possible to view the event from the muon's frame of reference and yields input parameters which represent the phenomenology of the energy loss more accurately. The selection of these input parameters is presented in 6.1.4.

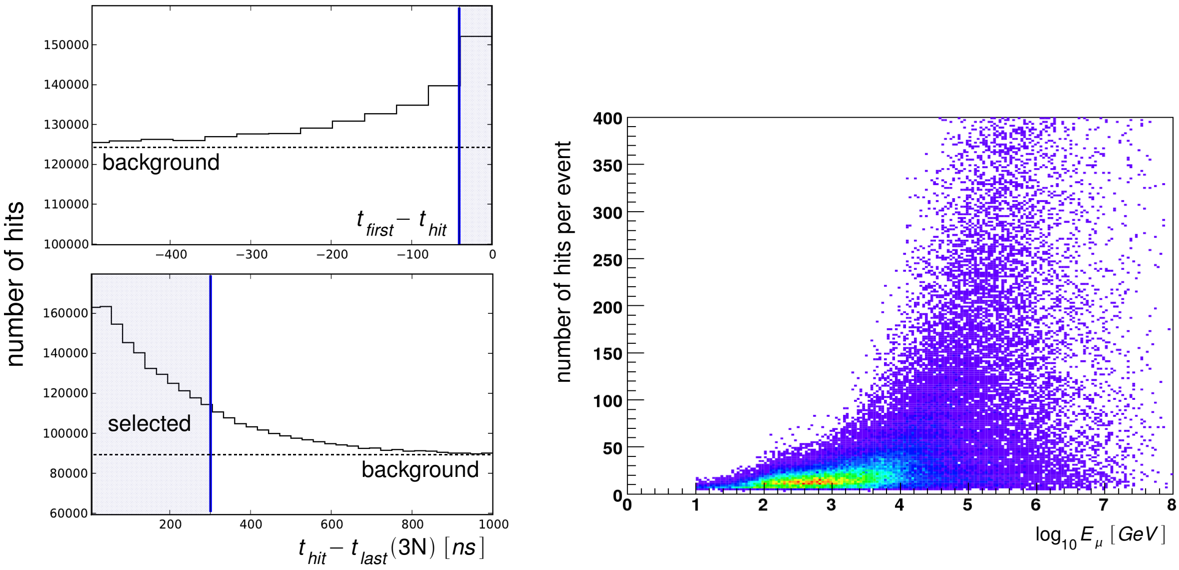
6.1.1 General Input Parameters

For each triggered event the available data contains a series of hits with a certain time, position and amplitude. These hits are either causally connected to the incident muon or background. A hit selection, which in the most basic case consists of the hits selected by a trigger algorithm, selects the most probably muon related hits from this event (see Chapter 2). These two hit series provide the only information initially available for energy estimation. Parameters derived here are of such fundamental importance that they are of use for simple estimator S and Cherenkov estimator C and will therefore be denoted by P^{SC}.

Charge Information As the expected amount of photons from the muon depends on its energy (see Sec. 3.1), obvious choices for basic input parameters P^{SC} are hits from the hit selection. If no data processing has been applied, the triggered hits are used. Basic charge information and reference for further input parameters are therefore:

- $\mathrm{P}_1^{\mathrm{SC}}$: number of selected OMs
- $\mathrm{P}_2^{\mathrm{SC}}$: number of selected hits.

The trigger might not select all hits caused by the muon, thus one can also use all hits within a certain time window around the triggered event as input in order to increase the number of signal hits passed to the net. As sometimes a noise hit is triggered as first hit or the first signal hits are left out of the trigger, it is possible that parts of the signal hits lie before or after the triggered event. Therefore the time window around the trigger is chosen to contain on average for a set of data at least 20% muon related hits in comparison to the constant number of background hits in the same time interval. Using training sample A_1^{up} as a reference the window is set 20 ns before the first and 300 ns after the last triggered event (see Figure 6.2a). This selection contains information about the background as well as additional signal hits and charge.
In order to use information about the total charge deposited in the detector, the average charge per hit can be used. While at low energies mostly direct Cherenkov photons from the muon are detected, high energies lead to more Cherenkov photons which can produce hits



(a)

(b)

Figure 6.2: Selection of a time window with significant hit contribution from the muon: (a) All hit times of set A_1^{up} before the first triggered hit (above) and after the last (below); Selected are all hits within the time interval where the overall hit count exceeds the background by at least 20 % (b) number of thus selected hits per event in A_1^{up} versus the muon energy

with higher charge and are produced in radiative processes (see Figure 6.2b). This change in the charge distribution is taken into account by adding the mean of the distribution and its RMS, which is the squareroot s of the variance presented in Section 5.1, to the input parameters. The higher probability of background photons of mostly low charge is included by considering the mean of all hits within the time window.

- P_3^{SC}: number of all hits in a time window from 20 ns before the first triggered hit until 300 ns after the last triggered hit
- P_4^{SC}: average charge per selected hit
- P_5^{SC}: average charge per hit for all hits within the time window
- P_6^{SC}: RMS of the charge distribution of selected hits

Symmetry of Detector Layout The amount of detected photons is obviously highly dependent on the detector geometry and the muon track's relative position to it. Due to the symmetrical position of the OMs in the detector, the number of storeys can be used to describe the vertical extension of the track within the detector. Accordingly the number of triggered strings mirrors the horizontal extension of the track in the detector. Both value can therefore be a useful handle to OM positions and the number of storeys and strings with triggered hits are therefore added to the input parameter set.

- P_7^{SC}: number of storeys with triggered hits
- P_8^{SC}: number of strings with triggered hits

6.1.2 The Simple Estimator

Track Estimation Information about the track geometry can only crudely be parametrised without track reconstruction and therefore estimates of important track parameters have

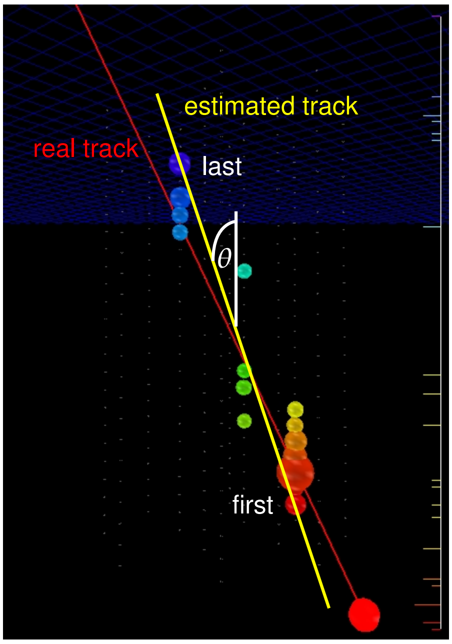


Figure 6.3: The first and last triggered hit are used to estimate the track's orientation (hits are colour-coded in time from red to blue and bubble size indicates the amplitude of the hit; event with $E_\mu = 1460\,\text{GeV}$).

to replace the exact reconstruction of the track. Two estimates were used here, namely a track which is defined by the first and last triggered hit and secondly the Centre of Gravity (CoG) of the event.

Energy loss takes place along the whole muon track. The amount of measured photons is therefore dependent on the length of the muon track within the detector which varies as the muon might originate or decay within the detector volume. The distance between the first and last triggered hit serves as an estimate of the path length of the muon within the detector.
Secondly a muon track which only passes the detector generally causes less hits than a a track which traverses the detector near its centre. The distance between the track and the detector centre is used to parametrise this fact. Instead of the point of closest approach of the reconstructed track the CoG of all hits of the event is used.
Finally, the zenith angle θ of the track is of great importance for the performance of the detector. On the one hand the detector is almost twice as high as wide and a vertical track generally traverses a larger detector volume. On the other hand the OMs are especially sensitive for photons coming from below (see Chapter 2) which increases the photon detection probability for these muons. It is difficult to find a simple and good estimate of the zenith angle without proper track reconstruction. However, the zenith angle between the first and last triggered hit are added to the parameter set for this purpose.

- P^S_9: distance between the OMs of the first and last triggered hit
- P^S_{10}: zenith angle between the OMs of the first and last triggered hit
- P^S_{11}: distance between CoG of triggered hits and CoG of detector

Timing and Local Distribution In energy reconstruction it is generally difficult to include all available parameters which are energy dependent in the reconstruction, e.g. the conventional energy estimator used in ANTARES includes only charge and hit count but leaves

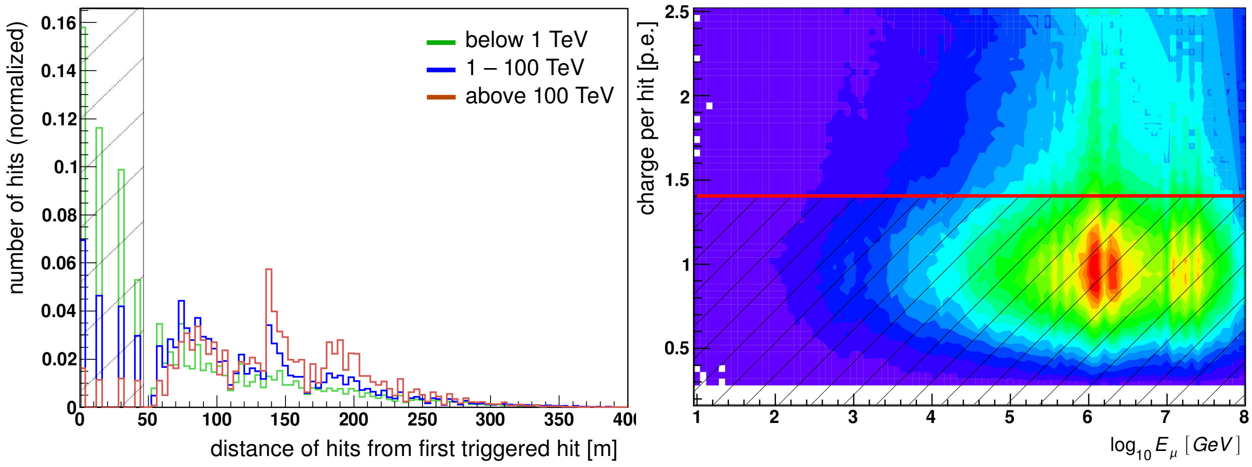


Figure 6.4: Input parameters for the simple estimator: (a) Distance of all triggered hits of A up1 from the first triggered hit (P^S_{17}); (b) amplitude of triggered hits of set A_1^{up} versus the muon energy (P^S_{18})

out geometry and time information. To increase the accuracy of the energy estimation in comparison to a reconstruction solely based on the charge information, the arrival times of photons at the OMs and the positions of the hits must be considered. As the number of input parameters for the network is constant for all data samples, global parameters have to be used to describe the geometrical hit distribution and the hit timing. The statistical choice for the parametrisation of these distributions is the mean of the distribution and its RMS. Therefore the mean value and RMS of arrival times from the first hit, of hit distances from the event CoG and the RMS of the hit charges proved useful as input parameters. As additional information in the time regime the event duration is considered for both strategies.

- P^S_{12}: average hit distance from event CoG
- P^S_{13}: RMS of the distribution of hit distances from event CoG
- P^S_{14}: average hit time after first triggered hit
- P^S_{15}: RMS of the hit time distribution of triggered hits
- P^{SC}_{16}: duration of triggered event

Hit Fractions Two further parameters have proven to be of use. On the one hand some more geometrical information can be contained in the fraction of triggered hits which lie on the first triggered string. On the other hand the fraction of hits with charge lower than 1.4 p.e. is used to account for the larger possibility at high energy for multiple photons hitting the PMT at the same time.

- P^S_{17}: fraction of hits on first string
- P^S_{18}: fraction of hits with charge below 1.4 p.e.

6.1.3 Track Reconstruction

The most commonly used strategy for track reconstruction, called the Aart Strategy consists of a series of hit selections and fitting algorithms. The strategy is described in detail in [14].

First Hit Selection It is assumed, that the largest hit in a sample is always part of the signal hits. All hits which are within a time window of $|\Delta t| = d/v_g + 100\,\mathrm{ns}$, where d is the distance of the OM from the largest hit and v_g the group velocity of light.

Linear Prefit A sub-sample of the hits with local coincidences and those above 3.0 p.e. are fed into a so called linear prefit which tries to fit a line between these hit positions.

M-Estimator Fit Hits within 100 m around the prefit track and within a window of ±150 ns of the calculated arrival times are handed to the M-Estimator.

Maximum Likelihood Fit The output of the M-Estimator and a renewed hit selection based on the estimator is the starting point for a maximum likelihood fit.

Repetition of M-Estimator and Maximum Likelihood The M-Estimator is rotated by 25^o and translated by 50 m and the new hit selection handed to the Maximum Likelihood Fit. The number of tracks which end up within 1^o from each other, are counted as N_{comp}. The track with the best likelihood result is handed to a

Maximum Likelihood Fit with improved PDF with hits with ±250 ns from the expected photon arrival times and those with charge above 2.5 p.e.. This is the final track reconstruction.

In order to characterize the fit quality of a reconstructed track, a parameter Λ is defined:

$$\Lambda = \frac{\log(\mathcal{L})}{N_{\mathrm{DoF}}} + 0.1(N_{\mathrm{comp}} - 1) \tag{6.1}$$

Here $\mathcal{L}$ is the likelihood and N_{DoF} the number of degrees of freedom of the final fit. The larger Λ, the better the track reconstruction is expected to be.

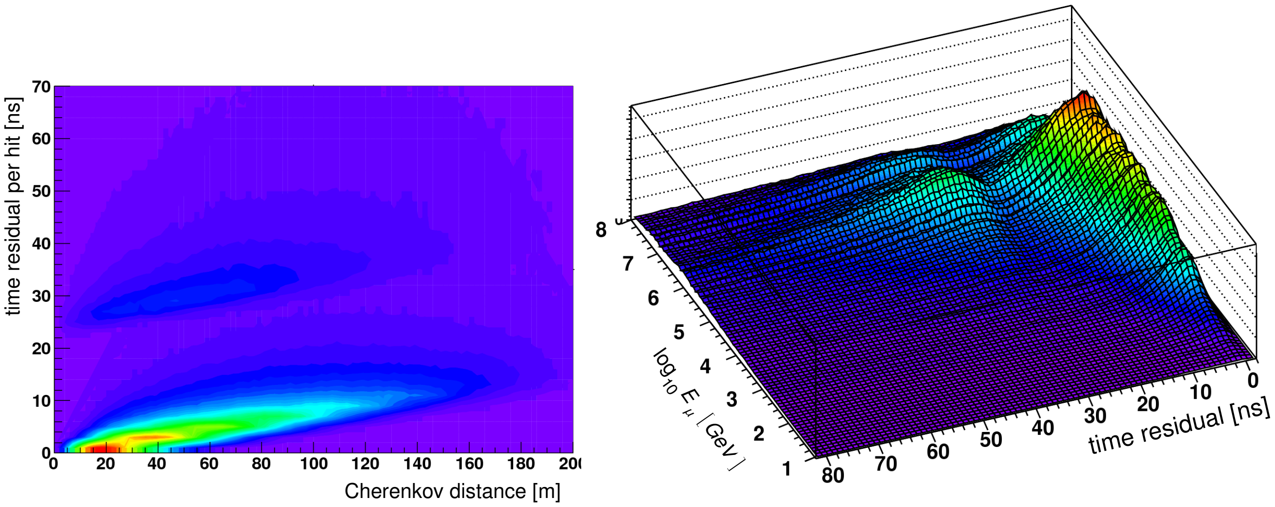
6.1.4 The Cherenkov Estimator

After application of the Aart Strategy to the data there are basically three pieces of information available for the energy reconstruction: the muon's reconstructed track, the selection of triggered hits and all hits in the above given time window. Nine basic parameters $\mathrm{P}_1^{\mathrm{SC}} - \mathrm{P}_8^{\mathrm{SC}}$ and $\mathrm{P}_{16}^{\mathrm{SC}}$ useful for both methods have already been mentioned above and will not be listed here again.

Muon Track Parameters The track estimates of Simple Estimator can now be replaced by the reconstructed parameters from the Aart Strategy. The length of the track in the detector can be approximated by taking the distance between the perpendicular projection onto the track to the first and last triggered hits. The zenith angle and closest approach distance of the track to the detector CoG are now taken directly from the track reconstruction.

- $\mathrm{P}_9^{\mathrm{C}}$: track length between first and last triggered hit
- $\mathrm{P}_{10}^{\mathrm{C}}$: track zenith θ
- $\mathrm{P}_{11}^{\mathrm{C}}$: closest approach distance between detector CoG and track

Hit Distribution and Hit Times With the help of the muon track it is possible now to view the whole event from the tracks frame of reference and describe hit times and distances in relation to the muon. The Cherenkov distance d_C of a hit is defined as the distance between the track and the hit OM under the assumption of photon emission under the Cherenkov angle of 42°. If a Cherenkov photon emitted by the muon would travel from the muon track



(a) cherenkov distance vs time residual

(b) time residuals vs energy

Figure 6.5: (a) cherenkov distance vs time residual of hits in A_1^{up} (b) time residuals of hits versus energy of emitting muon

along this path, a theoretical arrival time of the photon t_0 can be calculated. Arrival times can now be expressed relative to this theoretical arrival time, $t - t_0$. This is called the time residual of a hit and will be used instead of arrival times relative to the first hit. As photons with a large distance to the track might undergo multiple scattering and therefore show a systematically larger time residual, for each hit the time residual was divided by its Cherenkov distance $(t - t_0)/d_C$ and this combined parameter was passed to the net to account for the dependence between the two parameters.

- P_{12}^{C}: Average Cherenkov distance of all hits in the time window
- P_{13}^{C}: RMS of Cherenkov distances of all hits
- P_{14}^{C}: Average time residuals of all hits in the time window
- P_{15}^{C}: RMS of time residuals of all hits
- P_{17}^{C}: Average time residuals per Cherenkov distance of all hits in the time window
- P_{18}^{C}: RMS of time residuals per Cherenkov distance of all hits

Shower Parametrization Most Cherenkov photons are estimated to arrive at approximately their time residual t_0 at the OM. Photons which arrive later but still seem to be hit related are generally caused by energy loss mechanisms along the track. Finally, once the time residual is sufficiently large, one would only expect to measure background events which have no relation to the muon. Therefore three cuts in the time residuals of the hits with residuals $t - t_0 > -10\,\text{ns}$ were chosen at 25 ns to give the fraction of direct Cherenkov photons, 250 ns to account for the photons from energy loss mechanisms and 600 ns to account for background. The fraction of hits within these bins in time residuals was passed to the net as well as the charge within the first two time residual windows.

- P_{19}^{C}: fraction of hits with time residuals $t - t_0 \in [-10\,\text{ns}, 25\,\text{ns}]$
- P_{20}^{C}: fraction of hits with time residuals $t - t_0 \in [-10\,\text{ns}, 250\,\text{ns}]$

- $\mathrm{P}^{\mathrm{C}}_{21}$: fraction of hits with time residuals $t - t_0 > 600\,\mathrm{ns}$
- $\mathrm{P}^{\mathrm{C}}_{22}$: average charge of hits with time residuals $t - t_0 \in [-10\,\mathrm{ns}, 25\,\mathrm{ns}]$
- $\mathrm{P}^{\mathrm{C}}_{23}$: average charge of hits with time residuals $t - t_0 \in [-10\,\mathrm{ns}, 250\,\mathrm{ns}]$

Input Parameters	**Simple Estimator**	**Cherenkov Estimator**
3N triggered hits		
number of OMs	√	√
number of hits	√	√
$\langle A \rangle$ of hits	√	√
RMS(A) of hits	√	√
number of storeys	√	√
number of strings	√	√
event duration	√	√
all hits in time window		
number of hits	√	√
$\langle A \rangle$ of hits	√	√
track parameters	track estimate	reconstructed track
track length in detector	√	√
zenith angle	√	√
distance from detector CoG	√	√
statistic parameters	3N hits	hits relative to direct Cherenkov photons & track
$\langle d \rangle$	√	√
RMS(d)	√	√
$\langle t \rangle$	√	√
RMS(t)	√	√
$\langle \mathrm{t/d} \rangle$		√
RMS(t/d)		√
fraction $t < 25\,\mathrm{ns}$		√
fraction $t < 250\,\mathrm{ns}$		√
fraction $t > 600\,\mathrm{ns}$		√
fraction $d < 50\,\mathrm{m}$	√	
fraction $A < 1.4\,\mathrm{p.e.}$	√	
$\langle A \rangle$ if $t < 25\,\mathrm{ns}$		√
$\langle A \rangle$ if $t < 250\,\mathrm{ns}$		√

Table 6.1: Summary of input parameters to both estimators; for each hit, charge A, distance d and hit time t are used.

6.2 Transformation of Input Parameters

ANNs only allow input values x_i within a certain interval, normally $x_i \in [-1, 1]$. The chosen input parameters $\mathrm{P}^{\mathrm{SC}}_{x_i}$ must therefore be rescaled to this interval. As finding the largest possible value is difficult for some input parameters, this is done by using the same function which also serves as activation function.

$$x_i^{scaled} = \frac{2}{\pi} \arctan\left(\frac{\mathrm{P}^{\mathrm{SC}}_{x_i}}{\alpha} - \delta\right) \qquad (6.2)$$

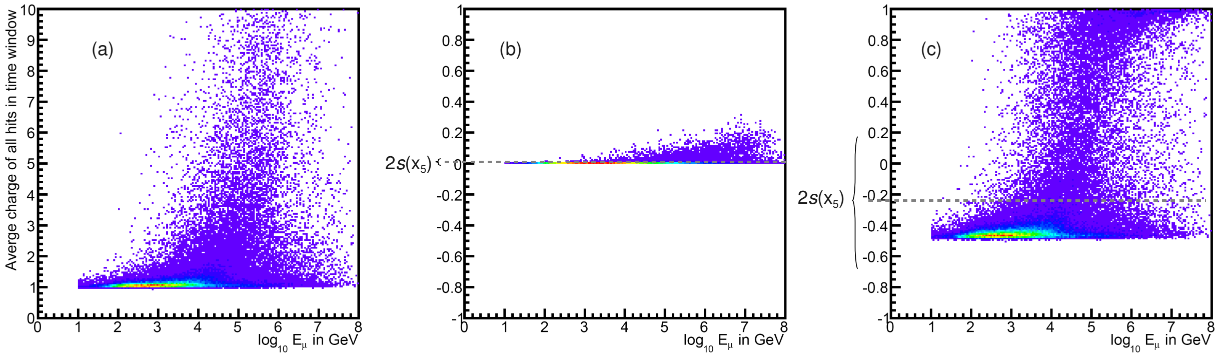


Figure 6.6: Distribution of P_5^S over energy (a) in natural units, (b) transformed using ($\alpha = 2\langle x \rangle$, $\delta = 0$) and (c) with maximized $s(x_5, \alpha)$

For α any value can be chosen to distribute the training set suitably while the offset δ allows generally positive values to be scaled down below 0. It has been suggested in [4] to use $\alpha = 2\langle x \rangle$ and $\delta = 0$, which limits most positive input parameters to a small interval around $[0, 0.2]$ with only a few exceptions at very high energies. This would limit the neural nets ability to extract significant features from the dataset.

To offer the net a larger dynamic range, an offset was introduced with $\delta = 2$. The input values of the training sample should be distributed widely within the available interval, which can be achieved by adapting α. For this purpose a program was written which maximizes the RMS $s(x_i, \alpha)$ of the normalized input values of the training set by varying α. Starting with $\alpha = \langle x_i \rangle$ it iteratively increased and decreased α by 5% and calculated $s_+(x_i, 1.05\alpha)$ and $s_-(x_i, 0.95\alpha)$. If s_- or s_+ proved to be larger than $s(x_i, \alpha)$, the process was repeated, using the increased or decreased α. This procedure was continued until a maximum was found for $s(x_i, \alpha)$.

In [4] the energy of the muon E_μ was scaled using

$$x_e = \frac{1}{10} \log_{10}(E_\mu), \tag{6.3}$$

as $\log_{10}(E_\mu)$ is generally applied to the energy scale for plotting. To use the input range to a larger extent, the energy is effectively transformed using

$$x_e = \frac{log_{10}(E_\mu)}{4} - 1. \tag{6.4}$$

Through the maximisation of $s(x_i, \alpha)$ for a given training set, the optimal performance of the net for the complete dynamic range is achieved. However, it might sometimes be of use to highlight a certain energy range of the dataset when specializing the net for a special task, e.g.. low energy reconstruction. In this case it should be considered to stress a certain range in the input parameter set by e.g. spreading small hit numbers wider than large hit numbers and adapting α and δ manually. For the following analysis only $s(x_i, \alpha)$ optimized transformation was used. All values for α can be found in appendix A

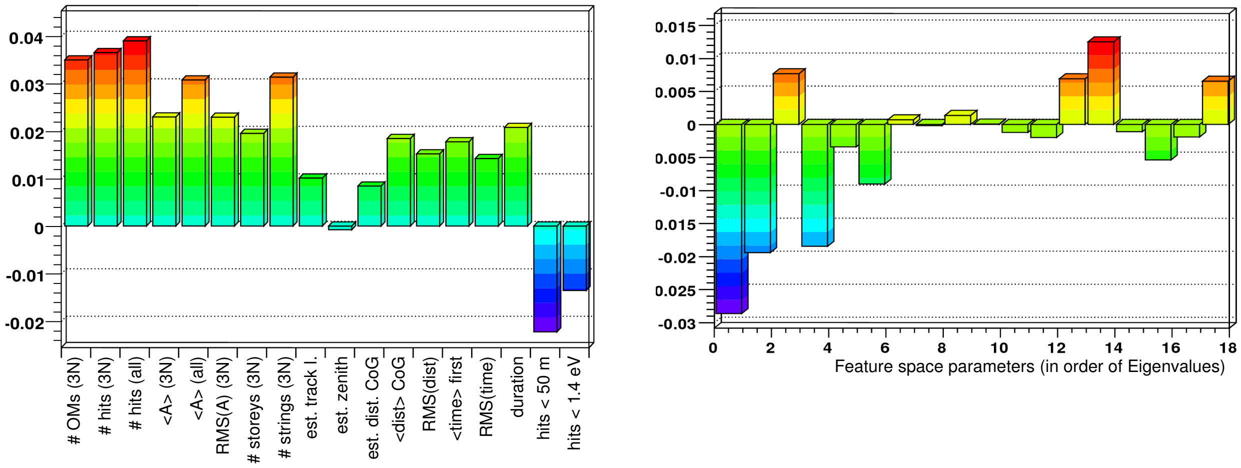


Figure 6.7: Covariance between input parameters of the simple estimator and muon energy: (a) the normalized pattern space parameters (b) the feature space parameters (starting with the parameter with largest Eigenvalue on the left)

6.3 Principle Component Analysis

When selecting a sample of possible input parameters two criteria must be met by each selected parameter. On the one hand it should obviously be energy dependent or at least be of relevance for the energy dependence of the other input parameters, on the other hand it should characterise the available parameter space in a good way in order to pass additional information to the neural net. The covariance between parameters was introduced in Section 5.1 to evaluate dependencies between parameters. Now the covariance matrix $\mathbf{C}$ will serve to find a representation for the input parameters which displays underlying features of the data set more effectively.

6.3.1 Principle Components Method

The covariance matrix can serve to find the main pattern forming components in a dataset. The “natural” space in which the parameter dependencies are displayed as they are derived from the available data, is called pattern space. The Principle Components Method (PCA) uses the Eigenvectors $\mathbf{e}_{c_i}$ and Eigenvalues λ_i of the covariance matrix $\mathbf{C}$ in its pattern space to find a transformation matrix $\mathbf{T}$ for the data set into its feature space. Feature space displays explicitly the independent features of the parameters and the covariance between the parameters is zero. In feature space the greatest variance in data lies on its first component while the following components vary only as much as they are independent from this principle component. The entries of the transformation matrix $\mathbf{T}$ therefore consist of the Eigenvectors $\mathbf{e}_{c_i}$ of $\mathbf{C}$ ordered by the value of their corresponding Eigenvalues λ_i (for derivation see [3]). The smaller λ_i, the less important is the corresponding Eigenvector $\mathbf{e}_{c_i}$. Input parameters $\mathbf{x}_i$ which contribute strongly to an important vector in feature space should thus also be of importance for pattern recognition. Finally, applying $\mathbf{T}$ to each input parameter set transforms the input into the feature space which can improve the networks adaptability to the problem, as will be shown in the next chapter.

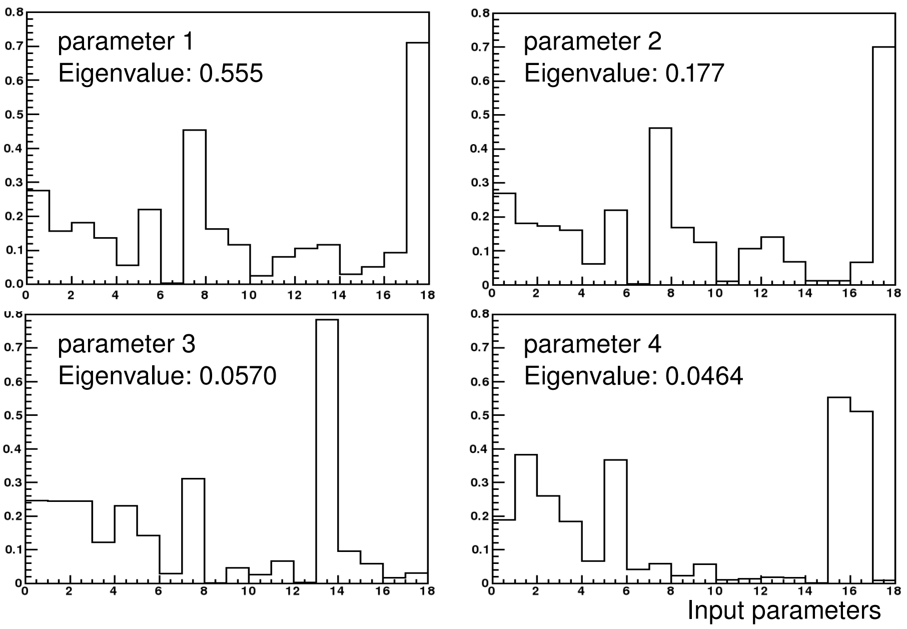


Figure 6.8: Contribution of input parameters to feature space parameters of parameters with largest Eigenvalues for the simple estimator

6.3.2 Input Parameters in Feature Space

Transforming pattern space parameters now generates feature space parameters which contain some information of almost all pattern space parameters to a certain extent. The parameters in feature space display a different energy dependence, as can be seen in Figure 6.7b, concentrating the energy dependent features on fewer parameters than in pattern space. Feature extraction has therefore already partly served the purpose of minimizing the degrees of freedom, as a smaller number of energy dependent parameters simplifies the procedure of adapting to these features for the ANN. It is interesting to note that the four feature space parameters with largest eigenvalues, which also display a high energy dependence, contain P_1^S to P_6^S to almost the same extend, which already here underlines the relative importance of these parameters. Whether the neural net trains actually on these main parameters and to which extent they are important for the result of the estimation will be explored in Chapter 8.

6.3.3 Rescaling of Parameters in Feature Space

After transformation the feature space parameters again have to be scaled down to $[-1, 1]$ for neural net use. This is now done through a simple linear transformation, as most parameters lie compactly within an interval around 0. The transformation applied is

$$x_j^{scaled} = \frac{x_j}{\gamma} - \eta \tag{6.5}$$

with γ and η being integer values. A listing of the exact values for each set can be found in appendix A. If any x_j^{scaled} lie above or below $[-1, 1]$ after transformation, they are set to 1 or -1 respectively.

Discarding feature space parameters can be helpful for the training of the neural net but has to follow analysis of the connection weights set in the training process. It will therefore be postponed to chapter 8.

Chapter 7

Adapting the Neural Net

7.1 Topology of the ANN

The topology of the network is defined by the network layout, i.e. the number of hidden layers and nodes therein, and the activation function. The Fast Artificial Neural Network (FANN) library [18] was used for the network training and thus some options of the functionality of this library which are commonly used in network training were explored to achieve good training results.

7.1.1 Network Layout

Generally feed-forward networks with two or three hidden layers should be sufficient for the task of energy estimation (see Section 4.2.1). To find the best topology, a fixed training set (A_1^{up} in estimator C) was processed through nets with varying numbers of nodes in two and three hidden layers. Using one training set is sufficient for these tests, as the phenomenology of the various training sets resemble each other in their main features. The training was stopped after 200 epochs and the MSE of the training sample plotted against the number of nodes in Figure 7.1.

Although a certain variation can be expected in the achieved MSE for the same topology in various training processes (see Section 7.2), a tendency of improving MSE for nets with two hidden layers can be found with a large number of nodes in the first hidden layer, although only 4 nodes in the second hidden layer seem to lead more often to failed training processes (red). Three hidden layers show the same tendency, but no general improvement can be seen in comparison to nets with only two hidden layers. Therefore the network layout was always chosen with two hidden layers and numbers of nodes indicated by the red circle in Figure 7.1. After repetitive training, the chance to reach an especially low MSE seems to be largest with this topology.

7.1.2 Activation Function

An adequate activation function is the main tool which can generalize the decision units derived from biological neurons, which only contain a step function, towards global application with smooth mappings between input and output layer. Some commonly used activation functions offered by FANN were tested and their steepness varied to find the most suitable solution.

Common functions The tested activation functions $g(a)$ for node input a, which were used in the hidden nodes as well as in the output layer, were

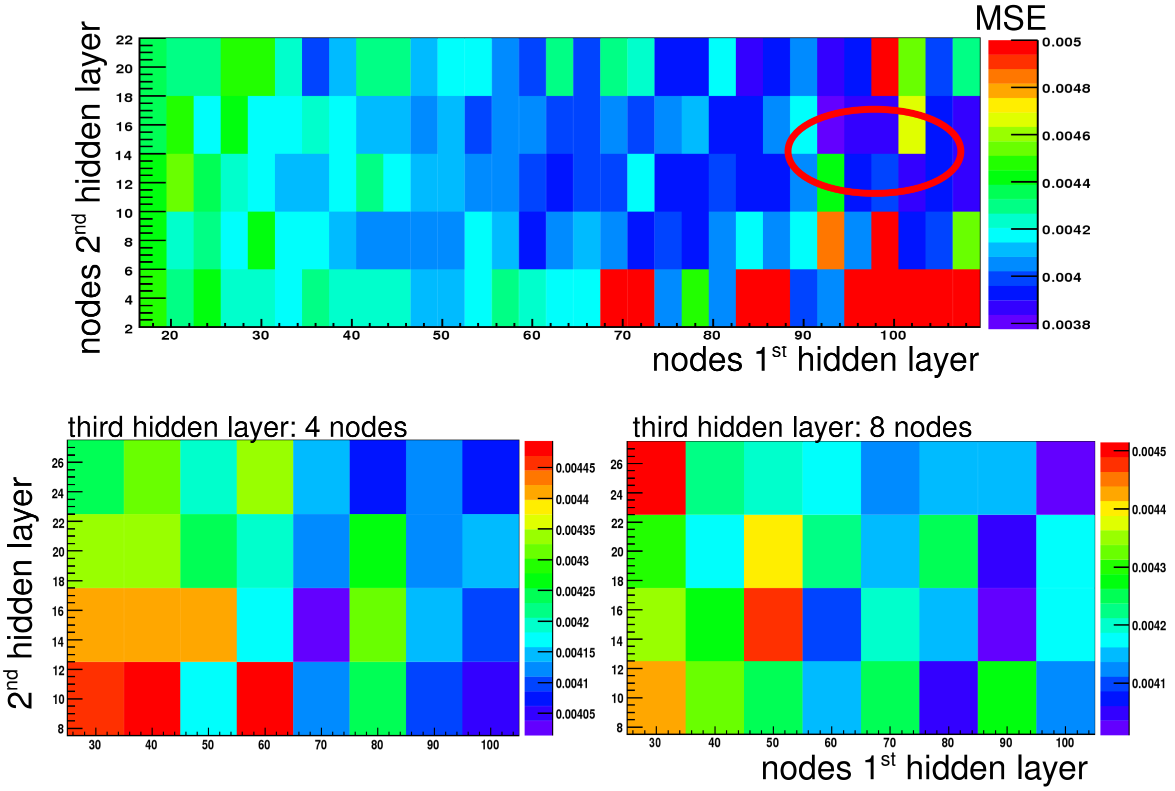


Figure 7.1: MSE after 200 epochs of training estimator C on set A_1^{up} with various nodes in the hidden layers. On top nets with two hidden layers, below nets with three hidden layers and 4 and 8 nodes in the third hidden layer respectively.

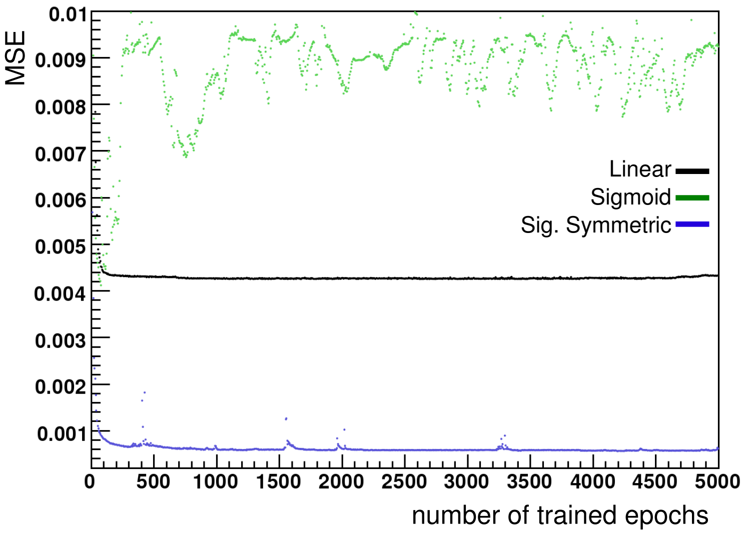


Figure 7.2: Comparison of MSE in training process with linear, sigmoidal and sigmoid symmetrical activation function

Linear : $g_{\text{lin}}(a) = s \cdot a$

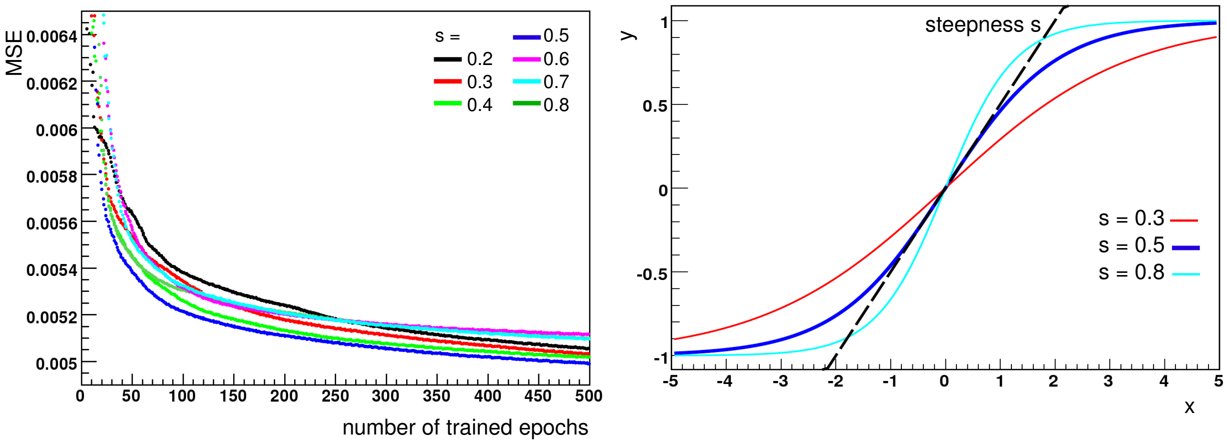
Sigmoid : $g_{\text{sig}}(a) = 1/(1 + e^{-2sa})$

Sigmoid Symmetric : $g_{\text{sigsym}}(a) = 2/(1 + e^{-2sa} - 1)$

The steepness s gives the gradient of the function at $a = 0$ and can be chosen freely. For this test it was kept constant on an average value, to $s = 0.5$. Three nets were trained using these functions, the result of which can be seen in Figure 7.2.

The sigmoidal activation function seems to cause large fluctuations during training and its output is limited to positive values, which adapts only poorly to the given input parameters. The linear and sigmoid symmetrical activation function are much more stable during training, although the linear function does not reach as good a performance as the latter. This might be connected to poor handling of extreme input values by the linear function, which would be smoothed out by the sigmoid symmetrical. Therefore the sigmoid symmetric activation function is used in both hidden and output layer.

Activation steepness The sigmoid symmetric activation function is shown in Figure 7.3b together with the definition of its activation steepness. The steepness can be adjusted freely and influences the adaptability of the network by either reducing nodes to decision units, if a great steepness is chosen, or smoothing out very large fluctuations in the input parameters through application of a small steepness.



(a) comparison of activation steepness

(b) sigmoid symmetric activation function

Figure 7.3: (a) Comparison of the MSE at training with various activation steepnesses (b) definition of the activation steepness in a sigmoid symmetric activation function

For the comparison in Figure 7.3a net 23/90/12/1 A_1^{up} as estimator C was trained three times to reduce the uncertainty in the comparison introduced through the varying behaviour in comparable training processes (see Section 7.2). The average for these three training processes per steepness s in steps of 0.1 is displayed. Although the difference is not large, an improvement can be observed towards smaller activation steepness. As $s = 0.5$ performs best in this comparison and a good balance between decision unit and limiting performance for small input fluctuations also seems to be a logical choice, the steepness in all nets is set to this value.

7.2 Training process

After setting the networks number of hidden layers, number of nodes in each layer and the activation function, the training process only has to be monitored and ended at the right moment to avoid over-fitting of the net to features which are peculiar to the training sample. To this end the MSE of an independent validation set is recorded. The stop criterion of the training process is met when the MSE of the validation sample does not rise in five subsequent steps. With the training conditions used, this was normally the case after a few hundred epochs (see Figure 7.4b).

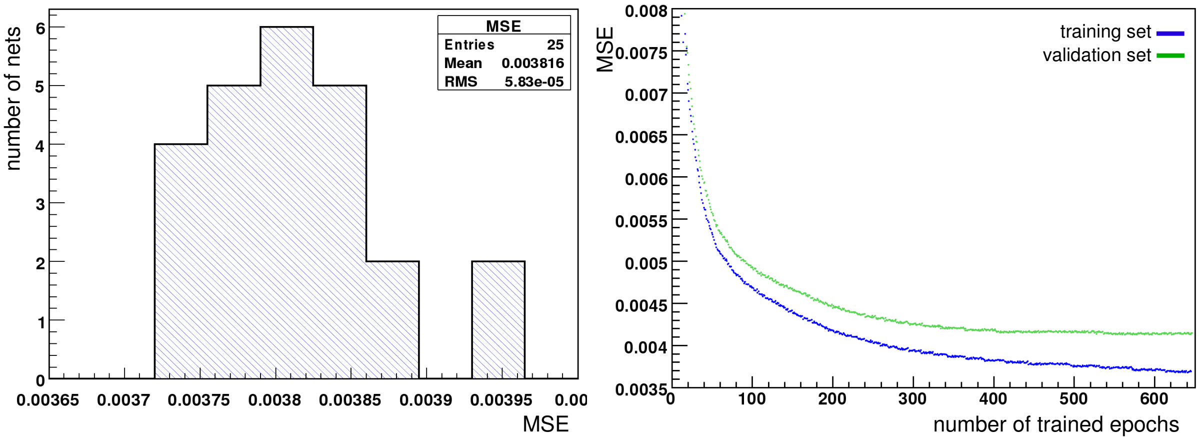


Figure 7.4: (a) distribution of the MSE of estimator C on A_1^{up} in various training processes finished by the stop criterion (b) MSE of training and validation sample in training until stop criterion is met through rising MSE in validation sample

It can generally be observed that, although the same topology and data set is used for a neural net, different MSE are reached until the stop criterion is met, as can be seen for the repeated training in Figure 7.4a. This has its cause in the randomized setting of the connection weights at the beginning of the training and the varying response of the minimizer on these initial conditions. Therefore, after repeated training of the same network configuration, the net with the smallest MSE is selected for the energy estimation and validation.

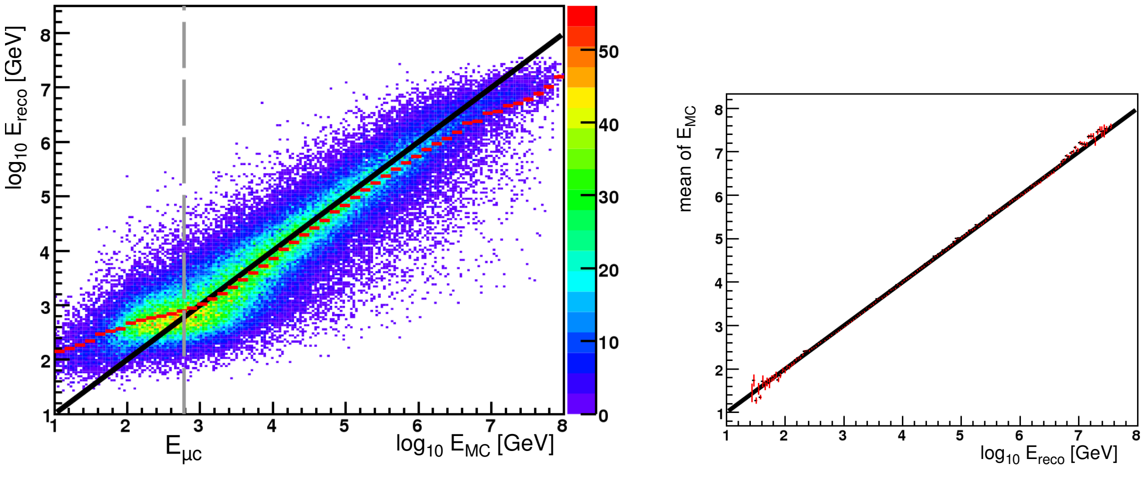
With the modifications mentioned here, neural nets were trained with the FANN library for all four training sets with input parameters from both estimators. While the exact numbers for each net can be found in Appendix A, the performance of these nets will be examined in the next chapter.

Chapter 8

Quality of ANN Energy Estimation

The selection of data sets and the choices for network configuration made in the last two chapters results in eight nets: Simple and Cherenkov estimator on ideal and real detector conditions, upgoing only and all zenith angles. The estimator performance can be analysed by evaluating the error of their reconstruction of testing sets B. After this, two questions need to be answered: Are the chosen parameters are actually contributing to the estimator performance? And secondly, can the method be applied to various detector conditions and how stable is the performance on real data? To this end, the Connection Weight Method and HIPR are applied in Sections 5.2 and 5.3.

8.1 Estimator Performance

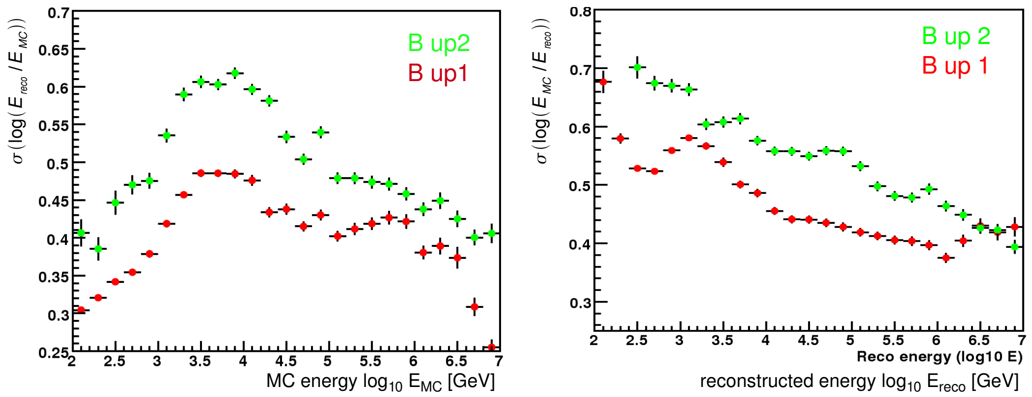


(a) MC vs reconstructed energy

(b) mean of MC energy

Figure 8.1: Performance of estimator S on set B_1^{up} (a) MC versus reco energy with mean reconstructed energy (red) (b) mean of the Monte Carlo energy per bin in reconstructed energy

The quality of a reconstruction method can be estimated in various ways. First, one can view the distribution of the reconstructed value from the true values point of view. This is done when evaluating the mean of the reconstructed energy per bin in MC energy (red lines in Figure 8.1a). In each of those bins a Gaussian fit can be performed on the difference between reconstructed and Monte Carlo energy. The width σ of the fit gives the standard deviation in case of an ideal Gaussian distribution. This is used as an error estimate



(a) σ over MC energy

(b) σ over reconstructed energy

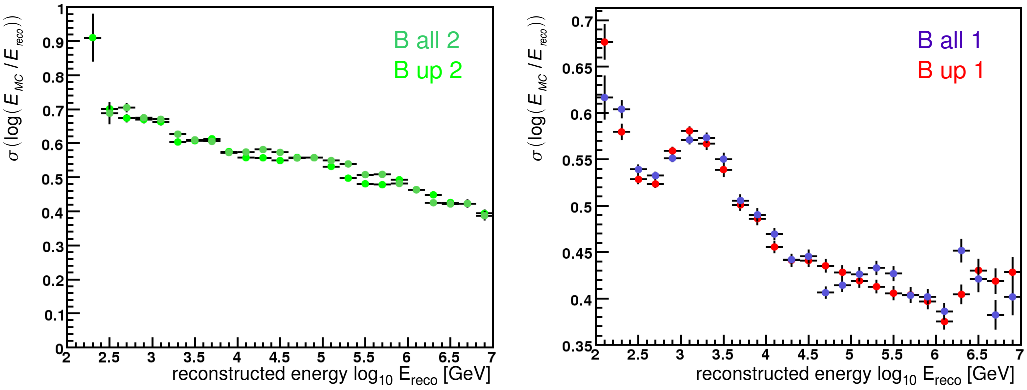
Figure 8.2: Performance of estimator S for testing sets B of real (2) and ideal (1) detector conditions in comparison: (a) over Monte Carlo energy and (b) over reconstructed energy

e.g. in Figure 8.2a. In experimental data the true value is not accessible and thus it is of importance to know the distribution of the true value for a measured result. Thus the mean of the MC energy in bins over reconstructed energy (Fig. 8.1b) and especially the width of this distribution (Fig. 8.2b) serve as a benchmark for reconstruction quality.

8.1.1 The Simple Estimator

The nets for both estimators were trained over the full range of detectable energies of muons in ANTARES, keeping in mind that the phenomenology of energy loss changes within this range at the critical energy $E_{\mu c}$ from almost linear ionization loss to exponential radiative losses. The Simple Estimator already displays this fact when applied to an upgoing muon set B_1^{up} (Fig. 8.1a). Estimator S can, for all used data sets, roughly estimate the energy to an accuracy of $\sigma = 0.5$ above $10 TeV$ (Fig. 8.2a) with the mean of the reconstructed energies showing a slight deviation from the MC energy towards higher energies(red lines in Figure 8.1a). Below $E_{\mu c}$, however, lower MC energies are interpreted towards higher energies, which indicates that the neural net trained to the phenomenology of radiative losses rather than ionization. Despite that, the mean of the Monte Carlo energies, when viewed in bins of the reconstructed energy, shows basically no deviation from the reconstructed mean (see Figure 8.1b). This might be attributed to the operating mode of the training algorithm which minimizes the MSE of the complete training set and thus balances the error of the various data samples until they display the smallest overall deviation from the true value. This good agreement of the mean of MC and reconstructed energy can be observed for all nets. Therefore the width σ of MC energy per reconstructed energy gives the overall error for a reconstructed energy without need for mean correction.

Looking closer at Figure 8.2, the same features which have already been observed in Figure 8.1a can be seen here again, as these Figures can be viewed slices in x and y direction through the distribution in Figure 8.1a. The distribution over Monte Carlo energy (Fig. 8.2a) shows a smaller width for low energies, as all event below $E_{\mu c}$ are interpreted towards higher energies, resulting in distributions which show a mean deviation towards those higher energies but a smaller σ. The same effect accounts for the “bump” at about 1 TeV in Figure 8.2b. Around this reconstructed energy, events with much smaller MC energies can be reconstructed thus due to the incapability of the estimator to distinguish low energy events with ionization losses from events at higher energies which cause photons primarily through

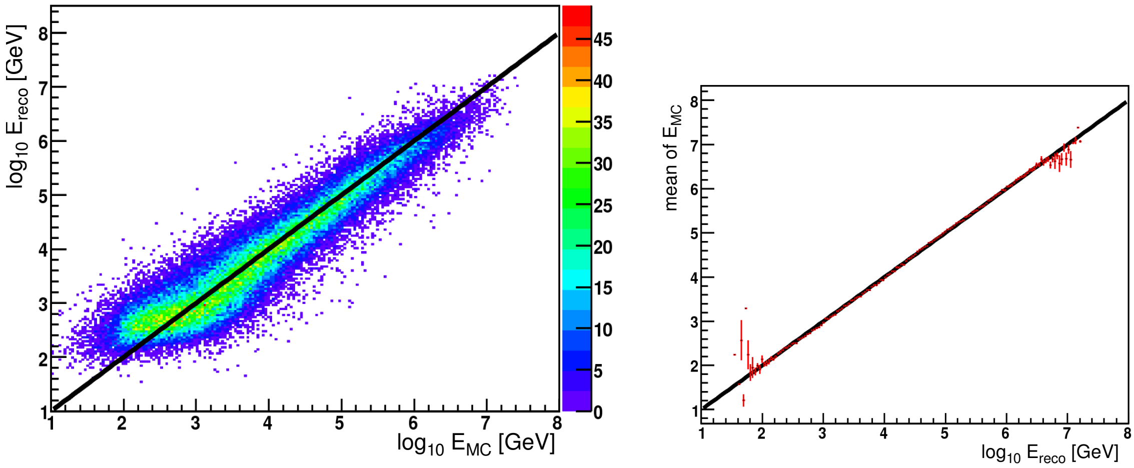


(a) σ for real detector

(b) σ for ideal detector

Figure 8.3: Performance of estimator S for sets of only upgoing muons (up) and from all zenith angles (all) (a) for real detector (2) (b) for ideal detector configuration (1)

radiative process, resulting in a large σ of the distribution of MC energy.



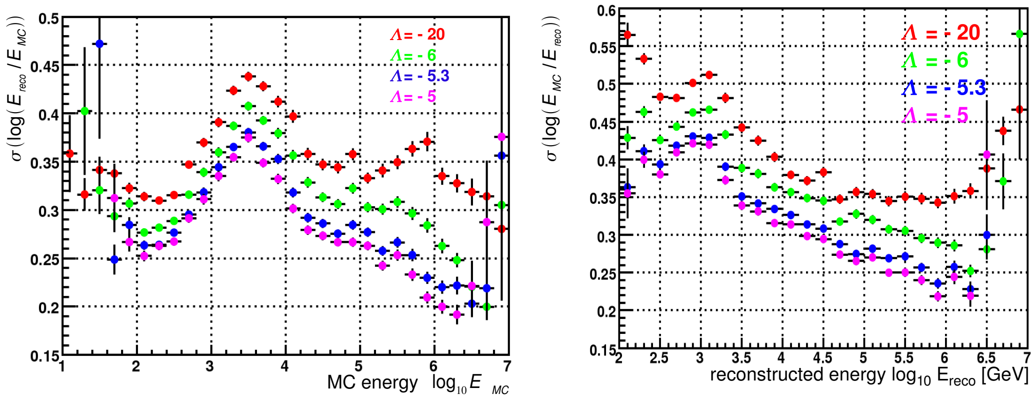
(a) MC vs reco energy

(b) mean of MC energy

Figure 8.4: Performance of estimator C on set B_1^{up} for $\Lambda \geq -5.3$ (a) MC versus reco energy (b) mean of the Monte Carlo energy per bin in reconstructed energy

As will be seen in Section 5.2, estimator S uses mainly input parameters which are strongly dependent on the background rate and detector efficiency, such as the number of triggered and detected hits and the number of triggered strings and storeys. For comparison estimator S was trained on the real detector conditions with four missing lines and 100 kHz mean background rate (sets B_2) and compared to the ideal detector in Figure 8.2. As expected the performance of estimator S is strongly impaired by these worse conditions. However, the simple estimator can still be used for a rough selection between high and low energy events, which also was its initial purpose.

The detector's larger sensitivity for photons coming from below compared to those coming from above lead to the introduction of the track estimates in Section 6.1.1. Although the main aim of the ANTARES detector is the measurement of particles coming through the Earth, analysing downgoing muons is of interest for a broad range of physics goals. For



(a) σ over MC energy

(b) σ over reconstructed energy

Figure 8.5: Resolution of energy reconstruction of Cherenkov estimator on set B_1^{up} for various fit quality cuts on the reconstructed tracks: (a) vs MC energy (b) vs reconstructed energy

the application of estimator S for muons from all directions, the zenith angle of the track estimate seems to be sufficient, as no large deviation from the results for upgoing samples can be seen in Figure 8.3.

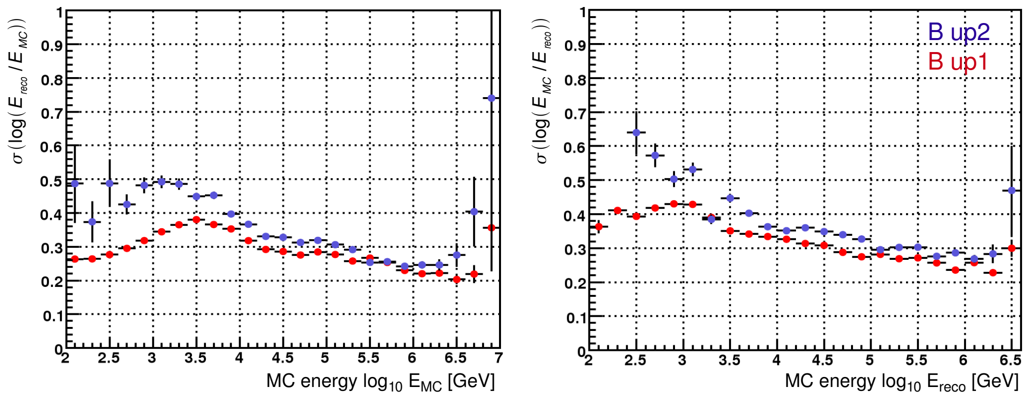
8.1.2 The Cherenkov Estimator

The performance of the simple estimator can be seen as an upper limit for the performance that can be reached for events where track reconstruction fails. For events with successful track reconstruction, the Cherenkov estimator increases the quality of the energy estimation. As can be seen in Figure 8.4a, the problem of overestimation at low energies remains, although being less pronounced than for estimator S. The mean of the Monte Carlo energies for a fixed reconstructed energy coincides with the reconstructed energy (Figure 8.4b), as was also already the case for estimator S.

The quality of the track reconstruction is measured by the parameter Λ, which increases with the quality of the reconstruction (see Section 6.1.3). A selection of events with $\Lambda \geq -20$ signifies basically no cut and shows only a slight increase of performance compared to estimator S, stronger cuts on Λ also increase the performance of the energy estimation (Fig. 8.6). While MC energies below 1 TeV are still more difficult to estimate and reach only a factor of 2.5 ($\sigma = 0.4$) for the deviation of the reconstructed energy with $\Lambda \geq -5.3$, above 10 TeV this resolution can be improved below factor 2. Above 1000 TeV finally the resolution decreases due to boundary effects.

The Cherenkov estimator is trained not only on the basic input parameters but also on time residual distributions and distances (see Section 6.1.4), which are not very dependent on detector layout and background rate. Therefore applying the Cherenkov estimator on the "real" set B_2^{up} shows only a slight decrease in the energy resolution (Figure 8.6). This makes estimator C applicable to background rates around 100 kHz and for the 10 or 12 line detector configuration without expecting a significant loss of performance. Most significantly, the decrease can again be noted in the low energy regime as here the signal events are drowned in the increasing amount of background hits.

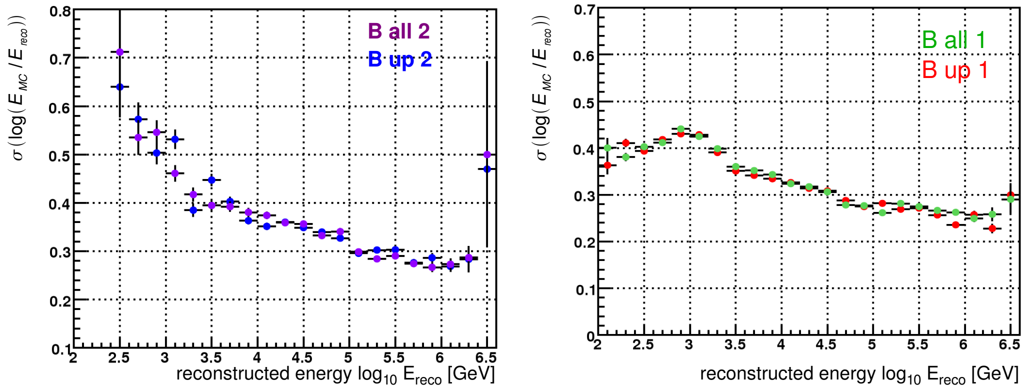
The zenith angle given by the track estimate of Estimator S already proved to be good enough to allow the neural net to compensate for the varying sensitivity of the OMs with the direction of the detected photon. The reconstructed track serves this purpose even



(a) σ vs MC energy

(b) σ vs reconstructed energy

Figure 8.6: Energy resolution of Cherenkov estimator for real (2) and ideal (1) detector in comparison (a) Monte Carlo energy (b) reconstructed energy ($\Lambda \geq -5.3$)



(a) σ for real detector

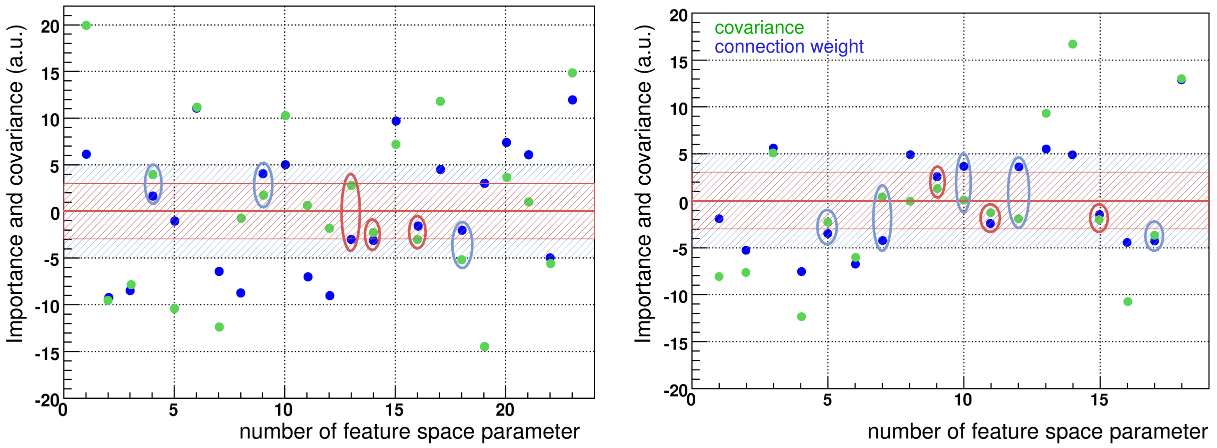
(b) σ for ideal detector

Figure 8.7: Performance of estimator C ($\Lambda \geq -5.3$) for sets of only upgoing muons (up) and from all zenith angles (all) (a) for real detector (2) (b) for ideal detector configuration (1)

better and therefore basically no difference in performance can be seen for upgoing muons compared to muons from all zenith angles (see Figure 8.7). Estimator C has therefore a wide range of application for various detector layouts, background rates and incident angles.

8.2 Parameter Importance in the Estimators

It is difficult to analyse the overall functionality of a neural net due to the multidimensionality of the problem. Nevertheless, the connection weight method (Sec. 5.2) offers a tool to estimate the importance Imp$CW(x_i)$ of an input parameter to a certain extend. It is used here to compare the importance of the feature space (fs) parameters which have been transformed through PCA with their energy covariance. Secondly, taking into account that the pattern space (ps) parameters contribute all to a certain extend to these fs parameters, the importance of the primary ps parameters is calculated by application of the transformation matrix **T**.



(a) importance of feature space parameters estimator C

(b) importance of feature space parameters estimator S

Figure 8.8: Comparison of connection weights and energy covariance of the feature space parameters in A_1^{up} of: (a) estimator C, (b) estimator S, covariance scaled to the same magnitude as importance

8.2.1 Evaluating the Feature Space Parameters

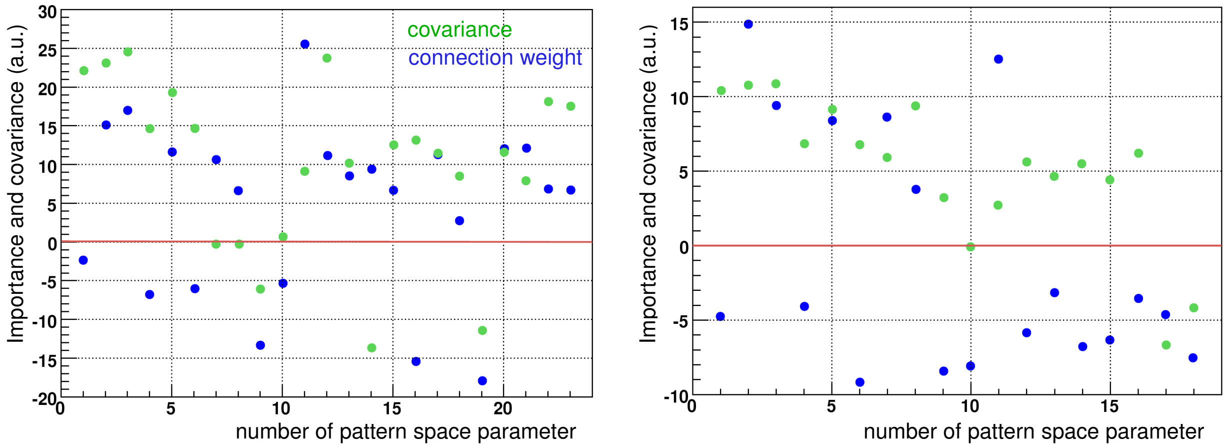
Some feature space parameters show a high energy dependence while covariance between some other parameters and energy is extremely small (Sec. 5.1). If also the importance of the latter is small, these parameters might be discarded to increase the adaptability of the neural net to the remaining important features. For both estimators the direct input parameters in feature space were scaled to the same magnitude (Sec. 6.3.3). For better comparison the covariance of each fs parameter was therefore scaled by dividing its value by γ. Although not always a correlation between importance and covariance can be observed, a general tendency exists for important input parameters to also have higher energy covariance (Fig. 8.8). Those parameters with little influence, are singled out in the red and blue dashed bands in Figures 8.8. If both covariance and importance of a parameter lies within these boundaries, it is selected (blue and red circles). After applying HIPR to the estimators for confirmation in Section 5.3, the influence of these selections to the estimator performance will be tested.

8.2.2 Evaluating the Pattern Space Parameters

The connection weights can also be assigned to the original input parameters in pattern space by summing up the connection weight of the feature space parameters weighted with the contribution of the patterns space parameter to it, expressed in the transformation matrix T_{ij}.

$$Imp_{CW}(x_i) = \sum_j T_{ij} Imp_{CW}(\tilde{x}_j) \tag{8.1}$$

A comparison in Figure 8.9 of the importance of the input parameters common to both estimators shows that both estimators seem to use them to roughly the same extent. For the overall impact of a parameter on the estimator performance, only the absolute value of covariance and connection weights is of importance, as parameters which are inversely proportional to the energy can contribute to the ANNs output when weighted negatively. For



(a) importance of feature space parameters estimator C

(b) importance of pattern space parameters estimator S

Figure 8.9: Comparison of connection weights and energy covariance of the pattern space parameters in A up1 of: (a) the Cherenkov estimator , (b) the Simple estimator, covariance scaled to the same magnitude as importance

estimator S the importance of the common parameters also seems to exceed the importance of parameters $P^S_{12} - P^S_{18}$, underlining the greater response of the S estimator's performance on varying detector conditions (see Section 8.1.1). It also shows that the zenith angle P^{SC}_{10} is important for both estimators, although it is obviously not energy dependent. This explains the good estimator performance for all zenith angles.

8.3 Finding a Minimal Representation of the Input Parameters

The response Resp of the HIPR method (Sec. 5.3) can be used to confirm the small importance of a selection of parameters as described in Section 5.4. The energy covariance, importance from the CW method and response from HIPR should be low for these parameters. For better comparison, the output of these methods was normalized to sum up to 100 in the Table 8.1, which displays the result for the methods for the simple estimator tested on set A_1^{up}.

The comparison suggests to discard the red selection from Figure 8.8b. To finally discard these parameters, a new net must be trained without these parameters and its variance compared to that of the net with full input sample. However, discarding these parameters only seems to lessen the performance (Figure 8.10). The same observation can be made when trying to discard the selection for the Cherenkov estimator. The selected parameters might be the least significant, but are still of importance to the overall performance. This indicates that the parameter selection for estimators S and C is diverse enough to hand additional relevant features to the net in each parameter, although this is no guarantee that it is exhaustive.

FS parameter	$Resp(x_i)$	covariance	$Imp_{CW}(x_i)$	discard it?
1	8.92794	52.19	-3.74	never
2	14.5679	35.36	-28.27	never
3	4.74376	-13.98	32.5	never
4	21.7843	33.62	-56.98	never
5	3.88651	6.19	-11.96	
6	7.57665	16.45	-43.93	never
7	2.68133	-1.28	-18.63	
8	3.65291	0.34	24.79	
9	1.54247	-2.5	6.3	yes?
10	1.4078	-0.12	13.54	
11	2.14901	2.21	-5.91	yes?
12	2.06967	3.61	13.51	
13	5.93493	-12.6	30.54	
14	6.94829	-22.79	24.01	never
15	1.68429	1.99	-2.11	yes?
16	2.71197	9.79	-20.39	
17	1.92346	3.45	-17.99	
18	5.80685	-11.95	164.71	never
$\sum$	100	100	100	

Table 8.1: Response, covariance and Importance for Estimator S parameters

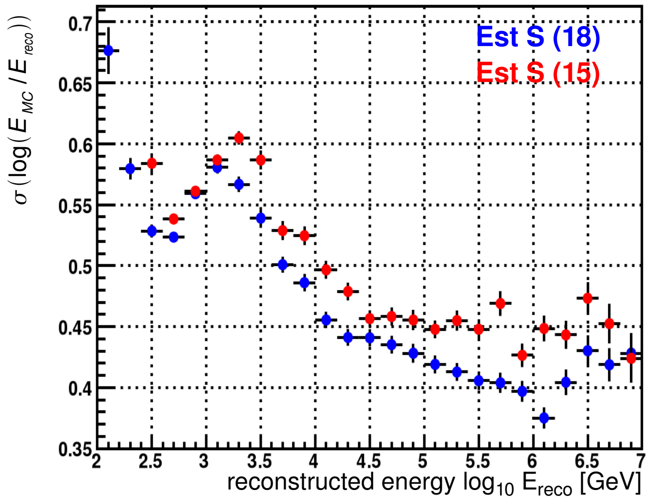


Figure 8.10: Comparison of estimator B with all 18 input parameters (blue) and after discarding three parameters (red)

Chapter 9

Comparison of Monte Carlo Simulation with Data

The recognition performance of an ANN for data can only be as good as the Monte Carlo simulation that has been used for training. In this chapter the quality of the energy estimation for data will be explored. The Cherenkov estimator is applied to data for five different runs in Section 9.1. As detector condition and background rates are subject to changes, the stability of the estimators for varying conditions are explored in Section 9.2. Finally, further improvements of energy estimation are briefly considered in Section 9.3.

9.1 Input Parameters in Data

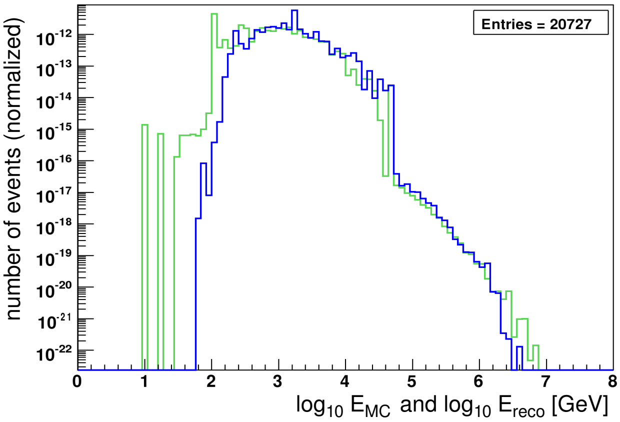


Figure 9.1: Muon spectrum from Monte Carlo (green) and reconstructed Monte Carlo (blue) in estimator C for $\Lambda \geq -5.3$, normalized to 1

As the ANN estimators train of Monte Carlo simulations, the validity of the reconstruction result depends on the agreement of data and simulation for the relevant input parameter. For the comparison, the distribution of the input parameters in Monte Carlo with real detector condition and background events from minimum bias data of a run, weighted to represent the expected muon fluxes at the ANTARES site was used. The simulated parameter was then plotted against the input parameter distribution for the respective run and normalized to an area of 1 (Fig. 9.2). The largest deviation for an input parameter can be seen for P_5^C, the other parameters show almost no deviation apart from statistical fluctuations. Training the neural nets on Monte Carlo simulation adapted as described above should therefore be sufficient to ensure a good performance for data.

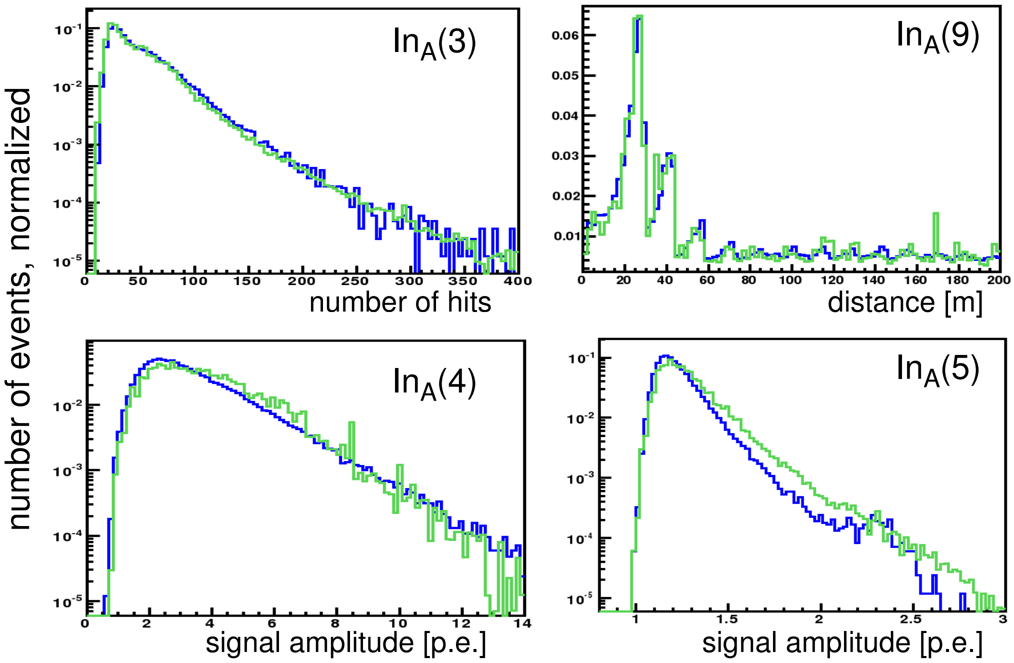


Figure 9.2: Input Parameters of Estimator A in data (blue) and Monte Carlo (green), distributions normalized for comparison

Reconstruction of the Monte Carlo spectrum with Estimator C shows the problem of the estimator at low energies mentioned in section 8.1.2. A good agreement can be observed above 1 TeV, and even the drop in statistics which is due to simulation can be seen in the reconstructed energy in Figure 9.1.

9.2 Stability of Estimator Performance

9.2.1 Estimator Response on Variation of Pattern Space Parameters

The response Resp of the HIPR method to variation of the pattern space input parameters instead of feature space parameters can be used to find the input parameters to which the estimator is most sensitive. If the pattern space parameters are varied between their minimum and maximum and the altered input set in feature space fed into the net after the complete preprocessing chain, the response of the neural net is a measure for the stability of the estimator to altered detector conditions. The results for estimator S of the HIPR method for set A up1 can be found in Appendix B. The greatest response for estimator S seems to lie in P_1^C, the number of triggered OMs. Unfortunately, the number of available OMs varies with the deployment status of the detector and bursts of bioluminescence that can saturate an OM's photomultiplier temporarily. Therefore new nets should be trained for detector layouts which show a larger difference in number of active OMs.

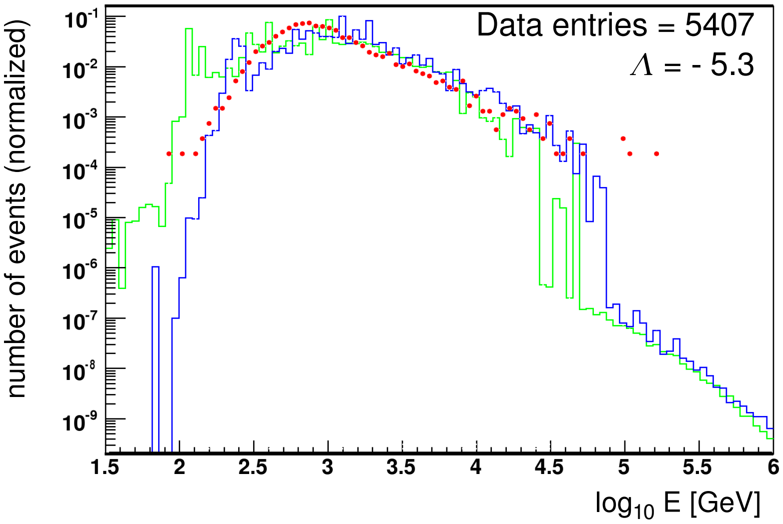


Figure 9.3: Muon spectrum from data (red), normalized for comparison with Monte Carlo (green) and reconstructed Monte Carlo (blue) in the Cherenkov estimator for $\Lambda = -5.3$

9.2.2 Testing against Background and Detector Condition

For basic input parameters like the number of hits in a certain time window one would expect a large dependence of the estimator performance on different background rates. As an example the net trained for a run with 63 kHz background and 153 failing OMs (run 42959) was used on the MC produced for a run with 101 kHz background and 260 failing OMs (run 43285). The result can be seen in Figure 9.4. Although muons around 1 TeV are reconstructed approximately right, the energy of muons with higher energy tend to be underestimated. The larger number of missing OMs in the Monte Carlo sample seem to be interpreted by the estimator as a lack of hits and thus a smaller energy is estimated compared to the real one. On the other hand the larger background rate seems to have not such a large effect on the reconstruction, which indicates that estimator C is more stable for different background rates than estimator S.

To use the estimators for all available data, one should therefore consider training various nets which take into account the different OM conditions and large variations in background rates.

9.3 Specializing for Various Tasks

The performance of the estimators below the critical muon energy $E_{\mu c}$ is worse compared to above $E_{\mu c}$, as input parameters were chosen to represent the phenomenology of radiative losses like the increasing number of photons from loss mechanisms. For the low energy regime, it could however be considered to train nets which put a greater emphasis on the track length of the muons in the detector, the existence of a vertex shower and similar criteria that are mostly connected to these events.

In the high energy regime the performance might still increase if more effective parameters can be found. If muons with energies above 1 PeV should be reconstructed, one can consider training nets only above the critical energy and increasing the statistic in the PeV region to

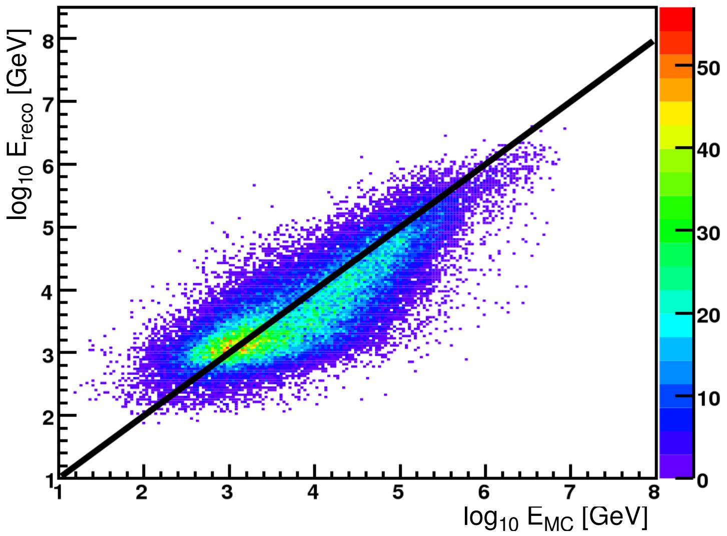


Figure 9.4: Reconstruction of the Monte Carlo sample with conditions of run 43285 reconstructed by estimator C trained on run 42959

avoid boundary effects visible in the nets presented in Chapter 8. The result of training on high energy events only can be seen in Figure 9.5, although here the statistics of those events for the training sample had not been increased. The performance remains the same for the largest part of the spectrum, but a slight improvement at the high end of the spectrum can be seen, although the effect is not significant due to small statistics.

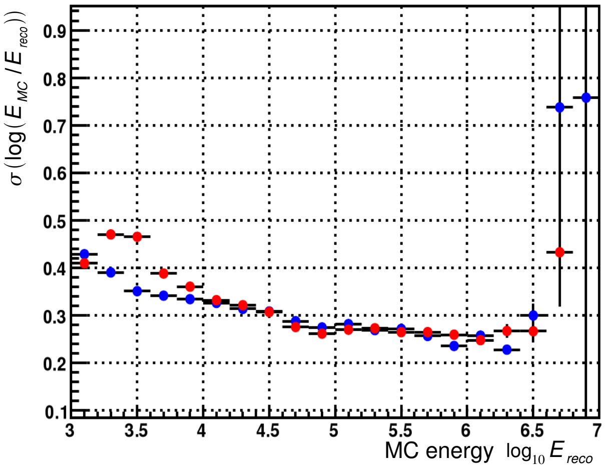


Figure 9.5: Comparison of the performance of estimator C trained on all available energies (blue) and on events with primary energies above 1 TeV only (red)

To allow users of the estimator to apply ANNs for special tasks like high energy or low energy reconstruction, one should consider supplying tools for training on independent Monte Carlo sets and input parameters when implementing the estimators in the ANTARES analysis framework.

Chapter 10

Summary and Outlook

Artificial neural networks are a powerful tool for probability density estimation. In the context of energy reconstruction for muons traversing the ANTARES detector their potential for feature extraction and modelling could be shown with two energy estimators, the Simple Estimator and the Cherenkov Estimator. For the Simple Estimator, input parameters from basic hit selections containing either only triggered hits or all hits in a time window around the triggered event were used. The ANN was trained on this parameter set after a parameter transformation into feature space and it could be demonstrated that it adapted well to the relevant features in the high energy regime. Above a few TeV, the Simple Estimator reached a resolution of $\sigma = 0.5$ in $\log_{10}(E_{\mathrm{MC}}/E_{\mathrm{reco}})$ or accuracy of a factor of 3. Although in comparison to other established methods not extremely accurate, the Simple Estimator has the advantage to be applicable to data at the trigger level and is therefore independent of track reconstruction. This property makes it fit for a pre-selection of high or low energy events before further analysis.
The most accurate description of muon energy loss is possible if the track of the muon is known. To this end the data set for the Cherenkov Estimator was derived from hit positions, time and charge with reference to direct Cherenkov photons emitted along the reconstructed muon track. The ANN training led to a good representation of the probability density function with an resolution better than $\sigma = 0.33$ in $\log_{10}(E_{\mathrm{MC}}/E_{\mathrm{reco}})$ or factor 2.1 for muon energies larger than 10 TeV, increasing towards higher energies. The Cherenkov Estimator can therefore be used for a precise analysis of high energy muon spectra, however it is only effective above a few hundred GeV to a factor of 3 depending on detector conditions and not suited for low energy estimation.

Specialization to low energies ANNs can adapt to various features inherent to a set of data, as long as the data is a good representation of the given problem. It was shown that Simple and Cherenkov Estimator adapt to the phenomenology of radiative energy losses, as specialization of the ANN to energies above 1 TeV did not significantly improve the estimator performance. Due to that, the estimators cannot be used below 100 GeV. If the method of ANN energy reconstruction should be applied to the low energy range, it should be considered to search for a new set of input parameters which better represent relevant low energy features, e.g. the track containment in the detector.

Application to a wide range of data Both estimators were tested for their zenith angle dependence, which proved to be negligible, and for various detector conditions and background rates. A strong fluctuation in the background rate was shown to be problematic for the Simple Estimator and both estimators are sensitive to the number of functioning

OMs. Therefore a new ANN was trained for each run for the application of the Cherenkov Estimator to data in Chapter 9. For an implementation as a tool in the standard analysis framework of ANTARES, this would be impracticable. Therefore the exact response of the estimators on variation in these two features must be explored and ANNs trained for various conditions. For each set of data, the tool would then pick the ANN which best resembles the conditions.

Last but not least, an error estimate has to be found which can be applied to the estimator results on an event by event basis, as can be seen from the missing error bars of the reconstructed muon spectrum in Figure 9.3. To this end, the similarities between Maximum Likelihood and ANNs might serve as a guideline.

Although once invented to build an artificial brain, artificial neural networks proved not to be capable to "think" on their own or arrive at independent behaviour. On the other hand they "outthink" us easily when it comes to modelling complex features from data - if we only think of the proper data first.

Appendix A

Datasets and Preprocessing

A.1 Testing and Validation Sets

The MC sets are described by

use of sample A = training set; B = testing set; V = validation set
direction of muon tracks up = upgoing; down = downgoing
background and detector ideal (1) = 60 kHz background, full detector (r12_c00_s01); real (2) = background from minimum bias events and detector conditions of run 44141 (mean rate: 100 kHz, line 6,9,10,12 not working, 203 not working OMs)

Testing Sets The Monte Carlo samples were taken from

upgoing km3 antineutrino and neutrino files, high and low energy
downgoing km3 antineutrino and neutrino files, high and low energy; corsika proton files (horizontal & vertical, all energies)

and mixed to display all available energies and zenith angle roughly to the same extend.

Validation Sets The validation sets V are a subset of the testing set by simply taking every fifth event of set B.

A.2 Preprocessing

A.2.1 Input Parameters

Available Data per event

- 3N triggered hits (3N)
- all hits with hittime $t \in [t_{first} - 20\,\mathrm{ns}, t_{last} + 300\,\mathrm{ns}]$ (all), with t_{first} and t_{last} being the first and last 3N triggered hit respectively

Simple Estimation Strategy (Estimator S)

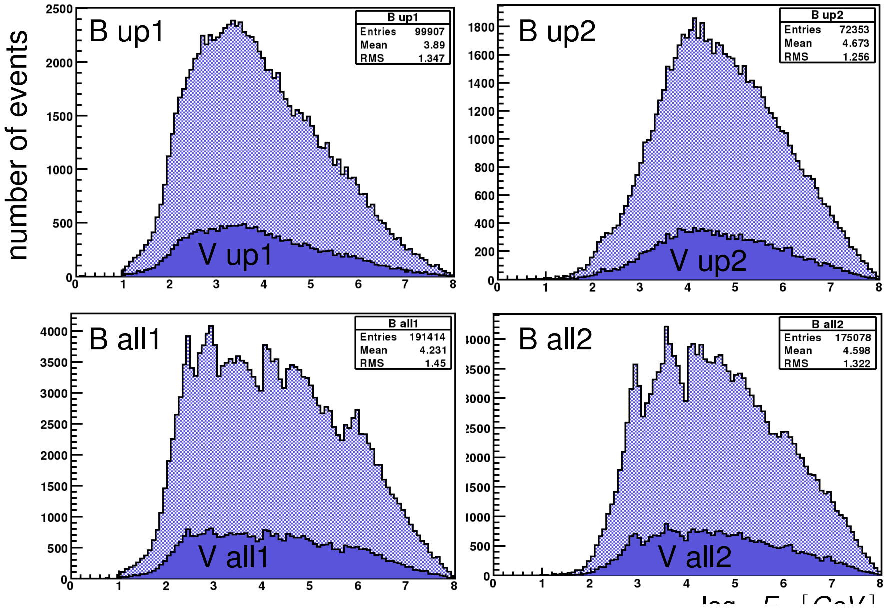


Figure A.1: Energy distribution of the test sets (B) and validation sets (V)

P^S_1	number of OMs (3N)	number of all OMs with a triggered hit
P^S_2	number of hits (3N)	number of all triggered hits
P^S_3	number of hits (all)	number of all hits in time window
P^S_4	average charge of hits (3N)	average charge of all triggered hits
P^S_5	average charge of hits (all)	average charge of all hits in time window
P^S_6	RMS of charge (3N)	RMS of the charge distribution on triggered hits (3N)
P^S_7	number of storeys (3N)	number of different storeys which have triggered hits
P^S_8	number of strings (3N)	number of strings with triggered hits
P^S_9	distance first and last hit (3N)	distance between OMs with first and last triggered hit
P^S_{10}	zenith first and last hit (3N)	zenith angle of track between first and last triggered hit
P^S_{11}	distance event and detector CoG (3N)	distance between detector center and the center of gravity (CoG) of the triggered hits
P^S_{12}	average distance from CoG (3N)	average distance of OMs with triggered hits from event CoG
P^S_{13}	RMS distance from CoG (3N)	RMS of the distances of OMs with triggered hits from event CoG
P^S_{14}	average hit time after first hit (3N)	average of all hit times starting from the first triggered hit
P^S_{15}	RMS hit times after first hit (3N)	RMS of all hit times starting from the first triggered hit
P^S_{16}	event duration (3N)	time difference between the first and last triggered hit
P^S_{17}	hit fraction on first string (3N)	number of triggered hits with distances $< 50\,\mathrm{m}$ from first triggered hit over number of all triggered hits
P^S_{18}	fraction with charge $< 1.4\,\mathrm{eV}$ (3N)	number of triggered hits with small charge over number of all triggered hits

Cherenkov Estimator (Estimator C) Additional Information: reconstructed track from the Aart Strategy.

P_1^C	number of OMs (3N)	number of all OMs with a triggered hit
P_2^C	number of hits (3N)	number of all triggered hits
P_3^C	number of hits (all)	number of all hits in time window
P_4^C	average charge of hits (3N)	average charge of all triggered hits
P_5^C	average charge of hits (all)	average charge of all hits in time window
P_6^C	RMS of charge (3N)	RMS of the charge distribution on triggered hits (3N)
P_7^C	number of storeys (3N)	number of different storeys which have triggered hits
P_8^C	number of strings (3N)	number of strings with triggered hits
P_9^C	track length in detector (3N)	distance between points along track perpendicular to first and last triggered hit
P_{10}^C	zenith of track	zenith angle of the reconstructed track
P_{11}^C	distance track and detector CoG (3N)	Closest approach distance between track and detector Center of Gravity (CoG)
P_{12}^C	average Cherenkov distance (all)	average distance of all hits in time window from track under Cherenkov angle
P_{13}^C	RMS Cherenkov distance (all)	RMS of distance of all hits in time window from track under Cherenkov angle
P_{14}^C	average time residual (all)	average time residuals of all hits in time window
P_{15}^C	RMS time residuals (all)	RMS time residuals of all hits in time window
P_{16}^C	event duration (3N)	time difference between the first and last triggered hit
P_{17}^C	average residual per distance (all)	average of time residual per Cherenkov distance of hits in time window
P_{18}^C	RMS residuals per distance (all)	RMS of time residual per Cherenkov distance of hits in time window
P_{19}^C	fraction residuals $< 25\,\mathrm{ns}$ (all)	number of hits with time residuals $t_r \in [-10\,\mathrm{ns}, 25\,\mathrm{ns}]$ over number of all hits in time window
P_{20}^C	fraction residuals $< 250\,\mathrm{ns}$ (all)	number of hits with time residuals $t_r \in [-10\,\mathrm{ns}, 250\,\mathrm{ns}]$ over number of all hits in time window
P_{21}^C	fraction residuals $> 600\,\mathrm{ns}$(all)	number of hits with time residuals $t_r > 600\,\mathrm{ns}$ over number of all hits in time window
P_{22}^C	average charge residuals $< 25\,\mathrm{ns}$ (all)	average charge of hits with time residuals $t_r \in [-10\,\mathrm{ns}, 25\,\mathrm{ns}]$
P_{23}^C	average charge residuals $< 250\,\mathrm{ns}$ (all)	average charge of hits with time residuals $t_r \in [-10\,\mathrm{ns}, 250\,\mathrm{ns}]$

A.2.2 Transformation of Parameters

Parameters for transformation in 6.2

$$x_i^{scaled} = \frac{2}{\pi} \arctan\left(\frac{\mathrm{P}_{x_i}^{\mathrm{S}}}{\alpha} - \delta\right) \tag{A.1}$$

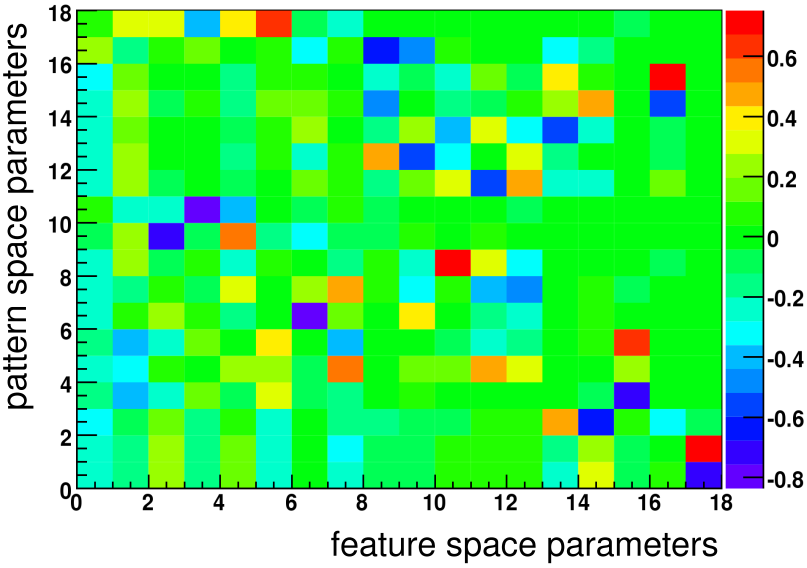
where $\delta = 2$ for all parameters where the transformation is applicable.

As an example the parameters for set $A_1^{\mathbf{up}}$ are listed for both estimators.

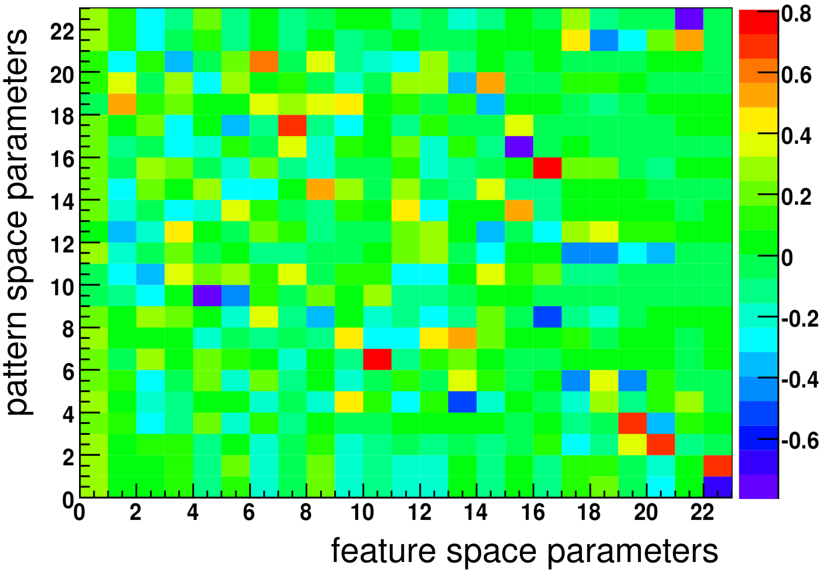
A.2.3 PCA

The Eigenvector matrix **T** transforms the parameters in pattern space into feature space; plotted here as an example is **T** for set A_1^{up}. The PCA of different training sets used in an estimator did not show a strong deviation from each other.

Simple Estimator



Cherenkov Estimator



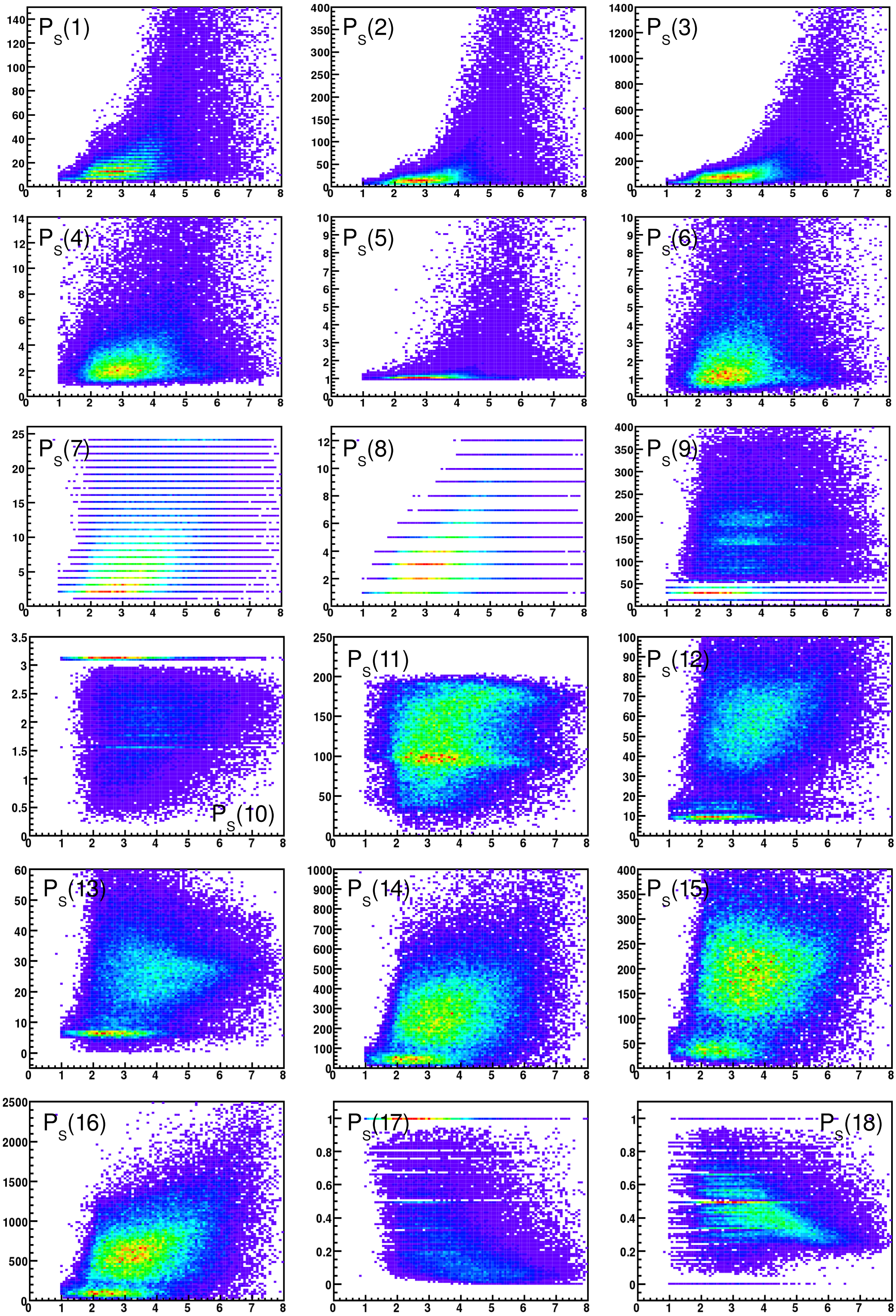


Figure A.2: Distribution of input parameters of Simple estimator from A_1^{up} over Monte Carlo energy [GeV] of event

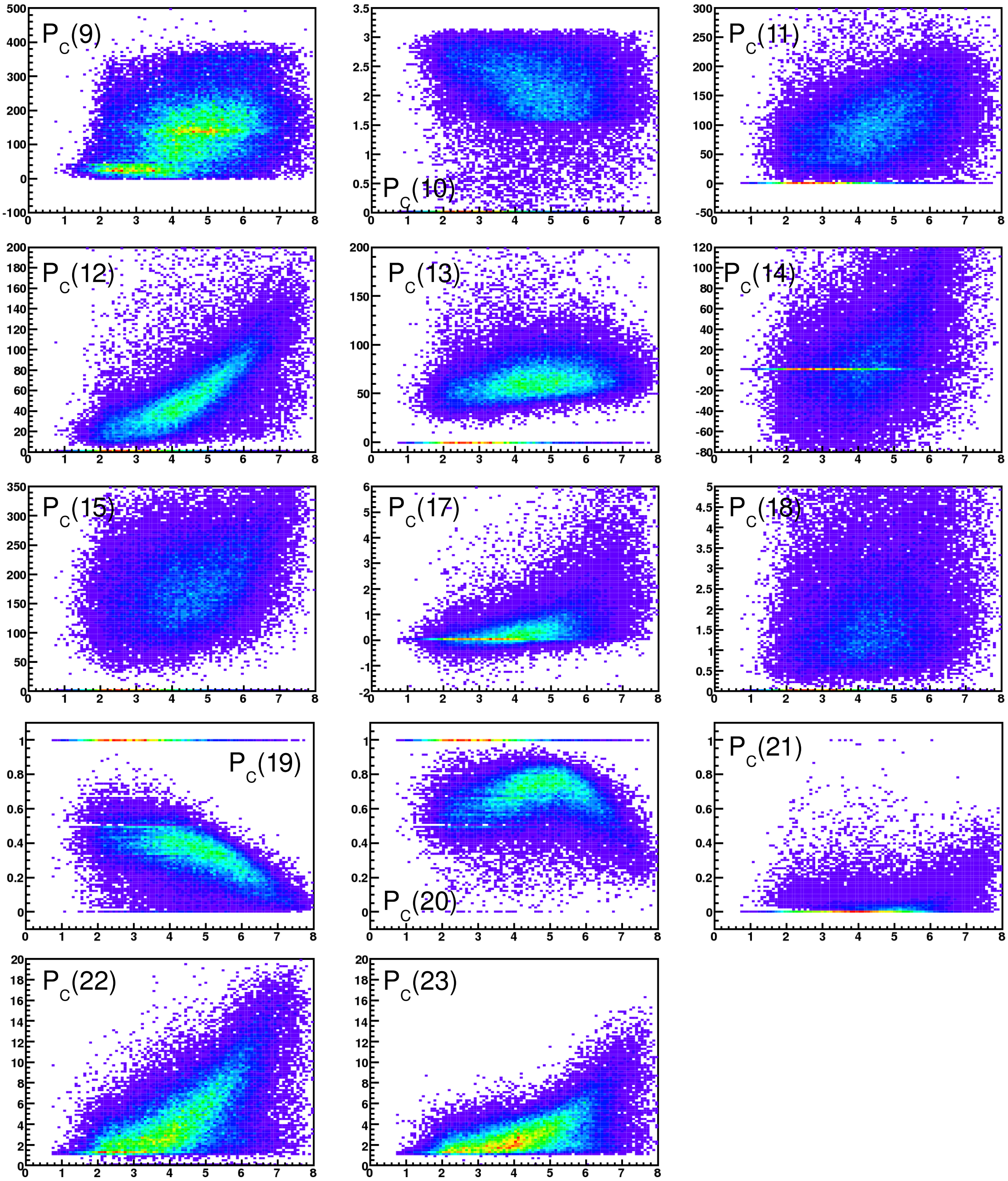


Figure A.3: Distribution of input parameters of Cherenkov Estimator from $A_1^{\mathrm{up}}1$ over Monte Carlo energy [GeV] of event

parameter	α	alternative transformation
P_1^S	12.8	
P_2^S	14.7	
P_3^S	50	
P_4^S	1.96	
P_5^S	0.95	
P_6^S	2.02	
P_7^S	-	$x_7^{scaled} = P_7^S/25$
P_8^S	-	$x_8^{scaled} = P_8^S/12$
P_9^S	52.6	
P_{10}^S	-	$x_{10}^{scaled} = \cos(P_{10}^S)$
P_{11}^S	48.3	
P_{12}^S	17	
P_{13}^S	7.8	
P_{14}^S	94.5	
P_{15}^S	61	
P_{16}^S	208	
P_{17}^S	-	no transformation needed
P_{18}^S	-	no transformation needed

(a) Simple Estimator

parameter	α	alternative transformation
P_1^C	12.8	
P_2^C	14.7	
P_3^C	50	
P_4^C	1.96	
P_5^C	0.95	
P_6^C	2.02	
P_7^C	-	$x_7^{scaled} = 2(P_7^S/25 - 0.5)$
P_8^C	-	$x_8^{scaled} = 2(P_8^S/12 - 0.5)$
P_9^C	50	
P_{10}^C	-	$x_{10}^{scaled} = \cos(P_{10}^S)$
P_{11}^C	41.4	
P_{12}^C	15.9	
P_{13}^C	30.2	
P_{14}^C	15	
P_{15}^C	74.4	
P_{16}^C	209	
P_{17}^C	0.5	
P_{18}^C	0.79	
P_{19}^C	-	$x_{19}^{scaled} = 2P_{19}^S - 1$
P_{20}^C	-	$x_{20}^{scaled} = 2P_{20}^S - 1$
P_{21}^C	-	$x_{21}^{scaled} = 2P_{21}^S - 1$
P_{22}^C	1.79	
P_{23}^C	1.35	

(b) Cherenkov Estimator

Figure A.4: Values for parameter transformations for both estimators

Appendix B

Net Parameters and Performance

B.1 Neural Network Configurations

activation function sigmoid symmetric

activation steepness 0.5

Simple Estimator

Training Set	layout	epochs	MSE set A	MSE set V
A up1	18/96/16/1	869	0.00449754	0.00501651
A up2	18/96/12/1	2848	0.00469878	0.00683802
A all1	18/90/12/1	747	0.00455631	0.00500232
A all2	18/96/15/1	503	0.00696346	0.00763673

Cherenkov Estimator

Training Set	layout	epochs	MSE set A	MSE set V
A up1	23/90/12/1	733	0.0036971	0.00422314
A up2	23/90/12/1	424	0.00437404	0.00610819
A all1	23/90/12/1	5100	0.0368185	0.00452613
A all2	23/95/12/1	471	0.00490199	0.00717923

B.2 Data Comparison

B.2.1 Data Selection for Reconstruction

Run number	run duration	number of events	mean rate	missing lines	missing OMs
42959	4:59:33	174588	63.13	9, 10, 12	153
43285	4:54:37	129001	101.59	9, 10, 12	260
44030	3:51:09	182201	62.0	9, 10, 12	150
44112	1:15:37	37466	84.7	6, 9, 10, 12	178
44156	4:08:39	104006	95.79	6, 9, 10, 12	187

B.2.2 Data-Monte Carlo Comparison of Input Parameters

For both estimators the distribution of all input parameters in weighted Monte Carlo simulation was compared to data of run 42959.

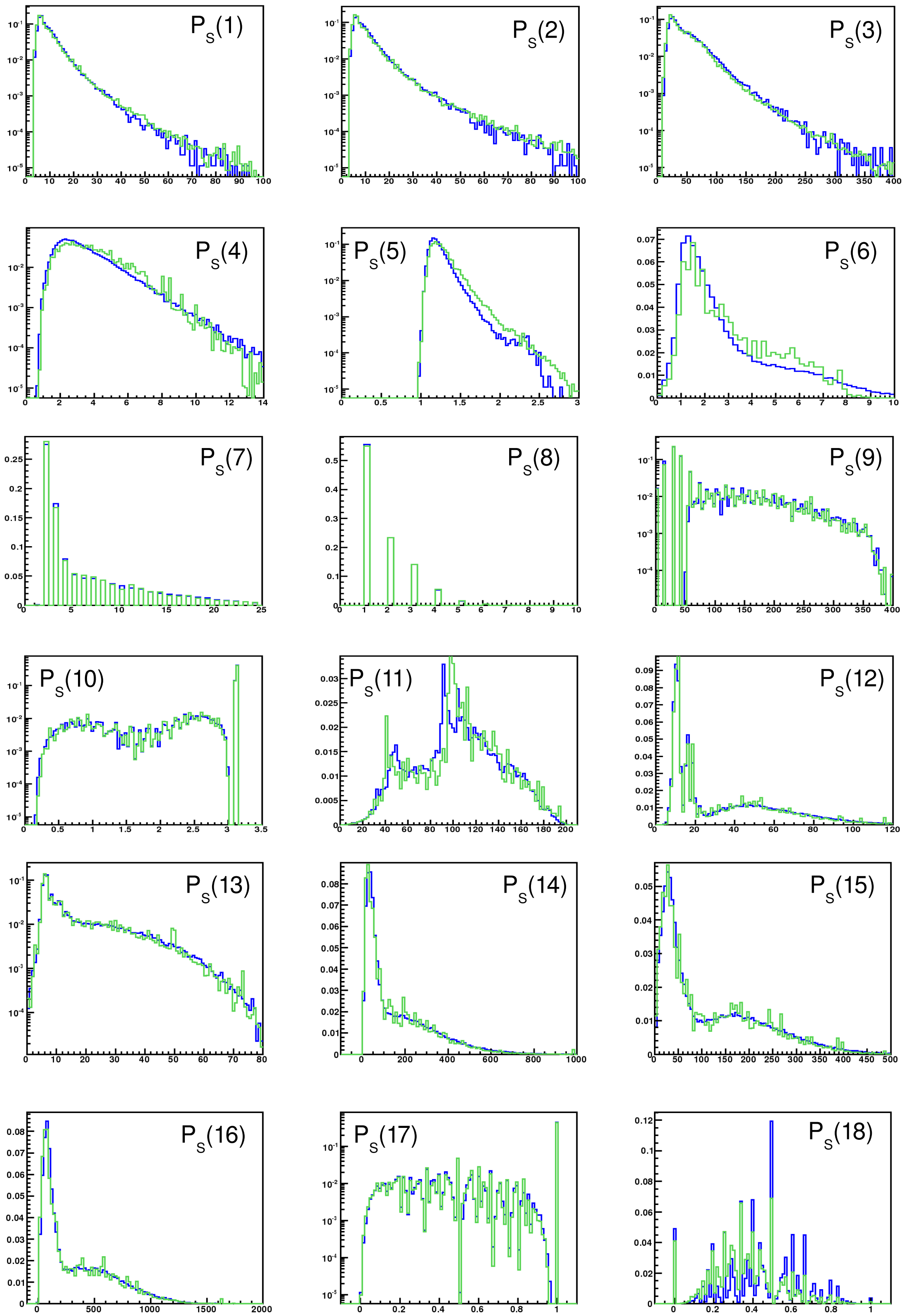


Figure B.1: Monte Carlo simulation (green) compared to data of run 42959 (blue) of Simple Estimator, normalized to 1

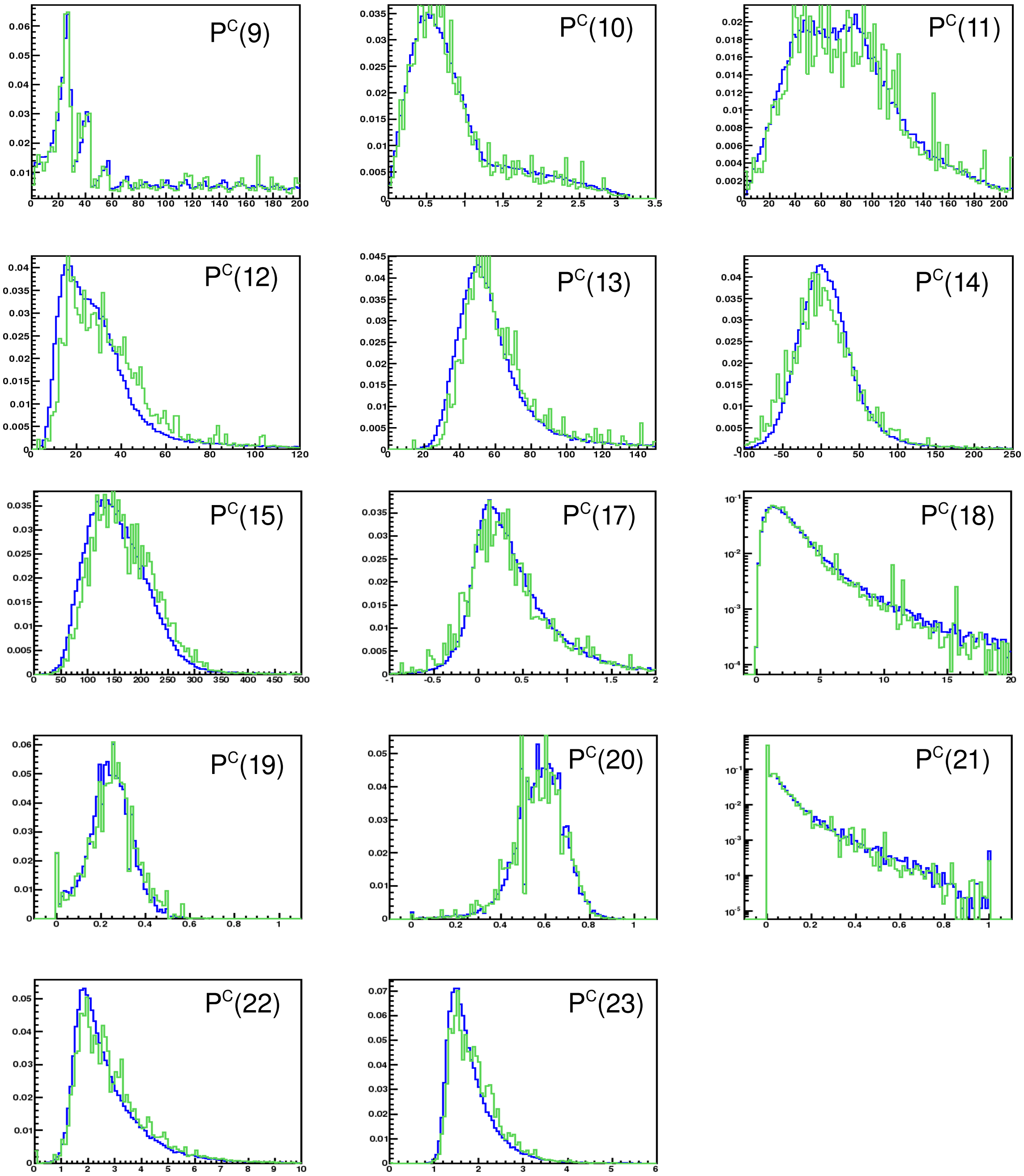


Figure B.2: Monte Carlo simulation (green) compared to data of run 42959 (blue) of Cherenkov Estimator, normalized to 1

Bibliography

[1] David J.L. Baily. *Genhen v5r1: Sofware Documentation*, 2002.

[2] David J.L. Baily. *KM3 v2r1: User Guide*, 2002.

[3] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[4] R. Bruijn. Muon energy reconstruction in ANTARES using neural networks. Master's thesis, Universiteit van Amsterdam, 2002.

[5] R. Bruijn. *The* ANTARES *Neutrino Telescope: Performance Studies and Analysis of First Data*. PhD thesis, Universiteit van Amsterdam, 2008.

[6] The ANTARES Collaboration. A deep sea telescope for high energy neutrinos. *astro-ph/9907432*, 1999.

[7] The ANTARES Collaboration. Technical design report for the ANTARES 0.1 km^2 project, 2001.

[8] The ANTARES collaboration. Transmission of light in deep sea water at the site of the ANTARES neutrino telescope. *ANTARES-site/2004-001*, 2004.

[9] M. de Jong. The ANTARES trigger software. *ANTARES-soft/2005-005*, 2005.

[10] Deutsches Elektronen-Synchrotron. Astropartical physics. homepage.

[11] Thomas K. Gaisser. *Cosmic Rays and Particle Physics*. Cambridge University Press, 1990.

[12] Claus Grupen. *Astroparticle Physics*. Springer, 2005.

[13] B. Hartmann. Reconstruction of neutrino-induced hadronic and electromagnetic showers with the ANTARES experiment, 2006.

[14] A. Heijboer. *Track Reconstruction and Point Source Searches with* ANTARES. PhD thesis, Universiteit van Amsterdam, 2004.

[15] C. Igel and M. Hüsken. Improving the rprop learning algorithm. In ICSC Academic Press, editor, *Proceedings of the Second ICSC International Symposium on Neural Computation*, pages 115 – 121. Institut für Neuroinformatik, Ruhr-Universität Bochum, 2000.

[16] Karl Mannheim John G. Learned. High energy neutrino astronomy. *Annu. Rev. Nucl. Sci. 2000*, 50:679–749, 2000.

[17] A. Margiotta. Monte carlo simulation of atmospheric muons. *ANTARES-soft/2008-005*, 2008.

[18] Steffen Nissen. Implementation of a fast artificial neural network library (fann). Master's thesis, University of Copenhagen (DIKU), 2003.

[19] I.A. Sokalski S.I. Klimushin, E.V. Bugaev. Precise parametrizations of muon energy losses in water. *arXiv:hep-ph/0106010v1*, 2001.

[20] F. Hansen S.J. Kemp, P. Zaradic. An approach for determining relative input parameter importance and significance in artificial neural networks. 2007.

[21] C. Amsler *et al. Review of Particle Physics*. Particle Data Group, 2010.

[22] Todor Stanev Thomas K. Gaisser, Francis Halzen. Particle astrophysics with high energy neutrinos. *MAD/PH/847*, 1994.

Danksagung

Am Ende gilt es denen zu danken, die mir diese Arbeit ermöglicht haben, und die Liste ist lang nach einem Jahr von Fragen, Anregungen und Hilfestellungen. Uli Katz hat die freie Herangehensweise an das Thema ermöglicht, Gisela Anton mit Anregungen neue Möglichkeiten aufgezeigt, die Arbeit zu entwickeln. Danke dafür und für ihre Arbeit als Korrektoren.
Alexander Kappes und Thomas Eberl waren diejenigen, zu denen man immer mit fundamentalen Fragen kommen konnte und fast immer mit einer aufschlussreichen Antwort wieder ging. Sie waren daher für mehr als nur das Korrektur lesen in dieser Arbeit verantwortlich.
Für alle Fragen fand man eigentlich immer einen Ansprechpartner in der ANTARES Softwaregruppe am ECAP, und so könnte man hier die gesammte Gruppe auflisten, deren freundlicher und offener Umgang die Zeit der Diplomarbeit zu einer der schönsten im Studium werden ließ. Besonders erwähnt seien dabei das Zimmer 208, die nicht nur zu meinen neuen Kenntnissen von ANTARES sondern auch des Schafkopf Spiels beigetragen haben. Klaus Geyer konnte man immer nerven, wenn es um neuronale Netze ging und die Zusammenarbeit mit Holger Motz im gegenseitigen Austausch von Netzen und Daten hat die Arbeit um Vieles erleichtert.
Schließlich gilt der grundlegendste Dank meinen Eltern, die mich in allen Dingen unterstützen und unterstützt haben. Durch sie war Physik für mich schon immer ein fester Bestandteil des Lebens.