Contents

1 Neural network training data

Over their relatively brief existance, neural networks have been shown to perform increasingly impressive tasks (e.g. [**openai2019dota**], [**openai2023gpt4**], and many more). However, they learn by example. The performance of a neural network is directly linked to the input data it receives during training. If the training data is not an accurate example of real world information a network later operates on, insight gained from it is at best an approximation, and at worst completely randomly generated data.

As such, it is not a question *if* some neural network architecture can learn to identify an extensive air shower from WCD data, but rather which implementation, fed with which information, does. For this purpose, this chapter explains the procedure with which training data is generated. As stated above, this must occur with a focus on being representative of data actually measured in the SD array. The elected approach to create time traces is modularized. The structure of this chapter reflects this. First, general comments about the characteristics of background data (i.e. the WCD detector response in the absence of an extensive air shower) are made in **??**. Next, the process to extract signal originating from CRs is detailed in **??**. Lastly, building the time trace from the aforementioned modules and drawing samples from it for a neural network to train on is done in **??** and **??**.

1.1 Background dataset

While a flux of partices causes elevated ADC levels in both the HG and LG channels of a WCD PMT during a shower event, the lack of such a phenomenon does not imply the readout information is uniformly flat. Instead, it hovers around the channels' baseline (c.f. **??**) with occasional spikes upwards due to low-energy particles impinging on the detector. Coupled with electronic noise from the many digital components in the station electronics, the **U**pgraded **U**nified **B**oard (UUB), this constitutes the data that is collected inbetween air shower events.

1.1.1 Accidental muons

Most low-energy background particles present in the detector are muons. These are produced in the upper atmosphere during cascading processes analog to **??**. Due to the low primary energy the electromagnetic component of the shower is thermalized before it reaches surface level. The muonic component by itself does not contain enough information to enable an accurate reconstruction of primary energy and origin. This fact, coupled with the high flux of events at lower energies ($\Phi|_{E=100\,\mathrm{GeV}} \approx 1\,\mathrm{m}^{-1}\,\mathrm{s}^{-1}$

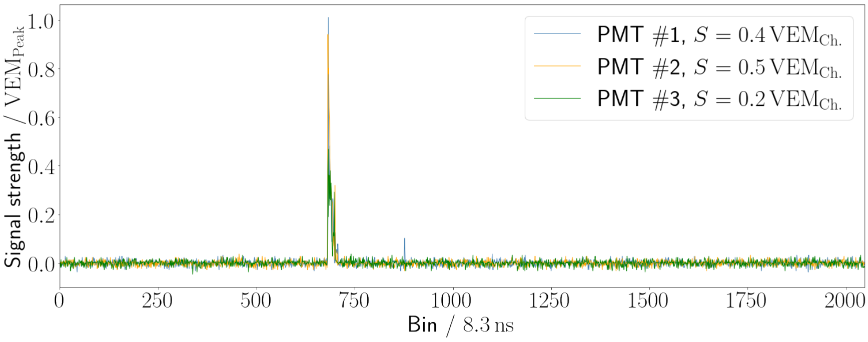


Figure 1.1: The simulated time trace from a single muon. The maximum peak of the time trace is equal to 1 $\mathrm{VEM_{Peak}}$. The integral over each PMT, S, sums to $\approx 1\,\mathrm{VEM_{Ch.}}$.

[**boezio2000measurement**]) make these events unsuitable for analysis. Stray muons, even though they originate from extensive air showers, must consequently be considered background events.

The rate at which such particles traverse a WCD tank is $f_{\mathrm{Acc.}} \approx 4.665\,\mathrm{kHz}$ [**DavidBackgroundSim**]. Their arrival time is Poisson-distributed. This implies that generally, one in 14 time traces contains signal from a low-energy background event. Coincidences of two accidental muons occur on a sub-percent level. Any higher order of coincidences is likely originating from a single air shower process. The typical signal recorded by the surface detector from a single muon is presented in **??**.

A library of background traces of this type was provided by David Schmidt [**DavidBackgroundSim**]. However, only the largest response of the three WCD PMTs is available for this library. Due to the lack of information one is either forced to assume the response to a low-energy background particle is the same across all PMTs, or neglect the response of the two remaining PMTs altogether upon injectiong a background muon into a signal trace (c.f. **??**). In both cases, neural networks are provided an easily detectable pattern to discern such particles from "real" shower signal. As a result, it should be refrained from training AI triggers on this dataset.

1.1.2 Electronic noise

Electronic noise is the umbrella term given to the distortions that some signal is subject to during digital readout. Such noise can have many different origins. An illustrative example is given by the **L**aser **I**nterferometer **G**ravitational wave **O**bservatory, which excludes the 60 Hz band and its' harmonics from analysis. This is owed to the fact that the DC frequency standard in the United States introduces systematic uncertainties in the detector [**martynov2016sensitivity**]. In the electronics of Pierre Augers' SD array, electronic noise is assumed to be Gaussian. That is to say that the ADC values of a time trace that was measured while no particle produced signal in the tank are normally distributed around the baseline. The standard deviation can be estimated from monitoring data, as is shown in **??**.

1.1.3 Random traces

Both above mentioned phenomena can be simulated, and the simulation results used as background training data for the neural networks discussed in the next chapter. A more accurate method however, and the approach elected for this work is to utilize directly measured data from the field. Thanks to the work of David Nitz, there exist collections of so called random-traces[1] that were gathered by forcing DAQ readout via a manually set trigger.

In particular, two datasets of UUB random-traces have been created until now. They were taken from 13th-18th March 2022, and 14th-18th November 2022 respectively. The first set contains data a total of sixteen million time traces distributed over four different SD stations. For reasons explained in **??**, only data from the first set is used in the analysis presented in this work.

Characteristics

Contrary to the naming of the random trigger, it occurs at a deterministic time. More accurately, the process of measuring random-traces is as follows; A single time trace ($2048 \cdot 8.333\,\text{ns} = 17.07\,\mu\text{s}$) is written to the local station buffer every 10 ms. Once the buffer has accumulated enough data, it is written to a USB storage device. Because of a bottleneck in the last step, the process takes about 22 h per station [**nitzCorrespondence**].

It is thus not the trigger that is unpredictable, but the data measured by each trigger. Due to the read/write process being indepentant of the measured data (as opposed to the algorithms in **??**) the latter must be considered to be essentially random. For the most part, random-traces are assumed to consist solely of electronic noise. However, signal of cosmic origin - be it accidental muons or extensive air showers - will appear in the data at a rate at least consistent with **??**.

A crude analysis of the type of noise in the random-traces can be made by examining the spectral density of the dataset, shown in **??**. Harmonic modulations visible in both spectra might originate from an offset between last and first bin of the random-traces. If this offset is nonzero, the periodic extension of $f(x)$ exerts a step-function-like behaviour. The Fourier transform consequently reflects this [**burrows1990fourier**]. Still, several features of $|\hat{f}(\xi)|^2$, espically present at 10 MHz, imply the presence of systematic noise in the UUB. Nevertheless, the large scale form of the spectral density is compatible with at least two noise models, that cannot be distinguished based on the data at hand:

- $|\mathbf{f}(\xi)|^2 \propto \exp\left(\frac{(\xi-\mu)^2}{2\sigma^2}\right)$. The spectral density is Gaussian. This implies the noise is Gaussian distributed as well, confirming the assumption in **??**.

[1] to avoid possible confusion between this dataset and a *random* trace in the statistical sense, the traces recorded by David Nitz are referred to as random-trace, with a hypen.

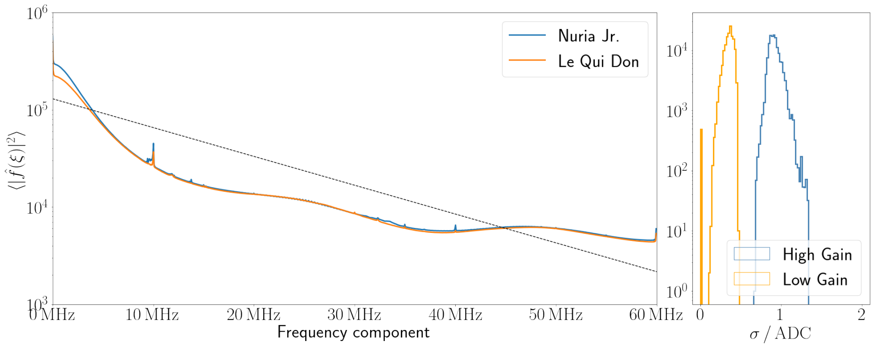


Figure 1.2: (*Left*) The random-trace spectral density for two stations. Plotted with a dashed black line is reference attenuation curve falling at −6 dB/Oct. The spike at 10 MHz is of unknown origin and represents systematic noise in the UUB electronics. (*Right*) Example variance of all UUB stations in the surface detector array. The data shown in this plot was recorded on November 15th 2022.

- $|\mathbf{f}(\xi)|^2 \propto \exp(-m\xi + b)$. The spectral density is proportional to ξ^{-n} for some power n. The case $n = 2$ (−6 dB/Oct attenuation) seems to describe the observations well, hinting that the generating function could be Brownian.

Calibration

The random-trace files contain raw measurement data in units of ADC for the HG and LG channel of the three WCD PMTs. In a first step to standardize this information, the baseline is substracted from each FADC bin. This is done via the baseline finding algorithm described in **??** and [**tobiasBaseline**, **tobiasBaselineUUB**]. Note that this approach differs from the baseline finding algorithm that runs on each station (c.f. **??**). However, the difference is negligible ($<< 1$ ADC) for traces that do not contain any signal, which is the case for the vast majority of the dataset.

Next, the baseline-substracted time traces are converted from units of ADC to $\mathrm{VEM}_{\mathrm{Peak}}$. This conversion is not straight forward, as it requires knowledge of I_{VEM} at the time of data taking. Each station estimates this value in periodic time intervals in the context of monitoring diagnostics.

For the second dataset of random-traces (taken from 14th-18th November 2022) a UNIX timestamp packaged with each time trace may be related to monitoring data. This reveals that no information regarding I_{VEM} was forwarded to CDAS for any station while it recorded random-traces. As a result, the entire dataset is unfortunately rendered useless for this work.

For the first collection of random-traces, monitoring data is available, but there exists no timing information for the individual traces. Only the date of the measurement is known. The elected procedure to evaluate data as accurately as possible is thus to calculate the day average of I_{VEM} and Q_{VEM} and take this as the best (first) estimate for each trace. As can be seen in **??**, this eliminates half of the remaining dataset, as two of the four stations show a large variance in I_{VEM}. The day average in these particular

Table 1.1: Calibration constant I_{VEM} for random-traces.

PMT #	Nuria	Le Qui Don
1	159.34 ADC	145.79 ADC
2	161.37 ADC	144.63 ADC
3	149.91 ADC	154.62 ADC

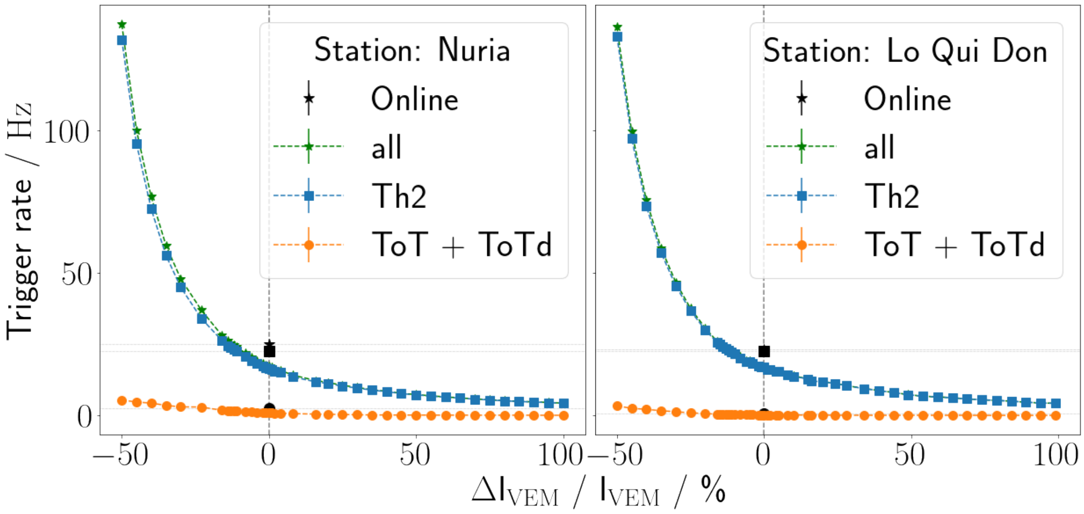


Figure 1.3: The online reported T2 trigger rate (black) does not match the calculated trigger rate. Only a decrease ΔI_{VEM} by 1/10th of the original I_{VEM} gives a close approximation of the observed rate when manually calculating trigger frequencies.

cases is not a good estimator for calibration purposes. For this reason, only random traces from the stations Nuria and Le Qui Don, measured in the March dataset are used for analysis purposes, as their $I_{\text{VEM, Rand.}}$ is stable.

As a crosscheck to verify the goodness of the approximation, the T2 trigger rate as reported by the calibration process is related to the trigger rate obtained by direct calculation over all (calibrated) random-traces. By extension, this also serves as a unit test for the classical triggers as they will be implemented in **??**. The results of this analysis are shown in **??**. As is clear from the plot, the rates calculated from the two different approaches are not in accordance. This indicates systematic errors in the calibration (a wrong implementation of trigger algorithms is disfavoured from the discussion in **??**). The errors are consistent across both considered stations. It is found that calculated trigger frequencies are ≈ 25% lower than what is taken from monitoring. It is unclear why this discrepancy occurs, as it implies that the stations do not use the same threshold values for triggering as they report.

In any case, $I_{\text{VEM, Rand.}}$ must be adjusted to reflect this. A 25% increase in trigger rate is relatable to a ≈ 10% decrease in I_{VEM} . The calibration scaling factors for random-traces thus become the values listed in **??**.

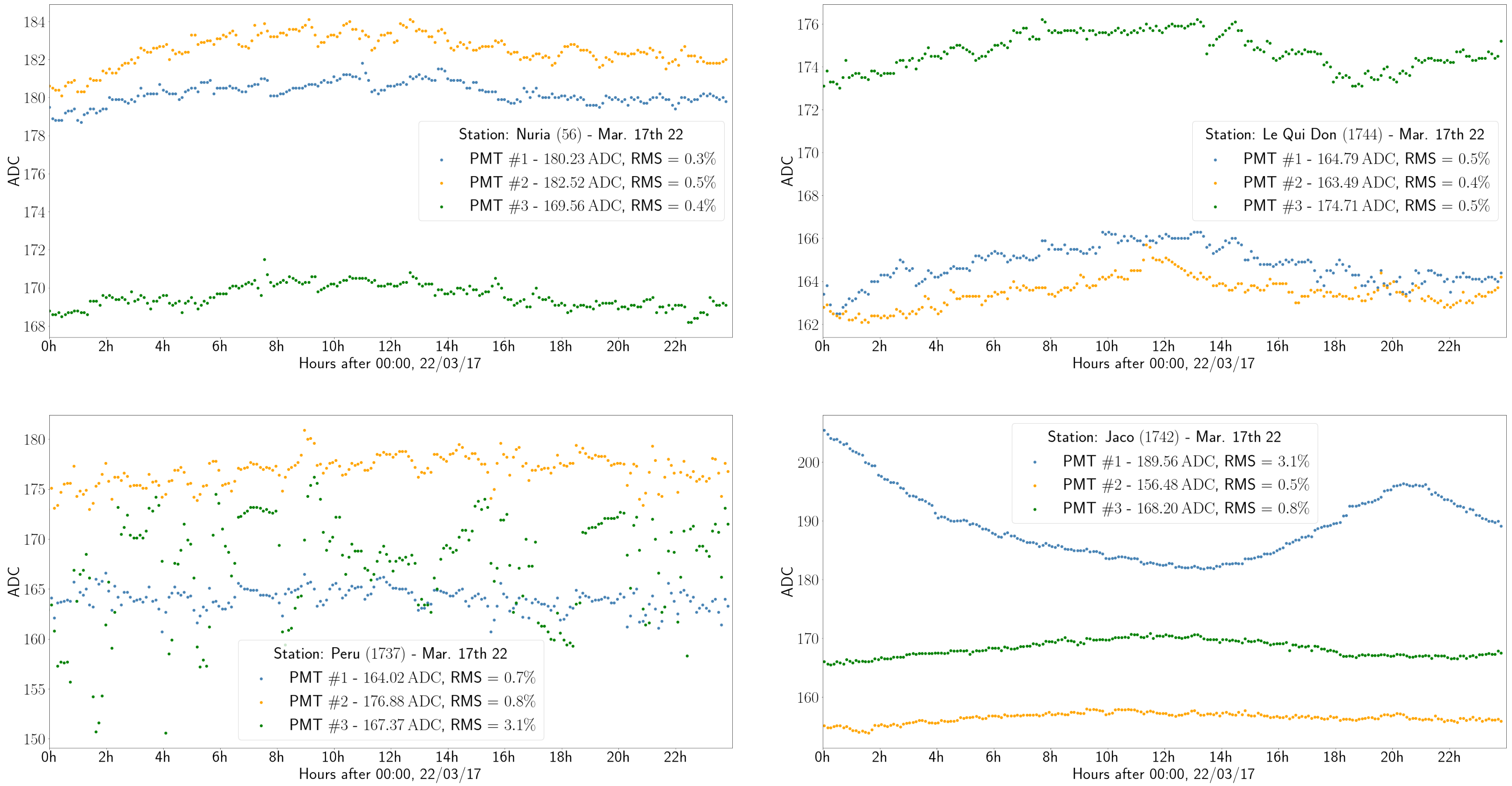


Figure 1.4: Monitoring values for the four stations available in the random-trace dataset measured in March '22. The bottom two stations show a large variance in I_{VEM} for at least one PMT. The two stations in the first row (Nuria, Le Qui Don) are more stable (σ / μ < 1%).

1.2 Signal dataset

In the context of this work, a "signal" (as opposed to background) is *any* detector response caused by extensive air showers. Admittedly, this choice of classification is not ideal, as the particle density far from the shower core grows sparse. Time traces recorded at those locations look very similar to ones raised by accidental muons. In any case, ramifications and possible solutions to this problem are further discussed in the following chapters.

In order to isolate the signal stemming from shower particles only, an $\overline{\text{Off}}\underline{\text{line}}$ simulation using Geant4 is executed on CORSIKA source files [**heck1998corsika**]. These are a total of 40557 simulated showers with a proton primary of energy $16 < \log(E/\text{eV}) < 19.5$. All showers are simulated using the hadronic interaction model QGSJET-II.04 [**ostapchenko2007status**].

In the process, the electronic feedback of the WCD PMTs is evaluated without any disturbance factor (see **??**). That is to say that the time trace obtained from such simulations is identically zero (in units of ADC) at any point in time where no ionizing particles are present in the WCD. An example trace that visualizes this is shown in **??**. Next, the trigger conditions both for individual stations and on the event-level are altered to trigger on everything. This step is needed in order to save all traces to the simulation output, an **A**dvanced **D**ata **S**ummary **T**ree (ADST). If this was not the case, only traces that already satisfy current trigger conditions would be written to disk. A neural network training on such data could therefore at best be as efficient as the current triggers.

The choice of this approach forces some detours in the ADST readout. Instead of extracting the VEM calibrated traces directly, individual component traces, i.e. the PMT signal caused by muons, electrons and photons individually are summed to yield a total ADC trace. Signal stemming from hadrons or other components in the cascade is neglected. This does not impose any errors in the analysis, as the hadronic component espically lays close to the shower core, where the EM- and muonic component of the shower alone should already enable easy detection. Finally, the total trace as calculated above is extracted to a more easily accessible data format alongside shower metadata like primary energy, zenith, but also SPD, and particle count in the station the trace was recorded from.

1.3 Trace building

With the componentwise traces at hand, the total trace as would be recorded in the WCD PMTs for a given event, can be constructed. First, a trace container with default UUB trace length ($2048\,\text{bins} \cdot 8.3\,\text{ns}\,/\,\text{bin} = 17.07\,\mu\text{s}$) and three components per bin (the three WCD PMTs) is initialized with all values equal to zero. Next, an arbitrary random-trace is selected as baseline. Since the FPGAs fundamentally count in the integer domain, the ADC data in the random-trace contains only whole numbers. As is, this wouldn't correctly model rollover when adding integer random-traces to floating point simulation information. There two ADC signals by themself might not exceed the threshold to cross to the next higher value, but the sum would; $\lfloor 0.7\,\mathrm{ADC} \rfloor + \lfloor 0.4\,\mathrm{ADC} \rfloor = 0\,\mathrm{ADC} \neq 1\,\mathrm{ADC} = \lfloor 0.7\,\mathrm{ADC} + 0.4\,\mathrm{ADC} \rfloor$. To account for this, uniformly distributed random numbers from 0 (inclusive) to 1 (exclusive)are added to the random-trace.

Furthermore the $I_{\mathrm{VEM,\,Rand.}}$ from random-traces (c.f. **??**) will in general be different from I_{VEM} simulated by $\overline{\mathrm{Off}}\underline{\mathrm{line}}$ ($I_{\mathrm{VEM,\,Off.}} = 215.781\,\mathrm{ADC}$ compare [**offlineSource**]). Thus the random-trace must be scaled by a factor $\frac{I_{\mathrm{VEM,\,Off.}}}{I_{\mathrm{VEM,\,Rand.}}}$ before being added to the container.

If desired, accidental muons can be added to the trace container as well. This is done either by directly specifying a number of random injections, or throwing the dice according to the injection frequency specified in **??**. If the number of accidental muons is nonzero, a sample of random background traces from [**DavidBackgroundSim**] is drawn and each sample added to every PMT at a random uniform position somewhere in the trace. **??** shows an example where five muons are injected into the trace.

Last, the actual shower signal is added to the trace container. In principle, it can be added at any random position, similar to the random injections. However, for continuity reasons and ease of comparing to plots generated with other software, the latch bin for signal insertion is hardcoded to be the same as in $\overline{\mathrm{Off}}\underline{\mathrm{line}}$, bin 660. Since otherwise the data is in the correct ADC format, no further manipulation of the data is necessary.

The trace container now holds all necessary components together, but remains in units of ADC. To convert to $\mathrm{VEM}_{\mathrm{Peak}}$, each bin in each PMT trace is floored to mimic the FPGA digital counting, and divided by the appropriate scaling factor I_{VEM} . If traces need to be UB compatible the trace must be filtered and downsampled in an intermediate step. This influences the scaling factor $I_{\mathrm{VEM,\,compat.}}$ in a major way (compare **??**). If the so-called full-bandwidth, UUB time trace is analyzed, the appropriate factor becomes $I_{\mathrm{VEM,\,Off.}}$.

1.3.1 Filtering and downsampling

Altough the triggers discussed in the next chapter are meant to function completely autonomously in the SD field, their implementation requires some prior knowledge of the signal one desires to detect. For their use in the Auger observatory, several hyperparameters such as the thresholds of the Th-Trigger, or the window size of the ToT-trigger have been determined in studies ([**bertou2006calibration**], [**triggerSettings**], [**ToTtriggerSetting**]).

These studies were conducted using the predecessor, the **U**nified **B**oard (UB), of the hardware that is being installed during the AugerPrime upgrade of the observatory. Most importantly, the UUB has a sampling rate that is three times larger (120 MHz) than that of UB electronics (40 MHz). Not only does this raise the number of bins in a standard time trace from 682 to $2^{11} = 2048$, but also drastically reduces the efficiency

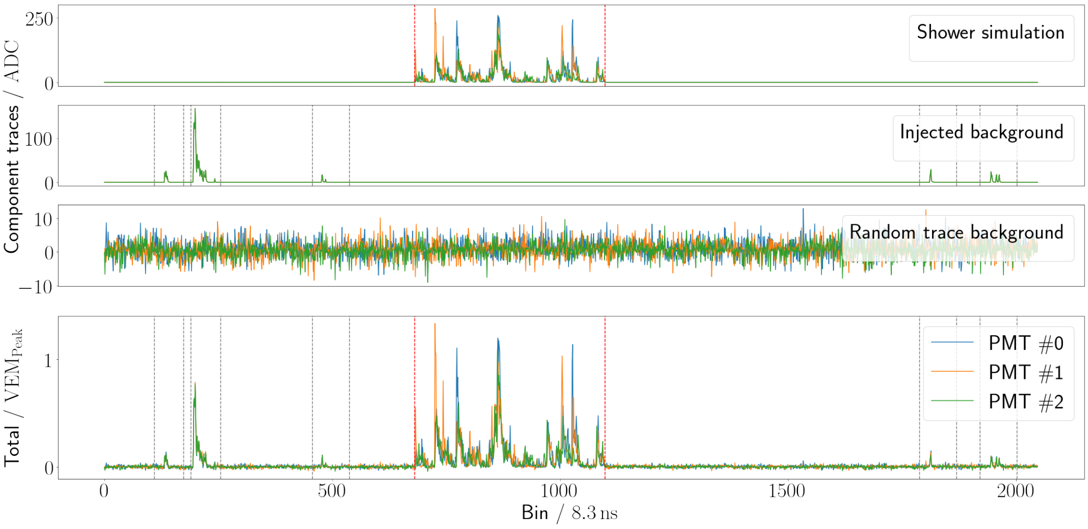


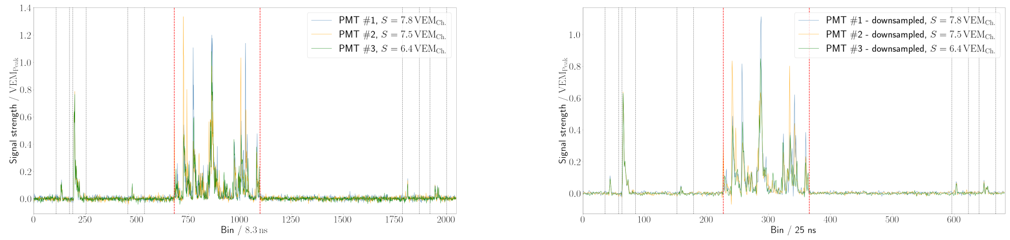
Figure 1.5: The individual component traces (in units of ADC, top three plots) make up the eventual VEM trace (in VEM$_{\text{Peak}}$, bottom plot). The dashed red (gray) lines signify where in the time trace the shower signal (injected muon signal) is located.

(in particular for ToT-like triggers) of the above discussed algorithms. Whereas a new bin is measured every 25 ns in a UB station, the triggers would receive a new input every ≈ 8.3 ns in a UUB setting.

The modus operandi elected by the Pierre Auger collaboration to circumvent this problem is to emulate UB electronics using the UUB electronics. This means that measured FADC bins are to be filtered and downsampled before any trigger runs over them. Software implementations by which this is achieved are listed in **??**. The effect the filtering and downsampling has on measured data is visualized in **??**.

While the features of the time trace largely remain intact, the absolute signal strength decreases due to a smearing effect imposed by the filtering. Overall, this amounts to a 30% difference in amplitude between UUB full-bandwidth traces and their filtered and downsampled counterpart. Since both measurements are derived from the same signal in the WCD though, this implies that I_{VEM} must be adjusted by 30% as well, if traces are to be downsampled. This results in a compatibility scaling factor $I_{\text{VEM, compat.}} = 163.235$ ADC [**OfflineSource**].

Recent contributions within the Auger collaboration ([**nitzTriggers**, **quentinComparison**]), and to some extent also this work have shown that issues arise in the comparison of **L**ateral **T**rigger **P**robabilities (LTPs) that are run in this compatibility mode. Namely, the UUB trigger efficiency (where full-bandwidth traces are filtered and downsampled) is lower than that of UB stations. This implies that the filtering and downsampling algorithms in **??** either make imprecise assumptions about the station electronics, or $I_{\text{VEM, compat.}}$ or the trigger thresholds themselves need to be adjusted further. This fact has to be kept in mind when discussing results and comparing lateral trigger probabilites from classical triggers and neural networks.



(a) UUB time trace

(b) UB time trace

Figure 1.6: (a) A simulated signal as it would appear to UUB electronics. The ionizing particles originating in the extensive air shower hit the tank around bin 660 (≈ 5.5 µs). **(b)** The same signal but filtered and downsampled to emulate UB electronics.

1.3.2 Sliding window analysis & Labelling

As will become apparent in **??**, three of the four trigger algorithms operate by examining only a window of measurement info, rather than evaluate the whole time trace as it is constructed in **??**. In a similar fashion, it is reasonable to assume neural networks do not need to receive the $3\,(\mathrm{PMT}) \cdot 2048\,(\mathrm{UUB\ bins}) = 6144$ input values from the entire trace to make an informed choice of whether or not a given signal stems from an extensive air shower.

For this reason, samples from the time trace are drawn via a sliding window analysis. A number of n_bins are extracted from the trace, to be analyzed by some classification algorithm. In order to not select the same information repeatedly, the window is moved by n_skip bins forward and the process can begin anew. Unless explicitly specified otherwise, the hyperparameters in this sliding window analysis are set as

$$n_\mathrm{bins} = 120, \qquad n_\mathrm{skip} = 10. \tag{1.1}$$

Whether or not a specific window contains signal from an extensive air shower - which is important for labelling data in the context of neural network training - is a simple exercise. The modular approach in **??** allows to simply check for nonzero bins in the shower signal component of the trace. In practice, upon creating a new combined trace, the first and last positive bin in the shower component are identified. This is e.g. visualized with dashed red lines in **??**. If any overlap - even just a single bin - exists between the sliding window and this signal region, the extracted window is consequently labelled as signal. If this is not the case, the window is labelled as background.

Of course, quality cuts can be applied, and a decision to count a given trace window can be made individually. This is further discussed in **??**.