1 The Hypotheses

Before performing a statistical test, is is important to formulate the exact hypotheses, which one wants to consider.

For the log-likelihood ratio tests that are considered here, a null-hypothesis and an alternative hypothesis is required. As null-hypothesis a data set with no signal events, i.e. with only background events, is usuallly considered. As alternative hypothesis a data set including signal events from a given signal (source) hypothesis is usuallly considered.

As general source hypothesis one could consider a source flux, Φ_{S}, for instance a neutrino or gamma-ray particle flux from a source. Such a general flux can be parameterized as

$$\Phi_{\mathrm{S}}(\alpha, \delta, E, t|\vec{x}_{\mathrm{s}}, \vec{p}_{\mathrm{s}}), \tag{1}$$

which is a function of the celestial coordinates right-ascension, α, and declination, δ, as well as the energy and time of the signal particle, given the source to be at location $\vec{x}_{\mathrm{s}}$ with source parameters $\vec{p}_{\mathrm{s}}$. At this stage $\vec{x}_{\mathrm{s}}$ doesn't have to be a single coordinate, but could also describe an extended source, e.g. the galactic plane.

This flux is a differential flux, thus can be written as

$$\frac{\mathrm{d}\Phi_{\mathrm{S}}(\alpha, \delta, E, t|\vec{x}_{\mathrm{s}}, \vec{p}_{\mathrm{s}})}{\mathrm{d}A\mathrm{d}\Omega\mathrm{d}E\mathrm{d}t}, \tag{2}$$

where A and Ω denotes area and solid-angle, respectively.

1.1 Flux Models

SkyLLH implements several flux model classes for Φ_{S}. The most generic flux model class is `FluxModel`, which provides the abstract base class for all flux models given by the differential flux (2).

1.2 Factorized Flux Models

The traditional point-like source searches in IceCube use a factorized flux model as signal hypothesis, where the spatial, energy, and time profiles of the source factorize, i.e. are independent of each other, and a constant Φ_0 provides the flux normalization. It should be noticed that such a flux model is already an assumption on the source. Such a factorized flux model is of the form

$$\Phi_{\mathrm{S}}(\alpha, \delta, E, t|\vec{x}_{\mathrm{s}}, \vec{p}_{\mathrm{s}}) = \Phi_0 \Psi_{\mathrm{S}}(\alpha, \delta|\vec{x}_{\mathrm{s}}, \vec{p}_{\mathrm{s}}) \epsilon_{\mathrm{S}}(E|\vec{p}_{\mathrm{s}}) T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s}}), \tag{3}$$

where Φ_0 is the flux normalization carrying the differential flux units, and Ψ_{S}, ϵ_{S}, and T_{S} are the spatial, energy, and time profiles of the source, respectively.

In SkyLLH the Python class `FactorizedFluxModel`, derived from class `FluxModel`, provides a base class for this mathematical class of flux models.

1.3 Point-like Source Factorized Flux Models

In IceCube the traditional searches for astro-physical neutrinos were searches for neutrino emission from point-like objects in the sky, where a factorized flux model has been used as source hypothesis. Thus, the spatial profile is given by two delta functions:

$$\Psi_{\mathrm{S}}(\alpha,\delta|\vec{x}_{\mathrm{s}}) = \delta(\alpha-\alpha_{\mathrm{s}})\delta(\delta-\delta_{\mathrm{s}}), \tag{4}$$

where the source location $\vec{x}_{\mathrm{s}}$ is given by a single point in the sky in equatorial coordinates: $\vec{x}_{\mathrm{s}} = (\alpha_{\mathrm{s}}, \delta_{\mathrm{s}})$. Hence, the flux model can be formulated as:

$$\Phi_{\mathrm{S}}(\alpha,\delta,E,t|\vec{x}_{\mathrm{s}},\vec{p}_{\mathrm{s}}) = \Phi_0\delta(\alpha-\alpha_{\mathrm{s}})\delta(\delta-\delta_{\mathrm{s}})\epsilon_{\mathrm{S}}(E|\vec{p}_{\mathrm{s}})T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s}}). \tag{5}$$

For such point-like source flux models SkyLLH provides the `PointlikeSourceFFM` class, which is derived from the `FactorizedFluxModel` class.

The emission energy-profile of the source flux might be assumed to be a power-law:

$$\epsilon_{\mathrm{S}}(E|\vec{p}_{\mathrm{s}}) \equiv \epsilon_{\mathrm{S}}(E|\gamma) = \left(\frac{E}{100\mathrm{TeV}}\right)^{-\gamma}, \tag{6}$$

where γ is the spectral index of the source.

As emission time-profile of the source, $T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s}})$, different functional forms could be imagined. A steady, i.e. time independent, source emission time profile would be

$$T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s}}) \equiv 1. \tag{7}$$

In cases of a time variable source, $T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s}})$ could be a box profile of length T_{W} with the box's middle time position, T_0,

$$T_{\mathrm{S}}(t|T_0,T_{\mathrm{W}}) = \begin{cases} 1 & \forall t \in [T_0 - T_{\mathrm{W}}/2; T_0 + T_{\mathrm{W}}/2] \\ 0 & \text{otherwise} \end{cases}, \tag{8}$$

or a Gaussian shaped time profile centered at T_0 with a time width of σ_T,

$$T_{\mathrm{S}}(t|T_0,\sigma_T) \equiv \exp\left(-\frac{(t-T_0)^2}{2\sigma_T^2}\right). \tag{9}$$

For efficiency reasons, the Gaussian shape source time profile could also be truncated at a certain distance from T_0, *e.g.* at $\pm\sigma_T$:

$$T_{\mathrm{S}}(t|T_0,\sigma_T) \equiv \begin{cases} \exp\left(-\frac{(t-T_0)^2}{2\sigma_T^2}\right) & |t-T_0| \leq \sigma_T \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

2 The Likelihood Formalism

This section describes the mathematical likelihood formalism used in SkyLLH. First it introduces the log-likelihood approach, second the likelihood ratio test and the used test statistic and then describes the used optimizations.

2.1 The Log-Likelihood Approach

SkyLLH implements the two-component likelihood approach with a likelihood function $\mathcal{L}(n_\mathrm{s}, \vec{p}_\mathrm{s} \mid D)$ of the form

$$\mathcal{L}(n_\mathrm{s}, \vec{p}_\mathrm{s} \mid D) = \prod_{i=1}^{N} \left[\frac{n_\mathrm{s}}{N} S_i(\vec{p}_\mathrm{s}) + (1 - \frac{n_\mathrm{s}}{N}) B_i \right], \tag{11}$$

where n_s is the number of signal events, hence, $(N - n_\mathrm{s})$ the number of background events in the data sample D of N total events. The set of source model parameters is denoted as $\vec{p}_\mathrm{s}$. For a point-like source model, the source model parameters include the source position $\vec{x}_\mathrm{s}$ and the spectral index γ of the source flux. $S_i(\vec{p}_\mathrm{s})$ and B_i is the value of the signal and background PDF for the ith data event, respectively.

The signal and background PDFs must incorporate the detector efficiency (yield), $\mathcal{Y}_i$, which, in general, depends on the celestial direction, the energy, and the observation time of the data event.

For computational stability reasons the logarithm of the likelihood function of equation 11 is used in SkyLLH:

$$\log \mathcal{L}(n_\mathrm{s}, \vec{p}_\mathrm{s} \mid D) = \sum_{i=1}^{N} \log(...) \tag{12}$$

2.2 Likelihood Ratio Test

For estimating the significance of an observation, the likelihood ratio Λ with respect to a null hypothesis of no observation, i.e. equation 11 at $n_\mathrm{s} = 0$ is tested:

$$\log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \log \frac{\mathcal{L}(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathcal{L}(n_\mathrm{s} = 0)} = \sum_{i=1}^{N} \log \left[1 + \frac{n_\mathrm{s}}{N} \left(\frac{S_i(\vec{p}_\mathrm{s})}{B_i} - 1 \right) \right] \tag{13}$$

By defining

$$\mathcal{X}_i(\vec{p}_\mathrm{s}) \equiv \frac{1}{N} \left(\mathcal{R}_i(\vec{p}_\mathrm{s}) - 1 \right), \tag{14}$$

with the signal over background PDF ratio value, $\mathcal{R}_i(\vec{p}_\mathrm{s})$, of the ith event,

$$\mathcal{R}_i(\vec{p}_\mathrm{s}) \equiv \frac{S_i(\vec{p}_\mathrm{s})}{B_i}, \tag{15}$$

this reads as:

$$\log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \sum_{i=1}^{N} \log(1 + n_\mathrm{s} \mathcal{X}_i(\vec{p}_\mathrm{s})). \tag{16}$$

By defining

$$\alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) \equiv n_\mathrm{s} \mathcal{X}_i(\vec{p}_\mathrm{s}) \tag{17}$$

the log-likelihood ratio function of the ith event can be defined as

$$\log \Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) \equiv \log(1 + \alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s})), \tag{18}$$

and the log-likelihood ratio function for all events can be written as

$$\log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \sum_{i=1}^{N} \log \Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s}). \tag{19}$$

In general, the argument of the log-function, Λ_i, might become close to zero, causing a divergence of the log-likelihood ratio function for a particular event. To circumvent this, a Taylor expansion of the log-likelihood ratio function of the ith event can be performed around a predefined threshold value α. The event-based log-likelihood ratio function, $\log \Lambda_i$, is then approximated by a second-order Taylor expansion for events with $\alpha_i \leq \alpha$:

$$\log \Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) \equiv \log(1 + \alpha) + \frac{\alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) - \alpha}{1 + \alpha} - \frac{1}{2}\left(\frac{\alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) - \alpha}{1 + \alpha}\right)^2 \tag{20}$$

By defining

$$\tilde{\alpha}_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) \equiv \frac{\alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) - \alpha}{1 + \alpha}, \tag{21}$$

the Taylor expanded log-likelihood ratio function reads more compactly:

$$\log \Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \log(1 + \alpha) + \tilde{\alpha}_i(n_\mathrm{s}, \vec{p}_\mathrm{s}) - \frac{1}{2}\tilde{\alpha}_i^2(n_\mathrm{s}, \vec{p}_\mathrm{s}). \tag{22}$$

2.3 Test Statistic

Assuming Wilks' theorem, a test statistic, TS, for the two-component log-likelihood ratio test can be formulated using the log-likelihood ratio function, $\log \Lambda$, at its maximum:

$$\mathrm{TS} = 2\mathrm{sgn}(\hat{n}_\mathrm{s}) \log \Lambda(\hat{n}_\mathrm{s}, \vec{\hat{p}}_\mathrm{s}), \tag{23}$$

where $\log \Lambda(\hat{n}_\mathrm{s}, \vec{\hat{p}}_\mathrm{s})$ is the maximum of the log-likelihood ratio function as defined by equation (13), with separation of an over- ($\hat{n}_\mathrm{s} > 0$) and under-fluctuation ($\hat{n}_\mathrm{s} < 0$). In case the assumptions of Wilks' theorem are met within the analysis, the test statistic value distribution will follow a χ^2-distribution with a degree-of-freedom equal to the number of fit parameters.

For the case $\hat{n}_\mathrm{s} = 0$, the log-likelihood ratio function is zero with degenerate source fit parameter values $\vec{\hat{p}}_\mathrm{s}$. When calculating the sensitivity of the analysis, the median test-statistic value for background-only data is required. Hence, $\hat{n}_\mathrm{s}$ is often zero in such cases. Having to deal with a delta-peak of the test-statistic distribution for $\mathrm{TS} = 0$ is cumbersome. In order to resolve the delta-peak, the log-likelihood ratio function can be approximated with a second-order Taylor expansion around $n_\mathrm{s} = 0$, and the apex of that Taylor function defines the value of log-likelihood ratio function for $\hat{n}_\mathrm{s} = 0$. In this case the test-statistic function is given by

$$\mathrm{TS} = -2\frac{\left(\frac{\mathrm{d}\log\Lambda(n_\mathrm{s}=0,\vec{p}_\mathrm{s}=\hat{\vec{p}}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}}\right)^2}{4\frac{\mathrm{d}^2\log\Lambda(n_\mathrm{s}=0,\vec{p}_\mathrm{s}=\hat{\vec{p}}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}^2}}. \tag{24}$$

2.4 Optimizations for Spatially Restricted Sources

For spatially restricted sources, e.g. point-like sources, most of the events in the data sample will be far away from the hypothesised source, hence, the value of the signal PDF, S_i, will be zero or very close to zero. By selecting only the signal-contributing N' events from the sample, the log-likelihood ratio function, $\log\Lambda$, reads

$$\log\Lambda(n_\mathrm{s},\vec{p}_\mathrm{s}) = \sum_{i=1}^{N'}\log\Lambda_i(n_\mathrm{s},\vec{p}_\mathrm{s}) + (N-N')\log(1-\frac{n_\mathrm{s}}{N}), \tag{25}$$

where for $N-N'$ events $\mathcal{R}_i(\vec{p}_\mathrm{s})$ equals zero and hence $\alpha_i(n_\mathrm{s},\vec{p}_\mathrm{s})$ becomes $-n_\mathrm{s}/N$, and $\log\Lambda_i$ equals $\log(1-n_\mathrm{s}/N)$ for all such pure background events.

2.5 Signal PDFs

The likelihood ratio function as given in equation (13) incorporates a signal probability, S_i, for an individual event i, given a signal hypothesis. Without loss of generality it can be expressed as a joint probability of a spatial and an energy p.d.f. at the observation time t_i:

$$S_i(\vec{p}_\mathrm{s}) \equiv S_i(\vec{x}_i, E_i, t_i|\vec{p}_\mathrm{s}) = \mathcal{S}_\mathrm{S}(\vec{x}_i|t_i,\vec{p}_\mathrm{s})\mathcal{E}_\mathrm{S}(E_i|\vec{x}_i,t_i,\vec{p}_\mathrm{s}) \tag{26}$$

It should be noted that at this stage the time-dependences of the spatial, $\mathcal{S}_\mathrm{S}$, and energy, $\mathcal{E}_\mathrm{S}$, p.d.f.s might still be different. In cases where the hypothesised source flux, Φ_S, has a spatial and energy independent time profile, i.e. usually named a "flare", the time-dependence factorizes:

$$S_i(\vec{p}_\mathrm{s}) \equiv S_i(\vec{x}_i, E_i, t_i|\vec{p}_\mathrm{s}) = \mathcal{S}_\mathrm{S}(\vec{x}_i|\vec{p}_\mathrm{s})\mathcal{E}_\mathrm{S}(E_i|\vec{x}_i,\vec{p}_\mathrm{s})\mathcal{T}_\mathrm{S}(t_i|\vec{p}_\mathrm{s}). \tag{27}$$

In general, the signal model parameters $\vec{p}_\mathrm{s}$ can be divided into spatial, energy, and time parameters, i.e. $\vec{p}_\mathrm{s} = (\vec{p}_\mathrm{s,spatial}, \vec{p}_\mathrm{s,energy}, \vec{p}_\mathrm{s,time})$:

$$S_i(\vec{p}_\mathrm{s}) = \mathcal{S}_\mathrm{S}(\vec{x}_i|\vec{p}_\mathrm{s,spatial})\mathcal{E}_\mathrm{S}(E_i|\vec{x}_i,\vec{p}_\mathrm{s,energy})\mathcal{T}_\mathrm{S}(t_i|\vec{p}_\mathrm{s,time}), \tag{28}$$

The spatial component, $\mathcal{S}_\mathrm{S}$, can be identified as the convolution, $(\Psi*\mathrm{PSF})(\alpha,\delta)$[1], of the spatial source extension, $\Psi(\alpha,\delta)$, and the point-spread-function, $\mathrm{PSF}(\alpha,\delta)$, of the detector. For a point-like spatial source extension at position $\vec{x}_\mathrm{s} = (\alpha_\mathrm{s},\delta_\mathrm{s})$,

[1] The 2D convolution on the sky is defined as $(f*g)(\alpha,\delta) = \int_0^{2\pi}\mathrm{d}\alpha'\int_{-\pi}^{\pi}\mathrm{d}\delta' f(\alpha',\delta')g(\alpha-\alpha',\delta-\delta')$.

that is $\Psi(\alpha, \delta) = \delta(\alpha - \alpha_{\mathrm{s}})\delta(\delta - \delta_{\mathrm{s}})$, where $\delta(\cdot)$ is the delta-distribution, this convolution collapses to a single point in the sky. With a 2D Gaussian PSF $\mathcal{S}_{\mathrm{S}}(\vec{x}_i|\vec{p}_{\mathrm{s,spatial}})$ is given as

$$\mathcal{S}_{\mathrm{S}}(\vec{x}_i|\vec{p}_{\mathrm{s,spatial}}) \equiv \mathcal{S}_{\mathrm{S}}(r_i, \sigma_i|\vec{x}_{\mathrm{s}}) = \frac{1}{2\pi\sigma_i^2} \exp\left(-\frac{r_i^2}{2\sigma_i^2}\right), \tag{29}$$

where r_i is the space angle between the source position and the recorded reconstructed event direction. In equatorial coordinates, $\vec{x} = (\alpha, \delta)$, the cosine of r_i is given by

$$\cos(r_i) = \cos(\alpha_{\mathrm{s}} - \alpha_i)\cos(\delta_{\mathrm{s}})\cos(\delta_i) + \sin(\delta_{\mathrm{s}})\sin(\delta_i). \tag{30}$$

The data quantity σ_i describes the angular reconstruction uncertainty of the event, hence the PSF is narrower for well-reconstructed events, and wider for events which have a large reconstruction uncertainty.

The energy signal PDF can be constructed from Monte-Carlo data using the assumed source energy spectrum. When considering a power law as source flux model, the energy source parameters, $\vec{p}_{\mathrm{s,energy}}$, consists of the spectral index γ and possibly an energy cut-off parameter E_{cut}.

The source time PDF, $\mathcal{T}_{\mathrm{S}}(t|\vec{p}_{\mathrm{s,time}})$, is the convolution of the source time profile, $T_{\mathrm{S}}(t|\vec{p}_{\mathrm{s,time}})$, as given in section 1.3, and the detector live-time function,

$$T_{\mathrm{live}}(t) = \begin{cases} 1 & \forall t \in \text{detector on-time window} \\ 0 & \text{otherwise} \end{cases}, \tag{31}$$

where the final result is normalized to unity.

For the source time emission profiles given in section 1.3 we here state their normalization factors, $\tilde{T}_{\mathrm{S}}$.

For a steady source profile over the entire observation (live) time, T_{obs},

$$\tilde{T}_{\mathrm{S}} \equiv \frac{1}{T_{\mathrm{obs}}}, \tag{32}$$

with

$$T_{\mathrm{obs}} = \int T_{\mathrm{live}}(t)\mathrm{d}t, \tag{33}$$

where $T_{\mathrm{live}}(t)$ is the detector live-time function as defined in equation (31).

For a box profile of length T_{W} with the box's middle time position, T_0, as given by equation (8), the normalization factor is given by

$$\tilde{T}_{\mathrm{S}} \equiv \begin{cases} \frac{1}{T_{\mathrm{W}}} & \forall t \in [T_0 - T_{\mathrm{W}}/2; T_0 + T_{\mathrm{W}}/2] \\ 0 & \text{otherwise} \end{cases}. \tag{34}$$

For a Gaussian shaped time profile centered at T_0 with a time width of σ_T, as given by equation (9),

$$\tilde{T}_{\mathrm{S}} \equiv \frac{1}{\sqrt{2\pi}\sigma_T}. \tag{35}$$

2.6 Background PDFs

In analog to the signal PDF, the background PDF can be formulated as

$$B_i \equiv \mathcal{S}_{\mathrm{B}}(\vec{x}_i|t_i)\mathcal{E}_{\mathrm{B}}(E_i|\vec{x}_i, t_i). \tag{36}$$

In cases where the background model has a spatial and energy independent time-profile, e.g. for a constant background flux, the time-dependence factorizes:

$$B_i \equiv \mathcal{S}_{\mathrm{B}}(\vec{x}_i)\mathcal{E}_{\mathrm{B}}(E_i|\vec{x}_i)\mathcal{T}_{\mathrm{B}}(t_i). \tag{37}$$

All the background PDF components can either be determined from the data itself or by using Monte-Carlo simulation.

For a background model of a constant background flux the background time profile, $T_{\mathrm{B}}(t)$, is given by

$$T_{\mathrm{B}}(t) \equiv \frac{1}{T_{\mathrm{obs}}}, \tag{38}$$

where T_{obs} is given by equation (33).

In analog to the signal time PDF, the background time PDF, $\mathcal{T}_{\mathrm{B}}(t)$, is the convolution of the background time profile $T_{\mathrm{B}}(t)$ and the detector live-time function, $T_{\mathrm{live}}(t)$, where the final result is again normalized to unity.

On short time-scales the uniform, time-independent, background hypothesis might not be a reasonable assumption possibly due to a varying event acceptance of the detector. In such cases the background time profile has to be determined from simulation or the experimental data itself.

2.7 Notes on the energy PDFs for Signal & Background

In general, the energy PDFs are detector response dependent. That means they depend on the local direction of the detected events. Hence, the spatial and energy PDFs cannot be factorized entirely in space and energy.

For IceCube the energy resolution mostly depends on the zenith angle, and hence on the declination, of the event. Thus, several energy PDFs are created for a set of (reconstructed) declination bands, both, for signal and background. At the data evaluation, the signal and background PDFs are selected corresponding to the declination band the event's declination is part of. Hence, for IceCube, the signal and background energy PDFs can be formulated as $\mathcal{E}_{\mathrm{S}}(E|\delta, \vec{p}_{\mathrm{s,energy}})$ and $\mathcal{E}_{\mathrm{B}}(E|\delta)$, respectively.

A lengthly discussion has been conducted in the past to clarify whether the true or reconstructed direction of the Monte-Carlo events should be used to generate the several signal energy PDFs. Since, we mainly use experimental data as background estimation it as been concluded to use the reconstructed event direction in order to be consistent in the data evaluation for signal and background PDFs.

2.8 Stacking of Sources

In general a likelihood value can be calculated for a set of K stacked sources in a weighted fashion. In this case the signal PDF expression of equation (28) becomes a bit more complicated due to the relative source weighting. The sources must be weighted according to their signal detection efficiency, $\mathcal{Y}_{\mathrm{s},k}$, and a relative strength weight of the sources, W_k, with $\sum_{k=1}^{K} W_k = 1$. Hence, the combined signal PDF is given as

$$\mathcal{S}_i(\vec{p}_\mathrm{s}) \equiv \frac{\sum_{k=1}^{K} W_k \mathcal{Y}_\mathrm{s}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}) \mathcal{S}_i(\vec{p}_{\mathrm{s}_k})}{\sum_{k=1}^{K} W_k \mathcal{Y}_\mathrm{s}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})}. \tag{39}$$

One should note that this formalism allows for different source properties, e.g. energy spectra, for the various sources.

2.9 Gradients of the Log-Likelihood Ratio

For maximizing the log-likelihood ratio function (equation (19)), or minimizing the negative of it, the minimizer algorithm requires the derivatives of the log-likelihood ratio function w.r.t. the fit parameters, n_s and $\vec{p}_\mathrm{s}$. Hence, here we provide the expressions of these derivatives for the optimized log-likelihood ratio function as given by equation (25).

The derivative w.r.t. n_s is given by

$$\frac{\mathrm{d}\log\Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}} = \sum_{i=1}^{N'} \frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}} - \frac{N - N'}{N - n_\mathrm{s}}. \tag{40}$$

For the numerical stable case, where $\alpha_i > \alpha$, the derivative of the log-likelihood ratio function of the ith event w.r.t. n_s is given by the derivative of equation (18) w.r.t. n_s:

$$\frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}} = \frac{\mathcal{X}_i(\vec{p}_\mathrm{s})}{1 + \alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}. \tag{41}$$

For the numerical unstable case, where $\alpha_i \leq \alpha$, this derivative is given by the derivative of the Taylor expansion of equation (22) w.r.t. n_s:

$$\frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}} = \frac{1}{1+\alpha}\left(1 - \tilde{\alpha}_i(n_\mathrm{s}, \vec{p}_\mathrm{s})\right) \mathcal{X}_i(\vec{p}_\mathrm{s}) \tag{42}$$

For calculating the test-statistic, *c.f.* section 2.3, the second derivative w.r.t. n_s become in handy for the case $n_\mathrm{s} = 0$. Hence, it is provided here as well:

$$\frac{\mathrm{d}^2\log\Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}^2} = \sum_{i=1}^{N'} \frac{\mathrm{d}^2\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}^2} - \frac{N - N'}{(N - n_\mathrm{s})^2} \tag{43}$$

The second derivative w.r.t. n_s of the log-likelihood ratio function for an individual event with $\alpha_i > \alpha$ is given by the derivative of equation (41):

$$\frac{\mathrm{d}^2\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}^2} = -\left(\frac{\mathcal{X}_i(\vec{p}_\mathrm{s})}{1 + \alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}\right)^2 = -\left(\frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}n_\mathrm{s}}\right)^2 \tag{44}$$

For the event case $\alpha_i \leq \alpha$ this second derivative would be a constant due to the second-order nature of the chosen Taylor expansion in that case. At the junction point α the second derivative would not be differentiable. Hence, equation (44) is used as well in this case, with $\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})/\mathrm{d}n_\mathrm{s}$ given by equation (42). This provides a second derivative that is differentiable for all $(1+\alpha_i)$ values, does not diverge for $(1+\alpha_i) \rightarrow 0$, and is closer to the second derivative of $\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})$ for the $\alpha_i > \alpha$ case.

The derivative w.r.t. an individual signal parameter, p_s, is given by

$$\frac{\mathrm{d}\log\Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}} = \sum_{i=1}^{N'} \frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}} \tag{45}$$

Again, one needs to distinguish between the numerical stable ($\alpha_i > \alpha$) and unstable ($\alpha_i \leq \alpha$) case. For the stable case the event-based derivative w.r.t. p_s is given by

$$\frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}} = \frac{n_\mathrm{s}}{1+\alpha_i(n_\mathrm{s}, \vec{p}_\mathrm{s})} \frac{\mathrm{d}\mathcal{X}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}}. \tag{46}$$

For the numerical unstable case this derivative is given by the derivative of the Taylor expansion of equation (22) w.r.t. p_s:

$$\frac{\mathrm{d}\log\Lambda_i(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}} = \frac{n_\mathrm{s}}{1+\alpha} \left(1 - \tilde{\alpha}_i(n_\mathrm{s}, \vec{p}_\mathrm{s})\right) \frac{\mathrm{d}\mathcal{X}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}}. \tag{47}$$

The derivative of $\mathcal{X}_i$ can be calculated using equation 14 and the expressions for the signal and background PDFs as given in equation 28 and 37, respectively. Depending on the type of fit parameter, i.e. spatial, energy, or time, the derivative of the PDF ratio, $\mathcal{R}_i(\vec{p}_\mathrm{s}) = \mathcal{S}_i(\vec{p}_\mathrm{s})/\mathcal{B}_i$, simplifies to the derivative of the respective type of PDF ratio:

$$\frac{\mathrm{d}\mathcal{X}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}} = \frac{1}{N} \frac{\mathrm{d}\mathcal{R}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_\mathrm{s}}, \tag{48}$$

with

$$\mathcal{R}_i(\vec{p}_\mathrm{s}) = \mathcal{R}_{S,i}(\vec{p}_{s,\mathrm{spatial}}) \mathcal{R}_{\mathcal{E},i}(\vec{p}_{s,\mathrm{energy}}) \mathcal{R}_{\mathcal{T},i}(\vec{p}_{s,\mathrm{time}}), \tag{49}$$

and

$$\frac{\mathrm{d}\mathcal{R}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_{\mathrm{s,spatial}}} = \frac{\mathrm{d}\mathcal{R}_{S,i}(\vec{p}_{s,\mathrm{spatial}})}{\mathrm{d}p_{\mathrm{s,spatial}}} \mathcal{R}_{\mathcal{E},i}(\vec{p}_{s,\mathrm{energy}}) \mathcal{R}_{\mathcal{T},i}(\vec{p}_{s,\mathrm{time}}), \tag{50}$$

$$\frac{\mathrm{d}\mathcal{R}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_{\mathrm{s,energy}}} = \mathcal{R}_{S,i}(\vec{p}_{s,\mathrm{spatial}}) \frac{\mathrm{d}\mathcal{R}_{\mathcal{E},i}(\vec{p}_{s,\mathrm{energy}})}{\mathrm{d}p_{\mathrm{s,energy}}} \mathcal{R}_{\mathcal{T},i}(\vec{p}_{s,\mathrm{time}}), \tag{51}$$

$$\frac{\mathrm{d}\mathcal{R}_i(\vec{p}_\mathrm{s})}{\mathrm{d}p_{\mathrm{s,time}}} = \mathcal{R}_{S,i}(\vec{p}_{s,\mathrm{spatial}}) \mathcal{R}_{\mathcal{E},i}(\vec{p}_{s,\mathrm{energy}}) \frac{\mathrm{d}\mathcal{R}_{\mathcal{T},i}(\vec{p}_{s,\mathrm{time}})}{\mathrm{d}p_{\mathrm{s,time}}}. \tag{52}$$

For stacked sources the expression for $\mathcal{R}_i(\vec{p}_\mathrm{s})$ in equation (49) becomes slightly more complicated due to the source strength weighting. With equation (39) and the definitions

$$a_k(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}) = W_k \mathcal{Y}_\mathrm{s}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}), \tag{53}$$

and

$$A(\vec{p}_{\mathrm{s}}) = \sum_{k=1}^{K} a_k(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}), \tag{54}$$

it is given by

$$\mathcal{R}_i(\vec{p}_{\mathrm{s}}) = \frac{\mathcal{S}_i(\vec{p}_{\mathrm{s}})}{\mathcal{B}_i} = \frac{1}{A(\vec{p}_{\mathrm{s}})} \sum_{k=1}^{K} a_k(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}) \frac{\mathcal{S}_i(\vec{p}_{\mathrm{s}_k})}{\mathcal{B}_i}. \tag{55}$$

The signal over background ratio $\mathcal{S}_i(\vec{p}_{\mathrm{s}_k})/\mathcal{B}_i \equiv \mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k})$ for the single source k is then given by equation (49).

Using the same set of source fit parameters $\vec{p}_{\mathrm{s}}$ for all sources, i.e. called global source fit parameters, the derivative of $\mathcal{R}_i(\vec{p}_{\mathrm{s}})$ for all stacked sources w.r.t. the single global source fit parameter, p_{s}, is then given by

$$\frac{\mathrm{d}\mathcal{R}_i(\vec{p}_{\mathrm{s}})}{\mathrm{d}p_{\mathrm{s}}} = -\frac{1}{A^2}\frac{\mathrm{d}A}{\mathrm{d}p_{\mathrm{s}}} \sum_{k=1}^{K} a_k \mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k}) + \frac{1}{A}\sum_{k=1}^{K} \left(\frac{\mathrm{d}a_k}{\mathrm{d}p_{\mathrm{s}}} \mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k}) + a_k \frac{\mathrm{d}\mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k})}{\mathrm{d}p_{\mathrm{s}}} \right). \tag{56}$$

Using $\mathcal{R}_i(\vec{p}_{\mathrm{s}})$ from equation (55) it simplifies to

$$\frac{\mathrm{d}\mathcal{R}_i(\vec{p}_{\mathrm{s}})}{\mathrm{d}p_{\mathrm{s}}} = \frac{1}{A(\vec{p}_{\mathrm{s}})} \left[-\mathcal{R}_i(\vec{p}_{\mathrm{s}}) \frac{\mathrm{d}A}{\mathrm{d}p_{\mathrm{s}}} + \sum_{k=1}^{K} \left(\frac{\mathrm{d}a_k}{\mathrm{d}p_{\mathrm{s}}} \mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k}) + a_k \frac{\mathrm{d}\mathcal{R}_{k,i}(\vec{p}_{\mathrm{s}_k})}{\mathrm{d}p_{\mathrm{s}}} \right) \right], \tag{57}$$

with the derivative of $A(\vec{p}_{\mathrm{s}})$ given by

$$\frac{\mathrm{d}A}{\mathrm{d}p_{\mathrm{s}}} = \sum_{k=1}^{K} \frac{\mathrm{d}a_k}{\mathrm{d}p_{\mathrm{s}}}. \tag{58}$$

In case one would fit each source individually with its own set of signal fit parameters, $\vec{p}_{\mathrm{s},k}$, $\vec{p}_{\mathrm{s}}$ would be a set of K sets of source fit parameters $\vec{p}_{\mathrm{s},k}$, and a derivative for each individual source fit parameter $p_{\mathrm{s},k}$ would have to be calculated. The expression for such a derivative would be similar to equation (57), but only the summand for the particular source, for which the fit parameter is for, would contribute.

2.10 Multiple Datasets

With SkyLLH a set of J different data samples (datasets), D_j, can be analyzed at once. Each data sample has its own detector signal yield, $\mathcal{Y}_{\mathrm{s}_j}$.

The composite likelihood function is the product of the individual dataset likelihood functions:

$$\log \Lambda = \sum_{j=1}^{J} \log \Lambda_j. \tag{59}$$

The total number of signal events n_{s} needs to get split-up into n_{s_j} for the individual datasets. The distribution of n_{s} along the different datasets is based on the detector signal yield, $\mathcal{Y}_{\mathrm{s}_j}$, of each dataset. For a single source it is given by:

$$n_{\mathrm{s}_j}(n_\mathrm{s}, \vec{p}_\mathrm{s}) = n_\mathrm{s} \frac{\mathcal{Y}_{\mathrm{s}_j}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})}{\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j'}}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})}, \tag{60}$$

where $\vec{x}_\mathrm{s}$ and $\vec{p}_\mathrm{s}$ denote the source position and flux fit parameters, e.g. the spectral index γ, respectively. The detector signal yield can be calculated via the detector effective area and the source flux (*c.f.* section 3).

By defining the dataset weight factor

$$f_j(\vec{p}_\mathrm{s}) \equiv \frac{\mathcal{Y}_{\mathrm{s}_j}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})}{\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j'}}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})} \tag{61}$$

with the property

$$\sum_{j=1}^{J} f_j = 1 \tag{62}$$

equation 60 reads

$$n_{\mathrm{s}_j}(n_\mathrm{s}, \vec{p}_\mathrm{s}) = n_\mathrm{s} f_j(\vec{p}_\mathrm{s}) \tag{63}$$

Using the dataset weight factor $f_j(\vec{p}_\mathrm{s})$ the likelihood ratio of equation (59) with equation (16) can now be written as

$$\log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \sum_{j=1}^{J} \sum_{i=1}^{N} \log(1 + n_\mathrm{s} f_j(\vec{p}_\mathrm{s}) \mathcal{X}_i(\vec{p}_\mathrm{s})). \tag{64}$$

From a reusability-of-software point of view it is advisable to be able to use the mathematical form of $\log \Lambda$ for the single dataset to calculate the combined $\log \Lambda$ value of the multiple dataset. This can be achieved by using the substitution of n_s as given by equation (63). Hence,

$$\log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s}) = \sum_{j=1}^{J} \log \Lambda_j(n_{\mathrm{s}_j}(n_\mathrm{s}, \vec{p}_\mathrm{s}), \vec{p}_\mathrm{s}). \tag{65}$$

For multiple point sources, i.e. a stacking of K point sources with positions $\vec{x}_{\mathrm{s}_k}$, the dataset weight factor of each single source needs to be taken into account via Bayes' theorem. Thus, $f_j(\vec{p}_\mathrm{s})$ can be written as the sum of the products of the dataset weight factor $f_j(\vec{p}_{\mathrm{s}_k})$ for source k, as given by equation (61), and the relative strength, $f_k(\vec{p}_{\mathrm{s}_k})$, of the kth source in all datasets compared to all the other sources in all datasets:

$$f_j(\vec{p}_\mathrm{s}) = \sum_{k=1}^{K} f_k(\vec{p}_{\mathrm{s}_k}) f_j(\vec{p}_{\mathrm{s}_k}). \tag{66}$$

The relative strength of source k can be written as

$$f_k(\vec{p}_{\mathrm{s}_k}) = \frac{\sum_{j=1}^{J} \mathcal{Y}_{\mathrm{s}_{j,k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})}{\sum_{\kappa=1}^{K} \sum_{j=1}^{J} \mathcal{Y}_{\mathrm{s}_{j,\kappa}}(\vec{x}_{\mathrm{s}_\kappa}, \vec{p}_{\mathrm{s}_\kappa})} \tag{67}$$

By combining equation 61 with $\vec{x}_\mathrm{s} \equiv \vec{x}_{\mathrm{s}_k}$ and $\vec{p}_\mathrm{s} \equiv \vec{p}_{\mathrm{s}_k}$, and equation 67, the expression for $f_j(\vec{p}_\mathrm{s})$ for multiple sources is given by:

$$f_j(\vec{p}_\mathrm{s}) = \sum_{k=1}^{K} \frac{\left(\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})\right) \mathcal{Y}_{\mathrm{s}_{j,k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})}{\left(\sum_{\kappa=1}^{K} \sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',\kappa}}(\vec{x}_{\mathrm{s}_\kappa}, \vec{p}_{\mathrm{s}_\kappa})\right) \left(\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})\right)} \tag{68}$$

The sum over the datasets of the detector signal yield for source k cancels out leaving the simplified equation

$$f_j(\vec{p}_\mathrm{s}) = \frac{\sum_{k=1}^{K} \mathcal{Y}_{\mathrm{s}_{j,k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})}{\sum_{k=1}^{K} \sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})}. \tag{69}$$

2.11 Gradients of the Multi-Dataset Log-Likelihood Ratio

By using equation (65) for the combined log-likelihood ratio, its derivative w.r.t. n_s is given by

$$\frac{\mathrm{d} \log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d} n_\mathrm{s}} = \sum_{j=1}^{J} \frac{\mathrm{d} \log \Lambda_j(n_{\mathrm{s}_j}, \vec{p}_\mathrm{s})}{\mathrm{d} n_{\mathrm{s}_j}} \frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} n_\mathrm{s}}, \tag{70}$$

with

$$\frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} n_\mathrm{s}} = f_j(\vec{p}_\mathrm{s}). \tag{71}$$

Its second derivative w.r.t. n_s is given by

$$\begin{aligned} \frac{\mathrm{d}^2 \log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d} n_\mathrm{s}^2} &= \sum_{j=1}^{J} \frac{\mathrm{d}}{\mathrm{d} n_\mathrm{s}} \left(\frac{\mathrm{d} \log \Lambda_j(n_{\mathrm{s}_j}, \vec{p}_\mathrm{s})}{\mathrm{d} n_{\mathrm{s}_j}} \right) \frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} n_\mathrm{s}} & (72) \\ &= \sum_{j=1}^{J} \frac{\mathrm{d}^2 \log \Lambda_j(n_{\mathrm{s}_j}, \vec{p}_\mathrm{s})}{\mathrm{d} n_{\mathrm{s}_j}^2} \left(\frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} n_\mathrm{s}} \right)^2. & (73) \end{aligned}$$

The derivative w.r.t. a single source fit parameter, p_s, consists of the partial derivatives of $\log \Lambda_j$ w.r.t. n_{s_j} and p_s:

$$\frac{\mathrm{d} \log \Lambda(n_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d} p_\mathrm{s}} = \sum_{j=1}^{J} \left(\frac{\partial \log \Lambda_j(n_{\mathrm{s}_j}, \vec{p}_\mathrm{s})}{\partial n_{\mathrm{s}_j}} \frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} p_\mathrm{s}} + \frac{\partial \log \Lambda_j(n_{\mathrm{s}_j}, \vec{p}_\mathrm{s})}{\partial p_\mathrm{s}} \right), \tag{74}$$

with

$$\frac{\mathrm{d} n_{\mathrm{s}_j}}{\mathrm{d} p_\mathrm{s}} = n_\mathrm{s} \frac{\mathrm{d} f_j(\vec{p}_\mathrm{s})}{\mathrm{d} p_\mathrm{s}}. \tag{75}$$

In case of a single source, the expression for the derivative of the dataset weight factor, where $f_j(\vec{p}_\mathrm{s})$ is given by equation (61), reads via the quotient rule of differentiation:

$$\frac{\mathrm{d} f_j(\vec{p}_\mathrm{s})}{\mathrm{d} p_\mathrm{s}} = \frac{\frac{\mathrm{d} \mathcal{Y}_{\mathrm{s}_j}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d} p_\mathrm{s}} \sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j'}}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s}) - \mathcal{Y}_{\mathrm{s}_j}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s}) \sum_{j'=1}^{J} \frac{\mathrm{d} \mathcal{Y}_{\mathrm{s}_{j'}}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s})}{\mathrm{d} p_\mathrm{s}}}{\left(\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j'}}(\vec{x}_\mathrm{s}, \vec{p}_\mathrm{s}) \right)^2}. \tag{76}$$

In case of multiple sources (stacking), the expression for the derivative of the dataset weight factor, where $f_j(\vec{p}_{\mathrm{s}})$ is given by equation (69) reads via the quotient rule of differentiation:

$$\frac{\mathrm{d}f_j(\vec{p}_{\mathrm{s}})}{\mathrm{d}p_{\mathrm{s}}} = \frac{\left(\sum_{k=1}^{K} \frac{\mathrm{d}\mathcal{Y}_{\mathrm{s}_{j,k}}}{\mathrm{d}p_{\mathrm{s}}}\right)\left(\sum_{k=1}^{K}\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',k}}\right) - \left(\sum_{k=1}^{K} \mathcal{Y}_{\mathrm{s}_{j,k}}\right)\left(\sum_{k=1}^{K}\sum_{j'=1}^{J} \frac{\mathrm{d}\mathcal{Y}_{\mathrm{s}_{j',k}}}{\mathrm{d}p_{\mathrm{s}}}\right)}{\left(\sum_{k=1}^{K}\sum_{j'=1}^{J} \mathcal{Y}_{\mathrm{s}_{j',k}}\right)^2} \tag{77}$$

3 Detector Signal Yield

The detector signal yield, $\mathcal{Y}_{\mathrm{s}_{j,k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k})$, is the mean number of expected signal events in the detector from a given source k in a given data sample j. For a differential source flux, $\mathrm{d}\Phi_{\mathrm{s}}/(\mathrm{d}A\mathrm{d}\Omega\mathrm{d}E\mathrm{d}t)$, it is the integral of the product of the detector effective area and this differential flux over the solid-angle, energy, and time of the source:

$$\mathcal{Y}_{\mathrm{s}_{j,k}}(\vec{x}_{\mathrm{s}_k}, \vec{p}_{\mathrm{s}_k}) \equiv \int_{\Omega_{\mathrm{s}_k}} \mathrm{d}\Omega \int_0^\infty \mathrm{d}E \int_{t_{\mathrm{start}_j}}^{t_{\mathrm{end}_j}} \mathrm{d}t A_{\mathrm{eff}_j}(E, t|\vec{x}_{\mathrm{s}_k}) \frac{\mathrm{d}\Phi_{\mathrm{s}}(E, t|\vec{p}_{\mathrm{s}_k})}{\mathrm{d}A\mathrm{d}\Omega\mathrm{d}E\mathrm{d}t} \tag{78}$$

In the most-general case, the source position $\vec{x}_{\mathrm{s}}$ consists of three quantities: right-ascension, declination, and observation time, i.e. $\vec{x}_{\mathrm{s}} = (\alpha_{\mathrm{s}}, \delta_{\mathrm{s}}, t_{\mathrm{obs}})$.

The time-dependent effective area $A_{\mathrm{eff}_j}(E, t|\vec{x}_{\mathrm{s}_k})$ must account for the detector off-time intervals within the data sample j. In cases, where the effective area is constant within a data sample, it can be written as

$$A_{\mathrm{eff}_j}(E, t|\vec{x}_{\mathrm{s}_k}) = A_{\mathrm{eff}_j}(E|\vec{x}_{\mathrm{s}_k}) T_{\mathrm{live}}(t) \tag{79}$$

with $T_{\mathrm{live}}(t)$ is the detector live-time function as given by equation (31).

3.1 Effective Area

In SkyLLH the effective area $A_{\mathrm{eff},j}$ of a data sample j is not calculated separately in order to avoid binning effects. However, the effective area can be calculated using the Monte-Carlo weights, `mcweight`[2], of the simulation events. The Monte-Carlo weights have the unit GeV cm^2 sr. Using the Monte-Carlo weight, $w_{m,j}$, of the mth event of data sample j, that corresponds to a signal event, i.e. an event that has similar properties as a signal event (*e.g.* same true direction), the effective area is given by the sum of the weights of those events, divided by the solid angle and the energy range ΔE of the summed selected events:

$$A_{\mathrm{eff}_j}(E) = \frac{\sum_{m=1}^{M} w_{m,j}}{\Omega \Delta E} \tag{80}$$

[2]In IceCube known as “OneWeight”, but which already includes the number of used MC files.

Table 1: IceCube specific detector signal yield implementation methods.

Name of Class & Description
FixedFluxPointLikeSourceI3DetSigYieldImplMethod IceCube detector signal yield implementation method for a fixed flux model and a point-like source. The flux model might contain flux parameters, but these are not fit in the likelihood maximization process. This implementation assumes that the detector effective area depends solely on the declination of the source. This method creates a spline function of given order for the logarithmic values of the $\sin(\delta)$-dependent detector signal yield. The constructor of this implementation method requires a $\sin(\delta)$ binning definition for the Monte-Carlo events and the order of the spline function. This implementation method create a detector signal yield instance of class `FixedFluxPointLikeSourceI3DetSigYield`.
PowerLawFluxPointLikeSourceI3DetSigYieldImplMethod IceCube detector signal yield implementation method for a power law flux model, implemented by the `PowerLawFlux` class, an a point-like source. This method creates a 2D spline function of given orders for the logarithmic values of the $\sin(\delta)$-dependent detector signal yield for a range of γ values. This implementation method supports multi-processing.

3.2 The DetSigYield Class

`DetSigYield` provides an abstract base class for a detector signal yield class to compute the integral given in equation (78). The detector signal yield depends on the flux model and its source parameters, which might change during the likelihood maximization process. It is also dependent on the detector effective area, hence is detector dependent. Thus, `DetSigYield` must be provided with a detector signal yield implementation method derived from the `DetSigYieldImplMethod` class.

Detector signal yield values can be retrieved via the call operator `__call__(src, src_flux_params)`, which takes the celestial source position(s), and the additional source flux parameters as arguments.

3.2.1 The DetSigYieldImplMethod Class

`DetSigYieldImplMethod` is an abstract base class and defines the interface between the detector signal yield implementation method and the `DetSigYield` class.

Table 1 lists all available IceCube specific detector signal yield implementation methods and their description.

4 The Concept of Source Hypothesis Groups

The analyses in SkyLLH rely heavily on the calculation of detector signal efficiencies. As seen in section 3, the detector signal efficiency depends on the source hypothesis (spatial model and flux model) and the detector response (dataset). Hence, the analyses require detector signal efficiency instances for each combination of source and dataset. However, the sources might be of the same kind, i.e. having the same spatial model and the same flux model. For such sources detector signal efficiency instances are needed only for each dataset. Thus, we define a group of sources with the same spatial model and flux model as a *source hypothesis group*, G_{s}.

A source hypothesis group has a list of spatial source models, e.g. point-like source locations in case of point-like sources, a flux model, and a detection signal efficiency implementation method assigned.

SkyLLH provides the `SourceHypoGroupManager` class to define the groups of source hypotheses.

5 Implemented Log-Likelihood Models

This section describes the implemented log-likelihood models. [1]

A Inverse CDF sampling of a bounded power-law

When working with the 10 years public release of IceCube's data, the generation of signal events requires as first step the generation of the true neutrino energy. In order to sample energies from a power-law, we use the technique of the inverse CDF sampling. When we are dealing with this specific release of IceCube's data, we need to sample the events from a 5 dimensional histogram which is giving us the probability of a certain reconstruction given a true neutrino energy E_ν and a true neutrino declination δ_ν.

The true neutrino energies stored in the 5-dimensional histogram are binned starting from $\log(E_\nu^{min}/\mathrm{GeV}) = 2$ up to $\log(E_\nu^{max}/\mathrm{GeV}) = 9$, which means that we can only generate energies in that energy range. In practice, we have to deal with a bounded power-law.

$$\Phi(E_\nu|\phi_0, E_0, \gamma) = \phi_0 \left(\frac{E_\nu}{E_0}\right)^{-\gamma}, \qquad E_\nu \in [E_\nu^{min}, E_\nu^{max}] \tag{81}$$

where ϕ_0 and E_0 are normalization factors, and γ is the spectral index of the power-law.

We need to consider two separate cases now:

1. $\gamma = 1$:
   In this case the power-law reads:

$$\Phi(E_\nu|\phi_0, E_0, \gamma = 1) = \phi_0 \left(\frac{E_0}{E_\nu}\right). \tag{82}$$

   The correct normalization for the bounded power-law in this case is:

$$N = \phi_0 \int_{E_\nu^{min}}^{E_\nu^{max}} \left(\frac{E_0}{E_\nu}\right) dE_\nu = \phi_0 E_0 \log\left(\frac{E_\nu^{max}}{E_\nu^{min}}\right) \tag{83}$$

   and the cumulative distribution function (CDF) is given by:

$$x = \phi_0 \int_{E_\nu^{min}}^{E_\nu} \left(\frac{E_0}{E_\nu^{'}}\right) dE_\nu^{'} = \phi_0 E_0 \log\left(\frac{E_\nu}{E_\nu^{min}}\right) \tag{84}$$

   Therefore, the correctly normalized CDF is given by:

$$x^{'} = \frac{x}{N} = \frac{\log\left(E_\nu/E_\nu^{min}\right)}{\log\left(E_\nu^{max}/E_\nu^{min}\right)}, \tag{85}$$

   where the constant factor $\phi_0 E_0$ cancels out, and $x^{'} \in [0, 1]$.
   The inverse of Eq. 85 gives the energy as function of the CDF:

$$E_\nu = e^{x \log(E_\nu^{max}/E_\nu^{min})} E_\nu^{max} \tag{86}$$

2. $\gamma \neq 1$:
   In this case the power-law reads:

$$\Phi(E_\nu|\phi_0, E_0, \gamma \neq 1) = \phi_0 \left(\frac{E_\nu}{E_0}\right)^{-\gamma}. \tag{87}$$

   The correct normalization for the bounded power-law in this case is:

$$N = \phi_0 \int_{E_\nu^{min}}^{E_\nu^{max}} \left(\frac{E_\nu}{E_0}\right)^{-\gamma} dE_\nu = \phi_0 \frac{E_0^\gamma}{1-\gamma} \left[(E_\nu^{max})^{1-\gamma} - (E_\nu^{min})^{1-\gamma}\right] \tag{88}$$

   and the cumulative distribution function (CDF) is given by:

$$x = \phi_0 \int_{E_\nu^{min}}^{E_\nu} \left(\frac{E_\nu}{E_0}\right)^{-\gamma} dE_\nu = \phi_0 \frac{E_0^\gamma}{1-\gamma} \left[(E_\nu)^{1-\gamma} - (E_\nu^{min})^{1-\gamma}\right] \tag{89}$$

   Therefore, the correctly normalized CDF is given by:

$$x^{'} = \frac{x}{N} = \frac{\left[(E_\nu)^{1-\gamma} - (E_\nu^{min})^{1-\gamma}\right]}{\left[(E_\nu^{max})^{1-\gamma} - (E_\nu^{min})^{1-\gamma}\right]} \tag{90}$$

   where the constant factor $\phi_0 E_0^\gamma/(1-\gamma)$ cancels out, and $x^{'} \in [0, 1]$.
   The inverse of Eq. 90 gives the energy as function of the CDF:

$$E_\nu = \left\{x \left[(E_\nu^{max})^{1-\gamma} - (E_\nu^{min})^{1-\gamma}\right] + (E_\nu^{min})^{1-\gamma}\right\}^{\frac{1}{1-\gamma}} \tag{91}$$

Hence, one can randomly draw energies according to the power-law distribution by generating uniformly distributed numbers between 0 and 1 and feedinf them to the inverse CDF formula, being careful of applying the correct normalization.

References

[1] Jim Braun, Mike Baker, Jon Dumm, Chad Finley, Albrecht Karle, Teresa Montaruli. Time-Dependent Point Source Search Methods in High Energy Neutrino Astronomy. *Astropart.Phys.*, 33:175–181, 2010.