An interpretable Neuro-probabilistic Regressor

ft. SHAP

Tommaso Perniola

Index

1. Project goals
2. Inspecting the satellite data
3. A statistical analysis of the satellite dataframe
4. Regression baselines
5. SHAP: intepreting model's learned features effects
6. Kolmogorov-Arnold Regression

1. Project goals

a) The objective of the project is to model, for each time bin, the expected probability **distribution** of **photon counts** by integrating temporal dynamics with satellite metadata (e.g., orbital coordinates, altitude, and positional information).

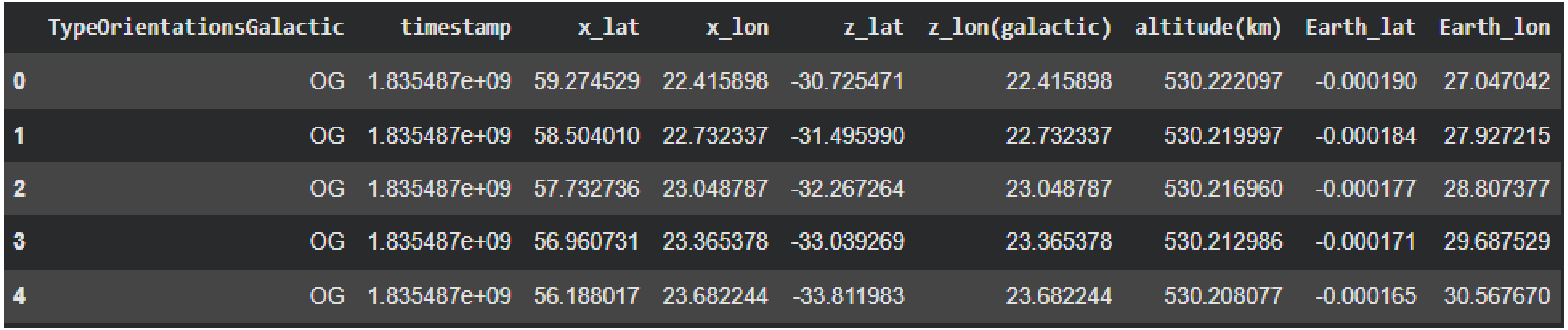
b) The orbital features will be exploited to estimate the **background count rate**, which serves as a baseline for subsequent **anomaly detection** tasks. The consistency between the predicted and theoretical distributions will be evaluated to assess the physical plausibility of the model.

c) Furthermore, **interpretability** techniques such as *SHAP* (SHapley Additive exPlanations) and *Kolmogorov–Arnold Networks* (KANs) will be employed to analyze the contribution of individual features and to uncover potential relationships between satellite configuration and photon count variability.

2. Inspecting the satellite data

A first glance

Our dataset mainly contains satellite **metadata**, that describes the orbital configuration of the satellite itself at different timestamps.



	TypeOrientationsGalactic	timestamp	x_lat	x_lon	z_lat	z_lon(galactic)	altitude(km)	Earth_lat	Earth_lon
0	OG	1.835487e+09	59.274529	22.415898	-30.725471	22.415898	530.222097	-0.000190	27.047042
1	OG	1.835487e+09	58.504010	22.732337	-31.495990	22.732337	530.219997	-0.000184	27.927215
2	OG	1.835487e+09	57.732736	23.048787	-32.267264	23.048787	530.216960	-0.000177	28.807377
3	OG	1.835487e+09	56.960731	23.365378	-33.039269	23.365378	530.212986	-0.000171	29.687529
4	OG	1.835487e+09	56.188017	23.682244	-33.811983	23.682244	530.208077	-0.000165	30.567670

2. Inspecting the satellite data

The input features

The columns of the orientation dataframe constitute the input features of our regressor. It is therefore useful to clarify their physical meaning.

The **x** and **z** coordinates describe the **orientation** (attitude) of the spacecraft, i.e., the pointing directions of two of its body axes. The **+X axis** is a lateral axis defined by the mechanical structure of the spacecraft, while the **+Z axis** corresponds to the **main** primary **pointing** direction of the satellite (the boresight).

- **x_lat / x_lon:** latitude and longitude of the spacecraft's **+X axis** direction, expressed in an Earth-centered celestial reference frame.
- **z_lat / z_lon:** latitude and longitude of the spacecraft's **+Z axis**, i.e., the **main pointing direction** (boresight) projected in the same reference frame.
- **altitude:** the **orbital altitude** (in km) of the satellite above Earth's surface.
- **Earth_lat / Earth_lon**: geographic latitude and longitude of the satellite's **sub-satellite point**, i.e., the point on Earth directly underneath the spacecraft at each instant. These coordinates describe the satellite's **position along its orbit**, independent of its pointing direction.

2. Inspecting the satellite data

Time/counts arrays and detectors

Our orientation dataframe does **not** include the target variable, which is the photon counts measured in a given time interval. Instead, these data live in two separate arrays:

- **time_array**: an array containing only timestamps, used to *align* with the “timestamp” column in the orientation dataframe.
- **total_rates:** an array containing the count rates for each detector, where:

$$r_i = \frac{c_i}{\Delta t} \quad \text{for each detector } i.$$

The rates array has shape **(num_samples, n_detectors)**, with **n_detectors = 6**. This reflects the actual hardware setup: the spacecraft has six detectors — two on each lateral side, and two on the bottom, with no detector on the top.

2. Inspecting the satellite data

Time resolution and linear interpolation

We previously stated that the orientation dataframe includes a timestamp column where each row is spaced by a fixed **15-second interval**. However, we also have three different time arrays available, each with a different temporal **resolution**:

15s　　1s　　50ms

The 15-second alignment is trivial, but for the 1-second and 50-millisecond arrays we need to **interpolate** the orientation features, so they match the finer time grids. We opt for simple **linear interpolation** (through `np.interp()`), deliberately ignoring any potential physical dynamics that might occur within those intervals.

2. Inspecting the satellite data

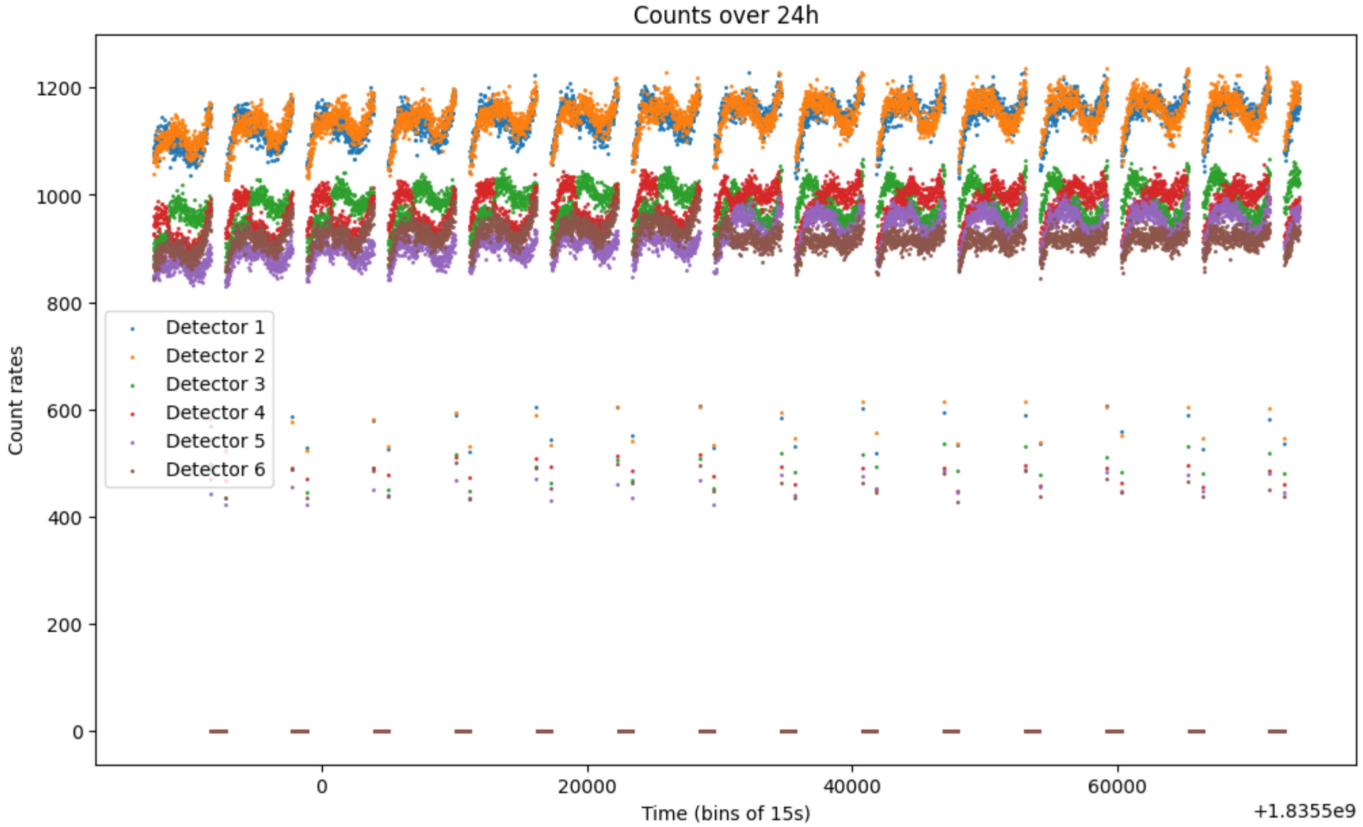
Visualizing count rates (numerically)

Num. Det.	50ms	1s	15s
Detector 1	1460.0	1073.0	1061.3
Detector 2	1120.0	1047.0	1038.7
Detector 3	1200.0	885.0	880.5
Detector 4	900.0	895.0	928.3
Detector 5	960.0	863.0	845.5
Detector 6	720.0	850.0	845.4

For the first bin
(the first row)

2. Inspecting the satellite data

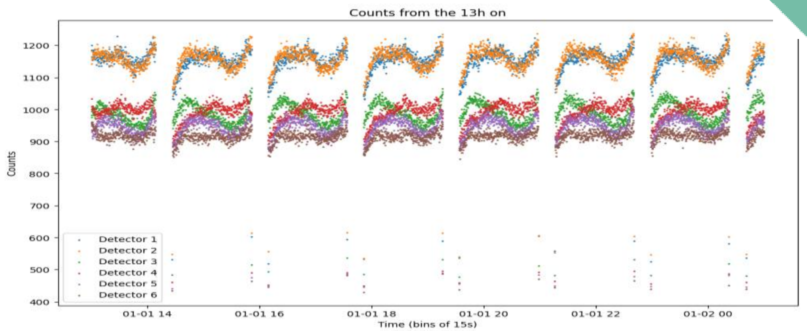
Visualizing count rates (graphically)



The plot on the side shows the global distribution of the count rates, for each detector. In particular, we can see how the **first two** detectors behave similarly, having a **mean count** rate a bit **higher** than the one for the other detectors.

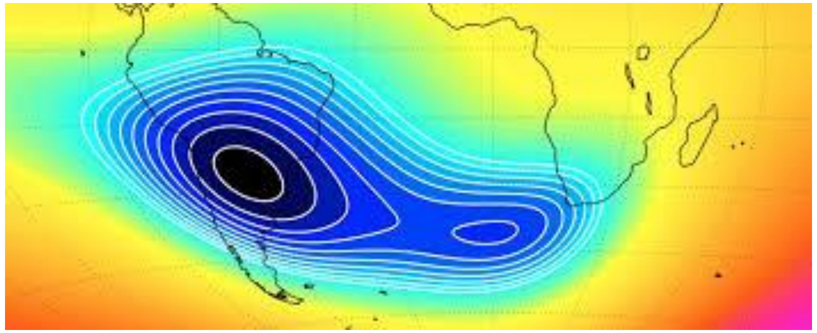
2. Inspecting the satellite data

Two clarifications



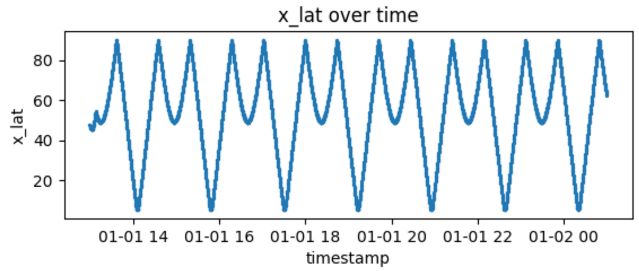
1. We will only consider data from the **second half** of the day, starting from the 13th hour. This is because during the first 12 hours the satellite is still activating, and its emission has not yet reached full intensity.

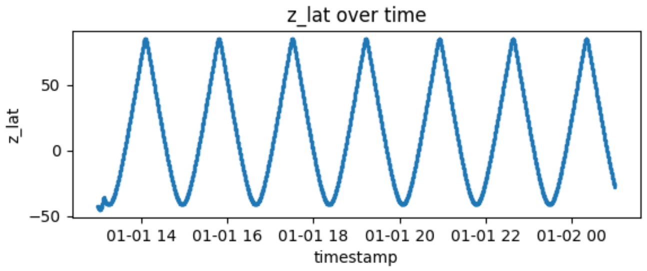
2. As we can clearly see from the previous plot, the count rates array is plenty of zeros. But why is that? This is due to the **South Atlantic Anomaly** (SAA) where the satellite is **off** while it transits the SAA. Indeed, the satellite gathers **zero** count rates in those temporal windows. Because of that, we can safely delete those rows which contain zero values in their count rates columns.

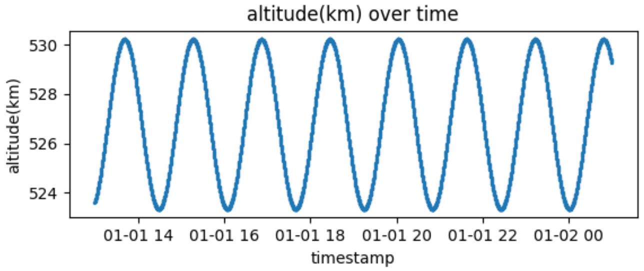


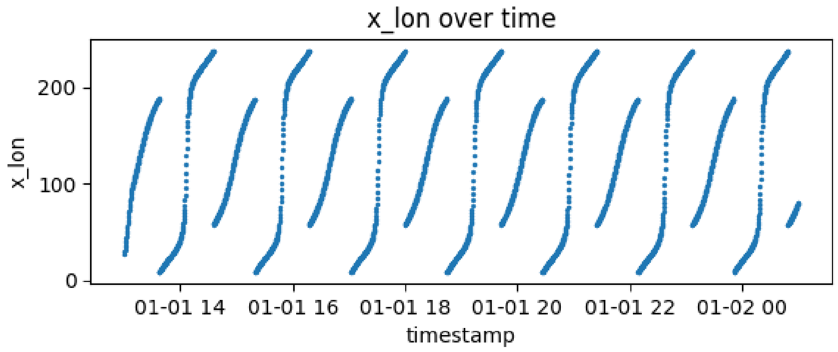
2. Inspecting the satellite data

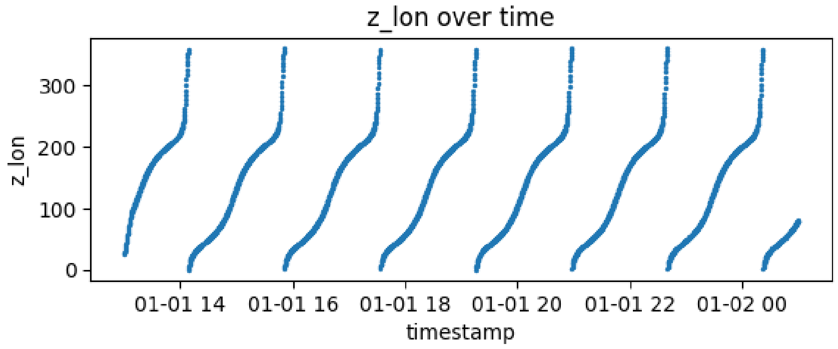
Input features distribution

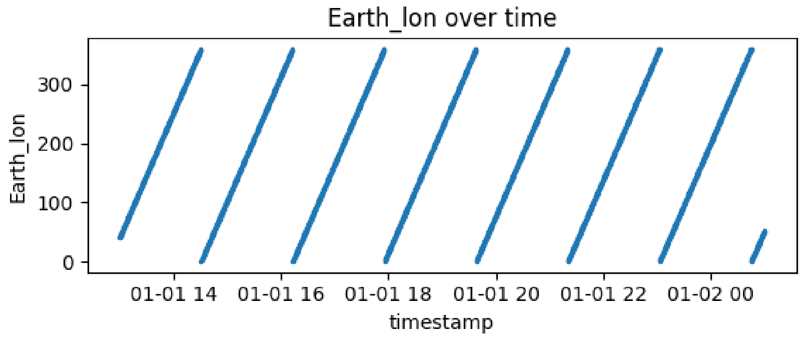

x_lat over time

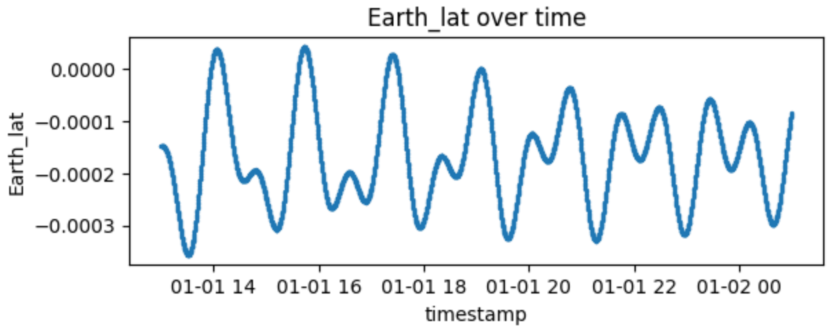

z_lat over time


altitude(km) over time


x_lon over time


z_lon over time

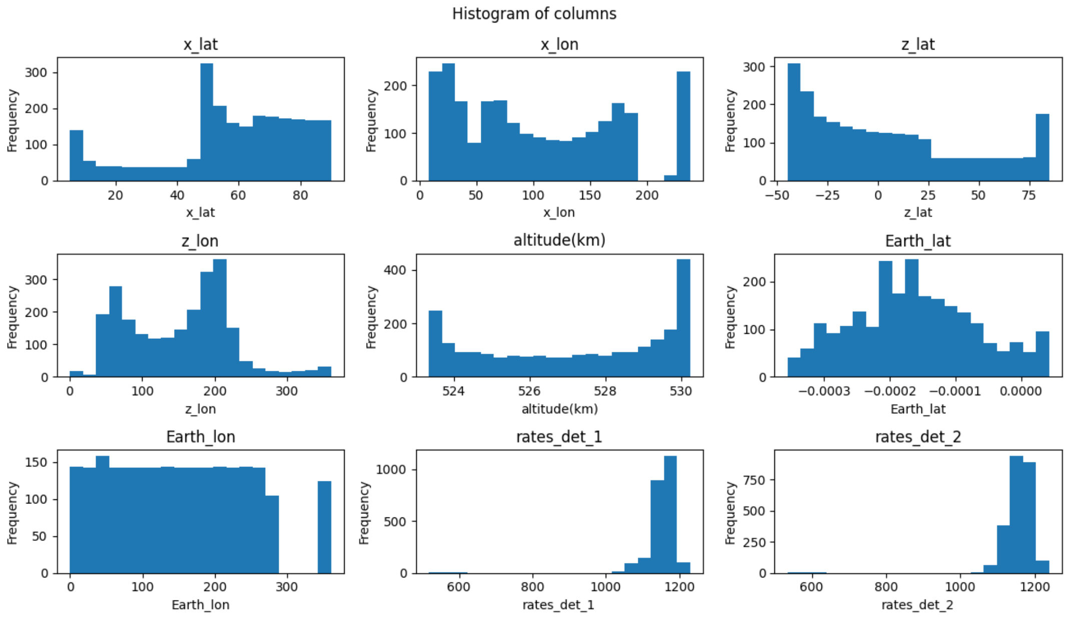

Earth_lon over time


Earth_lat over time

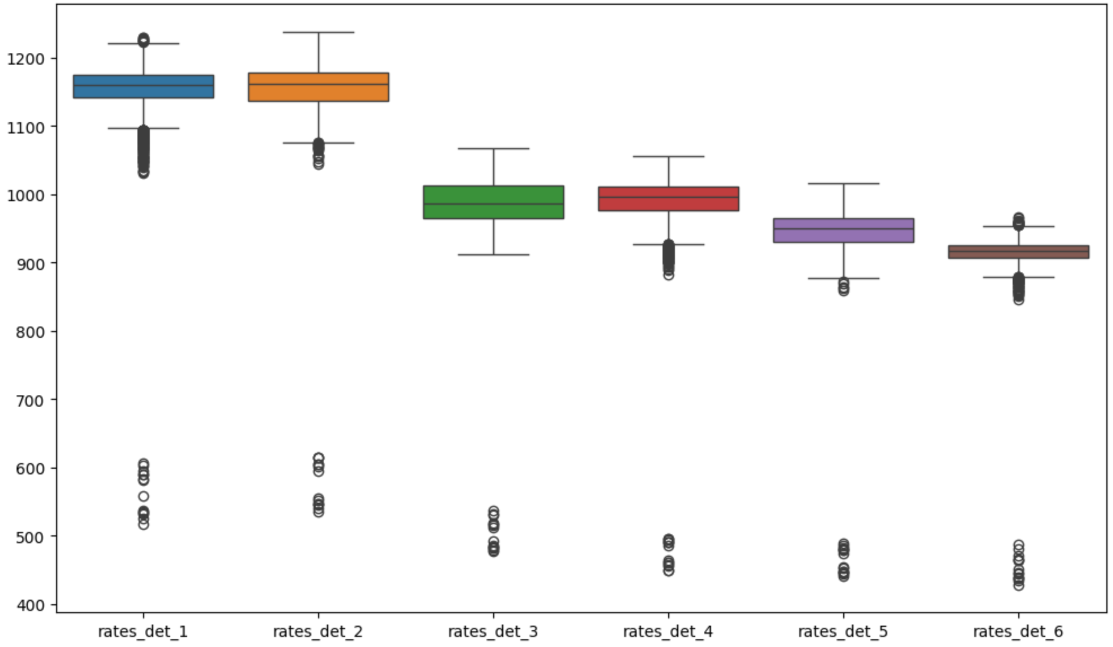
3. A statistical analysis

Feature boxplots and histograms

Features **histogram**



Count rates **boxplots**

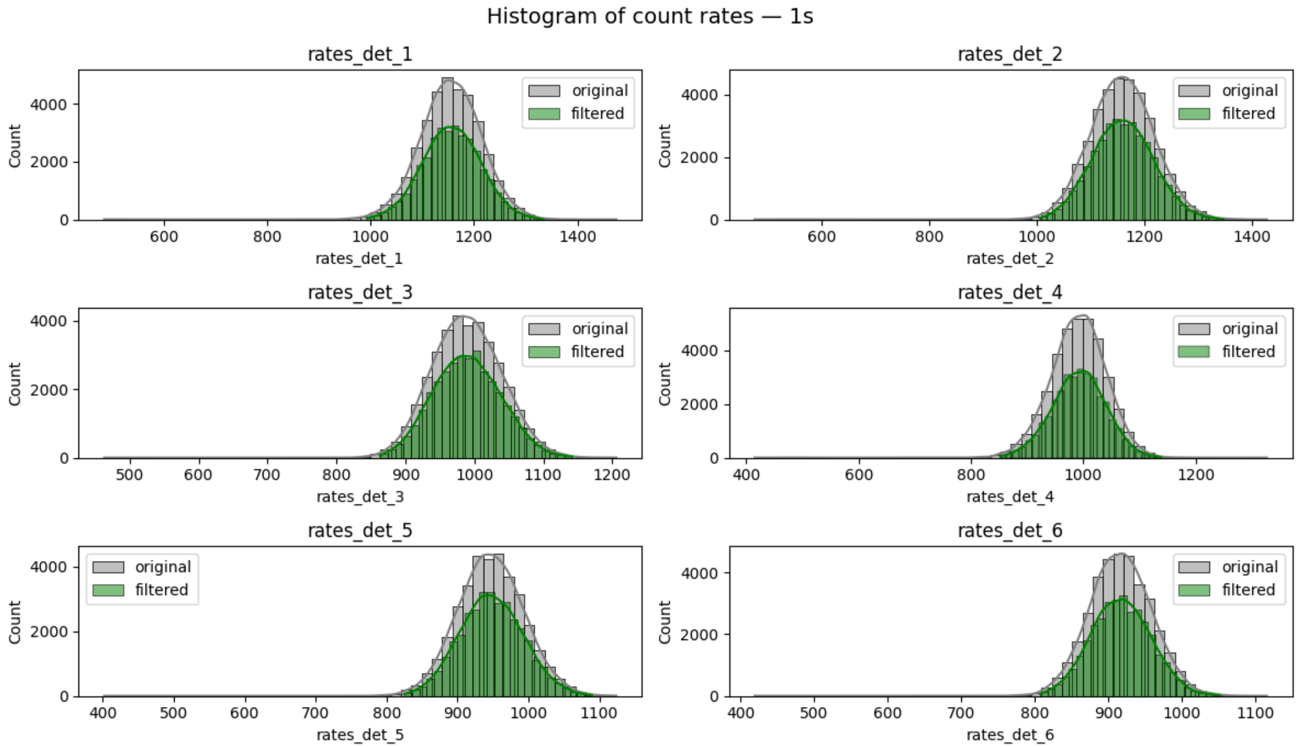


3. A statistical analysis

Removing outliers

Overall, we remove the 5% of the data, but the one on the tails. Since they are left-skewed, the quantiles are **asymmetric**.

```
def remove_outliers_quantiles(df):
  num_cols =
          df.select_dtypes(include=np.number).columns
  lower = df[num_cols].quantile(0.004)
  upper = df[num_cols].quantile(0.999)
  mask = ((df[num_cols] >= lower) & (df[num_cols]
          <= upper)).all(axis=1)
  df_filtered = df[mask]
  return df_filtered
```


Histogram of count rates — 1s

3. A statistical analysis

Theoretical count rate distribution

In order to analyze the count rate distribution, that in our case is a **per-bin** distribution (within each 15s bin).

Theoretically, the underlying process of a counting experiment is a **Poisson**, because we count independent events over a fixed time interval.

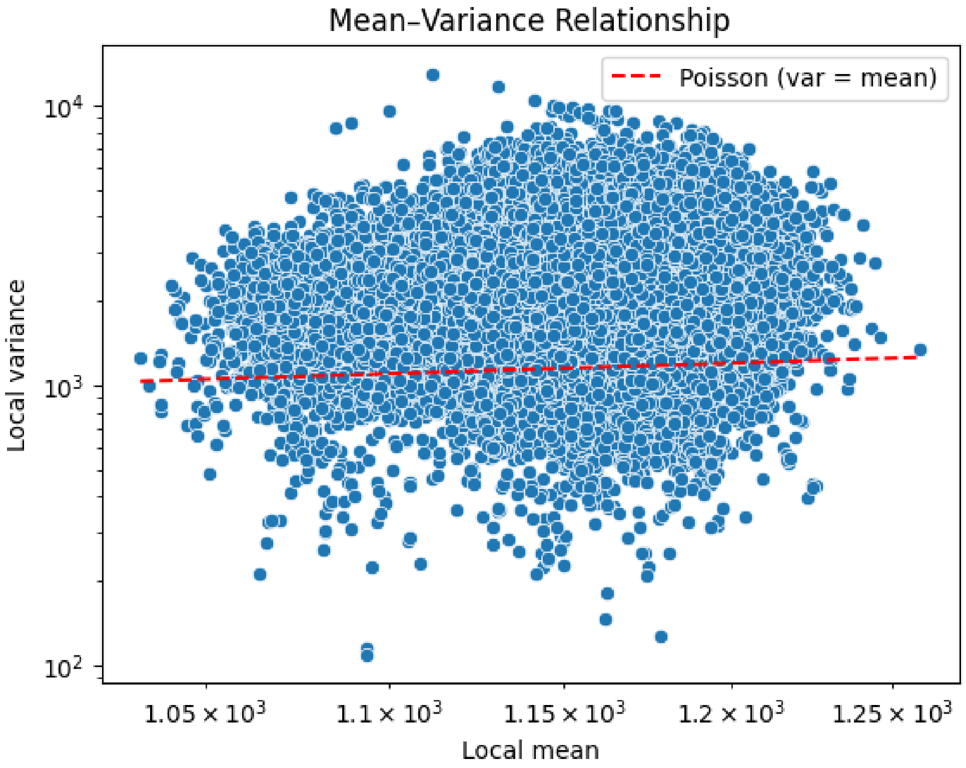
Indeed, our physical model is:

$$C_t \sim Poisson(\lambda_t) \qquad s.t. \qquad Poisson(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

where λ is the mean number of occurrences in the given interval.

3. A statistical analysis

First issue: overdispersion (var >> mean)



- Detector 1: median φ = 2.260
- Detector 2: median φ = 2.278
- Detector 3: median φ = 1.649
- Detector 4: median φ = 1.651
- Detector 5: median φ = 1.692
- Detector 6: median φ = 1.689

As we can see from the plot (in log-scale) and the results above, there is a clear sign of **overdispersion** (here 50s).

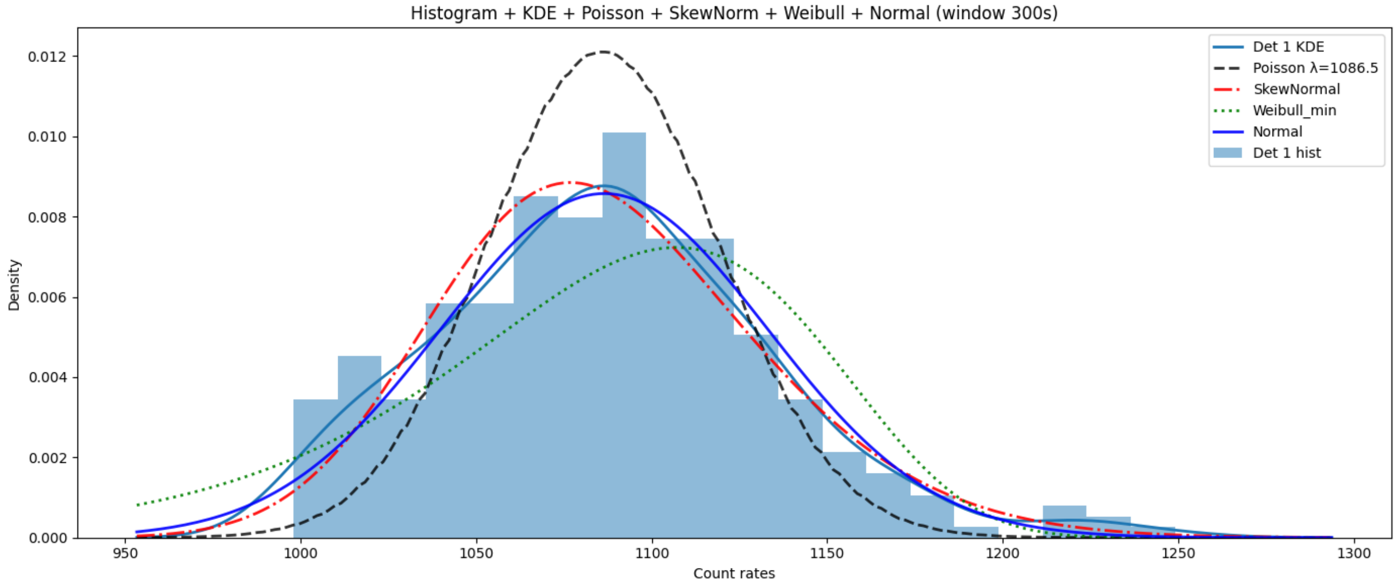
The observed variance is several times larger than the Poisson expectation, around the average count.

A plain Poisson regression will systematically underestimate uncertainty and produce biased λ estimates. Larger windows blend distinct states and inflate variance.

3. A statistical analysis

Count rate distribution | 1s

In order to properly analyze the count rate distribution, we should extract a **stationary** small window of data, to avoid to include any kind of orbital trend. Here, we will choose a representative window that is the closest to the **orbital mean** count.



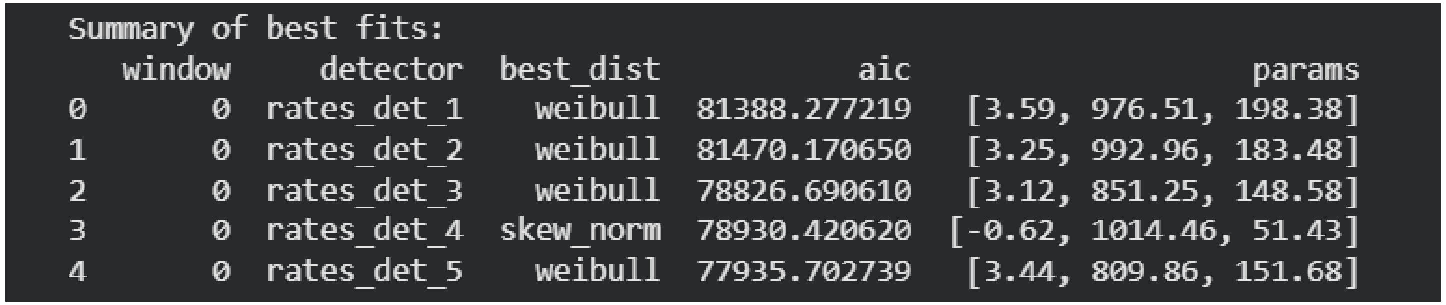
Here the chosen window is of *300s* (20 bins of data if $\nabla t = 15s$).

The orbital mean is computed over all the 6 detectors counts.

3. A statistical analysis

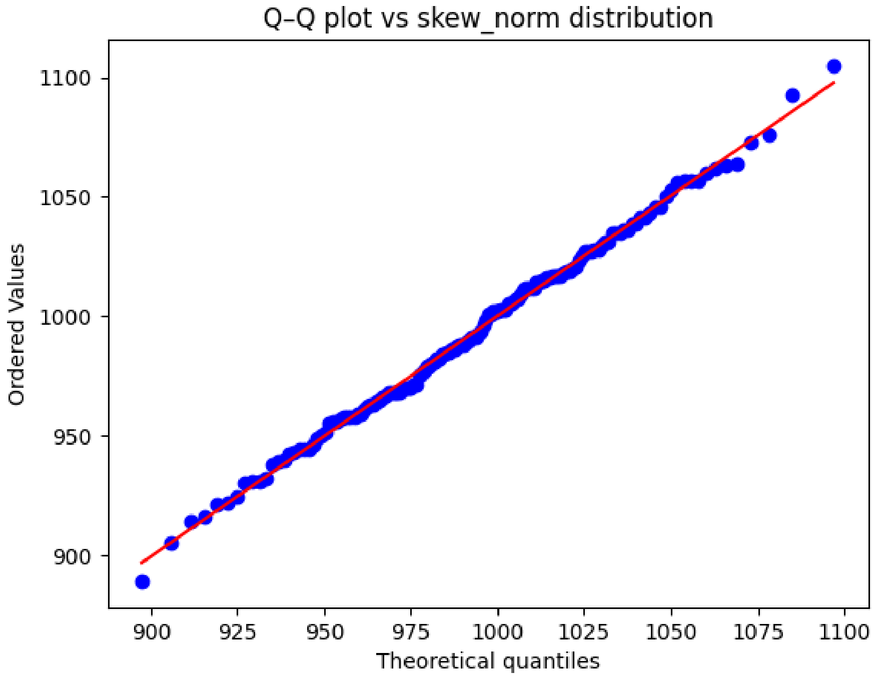
Count rate distribution | 1s

The best fitting distributions for a series of contiguous windows are the **Weibull (min)** and the **Skew Normal** distributions.



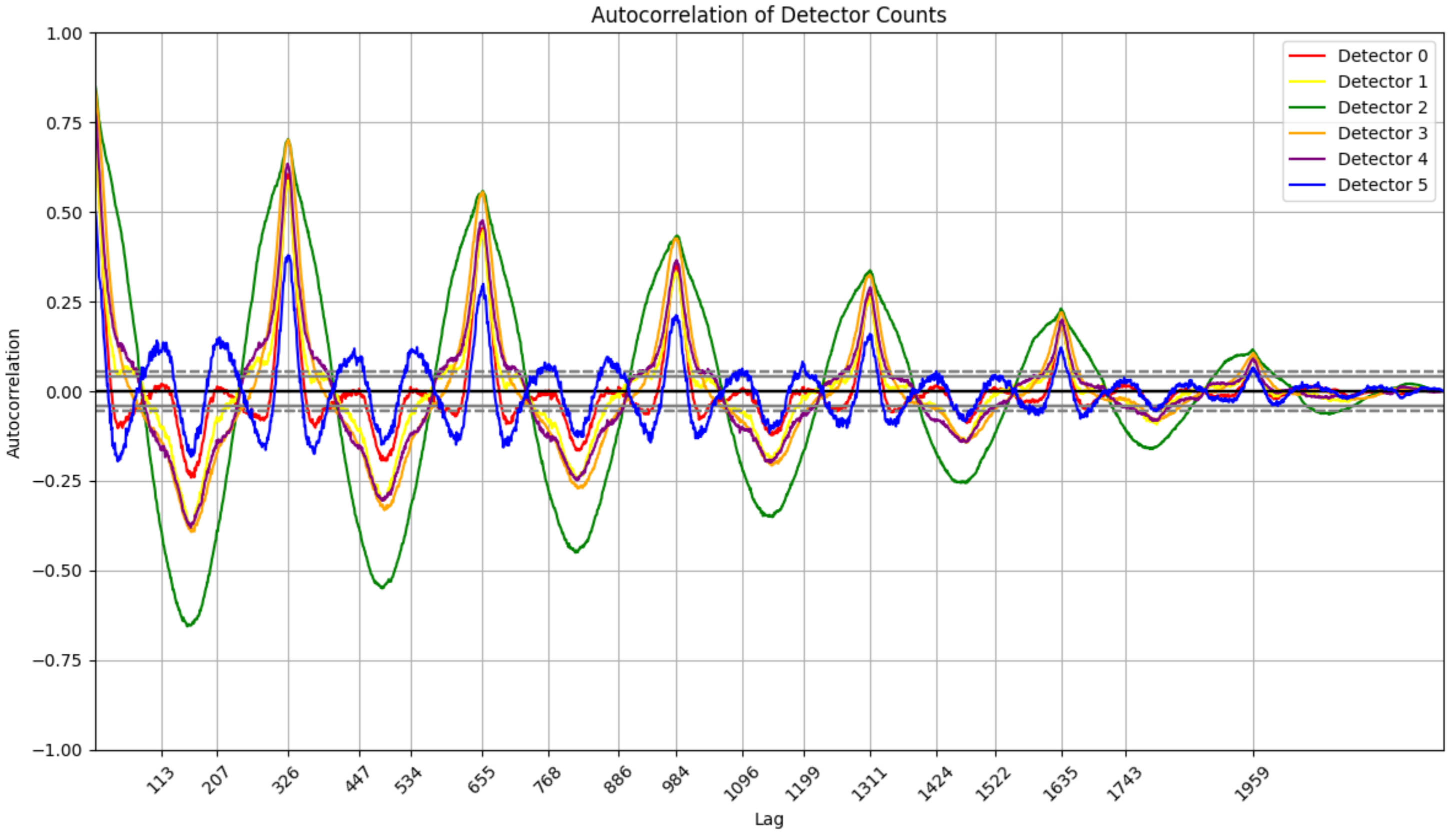
```
Summary of best fits:
   window     detector  best_dist          aic                  params
0       0  rates_det_1    weibull  81388.277219   [3.59, 976.51, 198.38]
1       0  rates_det_2    weibull  81470.170650   [3.25, 992.96, 183.48]
2       0  rates_det_3    weibull  78826.690610   [3.12, 851.25, 148.58]
3       0  rates_det_4  skew_norm  78930.420620  [-0.62, 1014.46, 51.43]
4       0  rates_det_5    weibull  77935.702739   [3.44, 809.86, 151.68]
```

To compare how well the different probabilistic regressors fit the data, we use the **Akaike Information Criterion** (AIC). It estimates the relative amount of information lost by a given model: the less information a model loses, the higher the quality of that model.



3. A statistical analysis

Autocorrelation plot: count rates



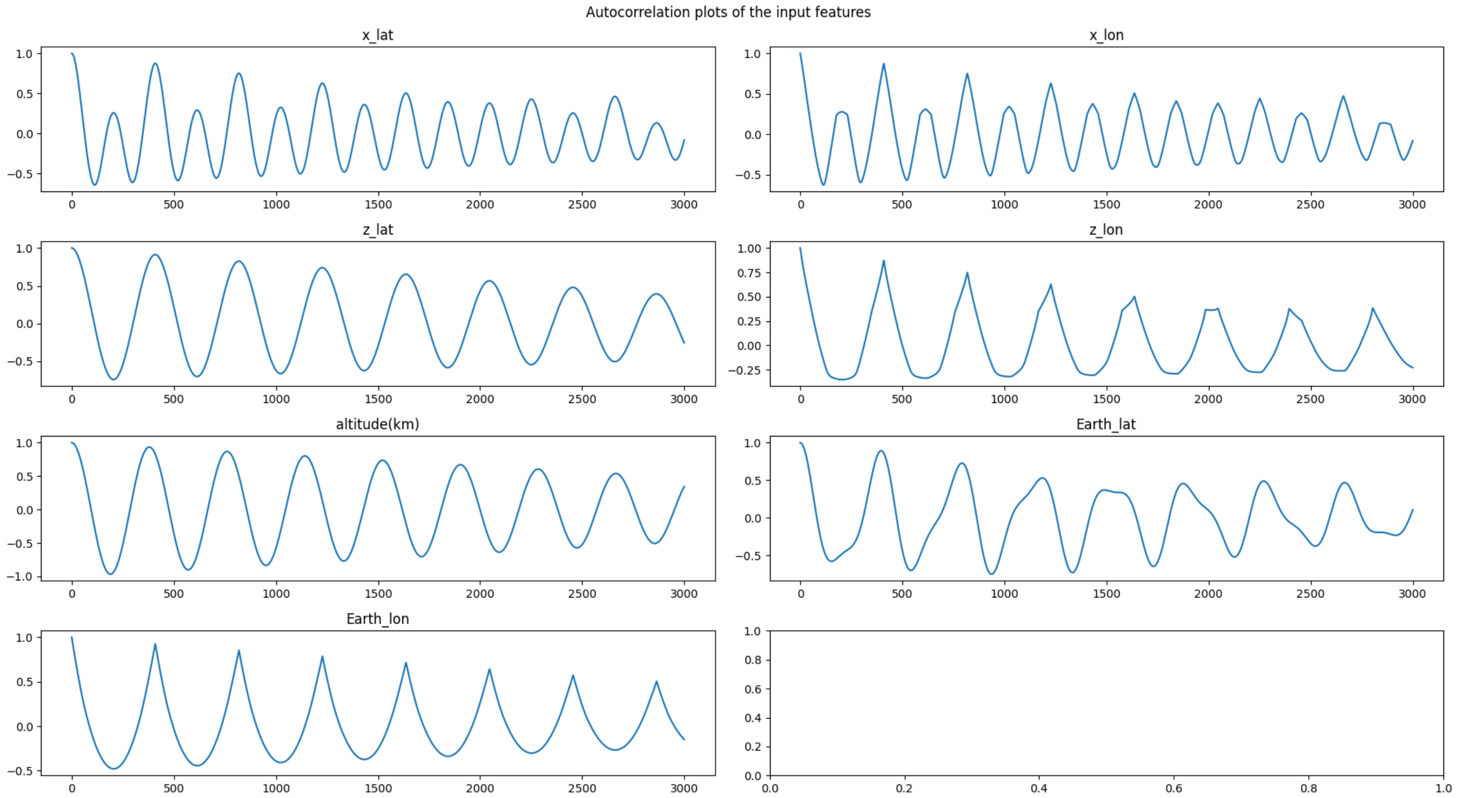
- All the detectors share somehow the same correlation pattern. Indeed, there are peaks that are repeating at regular intervals, which suggest **periodicity**. Noisier detectors shows a more irregular pattern.

- Each peak happens every *326* lags, **~1.36** hours; negative peaks indicate an inverse relationship

- We can also spot a temporal **decay**: the time series is still repeating, but each repetition is a little less like the original one. This correlation, both positive and negative, means that the pattern decreases over time. This is due to damping or the fact that noise accumulates over time.

3. A statistical analysis

Autocorrelation plot: input features



- Clear **periodicity**

- As before, we can also spot a temporal **decay.**

4. Regression

Train-test split and normalization

We split the available dataset into two distinct test subsets, each serving a different purpose:

- **Random Test Set:** This set consists of randomly **sampled**, non-contiguous timesteps. It is used to evaluate the model on heterogeneous orbital configurations and to compute global performance metrics (e.g., MAE, RMSE, NLL). These random samples represent the typical distribution of inputs seen during training.

- **Contiguous Test Set:** The second test subset contains a long, contiguous sequence of timesteps. This is required to generate continuous background **light curves** and to visually assess temporal coherence in the model's predictions. Contiguous windows allow us to inspect whether the regressor captures the slow orbital trends and geomagnetic modulations that appear in real background data.

Before training, all input features are standardized using a **StandardScaler** fitted exclusively on the **training** subset. This ensures that each feature has zero mean and unit variance, improving convergence during optimization and preventing information leakage from the test sets. The same scaler is then applied unchanged to both test subsets.

4. Regression

Evaluation metrics

To evaluate the regressor, we compute several standard error metrics that quantify different aspects of prediction quality:

- **RMSE**: it penalizes large errors more heavily because of the square. It measures the average magnitude of the prediction error with emphasis on **outliers**. A lower RMSE means the model follows the target values more closely, especially in regions with high variability.

- **MAE**: it computes the average absolute deviation between predictions and true values. It treats all errors equally and is easier to interpret than RMSE. A low MAE indicates that, on average, predictions are close to the ground truth with no bias from extreme cases.

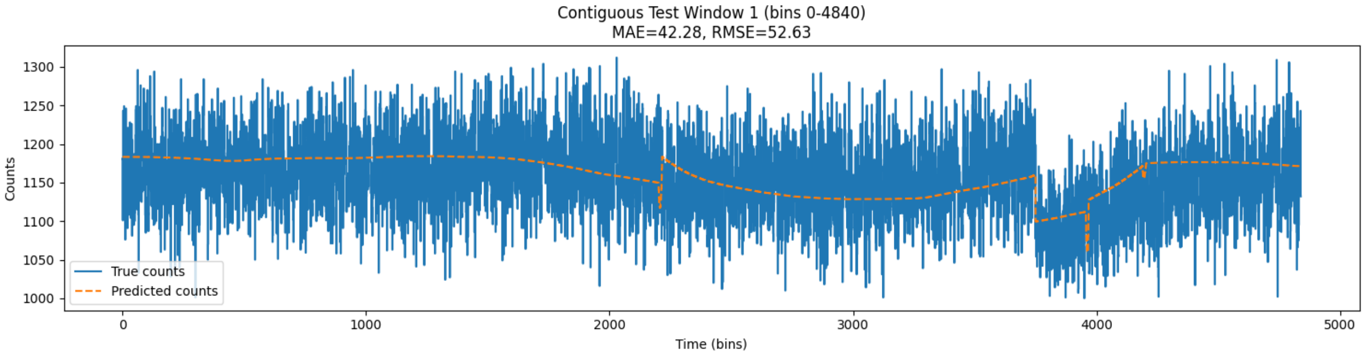
- **Mean Absolute Percentage Error**: It measures the average **relative** error, normalizing the deviation by the true value.

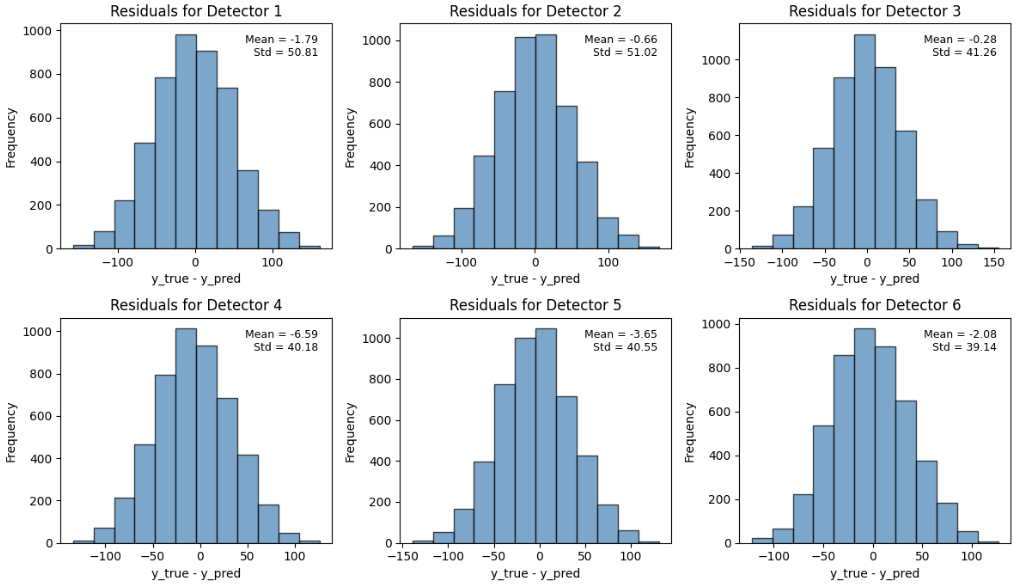
- **R^2**: it measures how much of the variance in the true signal is explained by the model, capturing how well the model reproduces the overall structure and variability of the target data.

4. Regression

Standard Regressor baseline | 1s

These are the results of a naïve dense regressor. The residuals are approximately **normally** distributed and the learned mean background signal seems faithful.

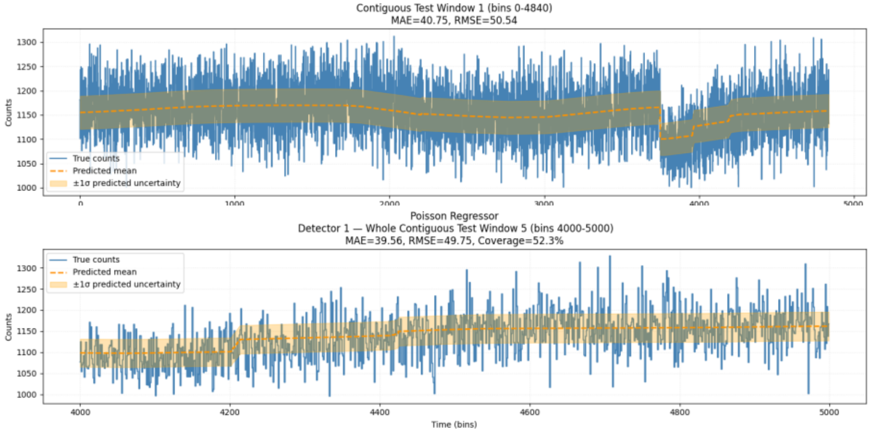




4. Regression

Neuro-probabilistic baselines

| 1s



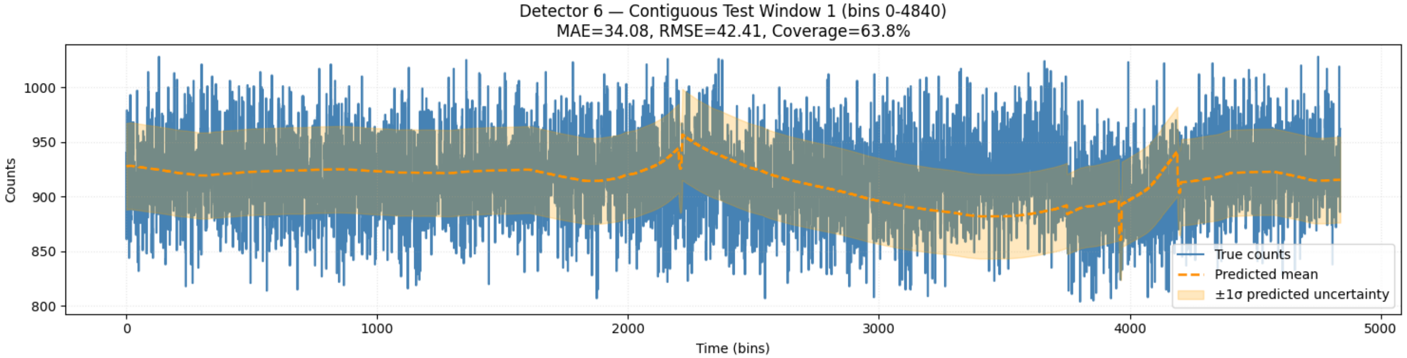
Contiguous Test Window 1 (bins 0-4840)
MAE=40.75, RMSE=50.54

Poisson Regressor
Detector 1 — Whole Contiguous Test Window 5 (bins 4000-5000)
MAE=39.56, RMSE=49.75, Coverage=52.3%

Poisson

Test MAE = 43.5

Percentage error = 4.25%

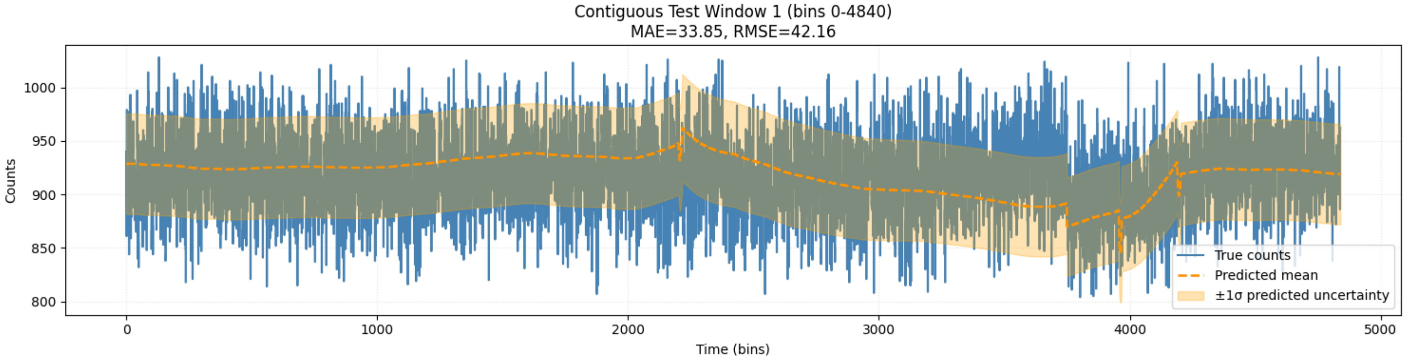


Detector 6 — Contiguous Test Window 1 (bins 0-4840)
MAE=34.08, RMSE=42.41, Coverage=63.8%

Skew-norm

Test MAE = 35.3%

Percentage error: 3.46%



Contiguous Test Window 1 (bins 0-4840)
MAE=33.85, RMSE=42.16

Weibull_min

Test MAE = 54.2%

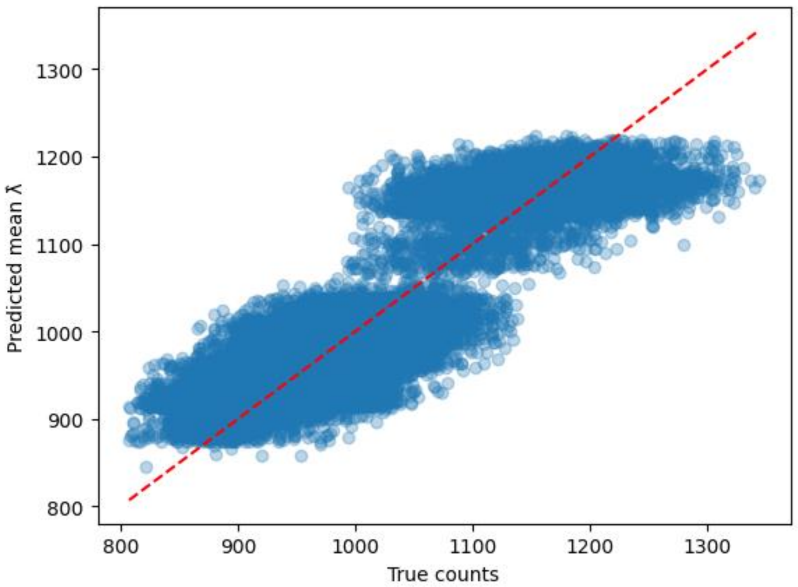
Percentage error: 5.29%

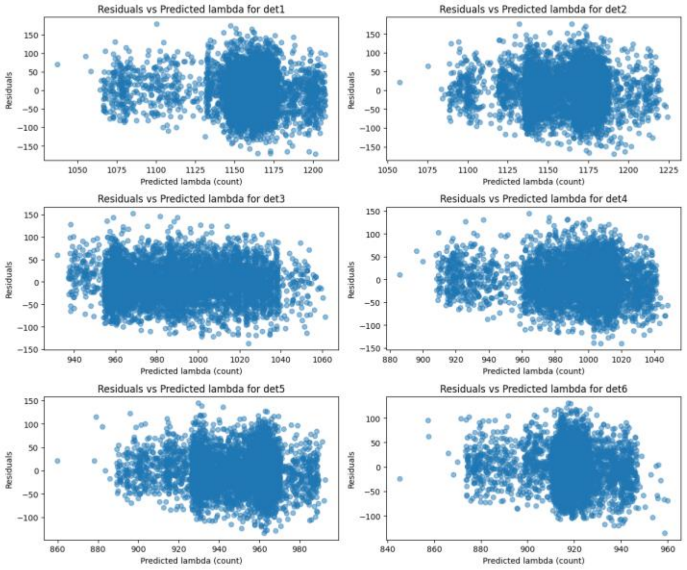
4. Regression

Neuro-probabilistic baselines

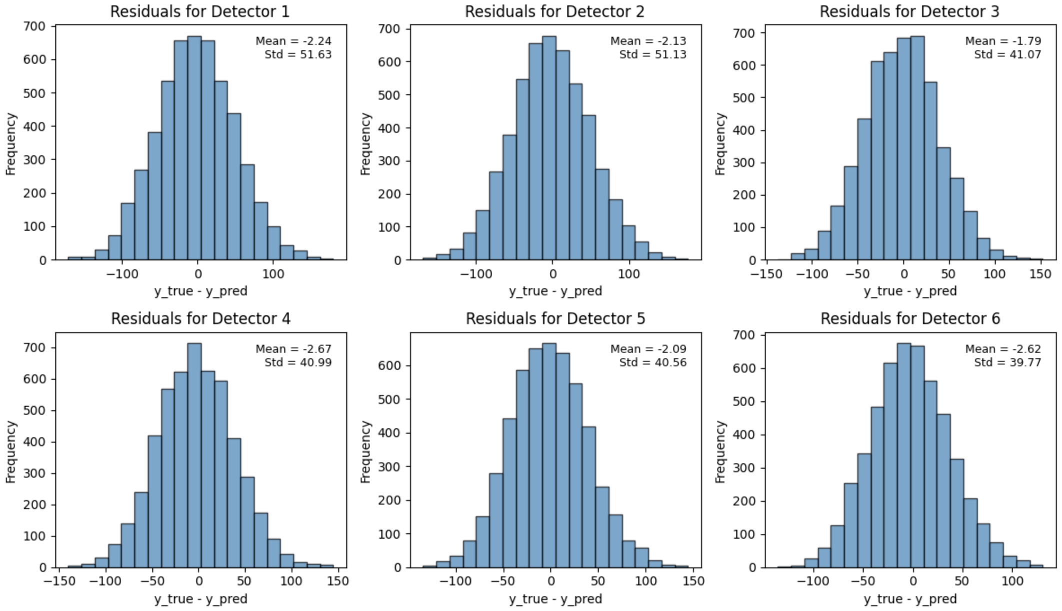
| 1s

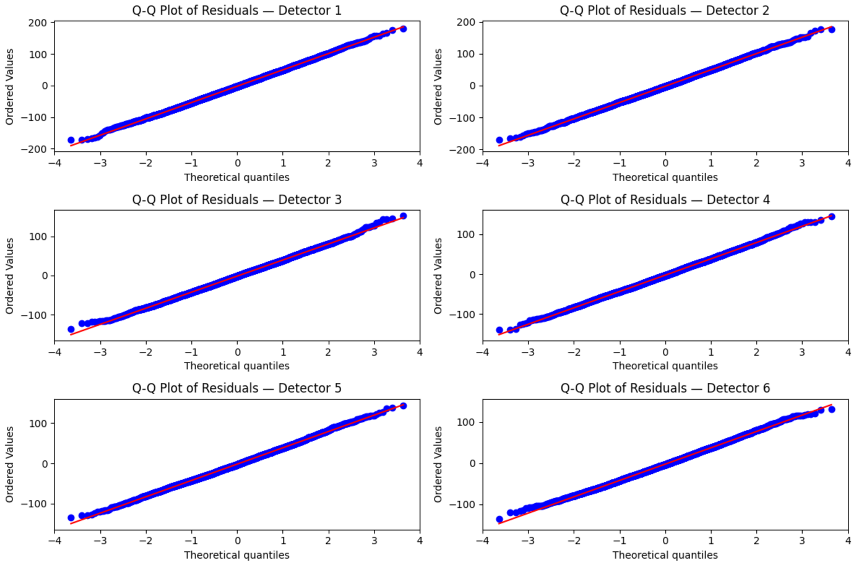
Skew-normal





Avg var/mean = 1.85





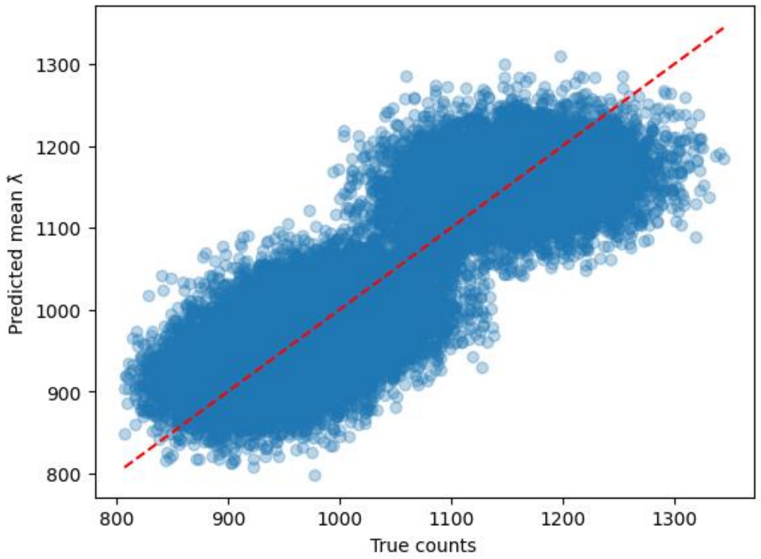
Avg mean = - 2.10

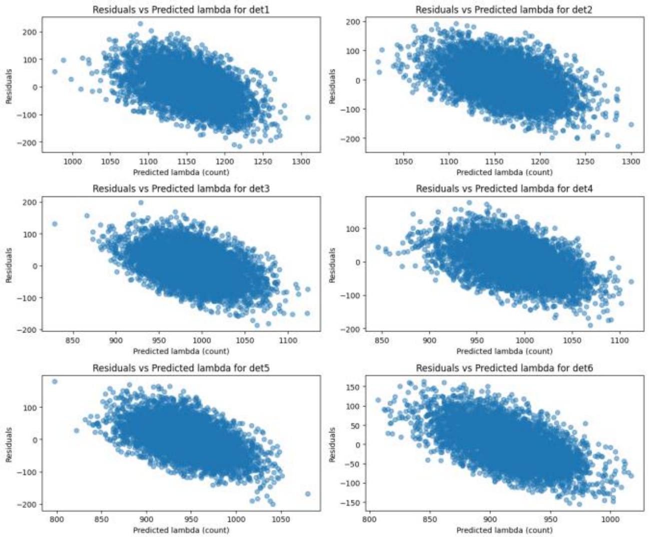
4. Regression

Neuro-probabilistic baselines

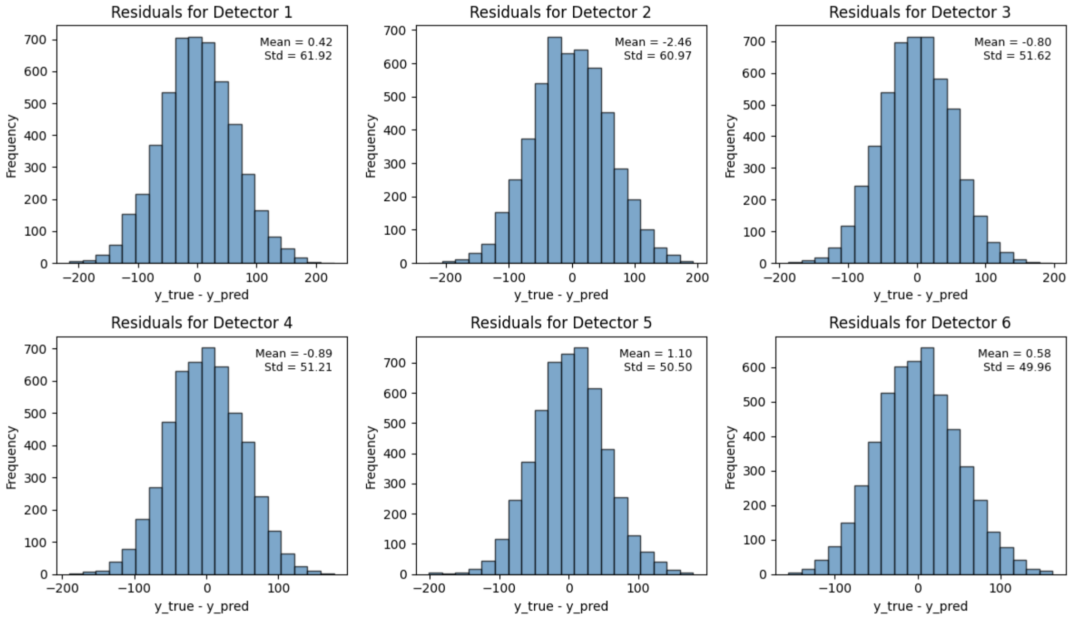
| 1s

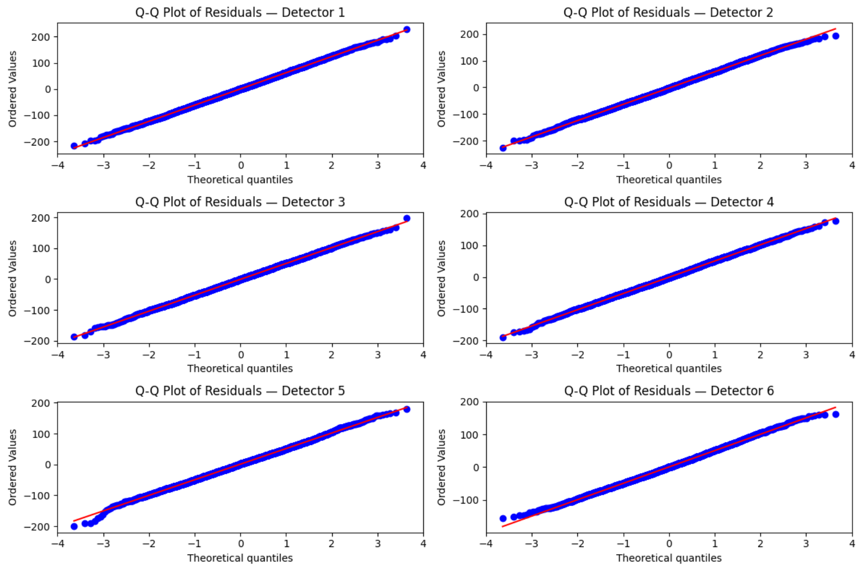
Poisson





Avg var/mean = 2.75





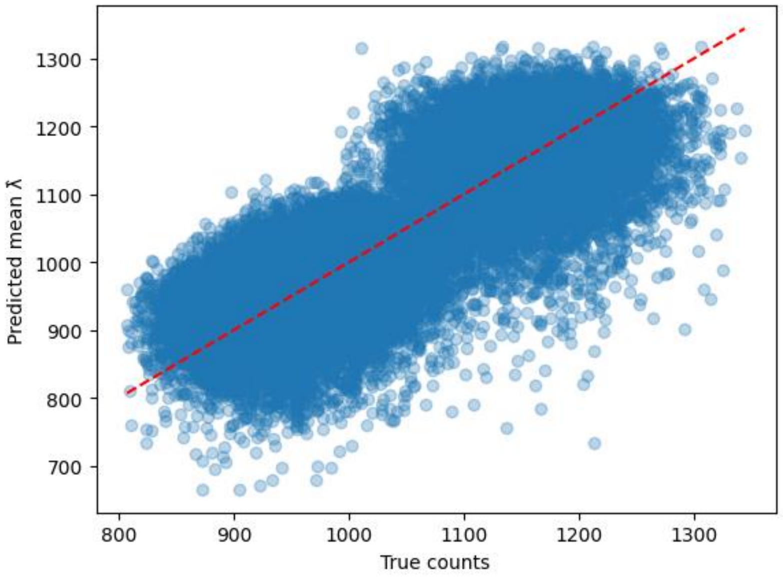
Avg mean = - 0.34

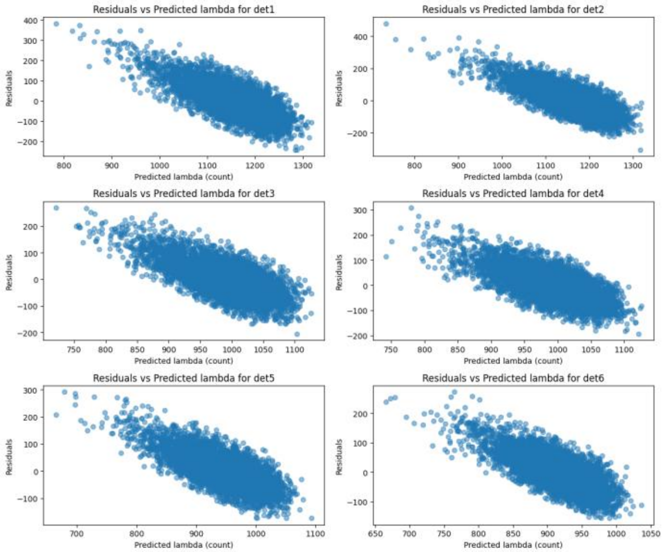
4. Regression

Neuro-probabilistic baselines

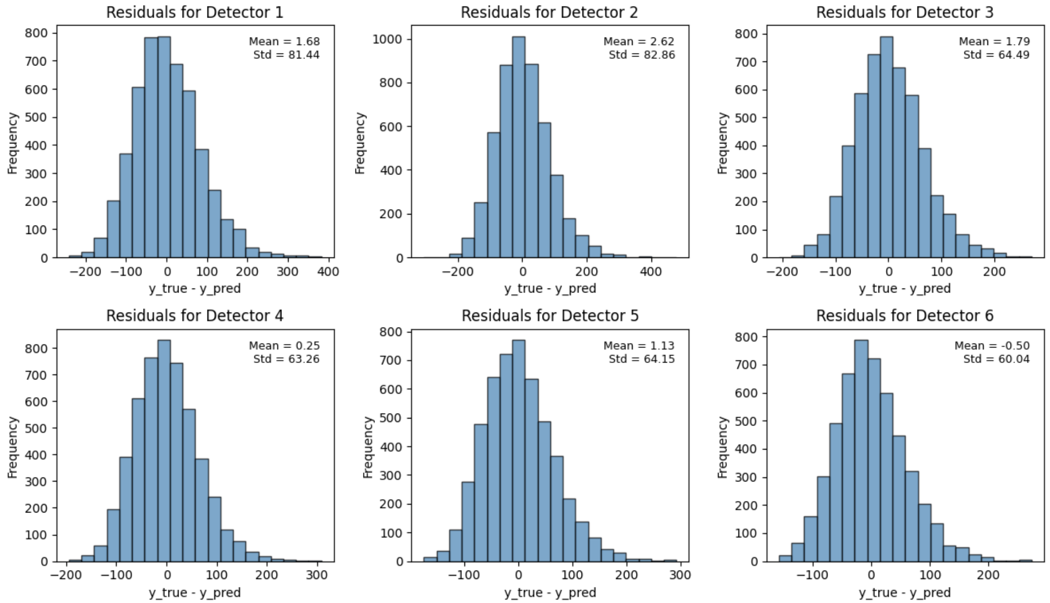
| 1s

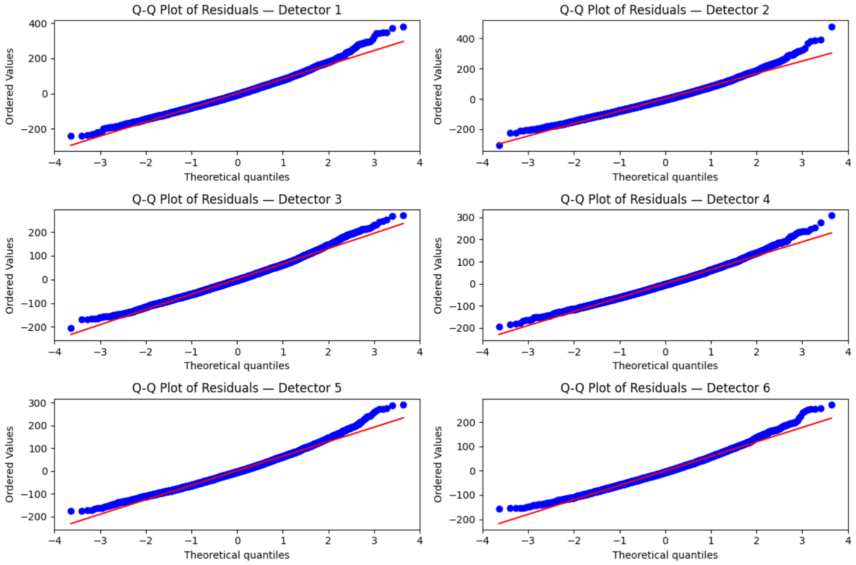
Weibull





Avg var/mean = 4.51



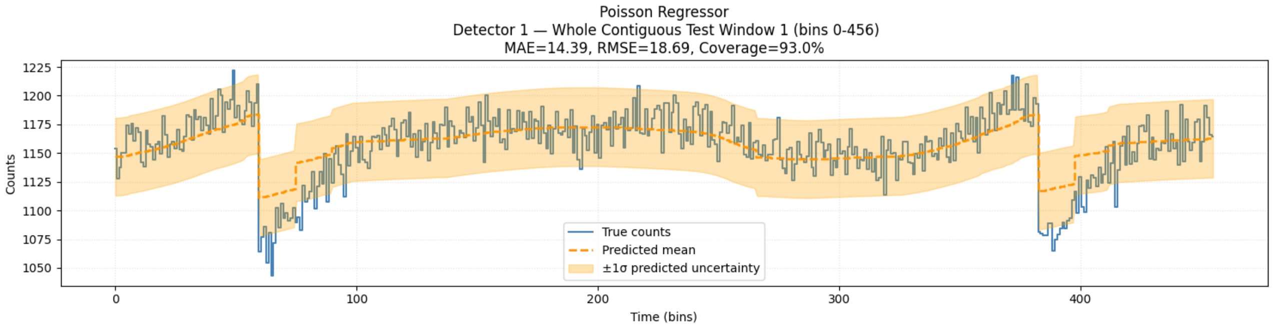


Avg mean = 1.16

4. Regression

Neuro-probabilistic baselines

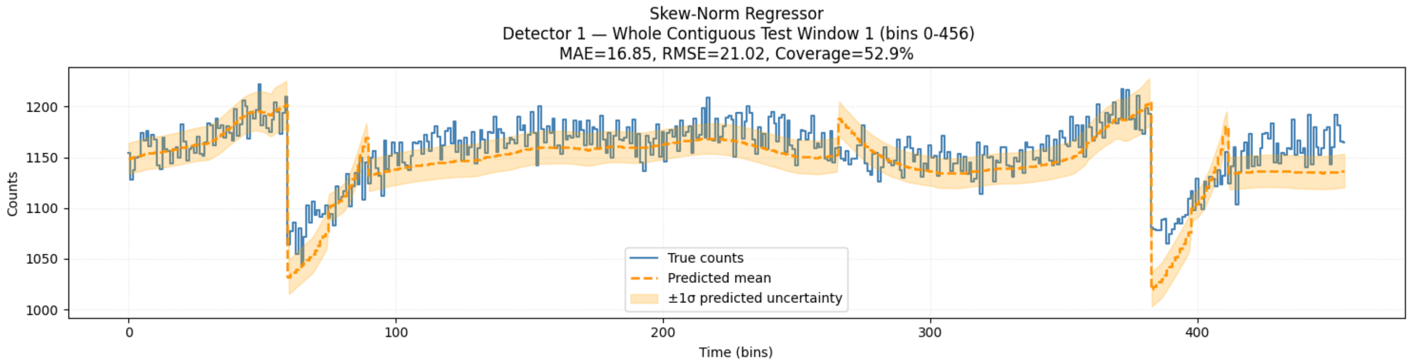
| 15s



Poisson Regressor
Detector 1 — Whole Contiguous Test Window 1 (bins 0-456)
MAE=14.39, RMSE=18.69, Coverage=93.0%

Poisson

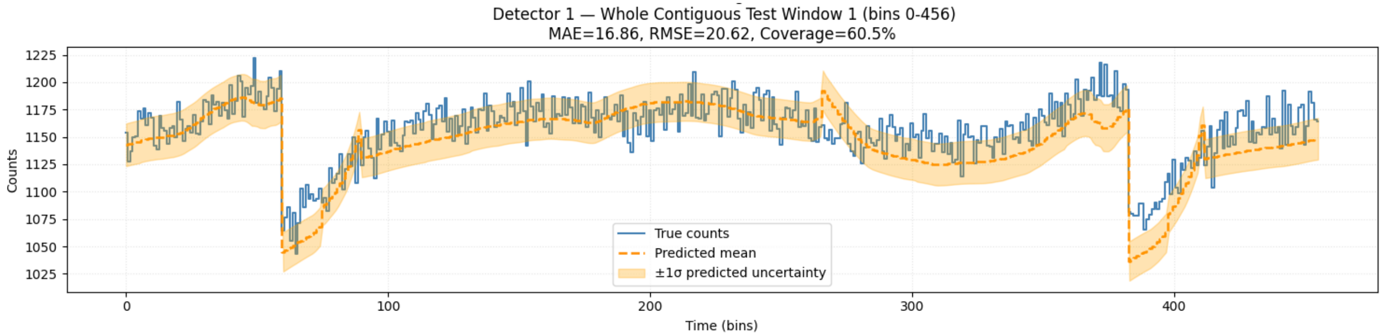
Percentage error = 2,72%



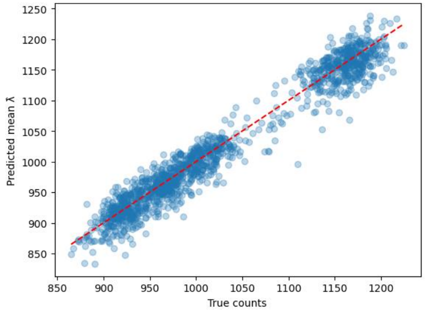
Skew-Norm Regressor
Detector 1 — Whole Contiguous Test Window 1 (bins 0-456)
MAE=16.85, RMSE=21.02, Coverage=52.9%

Skew-norm

Percentage error: 1,26%



Detector 1 — Whole Contiguous Test Window 1 (bins 0-456)
MAE=16.86, RMSE=20.62, Coverage=60.5%



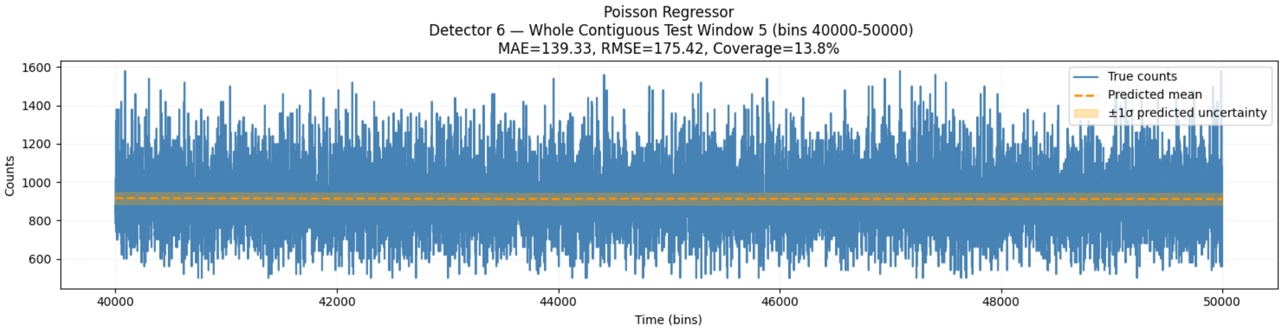
Weibull_min

Percentage error: 1,67%

4. Regression

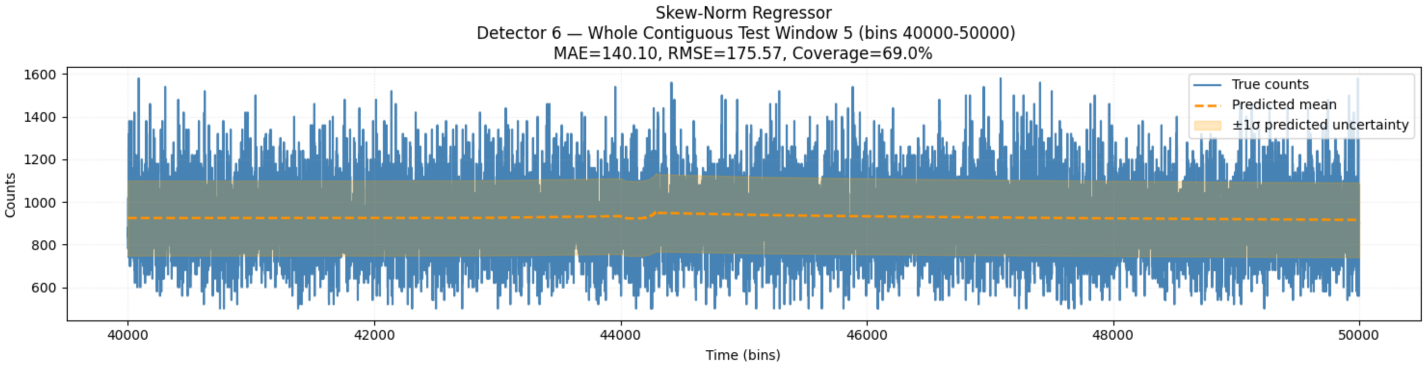
Neuro-probabilistic baselines

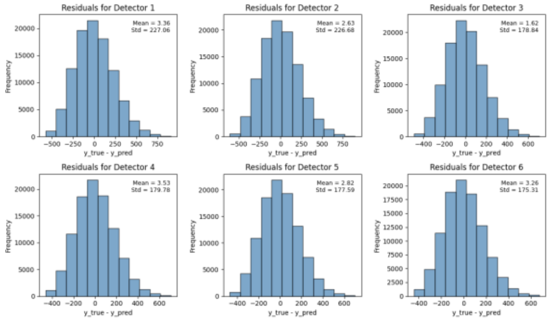
| 50ms



Poisson

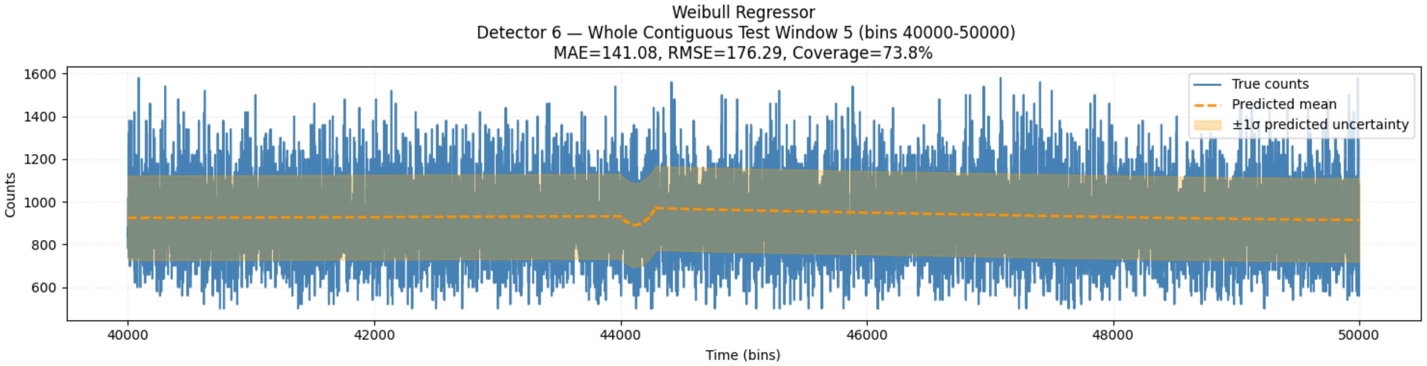
Percentage error = 15,98%





Skew-norm

Percentage error: 15,74%



Weibull_min

Percentage error: 23,52%

4. Regression

Considerations

As we have seen, the **Poisson regressor cannot model overdispersion**: its mean and variance are constrained to be equal. At **15-second** resolution, where the background exhibits strong orbital and geomagnetic **variability**, the data are highly overdispersed. In this regime the Poisson model has no dispersion parameter to absorb the excess variance, so it artificially inflates the mean λ to explain the noise. In practice, all the unmodelled variability is pushed into the predicted mean, producing large oscillations and poor calibration.

In contrast, at **50ms** the physical process is much **closer to a pure Poisson** regime—photon counts are low, and variance is naturally small. Consequently, the Poisson model behaves well and reproduces the observed variability.

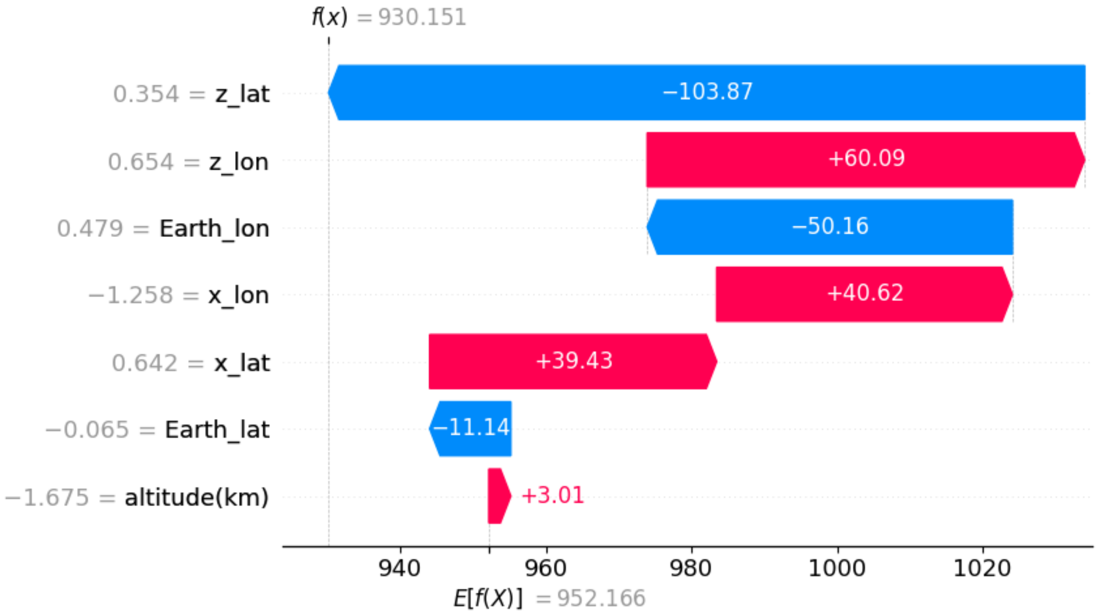
Skew-Normal and Weibull distributions include separate dispersion (and shape) parameters, allowing them to model the mean, the extra variance, and possible asymmetry independently. This flexibility enables them to **capture the overdispersed** behavior present at 15s without distorting the mean prediction.

5. SHAP

Local interpretability (SN – 1s)

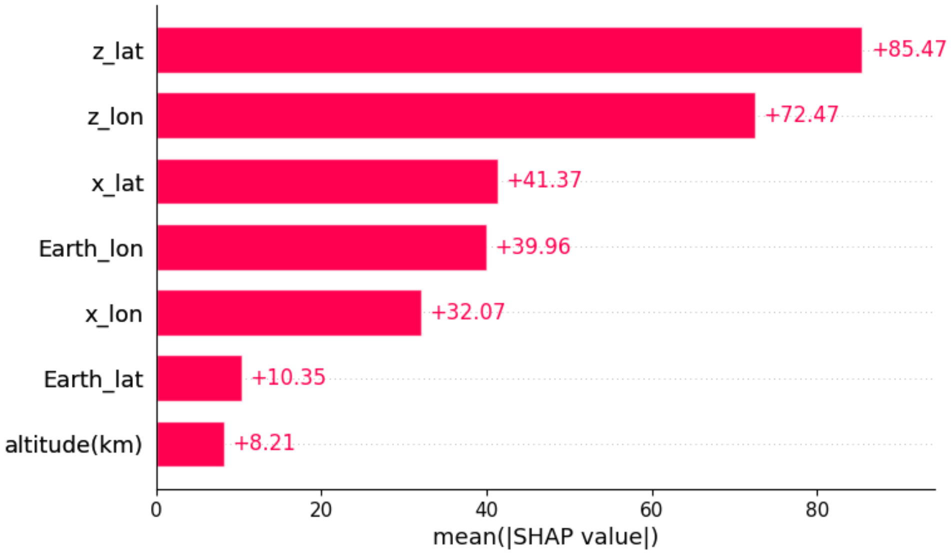
If we don't have access to any input feature, the predictions is supposed to be ~*952* (the population average).

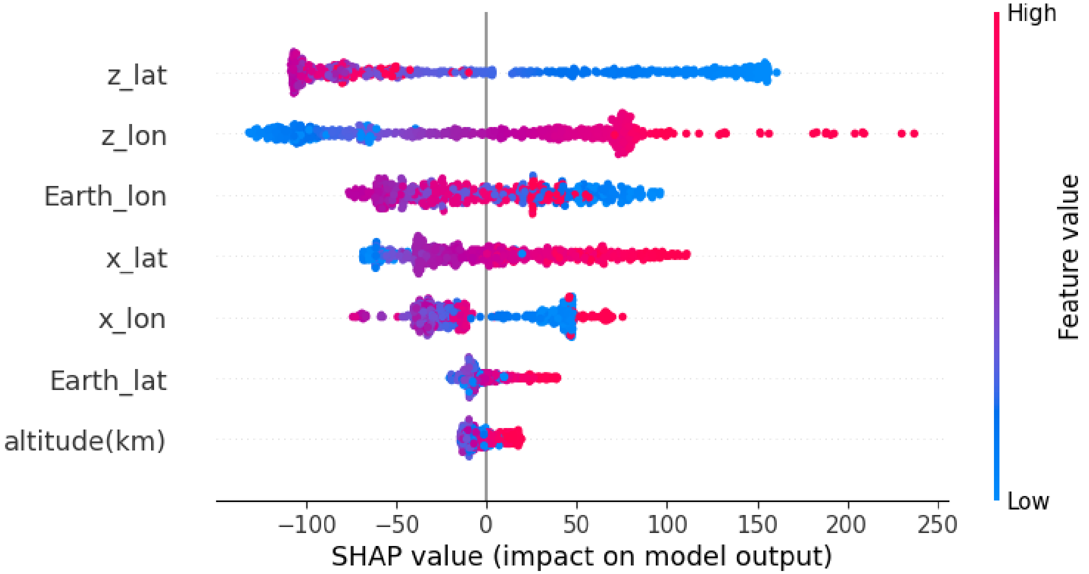
Our current prediction for the chosen sample is **lower** than the population average (the baseline), and the Shapley values of each features tells us that the *z_lat* and *x_lon* features are the one that are the most impactful.



5. SHAP

Global interpretability (SN – 1s)



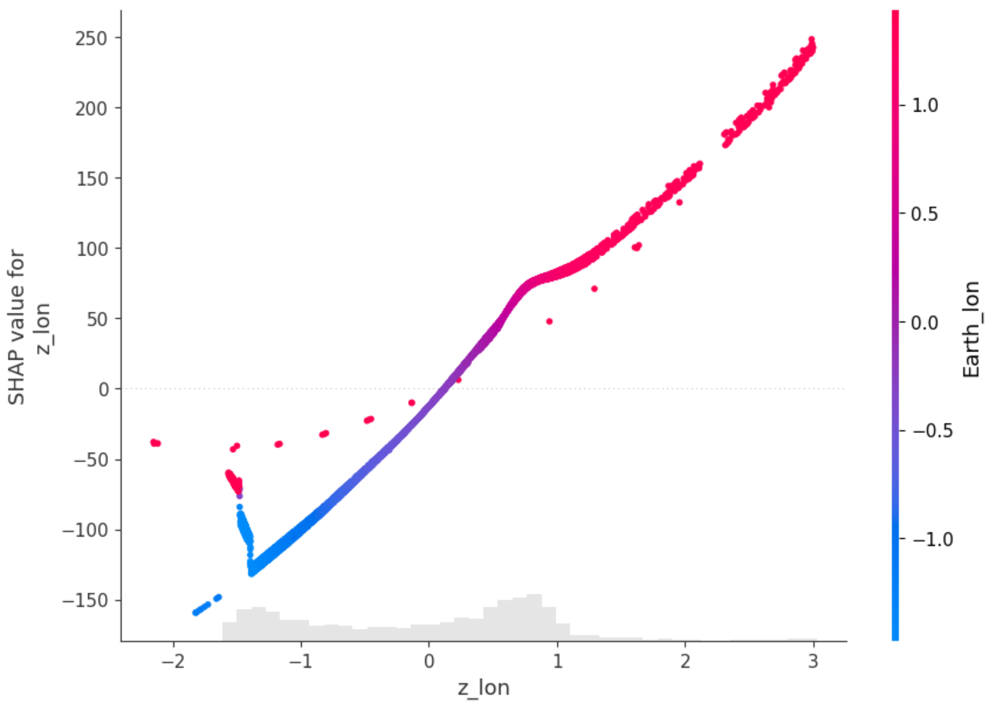


z_lat and *z_lon* are the most **influential** features

There is a **negative** correlation between *z_lat* and its shapely values; instead, there is a smoother and clearer **positive** one for *z_lon.*

5. SHAP

Global interpretability



There is a clear **positive** relationship between *z_lon* and its SHAP values: higher values of *z_lon* lead to increasingly positive contributions to the predicted background, while lower values lead to negative contributions. The point where SHAP values cross zero (roughly at *z_lon* ≈ 0.4) indicates the threshold at which the feature shifts from decreasing the prediction to increasing it.

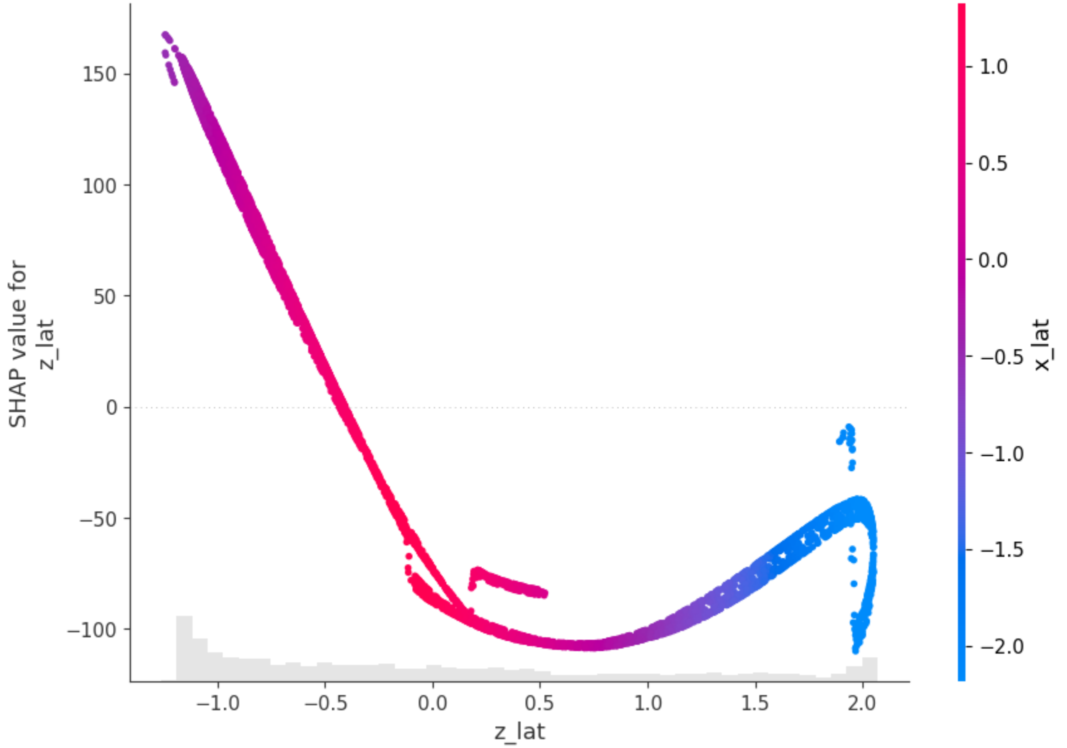
The dependence plot is colored by the feature that interacts most strongly with *z_lon*, identified as *Earth_lon*. The color pattern shows that *Earth_lon* modulates the effect of z_lon: when *Earth_lon* is close to 0, the SHAP values of *z_lon* tend to be small or positive, whereas for Earth longitudes farther from 0 the influence becomes strongly positive or strongly negative. This demonstrates a clear interaction between the two features

5. SHAP

Global interpretability

In this dependence plot we observe a behavior like the one seen for *z_lon*, but with an overall **negative** relationship between *z_lat* and its SHAP values. As *z_lat* increases, the SHAP contribution initially decreases, reaching a strong negative minimum, and then slowly rises again at higher values — producing the characteristic “golf-club” shape.

The color scale indicates the feature that interacts most strongly with *z_lat*, identified here as x*_lat*. The transition in colour shows that the effect of *z_lat* depends on the value of *x_lat*: when *x_lat* is close to zero or positive (purple/red), *z_lat* produces increasingly negative SHAP values as it grows. Conversely, when *x_lat* becomes more negative (blue), the SHAP values start increasing again at larger *z_lat*.



5. SHAP

Considerations | from a physical p.o.v.

The SHAP analysis is **consistent** with the expected **physical** behavior of the background.

As observed, *z_lat* and *z_lon* emerge as the **most influential** features. This is physically reasonable, since together they encode the latitude and longitude of the spacecraft’s main boresight direction—i.e., where the detector is pointed at each instant. The prominence of *z_lat* is also coherent with the physics: the pointing latitude should be more stable — it has a broader field of view — than the pointing longitude, which varies over a narrower range and contributes less strongly.

6. KAN