Noise mitigation with iDQ in MBTA

Amazigh OUZRIAT

PhD Candidate – Institut des 2 Infinis de Lyon

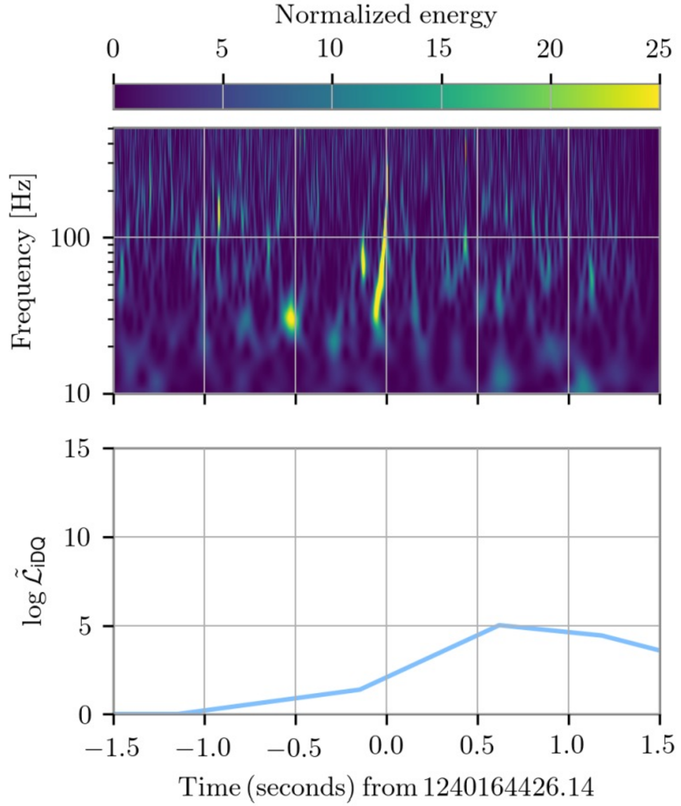
iDQ : Framework for statistical inference of glitches

- Used offline at Hanford & Livingston in O3, to be used at Virgo in O4
- iDQ correlates auxiliary data information and strain data
- Trains safe auxiliary channels only
- The iDQ timeseries are machine-learning based data quality products

  => Probabilistic quantities to estimate glitch presence in h(t)
- Incorporated as DQ flag in PyCBC & GstLAL

Motivation

- Use iDQ outputs to flag DQ problems => improve search sensitivity
- MBTA used CAT2 in O3, will not be produced by DetChar in O4
- Instead of vetoing : reranking events, allowing detection of loud signal during flagged times (Example: GW190424A)
- Current strategy : reweighting SNR single triggers using iDQ, expected improvement of VT



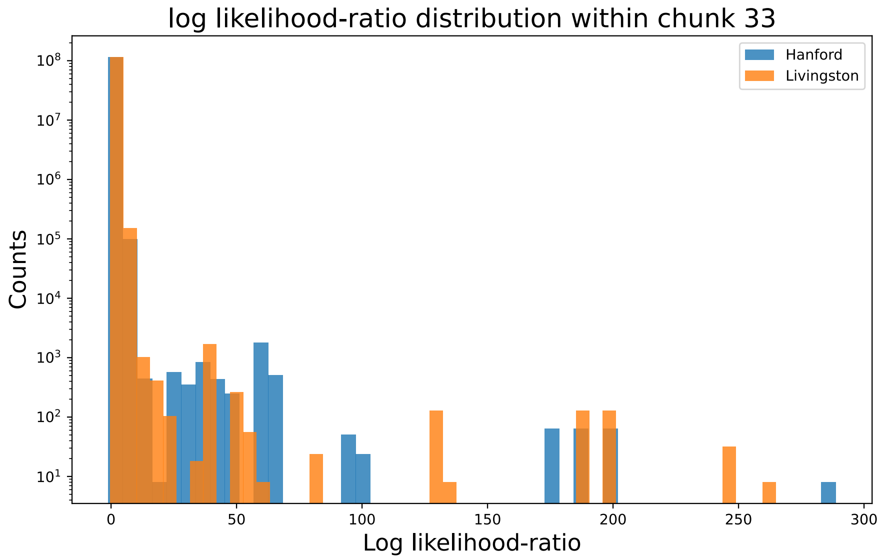
Time surrounding GW190424A vetoed by CAT2
Recovered after iDQ incorporation into GstLAL

iDQ timeseries

- 5 outputs : Rank, FAP, Eff/FAP, P_{glitch}, $\log(\mathcal{L})$
- Sampled at 128Hz
- P_{glitch} will not be produced during O4
- GstLAL uses $\log(\mathcal{L})$ to downgrade triggers : $\mathcal{L} = \frac{\mathcal{L}(glitch)}{\mathcal{L}(clean)}$
- PyCBC uses $\log(\mathcal{L})$ to correct trigger rate in the time dependent noise model

iDQ timeseries – Chunk 33

- We study MDC chunks (32-36)
- In these slides chunk 33, GPS Time within [1263751884, 1264528208]
- Use $\log(\mathcal{L})$ as DQ flag.
- Conservative approach : $\log(\mathcal{L}) < 0 \rightarrow \log(\mathcal{L}) = 0$
- Downsample from 128Hz to 1Hz
- 3 log(L) trend timeseries :
  - Maximum 1Hz $\log(\mathcal{L})$
  - Median 1Hz $\log(\mathcal{L})$
  - Mean 1Hz $\log(\mathcal{L})$

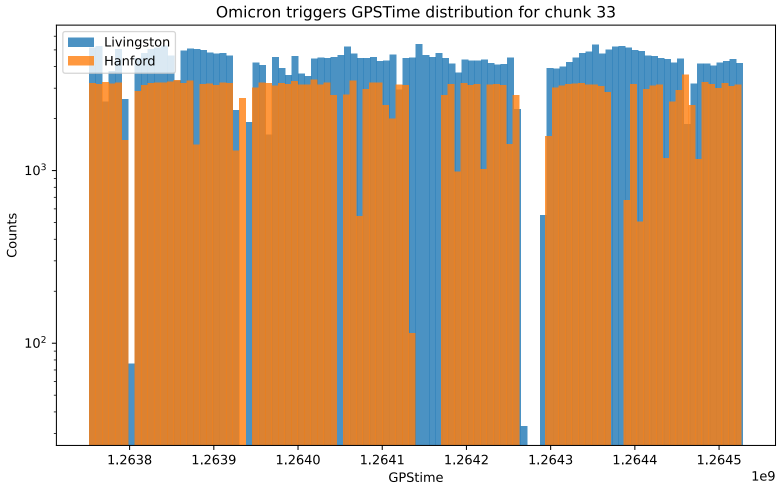


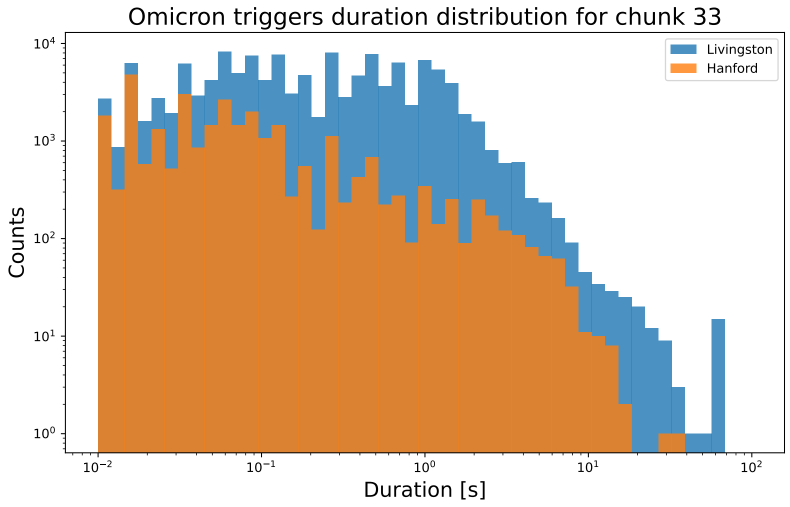
Omicron pipeline – Chunk 33

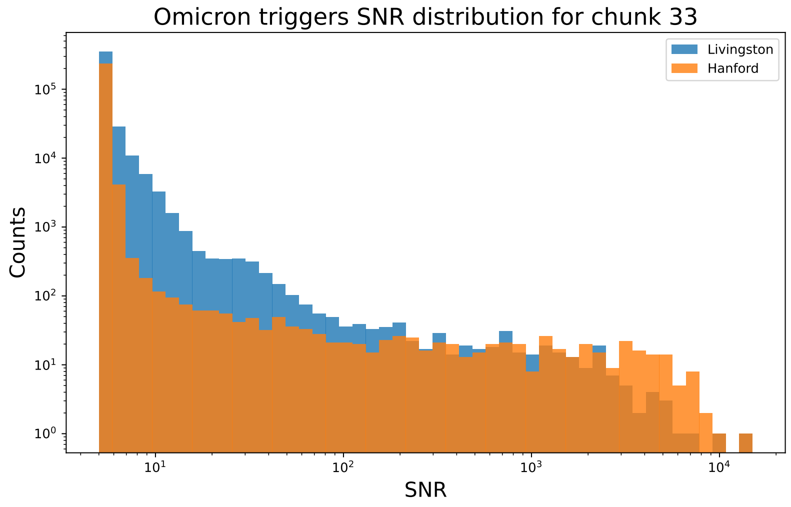
- Omicron is a glitch search pipeline used

as input to train iDQ

- We use SNR>5 triggers in h(t) with 0.1 clustering time
- Over 250k (400k) glitches in Hanford (Livingston)
- Sub-second glitches are dominant

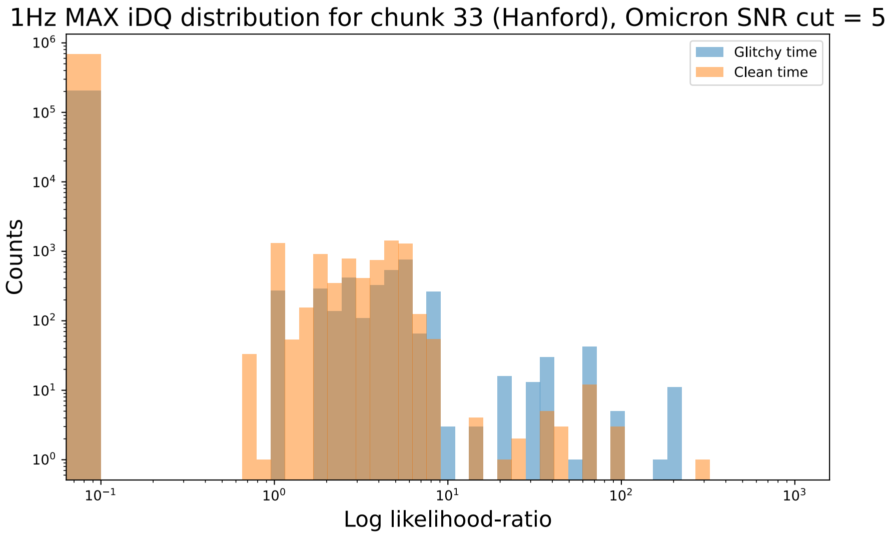


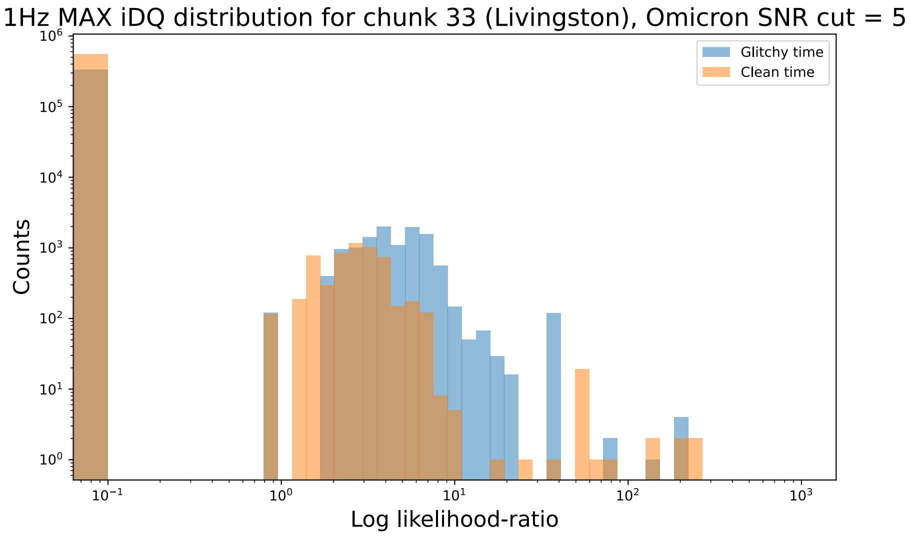




iDQ vs Omicron – Chunk 33

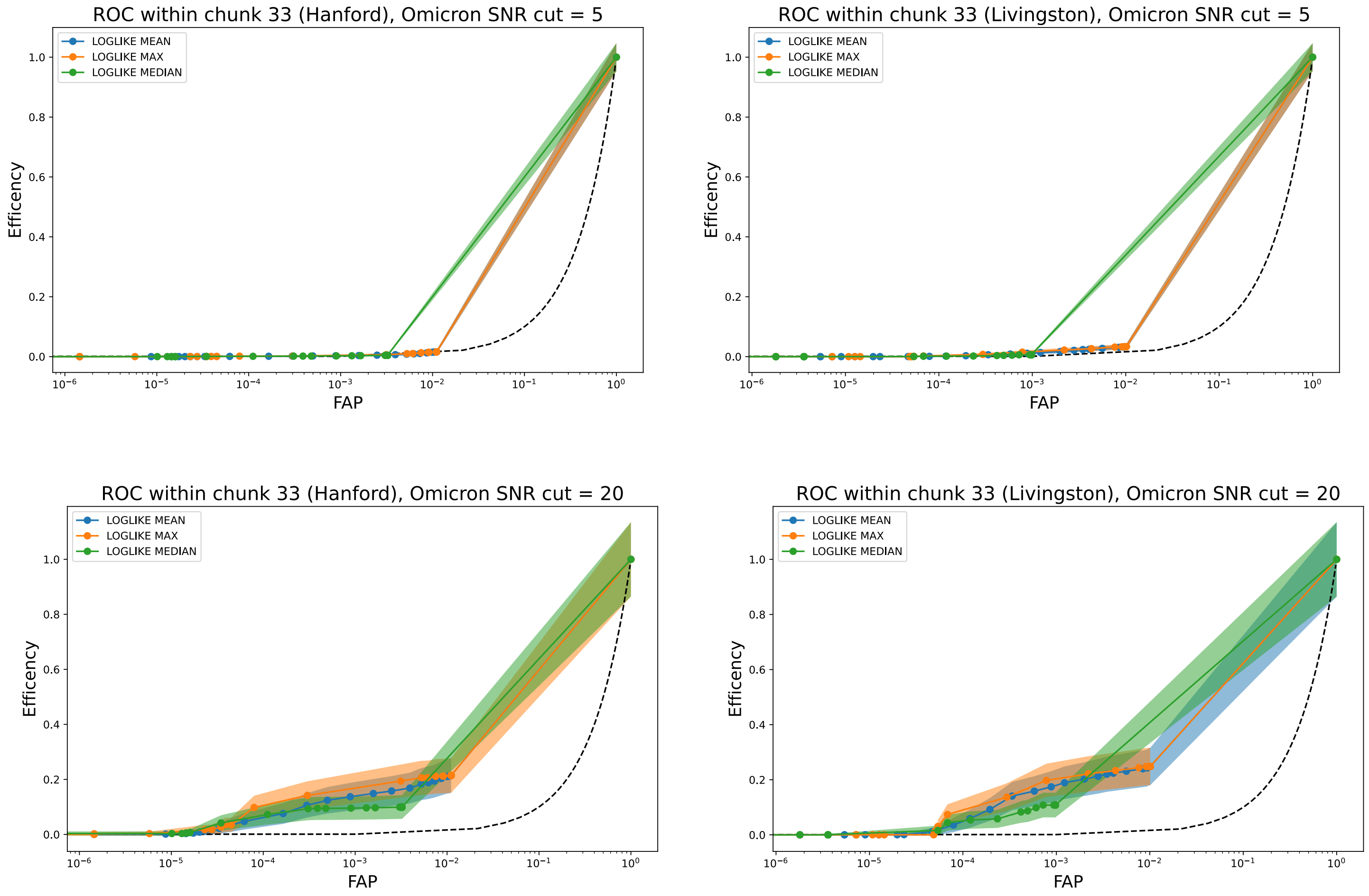
- Divide chunk 33 into two distinct timeseries (1Hz) : clean region (segments containing no Omicron triggers with SNR > 5), glitchy regions (segments containing at least a Omicron trigger with SNR > chosen SNR cut)
- Define several glitchy regions with different SNR cut
- For each segment we calculate Max/Mean/Median $\log(\mathcal{L})$





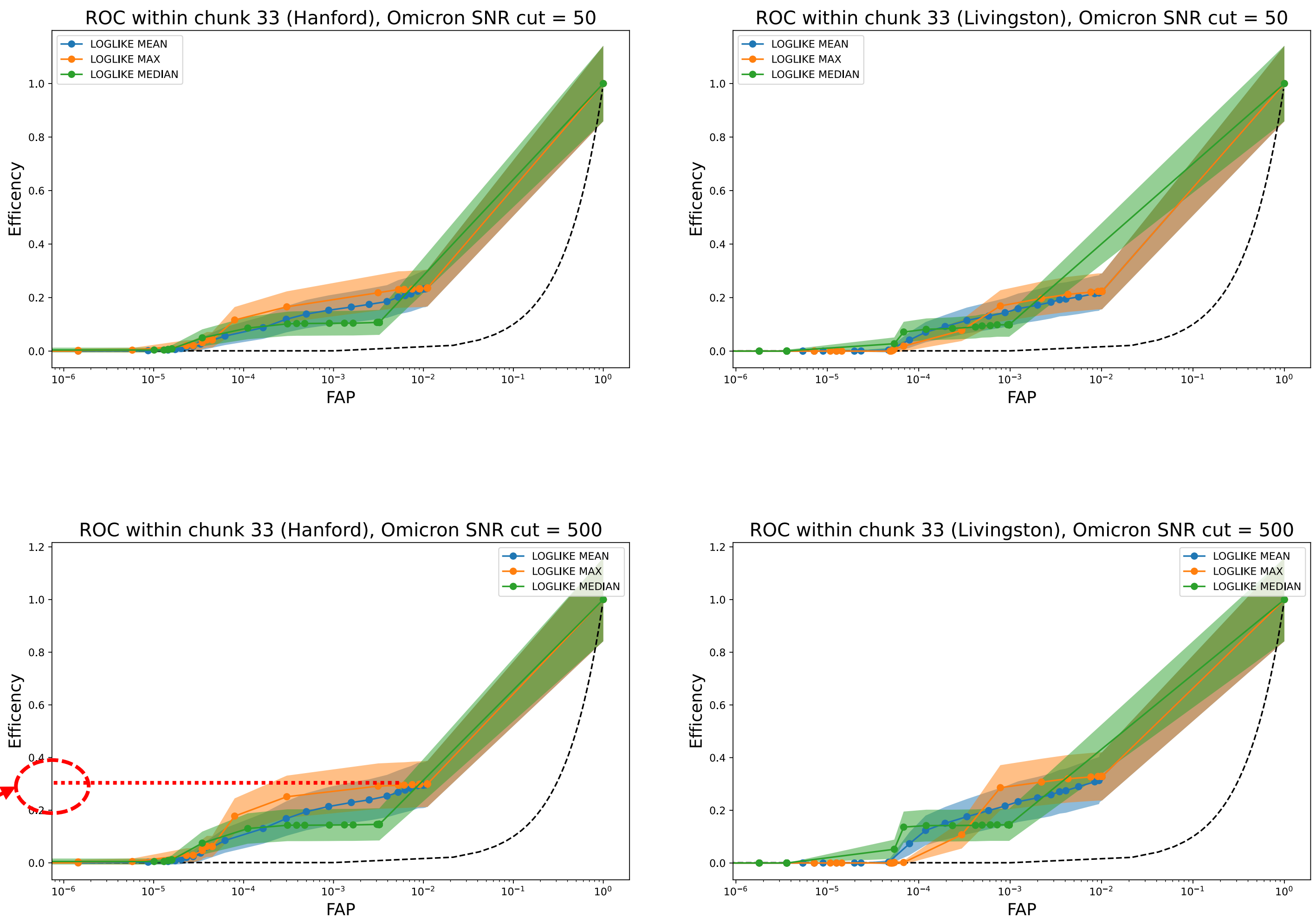
iDQ vs Omicron – Chunk 33

- ROC curves : Efficiency vs FAP
- **Efficiency** : fraction of glitchy samples removed by cut on $\log(\mathcal{L})$
- **FAP** : fraction of clean samples removed by cut on $\log(\mathcal{L})$
- SNR>5 poor efficiency/FAP
- Improvement when defined with SNR>10
- Using Maximum 1Hz $\log(\mathcal{L})$ more efficient



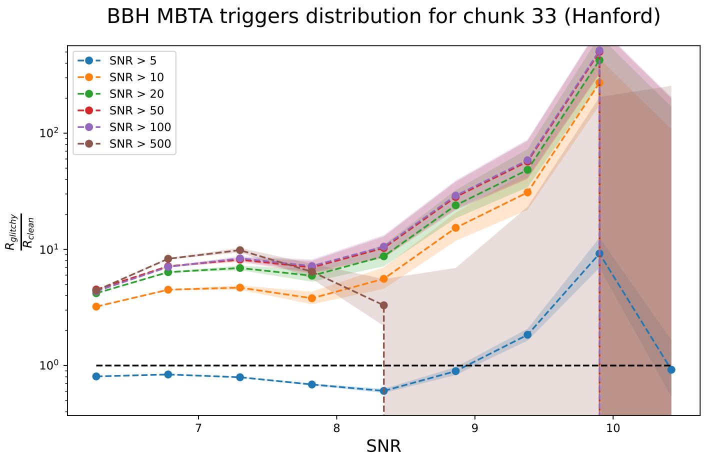
iDQ vs Omicron – Chunk 33

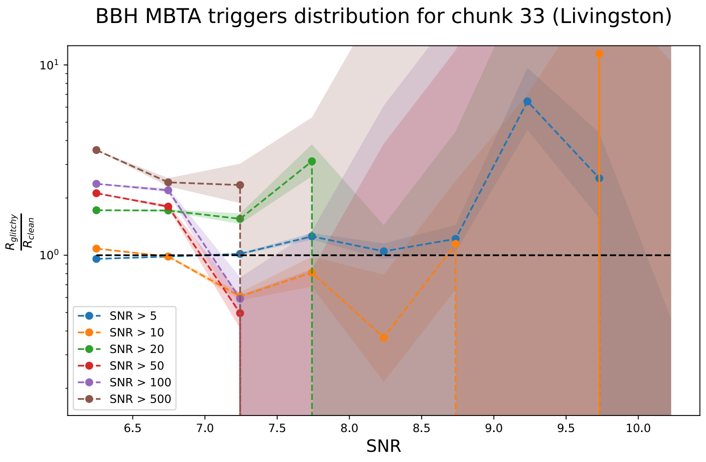
- For SNR>50 Omicron glitches, a cut at $\log(\mathcal{L}) = 5$:
  - 22% Eff vs 0.3% FAP in H for max $\log(\mathcal{L})$
  - 17% Eff vs 0.08% FAP in L for max $\log(\mathcal{L})$
- For SNR>500 Omicron glitches, a cut at $\log(\mathcal{L}) = 5$:
  - 29% Eff vs 0.3% FAP in H for max $\log(\mathcal{L})$
  - 29% Eff vs 0.08% FAP in L for max $\log(\mathcal{L})$
- Only a small fraction of triggers have $\log(\mathcal{L}) > 0$, meaning only this fraction can be rejected by iDQ

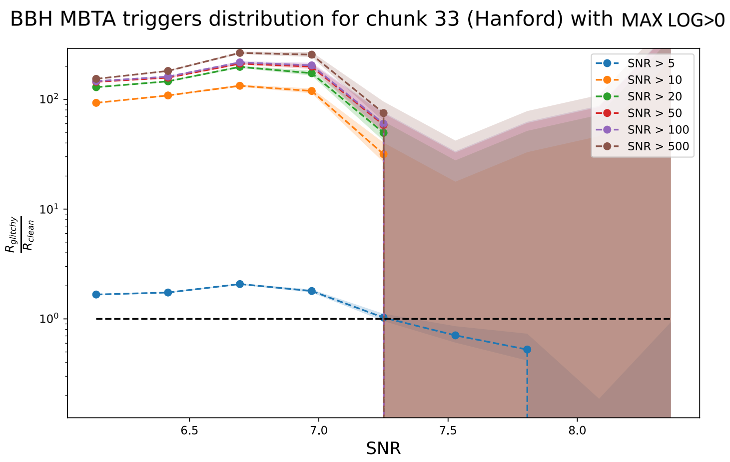


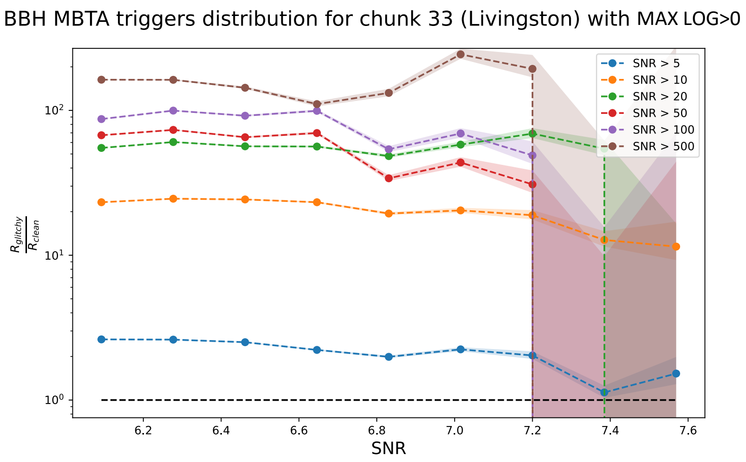
MBTA BBH triggers vs Omicron – Chunk 33

- We run MBTA to get single detector triggers in Hanford & Livingston
- We select only events with rwSNR > 6 (~200k triggers, enough stats)
- Separate the triggers into 2 populations:
  - Triggers in regions defined as glitchy by Omicron (at least an Omicron trigger with SNR > given SNR cut)
  - Triggers in clean region
- Calculate the MBTA triggers rate in both regions : $\frac{R_{glitchy}}{R_{clean}}$ in H > $\frac{R_{glitchy}}{R_{clean}}$ in L
- Higher cuts on Omicron SNR => ~ higher $\frac{R_{glitchy}}{R_{clean}}$
- No clear dependency on MBTA SNR



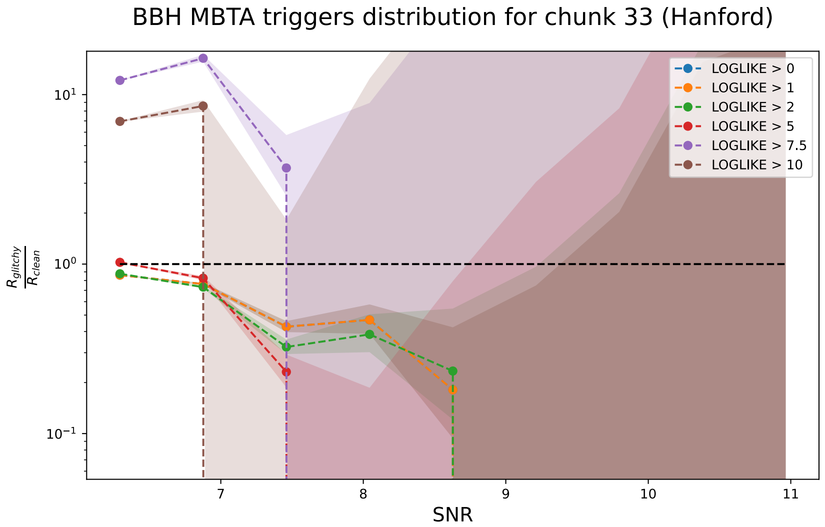


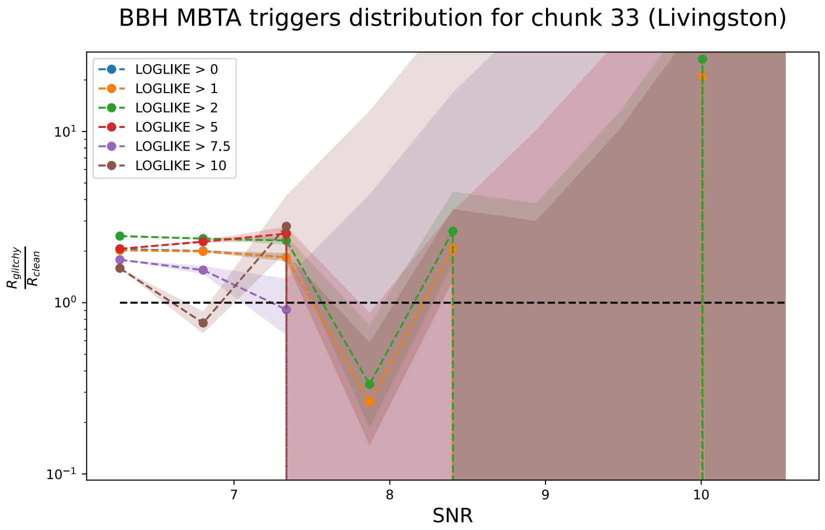




MBTA BBH triggers vs iDQ – Chunk 33

- We run MBTA to get single detector triggers in Hanford & Livingston
- We select only events with rwSNR > 6 (~200k triggers, enough stats)
- Separate the triggers into 2 populations:
  - Triggers in regions defined as glitchy by iDQ (samples with $\log(\mathcal{L})$ > given $\log(\mathcal{L})$ cut)
  - Triggers in clean region ($\log(\mathcal{L})$ = 0)
- Calculate the MBTA triggers rate in both regions
- No clear dependency on MBTA SNR
- Higher cuts on $\log(\mathcal{L})$ => ~ higher $\frac{R_{glitchy}}{R_{clean}}$





iDQ incorporation into MBTA

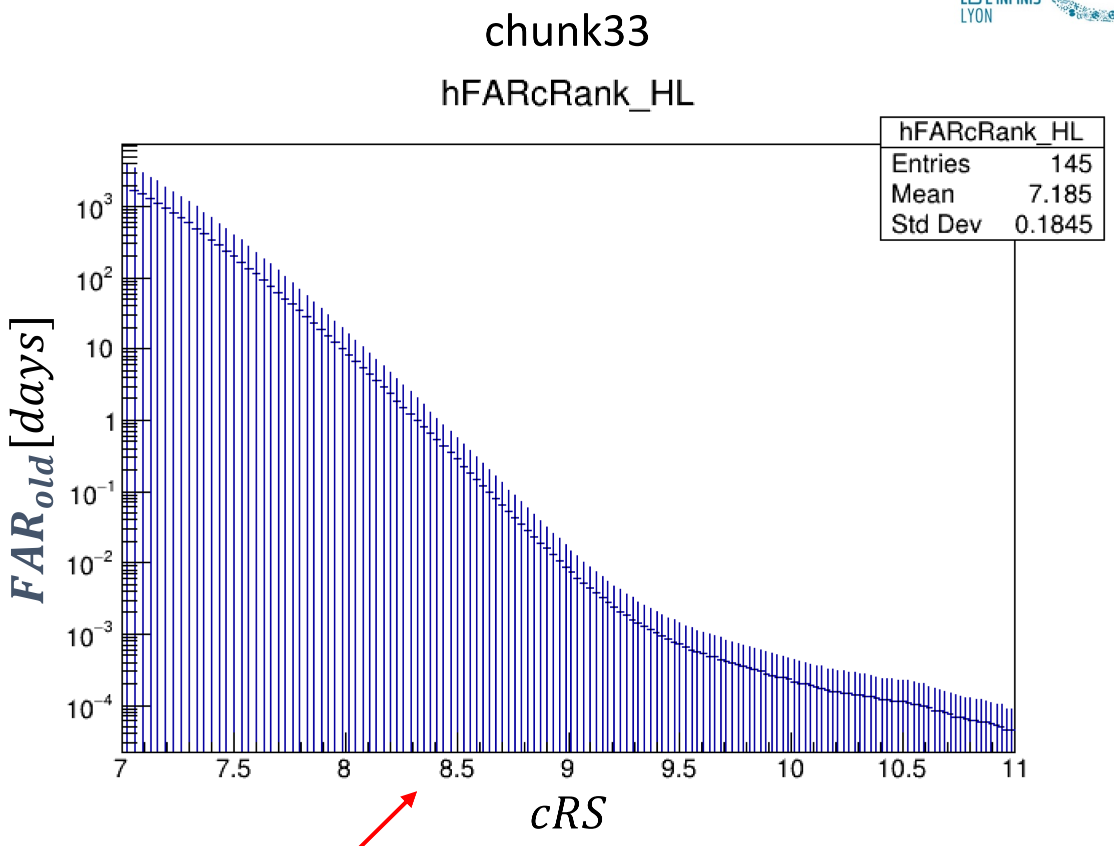
- Reweight single detector triggers SNR :

$$SNR_{iDQ} = \sqrt{SNR^2 - \alpha \log(\mathcal{L})}$$

- Run MBTA with injections, study efficiency & VT changes after reweighting SNR using iDQ
- iDQ useful if it globaly improves (reduces) FAR of injections
- Calculation of new FAR requires editing MBTA code (coming soon)
- Meanwhile : Approximative method to calculate new FAR of injections

$FAR_{iDQ}(\alpha)$ calcuation

- First MBTA run over chunk 33 without injections (noise only)
- $SNR_{iDQ} = \sqrt{SNR_{old}^2 - \alpha * \log(\mathcal{L})}$
- $R_H = \frac{N\ singles\ triggers\ in\ Hanf.with\ SNR_{iDQ}>6}{N\ singles\ triggers\ in\ Hanf.with\ SNR_{old}>6}$
- $R_L = \frac{N\ singles\ triggers\ in\ Liv.with\ SNR_{iDQ}>6}{N\ singles\ triggers\ in\ Liv.with\ SNR_{old}>6}$
- Second MBTA run over chunk 33 with injections



$$FAR_{iDQ} = FAR_{old} * R_H * R_L * \frac{FAR_{old}(SNR_{iDQ})}{FAR_{old}(SNR_{old})}$$

$FAR_{iDQ}(\alpha)$ calcuation – chunk33

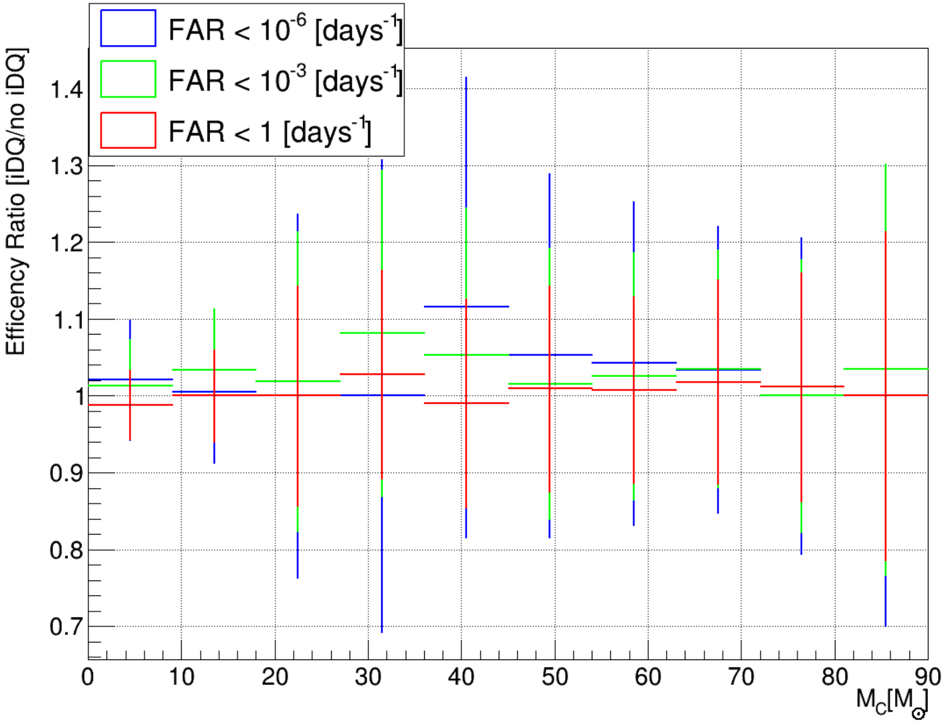
- α from 0 to 3 by step of 0.5
- $\forall\alpha$: max log(L) over template duration more efficient

$\Rightarrow$ α = 1.5 best choice so far

- $FAR < 10^{-3}\ [days^{-1}]$: 1218 injections recovered without iDQ
- $FAR < 10^{-3}\ [days^{-1}]$: 1248

$\Rightarrow$ +2.5% ~ +31 injections recovered

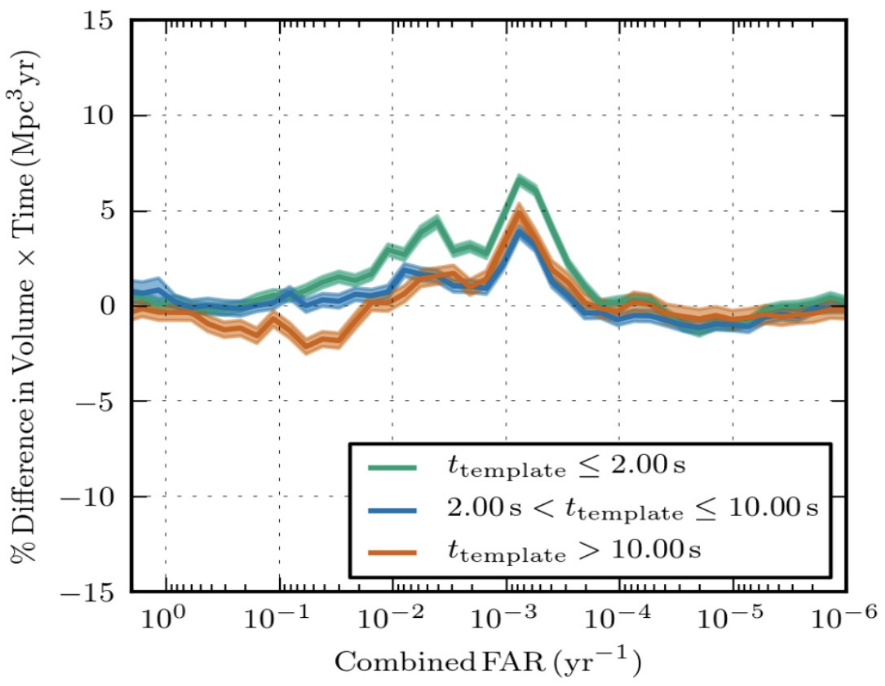
- Dependancy on chirp mass : more data needed



iDQ results in PyCBC & GstLAL

GstLAL

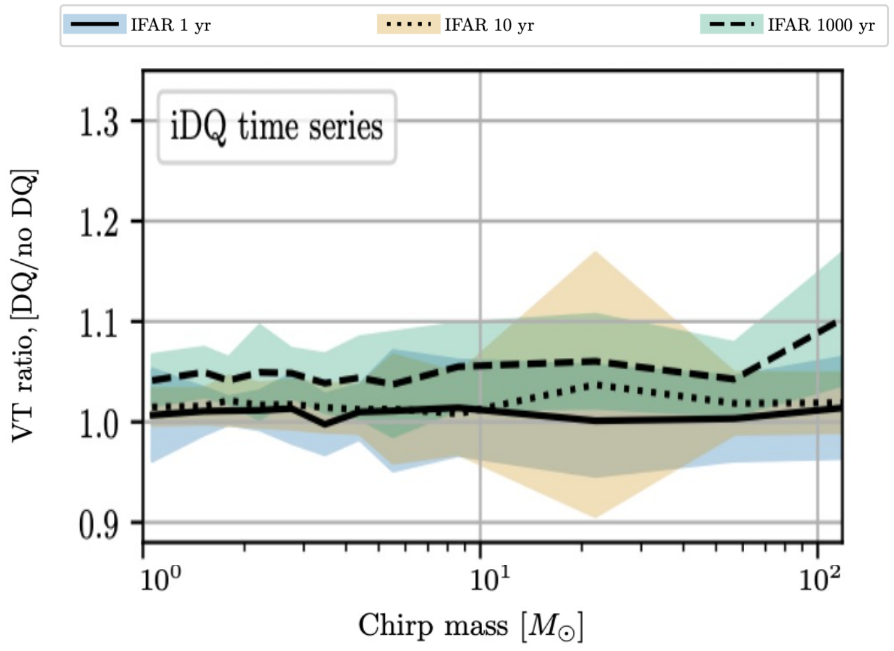
At most +6% in VT at $FAR{\sim}10^{-6}\ [days^{-1}]$



PyCBC

Average of +5% in VT at $FAR{\sim}\ 10^{-6}\ [days^{-1}]$

At most +10% for very high chirp mass



Conclusion

- Preliminary results for chunk 33 :
  - Using 1Hz Maximum trend of $\log(\mathcal{L})$ more efficient than Mean/Median
  - Large fraction of Omicron glitches are not flagged by iDQ
- Lower mass template can have large durations: consider iDQ on all template duration or some Δt around trigger time ?
  - Maximum $\log(\mathcal{L})$ over long durations could be overestimated → falsely downgrading events
  - Should iDQ be used for short BBH only ? BHNS/BNS events ?
- Reweight single detector triggers SNR : $SNR_{iDQ} = \sqrt{SNR^2 - \alpha \log(\mathcal{L})}$
- Approximative new FAR calculation : $\alpha = 1.5$ & using max $\log(\mathcal{L})$ more efficient
- Reach +2.5% of efficiency at FAR = $10^{-3}\ [days^{-1}]$
- Need to run MBTA once again after implementing iDQ reweighted SNR