LSU

Department of Physics & Astronomy

Gamma-Ray Burst Waterfalls and Machine Learning

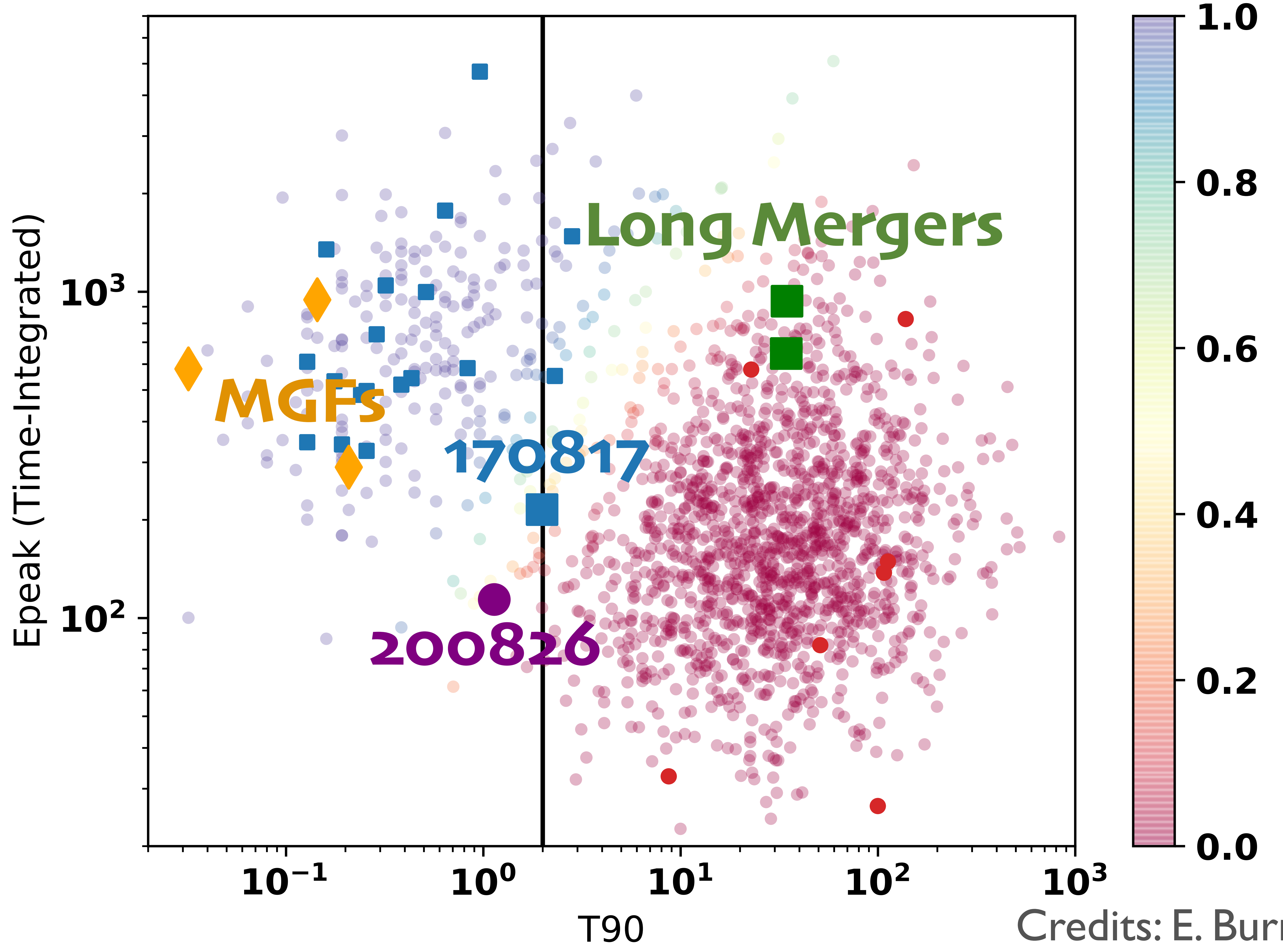
Michela Negro (michelanegro@lsu.edu) - Louisiana State University

on behalf of the team: N. Cibrario, E. Burns, J. Wood, A. Goldstein, T. dal Canton

Dec 2-6 2024: A WORKSHOP ON GRBS AND CENTRAL ENGINE POWERED TRANSIENTS

Gamma-ray bursts prompt classification

Epeak = the energy at which the photon energy distribution peaks (spectral hardness)
T90 = time required to for from 5% to 95% of the total burst counts (duration)



Credits: E. Burns

M. Negro - michelanegro@lsu.edu

Known Gamma-ray burst classes

Galama et al. (1998), Abbott et al (2017), Burns et al. (2021), Mereghetti et al. (2023)
Mochkovitch et al. (1993), Cano et al. (2017), Levan et al. (2013).

Key discoveries include:

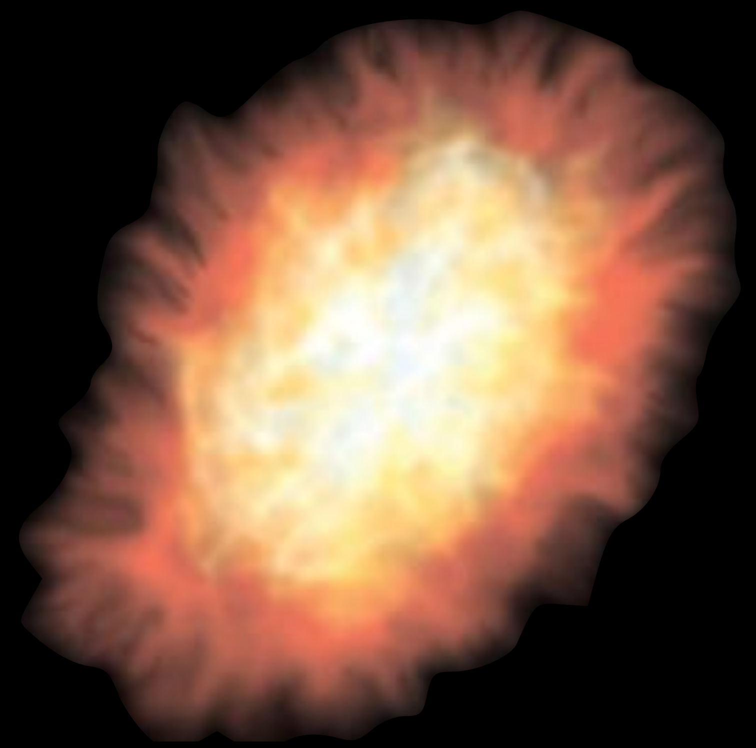
- identification of a collapsar origin for long GRBs (type Ic broad-lined supernova)
- gravitational-wave proof of short GRBs arising from binary neutron star (BNS) mergers
- unambiguous evidence of short GRBs from extragalactic magnetar giant flares (MGFs)

Additionally:

- Neutron star-black hole (NSBH) mergers are likely to be a third short GRB progenitor.
- Recently a long merger have been observed (Levan et al. 2023, Troja et al 2023)
- Long GRBs come in different flavors: low-luminosity GRBs, X-Ray Flashes, ultra-long GRBs

Ultimate goal:

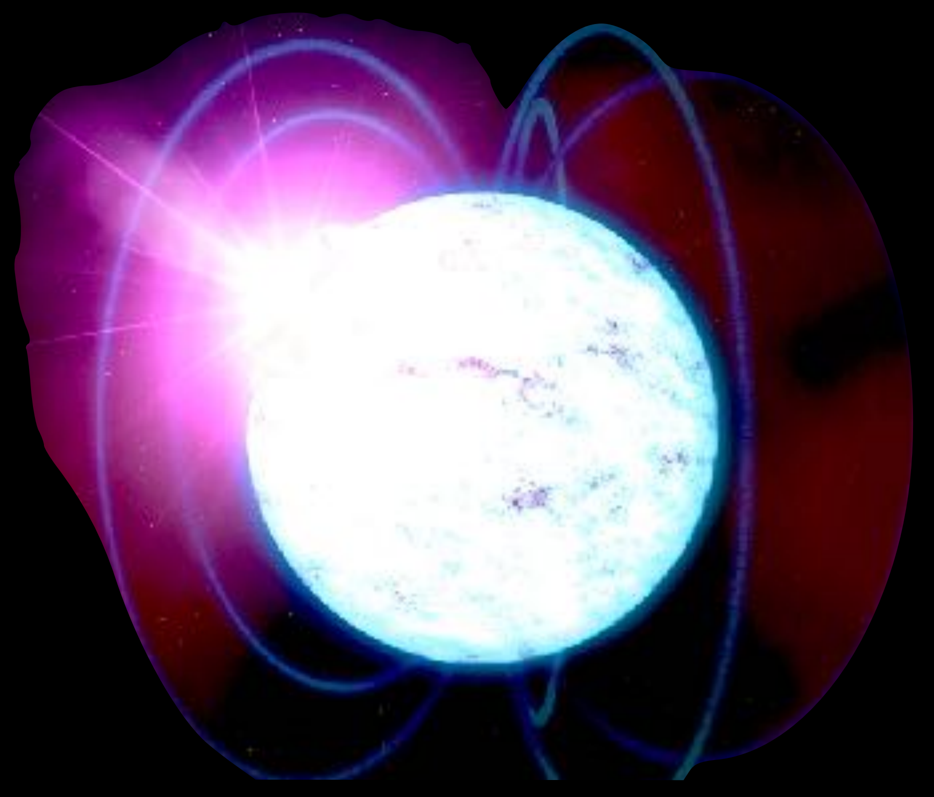
rapid identification of the progenitor of a given event, allowing for specific follow-up observations to occur,



Collapsars

Neutron stars binary

merger



Magnetar giant flare

M. Negro - michelanegro@lsu.edu

Overview of related works

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. Jul 2023;
Chen et al. Nov 2023;
Negro et al 5 June 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. 2024;

M. Negro - michelanegro@lsu.edu

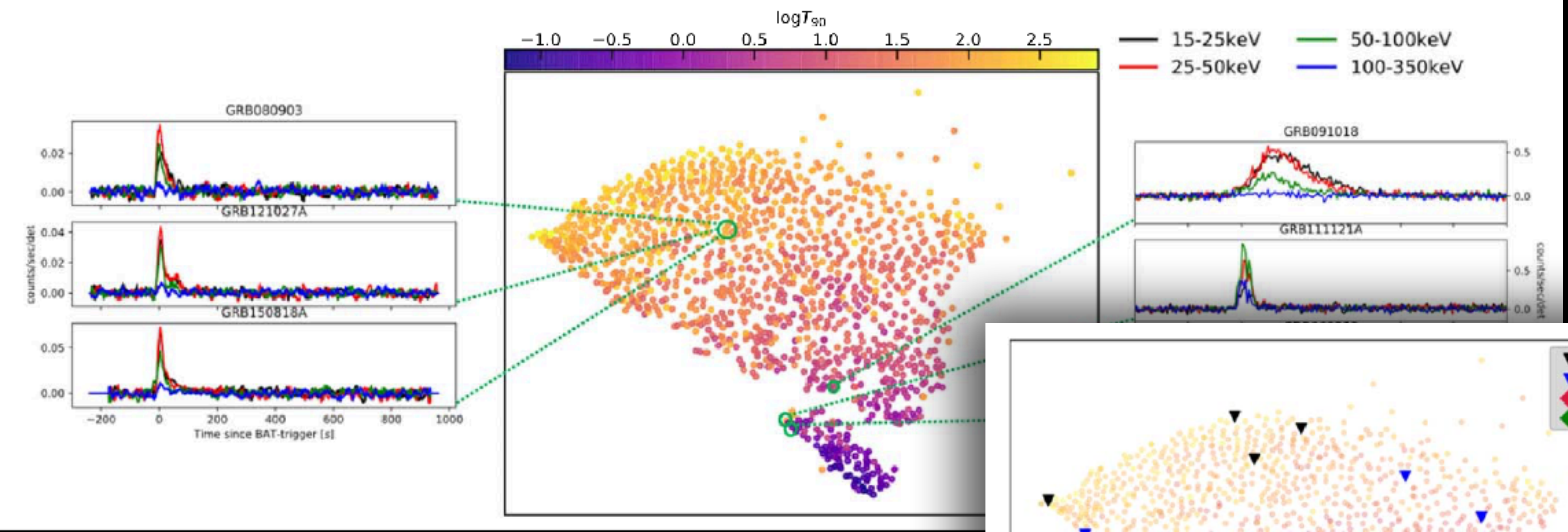
Overview of related works

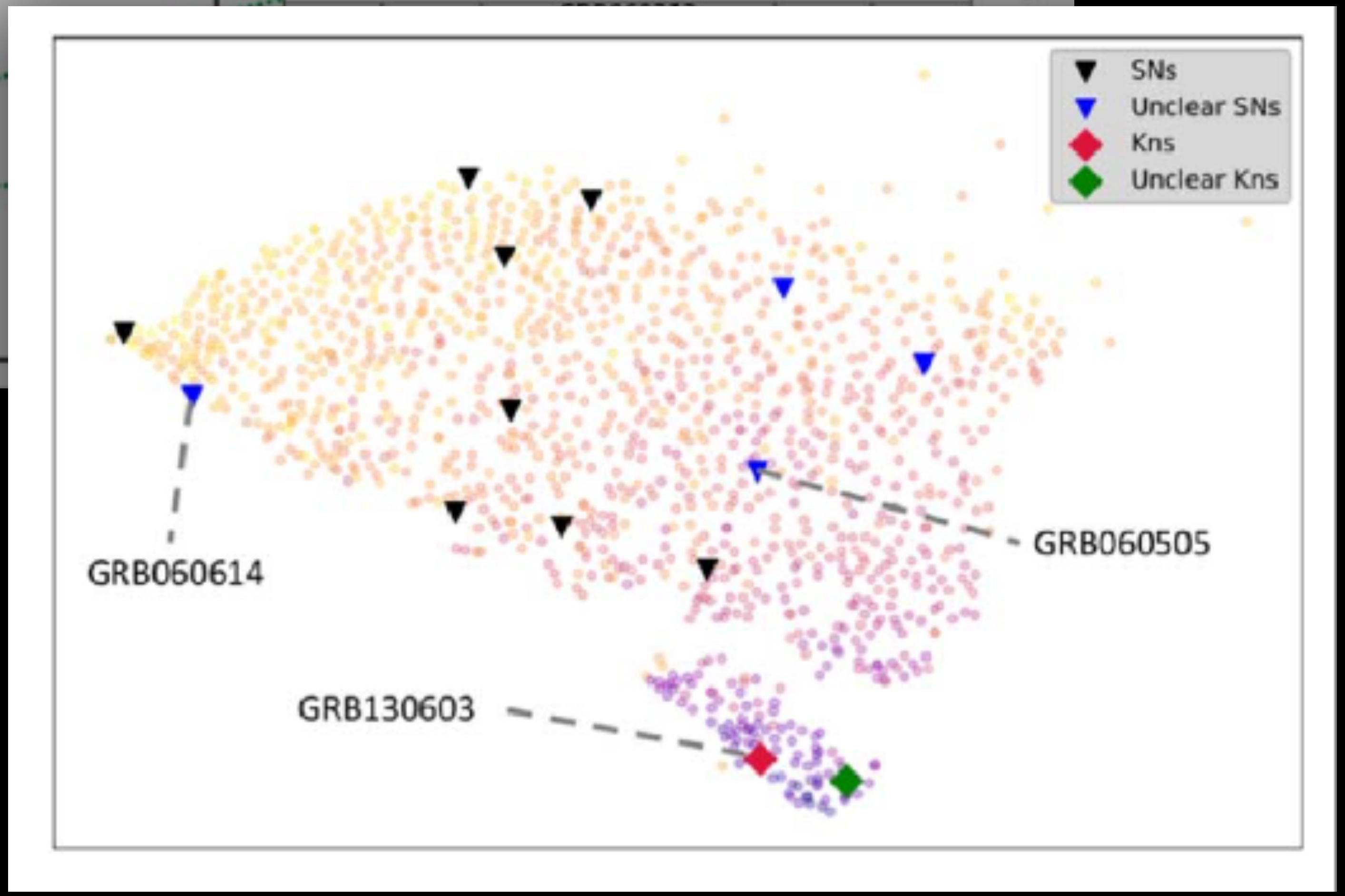
Methods: Applying t-SNE to Swift/BAT Light Curves

Input size:
~1500 × 30,000

Swift GRBs separated based on discrete-time fourier transform of prompt light curves

64 ms binned light curve in each band (limited to the interval out to T_{100} then zero-padded)





t-SNE = t-Distributed Stochastic Neighbor Embedding

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. Jul 2023;
Chen et al. Nov 2023;
Negro et al 5 June 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. 2024;
Nuessle et al. 2024

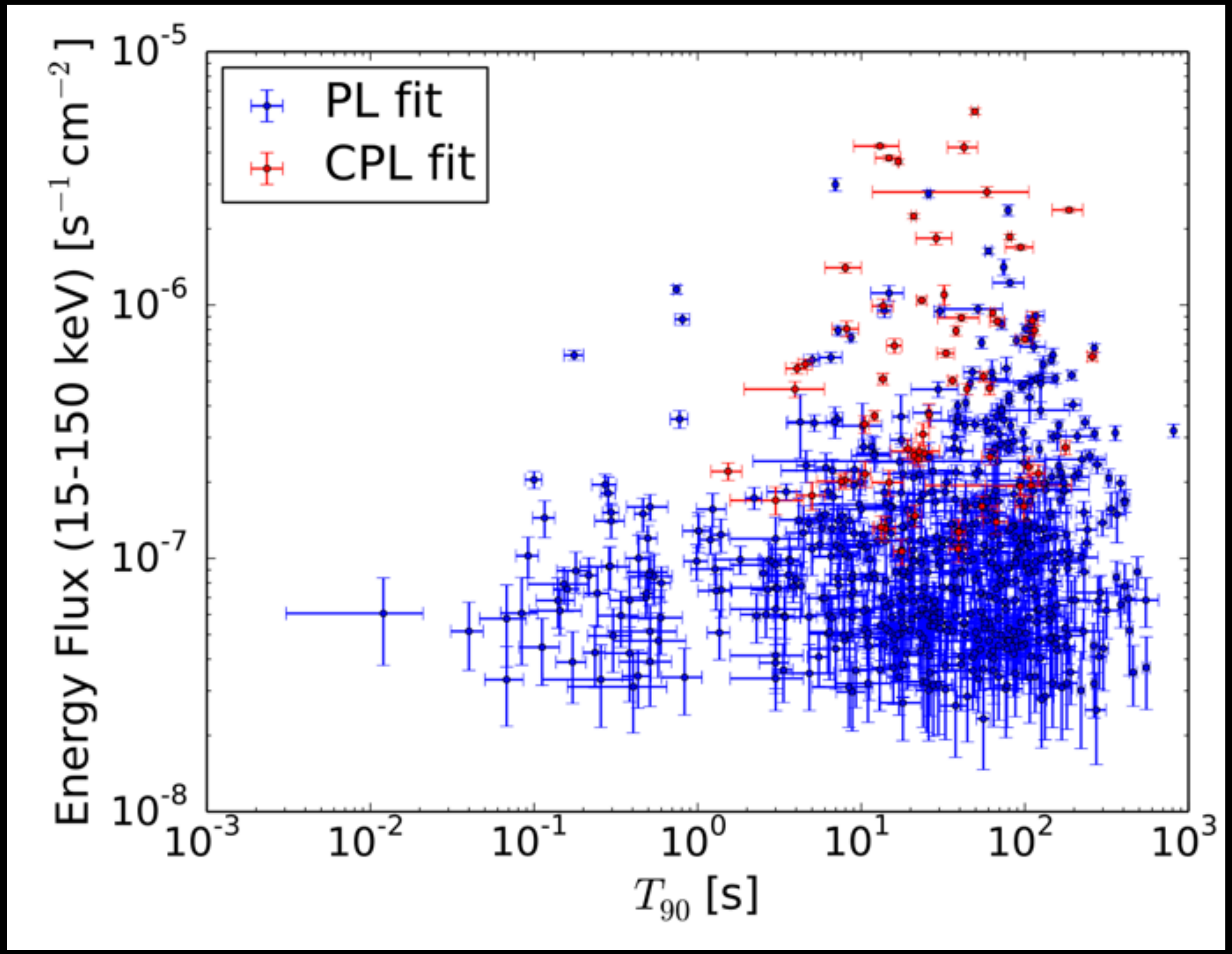
Test sample is small

GW170817 not a Swift burst

M. Negro - michelanegro@lsu.edu

Maybe an expected feature?

Lien et al 2016: The Third Swift Burst Alert Telescope Gamma-Ray Burst Catalog



1-s peak energy flux (15-150 keV) vs. T90

In Jespersen et al 2020 (and others):

"Individual light curves are normalized by the total fluence, obtained as the numerical integral of the flux across all bands"

M. Negro - michelanegro@lsu.edu

Overview of related works

Methods: Applying t-SNE and UMAP to Fermi GBM spectral parameters

Input size:
2297 × 5

2297 GBM GRBs using both time-integrated and peak-time fluxes and fluencies

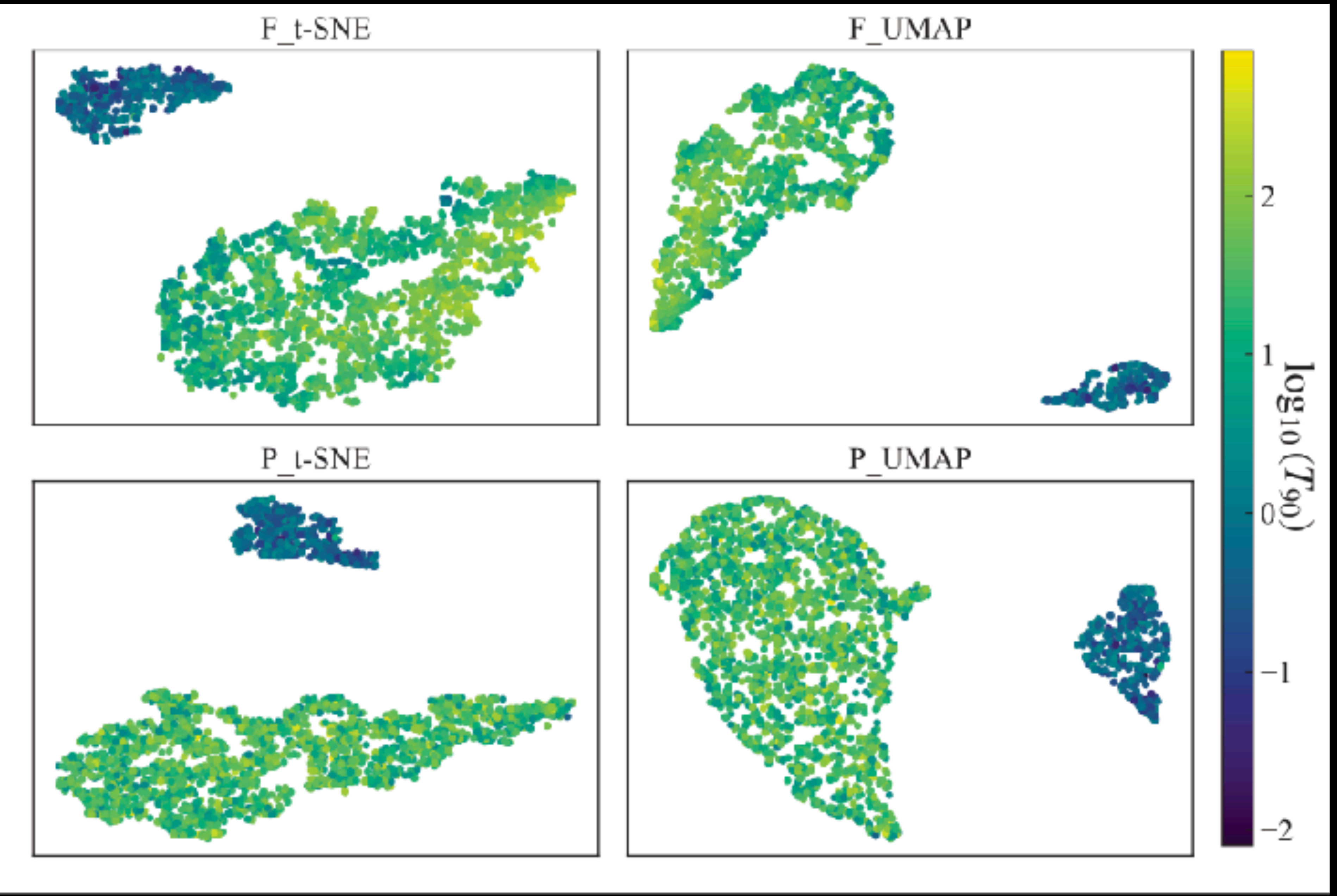
t-SNE and UMAP hyperparameters scanned and chosen to maximize the clustering

Clustering: k-means (k = 2)

UMAP = Uniform Manifold Approximation and Projection

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. Jul 2023;
Chen et al. Nov 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

Feature	Unit	Description
	Peak-flux	
$\log_{10}(E_{\mathrm{p\_P}})$	keV	Spectral peak energy of the Band model
$\log_{10}(f_{\mathrm{e\_P}})$	erg cm^{-2} s^{-1}	The energy flux of the Band model (10–1000 keV)
$\log_{10}(f_{\mathrm{p\_P}})$	photon cm^{-2} s^{-1}	The photon flux of the Band model (10–1000 keV)
$\log_{10}(F_{\mathrm{e\_P}})$	erg cm^{-2}	The energy fluence of the Band model (10–1000 keV)
$\log_{10}(F_{\mathrm{p\_P}})$	photon cm^{-2}	The photon fluence of the Band model (10–1000 keV)
	Time-integrated	
$\log_{10}(E_{\mathrm{p\_F}})$	keV	Spectral peak energy of the Band model
$\log_{10}(f_{\mathrm{p\_F}})$	photon cm^{-2} s^{-1}	The photon flux of the Band model (10–1000 keV)
$\log_{10}(f_{\mathrm{e\_F}})$	erg cm^{-2} s^{-1}	The energy flux of the Band model (10–1000 keV)
$\log_{10}(F_{\mathrm{e\_F}})$	erg cm^{-2}	The energy fluence of the Band model (10–1000 keV)
$\log_{10}(F_{\mathrm{p\_F}})$	photon cm^{-2}	The photon fluence of the Band model (10–1000 keV)



M. Negro - michelanegro@lsu.edu

Overview of related works

Methods: Applying ConvAE + UMAP to FermiGBM waterfalls

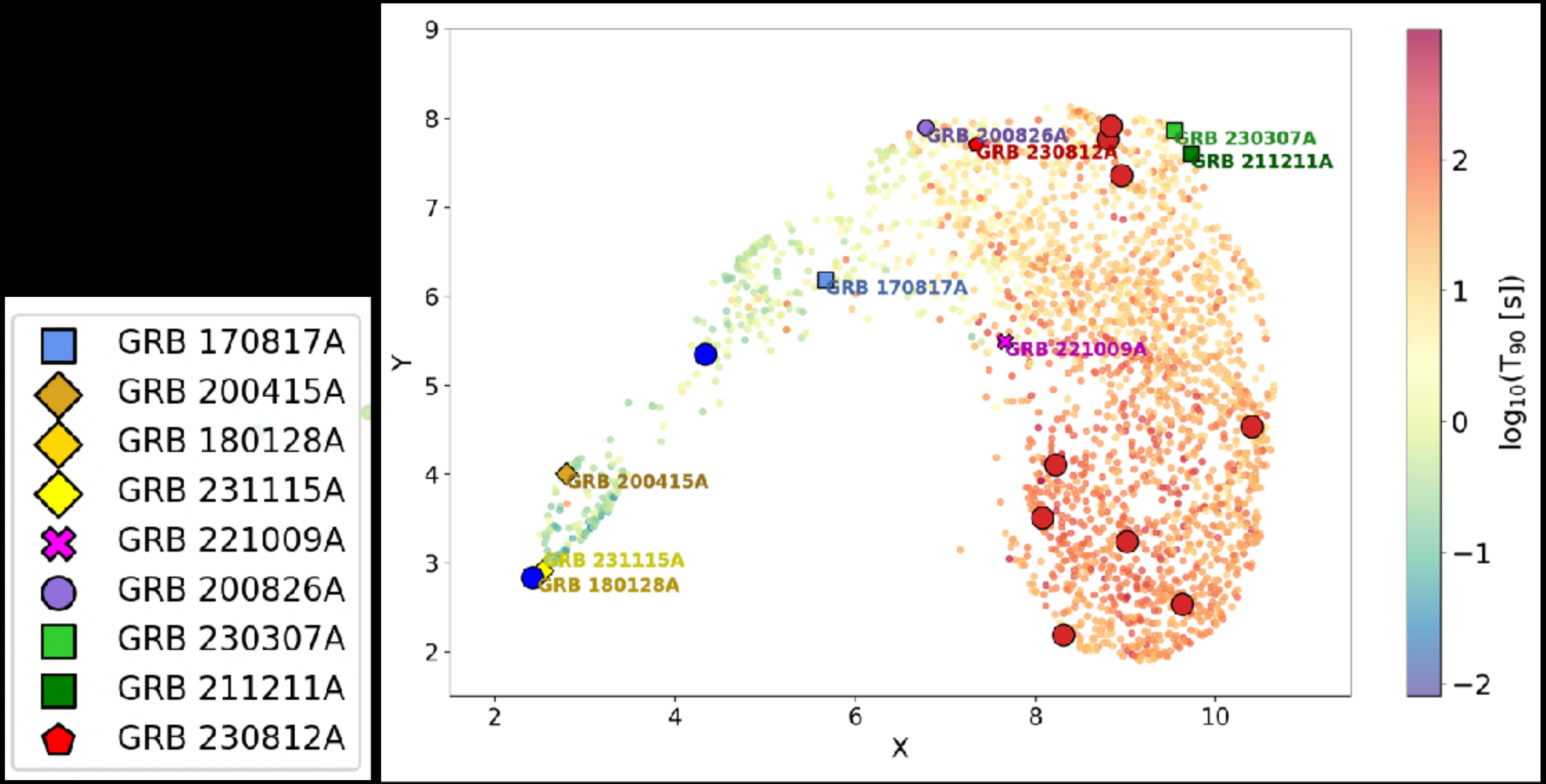
Input size:
see slide 13

2361 Fermi/GBM GRBs + 151 **test**

ConvAE architecture and UMAP hyperparameters chose a priory according to sample characteristics and expectations on characteristics of astrophysical progenitor classes

Clustering: semi-supervised label propagation on trusted events in the embedding

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. July 2023;
Chen et al. Nov 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;



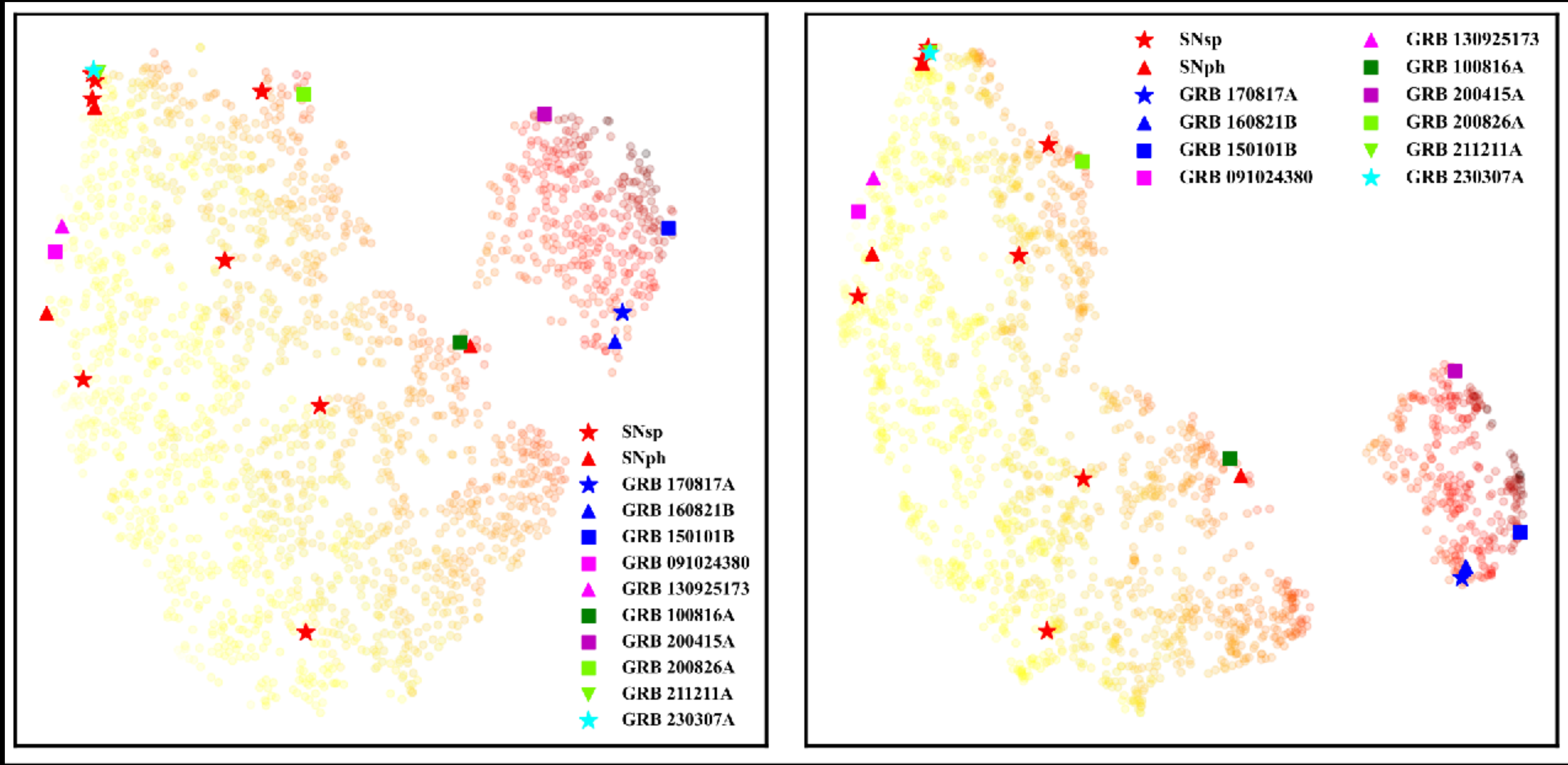
M. Negro - michelanegro@lsu.edu

Overview of related works

Methods: Applying t-SNE and UMAP on observed quantities

Input size:
3349 x 81798

2061 Fermi/GBM GRBs from fermi catalog:
duration (T90), peak energy (Ep), peak flux (Fp) and fluence (Sγ)



Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Chen et al. 2023;
Garcia-Cifuentes et al. 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

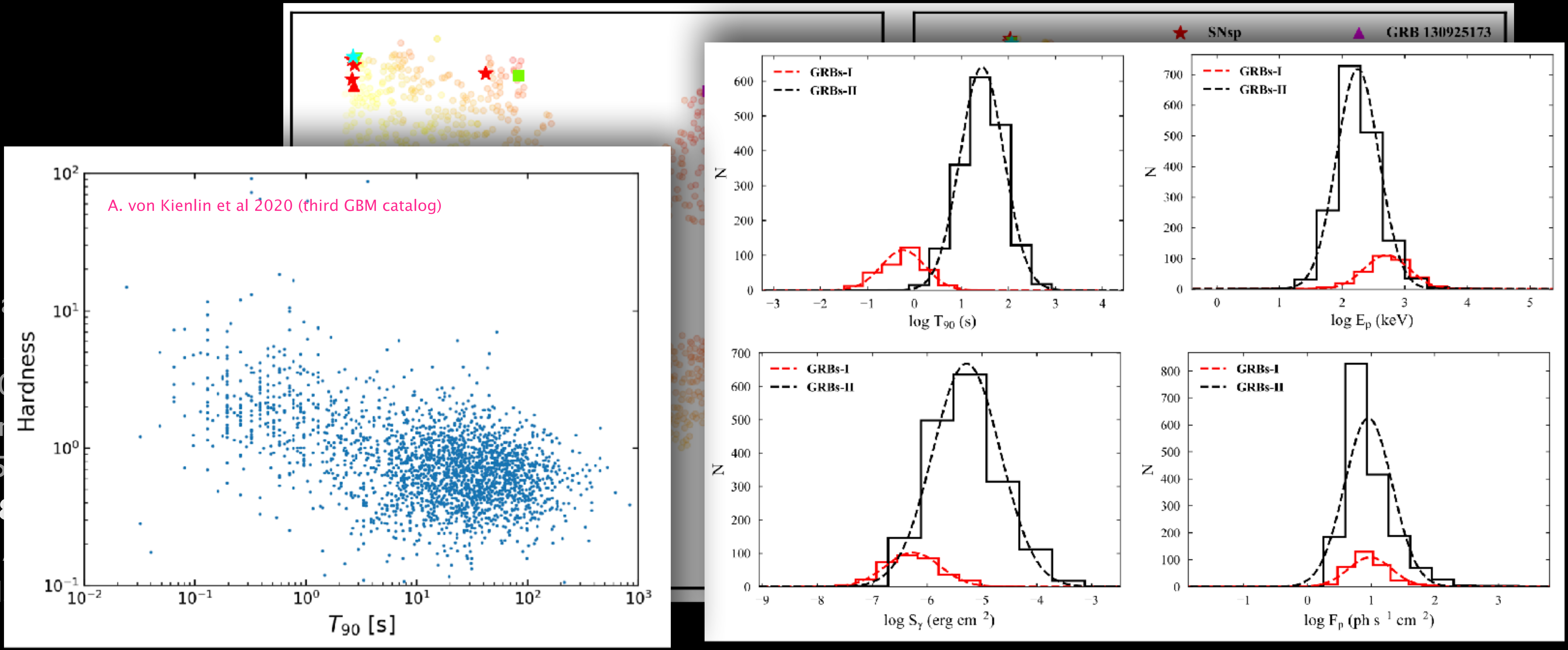
M. Negro - michelanegro@lsu.edu

Overview of related works

Methods: Applying t-SNE and UMAP on observed quantities

Input size:
3349 x 81798

2061 Fermi/GBM GRBs from fermi catalog:
duration (T90), peak energy (Ep), peak flux (Fp) and fluence (Sγ)



Jespersen et a
Dimple et al.
Chen et al. 20
Garcia-Cifuer
Negro et al 5
Zhu et al. 8
Dimple et al.
Nuessle et al

M. Negro - michelanegro@lsu.edu

Overview of related works

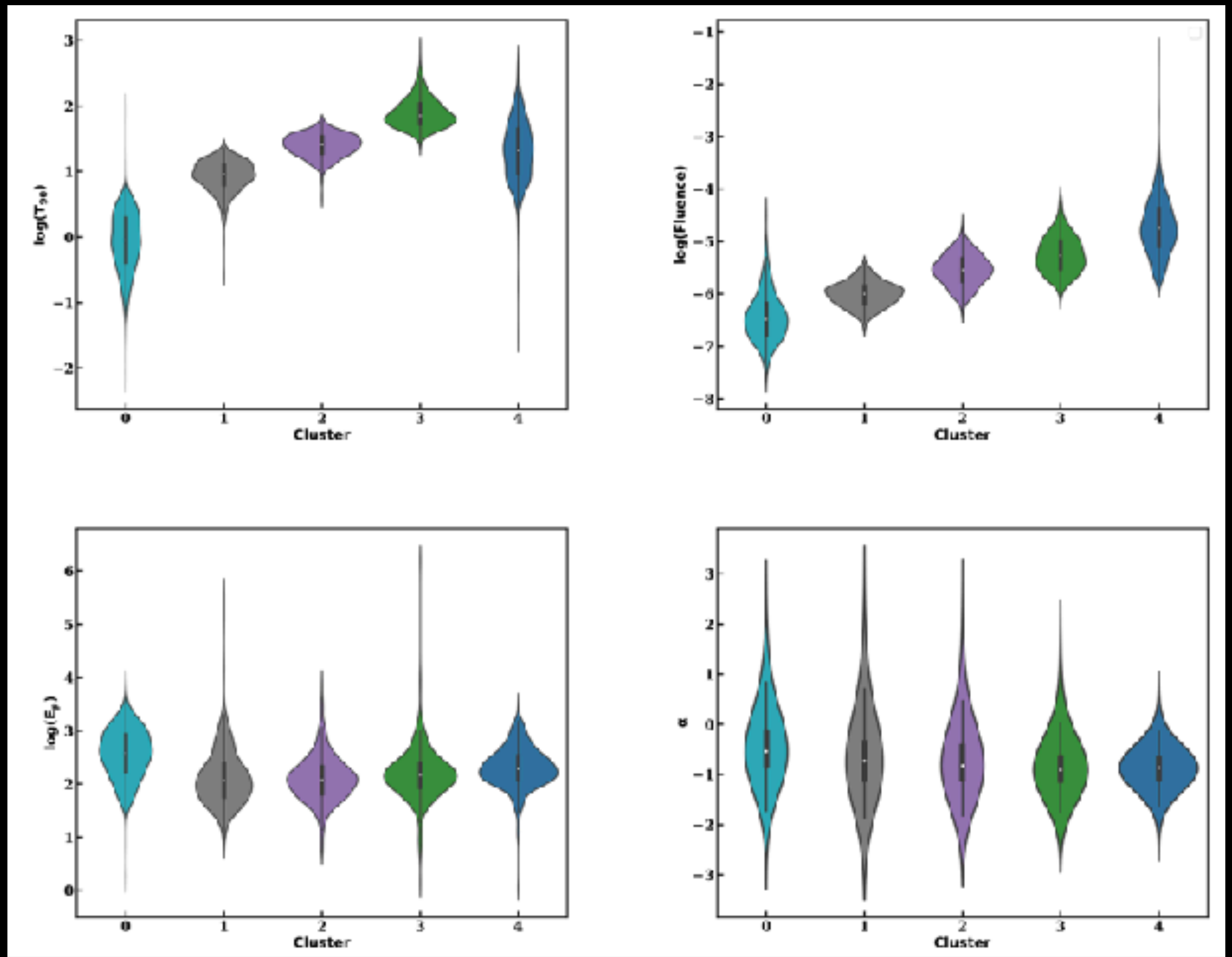
Methods: Applying UMAP to Fermi/GBM Light Curves

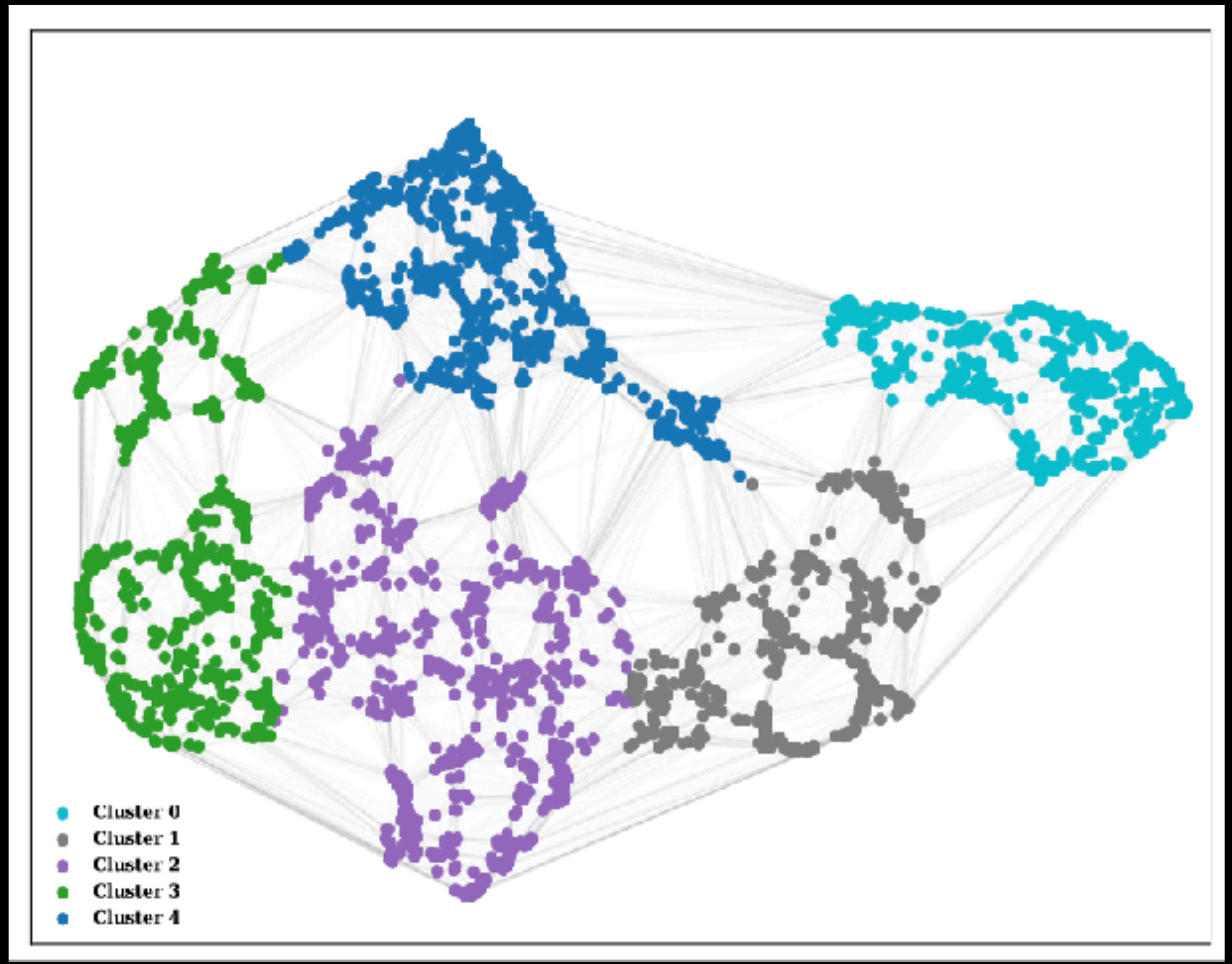
Input size:
3349 x 81798

3349 Fermi/GBM GRBs (similar data prep as Jespersen et al 2020, Dimple et al 2023)

t-SNE and UMAP hyperparameters scanned and chosen to maximize the clustering

Clustering: gaussian mixture models (AutoGMM): 5 clusters found





Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Chen et al. 2023;
Garcia-Cifuentes et al. 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

M. Negro - michelanegro@lsu.edu

Overview of related works

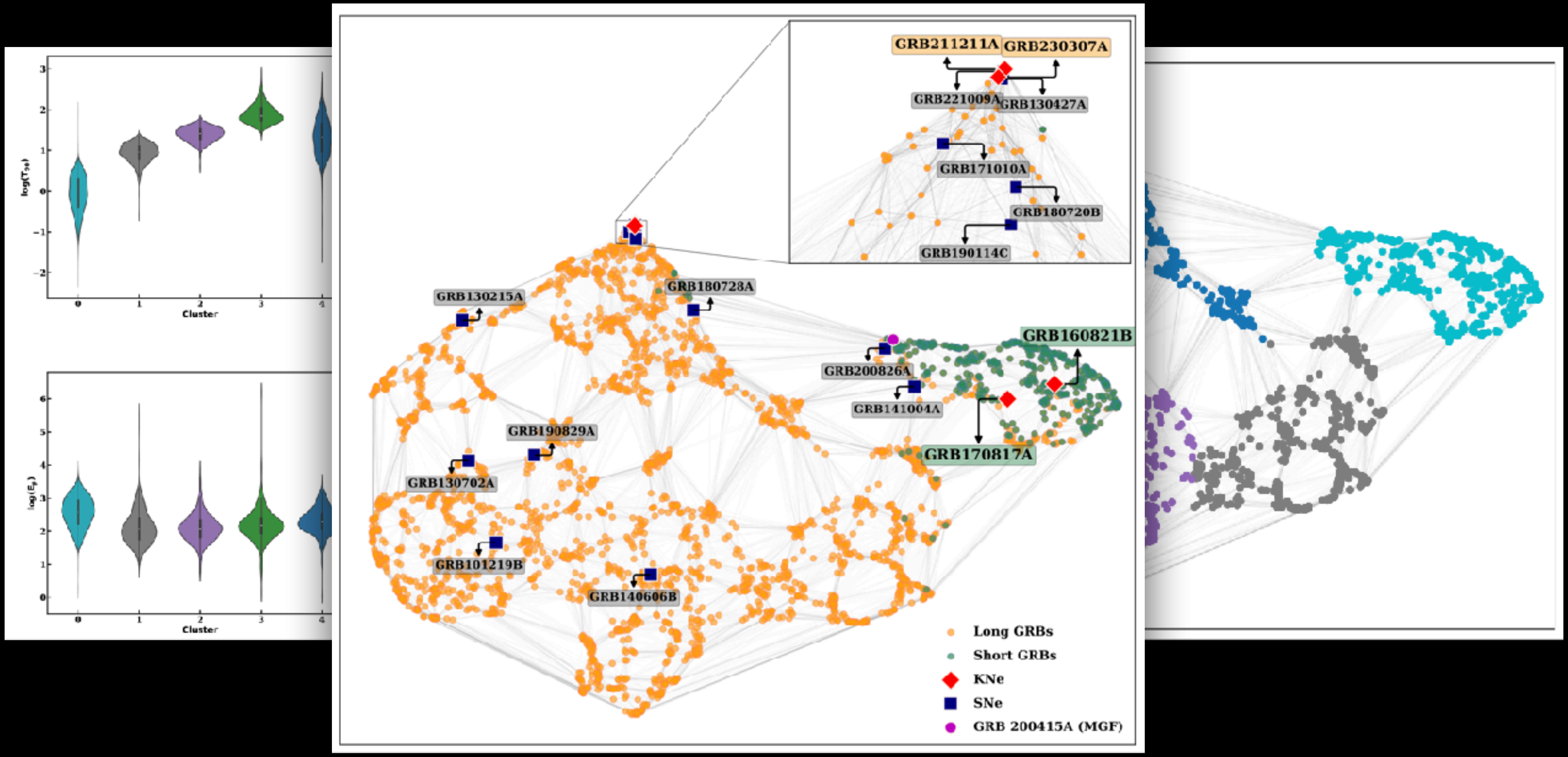
Methods: Applying UMAP to Fermi/GBM Light Curves

Input size:
3349 x 81798

3349 Fermi/GBM GRBs (similar data prep as Jespersen et al 2020, Dimple et al 2023)

t-SNE and UMAP hyperparameters scanned and chosen to maximize the clustering

Clustering: gaussian mixture method (AutoGMM): 5 clusters found



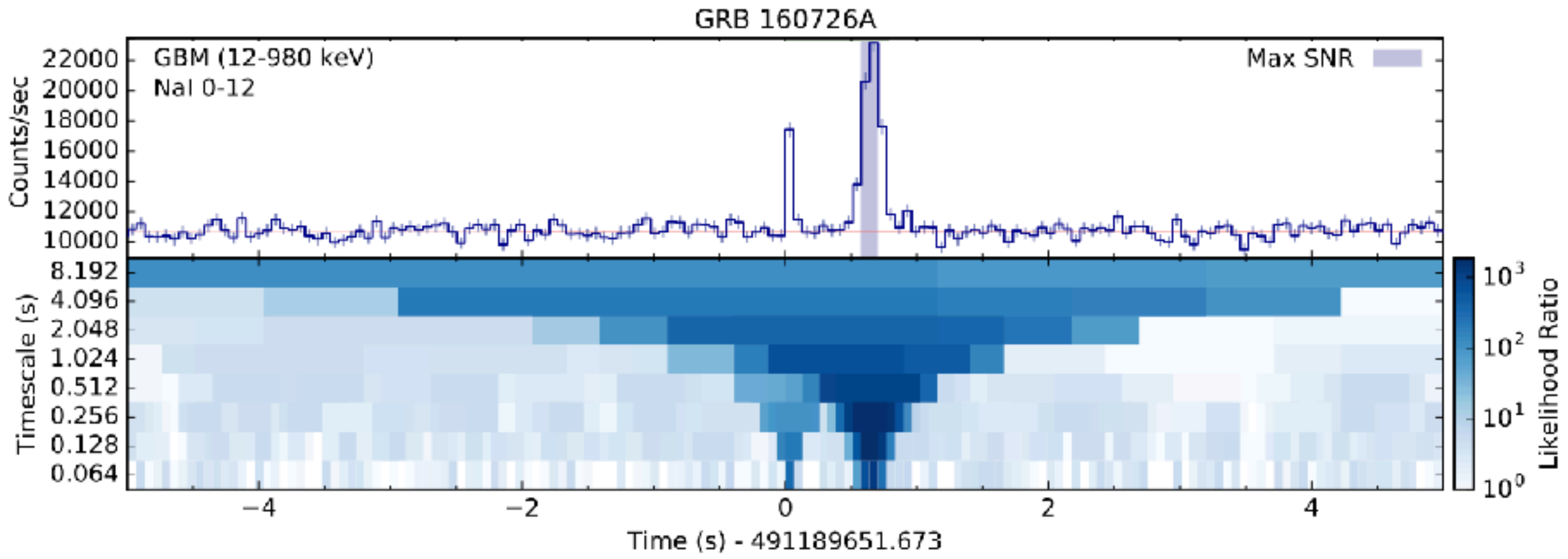
Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Chen et al. 2023;
Garcia-Cifuentes et al. 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

M. Negro - michelanegro@lsu.edu

GRB waterfalls

Blackburn et al. (2015), Goldstein et al. (2016), Kocevski et al. (2018)

We make use of the sub-threshold search data analysis developed but the GBM Team.
We extend the timescales down to 2 microseconds (64 ms default)



Probability of observed counts being from a source of amplitude s:

$$P(d|H_1) = \prod_i \frac{1}{\sqrt{2\pi}\,\sigma_{d_i}} \exp\left(-\frac{(\widetilde{d_i} - r_i s)^2}{2\sigma_{d_i}^2}\right) \qquad \widetilde{d_i} = d_i - \langle n_i \rangle$$

Probability of observed counts being from background (s=0)

$$P(d|H_0) = \prod_i \frac{1}{\sqrt{2\pi}\,\sigma_{n_i}} \exp\left(-\frac{\widetilde{d_i}^2}{2\sigma_{n_i}^2}\right)$$

Likelihood Ratio

$$\mathcal{L} = \ln \frac{P(d|H_1)}{P(d|H_0)} = \sum_i \left[\ln \frac{\sigma_{n_i}}{\sigma_{d_i}} + \frac{\widetilde{d_i}^2}{2\sigma_{n_i}^2} - \frac{(\widetilde{d_i} - r_i s)^2}{2\sigma_{d_i}^2}\right]$$

Different spectral shapes representative of typical GRB spectra:

- Hard / Normal / Soft / Blackbody

Different minimum values of Likelihood Ratio (MinVal) for background rejection:

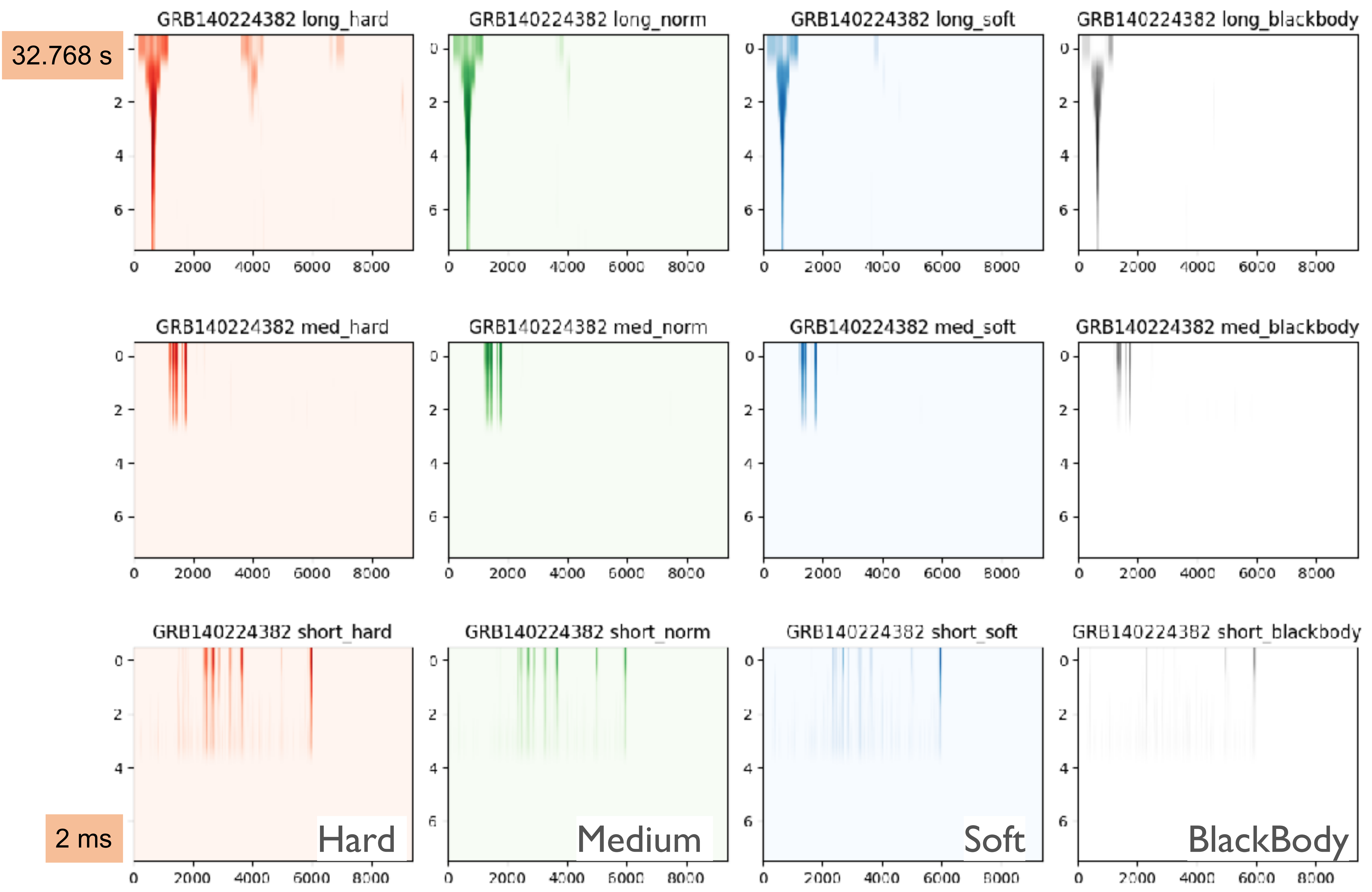
- MinVal = 0, **5**, 10

M. Negro - michelanegro@lsu.edu

The input

Training sample: 2361 GRBs (all GBM triggers Jan 2013 — May 2023)

Test sample: 151 GRBs (all GBM triggers May 2023 — Dec 2023)



AE 1 — Long timescales (4 images, 8x9376)

AE 2 — Medium timescales (4 images, 3x7500)

AE 3 — Short timescales (4 images, 3x7500)

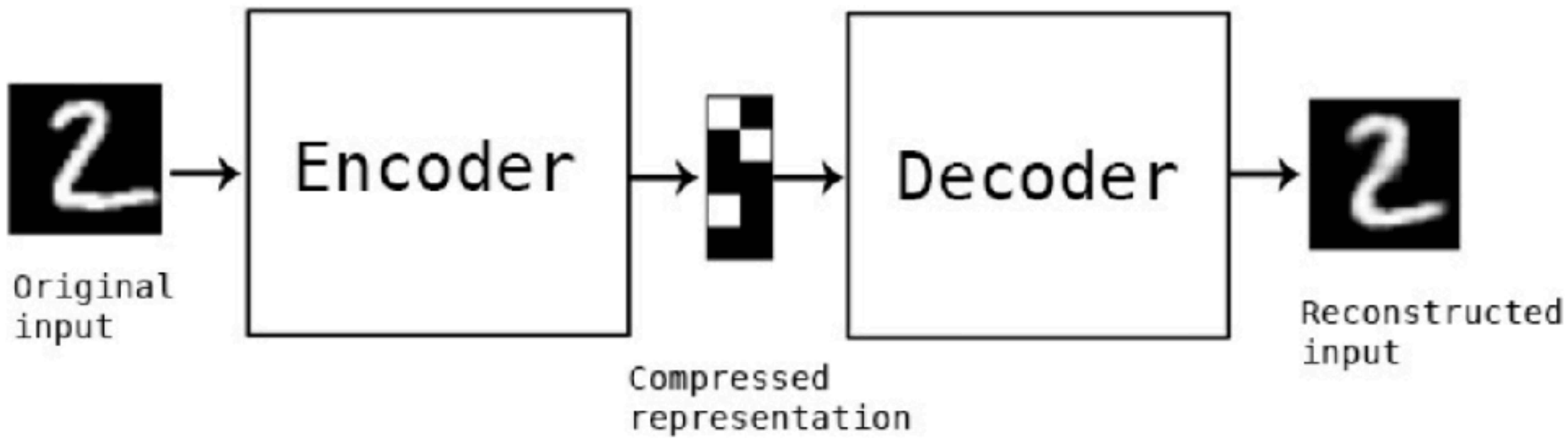
For each GRB we normalize the set of waterfalls to the max log-likelihood

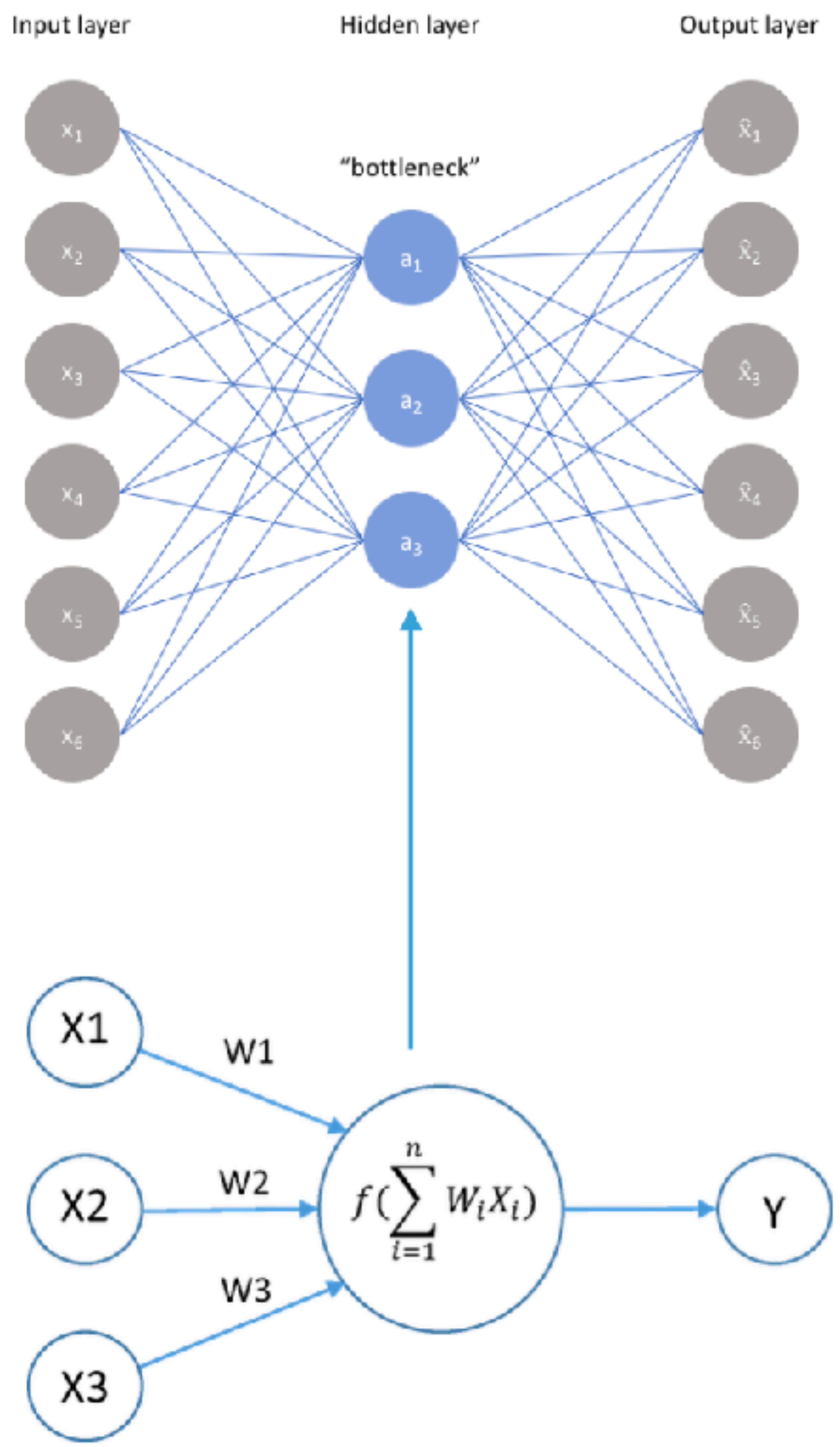
M. Negro - michelanegro@lsu.edu

Autoencoder

- Class of deep learning algorithms
- Unsupervised learning (no labels needed)
- Finds non linear
- Output = input

The autoencoder is trained using **backpropagation**: the loss gradient is used to update weights

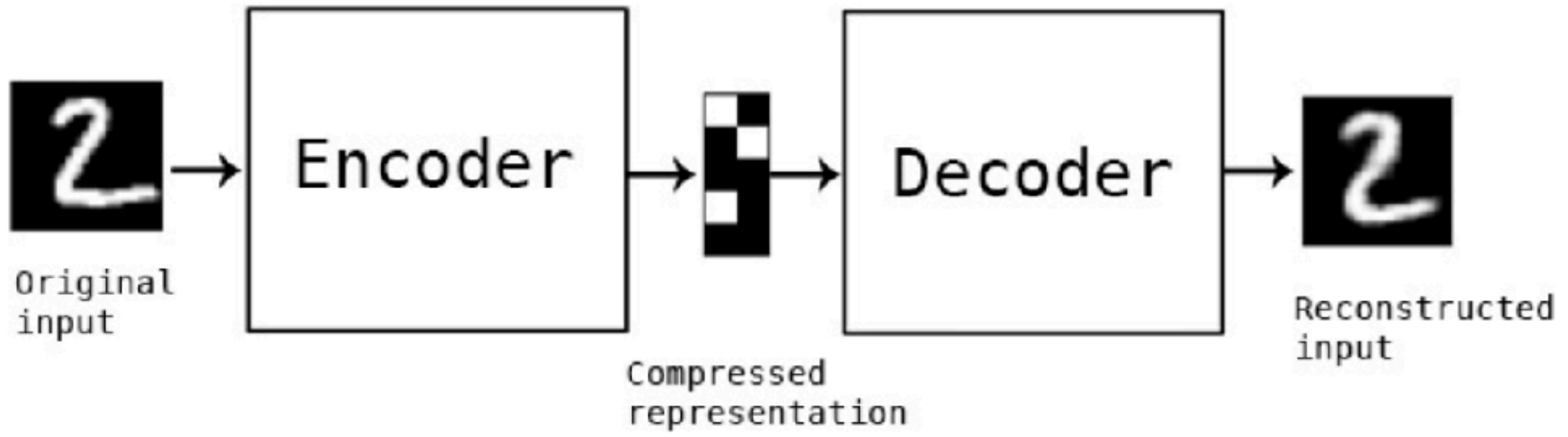


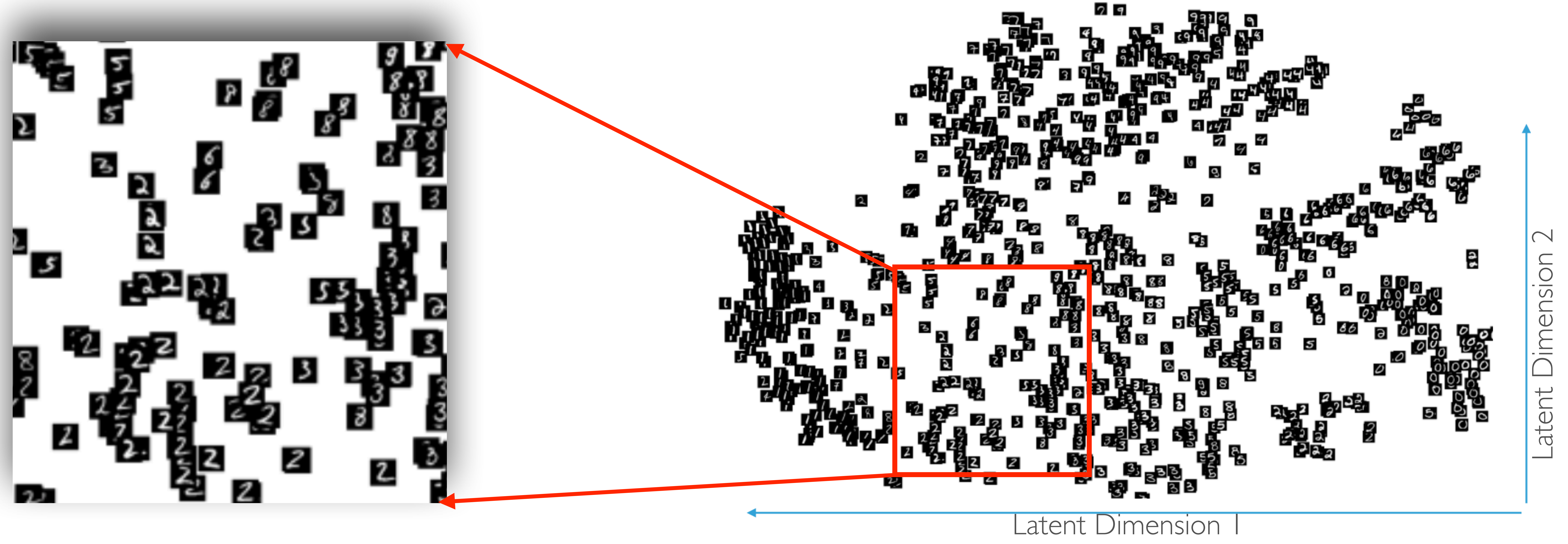


M. Negro - michelanegro@lsu.edu

Autoencoder's Latent Space

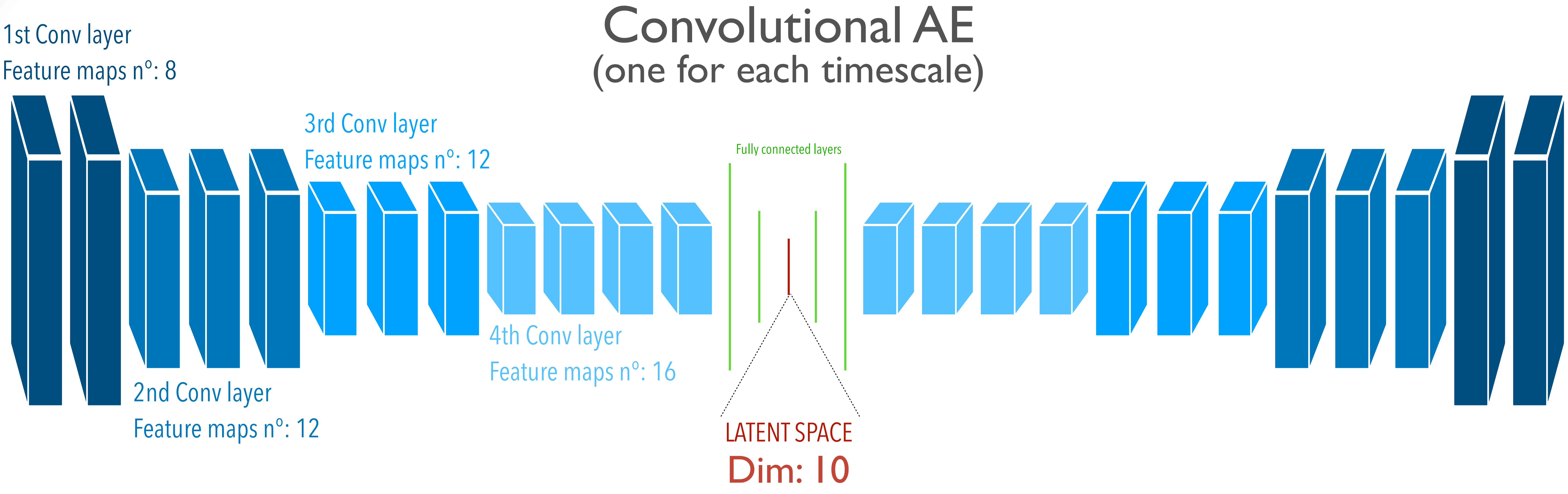
Simple case of a 2D Latent space.
No physical parameters in the latent space.
Elements with similar features "cluster" together.





The Convolutional Autoencoder

Dimensionality reduction by increasing kernel size and the stride



Loss function

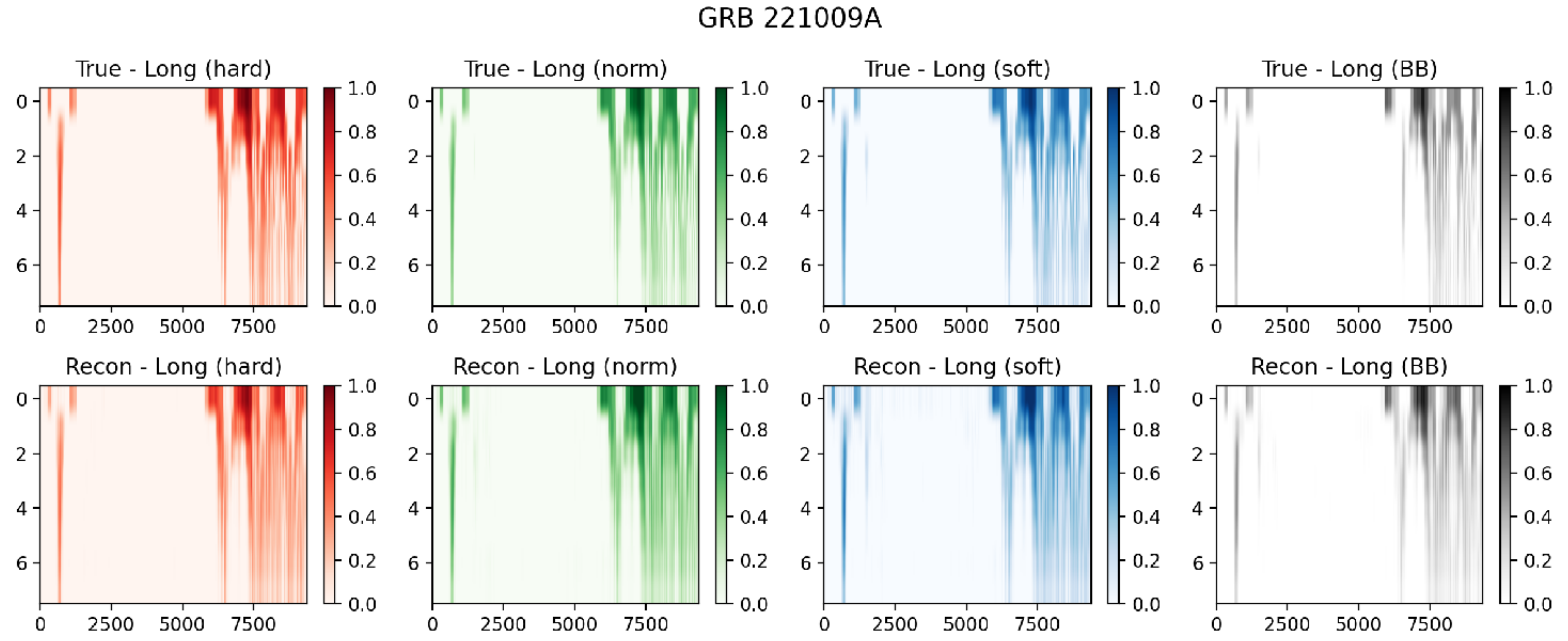
$$L(\mathbf{x}) = \frac{1}{N}\sum_i \left((x_i - d(e(x_i)))^2 \cdot w_i\right)$$

$$w_i = \begin{cases} 2 & \text{if } 0 < x_i < 0.6 \ \& \ \text{epoch} > \alpha \ \& \ \text{epoch} \ \% \ \beta \neq 0 \\ 1 & \text{otherwise} \end{cases}$$

The 10-D latent spaces of the 3 AE are then combined into one 30-D space

M. Negro - michelanegro@lsu.edu

Reconstructed vs original waterfalls



We optimize the architecture and the hyper parameters **blindly:** input-output comparison

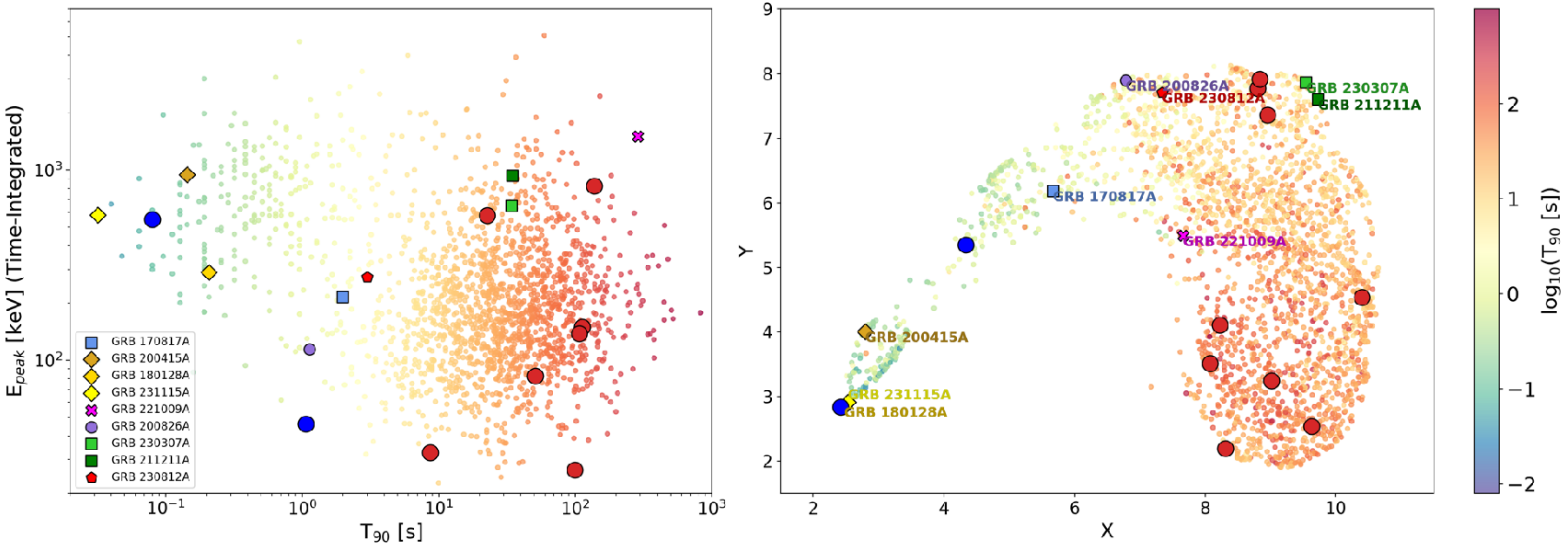
10-D is the minimal dimension for which input/output comparison was satisfactory

$$\text{Score} = \frac{\sum_i |I_i - \bar{I}_i|}{\sum_i i}$$

M. Negro - michelanegro@lsu.edu

Some famous GRBs

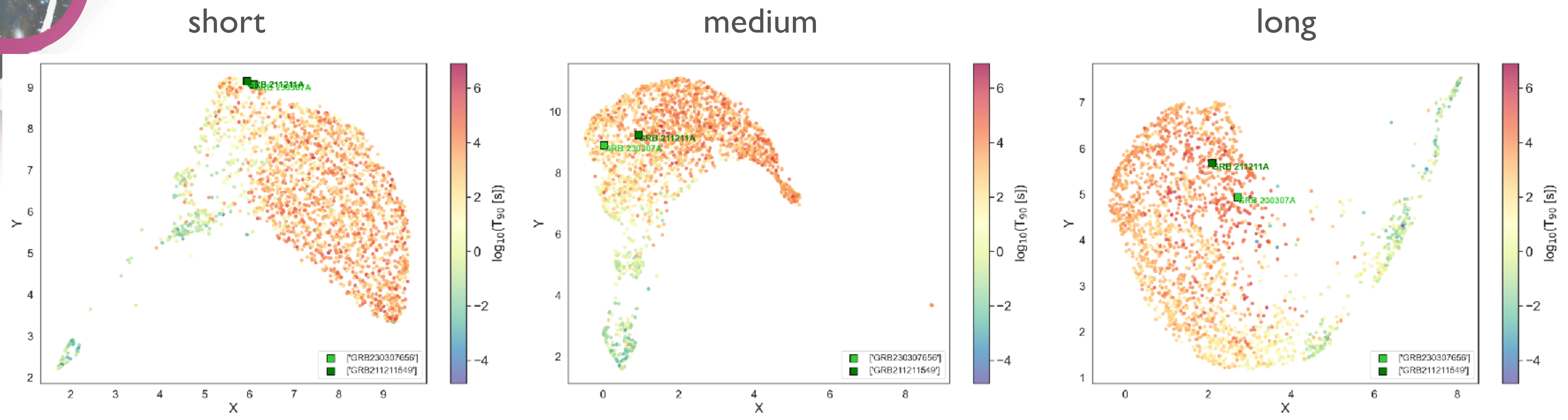
Groupings in the UMAP distributions seem to separate different "flavors" of GRBs



M. Negro - michelanegro@lsu.edu

The case of long merger

GRB 230307A and GRB 211211A are two long GRBs (~100 s) where the prompt emission is followed by a kilonova-like thermal transient, indicating a merger origin.



One would expect that the distinctive signature informing us that these two GRBs are mergers resides in the short-timescale (minimum variability timescale,).

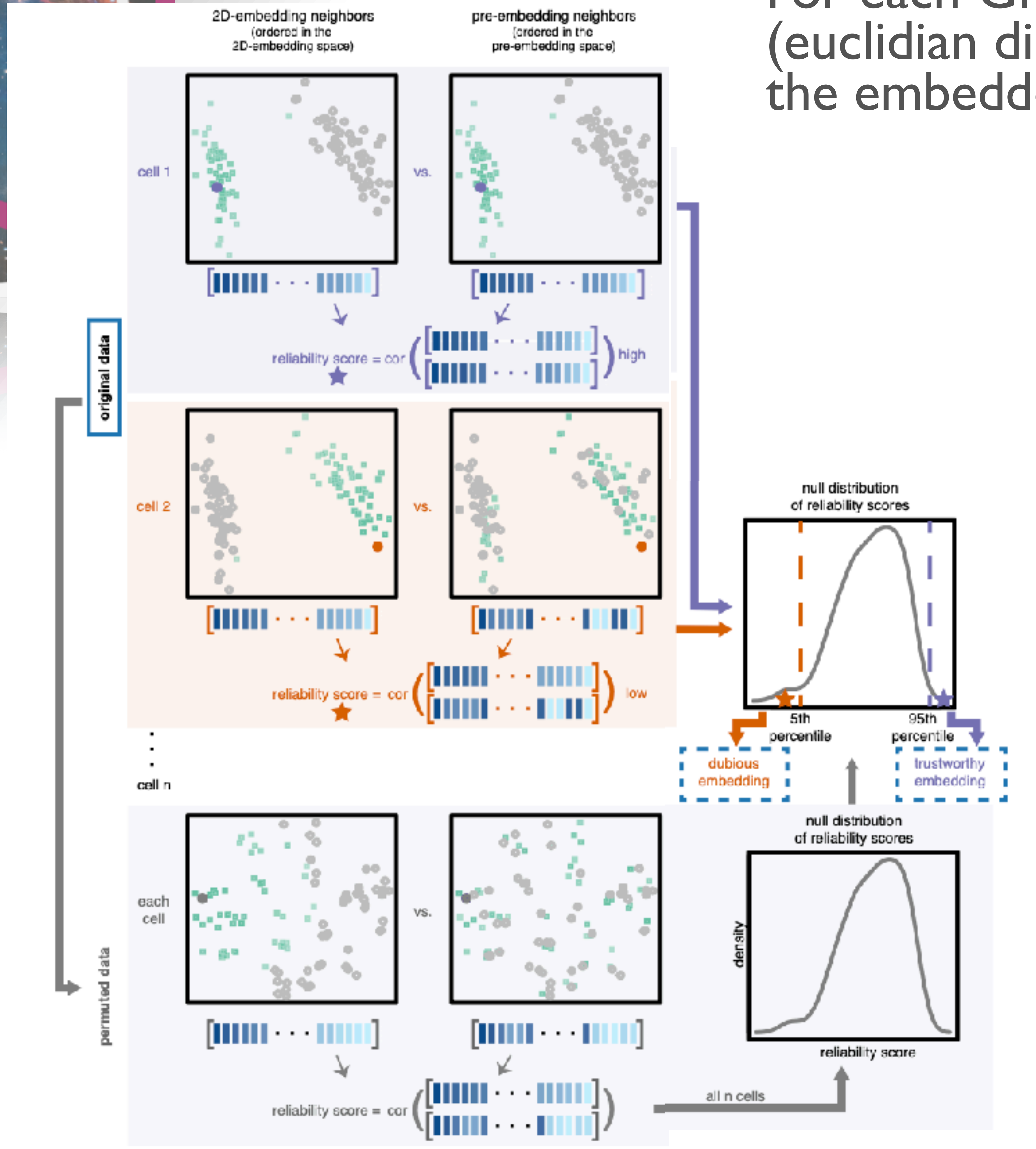
Veres et al 2023: Extreme Variability in a Long-duration Gamma-Ray Burst Associated with a Kilonova

M. Negro - michelanegro@lsu.edu

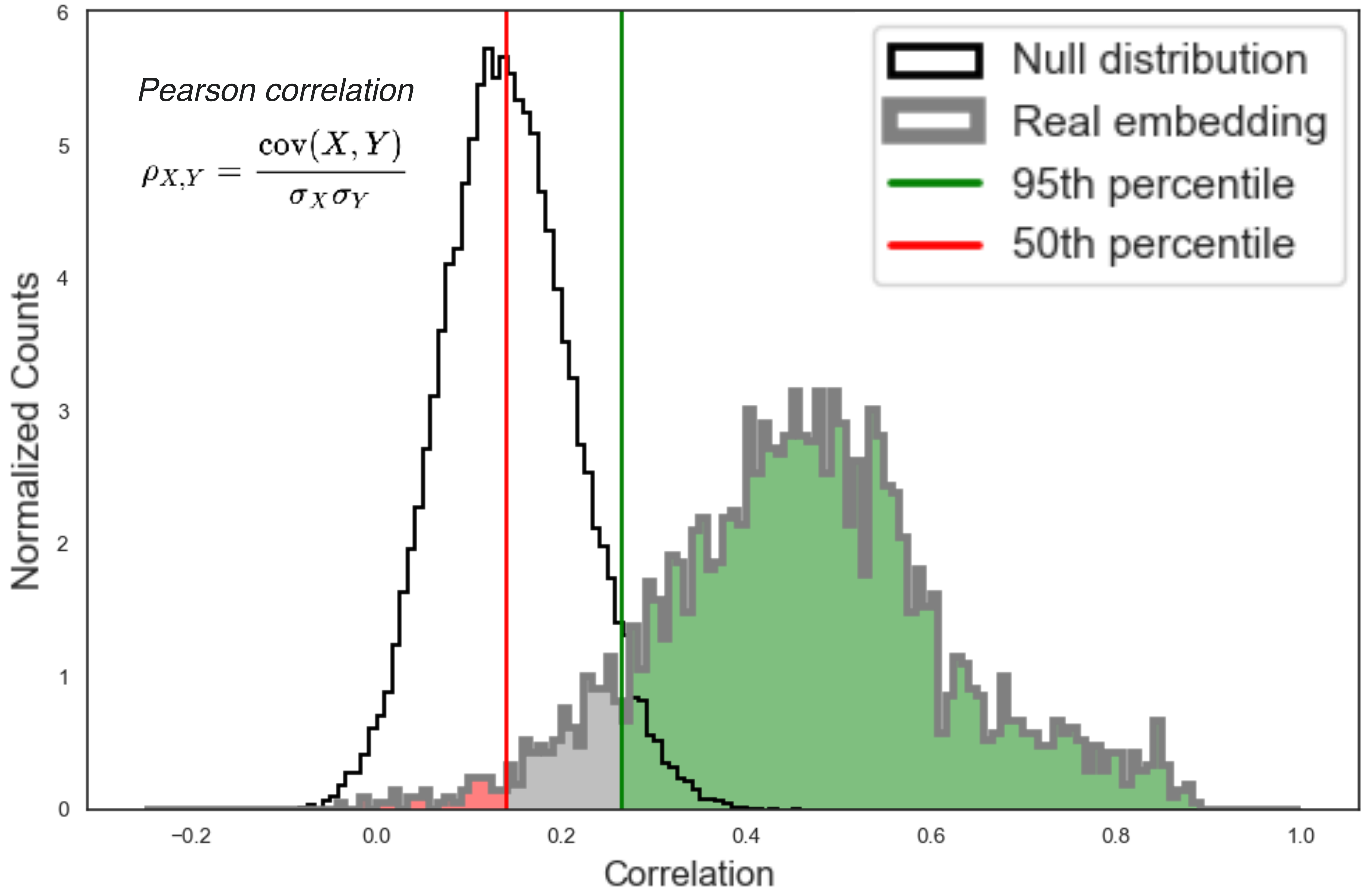
Towards the classification: embedding trustworthiness

scDEED Xia, Lee, and Li 2023

For each GRB we evaluate the correlation between the closest (euclidian distance) half neighbors in the pre-embedding space and the embedded space: the higher the correlation the better



Trustworthy: 89.5% Dubious: 2.0%

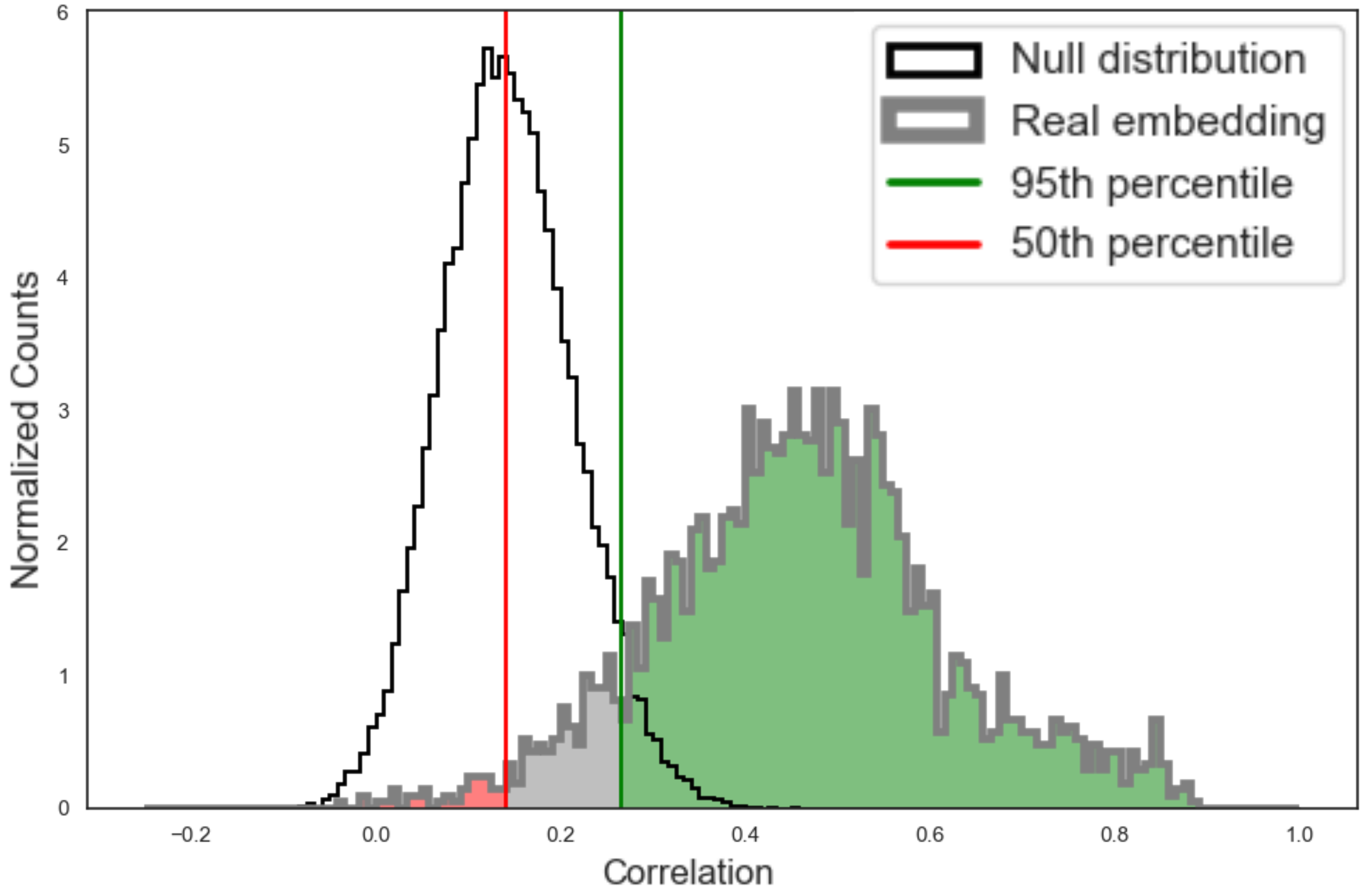


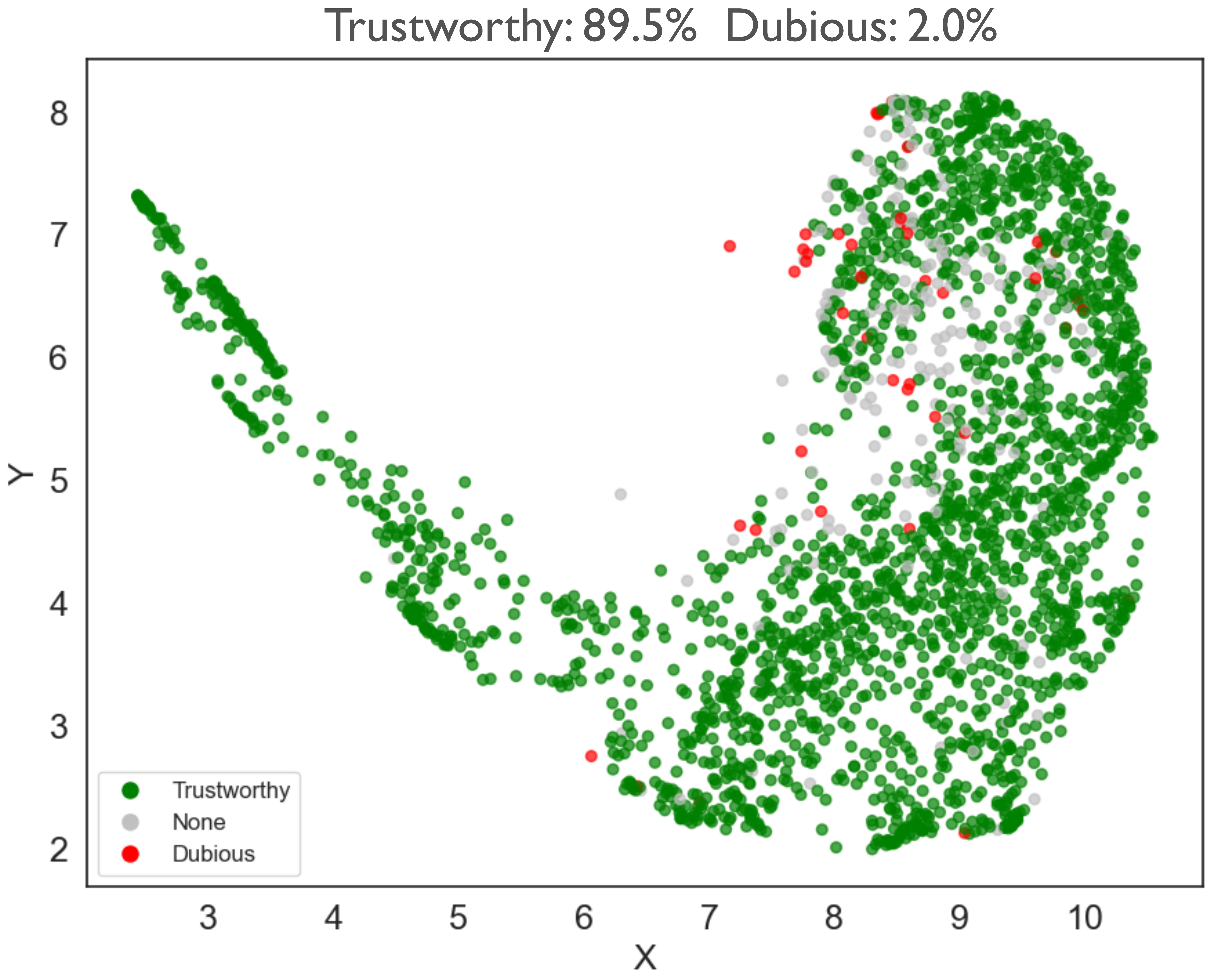
Towards the classification: embedding trustworthiness

N=30

N=2

- n_neighbors = 30
- min_dist = 0.0
- n_components = 3 (or 2)
- metric = 'euclidean'
- local_connectivity = 0.5
- n_epochs = 1000
- learning_rate=0.001



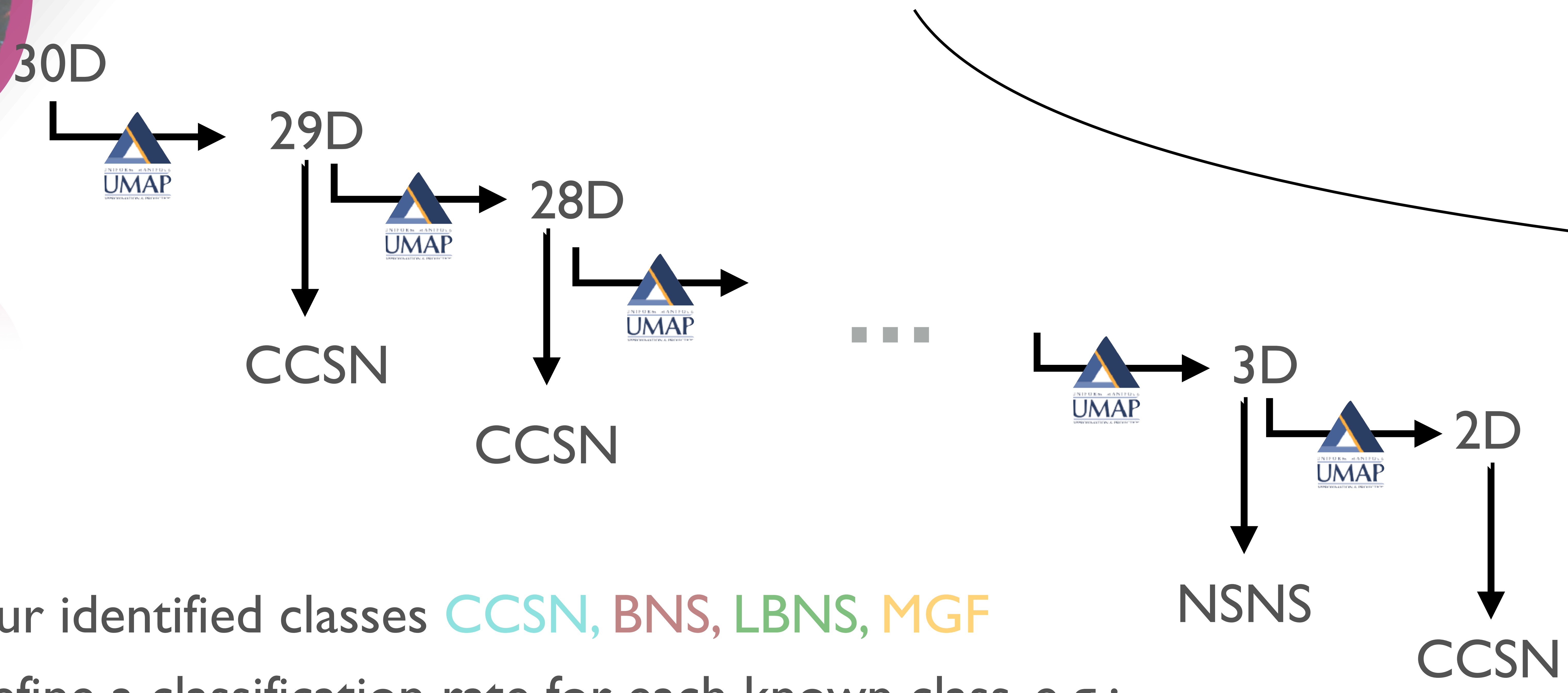


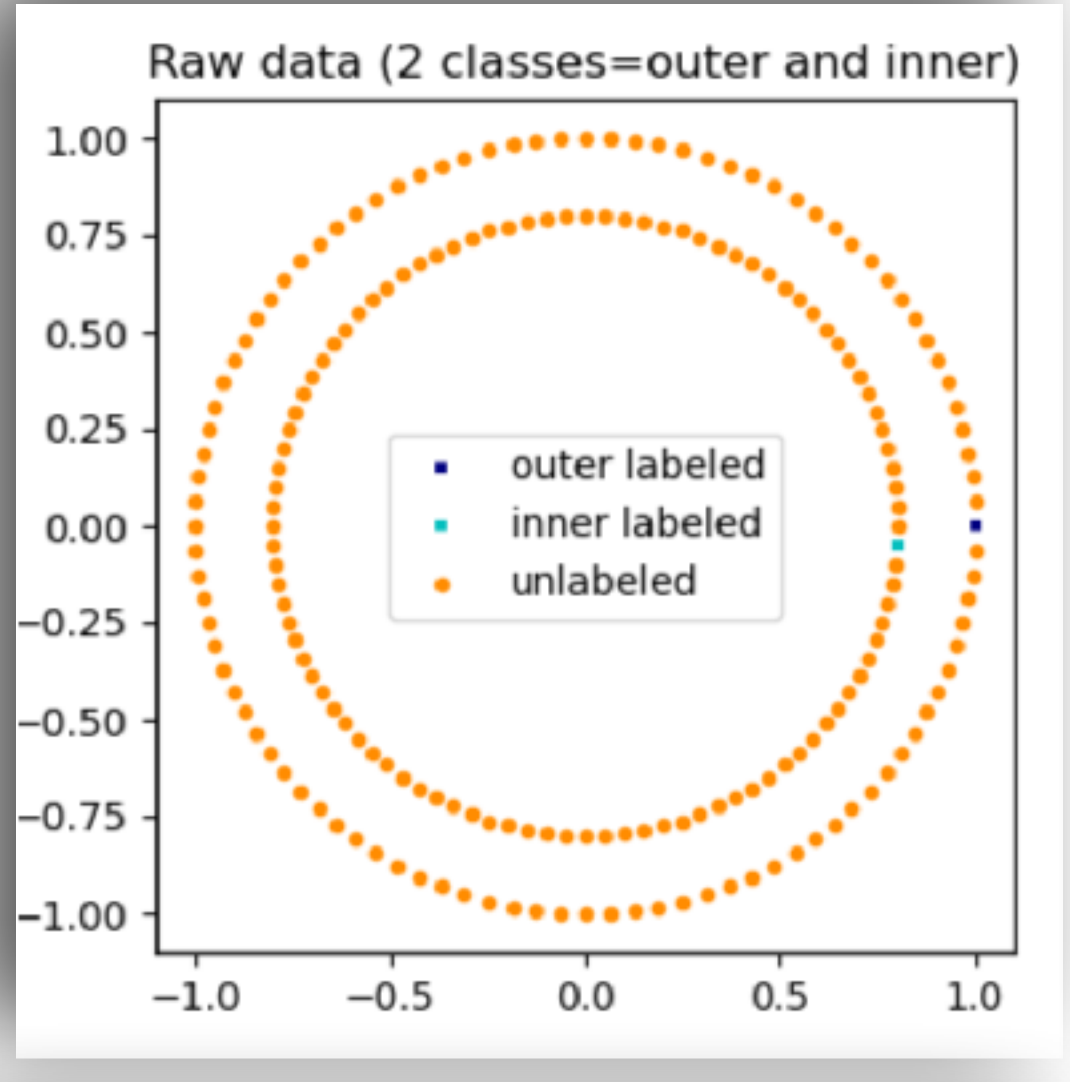
M. Negro - michelanegro@lsu.edu

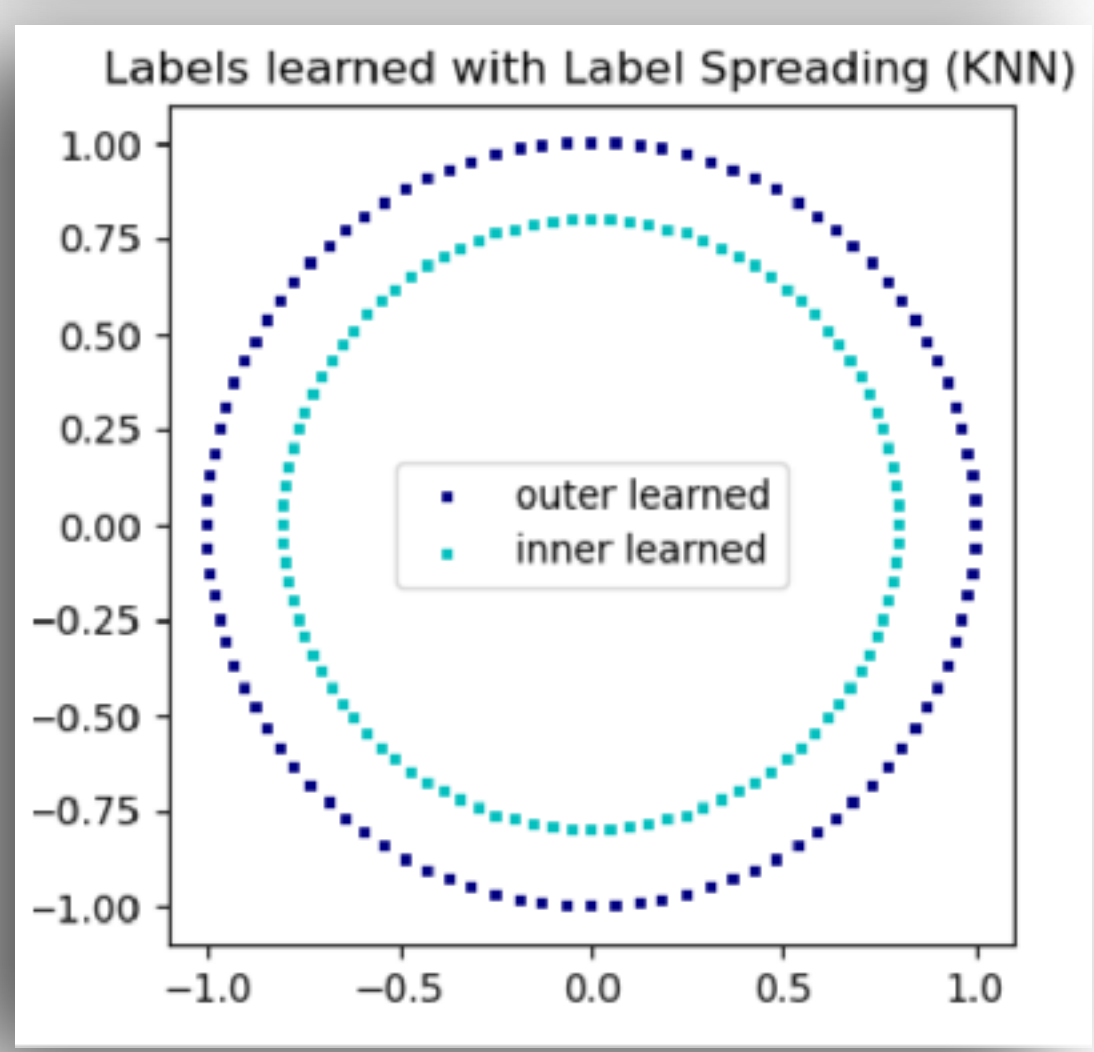
Towards the classification: semi-supervised classification

Can we classify the GRBs in the embeddings based on the known projenitors?

We can use **label propagation** (or label spreading)







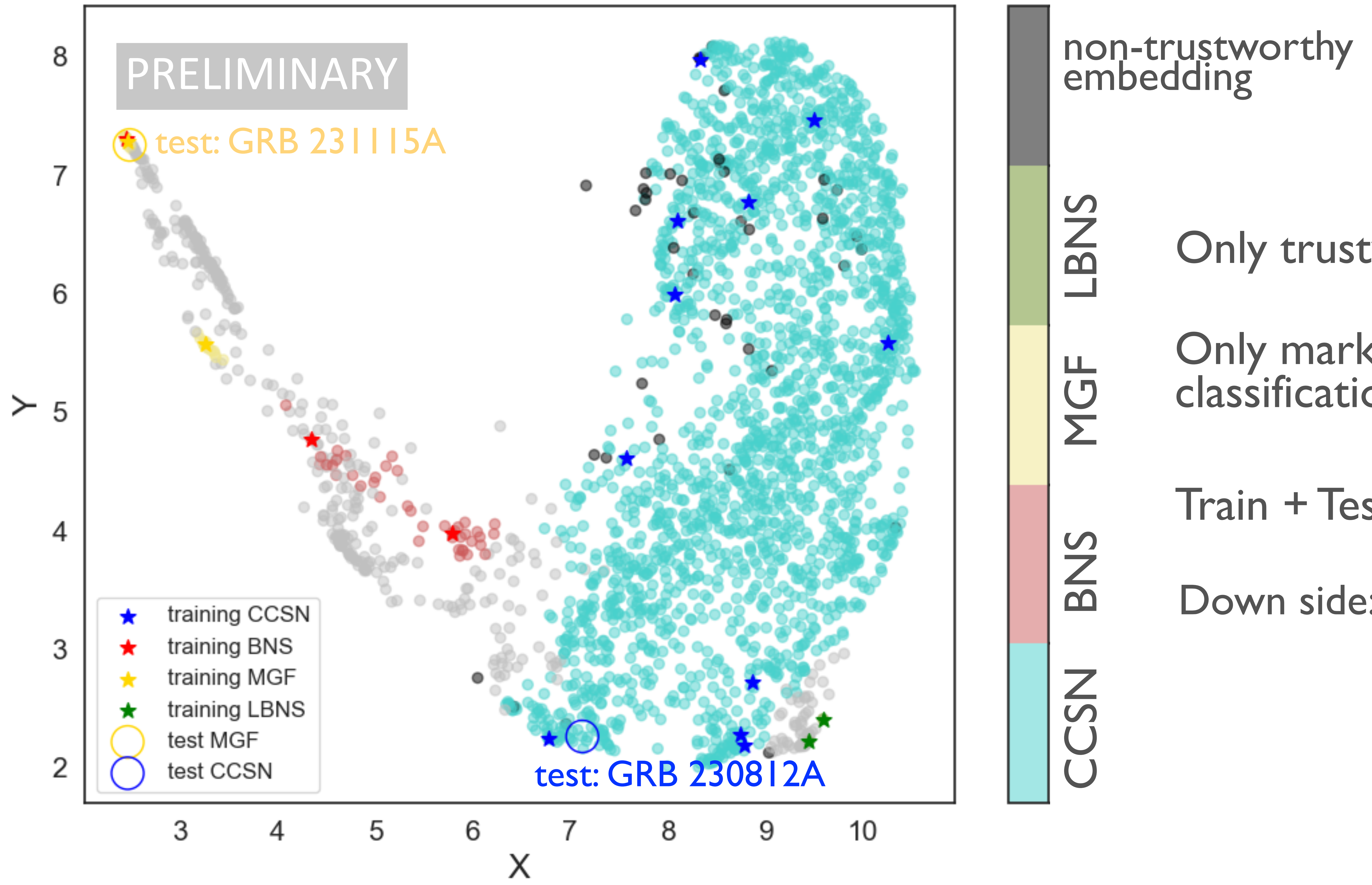
Four identified classes CCSN, BNS, LBNS, MGF

Define a classification rate for each known class, e.g.:

97.0% CCSN
2.7% BNS
0.3% LBNS
0.0% MGF

M. Negro - michelanegro@lsu.edu

Towards the classification: semi-supervised classification



Only trustworthy events

Only marking GRB with classification rate > 95%

Train + Test events

Down side: no room for "unknown class"

M. Negro - michelanegro@lsu.edu

Conclusion

Ultimate goal:
rapid identification of the progenitor of a given event, allowing for specific follow-up observations to occur,

- Unsupervised (or self-supervised) ML techniques are needed
- ML analysis techniques are no longer black boxes if one looks into it
- Waterfall plots can be improved/expanded (e.g., background fit procedure)
- The input format can be improved to allow one single AE to be trained for all timescales.
- Improved AE architecture, possibly avoiding dim. reduction algorithms
- If it works, the plan is to run this pipeline automatically on GBM GRBs and possibly expanded to other missions

Conclusion

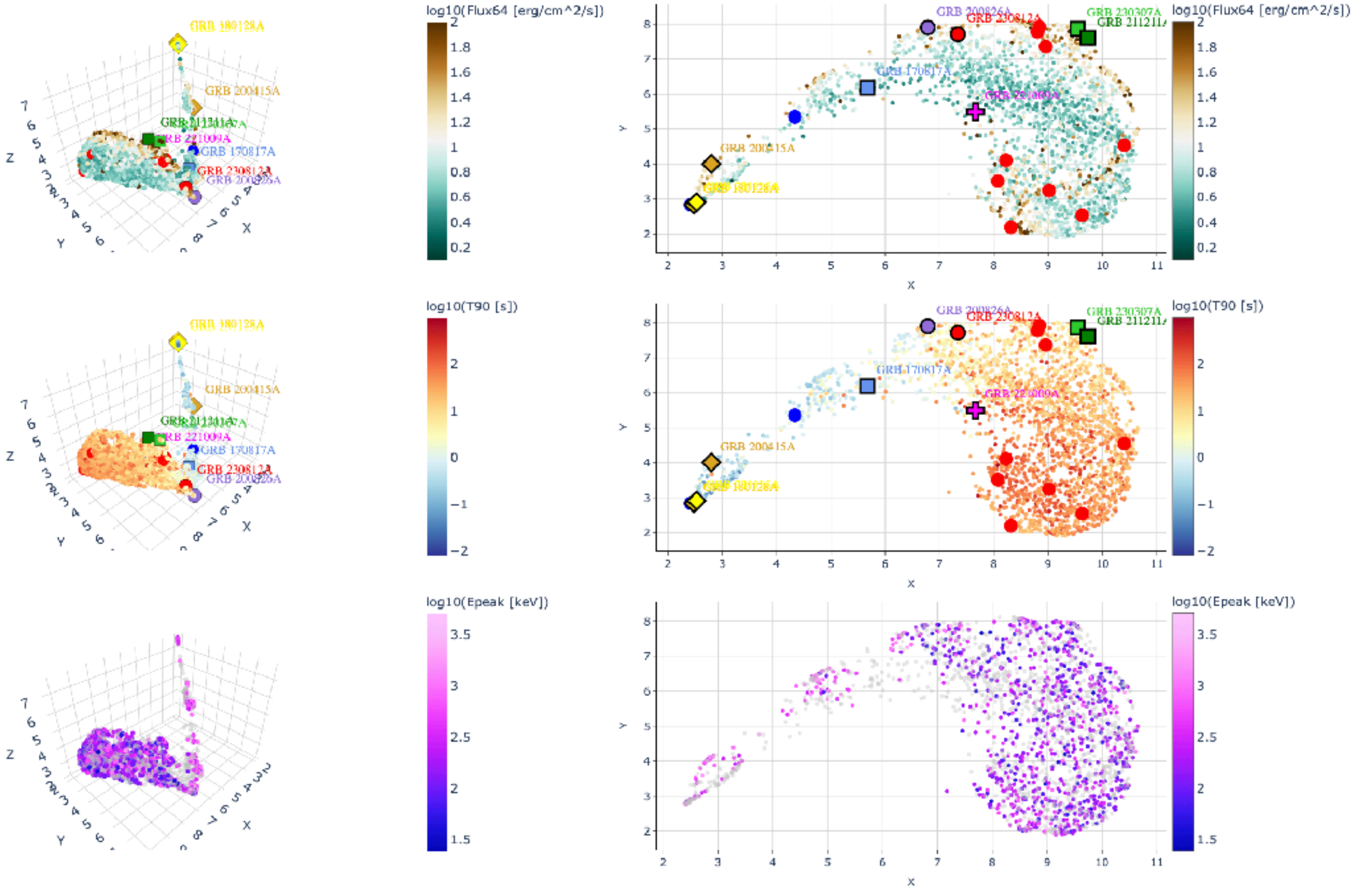
Ultimate goal:
rapid identification of the progenitor of a given event,
allowing for specific follow-up observations to occur,

- Unsupervised (or self-supervised) ML techniques are needed
- ML analysis techniques are no longer black boxes if one looks into it
- Waterfall plots can be improved/expanded (e.g., background fit procedure)
- The input format can be improved to allow one single AE to be trained for all timescales.
- Improved AE architecture, possibly avoiding dim. reduction algorithms
- If it works, the plan is to run this pipeline automatically on GBM GRBs and possibly expanded to other missions

Thank you for your attention!

!Taaac

Interactive view of the 3D and 2D embedding (preliminary)



M. Negro - michelanegro@lsu.edu

APS APRIL MEETING

QUARKS TO COSMOS

Backup

Michela Negro (michelanegro@lsu.edu) - Louisiana State University

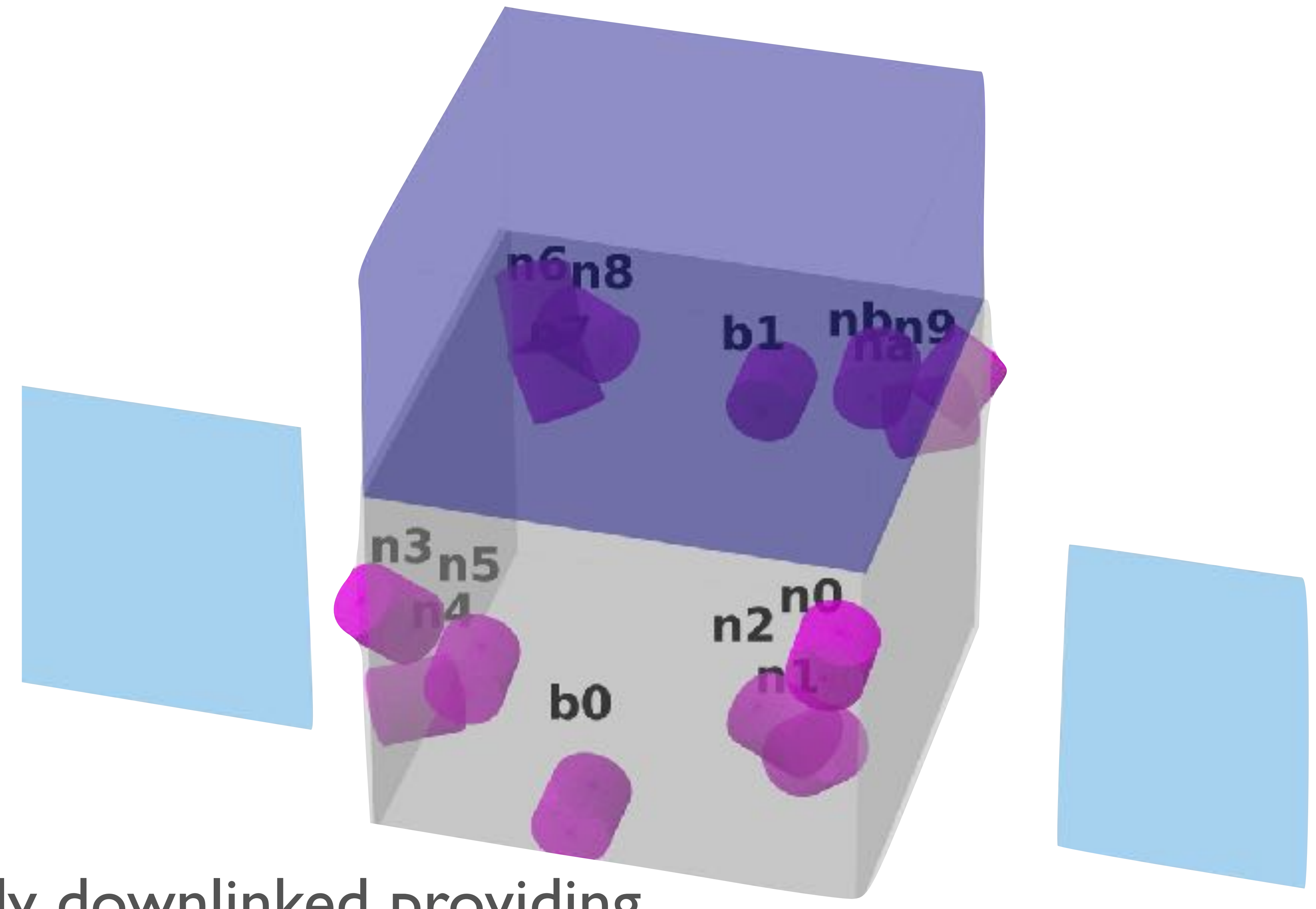
on behalf of the team: N. Cibrario, E. Burns

Fermi Gamma-ray Burst Monitor

(Meegan et al. 2009)

Energy Range: 8 keV — 40 MeV
14 scintillator detectors
- 12 NaI (8 –1000 keV)
- 2 BGO (200 – 40 MeV)

Each detector gets its own data file

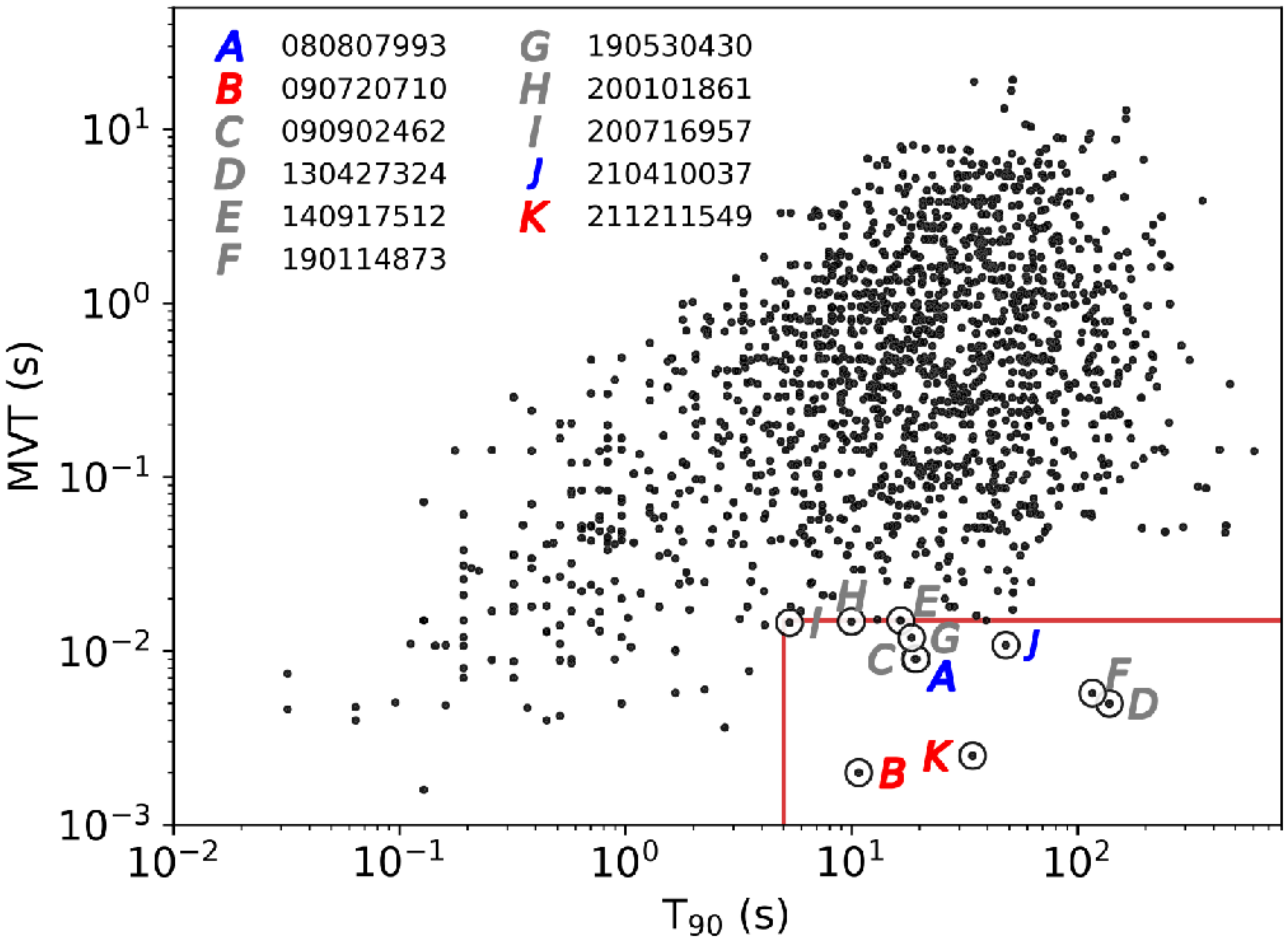


CTTE (Continuous time-target event) data are continuously downlinked providing information of individual photons at 2 us in 128 energy channels.

The basis for the offline sub-threshold analyses developed by the GBM team.

Minimum variability timescale

Veres et al 2023: Extreme Variability in a Long-duration Gamma-Ray Burst Associated with a Kilonova



M. Negro - michelanegro@lsu.edu

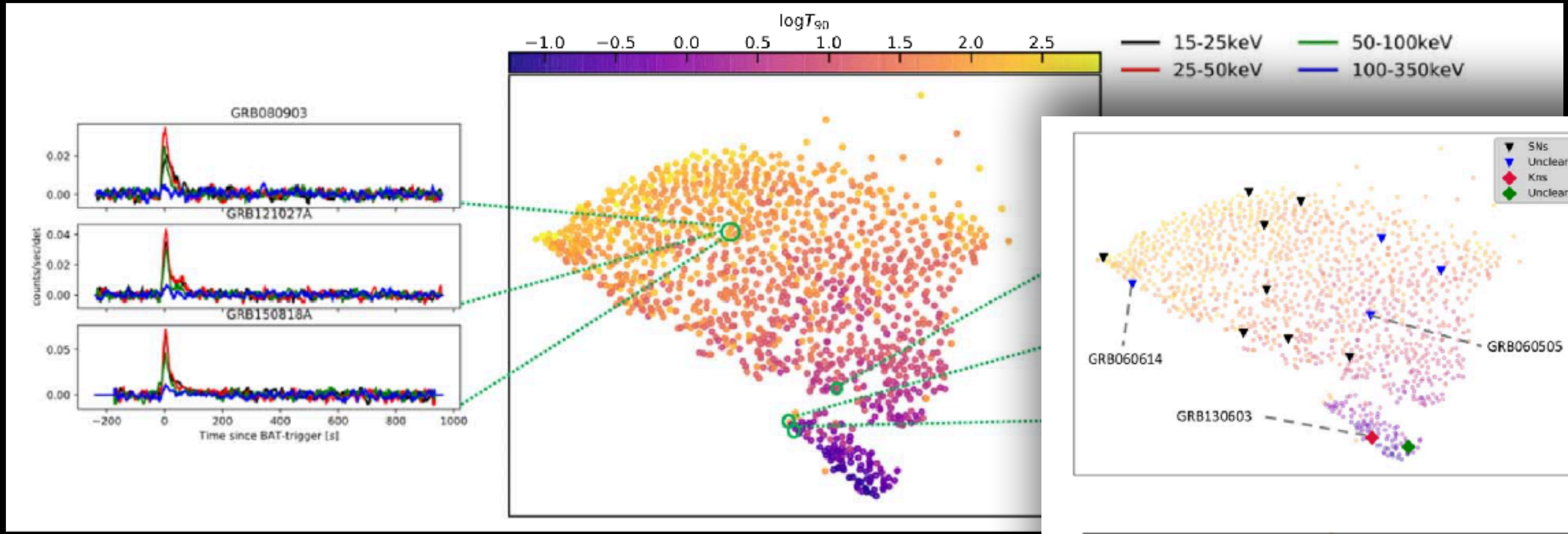
Overview of related works

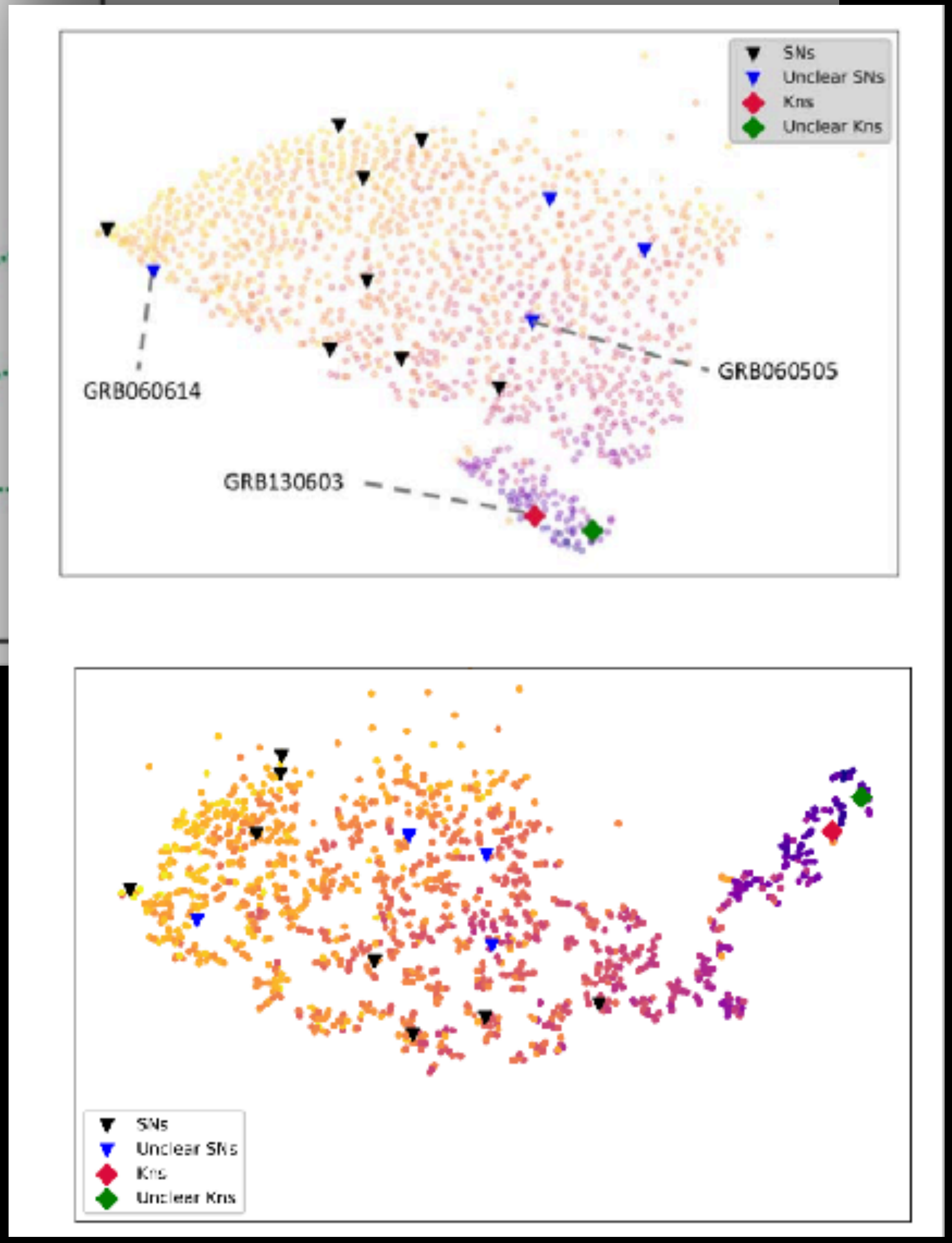
Methods: Applying t-SNE to Swift/BAT Light Curves

Input size:
1450 × 25,883

Swift GRBs separated into two groups based on prompt light curves (norm to tot fluence)

64 ms binned light curve in each band (limited to the interval out to T+100s then zero-padded)





Jespersen et al. June 2020;
Dimple et al. 2023;
Garcia-Cifuentes et al. Nov 2023;
Chen et al. Jun 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

Playing with the ***perplexity:***

Higher perplexity (30) = global structures

Lower perplexity (5) = local structure

Tadpole structure seems to appear

M. Negro - michelanegro@lsu.edu

Overview of related works

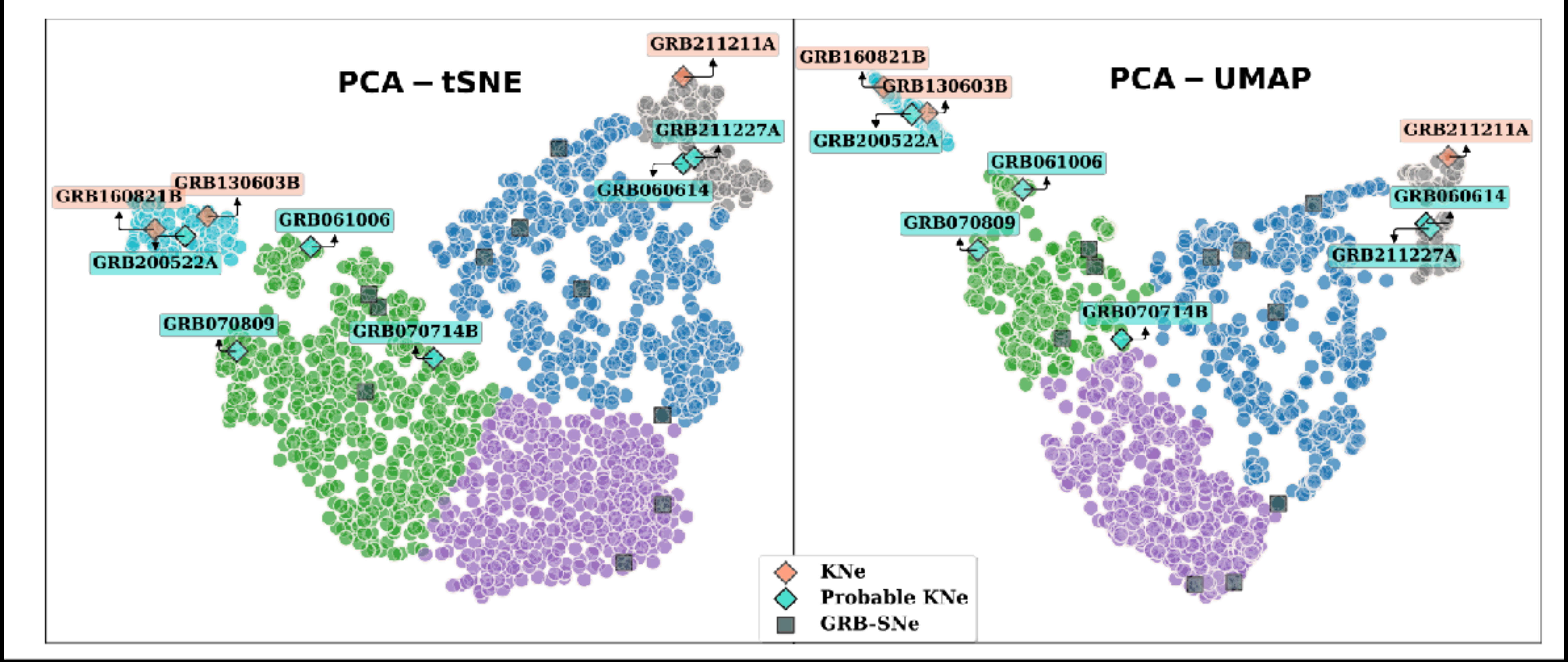
Methods: Applying t-SNE and UMAP to Swift/BAT Light Curves

Input size:
1450 × 25,883

1450 Swift GRBs(same data prep as Jespersen et al 2020)

t-SNE and UMAP hyperparameters scanned and chosen to maximize the clustering

Clustering: gaussian mixture method (AutoGMM): 5 clusters found



Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. Jul 2023;
Chen et al. Nov 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

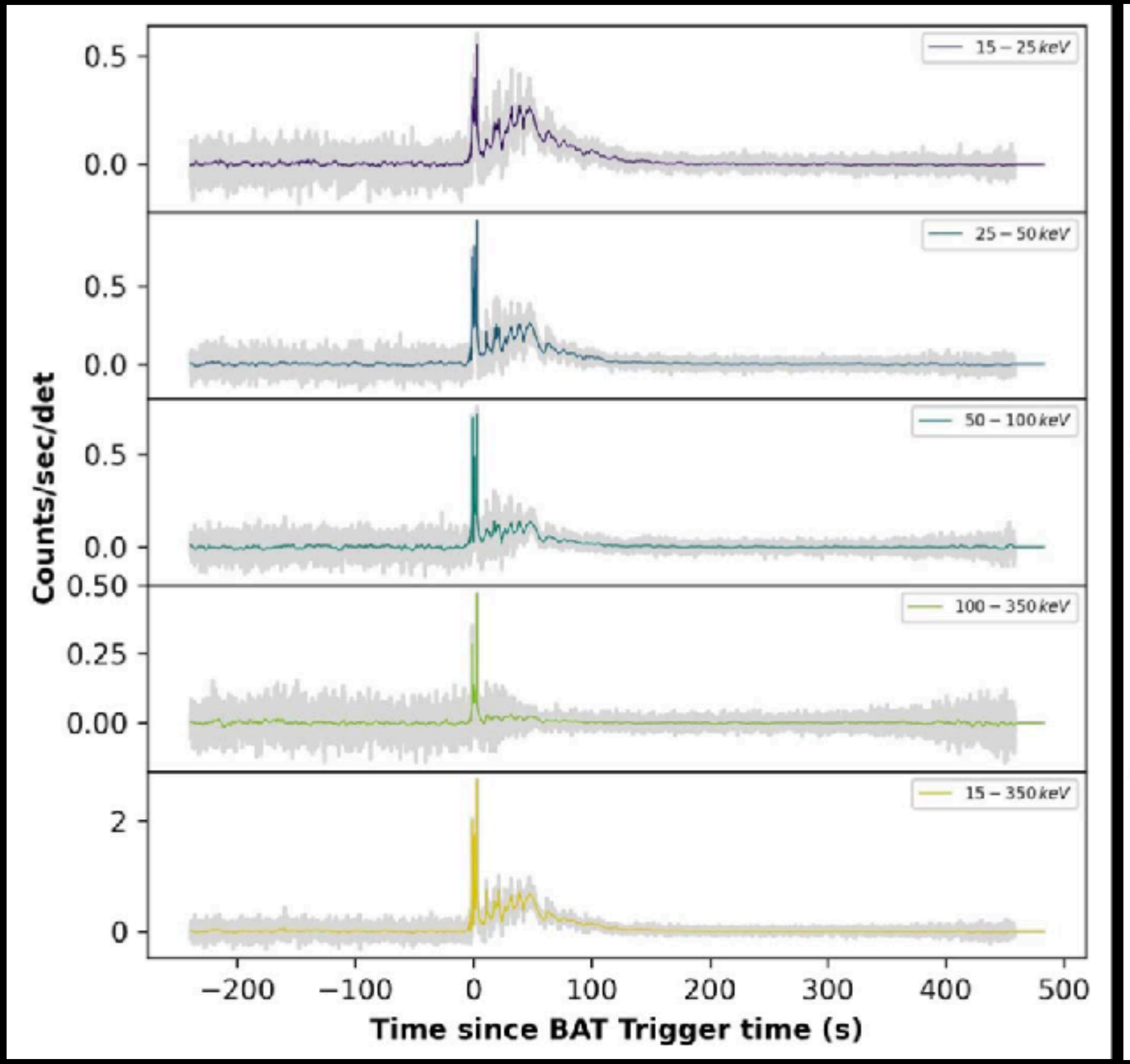
M. Negro - michelanegro@lsu.edu

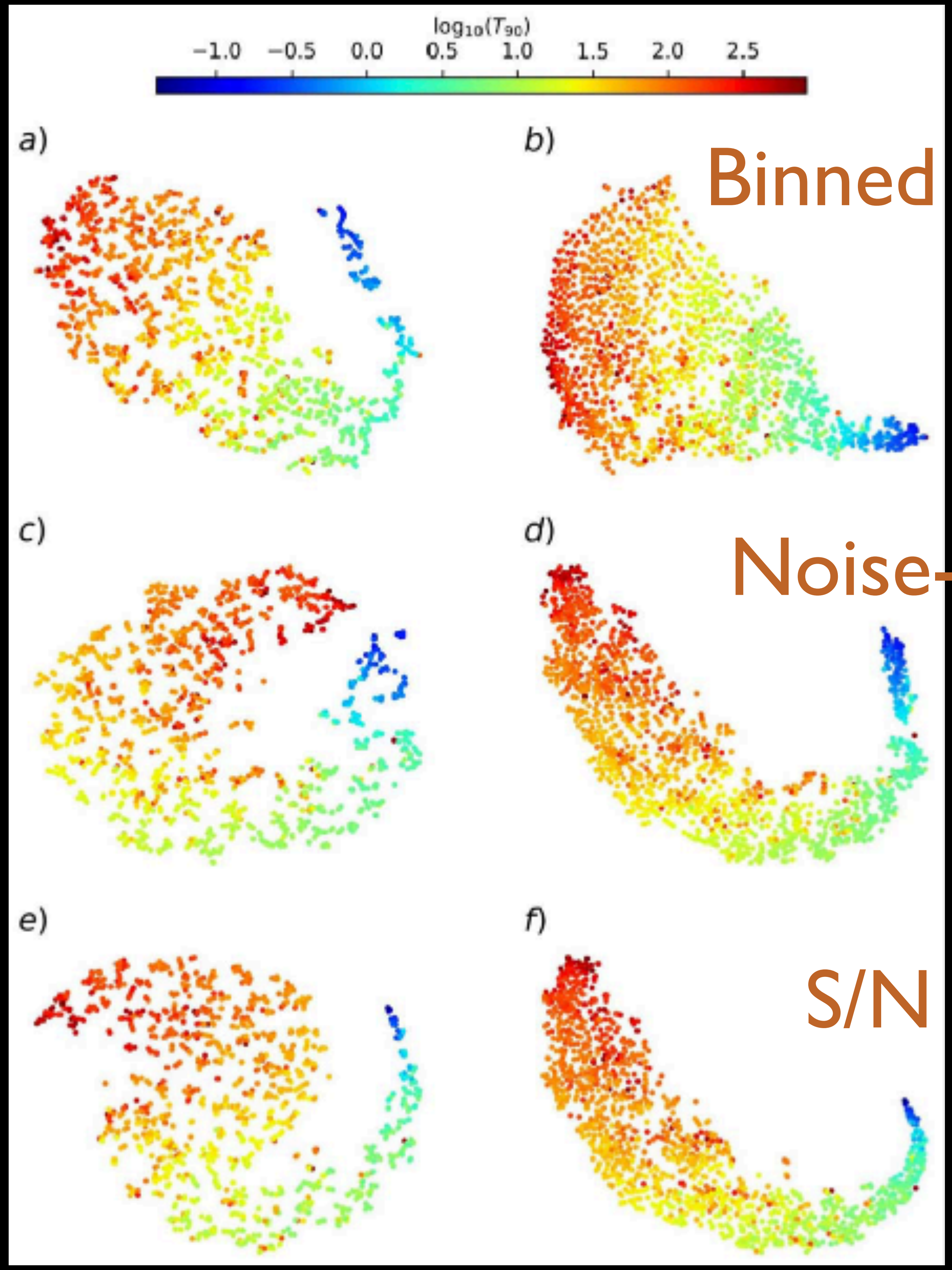
Overview of related works

Methods: Applying t-SNE to Swift BAT light curves

Input size:
2297 × 5

1527 BAT GRBs using light curves in 5 energy bins to identify GRBs with extended emission





Binned 64 ms

Noise-reduced

S/N selection

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. Jul 2023;
Chen et al. Nov 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

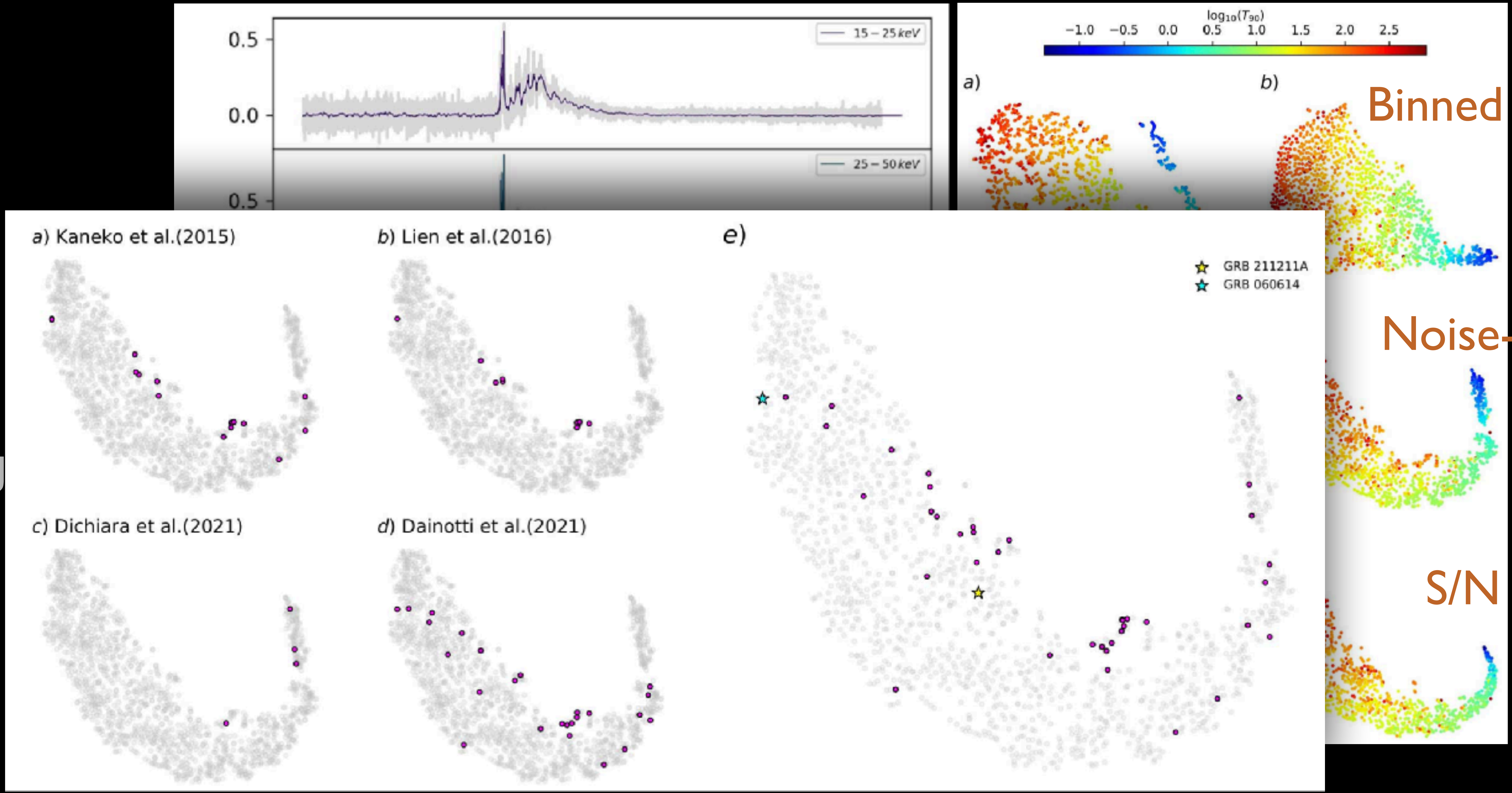
M. Negro - michelanegro@lsu.edu

Overview of related works

Methods: Applying t-SNE to Swift BAT light curves

Input size:
2297 × 5

1527 BAT GRBs using light curves in 5 energy bins to identify GRBs with extended emission



Binned 64 ms

Noise-reduced

S/N selection

Jespersen et al. Jun 2020;
Dimple et al. Jun 2023;
Garcia-Cifuentes et al. J
Chen et al. Nov 2023;
Negro et al 5 Jun 2024;
Zhu et al. 8 Jun 2024;
Dimple et al. Aug 2024;

M. Negro - michelanegro@lsu.edu

ConvAE and training parameters

	Short TS	Medium TS	Long TS
Epochs	250	250	200
Decaying LR (γ - step)	8×10^{-4} (0.9 - 10)	8×10^{-4} (0.9 - 10)	1×10^{-4} (0.9 - 10)
Optimizer		Adam	
Batch Size		4	
Loss function		Mean Squared Error (MSE)	
Activation functions		Leaky ReLU (Hidden layers) & ReLU (Output layer)	

Accounting for zeros in the loss function: weighted MSE loss function

$$\mathrm{L}(\mathbf{x}) = \frac{1}{N}\sum_i \left((x_i - d(e(x_i)))^2 \cdot w_i\right)$$

$$w_i = \begin{cases} 2 & \text{if } 0 < x_i < 0.6 \text{ \& epoch} > \alpha \text{ \& epoch } \% \ \beta \neq 0 \\ 1 & \text{otherwise} \end{cases}$$

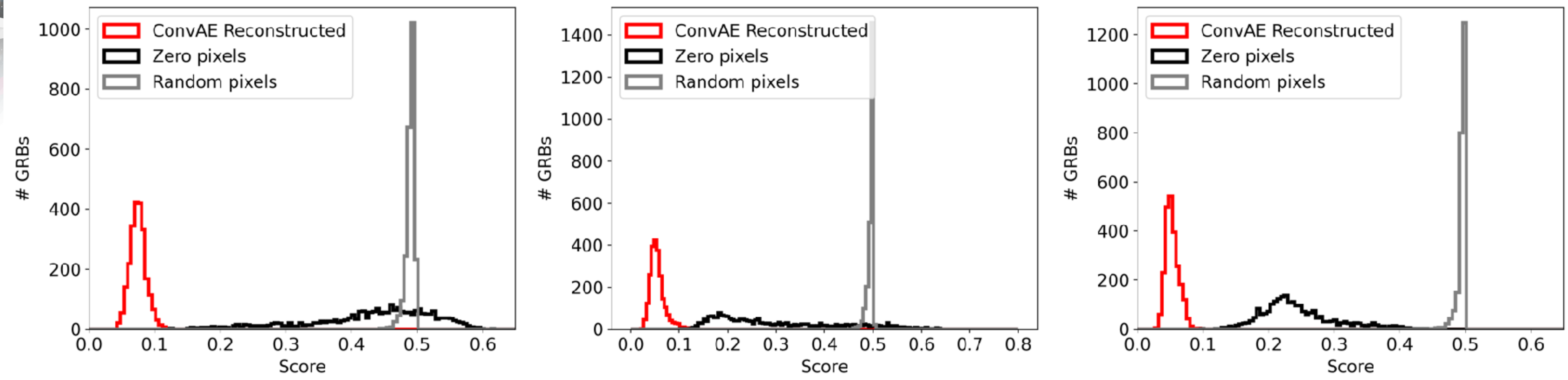
We optimize the architecture and the hyper parameters **blindly:** input-output comparison

M. Negro - michelanegro@lsu.edu

Testing performance of the AE

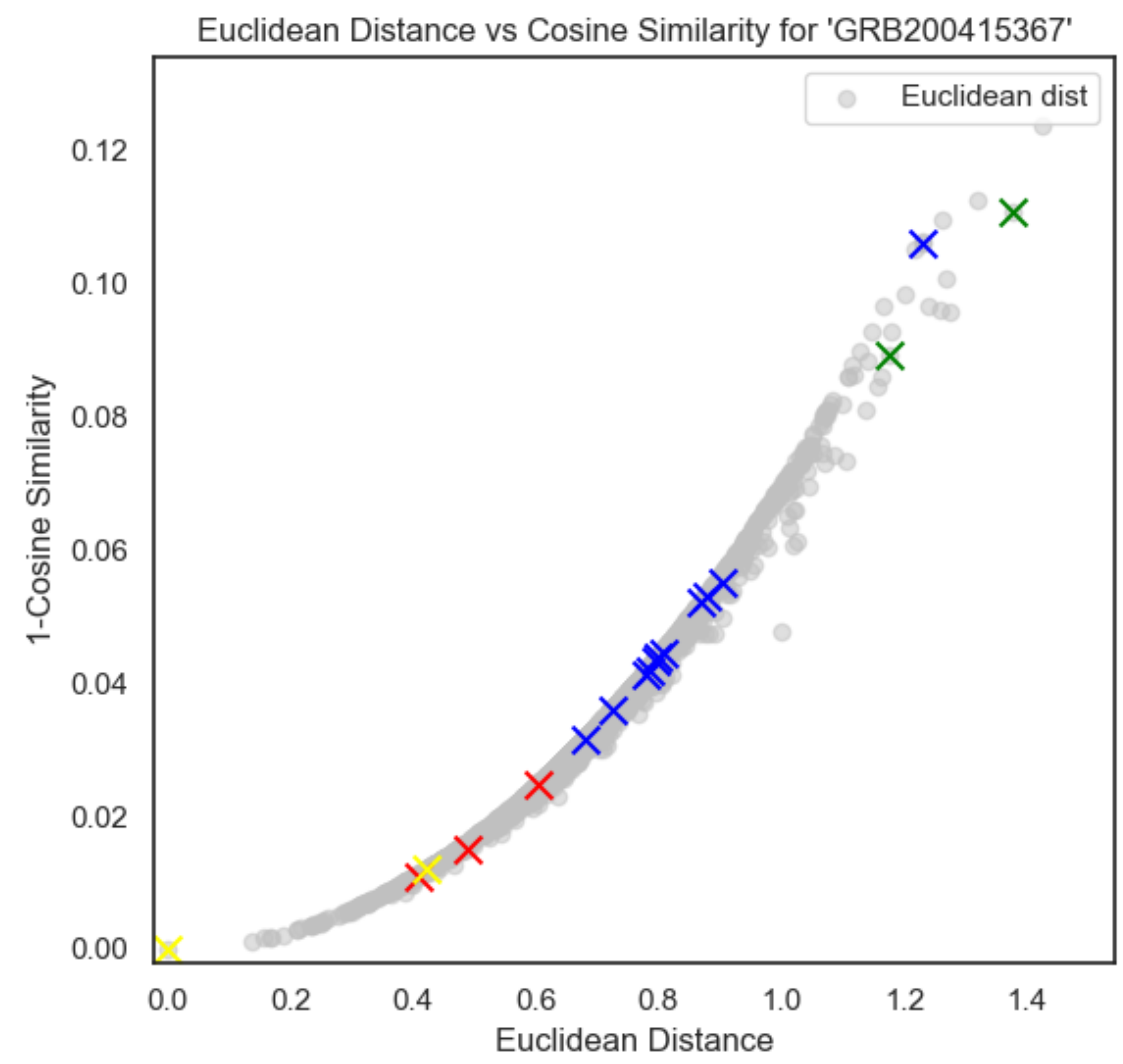
$$\text{Score} = \frac{\sum_i |I_i - \bar{I}_i|}{\sum_i i}$$

ConvAE reconstructed image
All-zeros image
All-random image



M. Negro - michelanegro@lsu.edu

Where do known GRB lie in the 30-D embedding?



Evaluating the distance between known GRBs

Euclidian distance

$$d(p,q) = \|p - q\|$$

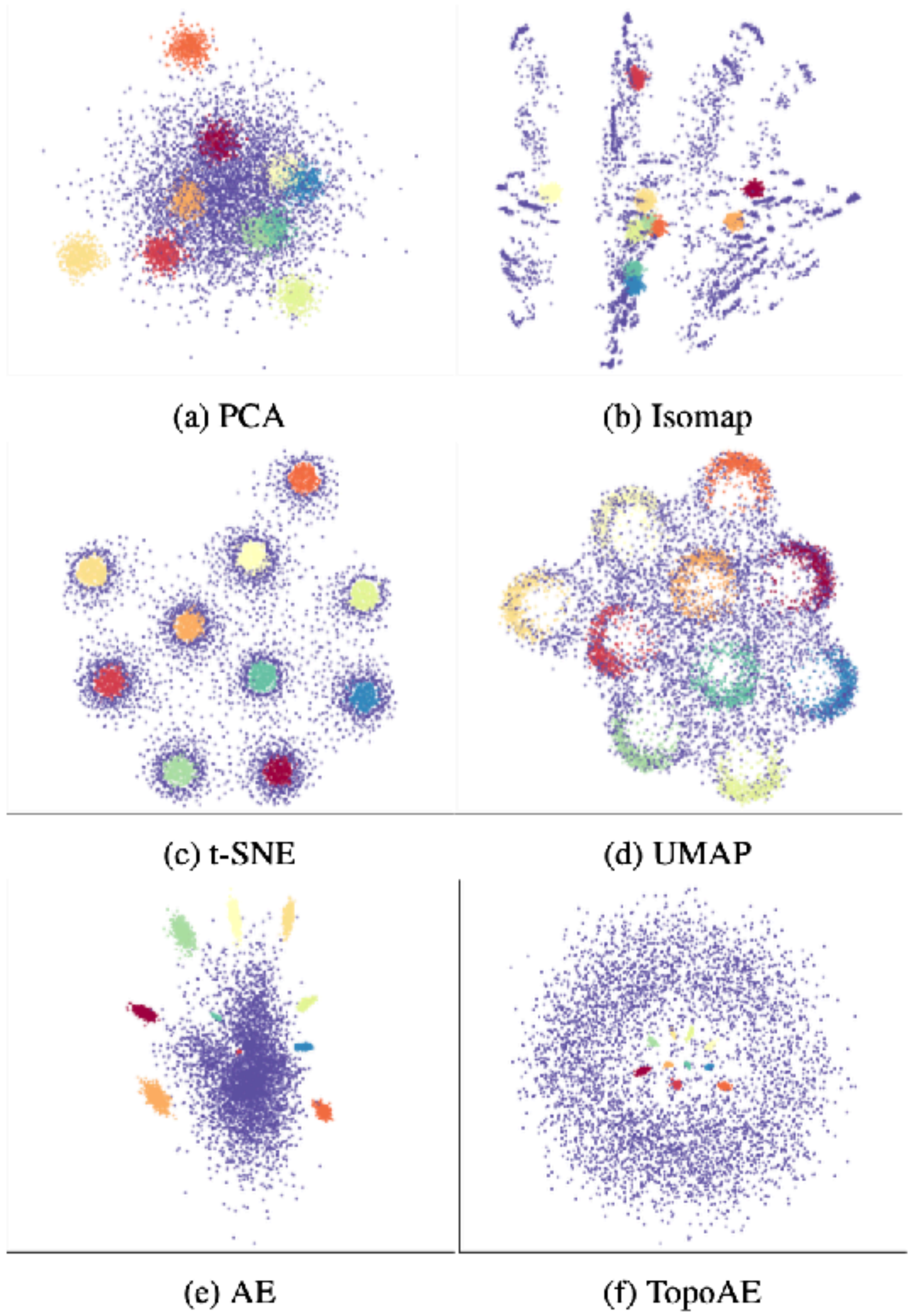
Cosine distance

$$S_C(A,B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}}$$

M. Negro - michelanegro@lsu.edu

Unsupervised dimensionality reduction

- We want to explore unsupervised deep learning approaches
- Dimensionality reduction algorithm
  - We started with **UMAP** McInnes et al.
    - Fast (>> faster than t-SNE)
    - Good scaling in terms of both dataset size and dimensionality.
    - better preserve the global structure of the data

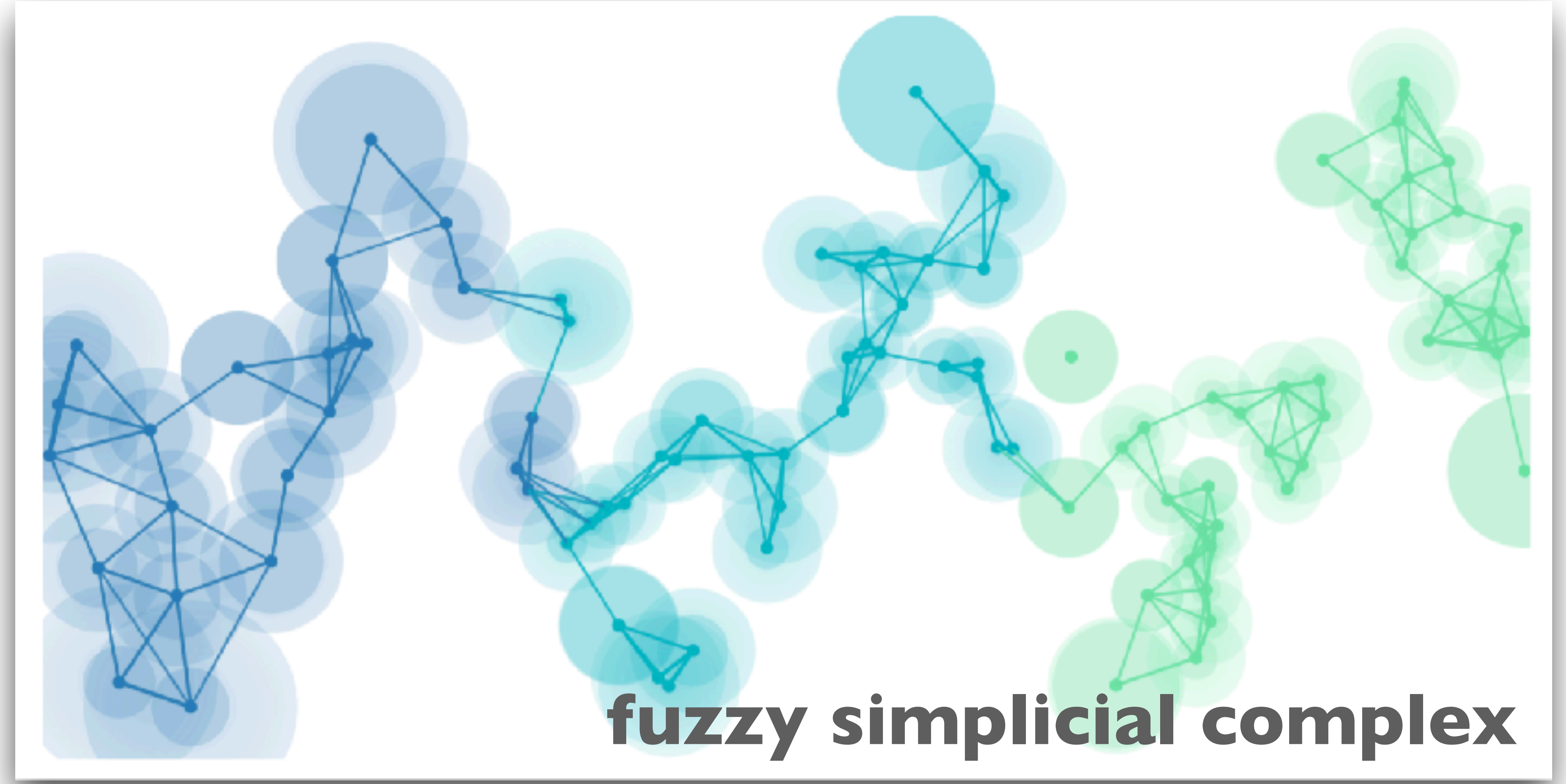


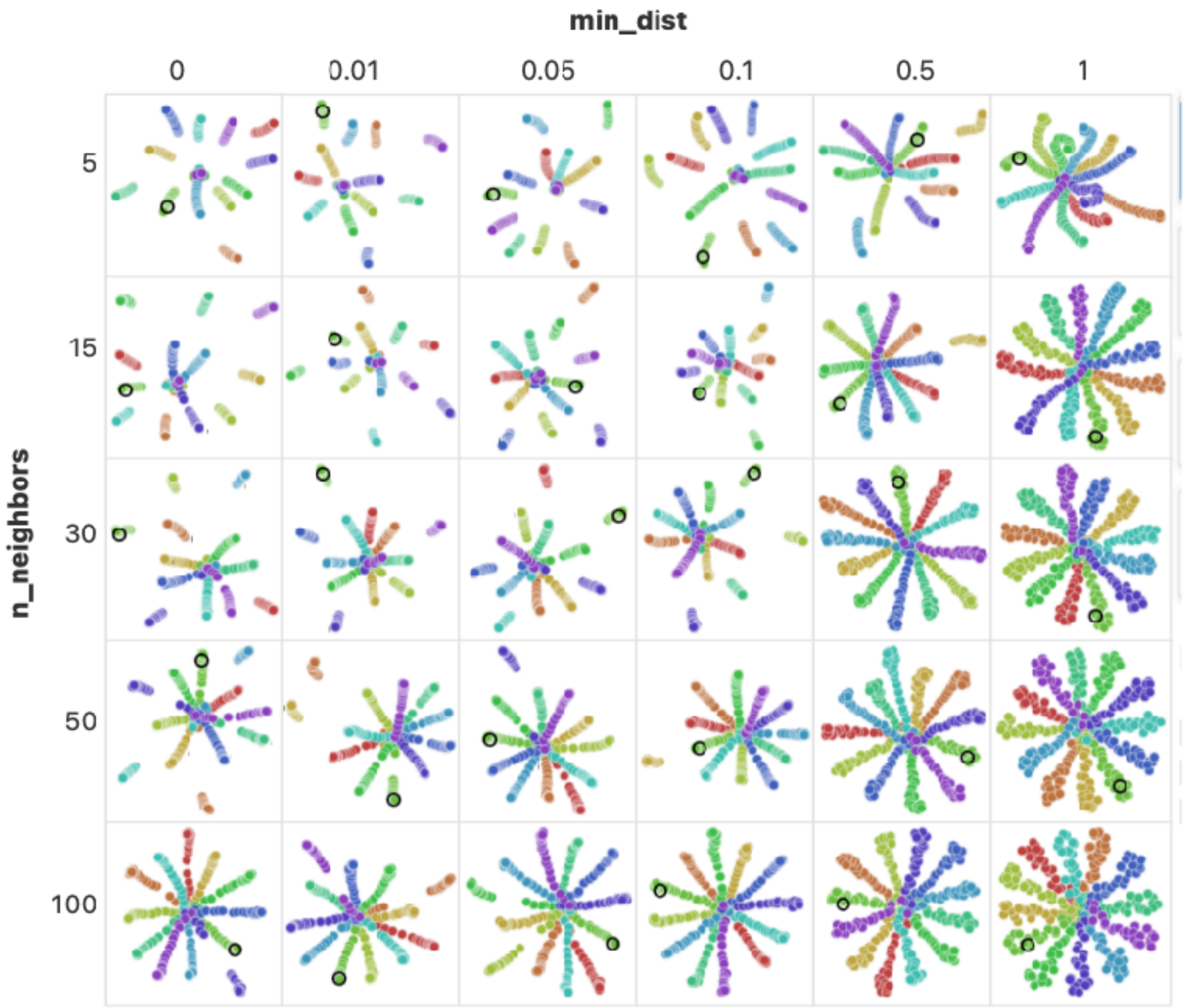
UMAP

UMAP uses weighted graph layout algorithms to arrange data in low-dimensional space

First step is building a “fuzzy simplicial complex”: a weighted graph, with edge weights representing the likelihood that two points are connected

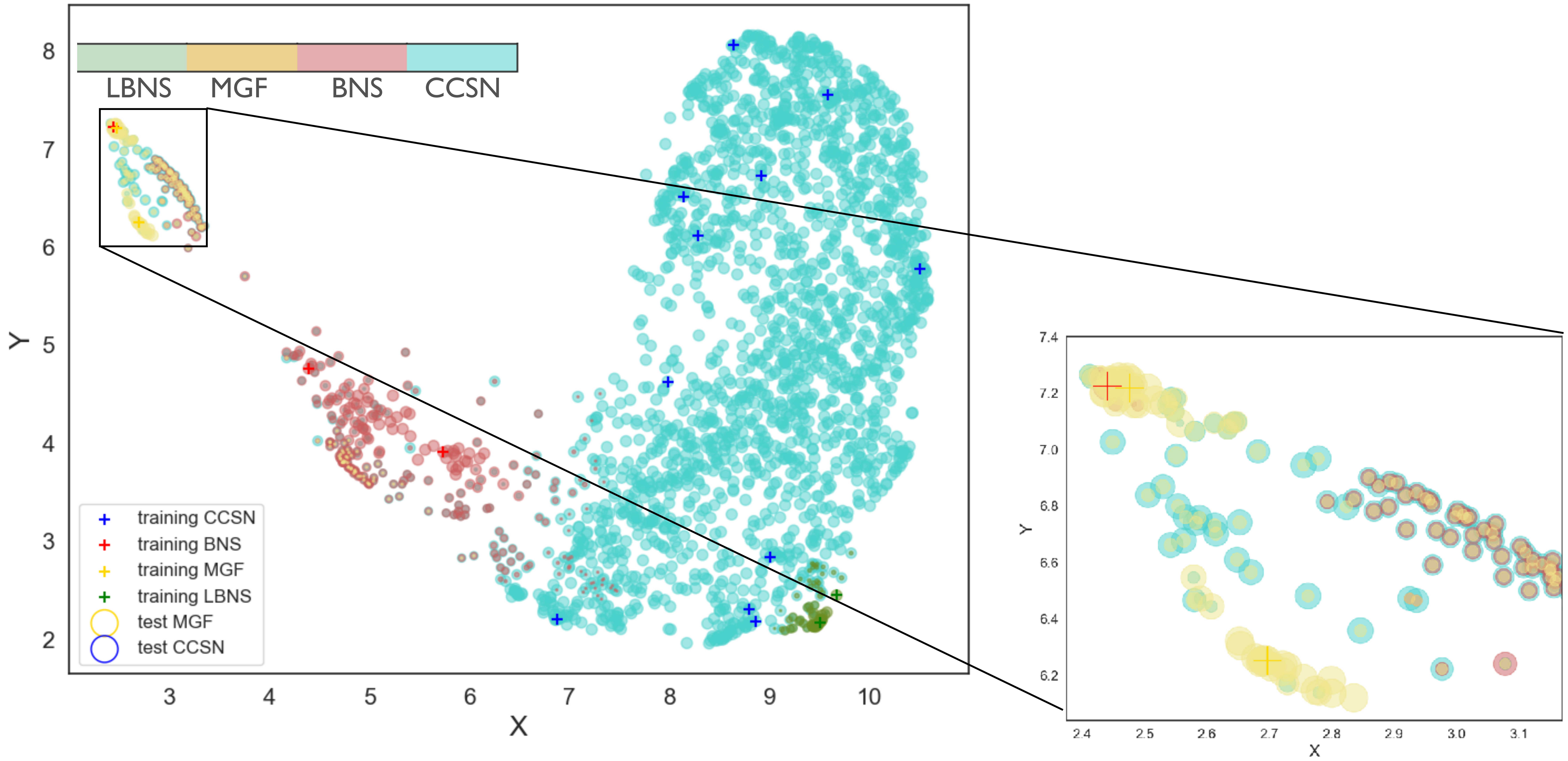
`n_neighbors` and `min_dist`, are effectively used to control the balance between local and global structure in the final projection.





https://pair-code.github.io/understanding-umap/#:~:text=Most importantly, UMAP is fast,-learn's t-SNE implementation.

Towards the classification: semi-supervised classification



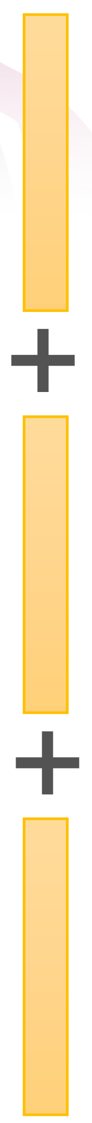
M. Negro - michelanegro@lsu.edu

Domentionality reduction with UMAP

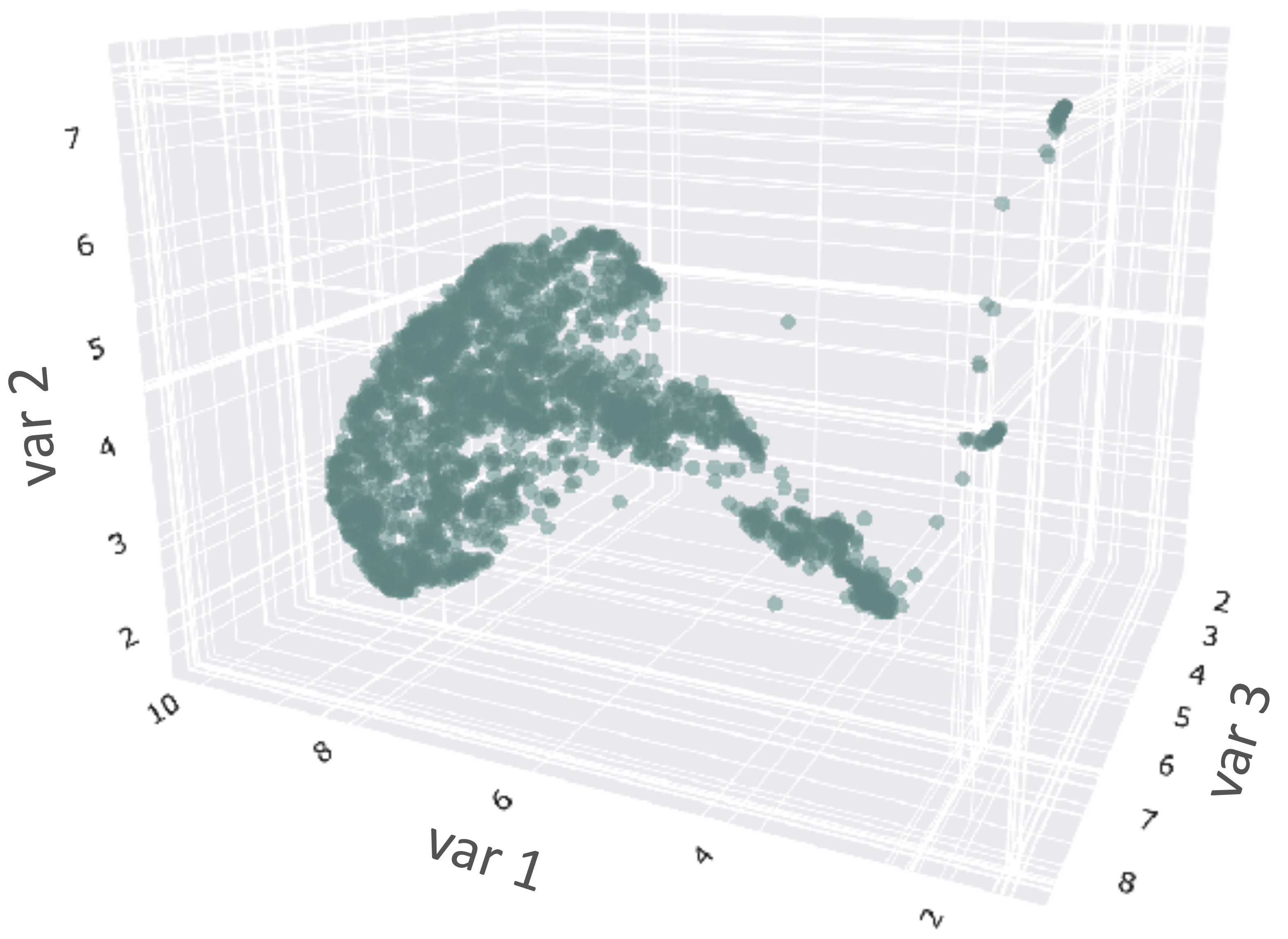
We further reduce the dimensionality of the Latent Space using UMAP.
We look at the 3D, 2D and 1D visualization

Uniform Manifold Approximation and Projection for Dimension Reduction

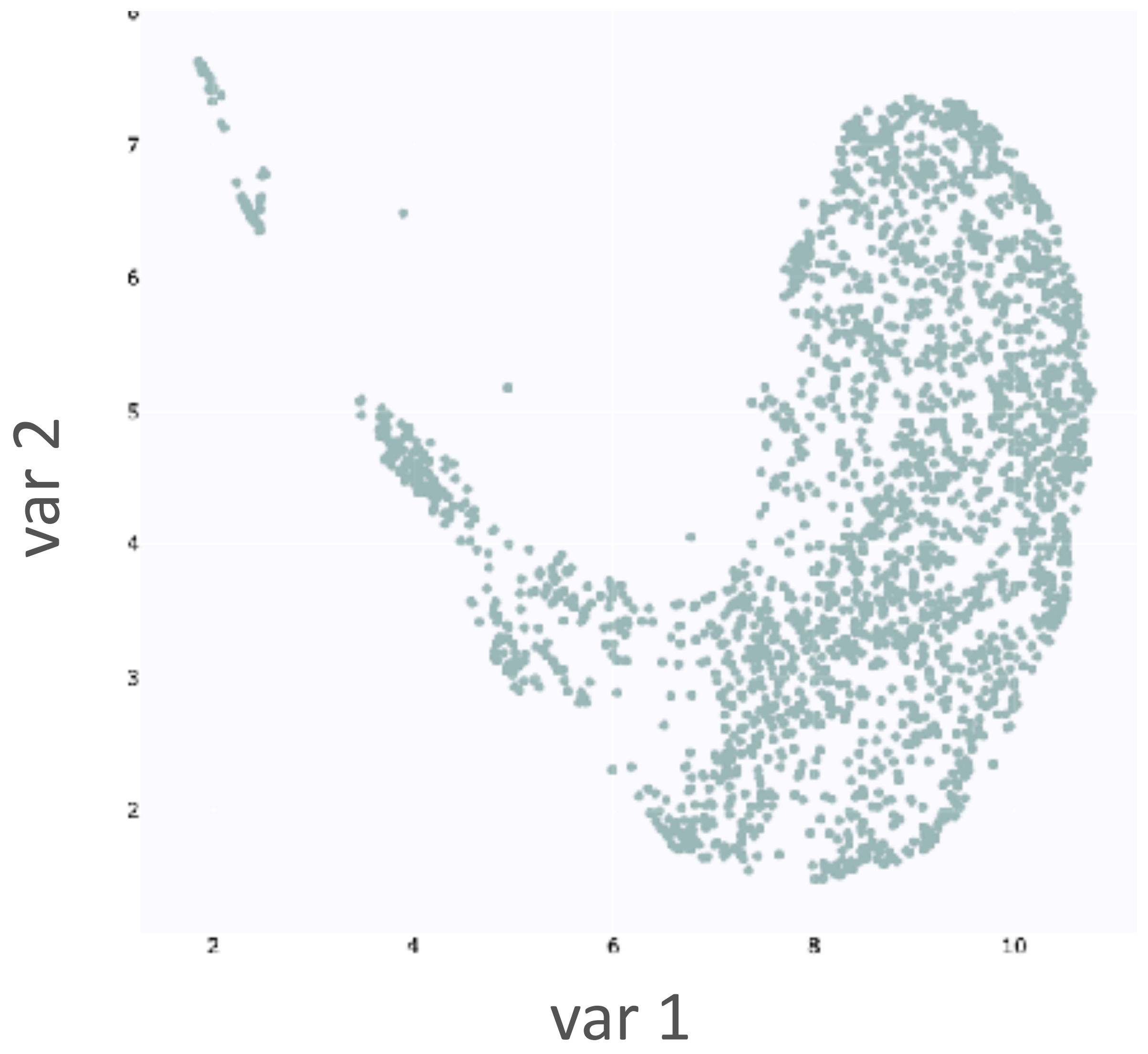
N=30



N=3

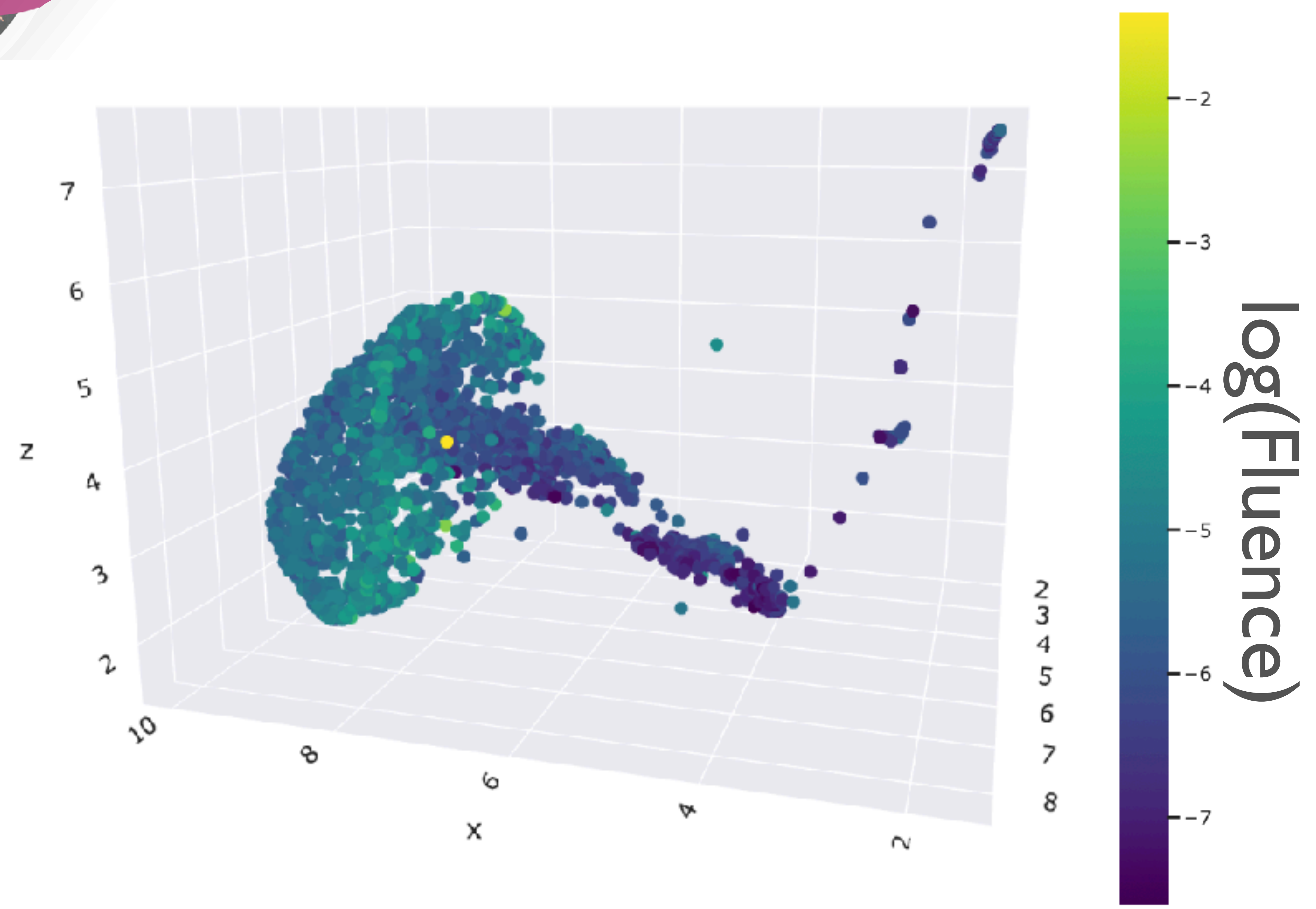


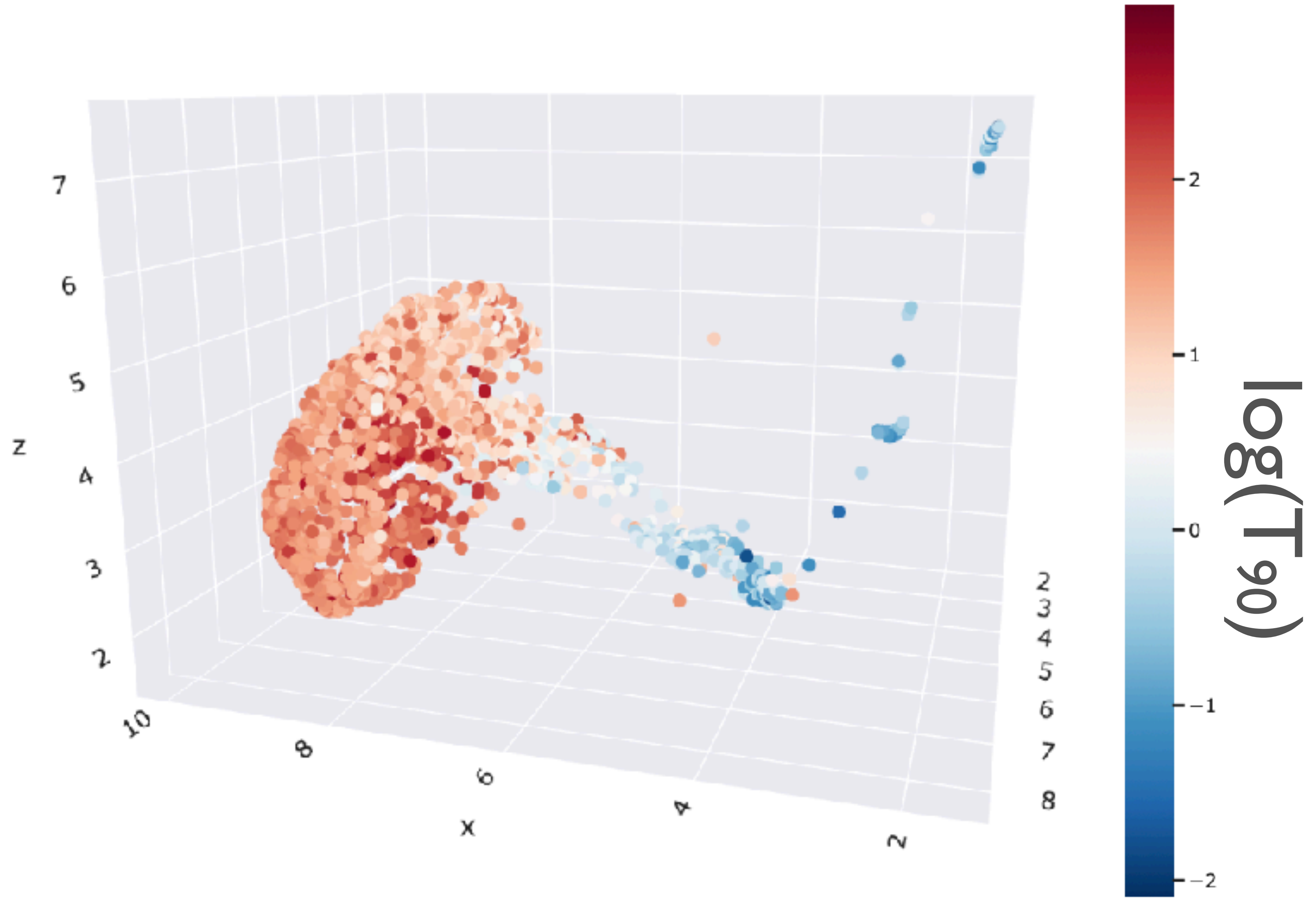
N=2



M. Negro - michelanegro@lsu.edu

Preliminary results and considerations





M. Negro - michelanegro@lsu.edu

Autoencoader

```
class autoencoder(nn.Module):
    def __init__(self, in_channels, n_e=10, xlen=9376, h1=512, h2=64):
        super(autoencoder, self).__init__()

        k2=int((xlen%8)/2.+3)

        self.conv1=nn.Conv2d(in_channels=in_channels, out_channels=in_channels*2, kernel_size=4, stride=2, padding=0)
        self.conv2=nn.Conv2d(in_channels=in_channels*2, out_channels=in_channels*3, kernel_size=(1,k2), stride=(1,2), padding=0)
        self.conv3=nn.Conv2d(in_channels=in_channels*3, out_channels=in_channels*3, kernel_size=(2,3), stride=(1,2), padding=0)
        #self.maxpool=nn.MaxPool2d(kernel_size=(2,k2), stride=(1,2), padding=0,return_indices=True)
        self.conv4=nn.Conv2d(in_channels=in_channels*3, out_channels=in_channels*4, kernel_size=(2,3), stride=(1,2), padding=0)
        self.linear1=nn.Linear(int((xlen/8 - (1+k2)/4)/2.-((xlen%8)/4+0.5))*in_channels*4, h1)
        self.linear2=nn.Linear(h1, h2)
        self.linear3=nn.Linear(h2, n_e)
        self.linear4=nn.Linear(n_e, h2)
        self.linear5=nn.Linear(h2, h1)
        self.linear6=nn.Linear(h1, int((xlen/8 - (1+k2)/4)/2.-((xlen%8)/4+0.5))*in_channels*4)
        self.unflatten=nn.Unflatten(1, (in_channels*4,1,int((xlen/8 - (1+k2)/4)/2.-((xlen%8)/4+0.5))))
        self.deconv1=nn.ConvTranspose2d(in_channels=in_channels*4, out_channels=in_channels*3,
                                        kernel_size=(2,3), stride=(1,2), padding=0, output_padding=(0,k2-3))
        self.deconv2=nn.ConvTranspose2d(in_channels=in_channels*3, out_channels=in_channels*3, kernel_size=(2,3),
                                        stride=(1,2), padding=0)
        #self.unpool=nn.MaxUnpool2d(kernel_size=(2,k2), stride=(1,2), padding=0)
        self.deconv3=nn.ConvTranspose2d(in_channels=in_channels*3, out_channels=in_channels*2, kernel_size=(1,k2), stride=(1,2), padding=
        self.deconv4=nn.ConvTranspose2d(in_channels=in_channels*2, out_channels=in_channels, kernel_size=4, stride=2, padding=0)
```

Autoencoader

```
def encoder(self,x):
        x=F.leaky_relu(self.conv1(x))
        #x,index=self.maxpool(x)
        x=F.leaky_relu(self.conv2(x))
        x=F.leaky_relu(self.conv3(x))
        x=F.leaky_relu(self.conv4(x))
        x=torch.flatten(x,1)
        x=F.leaky_relu(self.linear2(F.leaky_relu(self.linear1(x))))
        x=self.linear3(x)
        return x
```

```
    def decoder(self,x):
        x=F.leaky_relu(self.linear4(x))
        x=F.leaky_relu(self.linear6(F.leaky_relu(self.linear5(x))))
        x=self.unflatten(x)
        x=F.leaky_relu(self.deconv1(x))
        #x=self.unpool(x,index)
        x=F.leaky_relu(self.deconv2(x))
        x=F.leaky_relu(self.deconv3(x))
        x=F.relu(self.deconv4(x)) #F.leaky_relu(self.deconv4(x))
        return x
```

```
    def forward(self, x):
        x = self.encoder(x)
        x = self.decoder(x)
        return x
```

LeakyRELU

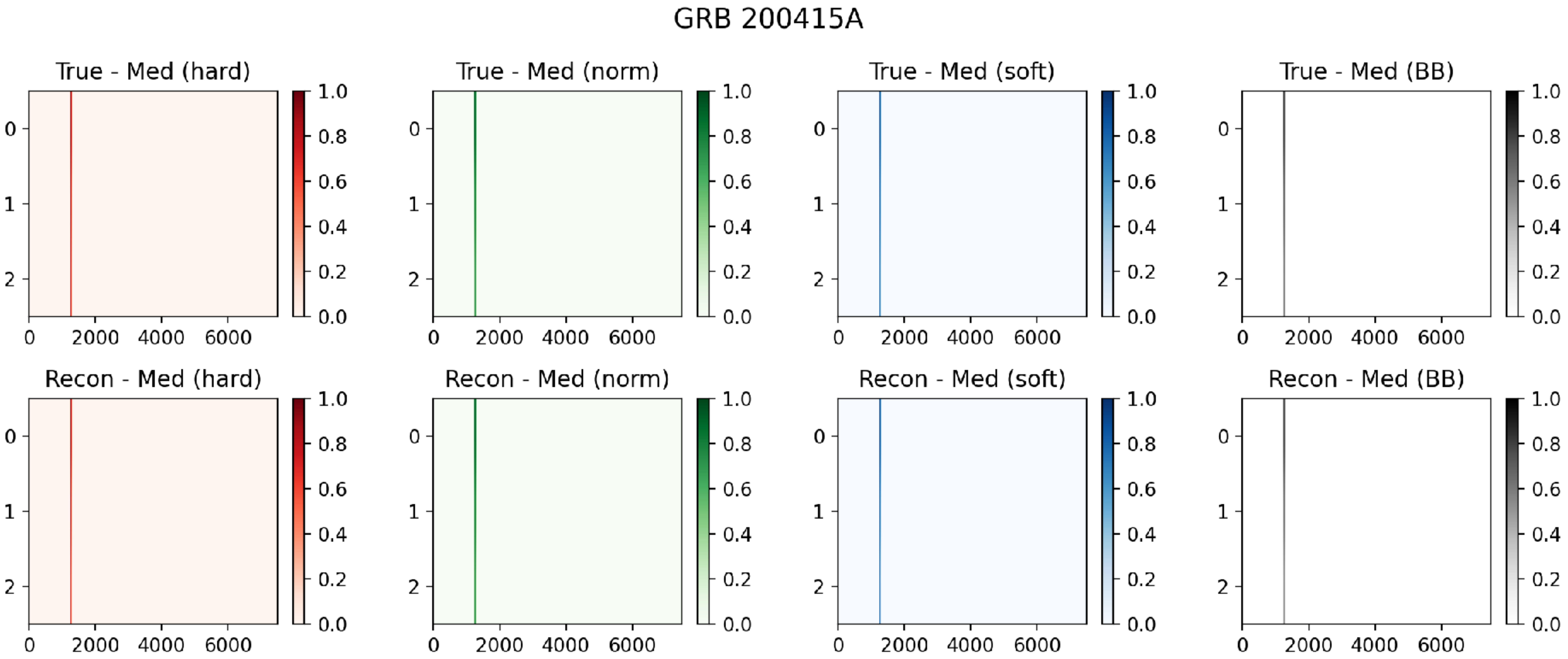
https://pytorch.org/docs/stable/generated/torch.nn.LeakyReLU.html#torch.nn.LeakyReLU

M. Negro - michelanegro@lsu.edu

Reconstructed vs original waterfalls

M. Ne

Reconstructed vs original waterfalls

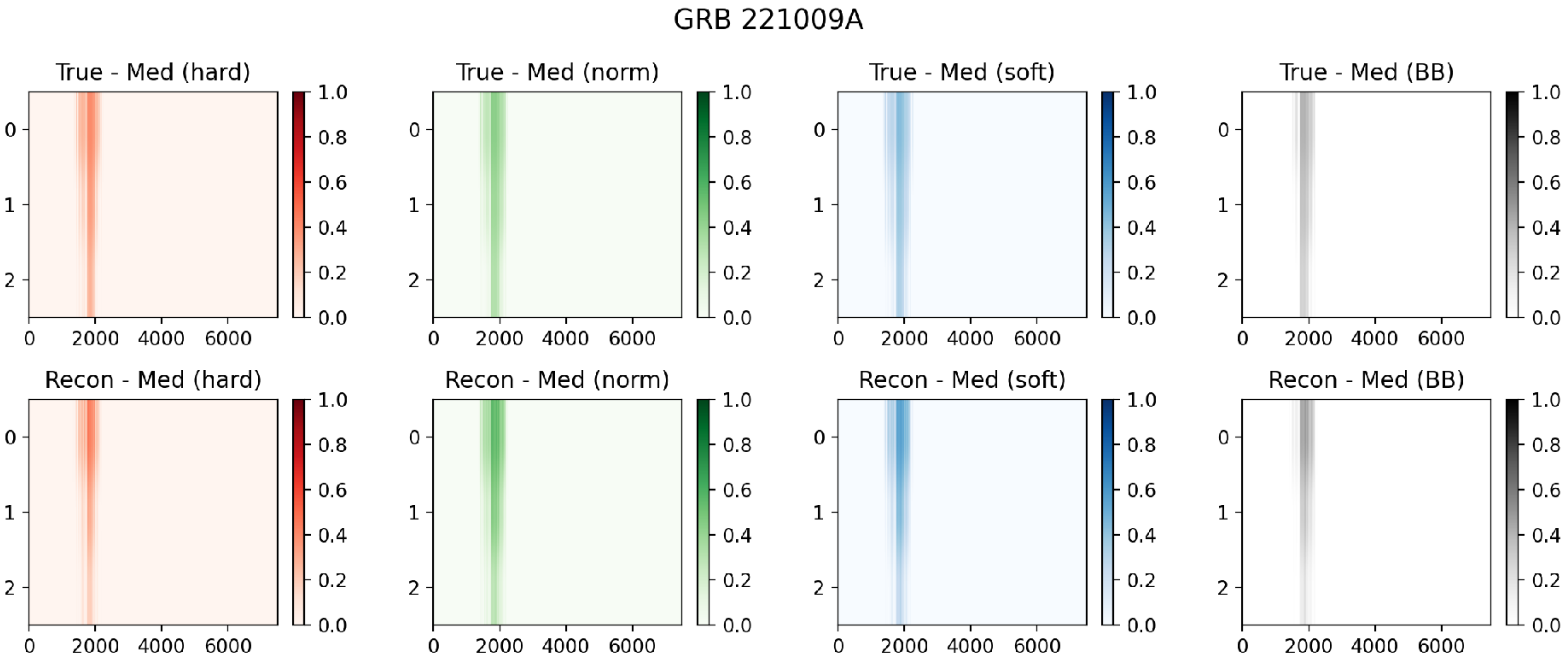


Reconstructed vs original waterfalls

M. Neg

Reconstructed vs original waterfalls

Reconstructed vs original waterfalls



M. Negr

Reconstructed vs original waterfalls

M. Negi