A multidimensional, event-by-event, statistical weighting procedure for signal to background separation

26^{th} ***International Conference CHEP — Norfolk, Va***

Physics Analysis Tools

Zachary Baldwin, May 8, 2023
for the GlueX Collaboration and
Carnegie Mellon University

Most common issue in many areas of research

- Separating regions of signal from background

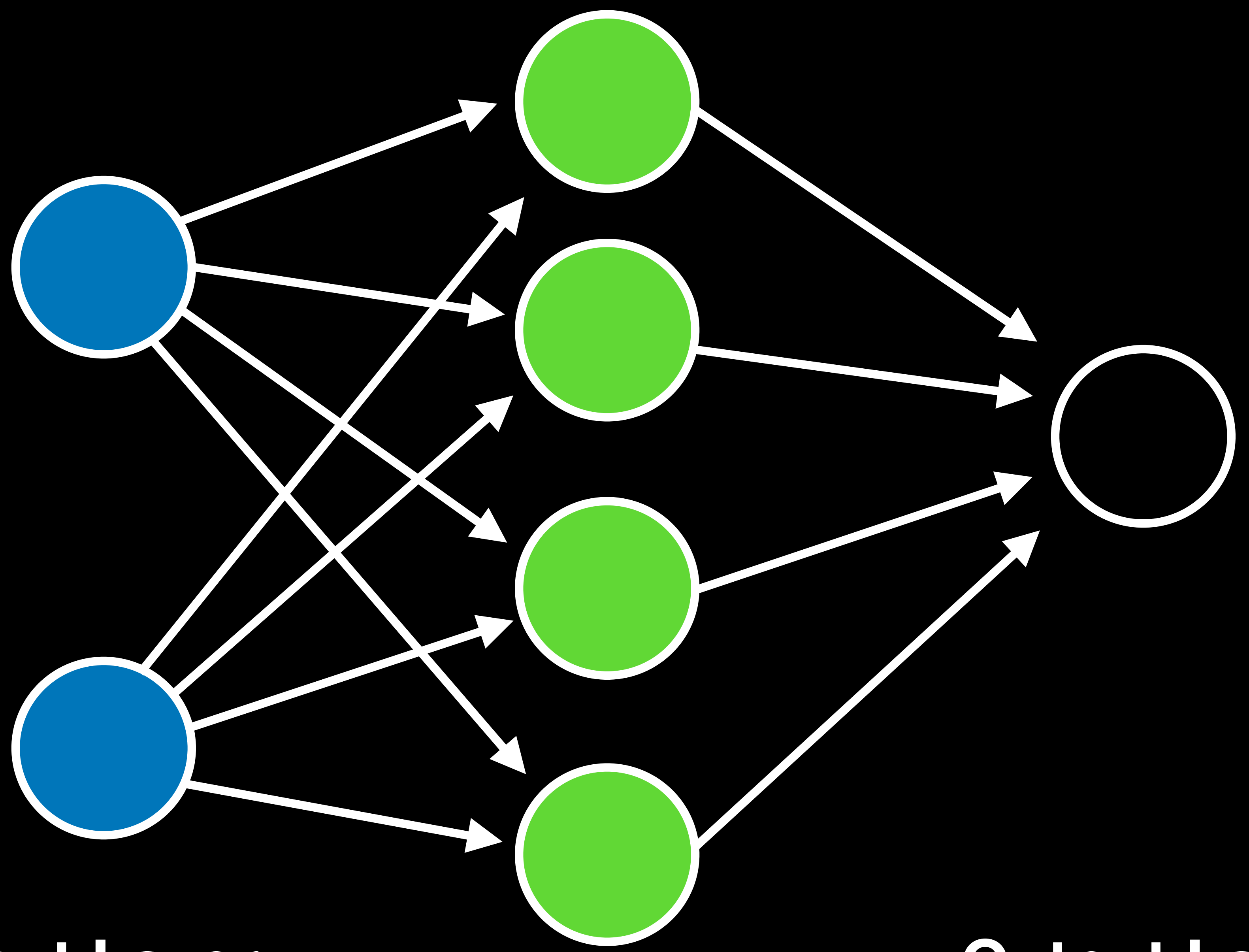
Solution? → Completely ignore the implications of keeping the background and just selecting around the region of interest

Most common issue in many areas of research

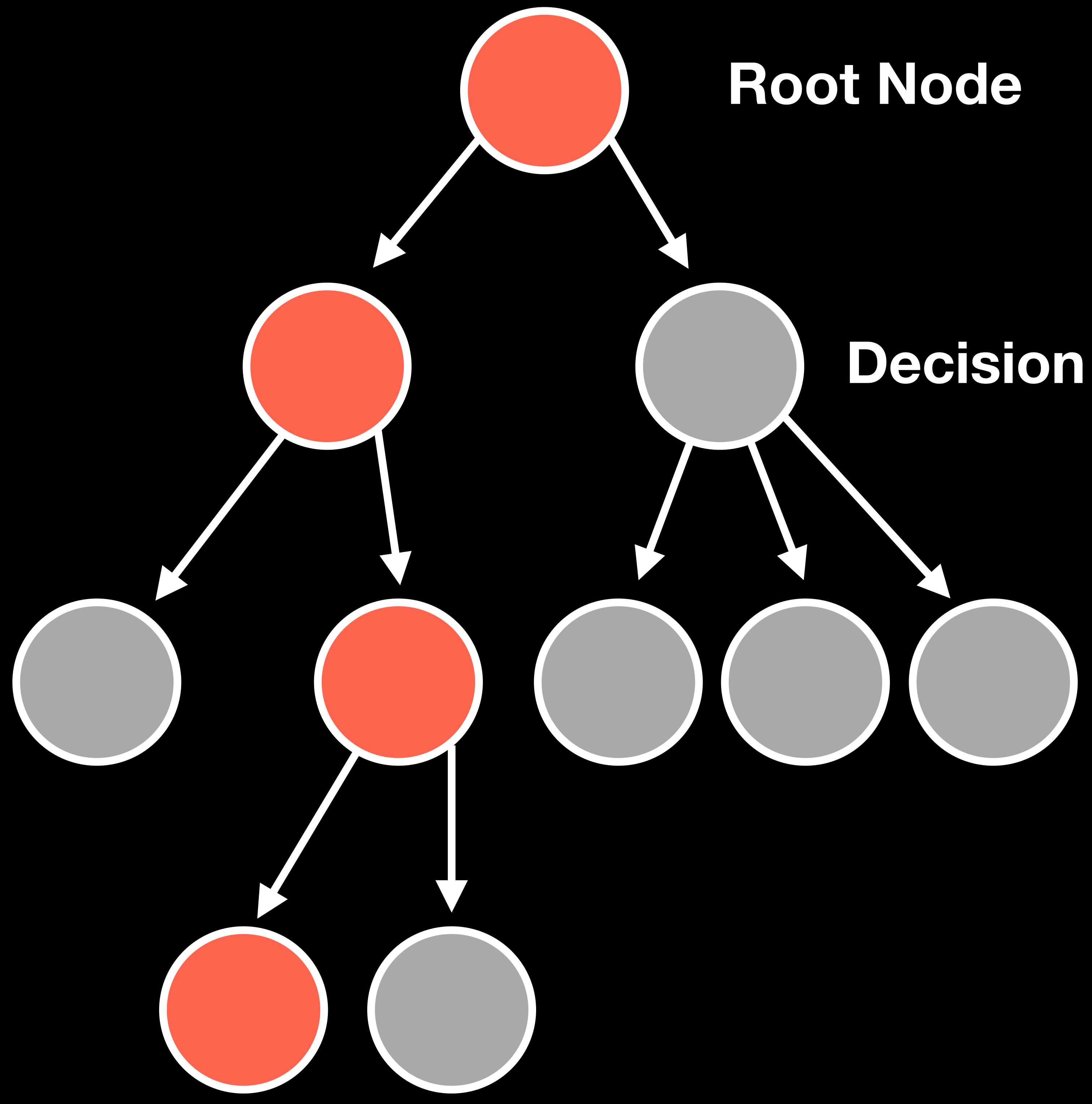
- Separating regions of signal from background

Solution? → Completely ignore the implications of keeping the background and just selecting around the region of interest

Neural Network



Decision Trees

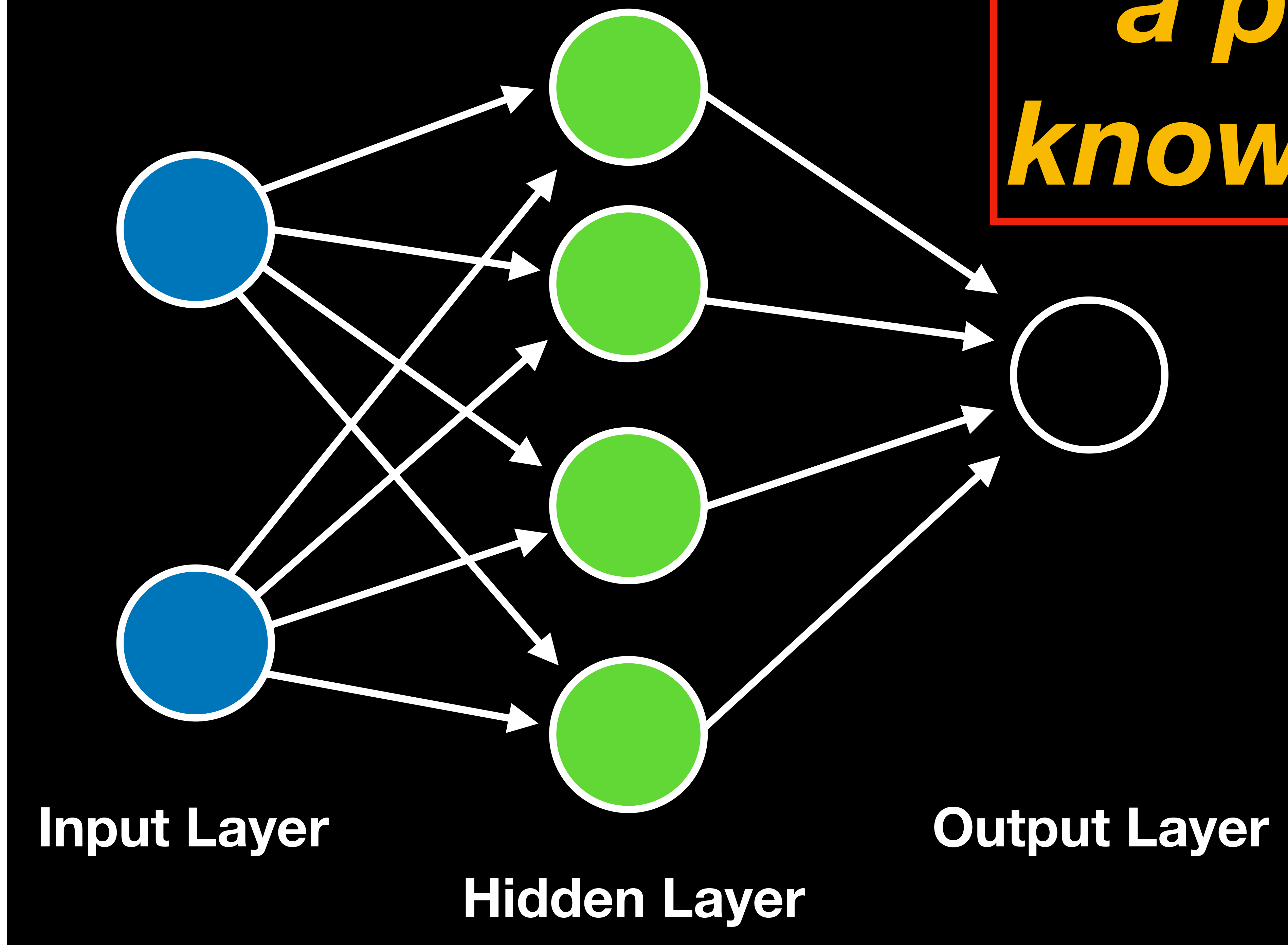


Most common issue in many areas of research

- Separating regions of signal from background

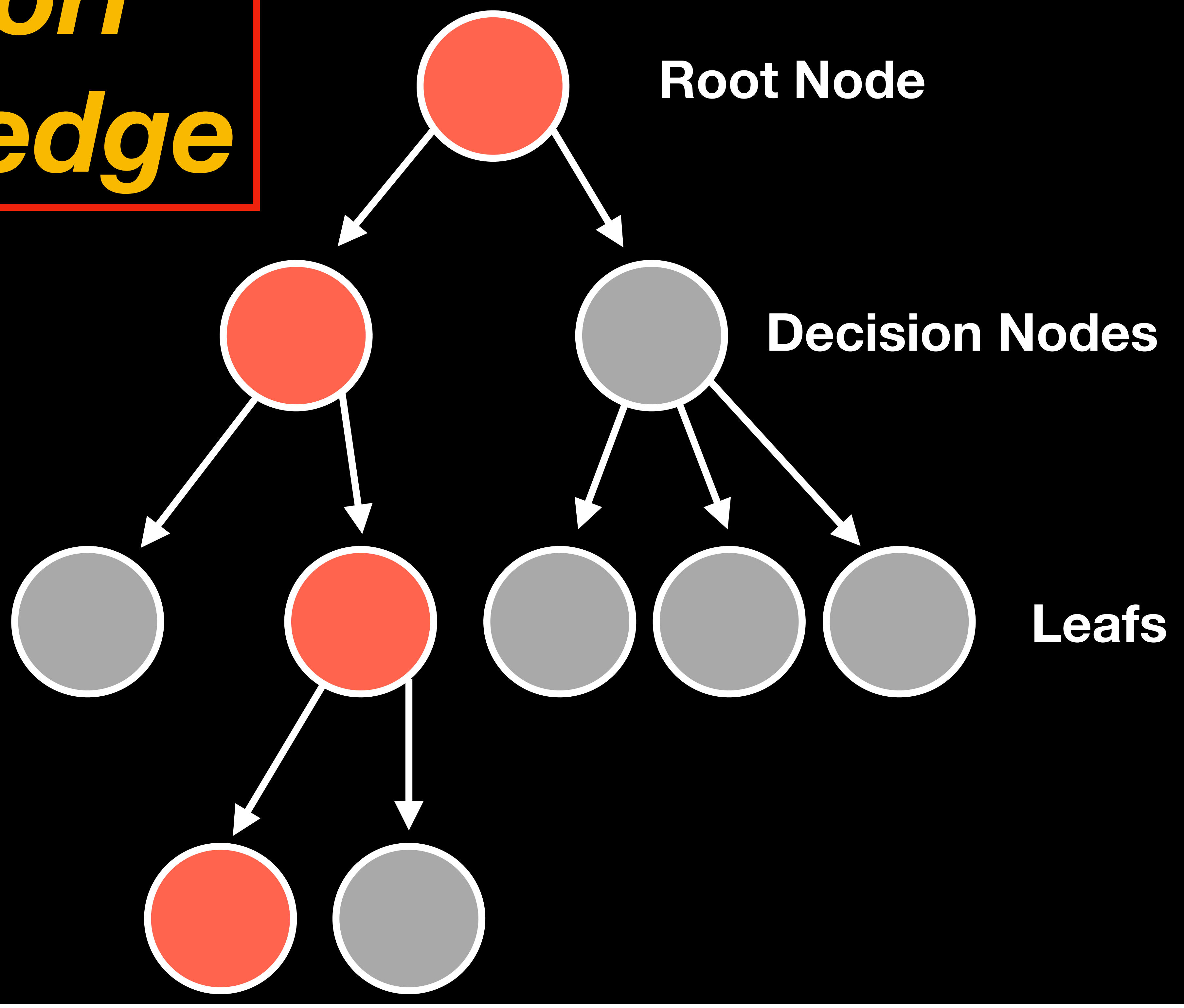
Solution? → Completely ignore the implications of keeping the background and just selecting around the region of interest

Neural Network



a priori knowledge

Decision Trees



What if distributions for signal and background are unknown and/or are irreducible?

Take for instance the background underneath $\gamma p \to p\eta$ (or $\gamma p \to p\omega$). Other production mechanisms can produce the same final state so can not differentiate between pure signal events using selection criteria therefore is *irreducible*

PDG values

η		ω	
$m(\eta) = 547.51\ MeV$		$m(\omega) = 782.66\ MeV$	
$\eta \to \gamma\gamma$	39 %	$\omega \to \pi^0\pi^+\pi^-$	89 %
$\eta \to \pi^0\pi^0\pi^0$	32 %	$\omega \to \pi^0\gamma$	8 %
$\eta \to \pi^0\pi^+\pi^-$	23 %	$\omega \to \pi^+\pi^-$	2 %

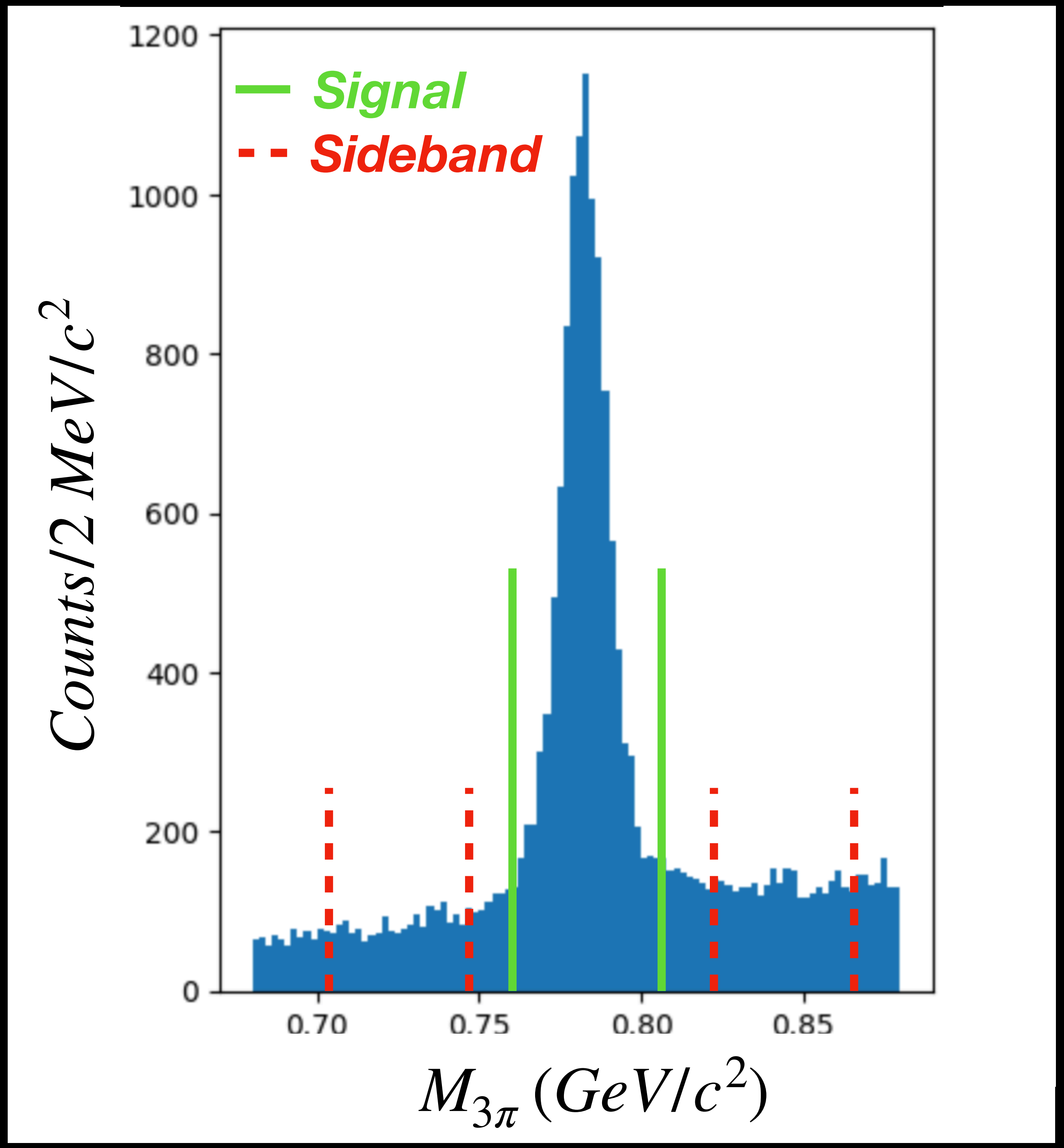
Take for instance the background underneath $\gamma p \to p\eta$ (or $\gamma p \to p\omega$). Other production mechanisms can produce the same final state so can not differentiate between pure signal events using selection criteria therefore is *irreducible*

PDG values

η		ω	
$m(\eta) = 547.51\ MeV$		$m(\omega) = 782.66\ MeV$	
$\eta \to \gamma\gamma$	39 %	$\omega \to \pi^0\pi^+\pi^-$	89 %
$\eta \to \pi^0\pi^0\pi^0$	32 %	$\omega \to \pi^0\gamma$	8 %
$\eta \to \pi^0\pi^+\pi^-$	23 %	$\omega \to \pi^+\pi^-$	2 %

Data outside the region of interest can be subtracted from inside the signal region

Toy Monte Carlo



$$\Phi_{Subtracted} = \Phi_{Signal} - \frac{A_{Signal}}{A_{Left} + A_{Right}} \left(\Phi_{Left} + \Phi_{Right}\right)$$

Can be limited in use!

Developed during analysis of $\eta^{(\prime)}$ and ω photo-production in CLAS

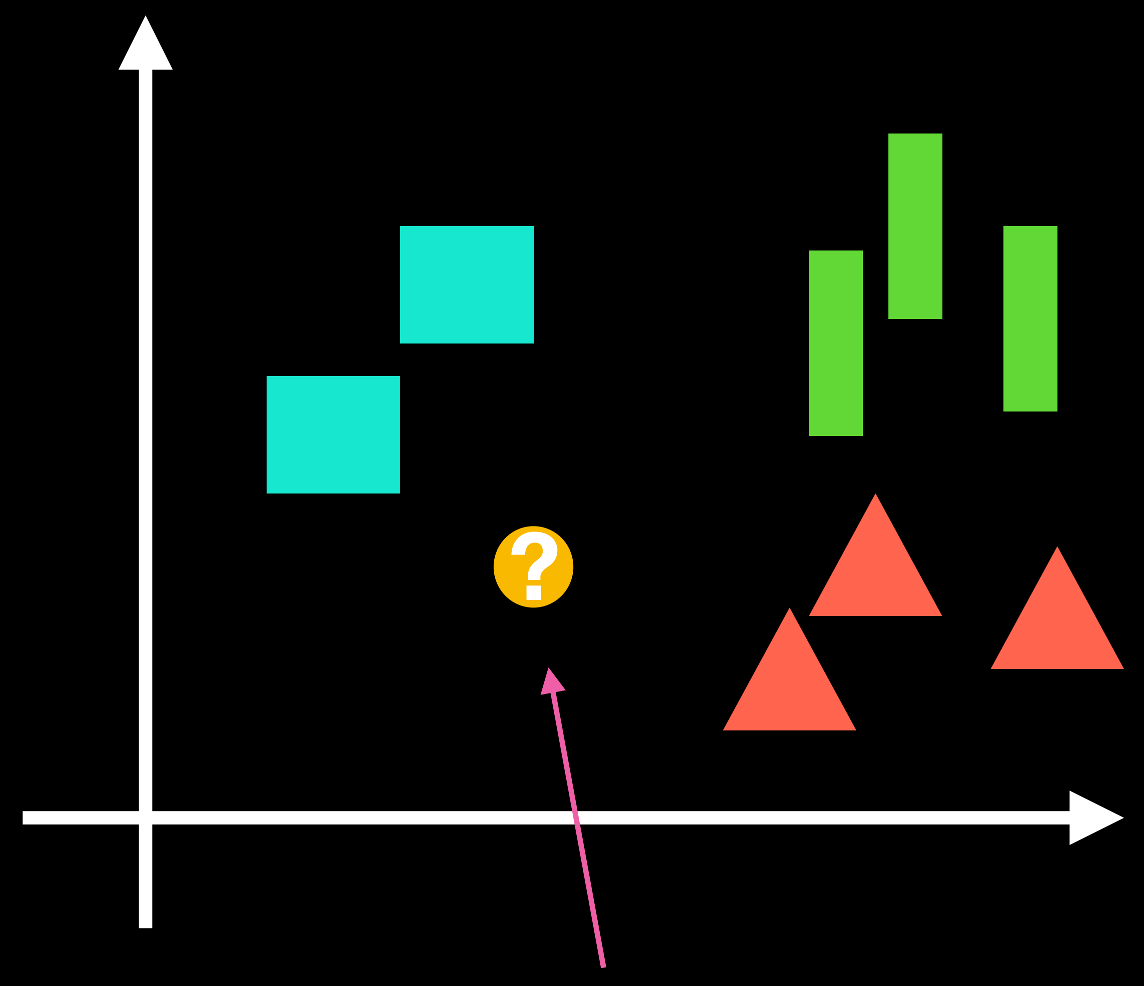
Generalizes sideband subtraction method to higher dimensions (no binning required)

What is the procedure?

Utilizes k-nearest neighbor technique to assign each signal candidate a Quality (Q) Factor (i.e. the probability that the event originates from desired signal)

What are K-Nearest Neighbors?

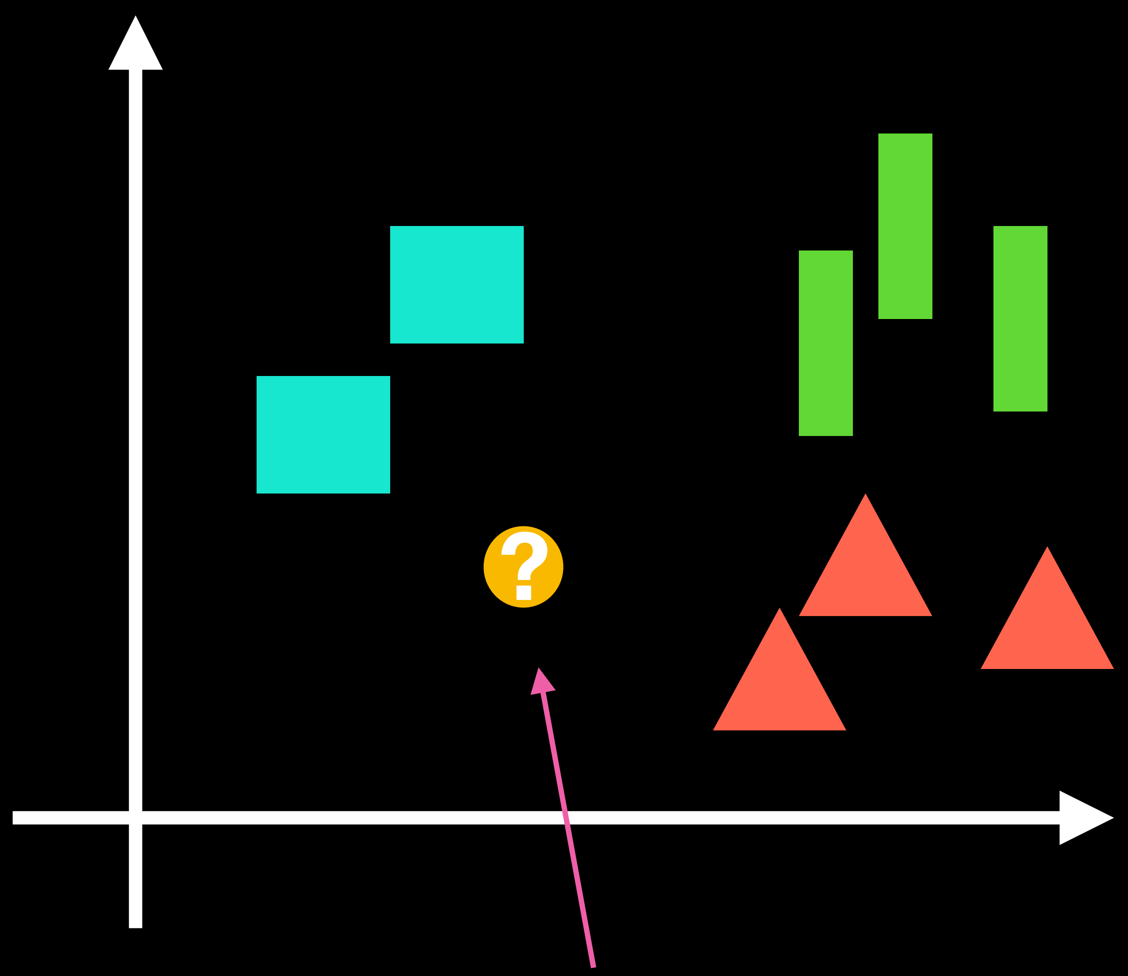
- Algorithm to look at data surrounding specific target data point, in order to predict what category that data should be



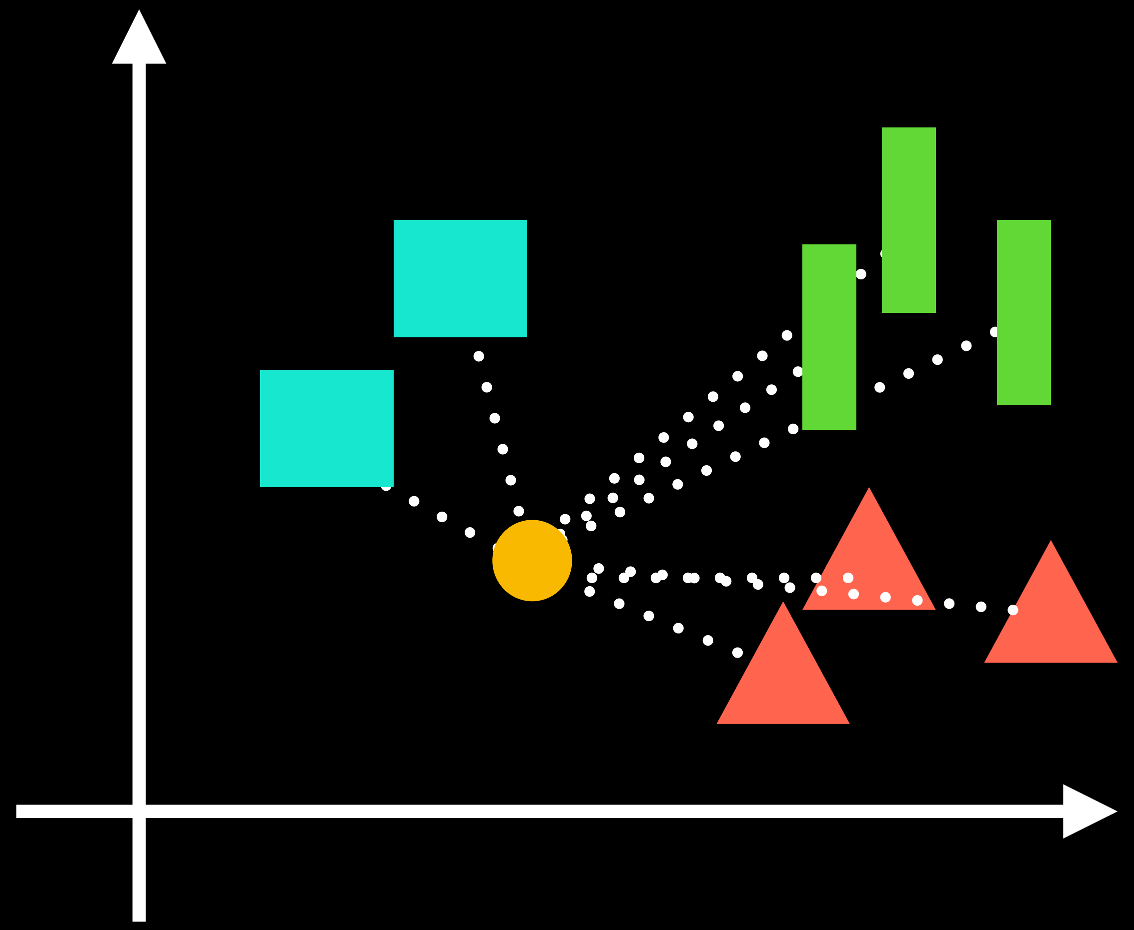
What category should this event belong to?

What are K-Nearest Neighbors?

- Algorithm to look at data surrounding specific target data point, in order to predict what category that data should be



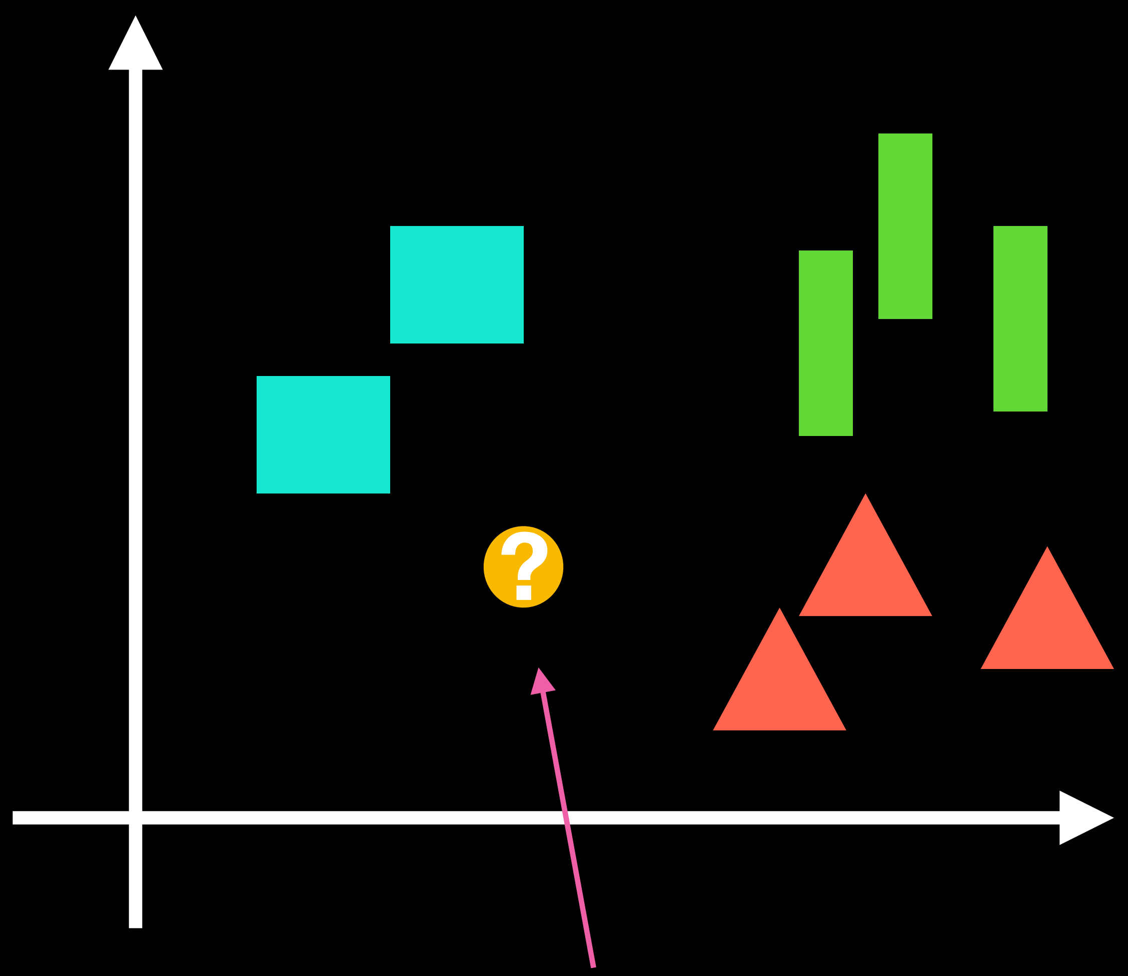
What category should this event belong to?



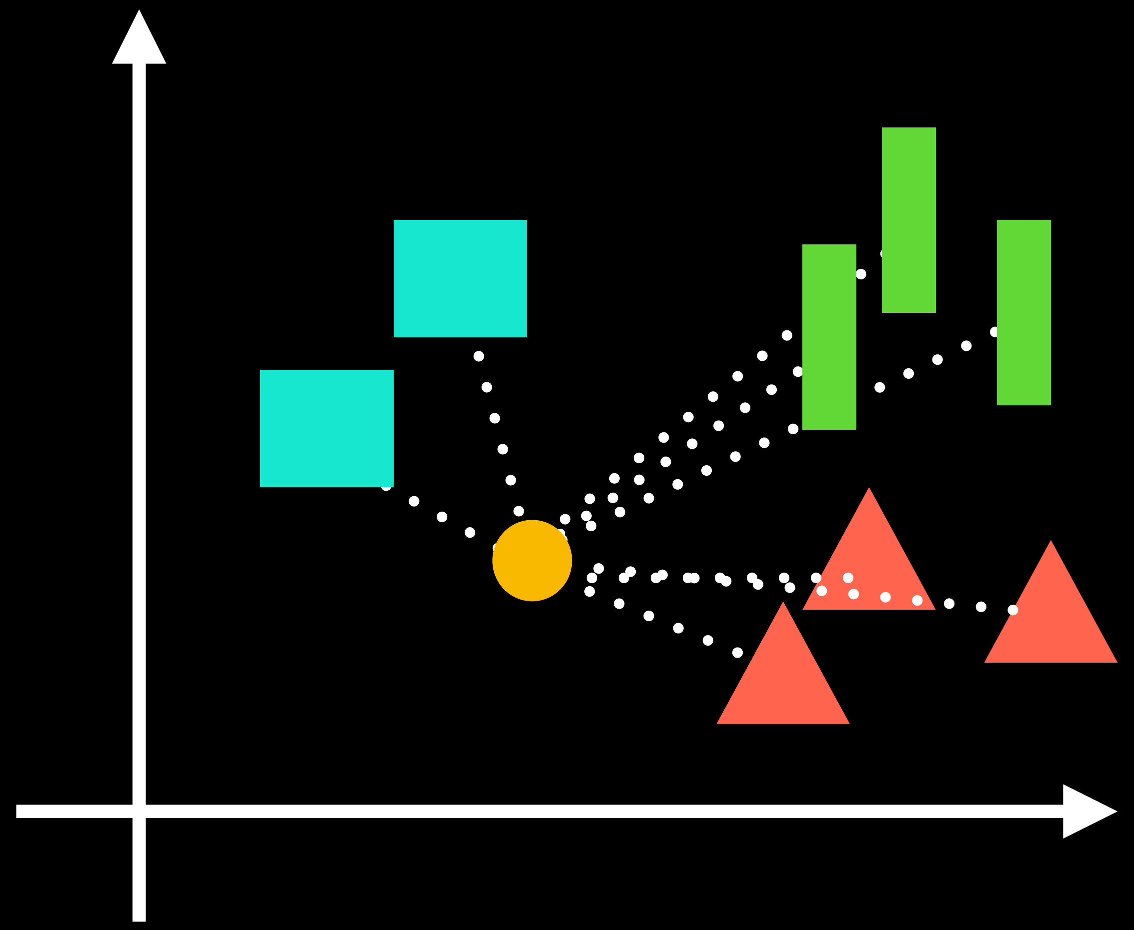
Measure distance to all points

What are K-Nearest Neighbors?

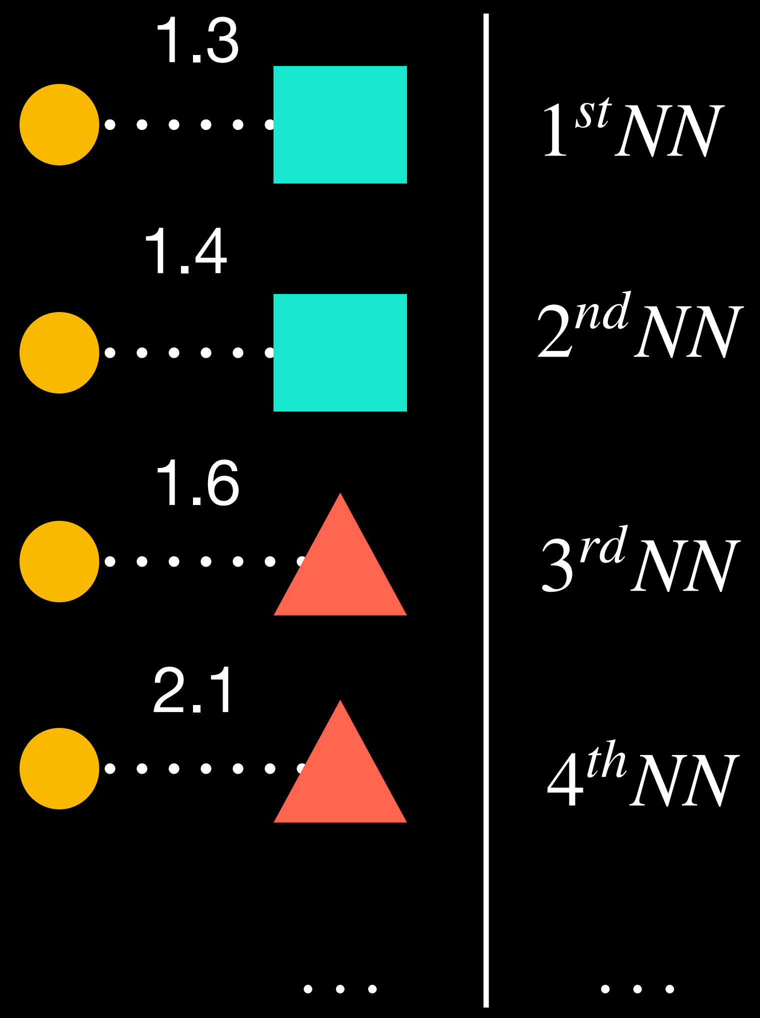
- Algorithm to look at data surrounding specific target data point, in order to predict what category that data should be



What category should this event belong to?



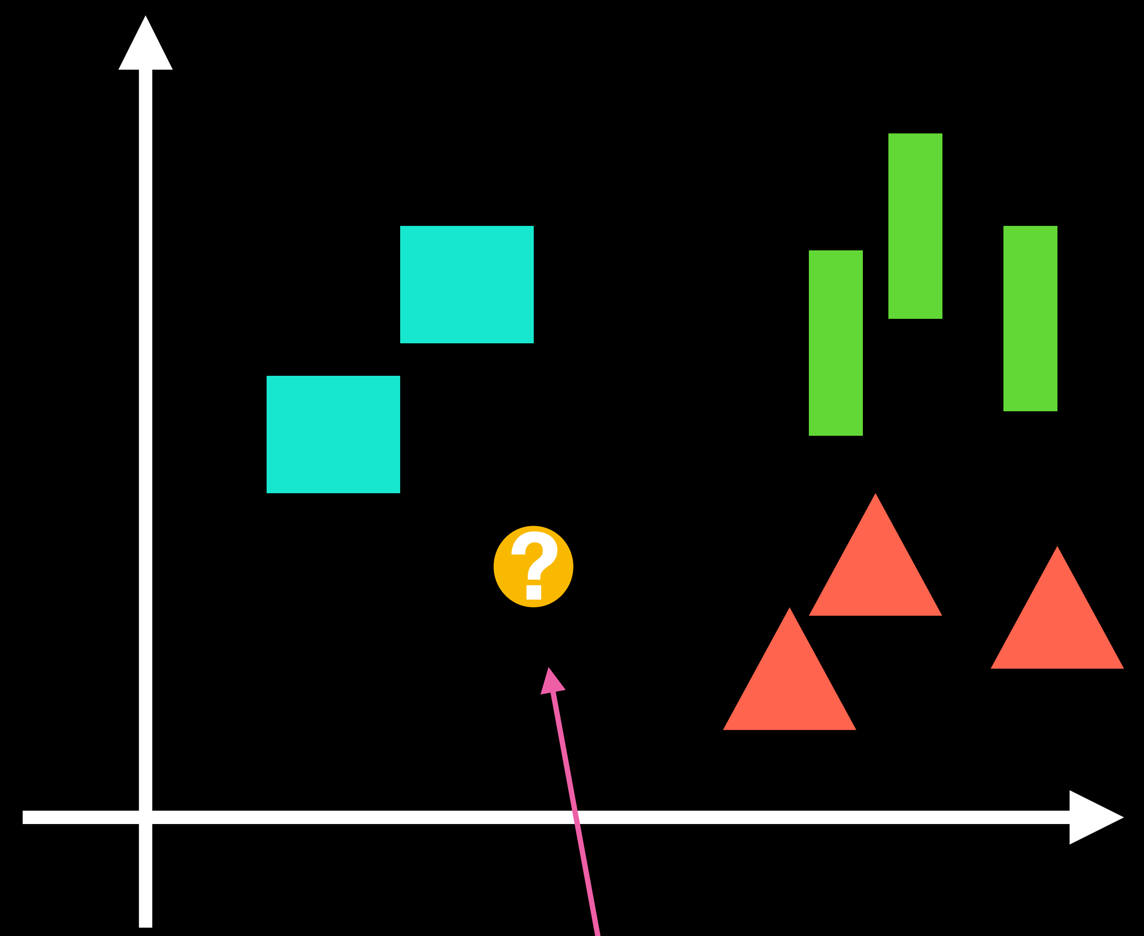
Measure distance to all points



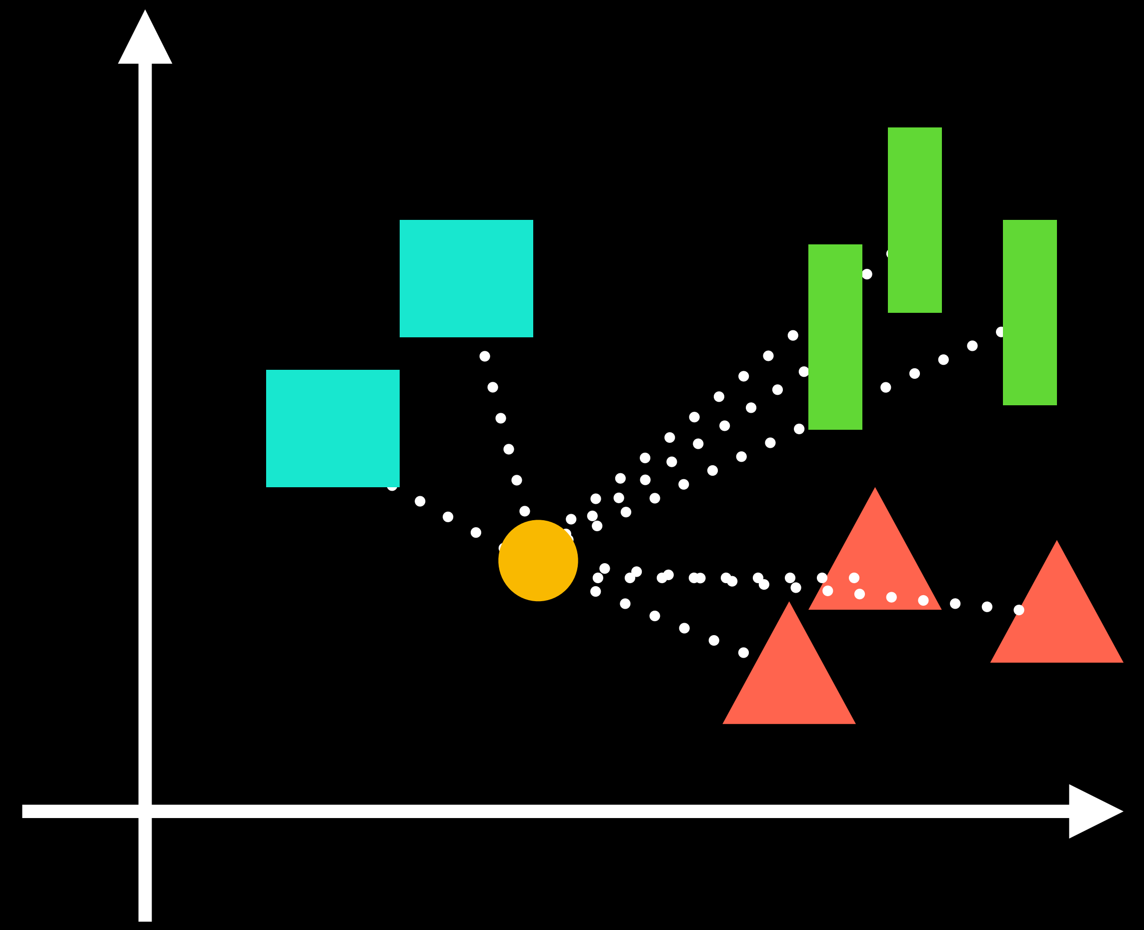
Find the neighbors

What are K-Nearest Neighbors?

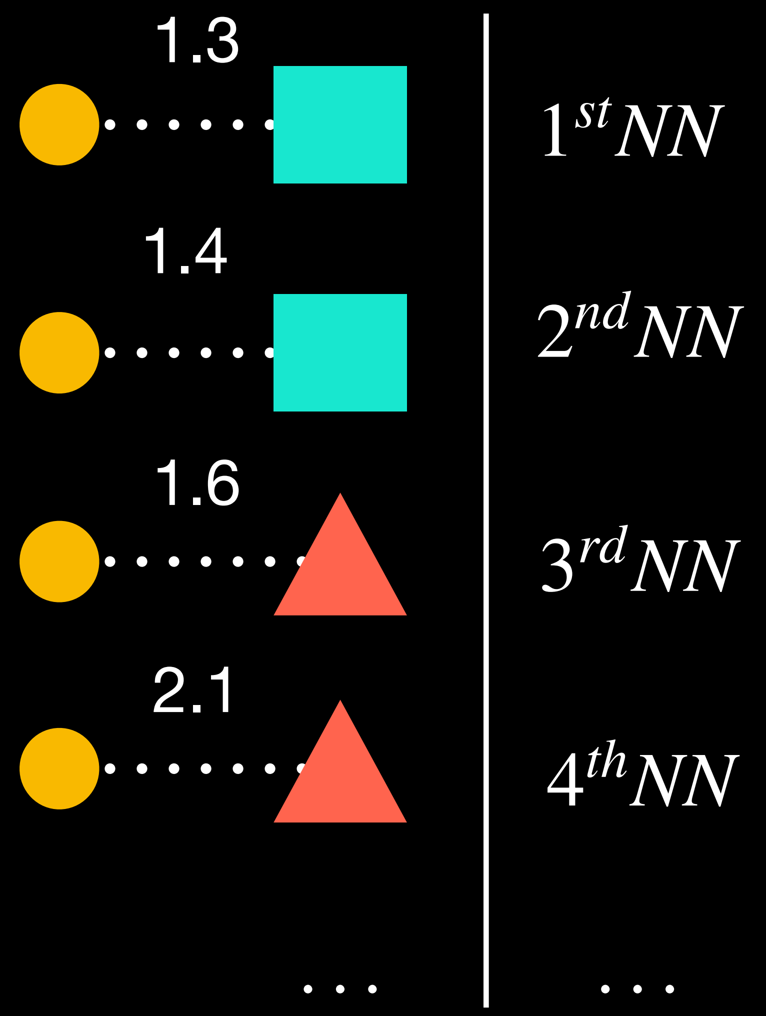
- **Algorithm to look at data surrounding specific target data point, in order to predict what category that data should be**



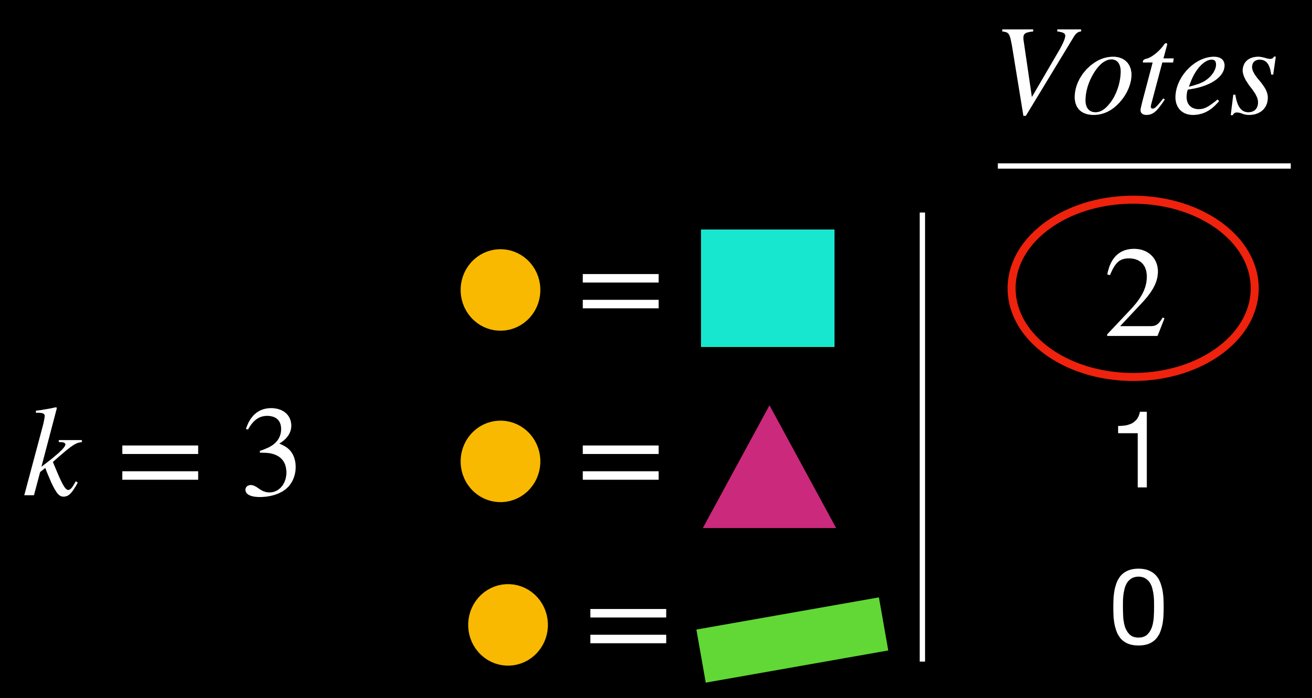
What category should this event belong to?



Measure distance to all points



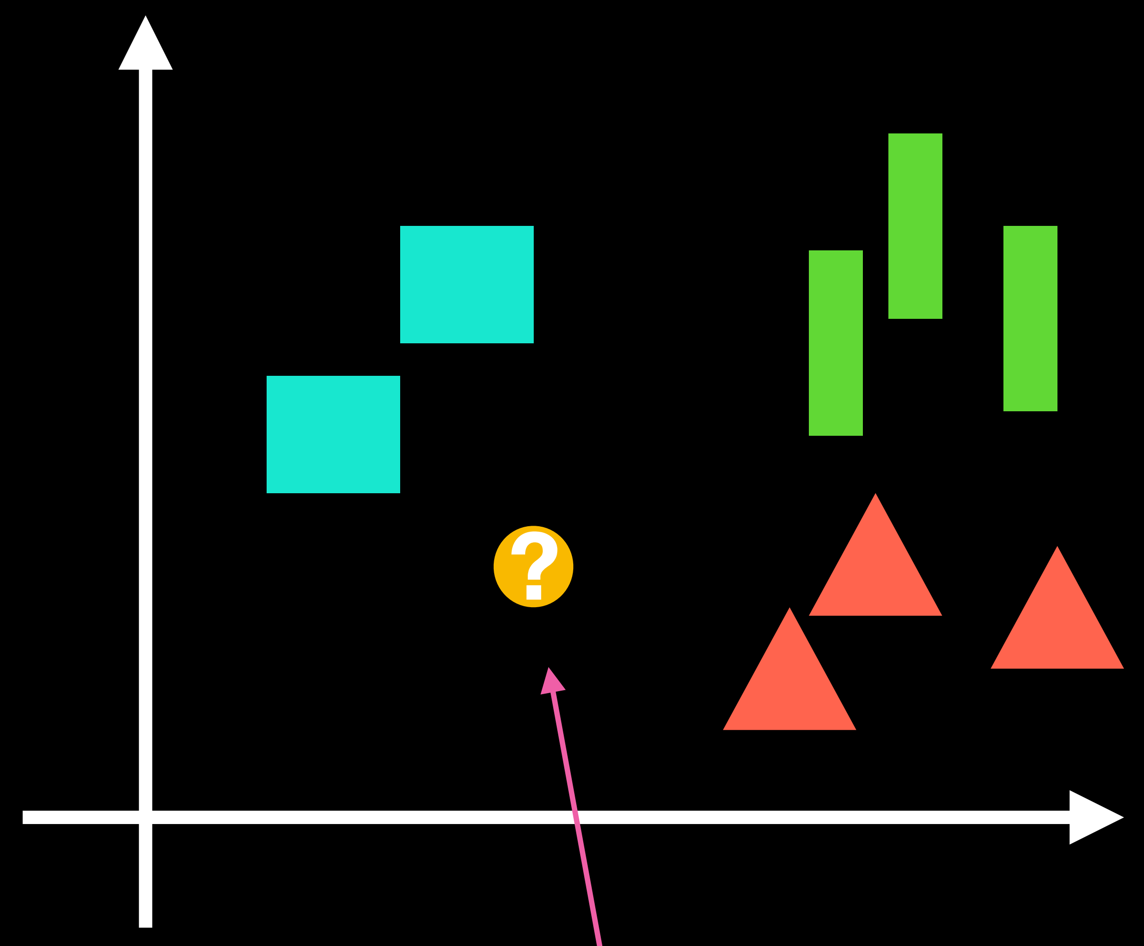
Find the neighbors



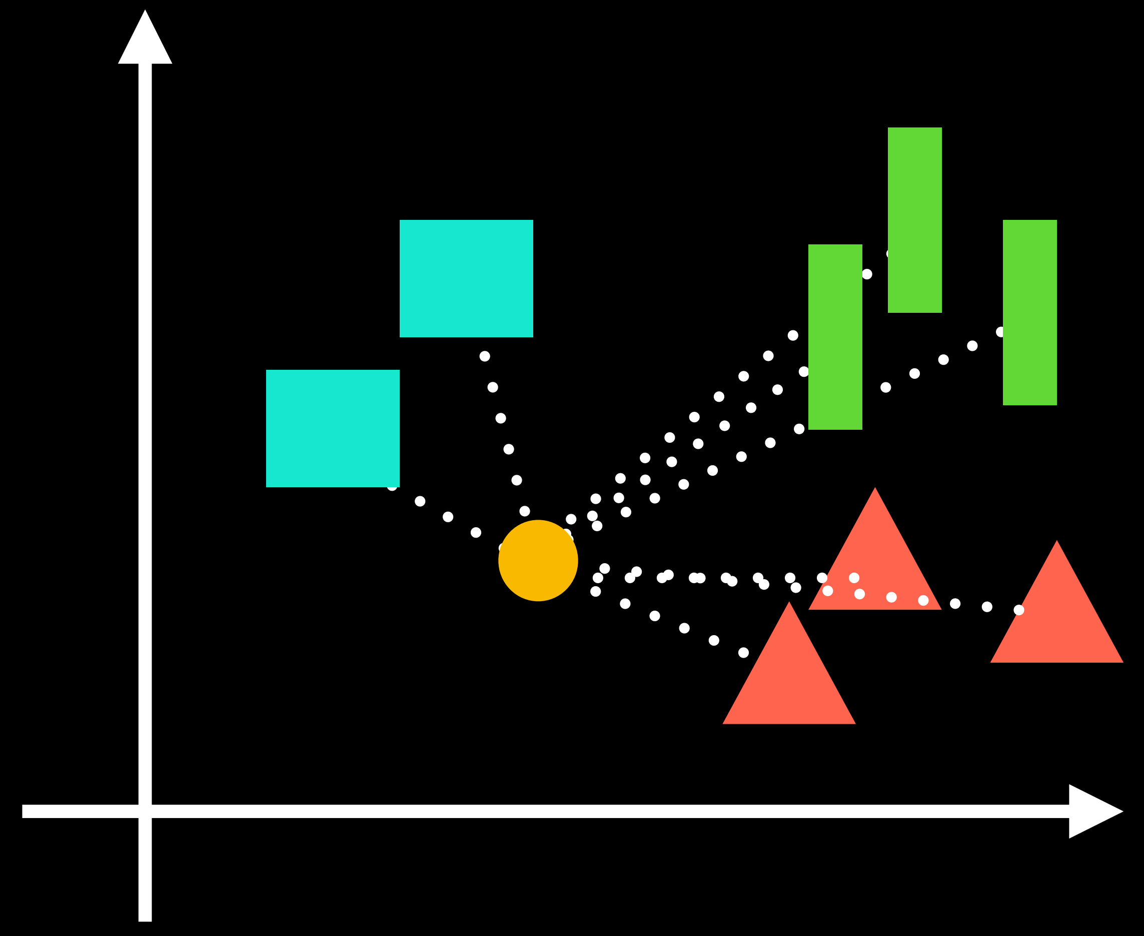
Vote on most nearest neighbor categories (based on k)

What are K-Nearest Neighbors?

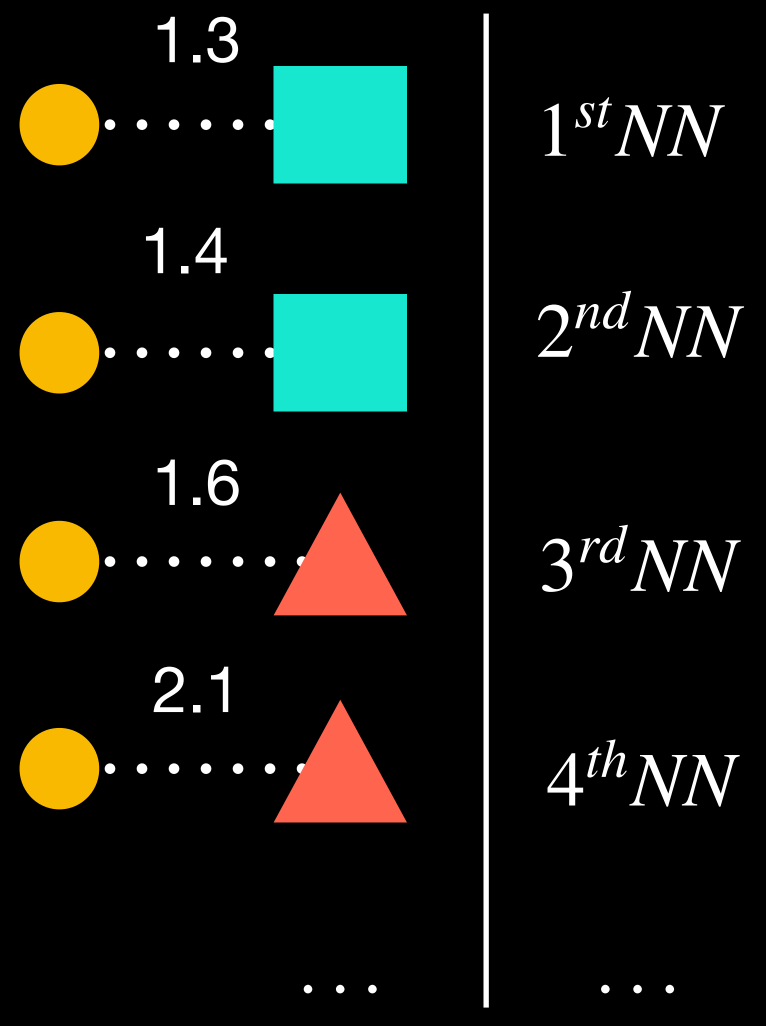
- **Algorithm to look at data surrounding specific target data point, in order to predict what category that data should be**



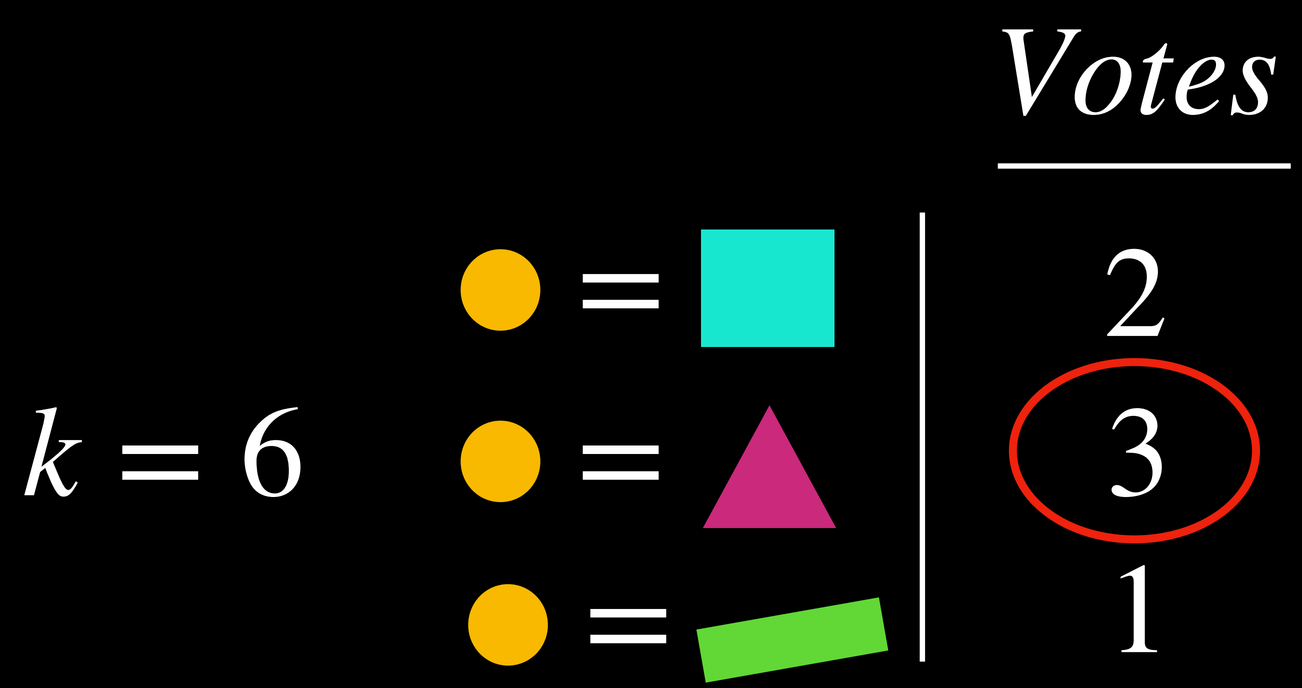
What category should this event belong to?



Measure distance to all points



Find the neighbors



Vote on most nearest neighbor categories (based on k)

Note: change k, could change the outcome

Assumptions

- The data should be in angles, masses, etc..
- Distributions of signal and background must be known in a subset of coordinates
- Signal and background do not vary rapidly in non-reference coordinates

Definitions

$\vec{\xi} \longrightarrow$ Coordinates

$\xi_{ref} \longrightarrow$ Reference coordinate

$S(\xi) \longrightarrow$ Signal function of coordinates

$B(\xi) \longrightarrow$ Background function of coordinates

No a priori information required

Assumptions

- The data should be in angles, masses, etc..
- Distributions of signal and background must be known in a subset of coordinates
- Signal and background do not vary rapidly in non-reference coordinates

Definitions

$\vec{\xi} \longrightarrow$ Coordinates

$\xi_{ref} \longrightarrow$ Reference coordinate

$S(\xi) \longrightarrow$ Signal function of coordinates

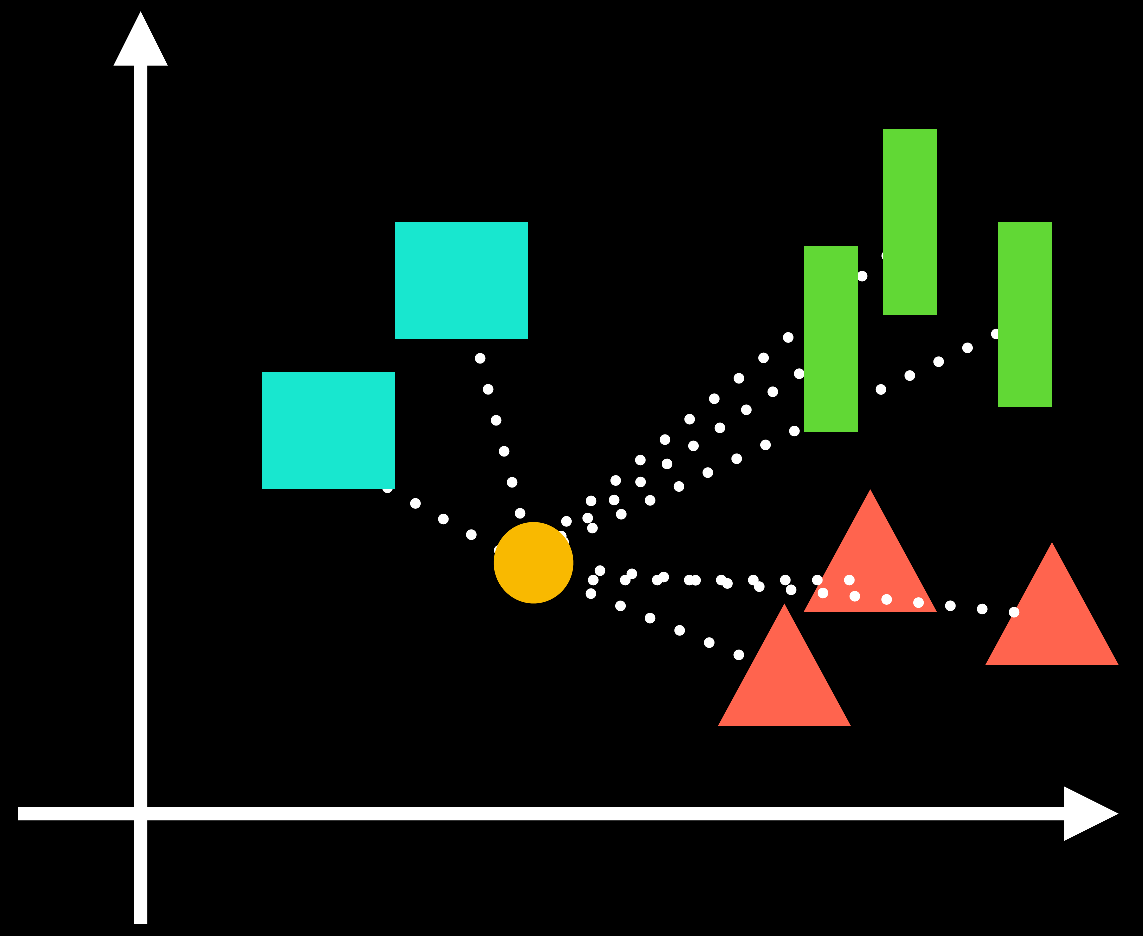
$B(\xi) \longrightarrow$ Background function of coordinates

No a priori information required

Normalized Euclidean Distance

- Need to assign a distance metric to phase space to determine how close two events are in non-reference coordinates

$$d_{ij} = \sum_{k \neq \xi_{ref}}^{m} \left[\frac{(\xi_k)_i - (\xi_k)_j}{R_k} \right]^2$$



R_k = maximal distance between any two events ξ_k

Measure distance to all points

- For each event, a computation of the distance between all other events in data is performed to obtain the nearest neighbor events
- Once these events are obtained, they are fit to gather fit parameters, $\vec{\alpha}$ to

$$F(\xi_r, \vec{\alpha}) = \frac{F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})}{\int [F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})]}$$

$F_s(\xi_r, \vec{\alpha})$ **(Signal)** $\longrightarrow$ $\int F_s(\xi_r, \vec{\alpha}) d\xi_r = n_{sig}$

$F_b(\xi_r, \vec{\alpha})$ **(Background)** $\longrightarrow$ $\int F_b(\xi_r, \vec{\alpha}) d\xi_r = n_{backgrond}$

$$\boxed{Q_i = \frac{F_s(\xi_r^i, \hat{\alpha}_i)}{F_s(\xi_r^i, \hat{\alpha}_i) + F_b(\xi_r^i, \hat{\alpha}_i)}}$$

Quality Factor Description w/ Toy Monte Carlo

- For each event, a computation of the distance between all other events in data is performed to obtain the **nearest neighbor events**
- Once these events are obtained, they are fit to gather fit parameters, $\vec{\alpha}$ to

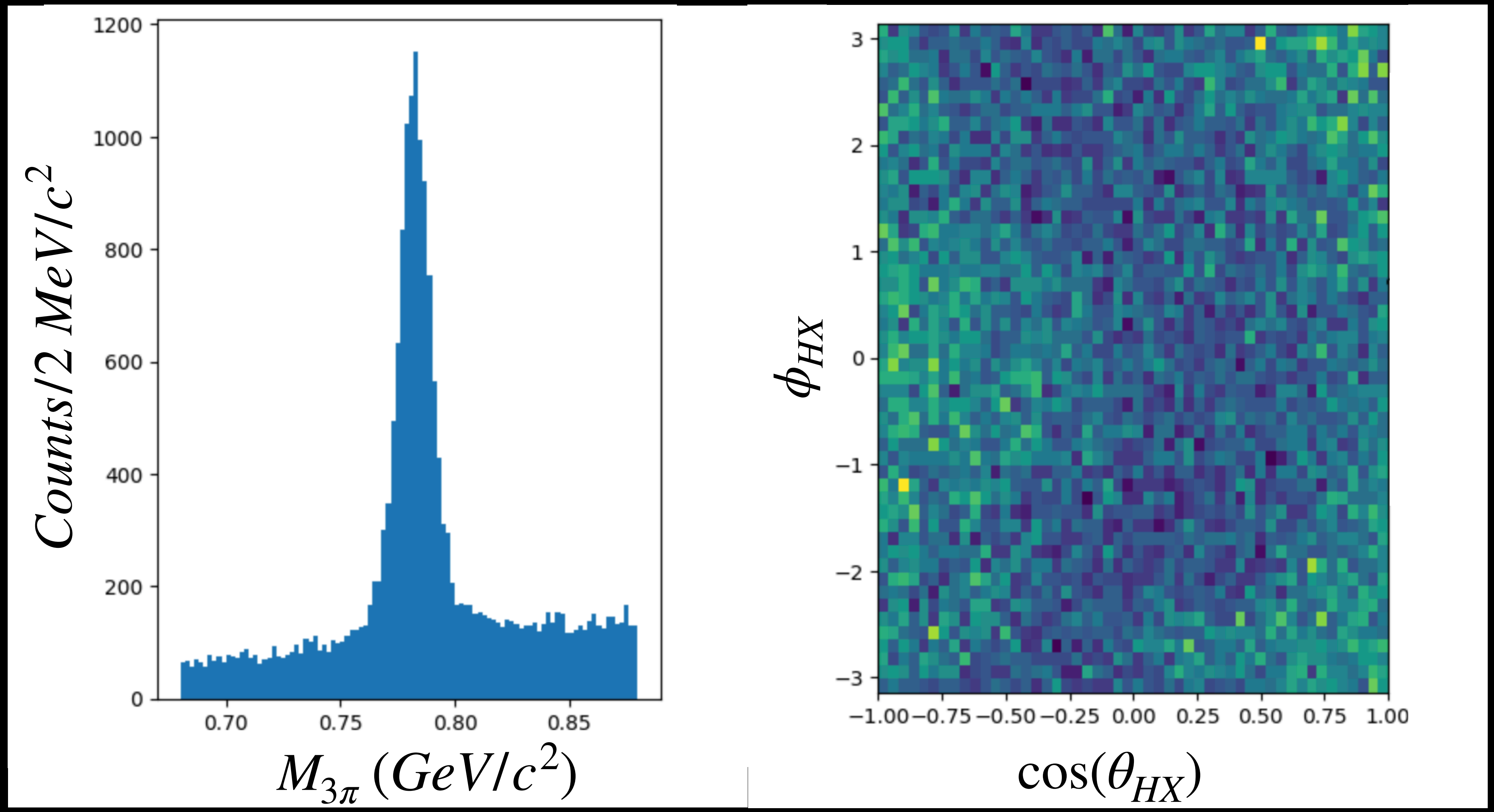
$$F(\xi_r, \vec{\alpha}) = \frac{F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})}{\int [F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})]}$$

$F_s(\xi_r, \vec{\alpha})$ **(Signal)** $\longrightarrow \int F_s(\xi_r, \vec{\alpha}) d\xi_r = n_{sig}$

$F_b(\xi_r, \vec{\alpha})$ **(Background)** $\longrightarrow \int F_b(\xi_r, \vec{\alpha}) d\xi_r = n_{backgrond}$

$$\boxed{Q_i = \frac{F_s(\xi_r^i, \hat{\alpha}_i)}{F_s(\xi_r^i, \hat{\alpha}_i) + F_b(\xi_r^i, \hat{\alpha}_i)}}$$

Signal + Background Toy Monte Carlo



$$\vec{\xi} = (m_{3\pi}, cos\theta_{HX}, \phi_{HX})$$

$$\vec{\xi}_{ref} = m_{3\pi}$$

$$F_s(m_{3\pi}, \vec{\alpha}) = s \cdot V(m_{3\pi}, m_\omega, \Gamma_\omega, \sigma)$$

$$F_b(m_{3\pi}, \vec{\alpha}) = b_1 \cdot m_{3\pi} + b_0$$

Quality Factor Description w/ Toy Monte Carlo

- For each event, a computation of the distance between all other events in data is performed to obtain the **nearest neighbor events**
- Once these events are obtained, they are fit to gather fit parameters, $\vec{\alpha}$ to

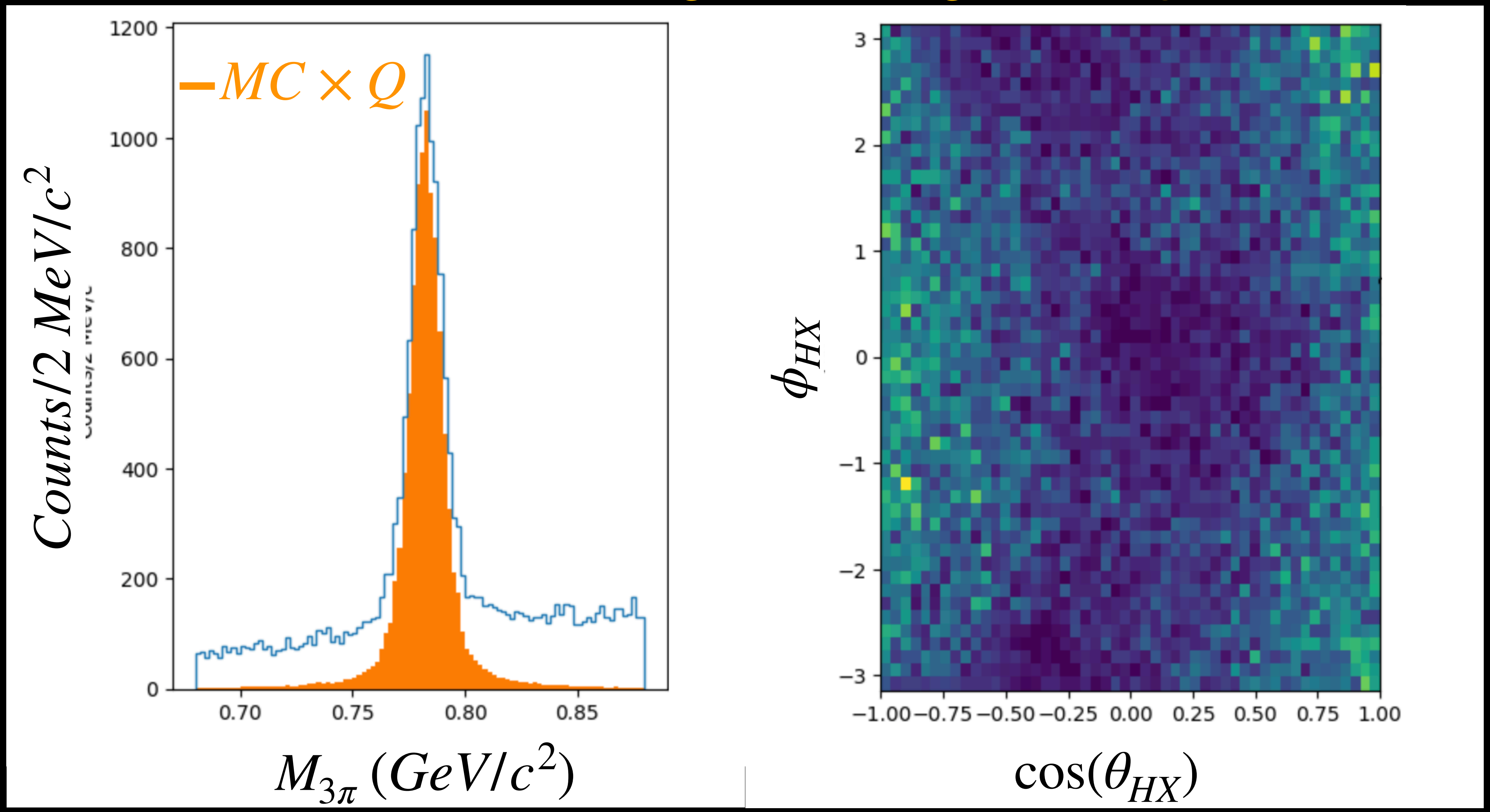
$$F(\xi_r, \vec{\alpha}) = \frac{F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})}{\int [F_s(\xi_r, \vec{\alpha}) + F_b(\xi_r, \vec{\alpha})]}$$

$F_s(\xi_r, \vec{\alpha})$ **(Signal)** $\longrightarrow$ $\int F_s(\xi_r, \vec{\alpha}) d\xi_r = n_{sig}$

$F_b(\xi_r, \vec{\alpha})$ **(Background)** $\longrightarrow$ $\int F_b(\xi_r, \vec{\alpha}) d\xi_r = n_{backgrond}$

$$\boxed{Q_i = \frac{F_s(\xi_r^i, \hat{\alpha}_i)}{F_s(\xi_r^i, \hat{\alpha}_i) + F_b(\xi_r^i, \hat{\alpha}_i)}}$$

Signal + Background Toy Monte Carlo



$$\vec{\xi} = (m_{3\pi}, cos\theta_{HX}, \phi_{HX})$$

$$\vec{\xi}_{ref} = m_{3\pi}$$

$$F_s(m_{3\pi}, \vec{\alpha}) = s \cdot V(m_{3\pi}, m_\omega, \Gamma_\omega, \sigma)$$

$$F_b(m_{3\pi}, \vec{\alpha}) = b_1 \cdot m_{3\pi} + b_0$$

Removed the generated background events!

GlueX Experiment

The main goal of the GlueX experiment is understand the underlying nature of confinement within QCD by mapping the spectrum of light quark states

With an emphasis on searching for evidence of a non-$q\bar{q}$ state (i.e. new QCD states)

$\gamma p \rightarrow \pi^0 \eta p$

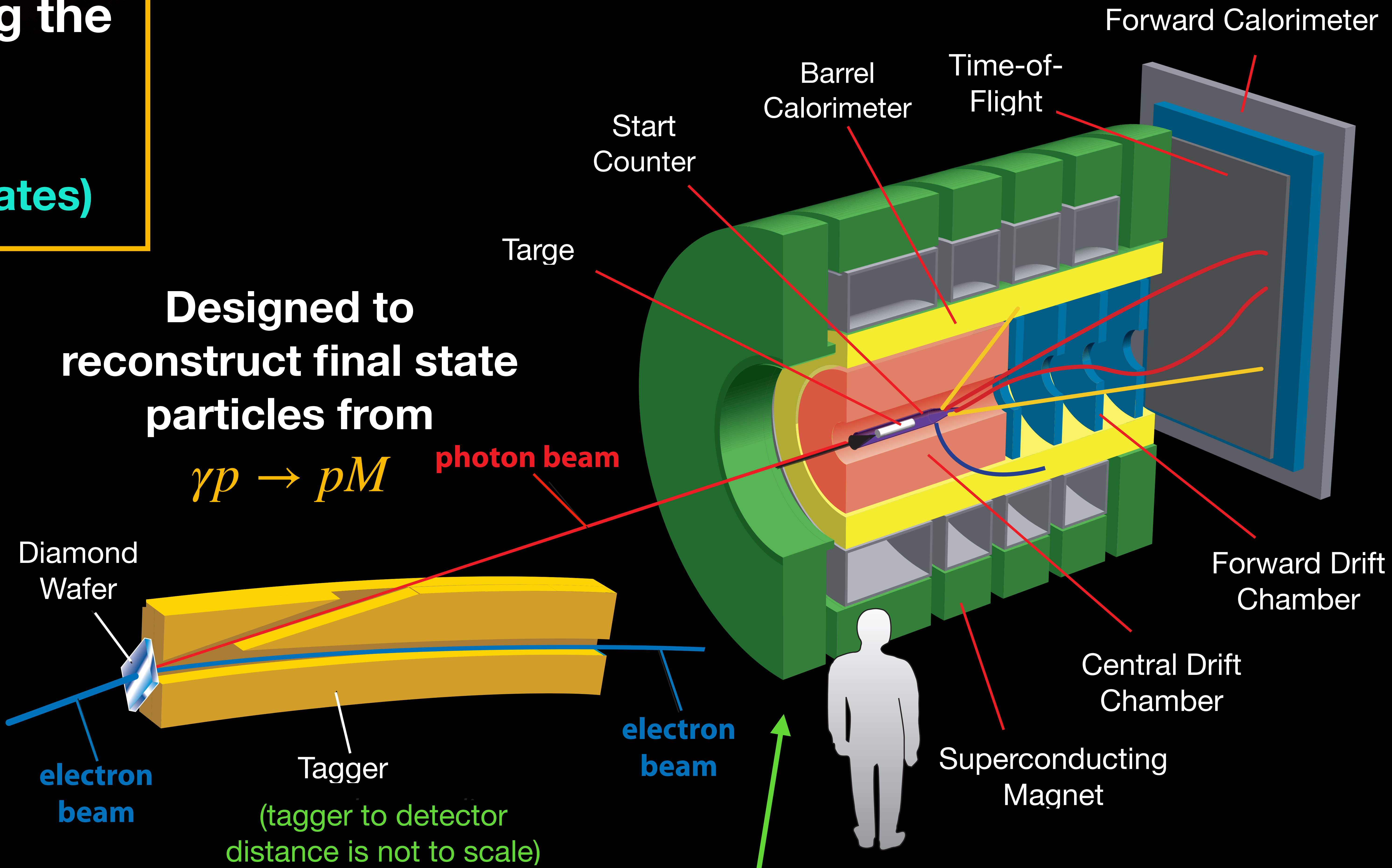
$\eta \rightarrow \pi^0 \pi^+ \pi^-$, $\pi^0 \rightarrow \gamma\gamma$

$\pi^0 \rightarrow \gamma\gamma$

$\rightarrow \boxed{4\gamma\pi^+\pi^-}$

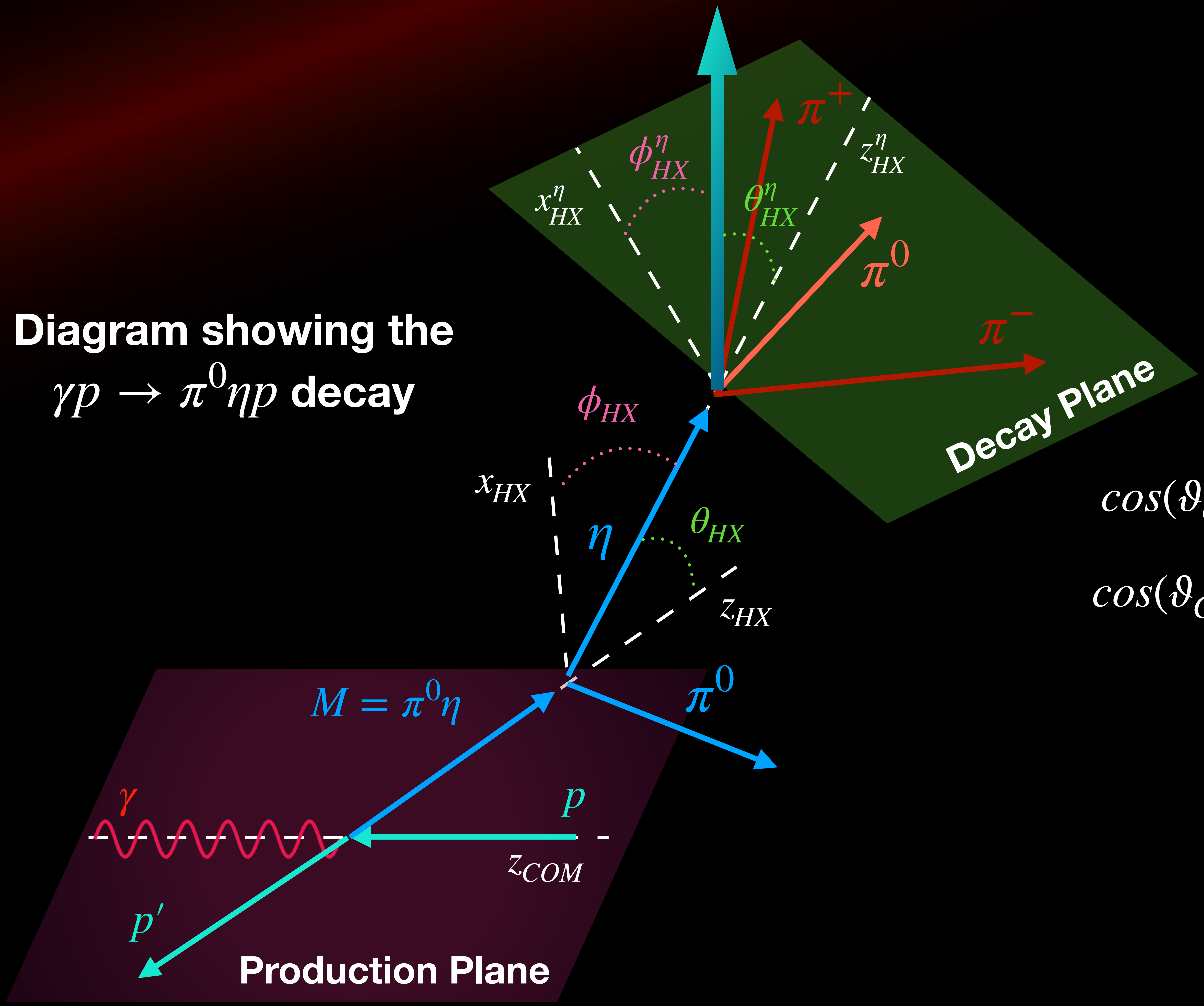
Designed to reconstruct final state particles from

$\gamma p \rightarrow pM$



- **Solenoid magnet operates at max 2 T magnetic field strength**

Diagram showing the $\gamma p \to \pi^0 \eta p$ decay



Reference Coordinate (ξ_r)

$$m(\eta)$$

Phase Space Coordinates (ξ_k)

$$\Phi_\gamma \to Polarization$$

$$\cos(\vartheta_{GJ}) \,\Big|\, \phi_{GJ} \to \eta$$

$$\cos(\vartheta_{COM}) \to \pi^0\eta$$

$$\boxed{\begin{array}{l} \cos(\vartheta^{\eta^{(\prime)}}_{HX}) \,\Big|\, \phi^{\eta^{(\prime)}}_{HX} \to \eta_{DECAY} \\ \cos(\vartheta^{\omega}_{HX}) \,\Big|\, \phi^{\omega}_{HX} \to \omega_{DECAY} \end{array}}$$

(Shown in backup slides)

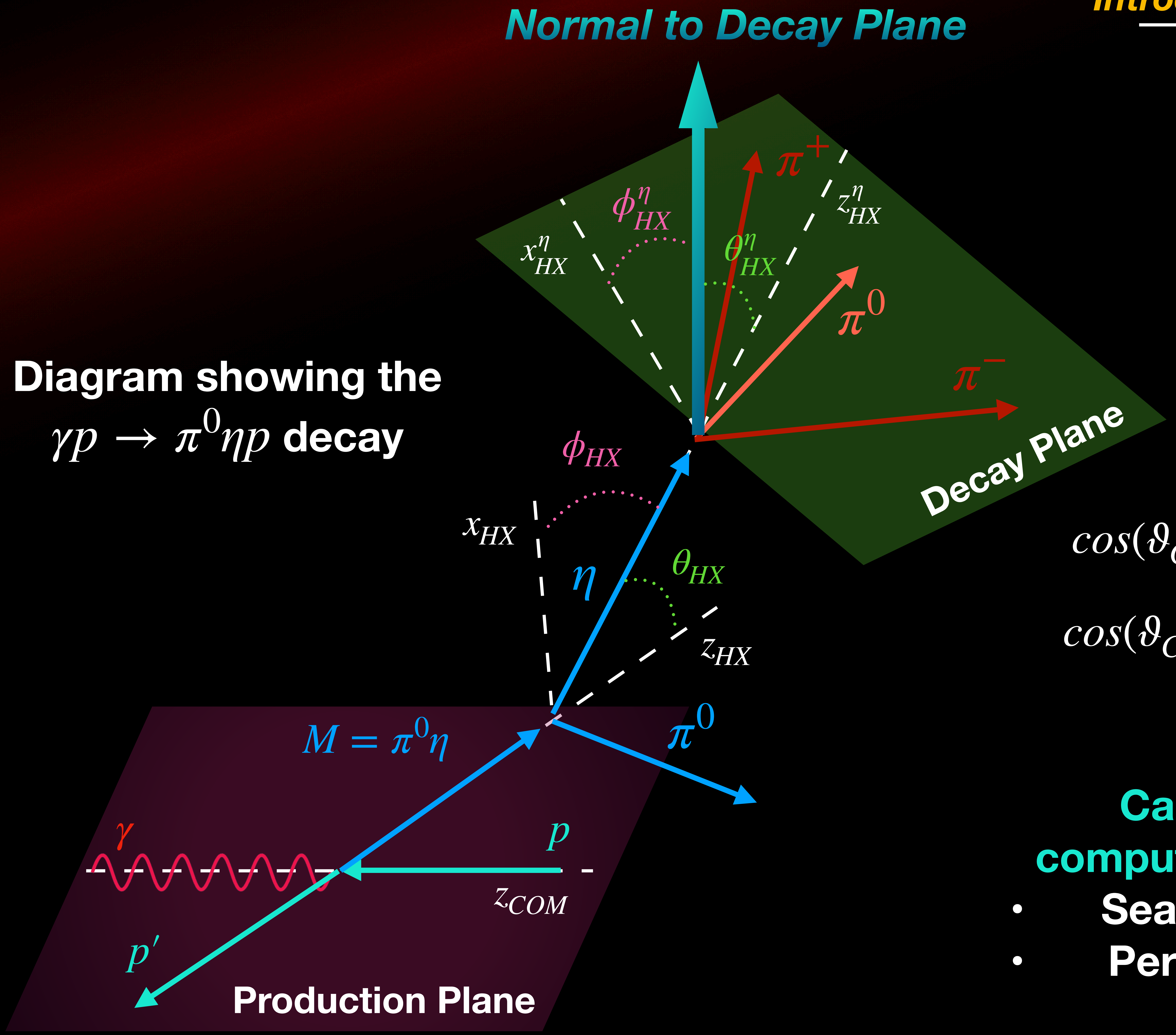


Diagram showing the $\gamma p \rightarrow \pi^0 \eta p$ decay

Reference Coordinate (ξ_r)

$$m(\eta)$$

Phase Space Coordinates (ξ_k)

$$\Phi_\gamma \rightarrow Polarization$$

$$cos(\vartheta_{GJ}) \,\Big|\, \phi_{GJ} \rightarrow \eta$$

$$cos(\vartheta_{COM}) \rightarrow \pi^0\eta$$

$$\boxed{\begin{array}{l} cos(\vartheta^{\eta^{(\prime)}}_{HX}) \,\Big|\, \phi^{\eta^{(\prime)}}_{HX} \rightarrow \eta_{DECAY} \\ cos(\vartheta^{\omega}_{HX}) \,\Big|\, \phi^{\omega}_{HX} \rightarrow \omega_{DECAY} \end{array}}$$

(Shown in backup slides)

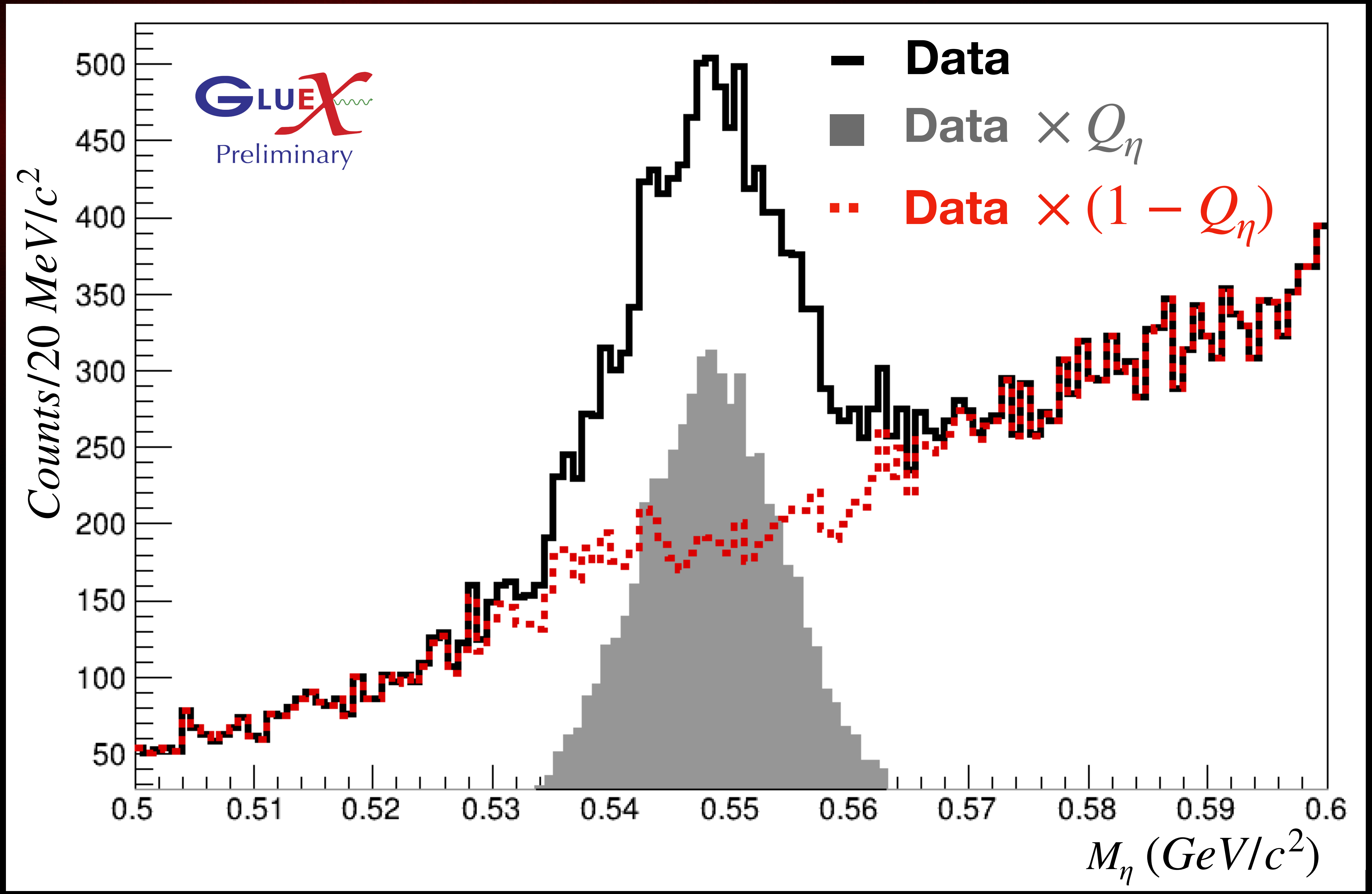
Calculations on data is a very computationally expensive technique:

- Searching for nearest neighbors
- Performing unbinned Maximum Likelihood Estimation

for each event …

Fits To Data

GlueX Data



Candidate has 50% probability it "originated" from an η

$\boxed{0.567}$

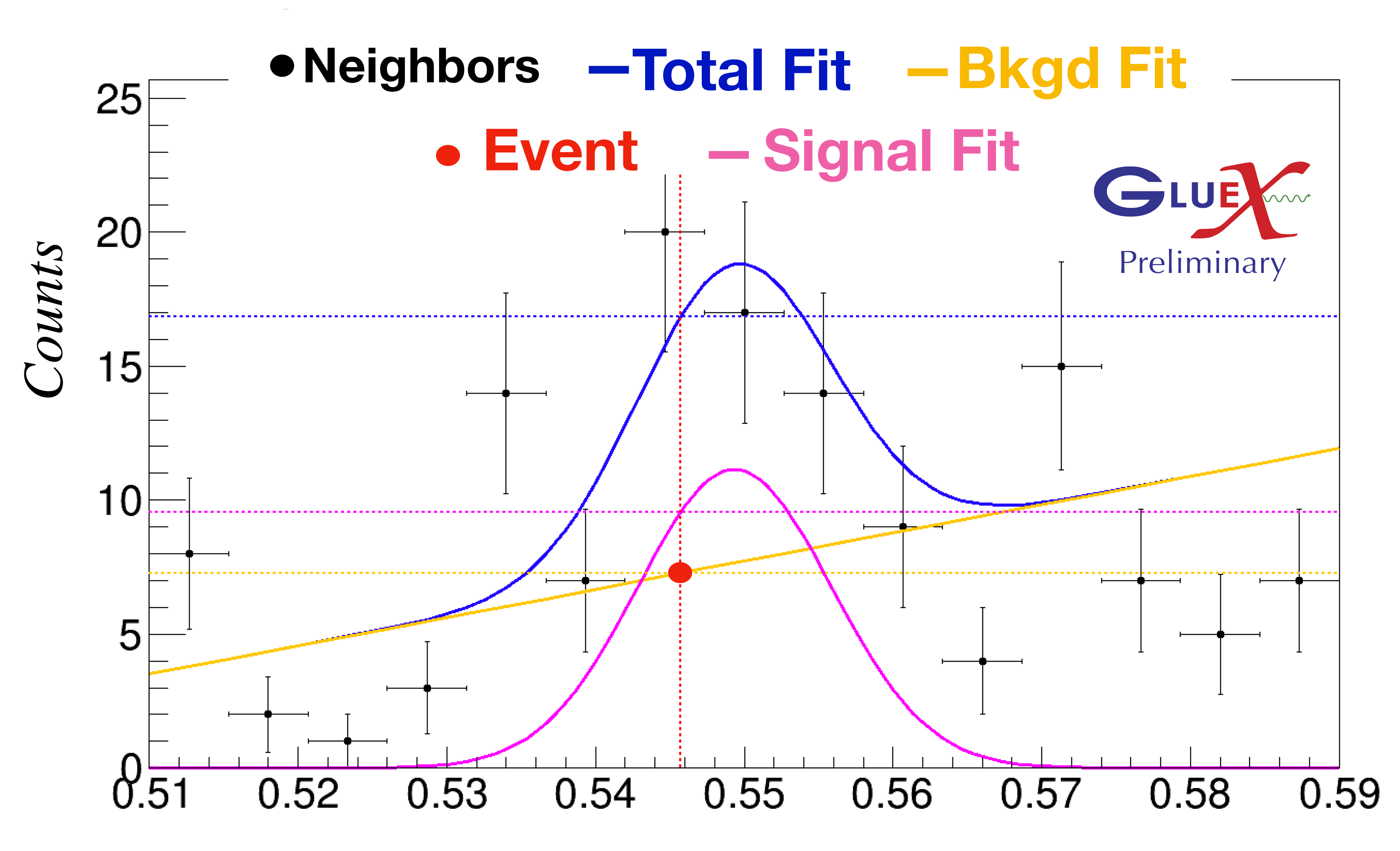
Q-value

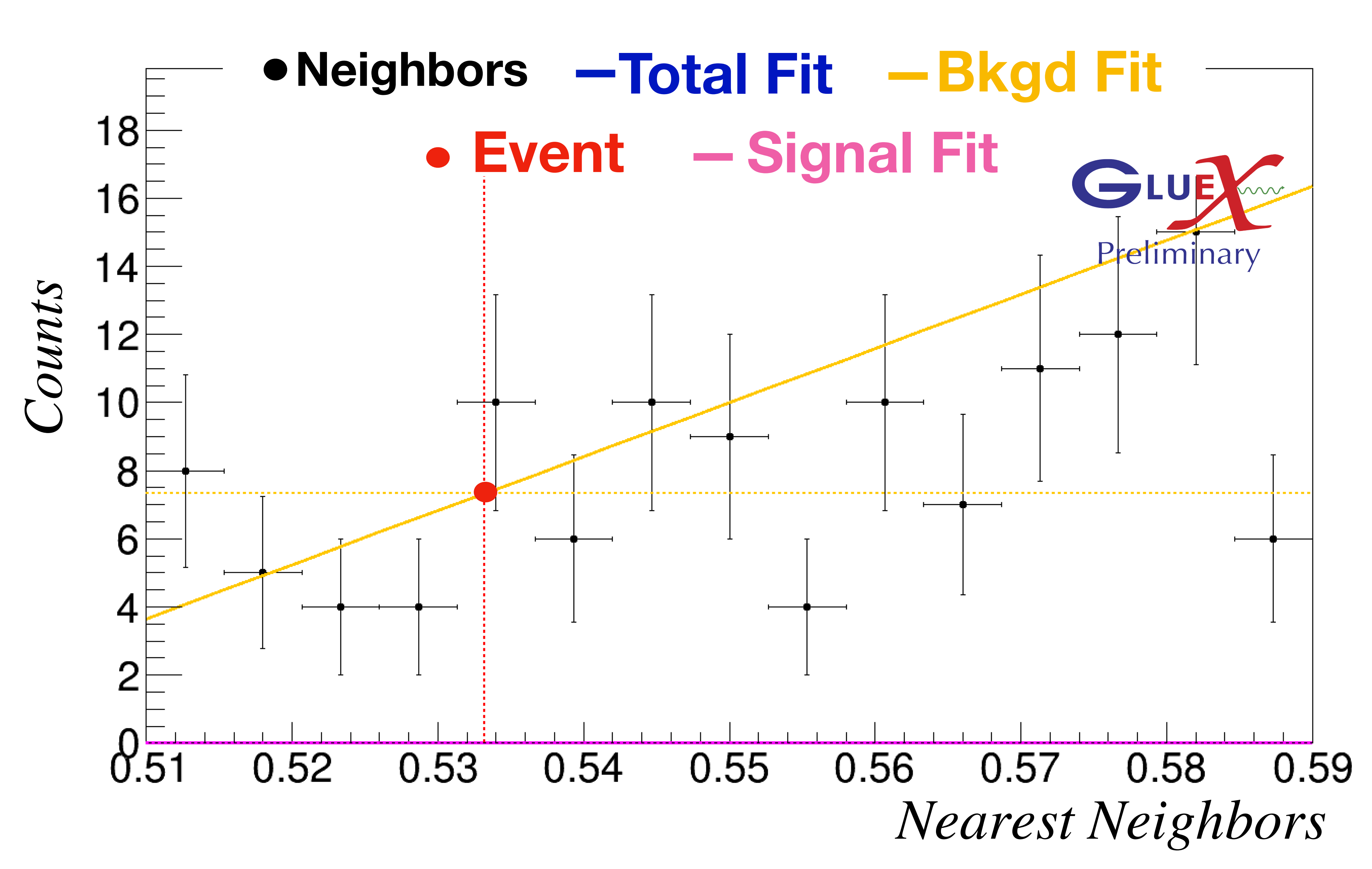
Candidate is definitely not an η event

$\boxed{0.000}$

Q-value

Individual Fits





Signal Fit ⟶ $G(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} exp\left[-\left(\frac{(m-\mu)^2}{2\sigma^2}\right)\right]$

Bkgd Fit ⟶ $b_{\nu,n}(x)\binom{\nu}{n} x^\nu (1-x)^{n-\nu}$

Pros

- Binning is not required
- Can weight the log likelihood when performing unbinned maximum likelihood fits
    - Therefore background subtraction carried out automatically
- Unlike other procedures no *a priori* knowledge of signal or background required

Cons

- Computationally expensive
- Potential inability to deal with correlated coordinates

Conclusion

- The Quality Factor procedure is proven to separate signal from non-interfering backgrounds
(on an event by event basis)

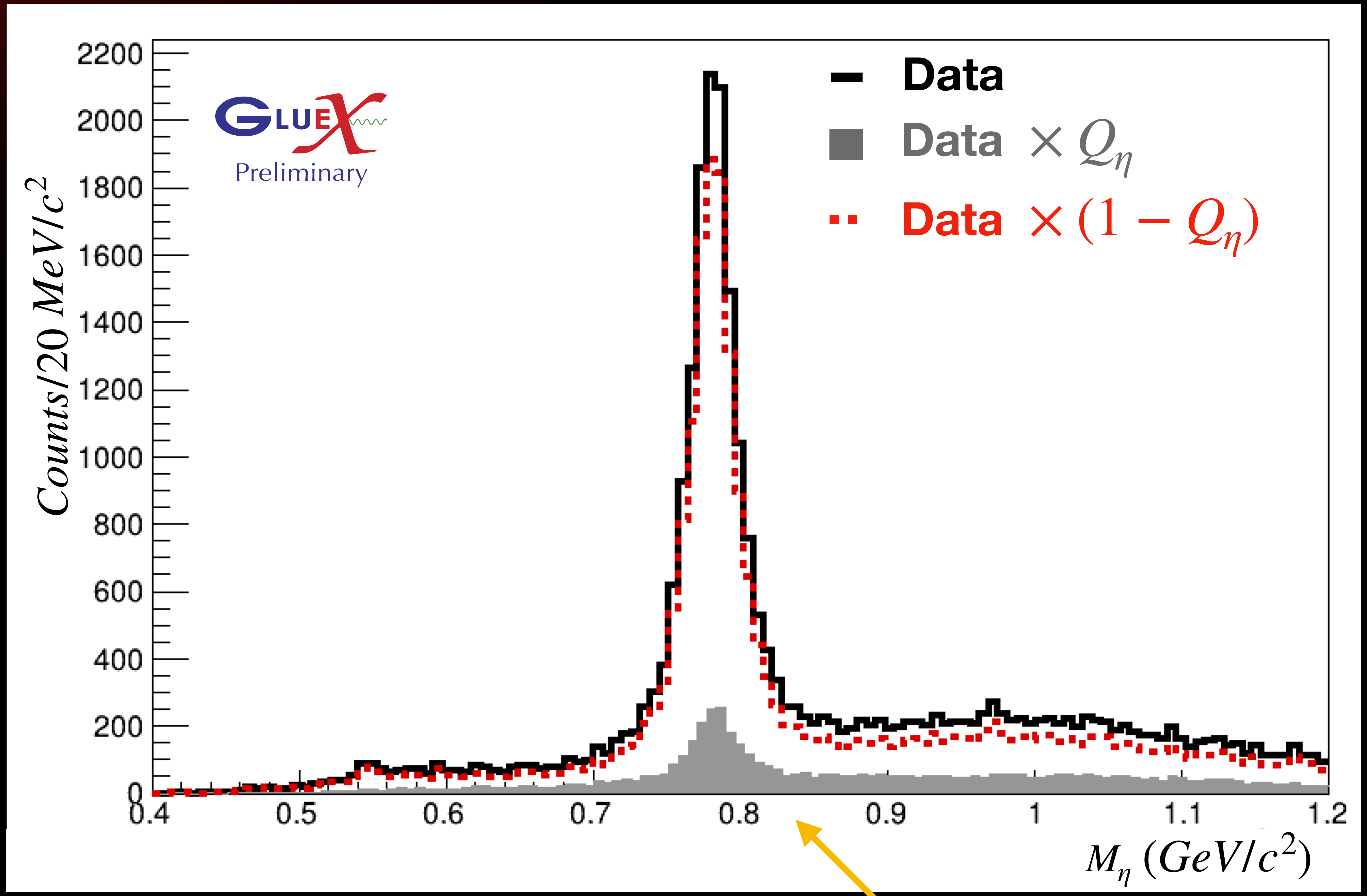
- Weights obtained from this procedure can be utilized in other analysis studies
(Cross-sections, PWA's, etc.)

C A Meyer, M Williams, M Bellis. Multivariate side-band subtraction using probabilistic event weights. Instrumentation, 2009

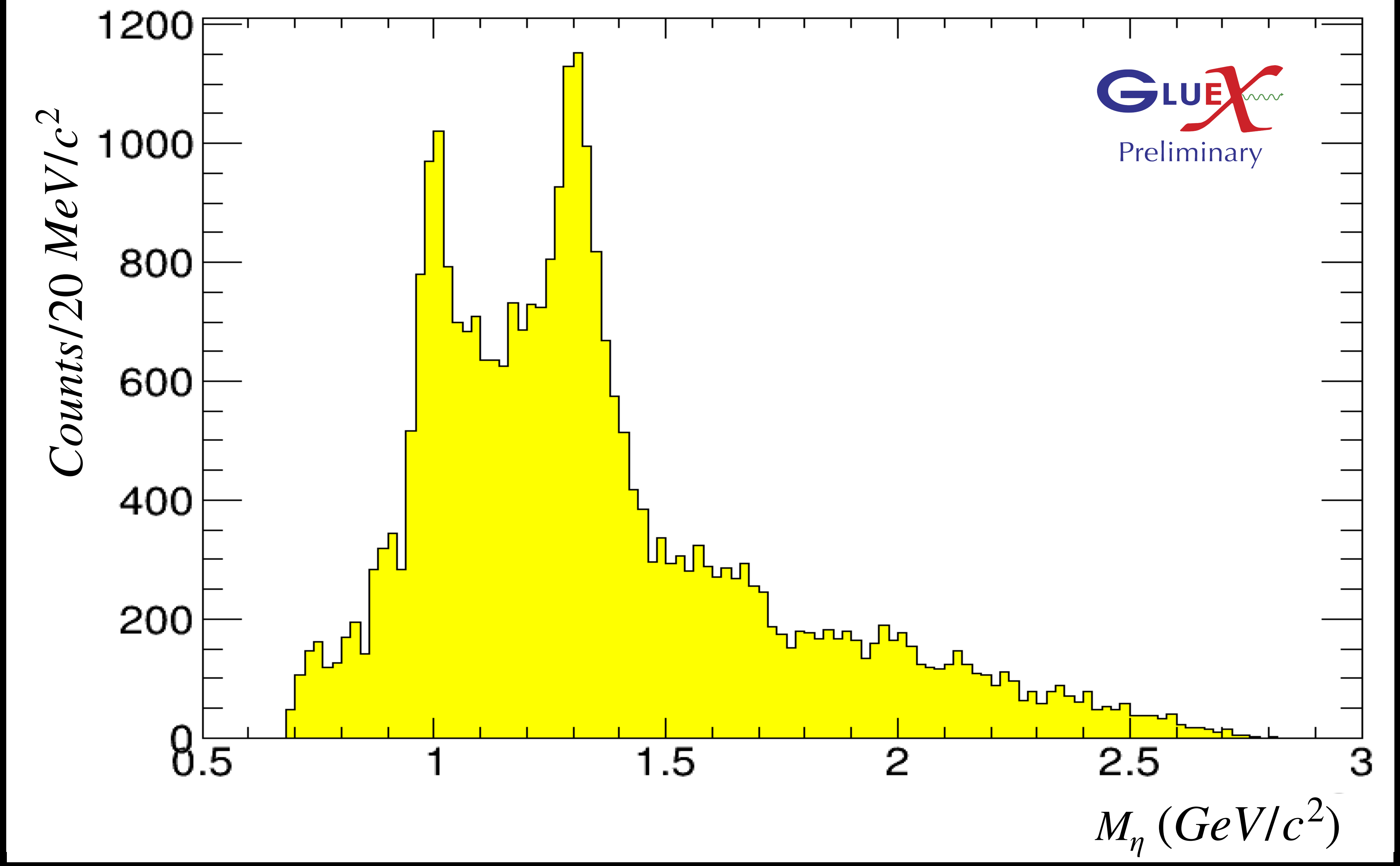
GlueX acknowledges the support of several funding agencies and computing facilities

gluex.org/thanks

BACKUP SLIDES



Q **factor eliminates most $\omega \to \pi^0\pi^+\pi^-$ background but not all**



Resonance M frame

$$\vec{y} = \frac{\vec{k} \times \vec{z}_{HX}}{|\vec{k} \times \vec{z}_{HX}|}$$

$$\vec{x} = \vec{y} \times \vec{z}$$

$\vec{k}$ vector
in beam direction

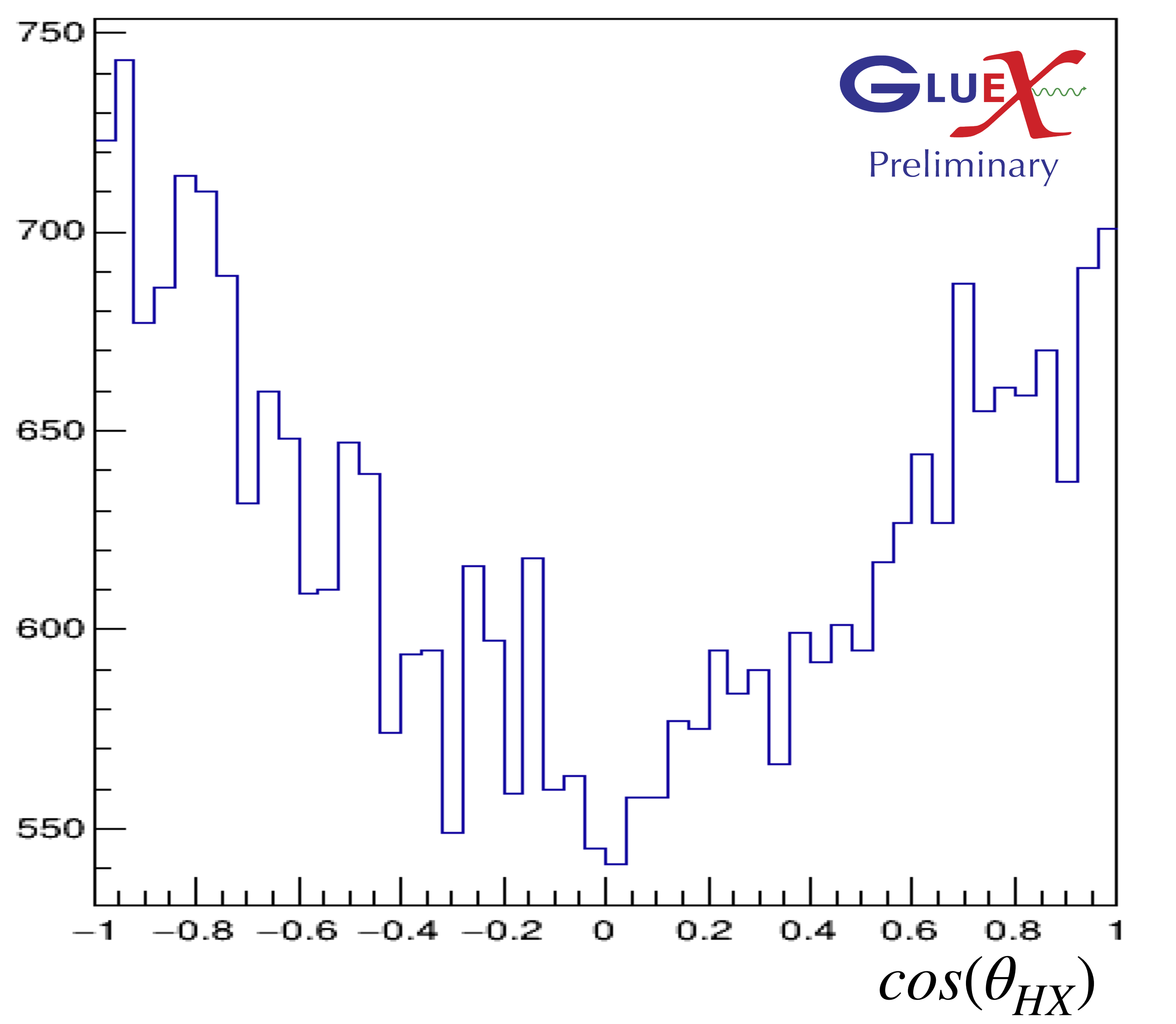
We see both $cos(\vartheta)_{HX}$ are not flat
as expected and have "wings" at edges

η Decay Frame

$$\vec{y}^{\eta}_{HX} = \frac{\vec{z}_{HX} \times \vec{z}^{\eta}_{HX}}{|\vec{z}_{HX} \times \vec{z}^{\eta}_{HX}|}$$

$$\vec{x}^{\eta}_{HX} = \vec{y}^{\eta}_{HX} \times \vec{z}^{\eta}_{HX}$$

$$\vec{n} = \frac{\vec{\pi}_{+} \times \vec{\pi}_{-}}{|\vec{\pi}_{+} \times \vec{\pi}_{-}|}$$



ω Decay Frame

$$\vec{y}^{\omega}_{HX} = \frac{\vec{z}_{HX} \times \vec{z}^{\omega}_{HX}}{|\vec{z}_{HX} \times \vec{z}^{\omega}_{HX}|}$$

$$\vec{x}^{\omega}_{HX} = \vec{y}^{\omega}_{HX} \times \vec{z}^{\omega}_{HX}$$

$$\vec{n} = \frac{\vec{\pi}_{+} \times \vec{\pi}_{-}}{|\vec{\pi}_{+} \times \vec{\pi}_{-}|}$$