Treasure Hunt: Search For Stop 4-body Decays

Using Methods Of Multivariate Analysis And Multiclass Neural Network Event Classification

Theoretical Framework

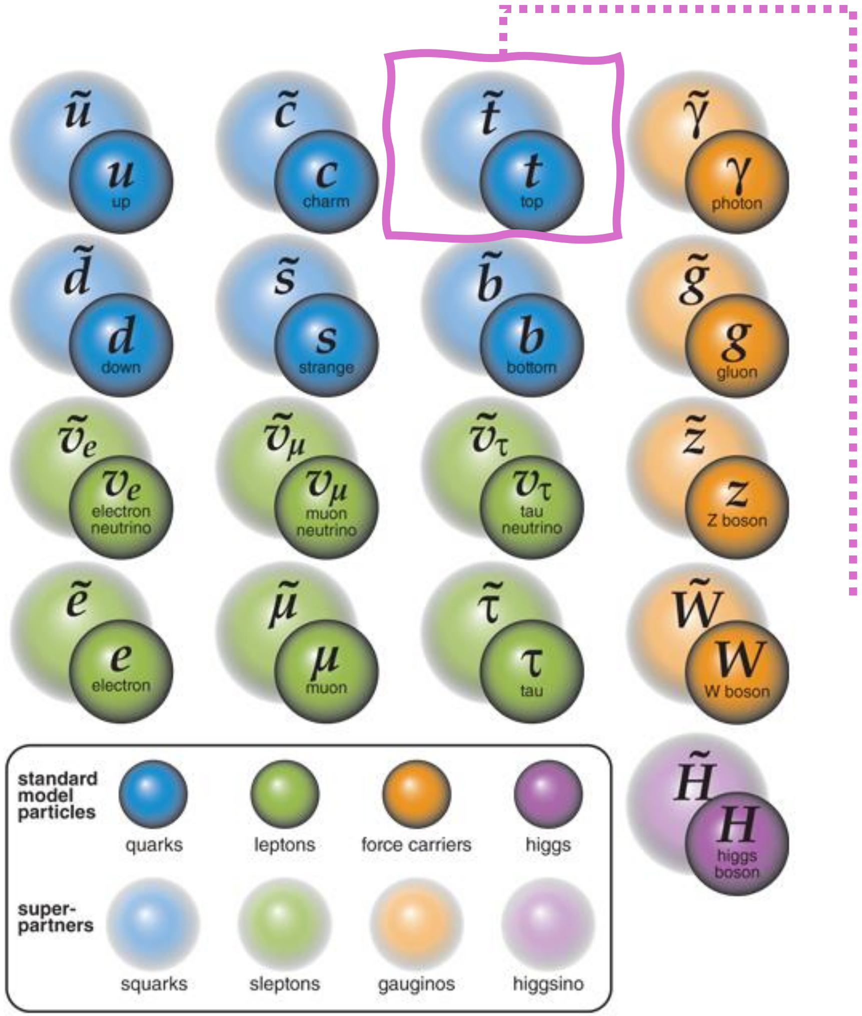
Legend says

…

There exists a symmetry between bosons and fermions not yet observed since it is of the broken sort.

…

Supersymmetry



- Every fermion has a corresponding bosonic supersymmetric partner
  - Spin ½ particle -> Spin 0 sparticle
  - Spin 1 force carrier -> Spin ½ gaugino
  - Higgs (spin 0) ->Higgsino (spin ½)

TOP SQUARK (STOP) $\tilde{t}$ -
supersymmetric partner of top quark

NEUTRALINO $\widetilde{\chi_0}$ - <u>lightest supersymmetric particle</u> (mix of neutral electroweak gauginos and higgsino)

Particles Relevant In This Search

Why 4-body decay?

$$\tilde{t} \longrightarrow \widetilde{\chi_0} + other\ decay\ products$$

$\Delta m = m(\tilde{t}) - m(\widetilde{\chi_0})$ = energy available to other decay products

$$\boxed{\text{ASSUMPTION: } \Delta m < 80\text{GeV}}$$

Consequences:

~~Two-body decay~~ ✗

Kinematically forbidden

- $\tilde{t} \rightarrow t + \widetilde{\chi_0} \quad (\,\Delta m < m(t)\,)$

Suppressed

- $\tilde{t} \rightarrow c + \widetilde{\chi_0}$

Kinematically forbidden

~~Three-body decay~~ ✗

- $\tilde{t} \rightarrow \widetilde{\chi_0} + W^+ + b$

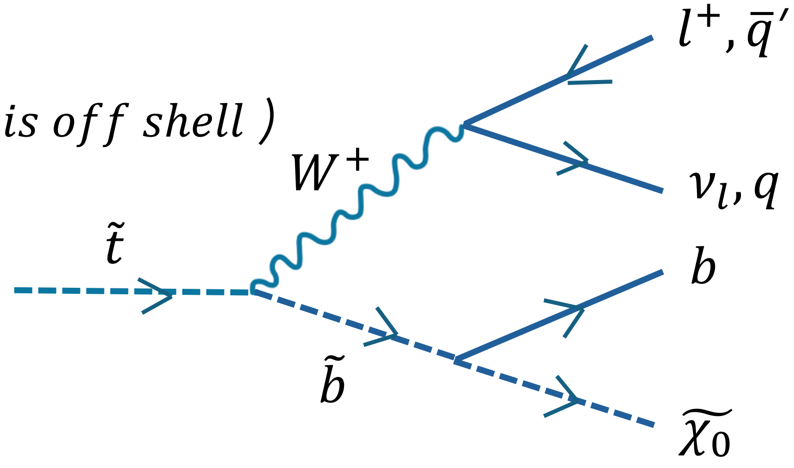
$(\,\Delta m < m(W)\,)$

$W\ is\ on\ shell$

The next best option: **4-body decay** ✓

$(\,W\ is\ off\ shell\,)$

$\Rightarrow$

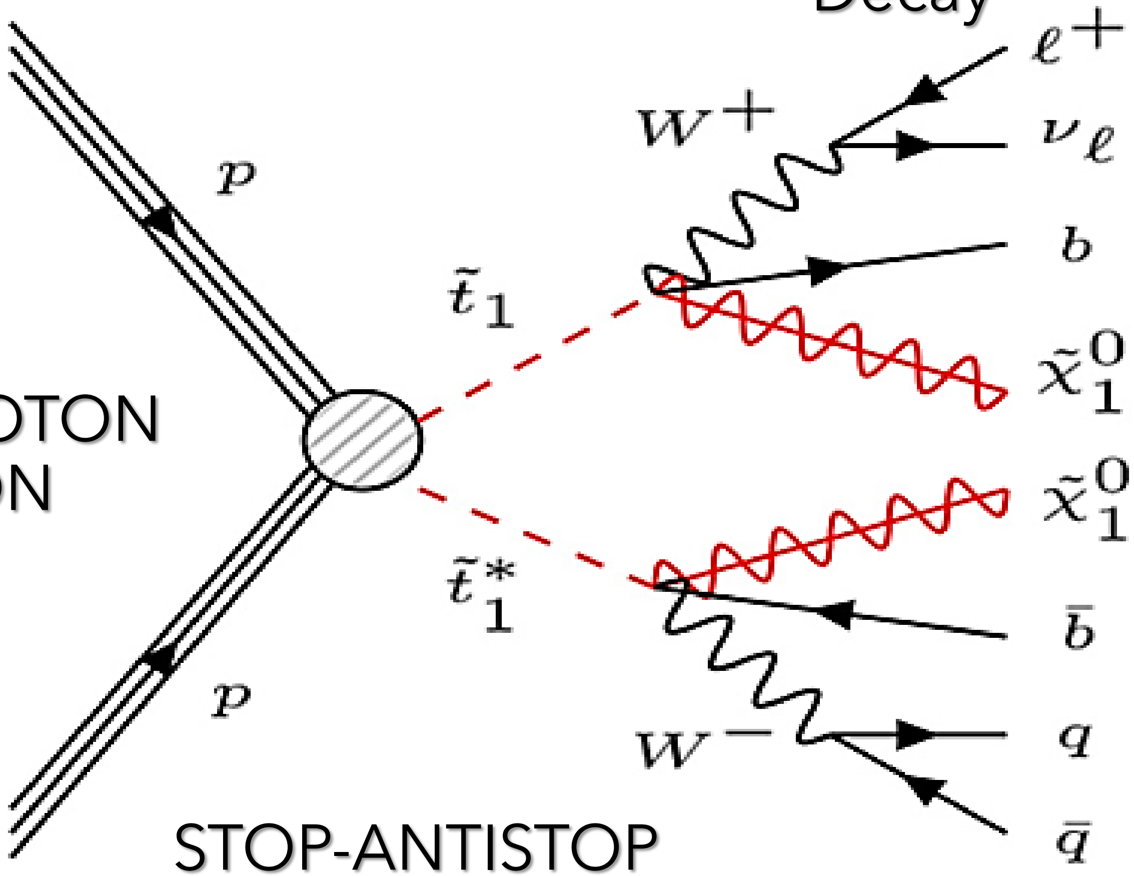


It can decay ***leptonically*** *or* ***hadronically****.*

Collider Picture

Stop Decay

PROTON-PROTON COLLISION



Leptonic Channel

Hadronic Channel

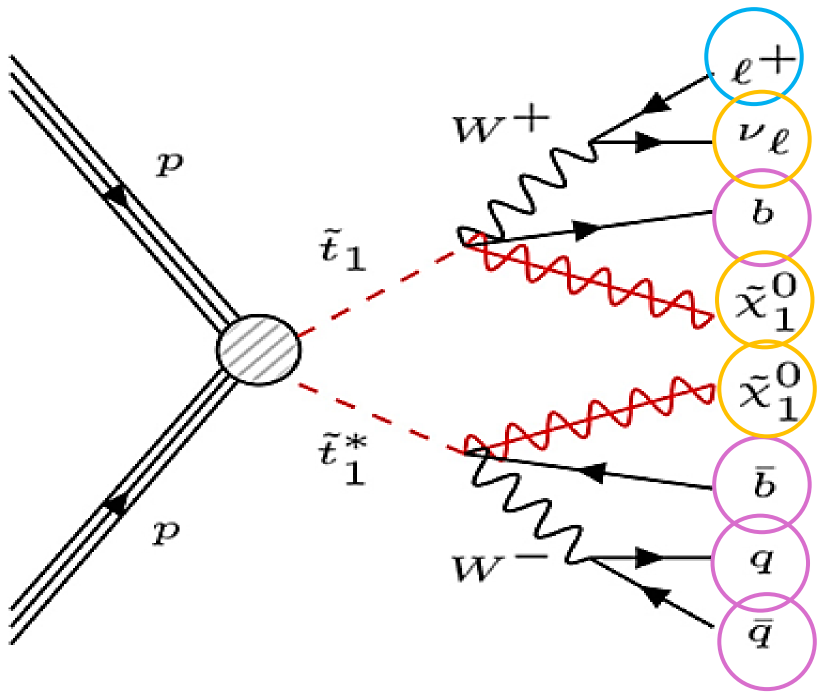
STOP-ANTISTOP PAIR PRODUCTION

Antistop Decay

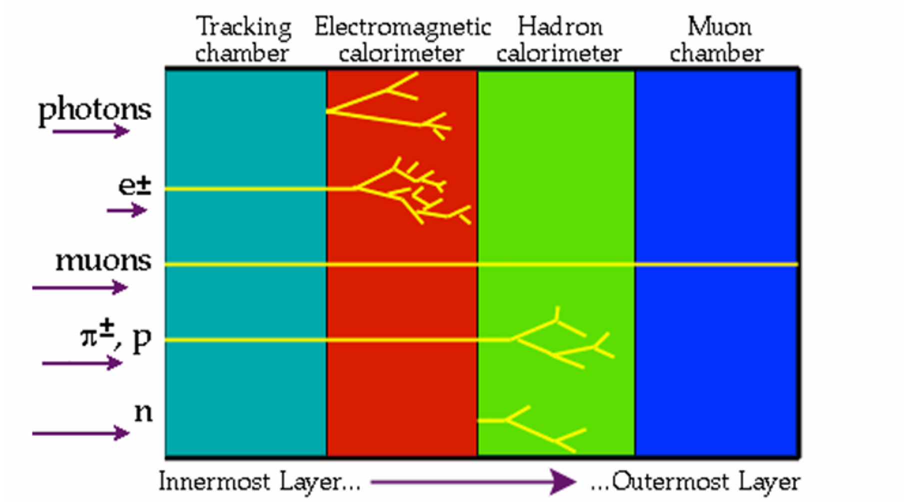
*Both stop and antistop can go through either **leptonic** or **hadronic** channel.*

Stop Decay

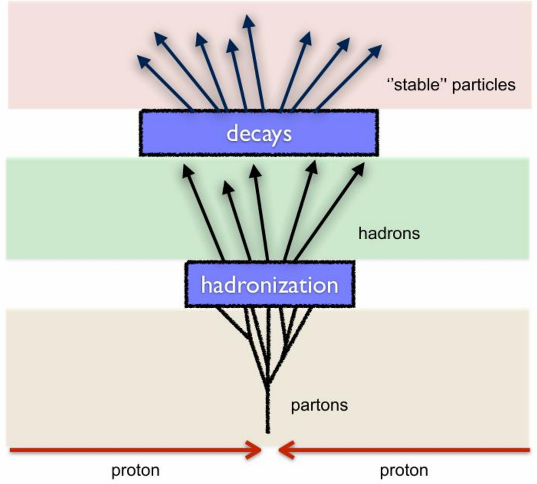
Detector Trace



"Stable" Particles:



Jets:



Missing Energy

Phase Space

- Construct the phase space that is optimal for the search of stop

1. step: Event Selection

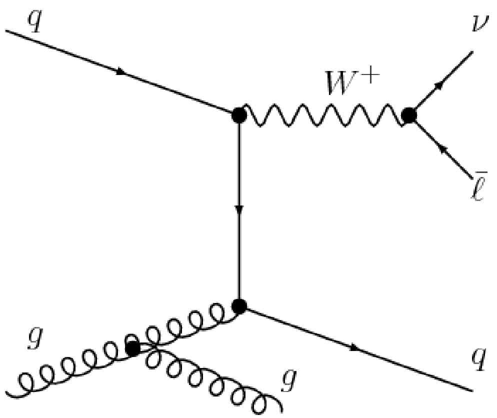
- Includes one lepton
  (one stop decays leptonically)
- Includes at least one jet
- High missing transverse energy
  (from two neutralinos and one neutrino)

2. step: Preselection

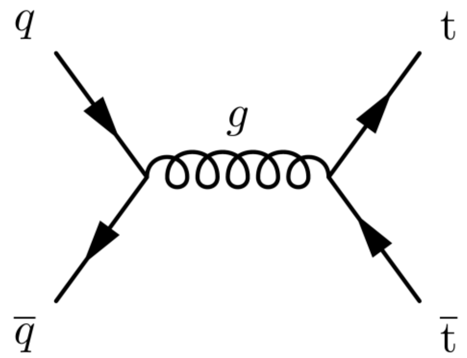
Cut on kinematic variables to further reduce the number of background events

Processes In The Phase Space

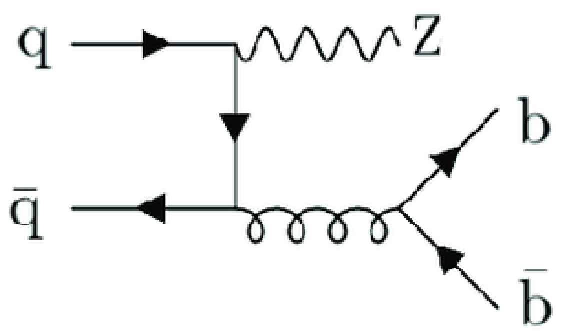
W Jets



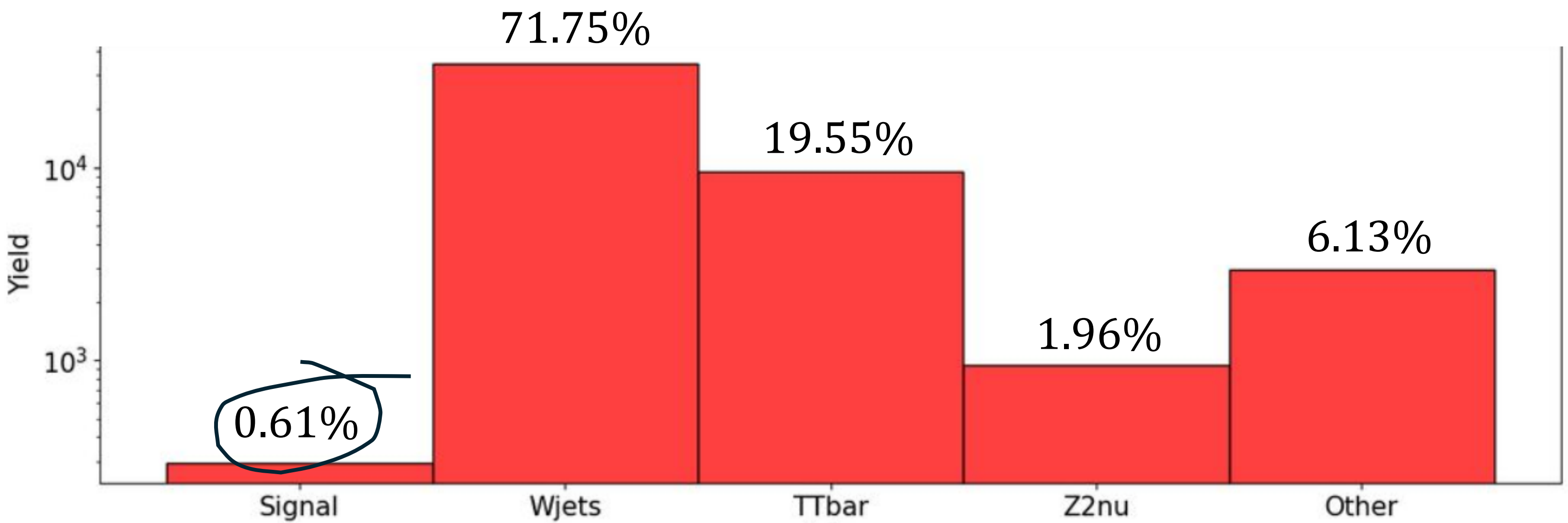
$t\bar{t}$ Pair Production

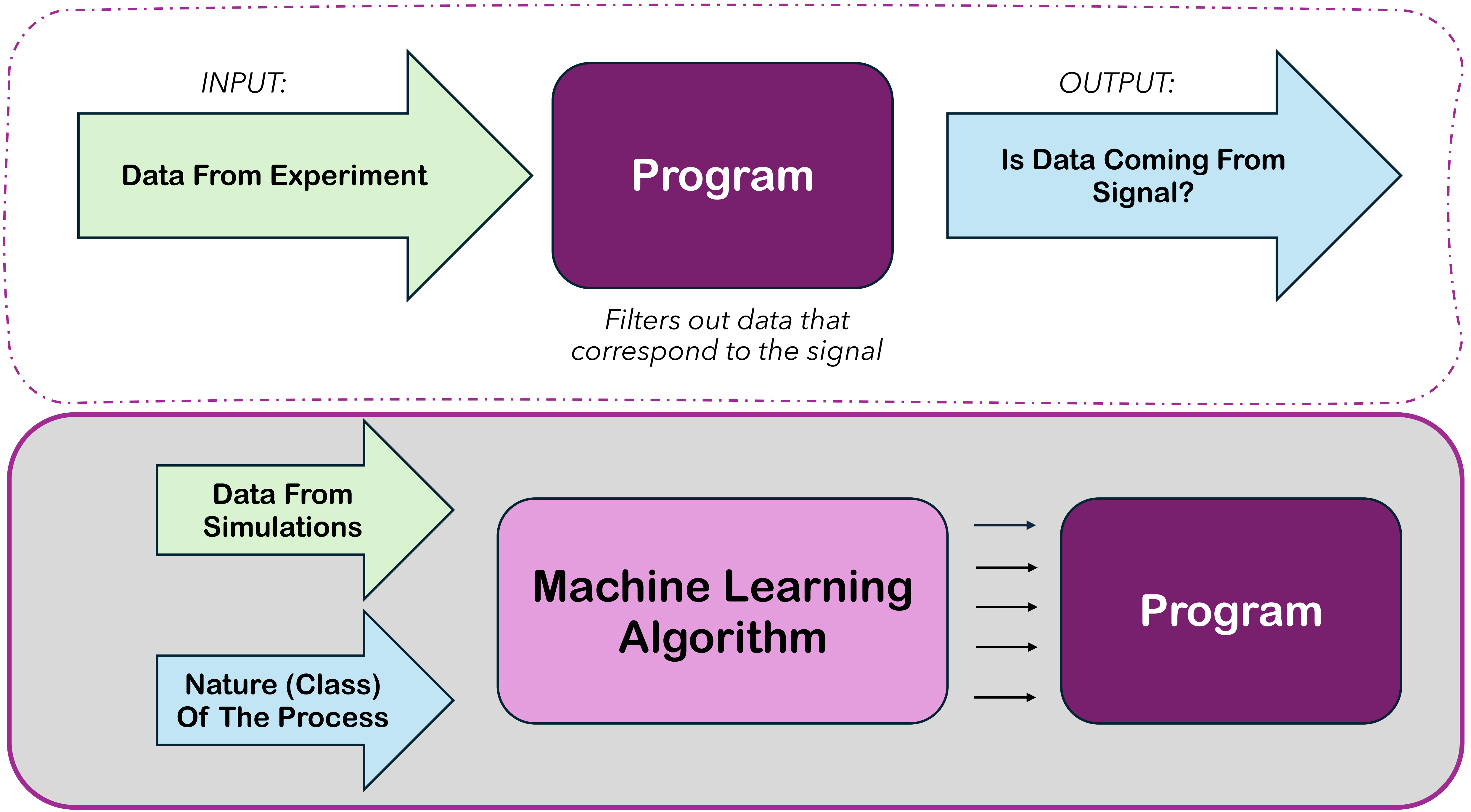


Z Jets



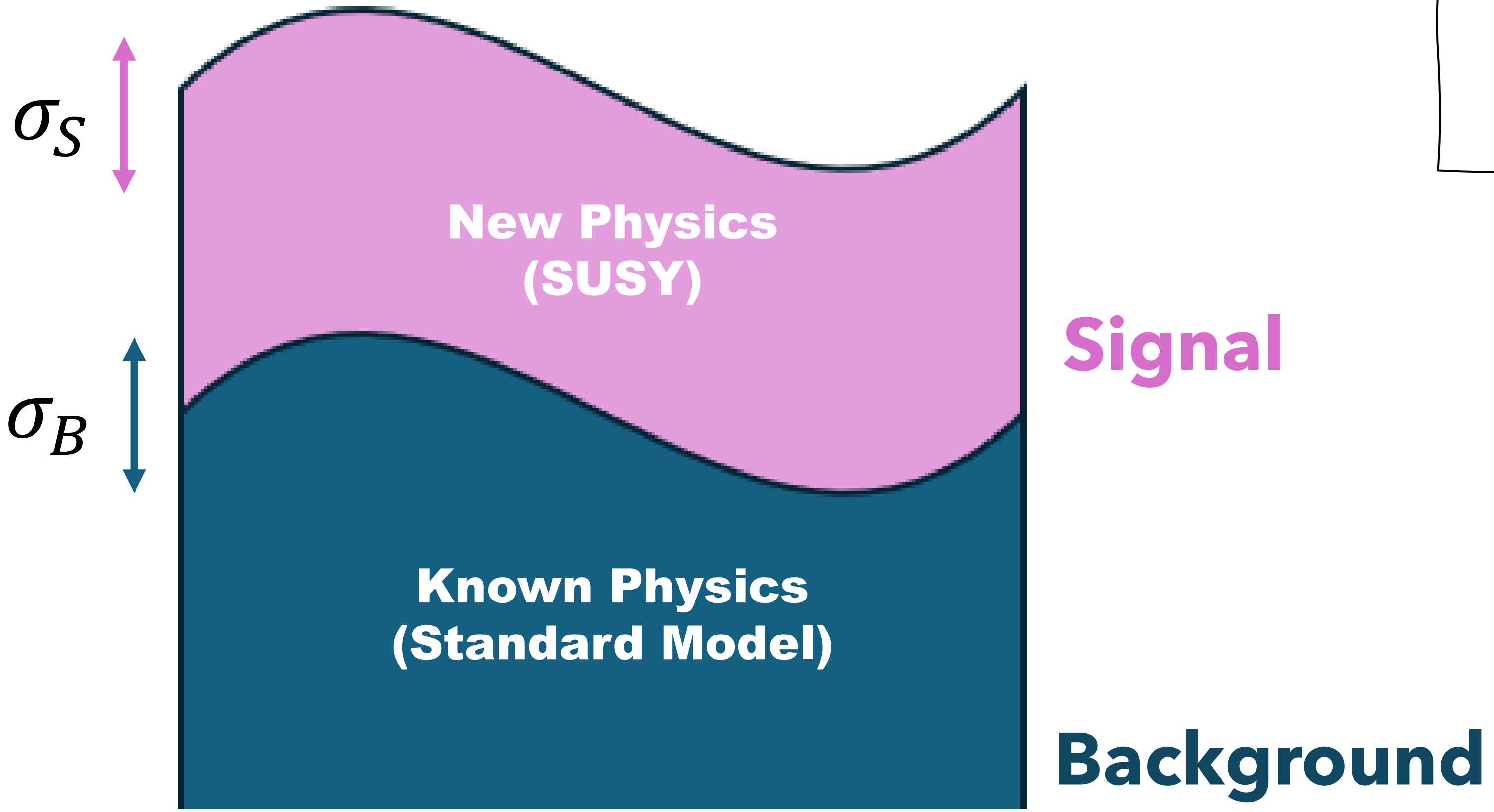
Other Smaller Backgrounds




INPUT:
Data From Experiment
Program
OUTPUT:
Is Data Coming From Signal?
Filters out data that correspond to the signal
Data From Simulations
Nature (Class) Of The Process
Machine Learning Algorithm
Program

One, two, three, four…

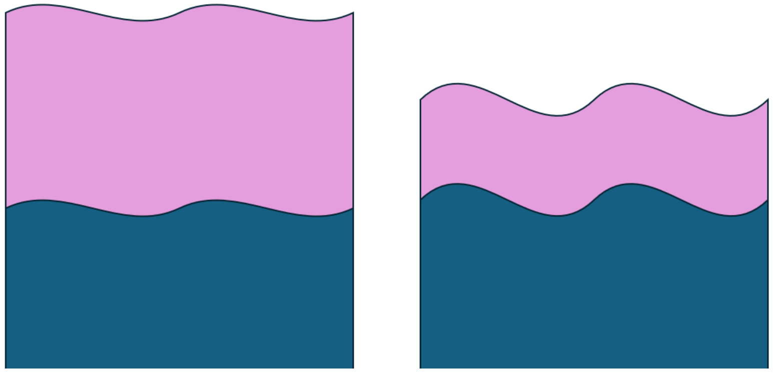
Counting the events:



How significant is the total number of signal events?

$$\boxed{\frac{S}{\sigma_{tot}} = \frac{S}{\sqrt{\sigma_{Sig}^2 + \sigma_{Bkg}^2 + \sigma_{sys}^2}}}$$

Figure Of Merit



...five sigma

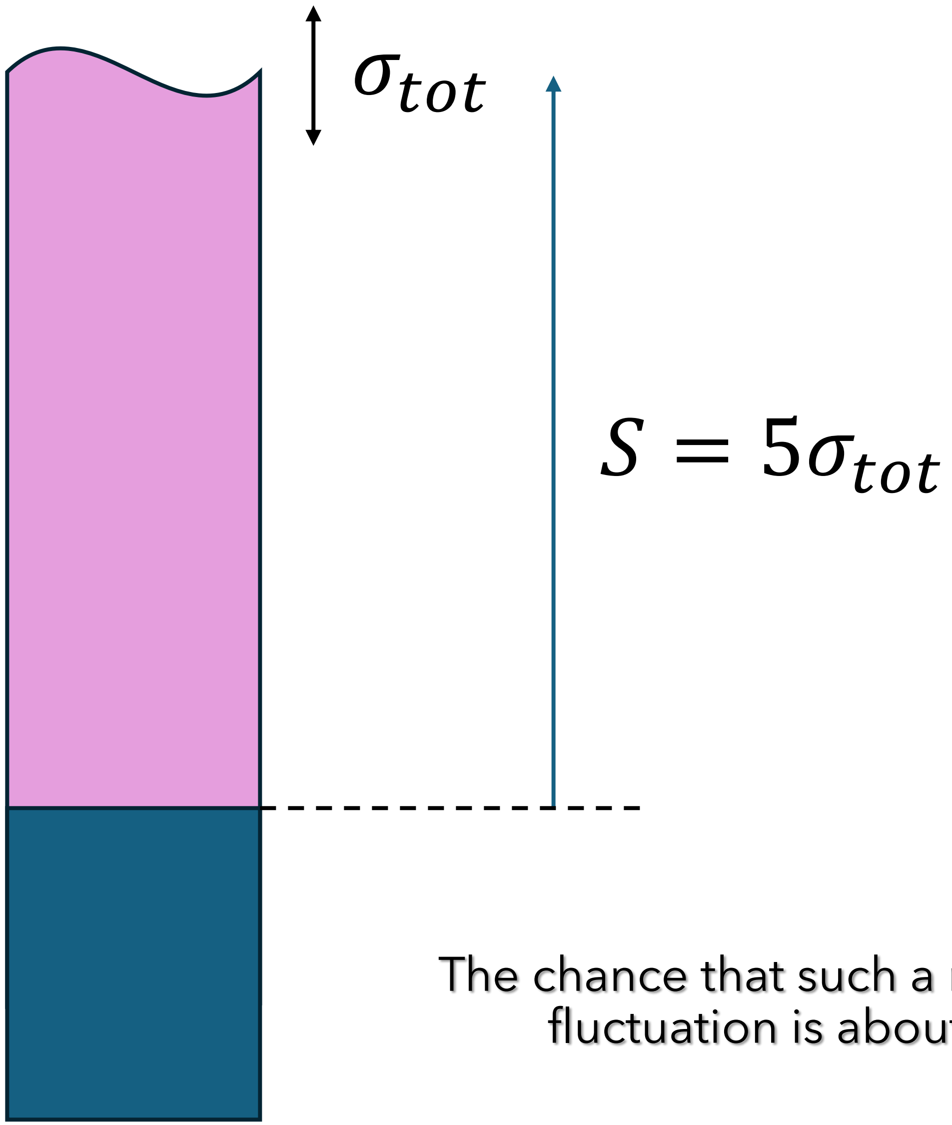


Figure Of Merit = 5

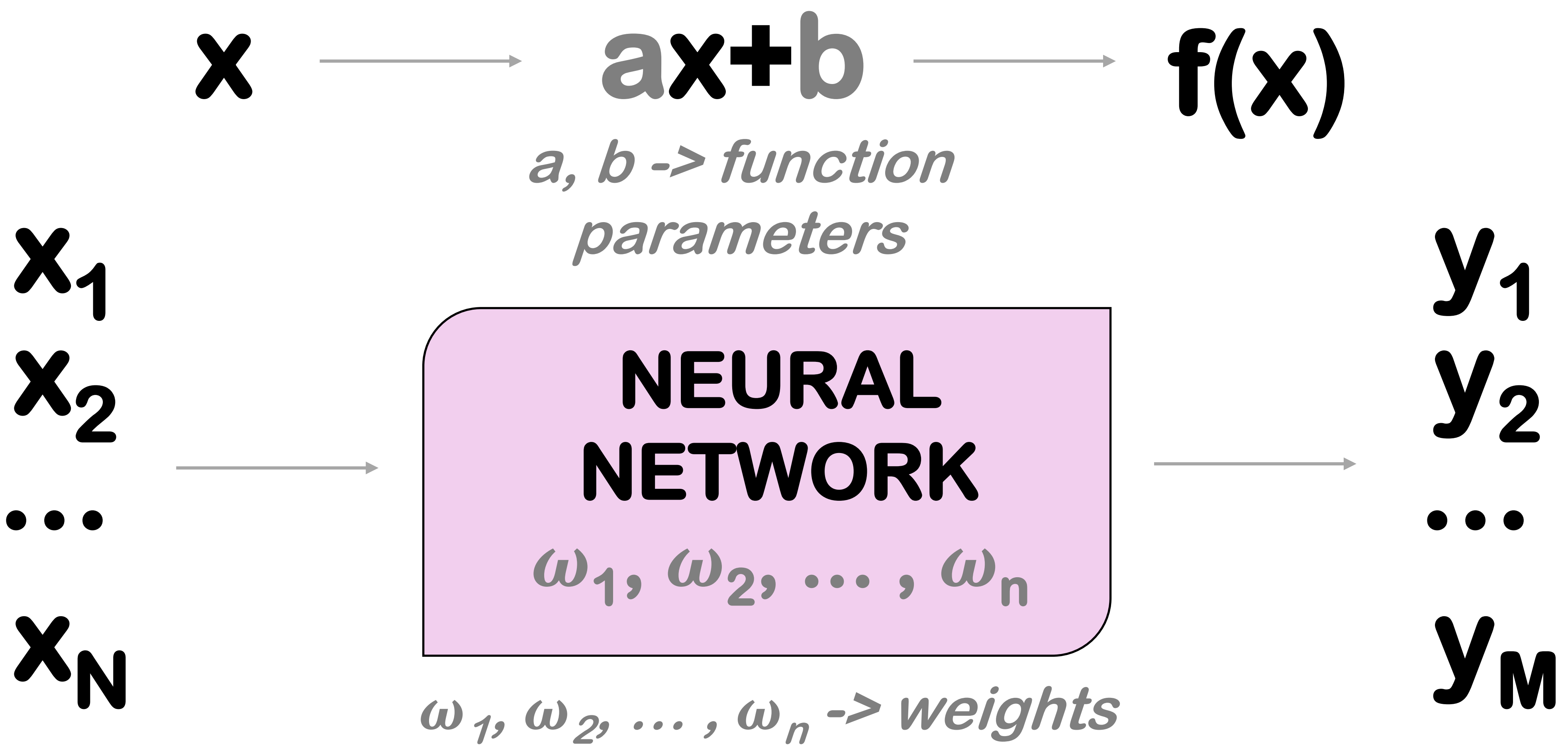
Illustration: Throwing a dice

After throwing the dice 8 times and getting the same number most people would conclude that it is weighted.

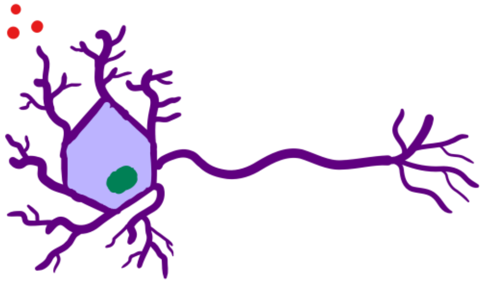
The chance that such a result is due to a fluctuation is about 0.00006%

Neural Network

Function That Produces An Output



Internal Configuration



1 ω_0

x_1 ω_1

x_2 ω_2

$\dots$

x_N ω_N

Σ

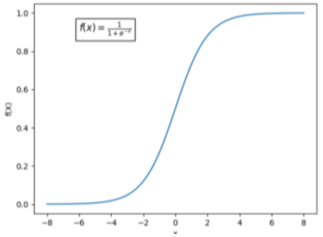
$$\omega_0 + \sum_{i=1}^{N} \omega_i x_i$$

Bias Term

Dot Product

$$g\left(\omega_0 + \sum_{i=1}^{N} \omega_i x_i\right)$$

Apply Nonlinearity (g - activation function)

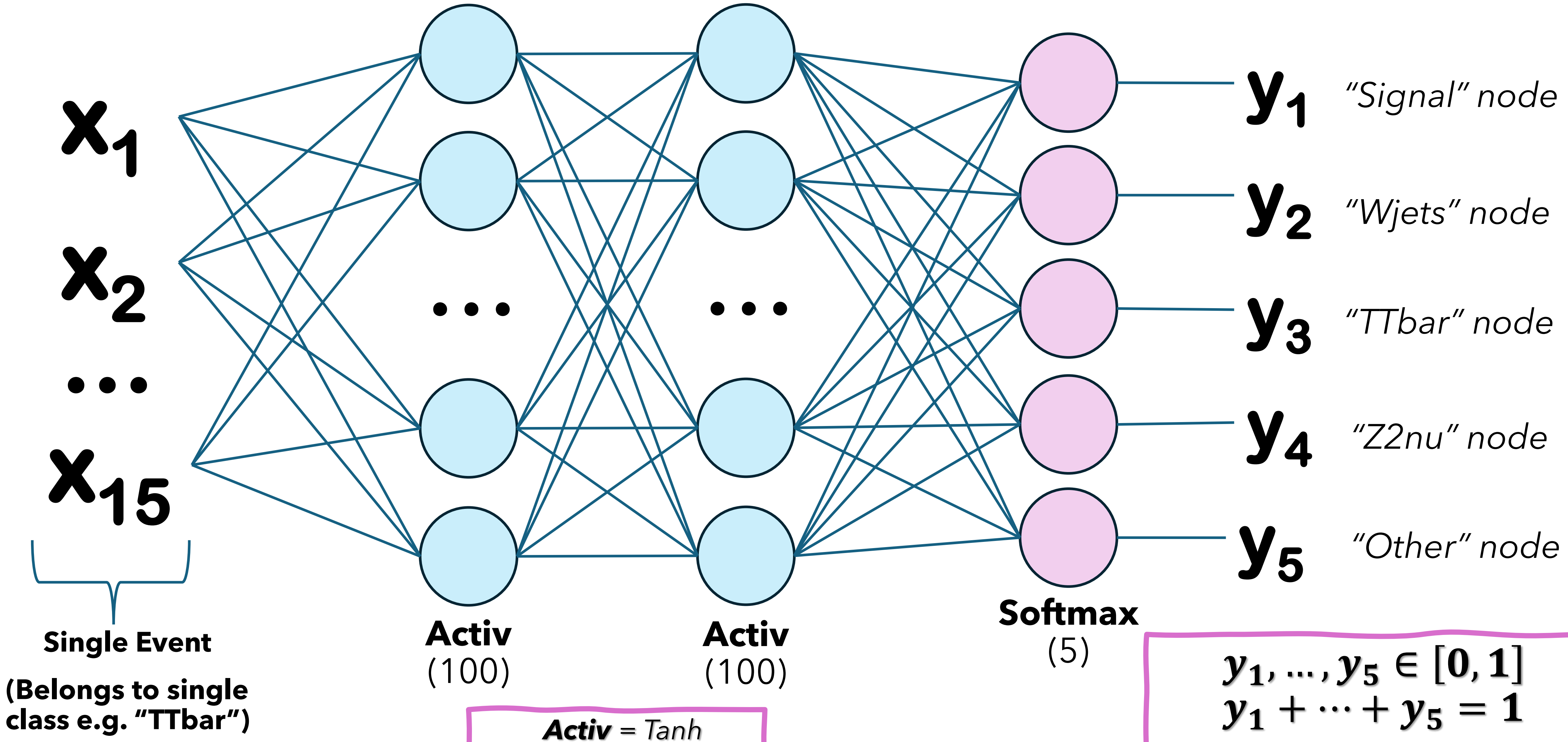


$$y = g\left(\omega_0 + \sum_{i=1}^{N} \omega_i x_i\right)$$

Output

PERCEPTRON (NEURON) – basic unit of neural network

Internal Configuration



Activ = *Tanh*

$$y_1, \dots, y_5 \in [0,1]$$
$$y_1 + \cdots + y_5 = 1$$

Choice Of Input Variables

Choose minimal set of maximally discriminating kinematic variables:

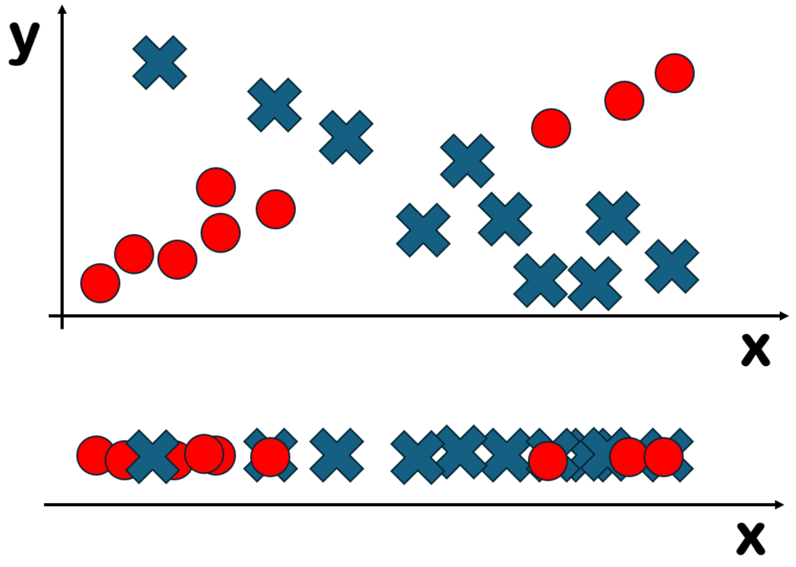
$\underbrace{\text{minimal set}}_{\text{Simplifying the phase space}}$

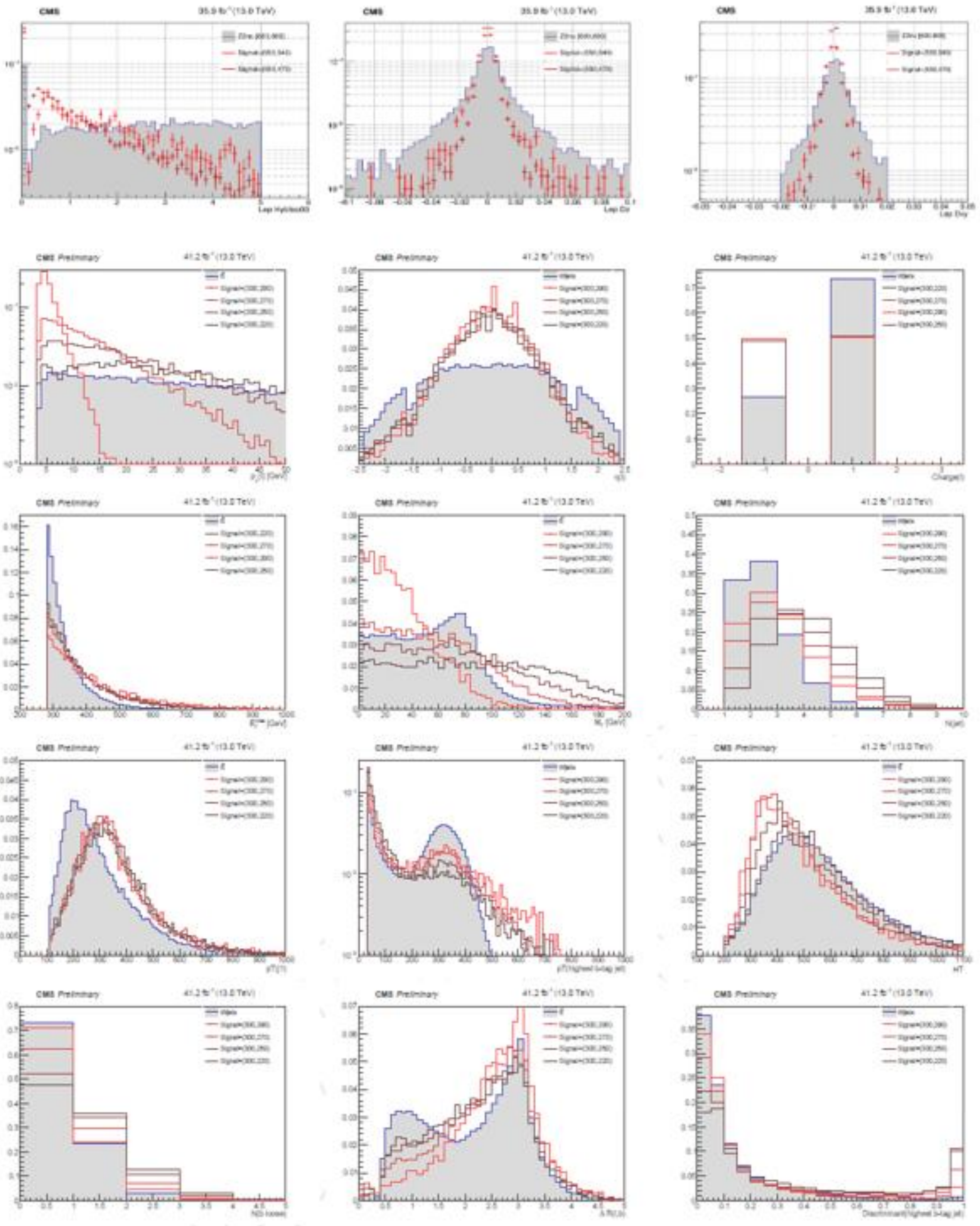
$\underbrace{\text{maximally discriminating}}$ → *Variables that show maximal separation of signal and background event distributions*

$$x_1, \ldots, x_{15}$$

p_T^{lep}	H_T	N_{bloose}
η^{lep}	N_{jet}	p_T^{jetHB}
$LepChg$	p_T^{jet1}	$btag^{jetHB}$
MET	D_z^{lep}	$\Delta R^{jetHB,lep}$
m_t	D_{xy}^{lep}	$AbsIso03^{lep}$

If variables correlate differently for different classes distributions will get further separated in multidimensional phase space





First attempt:

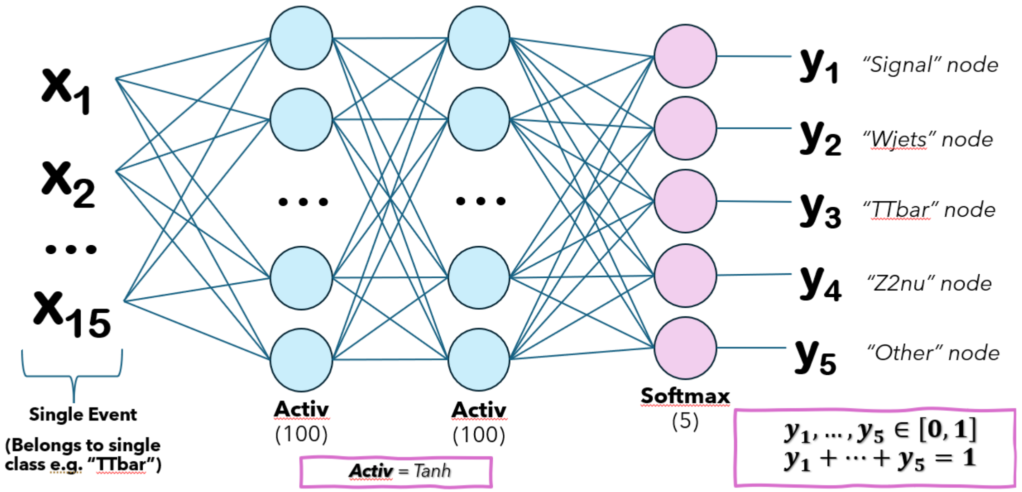
1D cuts on (normalized) event distributions…

$$FOM = 0.027$$

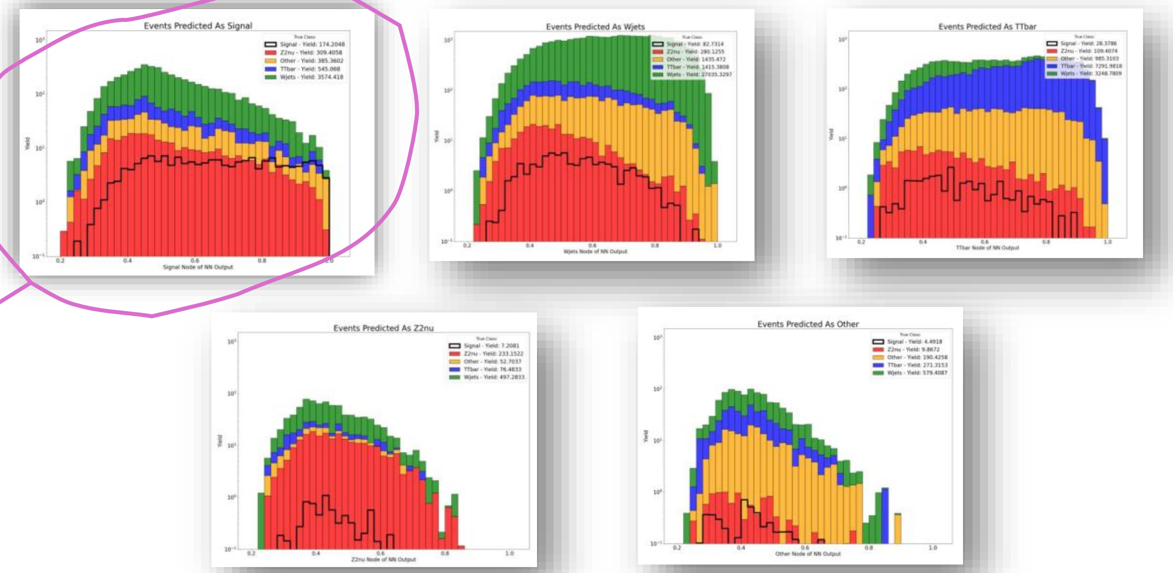
What about separation in ***15D****?*

Event Classification

Simulation Dataset



Classified based on which y_i has the highest value



SIGNAL NODE

- Used to linearly separate signal from background events
- Find cut on the signal node that maximizes FOM

Neural Network Training

$$\mathcal{N}_\omega : \mathbb{R}^N \rightarrow \mathbb{R}^M$$

$$\mathcal{N}_\omega(\mathrm{x}) = \mathrm{y}$$

$$\omega = \{\omega_i\}$$

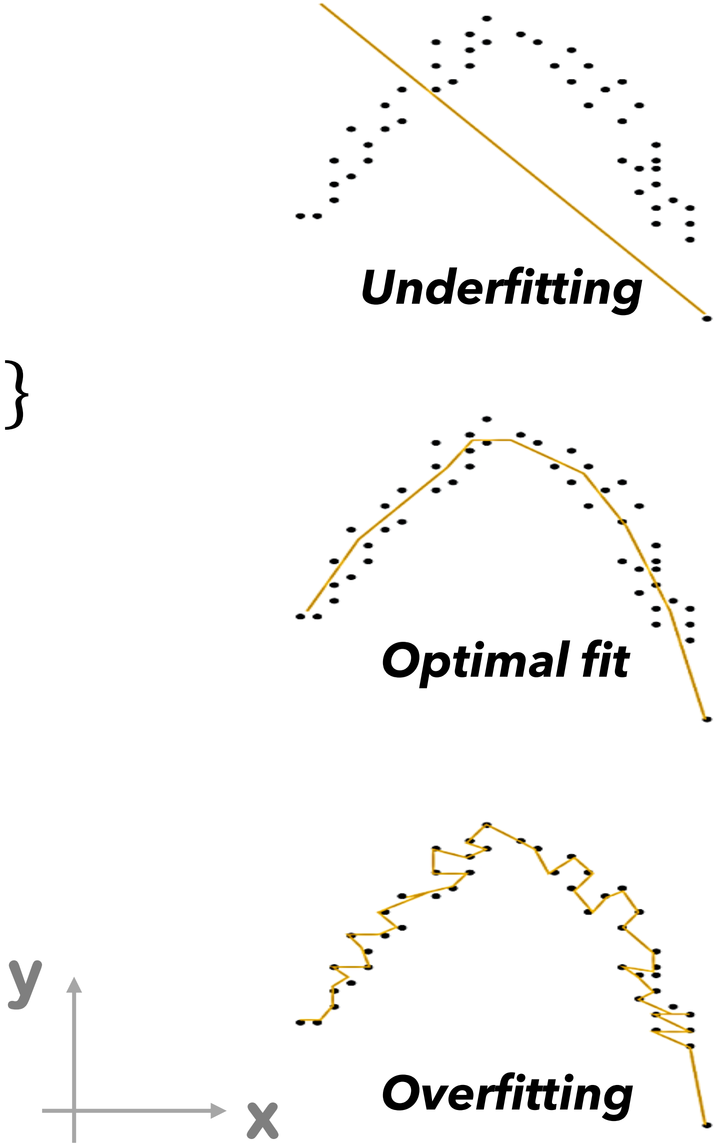
$$\{x^{(n)}\} \longrightarrow \{(x^{(n)}, y^{(n)}_{pred})\} \xleftrightarrow{\text{comparison}} \{(x^{(n)}, y^{(n)}_{true})\}$$

Vectors of kinematic variables — *value of the function $\mathcal{N}_w(\mathrm{x})$* — *Data from simulation*

Find the set of weights w that minimizes the loss function

$$\boxed{\mathcal{L}(\omega) = \frac{1}{K}\sum_k \ell(y^{(k)}_{true}, y^{(k)}_{pred}(\omega))}$$

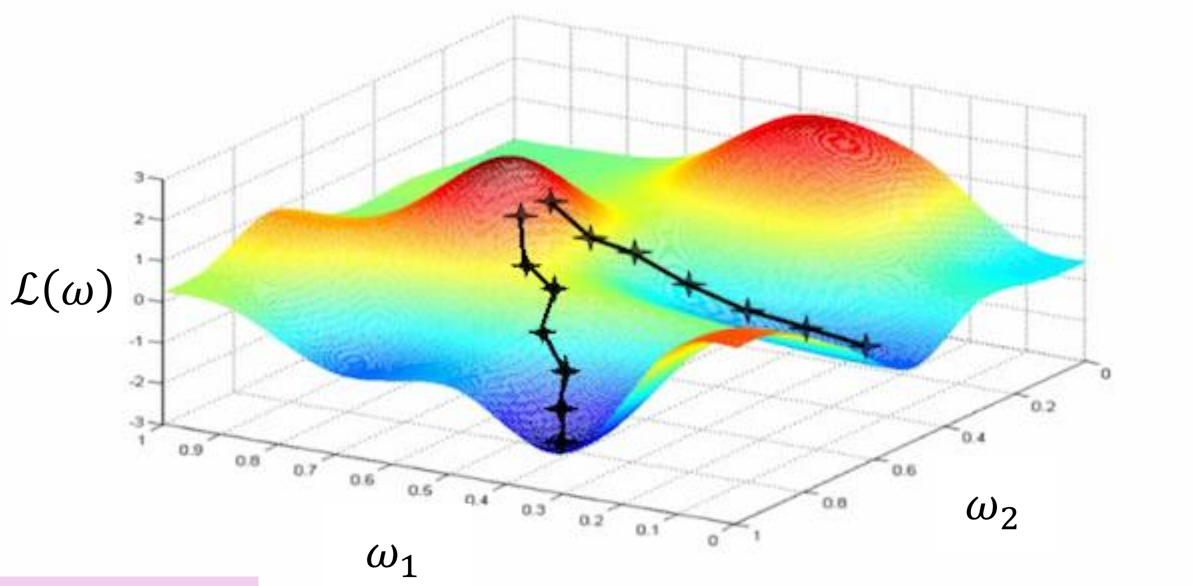
ℓ= Categorical Crossentropy or MSE



Basically, a fitting procedure.

Loss Minimization (Batch Gradient Descent)

$$\mathcal{L}(\omega) = \frac{1}{K}\sum_{k} \ell(y_{true}^{(k)}, y_{pred}^{(k)}(\omega))$$



After each epoch (t) weights are updated:

$$\omega_i^{t+1} = \omega_i^t - Re^{-Dt}\Delta\omega_i^t$$

R=Learning Rate

D=Decay Rate

$$\Delta\omega_i^t = \frac{\partial \mathcal{L}}{\partial \omega_i}\Big|_{\omega=\omega^t} \longrightarrow$$

Computationally too intensive work to calculate over entire training dataset at once!

$\Longrightarrow$ *Split the dataset into smaller batches and do the same procedure!*

Event Balancing

Give in the neural network training larger weight to events that are more likely to happen:

$$\mathcal{L}(\omega) = \frac{1}{K}\sum_{k} \tilde{w}_k \, \ell(y_{true}^{(k)}, y_{pred}^{(k)}(\omega))$$

Treat the weights of each class separately:

$$\mathcal{L}(\omega) = \frac{1}{K}\sum_{C=1}^{5}\sum_{k_C=1}^{N_C} \tilde{w}_{k_C}^{C} \, \ell(y_{true}^{(k_C)}, y_{pred}^{(k_C)}(\omega))$$

Larger weight means larger error for misidentification → "Neural network is more focused on events that are given larger weight"

Neural Network Training – Event Balancing

<u>*5 CLASSES*</u>: C = “Signal”, “Wjets”, “TTbar”, “Z2nu”, “Other”

$$\tilde{w}_i^C = \underbrace{\frac{w_i^C}{\sum_i w_i^C}}\,\underbrace{N_S} \times \boxed{?}$$

Put all classes on equal footing

- All classes are equally numbered – summation over all the events of a given class gives 1
- Reflects **local differences** (within a single class) on an event-to-event basis

Number of events in class C

- Scaling factor
- Doesn’t have large effect on the performance
- Chosen for numerical reasons

w_i^C - number of i-th events that is expected to be seen in the experiment (~ *probability that a given event will occur*)

Neural Network Training – Event Balancing

<u>5 CLASSES</u>: C = "Signal", "Wjets", "TTbar", "Z2nu", "Other"

$$\tilde{w}_i^C = \frac{w_i^C}{\sum_j w_j^C} N_S \times f_C$$

Choice: $f_C = \xi_C + \delta$ $\qquad \delta = 0, 1, 2, 3, 4, 5, \infty$ $\qquad \xi_C = \frac{\sum_i w_i^C}{\sum_i w_i^S}$

Case: $f_C = 1 \quad \forall C$

(Classes are equally balanced)

In principle, f_C could be any positive real number…

Neural Network Training – Event Balancing

Bookkeeping Suggestion:

→ Divide by the average value: $\tilde{w}_i^C = \frac{w_i^C}{\sum_j w_j^C} N_S \times \frac{f_C}{\bar{f}}$ $\quad \bar{f} = \frac{1}{5}\sum_C f_C$

Multiplying the scaling factor by a constant doesn't influence the performance of the neural network (doesn't change position of the minimum in the loss function)

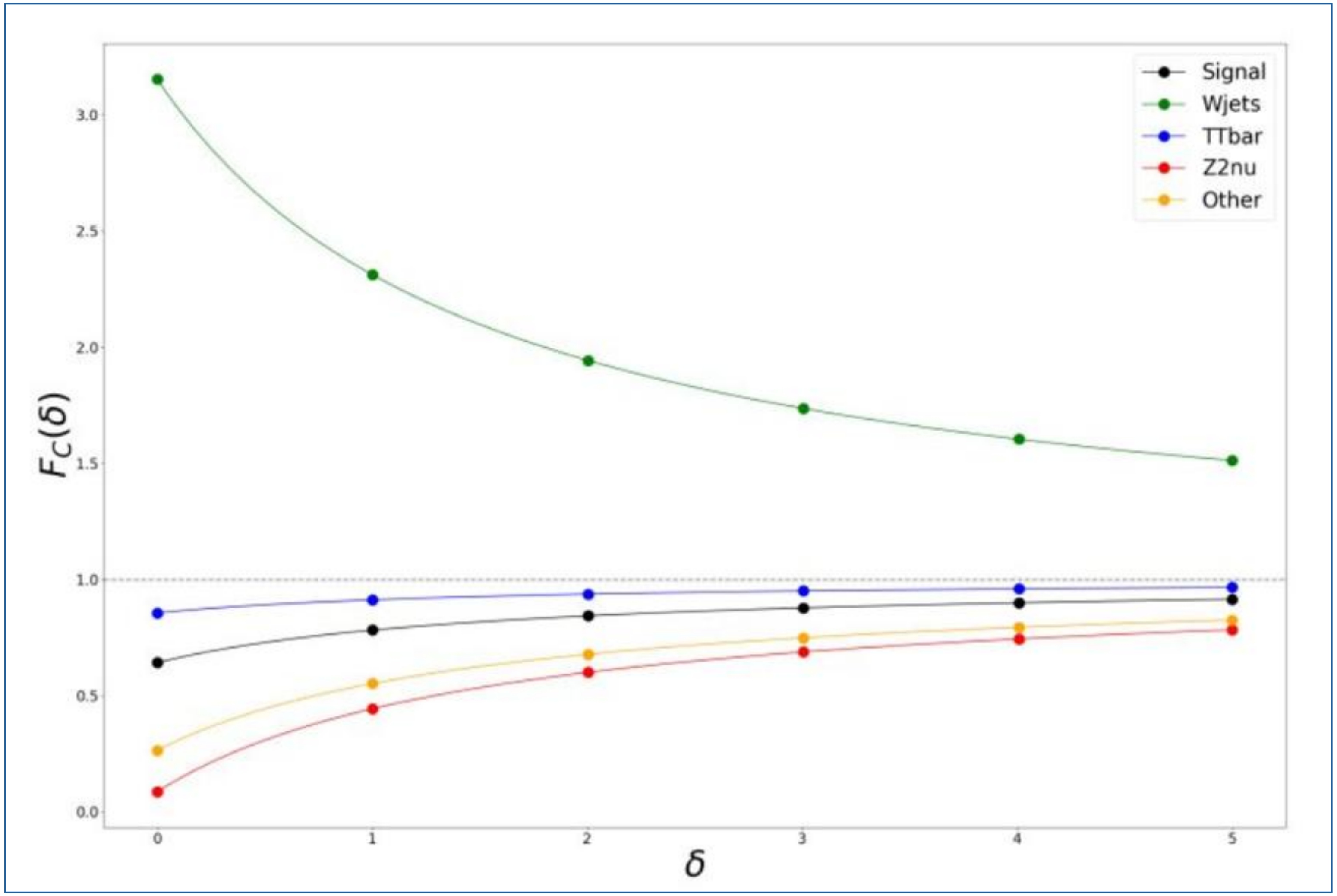
$$\boxed{f_C \;\Leftrightarrow\; \alpha f_C;\ \alpha = const.}$$

→ Idea: Scaling factors $\{f_C\}$ that have the same $\{F_C\}$ result in the same overall performance

Def. $$\boxed{F_C = \frac{f_C}{\bar{f}} \qquad \bar{f} = \frac{1}{5}\sum_C f(\xi_C)}$$

Neural Network Training – Event Balancing

With given choices for $\boldsymbol{f_C}$ *classes are gradually getting more and more balanced…*



$$\boldsymbol{f_C = \xi_C + \delta}$$

$$\boldsymbol{\delta = 0, 1, 2, 3, 4, 5, \infty}$$

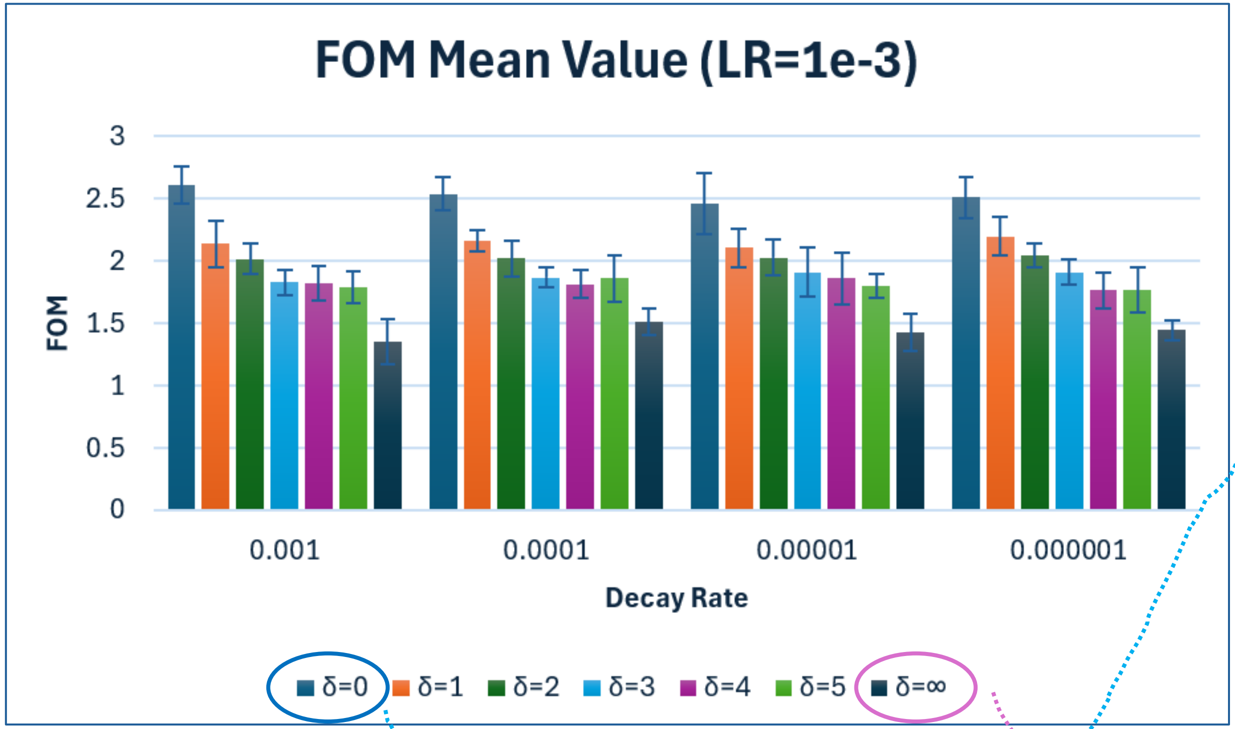
… this is reflected in the values of the $\boldsymbol{F_C}$ *factors.*

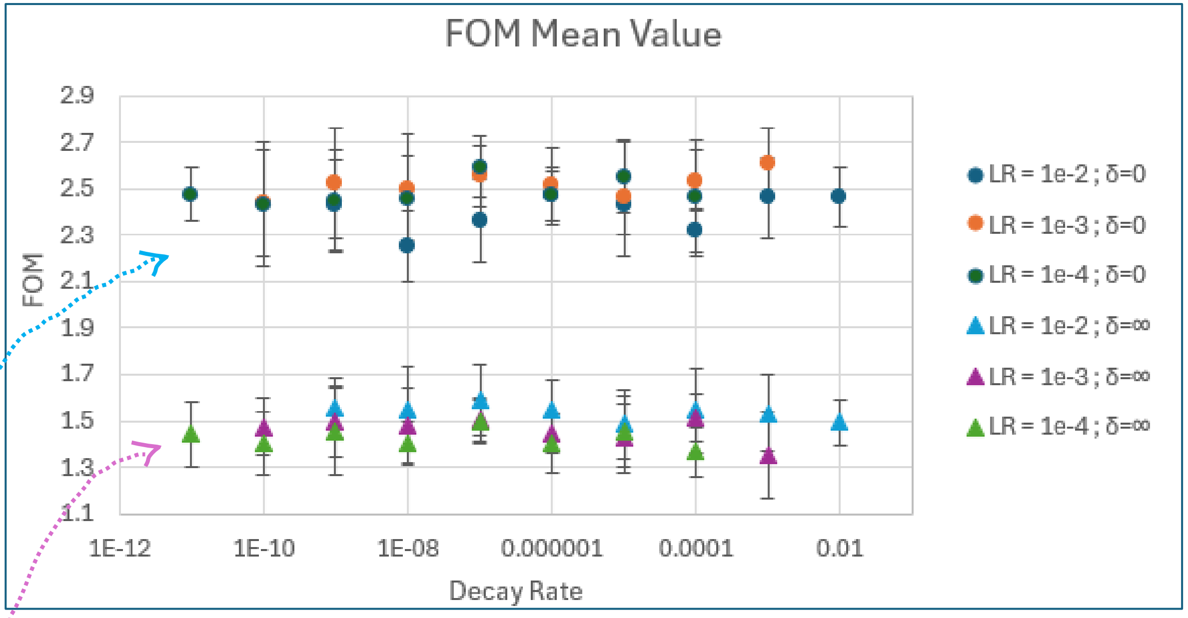
$$F_C = \frac{f_C}{\bar{f}}$$

Results

*For six different balancing schemes characterized with **δ** ten neural network trainings were conducted and FOM maximized*

<u>Average of the FOM over these 10 trainings</u> and its standard deviation are reported…



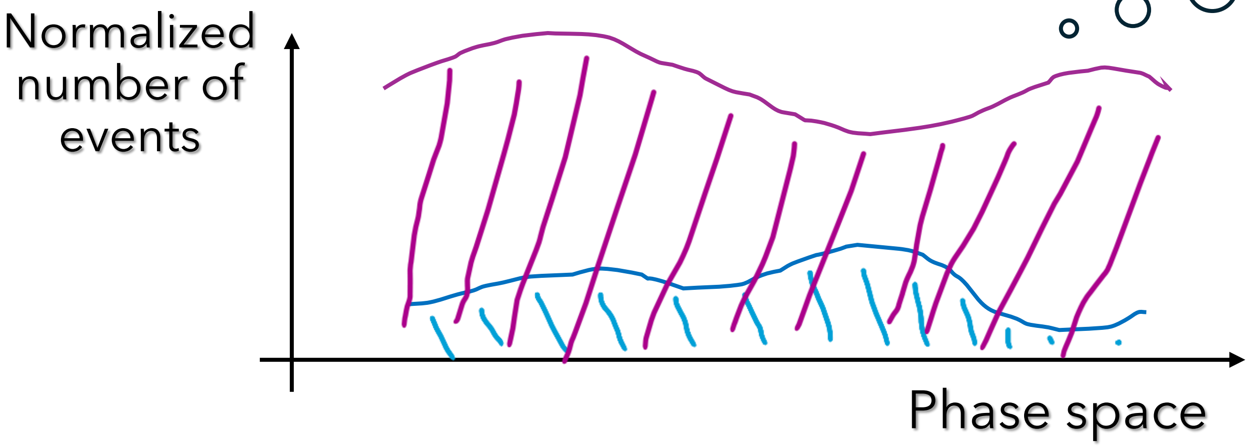


Learning Rate
Decay Rate
→ *NN training parameters*

Results

What is happening?

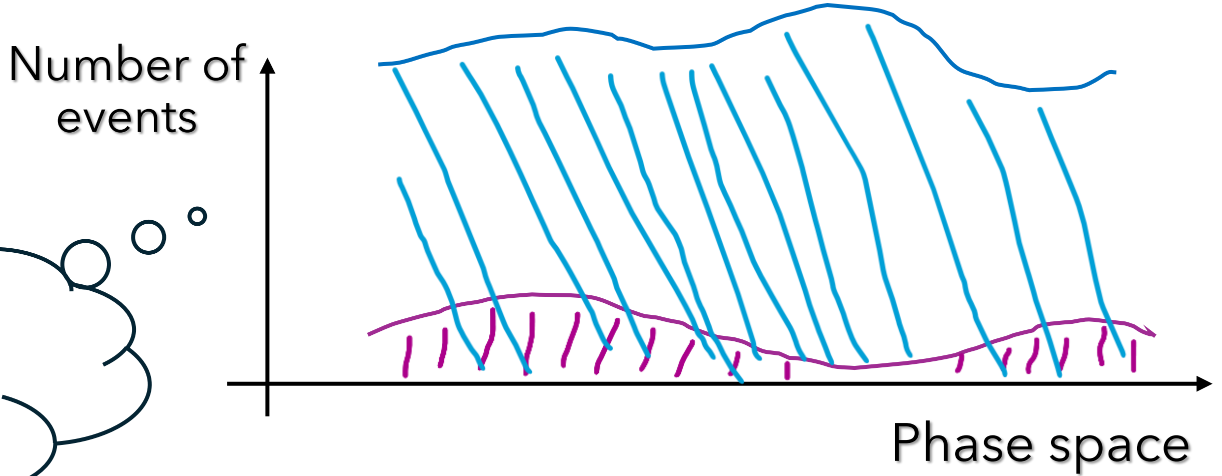
These events probably go to the high values signal output...



Wjets
Signal

Normalized distributions

$$\boxed{\boldsymbol{\delta} = \infty}$$



...but really, they shouldn't go there.

Real relationship

Mimicked with other choces of $\boldsymbol{\delta}$

What is the best choice?

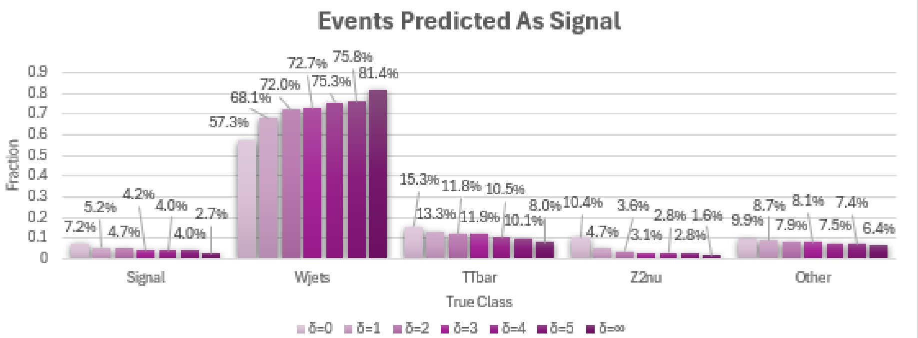
Depends on what is the goal…

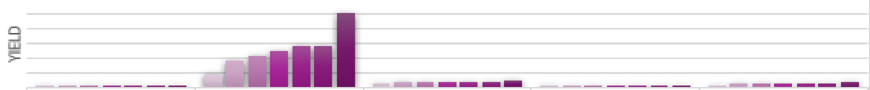
(1)	**Signal region** *(Areas in phase space enriched in signal)*	*Events behind the cut that maximizes FOM on the* ***signal node*** *of NN output*	*"Find signal"*
(2)	**Control regions** *(Areas in phase space that are enriched in dominant SM backgrounds)*	*Events distributed on the* ***background nodes*** *of NN output*	*Improves the precision of the final analysis (Reduces uncertainty)*

Signal Node

Control Regions

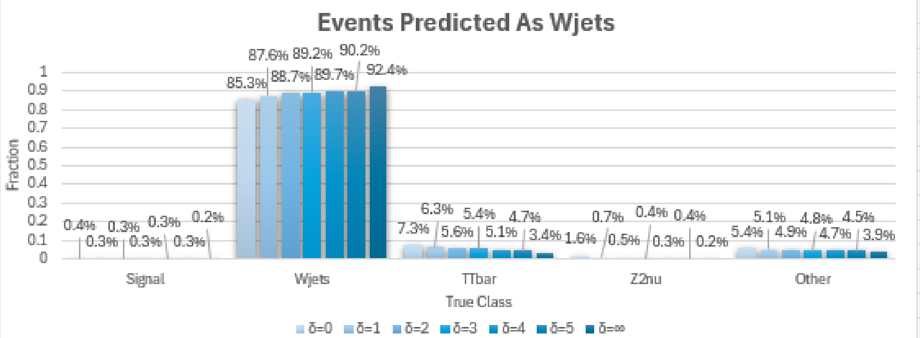
Monitor…
… SIGNAL CONTAMINATION
… CROSS CONTAMINATION

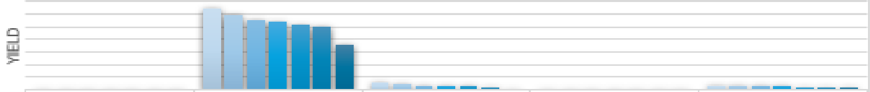




	Signal	Wjets	TTbar	Z2nu	Other
δ=0	114	912	244	166	157
δ=1	139	1820	356	127	232
δ=2	140	2150	352	107	237
δ=3	147	2520	411	106	281
δ=4	148	2790	389	103	277
δ=5	149	2840	377	103	276
δ=∞	168	5100	499	97.7	399

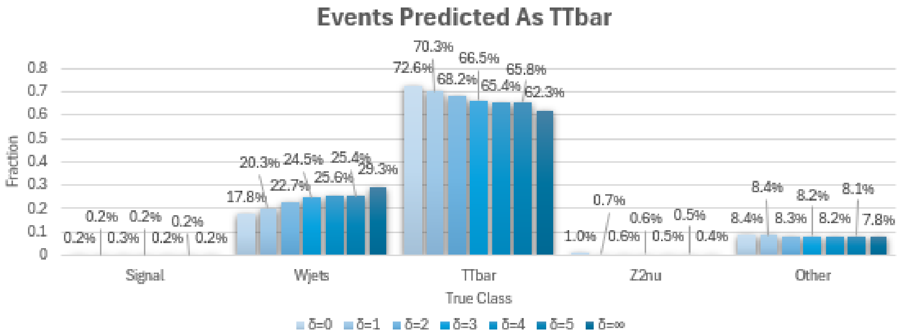
δ=0 δ=1 δ=2 δ=3 δ=4 δ=5 δ=∞

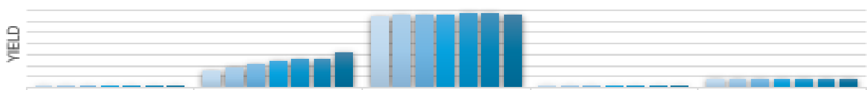




	Signal	Wjets	TTbar	Z2nu	Other
δ=0	157	32100	2730	599	2030
δ=1	109	29600	2130	229	1710
δ=2	93	27900	1760	151	1540
δ=3	84	26900	1620	127	1440
δ=4	76.4	25800	1460	94.9	1340
δ=5	70.5	25100	1320	97.5	1250
δ=∞	34.5	18200	668	32.1	762

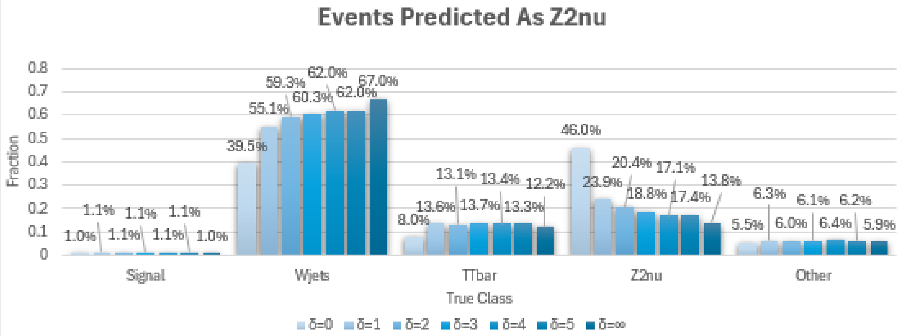
δ=0 δ=1 δ=2 δ=3 δ=4 δ=5 δ=∞

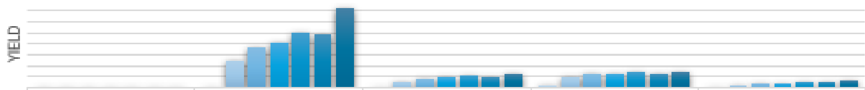




	Signal	Wjets	TTbar	Z2nu	Other
δ=0	20.8	1580	6450	90.6	748
δ=1	20	1890	6550	65.1	786
δ=2	24.6	2220	6670	59.4	811
δ=3	21	2460	6660	55.7	824
δ=4	22.3	2630	6710	53.3	843
δ=5	22	2580	6690	54.9	823
δ=∞	21.2	3130	6640	47.6	827

δ=0 δ=1 δ=2 δ=3 δ=4 δ=5 δ=∞





	Signal	Wjets	TTbar	Z2nu	Other
δ=0	1.94	76.4	15.4	88.9	10.7
δ=1	25	1200	295	519	137
δ=2	32.8	1800	398	620	183
δ=3	37.4	2080	474	649	210
δ=4	42	2480	536	685	256
δ=5	42.2	2400	516	674	241
δ=∞	54.4	3600	656	743	318

δ=0 δ=1 δ=2 δ=3 δ=4 δ=5 δ=∞

Conclusion

Two possible lines of work:

(1) Use multiclass neural network as a binary classifier (choose $\boldsymbol{\delta = 0}$)

Benefit: Large figure of merit FOM=2.6

Shortcoming: Control regions are searched in phase space orthogonal to the one where search of the signal region is conducted (affects precision of final analysis)

(2) Use multiclass neural network in a standard way (choose $\boldsymbol{\delta = 2}$)

Benefit: Control regions are found in the same space where signal search is conducted (better precision of the final analysis)

Shortcoming: Slightly lower value of the figure of merit FOM=2

Previous research (binary classifiers): FOM(BDT)=1.6 FOM(NN)=1.8