# Hunting dark matter signals with deep learning at the LHC

**Dr. Andres D. Perez**
**IFLP-CONICET, Argentina**

Work in colaboration with:
Ernesto Arganda (IFT), Anibal Medina (IFLP),
and Alejandro Szynkman (IFLP)

I F L P
CONICET
U N L P

April 28, 2021
@ I Workshop de Computing y Software de la Red Española de LHC

# Plan

- **Models and sample generation**
- **Neural Network algorithms**
  - Event-by-even data
  - Data as 2D histograms
    - Performance invariance with the number of background events
- **Multimodel Classifiers**
- **Conclusions**

# Models and sample generation

Simplified models
Kinematic features
Benchmark models

# Simplified models

**Monojet plus missing transverse energy channel**

- **DM with a spin-0 mediator**
- **DM with a spin-1 mediator**
- **DM with a spin-2 mediator**

$pp \to DM\, DM\, j$

- **Axion-Like Particle (ALP) as DM**

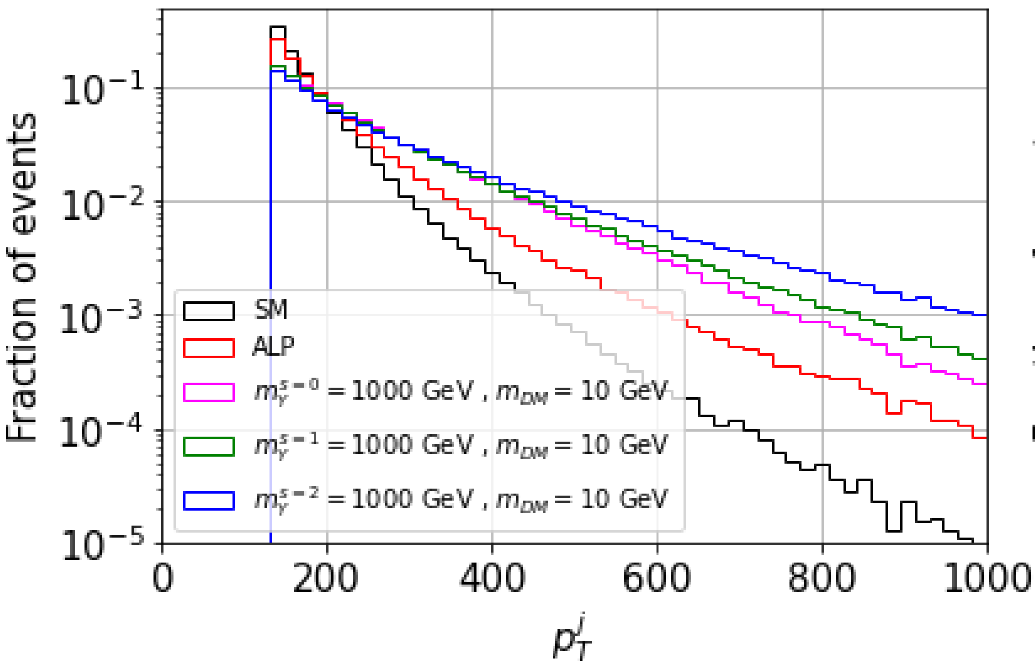
$pp \to a\, j$

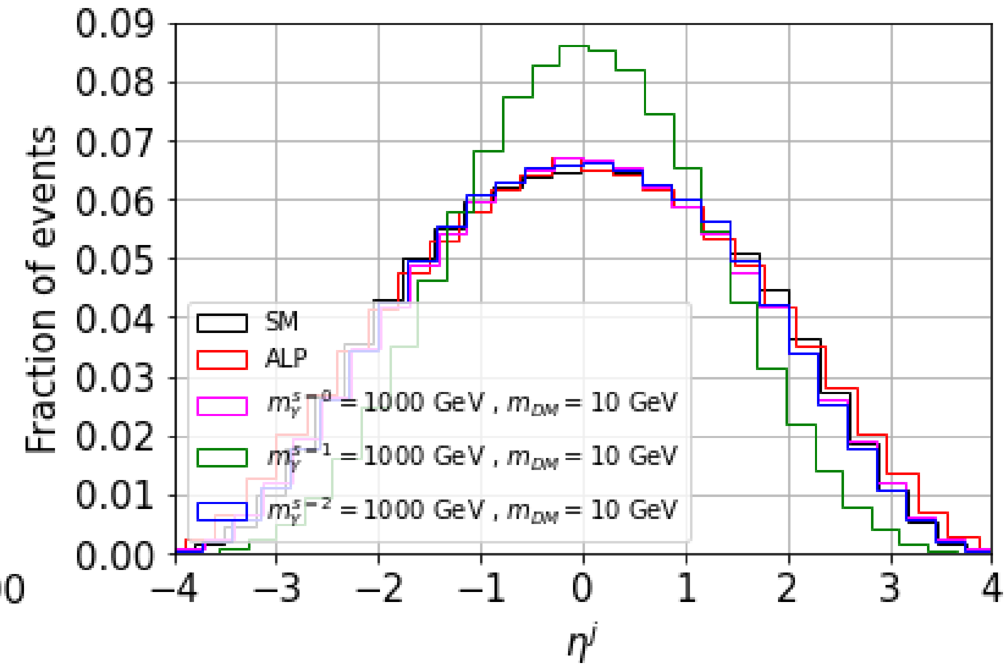
- **SM background**

$pp \to Z\, j\ (Z \to \nu\nu)$

# Simplified models

Each event has the monojet kinematic information **($p_T^j$, $\eta^j$, $\Phi^j$)**




**1.** The azimuthal angle distribution does not show any useful structure.

**2.** The coupling values do not modify the kinematic distributions.

We simulated **1.5M SM events** and **0.5M New Physics events**

**MadGraph5_aMC@NLO** to generate events with monojets plus missing energy at parton level.
Parton shower and hadronization are performed with **Pythia.**
Detector-level data is simulated using **Delphes** with the default ATLAS card.

**$\sqrt{s}$= 14TeV** generation level cuts: $p_j^T \geq 130$GeV and $|\eta_j| \leq 5$ for the leading jet.

# Neural Networks algorithms

- Event-by-even data
- Data as 2D histograms
  - Performance invariance with the number of background events

# DNN with Event-by-event data

We simulated **1.5M SM events** and **0.5M New Physics events**

Each event has the monojet kinematic information **($p_T^j$, $\eta^j$, $\Phi^j$)**

## Input

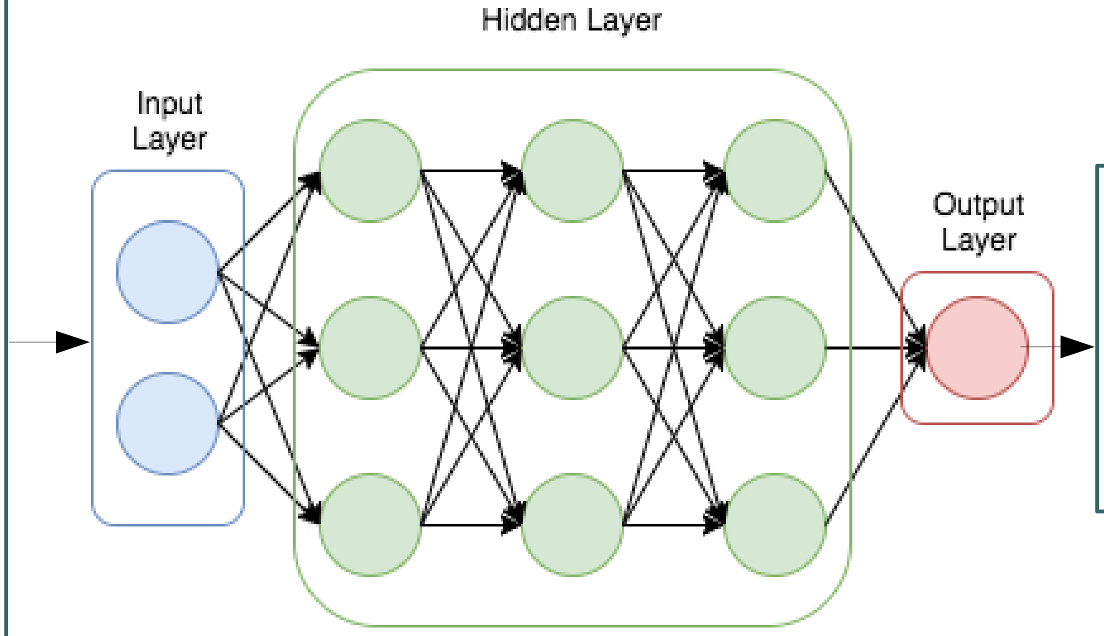
Each data sample is a single event:

**Event 1** → ($p_T^j$, $\eta^j$, $\Phi^j$) → SM → **labeled '0'**

**Event 2** → ($p_T^j$, $\eta^j$, $\Phi^j$) → New Physics → **labeled '1**

…

**Event N** → ($p_T^j$, $\eta^j$, $\Phi^j$) → SM → **labeled '0'**




## Output

Number:

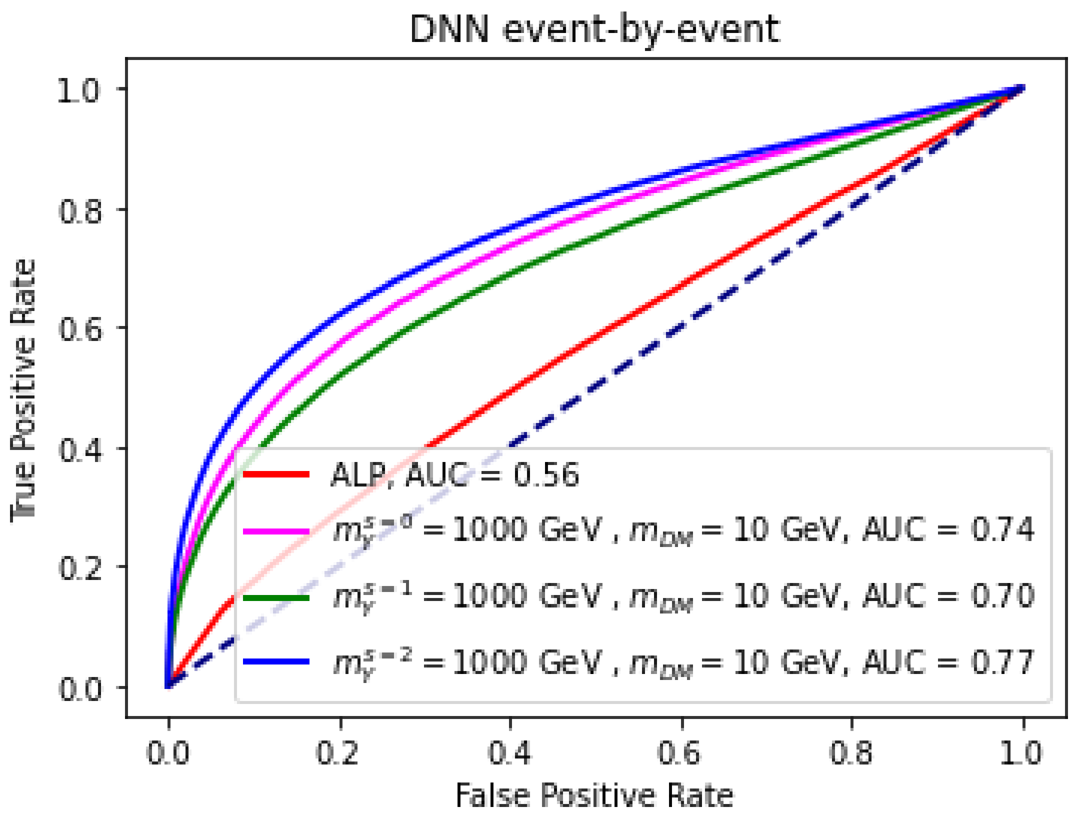
**[0,1]**

Data samples are divided with a 0.64:0.20:0.16 train-test-validation ratio

Trained each benchmark model vs SM **individually**.

# DNN with Event-by-event data

Receiver Operating Characteristic (ROC) curves:




**Poor performance**

The area under the ROC curve (AUC), a conventional metric to test the performance of binary classifiers

**AUC=1** is a perfect classifier, and **AUC=0.5** represents a random classifier

# DNN with data as 2D histograms

S: # NP events
B: # SM events

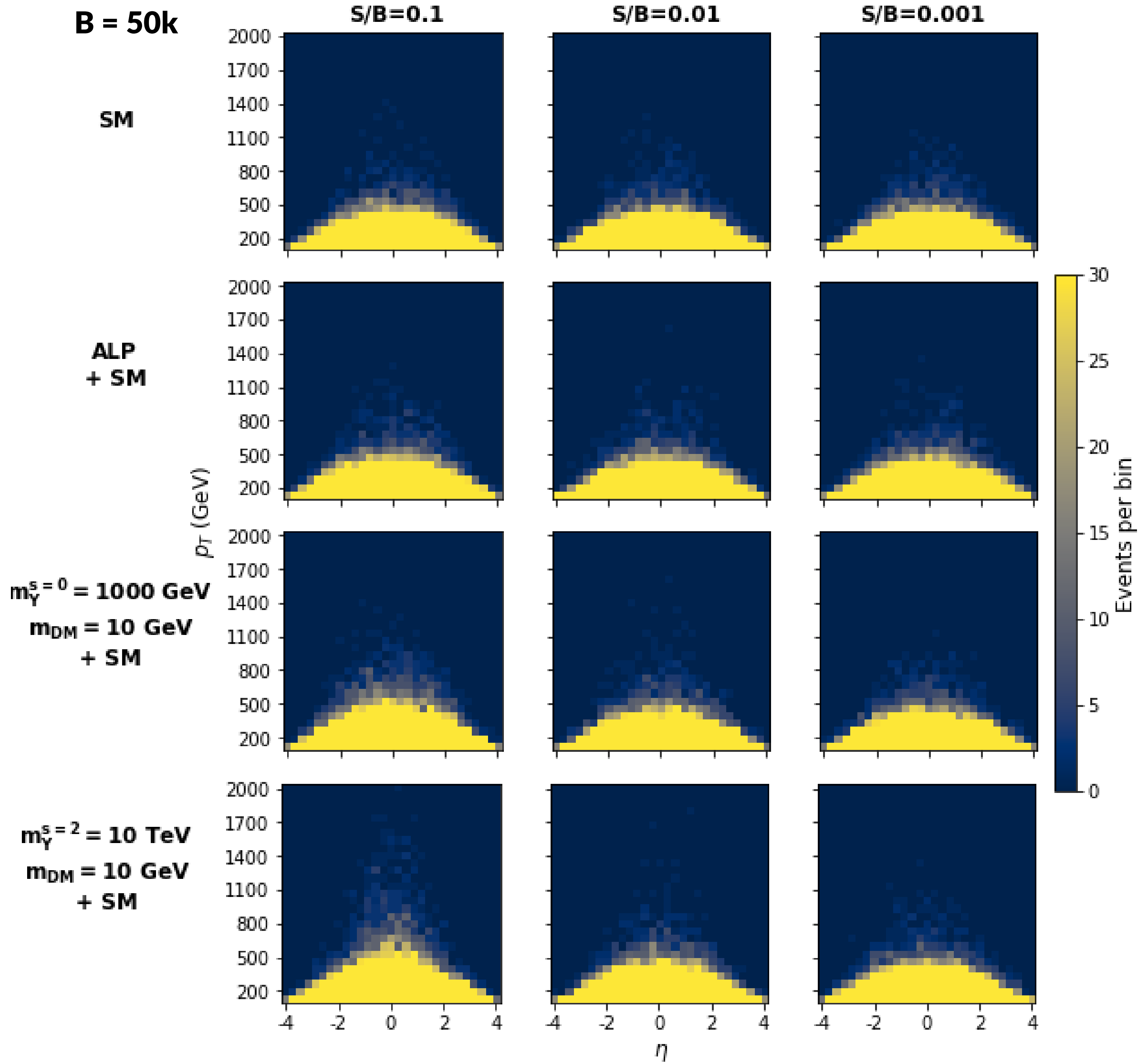
The jet azimuthal angle $\Phi^j$ does not provide any useful information.

We can construct 2D histograms made from the pair $(p_T^j, \eta^j)$

**. 20k histograms** with only SM events

**. 20k histograms** with NP + SM events

per benchmark model and per S/B ratio

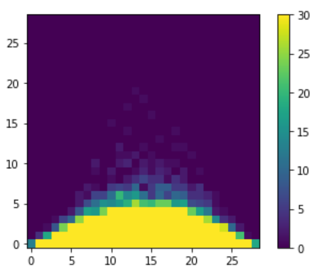


# DNN with data as 2D histograms

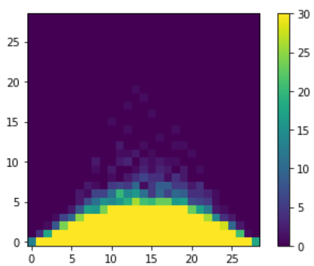
We simulated
**20k SM only histograms** and
**20k New Physics + SM histograms** (per benchmark model and per S/B ratio)

## Input

Each data sample is a single histogram:

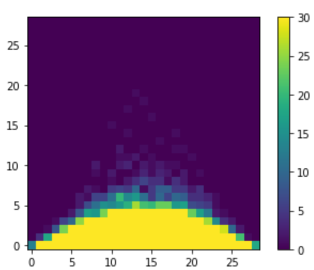


Data sample 1
**SM only**
**labeled ‘0’**

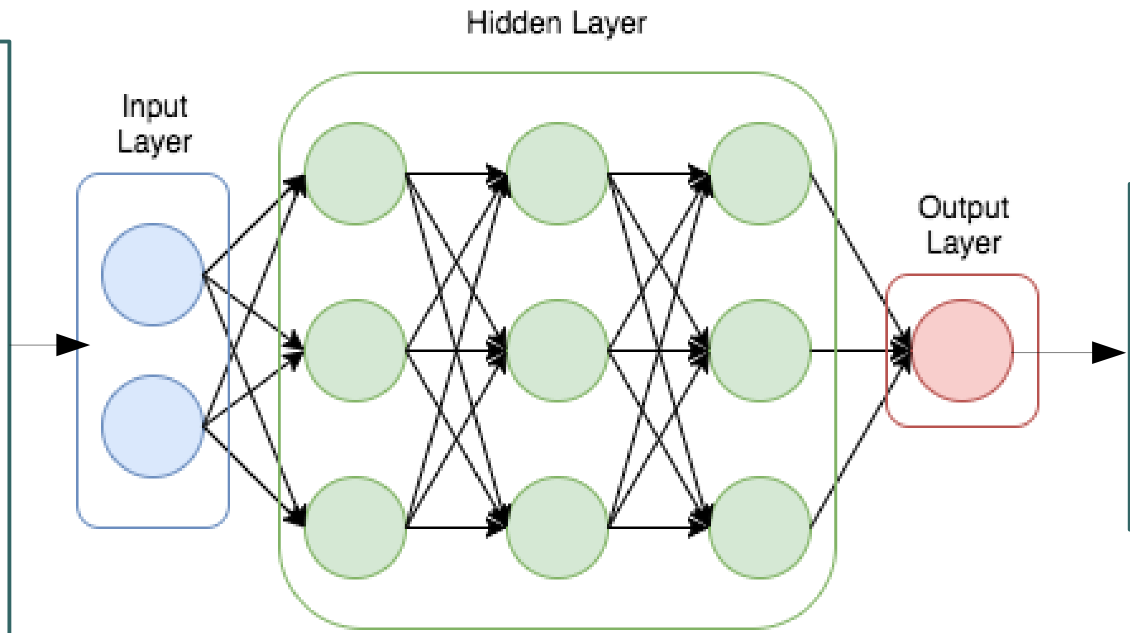


Data sample 2
**NP + SM**
**labeled ‘1’**

... ...



Data sample N
**SM only**
**labeled ‘0’**




## Output

Number:

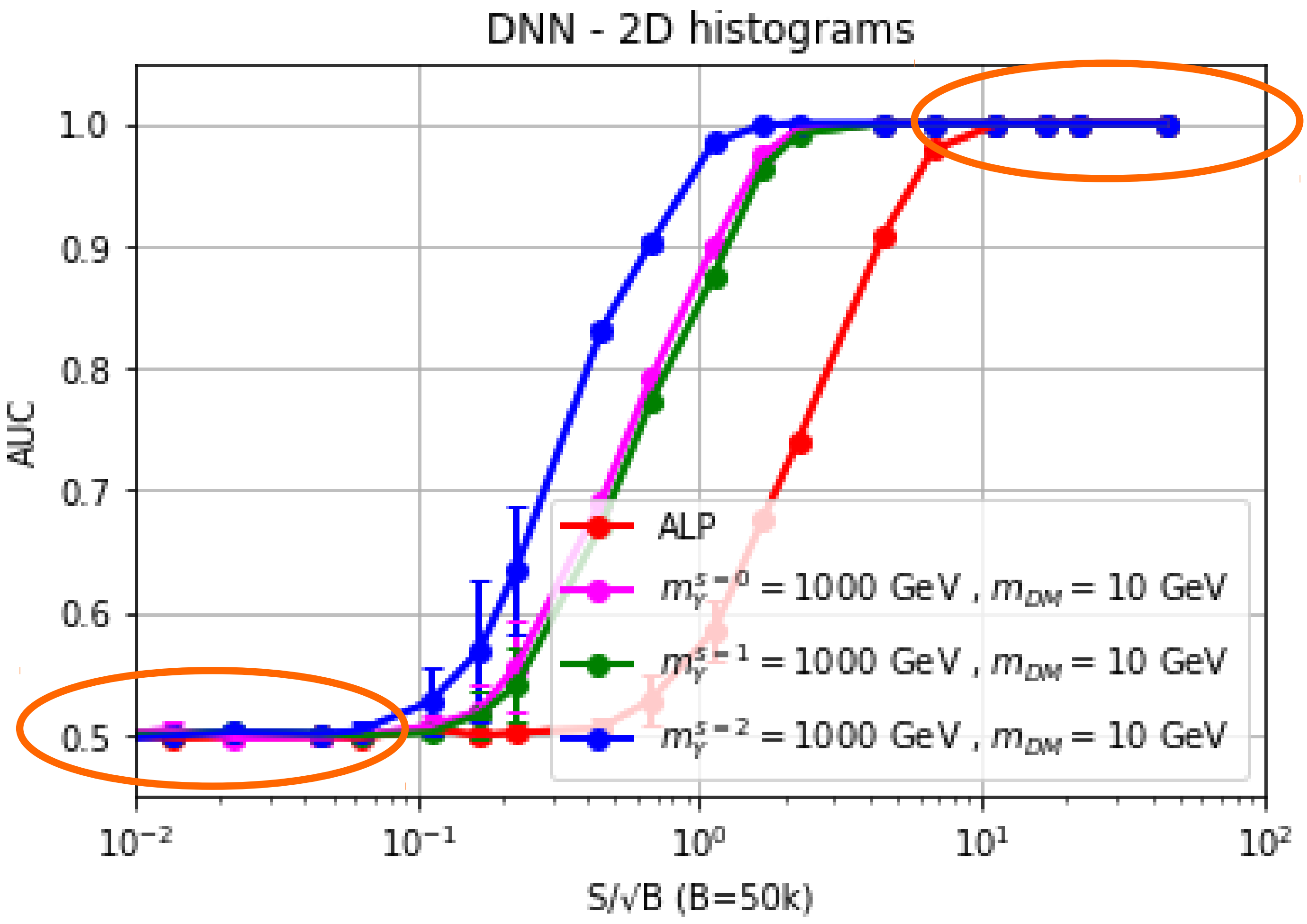
**[0,1]**

DNN trained to discriminate:
**histograms** with SM only events vs
**histograms** with NP+SM events

Trained each benchmark model vs SM **individually**.

# DNN with data as 2D histograms

Each point represents a DNN trained with a data set with a specific benchmark, S and B

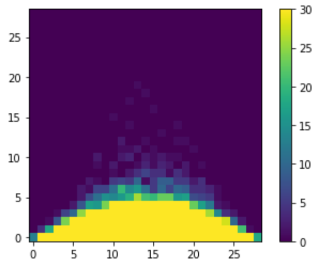



**Great performance!**

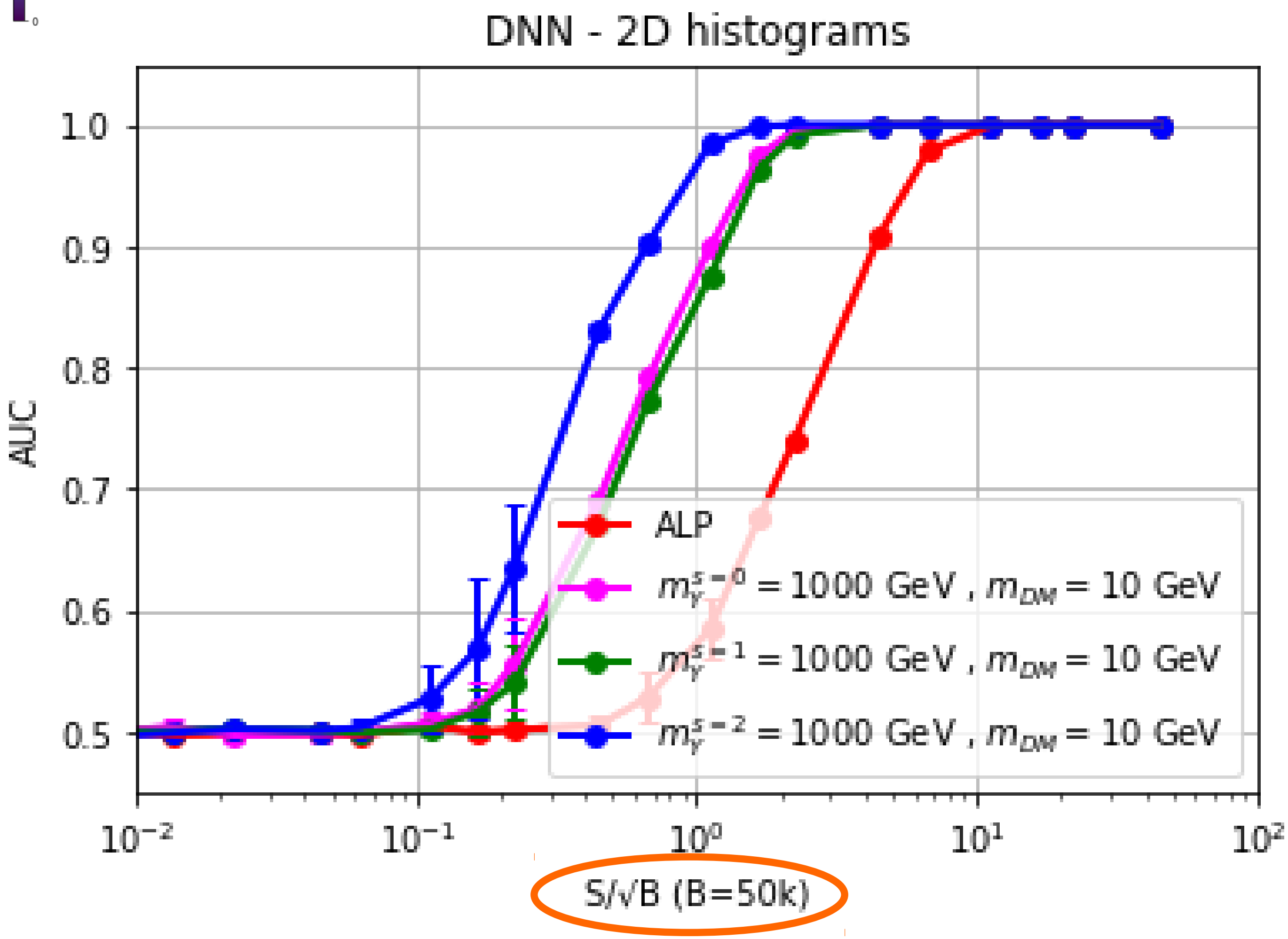
**AUC=1** is a perfect classifier, and **AUC=0.5** represents a random classifier

# Performance invariance with B

S: # NP events
B: # SM events

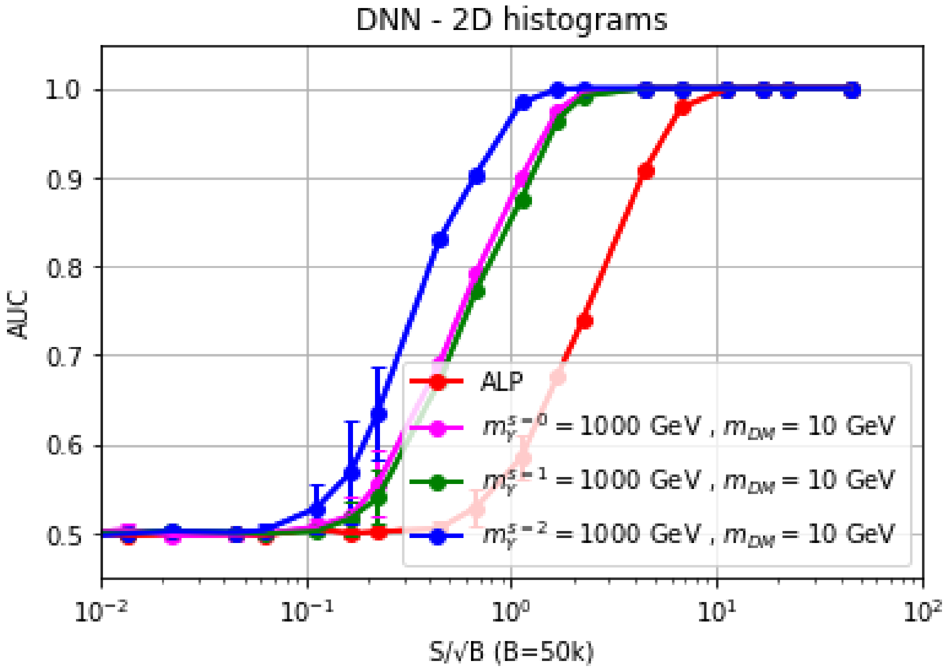


Before, each 2D histogram constructed with **B = 50k** SM events + **S** New Physics events




Performance is not modified significantly for different values of B, if the results are presented as a function of **S/√B**.

# Performance invariance with B




To know if a DNN with 2D histograms could distinguish a particular new physics model from the SM background, we only need to:

- **Identify the curve of the corresponding benchmark model**
- **Calculate the model cross section for the chosen couplings**
- **Calculate the SM background cross section**
- **Calculate S/√B for any luminosity, and check the corresponding AUC**

Also, we can have an idea of the luminosity needed to obtain a given efficiency. Change the last step for:

- **Identify the S/√B value for the corresponding AUC you would like to get and calculate the luminosity needed**

# events = cross section * luminosity * detector efficiency

# Multimodel classifiers

Multiclass classifier

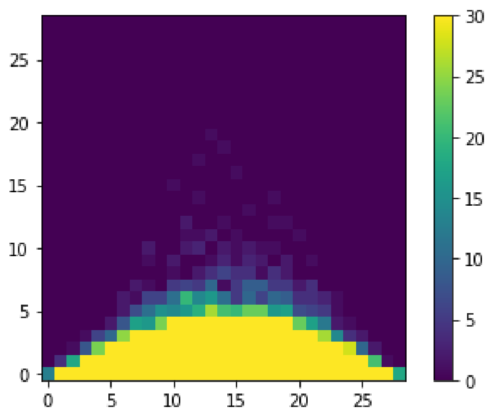
# Multiclass classifier

A single DNN trained with **several** new physics models:

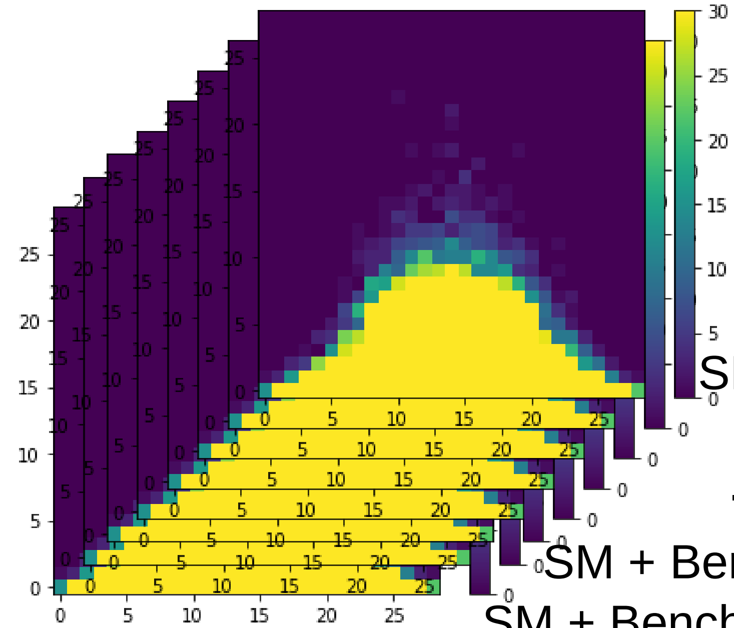
**SM only** vs **Benchmark 1 + SM** vs … vs **Benchmark N + SM**

**(labeled '0')** **(labeled '1')** **(labeled 'N')**

## Training the DNN



SM only → **"0"**



10k histograms per benchmark and per S/B ratio

SM + Benchmark 7 → **"7"**

...

SM + Benchmark 2 → **"2"**

SM + Benchmark 1 → **"1"**

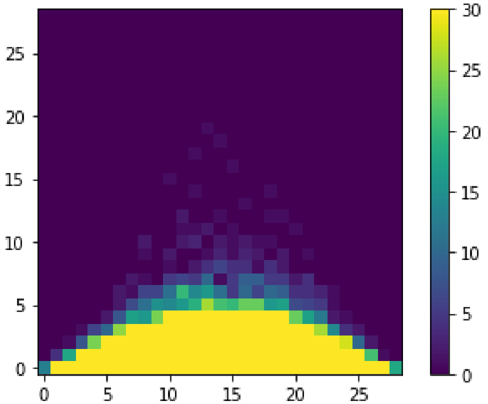
**DNN**

(In this work, 7 NP+SM models and SM only)

# Multiclass classifier

## Test the DNN

(trained with **7 NP+SM** models and **SM only**)




SM? NP? → **Label "??"**

→ **DNN** —*output*→ **Array of 8 elements** between **[0,1]**

Each element is the probability that the histogram belongs to a training model

For example

| **Label** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
|---|---|---|---|---|---|---|---|---|
| **Array** | [ 0.09 | 0.03 | 0.01 | **0.84** | 0.005 | 0.005 | 0.01 | 0.01 ] |

Prob(SM)

Prob(benchmark model 3 + SM)

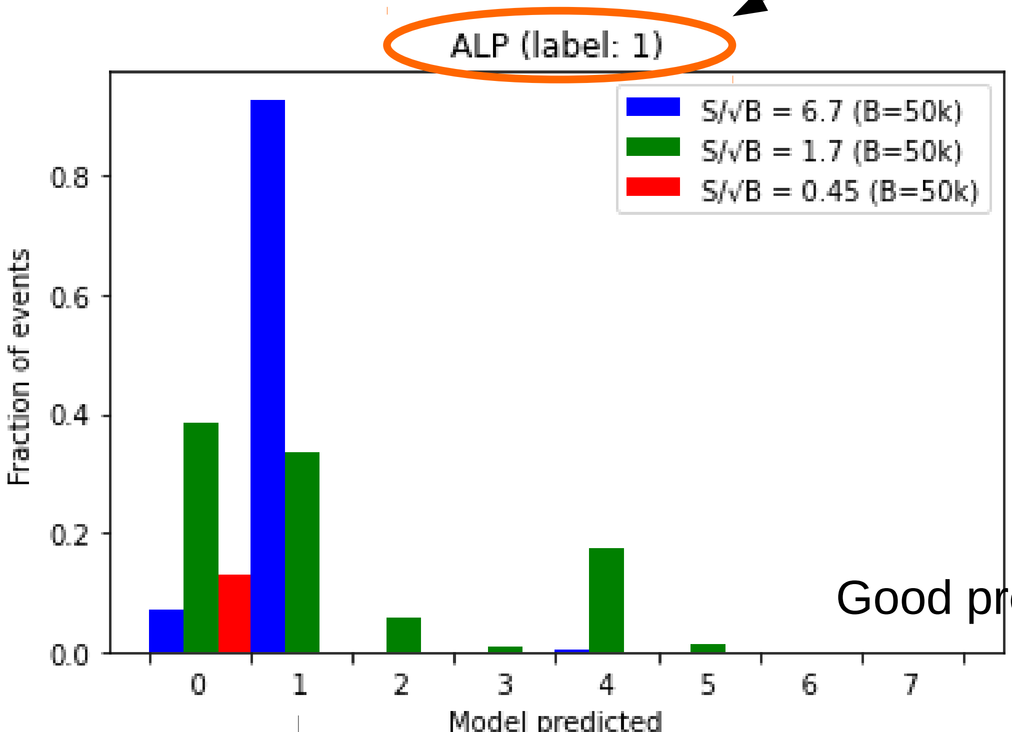
# Multiclass classifier

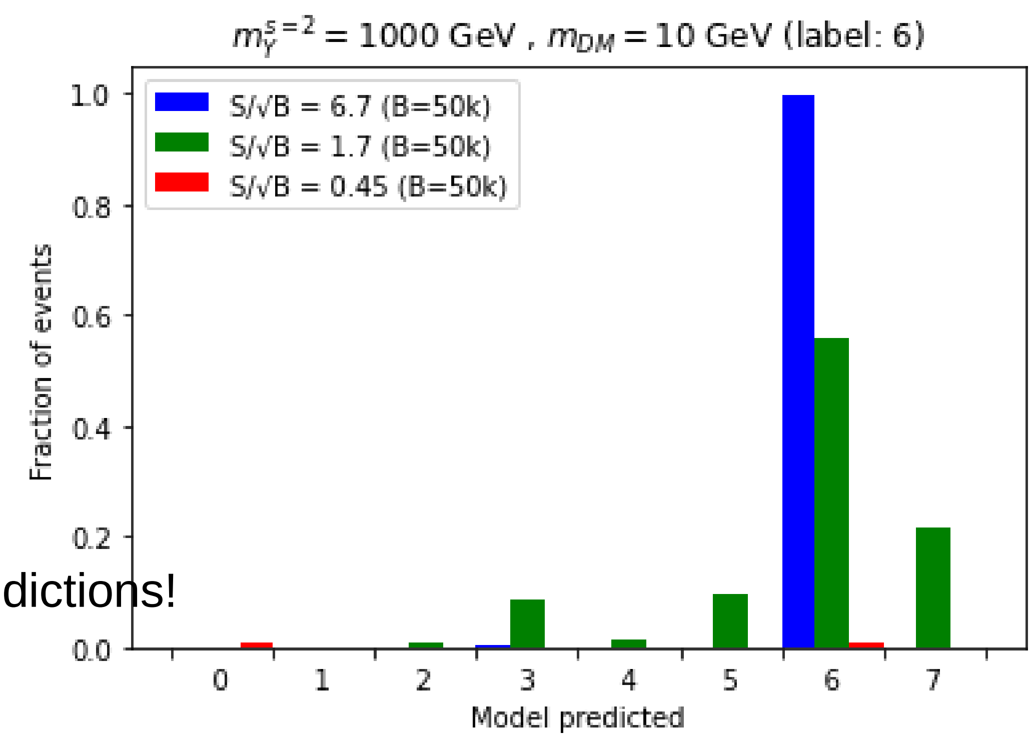
Histogram of the **frequency of occurrence** can be constructed

Which model is predicted by the DNN?

## Testing with training models

Input model



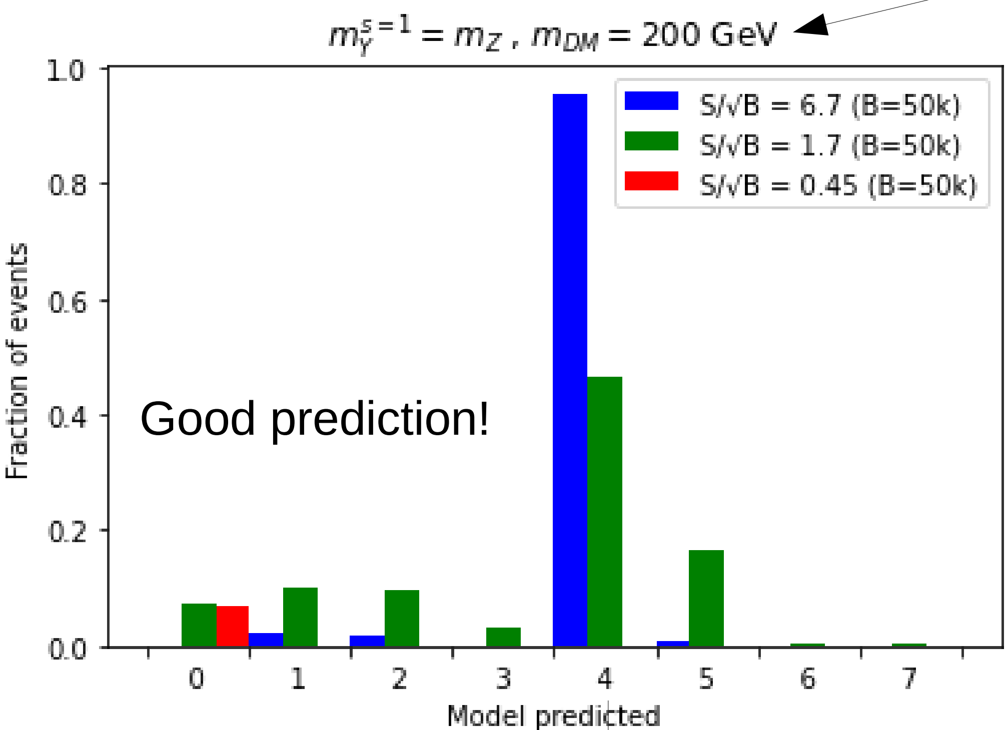

Good predictions!

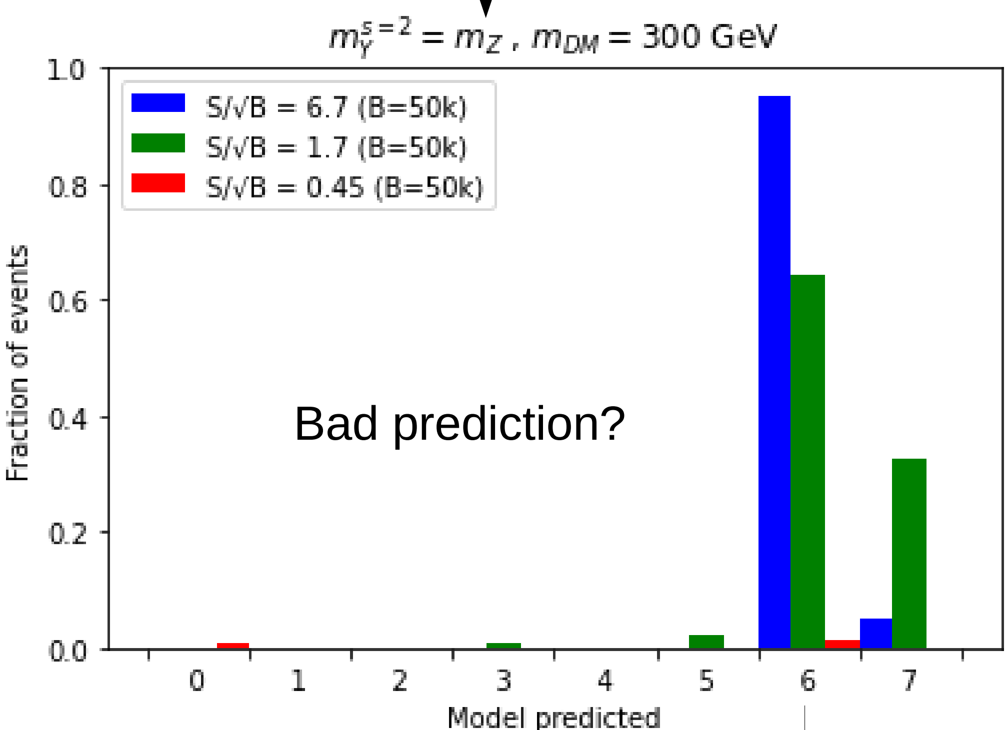
Model predicted

# Multiclass classifier

## Testing with non-training models

**Input models completely new to the DNN**



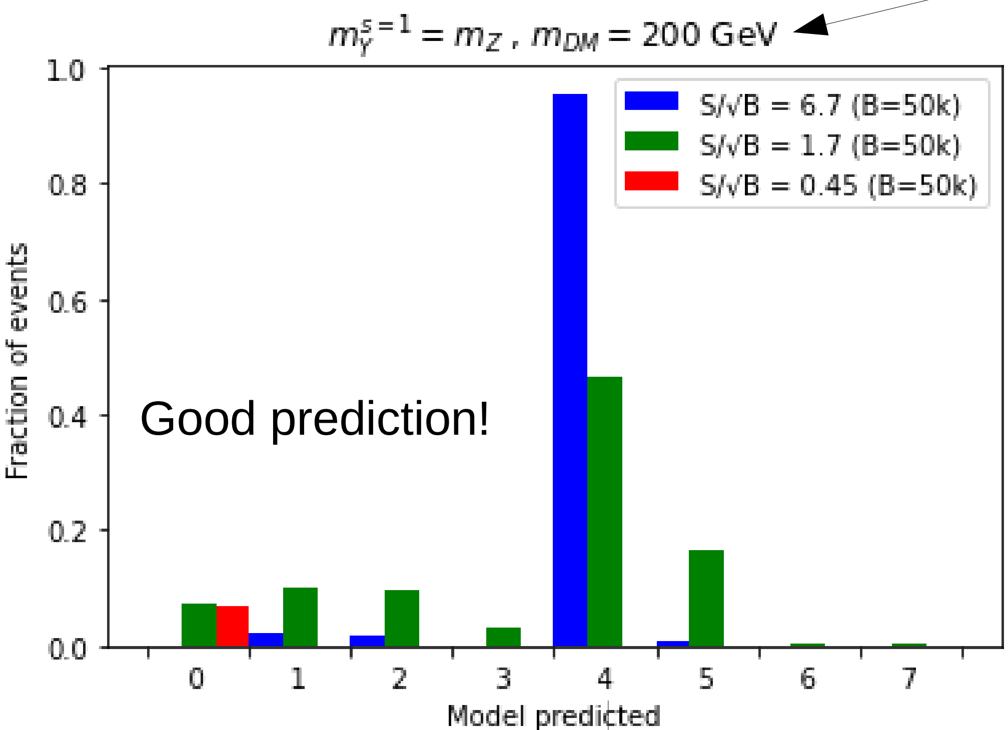

$m_Y^{s=1} = m_Z$ , $m_{DM}$=300 GeV

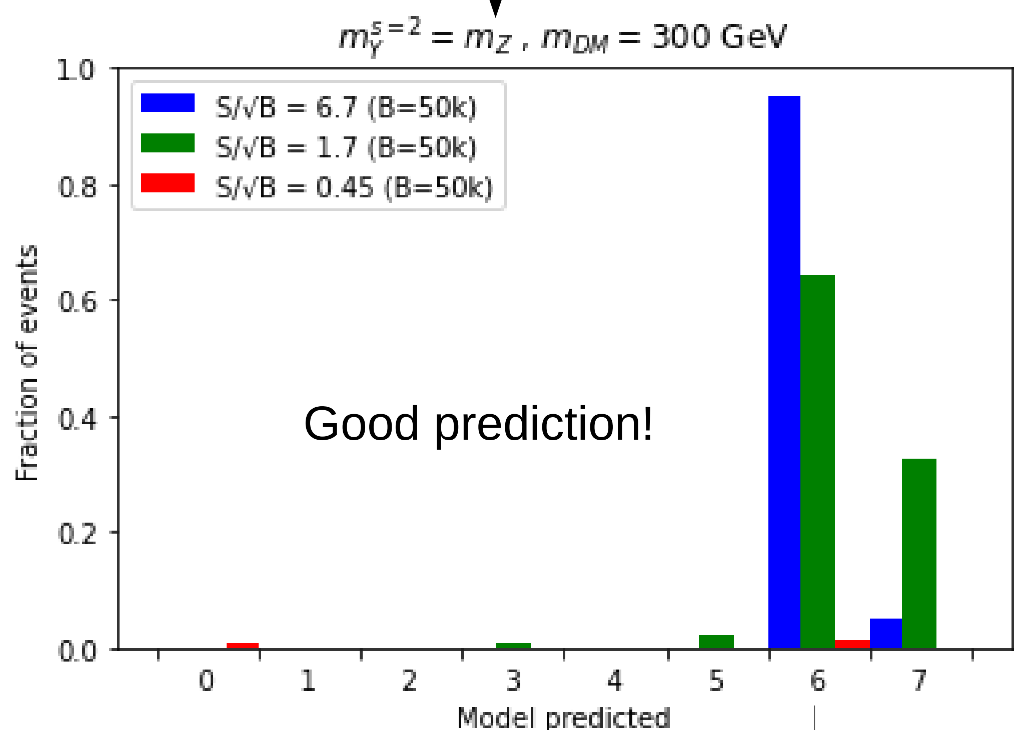
$m_Y^{s=2}$=1000 GeV , $m_{DM}$=10 GeV

# Multiclass classifier

## Testing with non-training models

**Input models completely new to the DNN**




$m_Y^{s=1} = m_Z$ , $m_{DM}$=300 GeV

$m_Y^{s=2}$=1000 GeV , $m_{DM}$=10 GeV

**The DNN classifies "kinematic distributions" not "models"**

**Predicts compatible kinetic distribution of the underlying model.**

# Conclusions

# Conclusions



- Monojet plus missing transverse energy channel of four simplified dark matter frameworks:
  **ALP and spin-0, spin-1, and spin-2 mediator models**

- One usual drawback of supervised techniques: the need of a specific data set per model
  → **we describe a family of models with a single data set**

## Neural Networks (individual classifiers):

- Discerning new physics signatures from SM background, two data representations:
  → event-by-event data → poor performance
  → **2D histograms** → **great performance**

- DNN performance independent of the number of background event with $S/\sqrt{B}$ as variable

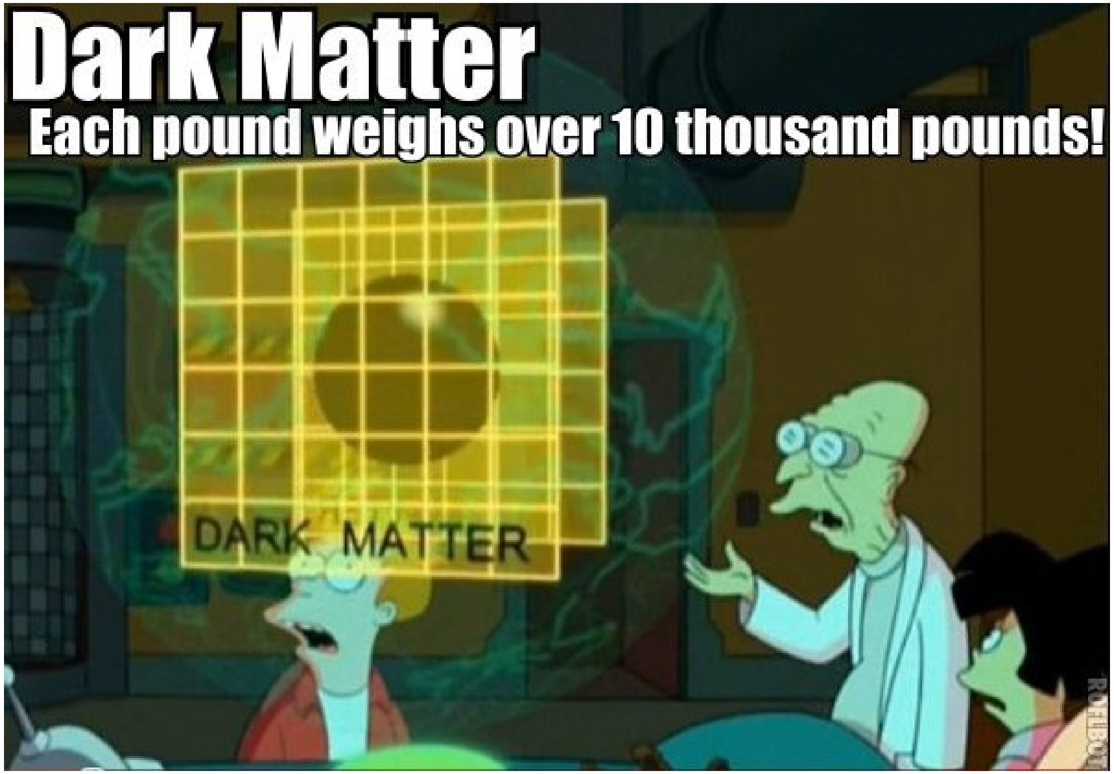
**Easy to check** if a DNN could discriminate a particular model from the SM, for any luminosity.
Or to **estimate the luminosity needed** to achieve a certain performance level.

## Multimodel classifiers:

- Supervised algorithms trained with several benchmark models per DNN.

  - a more challenging task, but a good performance is achieved.
  - result points towards a compatible kinetic distribution, a key tool to guide further analysis

Thank you!


Dark Matter
Each pound weighs over 10 thousand pounds!
DARK MATTER