# Domain Adaptation ML-INFN Hackathon

**Benedetta Camaiani, Piergiulio Lenzi, Lorenzo Viliani**

The basic idea

- Model agnostic measurements of Standard Model cross sections at the LHC.
- Simplified version of a study we recently published (arXiv:2207.09293).

- We want to reduce the model dependence in the signal extraction procedure of typical cross section measurements at the LHC, but keeping the best possible signal sensitivity.

- y : output of a DNN classifier (optimal S/B separation)
- Used as fit variable to measure the signal

- y must not be able to distinguish the physics model of signal events
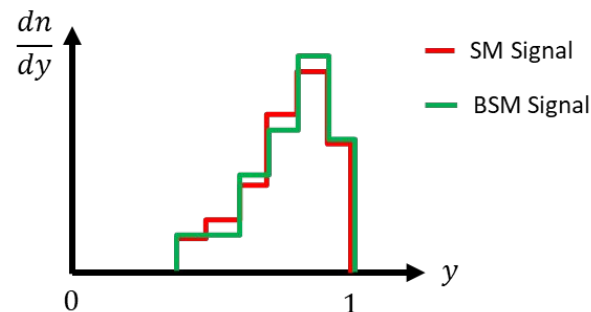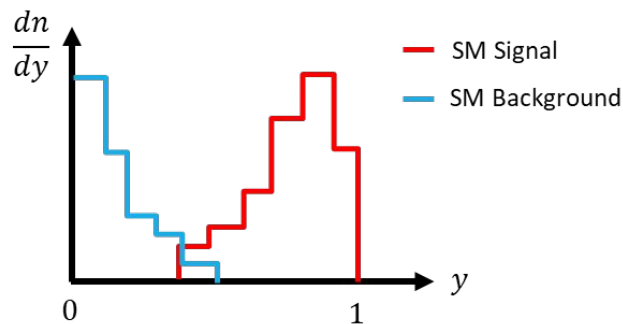
y does not introduce a bias in the fit result since the shape of its distribution is roughly the same regardless of the theoretical model describing the data
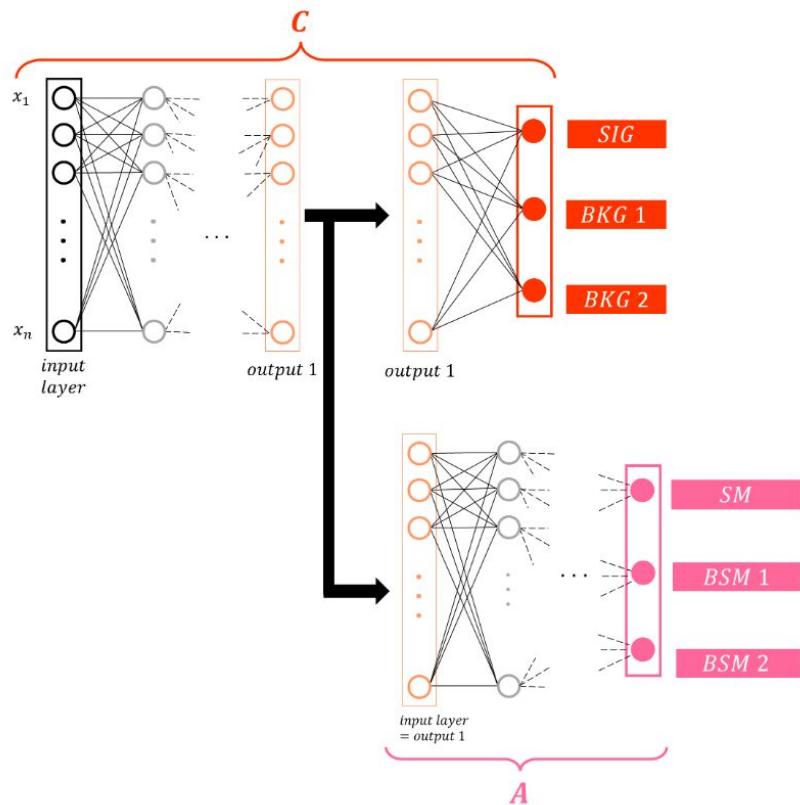
Classifier

- Takes as input the measurable kinematic variables of an event

- Aims to determine if the event is signal- or background-like

- Each output represents the probability that an event belongs to the corresponding class

- Is trained on data sample including events coming from different "domains", i.e. different signal models1

Adversary

- Trained only on signal (SM or BSM) events

- Guess the physics model of signal events, regressing the domain from the second-to-last layer of C

1. $Loss = Loss(C) - \alpha \cdot Loss(A)$

2. $Loss(A)$ *Two-step training procedure*

Dataset and Jupyter Notebook

- The case study is a cross section measurement of the Higgs boson produced via Vector Boson Fusion and decaying to WW→2l2v.

- Labelled simulated CMS data.
  - Generator-only information, no detector simulation!

- Contains the following classes of events in equal proportions.
  - Background events (SM processes such as ttbar, WW, etc.);   →   **BKG label**
  - Higgs boson produced via Gluon Fusion;   →   **GGH label**
  - Signal (SM and BSM).   →   **One label for SM signal + one for each of the 6 alternative BSM model**

- Includes a set of high-level input features (~26) mainly related to kinematics of leptons and jets.

- The Jupyter Notebook is currently running on a ML_INFN machine instantiated through INFN Cloud using hardware resources at CNAF.