Norfolk, Virginia, USA • May 8-12, 2023
CHEP 2023
Computing in High Energy & Nuclear Physics

# Deep Learning to improve Experimental Sensitivity and Generative Models for Monte Carlo simulations for searching for New Physics in LHC experiments

CHEP23 - International Computing in High Energy & Nuclear Physics Conference, 8-12 May 2023, Norfolk Waterside Marriot, USA

**J. Salt [1*], R. Balanzá[2] , A. García[4] J.A. Gómez[2], S. González de la Hoz[1], J. Lozano[3] ,R. Ruiz de Austri[1], M. Villaplana[1]**

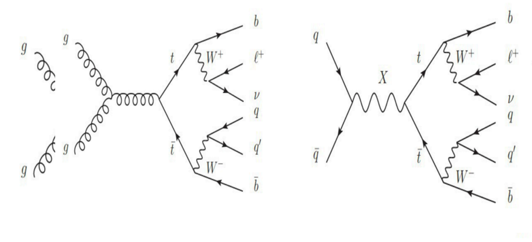
[1]Instituto de Física Corpuscular (IFIC), University of Valencia and CSIC, Valencia, Spain; [2]Pattern Recognition and Human Language Technology research center (PRHLT), Universitat Politècnica de València, Spain; [3] Departamento de Física y Matemáticas, Universidad de Alcala de Henares, Madrid, Spain; [4]Facultad de Física – Universitat de València, Valencia, Spain

*Corresponding author (jose.salt@ific.uv.es)

## Use of different ML methods in the classification of signal/background for searches for ttbar resonances

### Physics Problem: classification into ttbar resonances

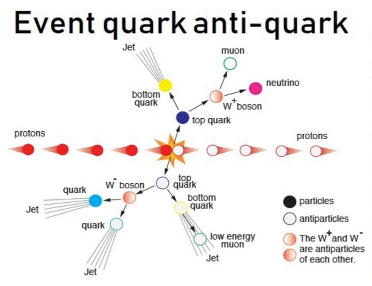
Our use case is the application of these methods to the search for a new particle X, of unknown mass, which is a resonance decaying into a top-antitop pair. The decay scheme of the top-antitop pair is the same for SM processes as for the resonance case. It is the kinematic variables and the invariant masses that can be calculated from them that have different types of distributions.



SM Process -Background

BSM Process Signal

Event quark anti-quark



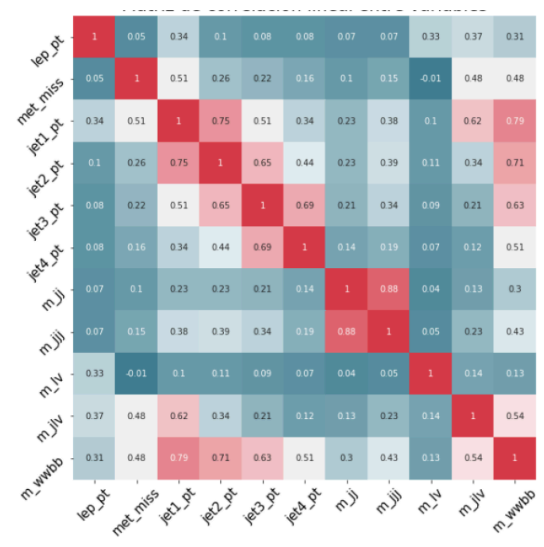
**Input data is Simulated Data by:**

- Generation by using Pythia & MADGRAPH
  - Background events datasets (SM events)
  - Different resonance masses:
    - 500GeV 1250GeV 750GeV 1500GeV 1000 GeV

| FEATURE | TIPO | DESCRIPCIÓN | PARTÍCULA |
|---|---|---|---|
| label | binaria | Variable output señal vs ruido | - |
| lep_pt | float | Momento transversal del leptón | Leptón |
| lep_eta | float | Pseudorapídity | |
| lep_phi | float | Azimut | |
| met_miss | float | Momento transversal faltante | Neutrino |
| met_phi | float | Azimut faltante | |
| jets_no | entero | Número de jets de la colisión | Jets resultantes |
| jet1_pt | float | Momento transversal | Jet más energético |
| jet1_eta | float | Pseudorapidity | |
| jet1_phi | float | Azimut | |
| jet1_btag | binaria | Correspondencia con un quark $b$ | |
| jet2_pt | float | Momento transversal | Segundo jet más energético |
| jet2_eta | float | Pseudorapidity | |
| jet2_phi | float | Azimut | |
| jet2_btag | binaria | Correspondencia con un quark $b$ | |
| jet3_pt | float | Momento transversal | Tercer jet más energético |
| jet3_eta | float | Pseudorapidity | |
| jet3_phi | float | Azimut | |
| jet3_btag | binaria | Correspondencia con un quark $b$ | |
| jet4_pt | float | Momento transversal | Cuarto jet más energético |
| jet4_eta | float | Pseudorapidity | |
| jet4_phi | float | Azimut | |
| jet4_btag | binaria | Correspondencia con un quark $b$ | |
| m_jj | float | Masa invariante | $W^-$ |
| m_jjj | float | Masa invariante | $\bar{t}$ |
| m_lv | float | Masa invariante | $W^+$ |
| m_jlv | float | Masa invariante | t |
| m_wwbb | float | Masa invariante | $X$ |
| mass | float | Masa de la partícula $X$ en la simulación (0 si es ruido) | $X$ |

- 10,000,000 events
  - No normalized and Normalized datsets
- Variables/features:
  - 5 high-level features
  - 21 low-level features

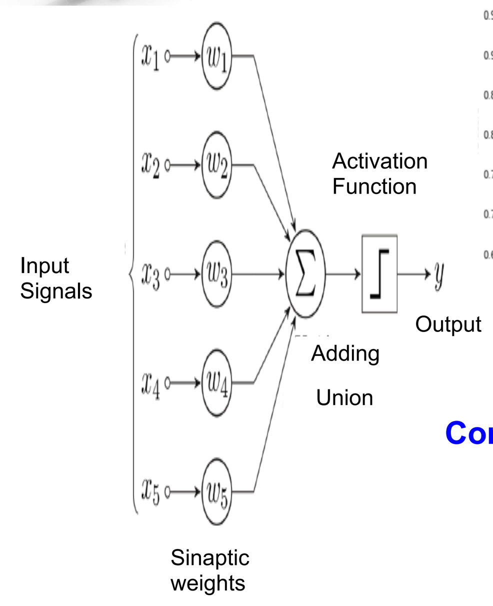
Available at: http://archive.ics.uci.edu/ml/datasets/HEPMASS

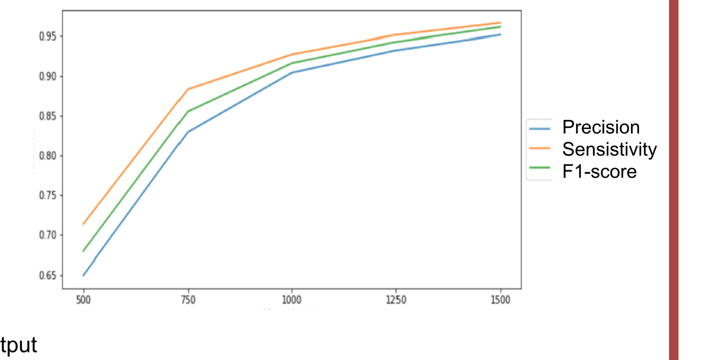


Correaltion between 11 variables, the Ones with better discrimination power

## MACHINE LEARNING METHODS

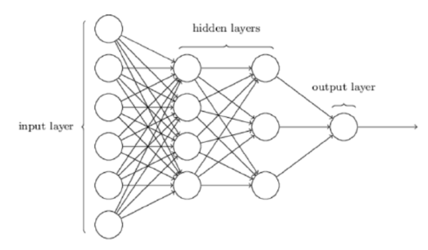
### Simple Neural Network




F1-score, Precision and Sensitivity of Simple NN for the 5 values of the signal masses

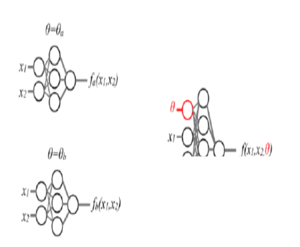
### Complex Neural Network



Scheme of a Complex NN with two hidden layers

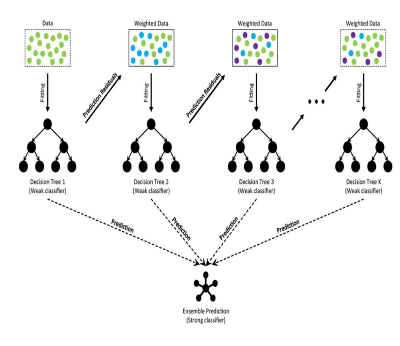
### Parameterized NN:

One of the possible solutions to this problem would be the use of parameterized models, which are based on including mass as one additional feature. In a real case, the idea would be to train the model with masses from the simulations. When making predictions on real data, a mass parameter would be added to them. Several tests could be done with masses suspected to be the mass of particle X, and they do not have to be the masses used for training. These possible masses could be to estimate for example from visualizations of the final invariant mass.

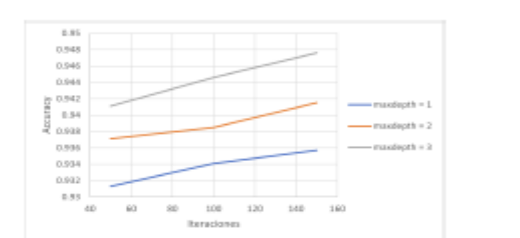


Structure of a Parameterized Neural Network

### Decision Trees:BDT




- *Boosted Decision Trees* allow the use of trees with little classifying power, such as trees with little complexity or composed of a single node (stumps) as good classifiers.
- *Adaptive Boosting (AdaBoost) algorithm has beeen used.*
- The training data set is assigned a weight uniformly to each event. Depending on whether the simple tree has been able to correctly classify an event or not, a new weight is reassigned.
- A different data set is made with the new weights, which is used to fit the next tree and so recursively

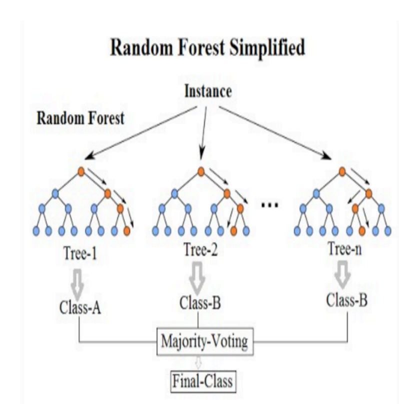



Accuracy vs number of iterations for 3 different depths (with resonance mass = 1500 GeV)

### Decision Tree Random Forest

- Random Forest can be used for classification problems, as in our case, or for regression.
- They are fast to train and make predictions, are easy to fit and easily estimate a general error of the model (Out-of-the-Bag Data)
- Starting from multiple random subdivisions of the dataset train, DTs are generated whose nodes collect a combination of variables that are different from each other due to the randomness of the data chosen to create them. The result of these trees are combined homogeneously, without taking into account any type of weight between the trees.

Random Forest Simplified




Hyperparameter optimization:

- number of variables/features: 16, 11, 5
- number of DT estimators: 500

### Comparison between different ML methods

| | Better RF | | | Better BDT | | | Better NN | | |
|---|---|---|---|---|---|---|---|---|---|
| *Mass (GeV)* | *Accuracy* | *Kappa* | *F1-Score* | *Accuracy* | *Kappa* | *F1-Score* | *Accuracy* | *Kappa* | *F1-Score* |
| 500 | 0.787 | 0.574 | 0.756 | 0.819 | 0.638 | 0.785 | 0.663 | 0.326 | 0.679 |
| 750 | 0.851 | 0.700 | 0.841 | 0.852 | 0.704 | 0.837 | 0.850 | 0.699 | 0.855 |
| 1000 | 0.895 | 0.790 | 0.891 | 0.889 | 0.779 | 0.882 | 0.914 | 0.829 | 0.915 |
| 1250 | 0.925 | 0.849 | 0.923 | 0.922 | 0.844 | 0.919 | 0.941 | 0.882 | 0.941 |
| 1500 | 0.946 | 0.892 | 0.946 | 0.944 | 0.888 | 0.942 | 0.958 | 0.916 | 0.958 |

- **BDT's** : better results with **3 levels** of depth and it's optimized with **300 iterations**
- **BDT** gives better results than **RF** at low masses but RF is a little bit better at high masses. Improvement of RF results with to 5 variables and 500 DT estimators
- **NNS vs BDT&RF**: NN simple gives worst results at low mass wrt BDT&RF but they give better results at high mass
- **Parameterized NN don't get a significant improvement** with respect NNsimple/ complex: they interpolate well except for low masses

### Comparison Rstudio - Python

- The same task is 7-8 times faster in python than in Rstudio
- Comparing RF with 500 trees with BDT ADA 300 iterations : about 2 times faster RF 500 Trees
- RStudio is more user friendly than python and is used for first steps in ML

## Use of Deep Learning Generative Models for production of Simulated Data in ATLAS

To simulate proton-proton LHC collisions:
In the standard way implies (1) generation , (2) hadronization/fragmentation (3) to pass the particles through the detectors (detector simulation)
To produce billions of events -> Time consuming and expensive
On top of that, when systematic errors have to be evaluated, more MC production is needed

We can try to use another methodology:
To generate SM background events and new physics scenario and to process the data in a easy-format ( sequence of 4 –vectors)
Run ML Methods: VAE's, FAN's and NF
With computational time savings
To define a metrics to compare the performance
To define a Good estimation of systematic errors

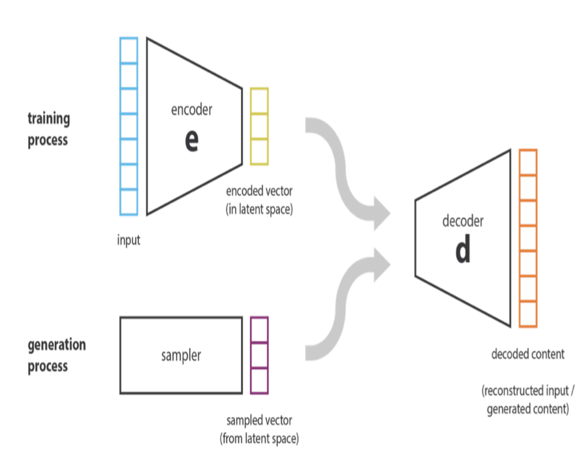
Datasets used in this work have been taken from a uptodate repository; the ones generated by DarkMachines community. LHCsimulationProject, Feb 2020, doi:10.5281/zenodo.3685861. Available at: https://zenodo.org/record/3685861.

Reason to study these datasets:
- more processes in the repository which can be studied subsequently
- extended information per event (leptons, photons,…)
- well documented and access to the authors

CSV file one line per event

List of variables: event id, process ID, event weight , MET, METphi, obje1, E1, pt1, eta1, phi1, obj2, E2, pt2, eta2, phi2, …
Process ID: ttbar / New Physics: stop_2
objects: b-jets, leptons, jets, photons,; in each type they are ordered in descending order according its p_t the latent space is composed by a vector containing the mean and the standard deviation corresponding to the different distributions of the features /variables of each event

**AUTOENCODERS:** An architecture of artificial neural networks composed of two parts: encoder and decoder, which is trained as a whole in order to reproduce the input at the output while learning an intermediate coded representation. And then generate the input data as close as possible to the input from the learned coded representation.




Scheme of the internal structure of a Variational Autoencoder

### Loss Function

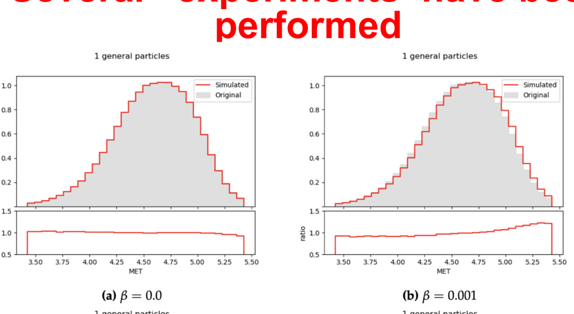
$$L_{VAE} = (1-\beta)MSE + \beta KL$$

MSE: Mean Squared Error. Reconstruction term on the Final layer, which tends to improve The performance of the encoding-decoding schema

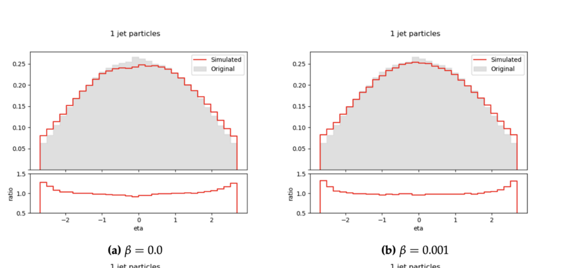
a regularisation term on the latent layer, that is proportional to the Kullback-Leibler (KL) divergence and tends to regularise the organisation of the latent space by making the distributions returned by the encoder close to a standard normal distribution with zero mean and unit variance

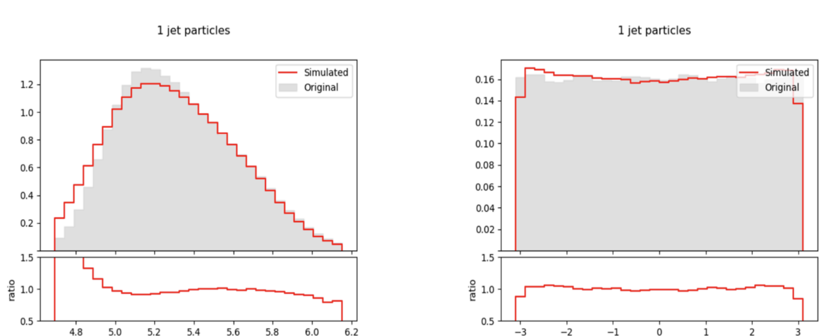
To avoid Overfitting

### Several 'experiments' have been performed



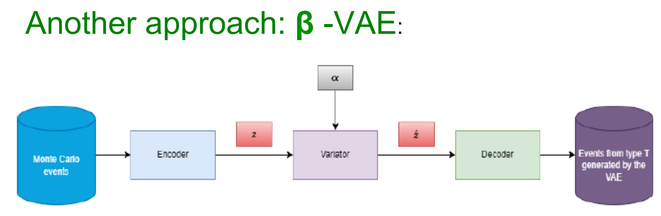

Application to a data sample of SM ttbar events for variables:
MET (missing E_t) and η (psudorapidy)



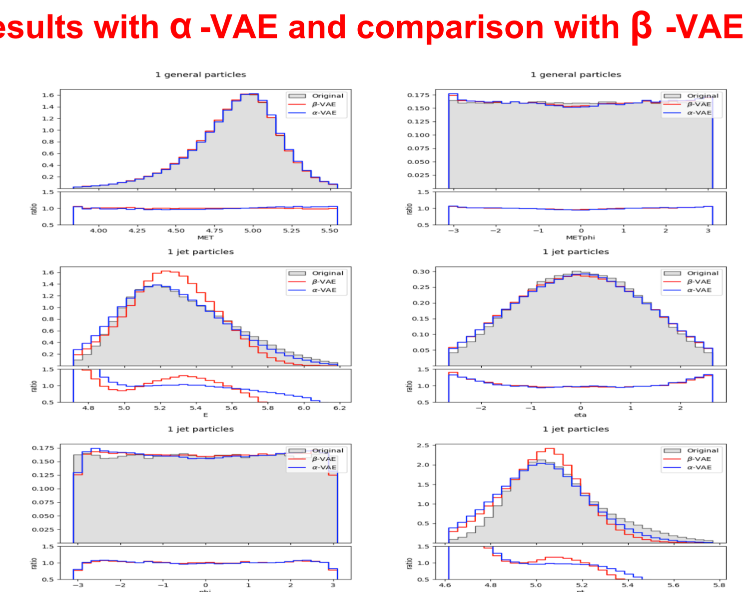

Histograms of first jet parameter E and Φ with β =0.001

Another approach: β -VAE:



Flow diagram for generating events with β- VAE approach

### Results with α -VAE and comparison with β -VAE




Histograms of fist jet of particles comparing the best model of β -VAE and α-VAE with stop_02 events ( β =0.001 and α =0.2 )

### Next steps:

Continuation of the studies with Generative Models in the context of IFIC-UPV collaboration:

- Improvements in the quality of the simulated events generated using VAE
- Further cross-checks to validate the samples
- To study the performance using Normalizing Flows (NF) and comparison VAE-NF

To apply these GGMM to another physics processes

Possible limits in the use of these additional simulated data
- At the statistical level
- In the phase of evaluation of systematic errors

## Conclusions:

### Part 1

1.-A complete study with different ML methods applied to simulated datasets ; Decision Trees: BDT and Random Forest ; Neural Networks: Simple , Complex and Parameterized NN
2.- NN give better classification performance than Decision Trees,l except in the case of low masses of ttbar resonances
3.- Rstudio and Python comparison: Python is faster but RStudio provides a more didactic framework
4.- Out Of Bag error estimate: using RF one can have access to the error estimates of the accuracies

### Part 2

1.- Study of different VAE for creating large amounts of analysis-specific simulated LHC events with limited computing cost
2.- Method β-VAE yields initially promising results
3.- Method α-VAE: : by adding a variator one can obtain a better agreement between the original dataset and the Generative Models simulated dataset
4. Further studies will be focused in the control of the metrics and the study of other Generative Models

IFIC INSTITUT DE FÍSICA CORPUSCULAR · VNIVERSITAT DE VALÈNCIA · CSIC · UNIVERSITAT POLITÈCNICA DE VALÈNCIA · GENERALITAT VALENCIANA Conselleria d'Educació, Cultura i Esport · GenT · MINISTERIO DE CIENCIA E INNOVACIÓN

**Funded by MICINN and Generalitat Valenciana under contracts PID2019-104301RB-C21 and CIDEGENT/2019/027**