Uncertainty estimation in deep learning based-classifiers of High Energy Physics events using Monte Carlo Dropout.

Raquel Pezoa[1,2], Sebastián Bórquez[3], William Brooks[2,4], Luis Salinas[2,3], and Claudio Torres[2,3] (1) Escuela de Ingeniería Informática, Facultad de Ingeniería, Universidad de Valparaíso, Chile. (2) Centro Científico Tecnológico de Valparaíso. (3) Departamento de Informática, Universidad Técnica Federico Santa María. (4) Departamento de Física, Universidad Técnica Federico Santa María. raquel.pezoa@uv.cl, sebastian.borquez@sansano.usm.cl, william.brooks@usm.cl, claudio.torres@usm.cl

Universidad de Valparaíso CHILE

ACAT 2022 BARI

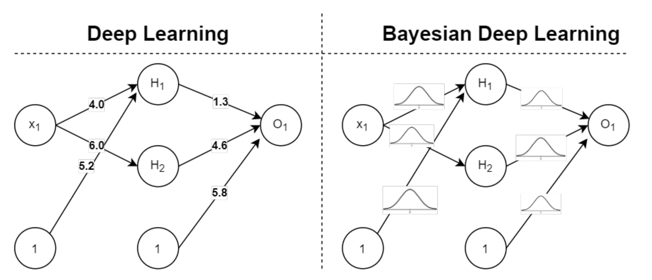
CCTVal CENTRO CIENTÍFICO TECNOLÓGICO DE VALPARAÍSO

UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

1. Introduction

- Classifying HEP events, or separating *signal* events from the *background*, is an important analysis task, in the search for new phenomena.
- Complex deep learning-based models have been fundamental for achieving accurate performance.
- However, the **uncertainty estimation** has been traditionally neglected when deep learning-based methods are used, despite its critical importance in scientific applications.

2. BDL and Uncertainty Estimation

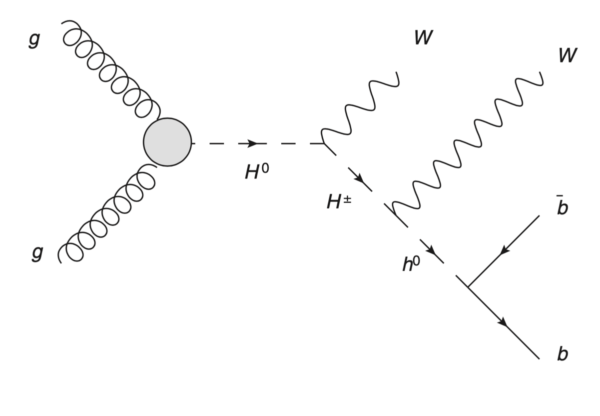


- Bayesian deep learning (BDL)-based methods allow measuring uncertainty when classification is performed using deep learning architectures.
- **BDL** models usually estimate uncertainty by either **placing probability distributions over model weights**, or by learning a direct mapping to probabilistic outputs.
- This work is focused on the use of the **Monte Carlo Dropout (MC-Dropout) method**, a BDL technique proposed in [1] that is based on Dropout [2].

3. Event Classification

We classify the events from two datasets:

- **Higgs dataset**[a] for identifying the **signal**: $gg \to H^0 \to W^{\mp}H^{\pm} \to W^{\mp}W^{\pm}h^0 \to W^{\mp}W^{\pm}b\bar{b}$ from the background.



Each event has with 21 low-level features (lepton_pT, jet1pT, ...), and 7 high level features (m_bb, m_wbb, ...).

- Hadronization of the ω **meson production off nuclear targets**[b].

$$\omega \to \pi^+\pi^-\gamma\gamma$$

[a] Baldi et al. Searching for Exotic Particles in High-energy Physics with Deep Learning. Nature Communications, 2014.

[b] A. Bórquez, Master's thesis, UTFSM, Valparaíso, Chile, 2021.

4. Monte Carlo Dropout

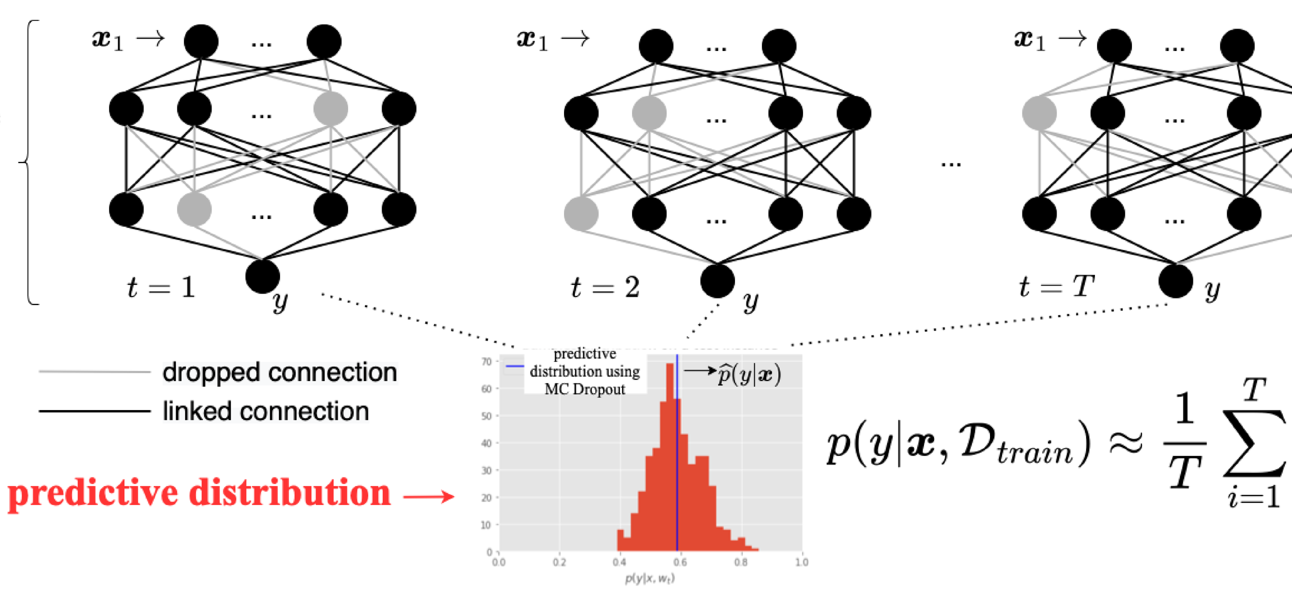
- Let $\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{\omega})$ a DNN model with parameters $\boldsymbol{w}$, the training set $\mathcal{D}_{\text{train}} := \{\boldsymbol{X}, \boldsymbol{Y}\}$, $\boldsymbol{X} := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and $\boldsymbol{Y} := \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ are the inputs and outputs, respectively. Bayesian models allow predictions on a new input point $\boldsymbol{x}^*$, predicting $\boldsymbol{y}^* = \boldsymbol{F}(\boldsymbol{x}^*, \boldsymbol{\omega})$ given the learned weights $\boldsymbol{\omega}$.

$$\text{The predictive distribution is given by } p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{X}, \boldsymbol{Y}) = \int p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\omega}) \underbrace{p(\boldsymbol{\omega}|\boldsymbol{X}, \boldsymbol{Y})}_{\text{posterior dist.}} d\boldsymbol{\omega} \quad (1)$$

- $p(\boldsymbol{\omega}|\boldsymbol{X}, \boldsymbol{Y})$ in Eq. 1 is usually intractable, and **Monte Carlo Dropout** [1] allows us to approximate it.

$$\boldsymbol{x}_1 \in \mathbb{R}^n, \boldsymbol{x}_1 = [lepton_{pt}, \ldots, m_{jlv}, m_{bb}], y \in [0, 1]$$

$\boldsymbol{F}(\boldsymbol{x}, \widehat{\boldsymbol{w}})$ with parameters $\widehat{\boldsymbol{w}}$ is used to approximate the predictive distribution, obtaining the predictive mean: $\widehat{p}(y|\boldsymbol{x}, \mathcal{D}_{train})$



predictive distribution $\longrightarrow$

$$p(y|\boldsymbol{x}, \mathcal{D}_{train}) \approx \frac{1}{T}\sum_{i=1}^{T} p(y|\boldsymbol{x}, \widehat{\boldsymbol{\omega}}_t) \approx \widehat{p}(y|\boldsymbol{x}, \mathcal{D}_{train}),$$

$$\widehat{\boldsymbol{\omega}}_t \sim q(\boldsymbol{\omega})$$

MC Dropout includes **dropout layers** after each dense layer, and **keep the dropout layers active during the training and inferences stages.**

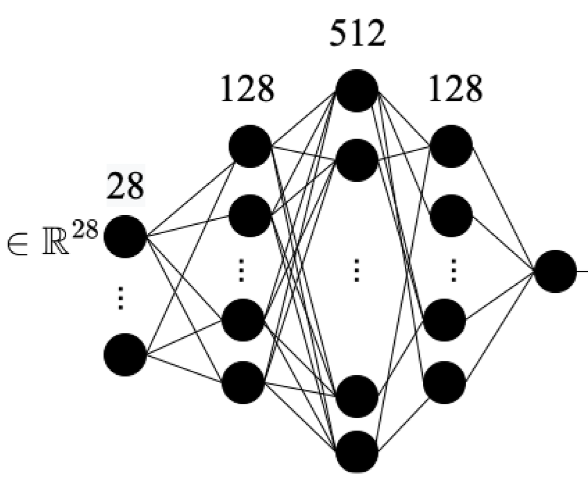
More info here

5. Experiments and Results

We used **`autokeras`** to select the DNN architecture using the **Higgs** dataset and the following hyperparameters values:

Hyperparameter	Description	Values
num_layers	Number of layers	3, 4, 5, 6
num_units	Number of neurons per layer	32, 64, 128, 256, 512
batch_size	Batch size	8
Activation	Activation function	ReLu and sigmoid (output)
Loss	Loss function	binary cross-entropy

autokeras $\boldsymbol{x} \in \mathbb{R}^{28}$

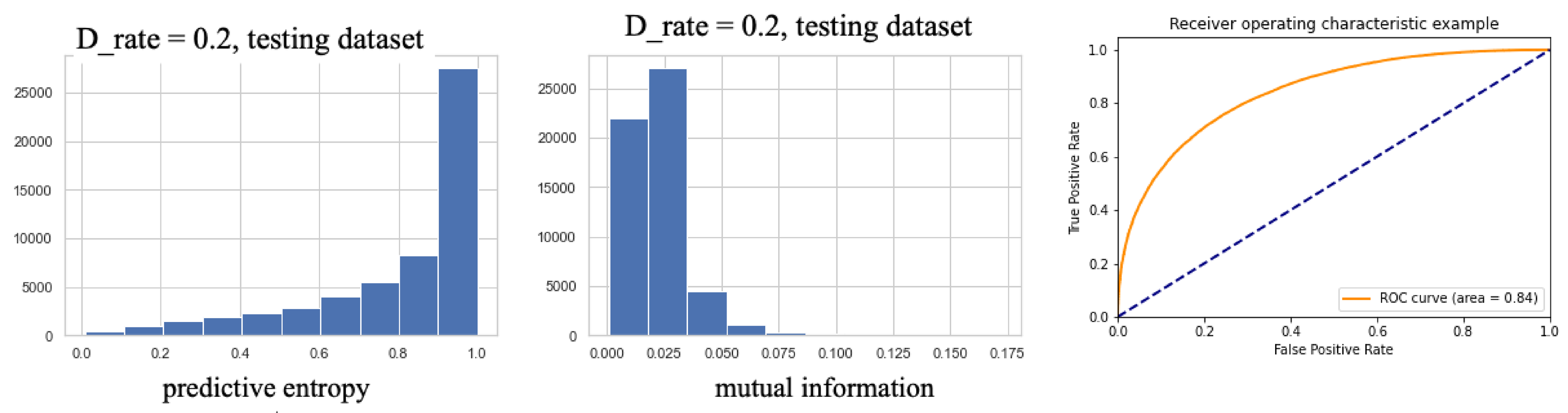


$y \in [0, 1]$

Using **MC Dropout**, we measure epistemic uncertainties: **predictive entropy** and **mutual information.**

Higgs dataset

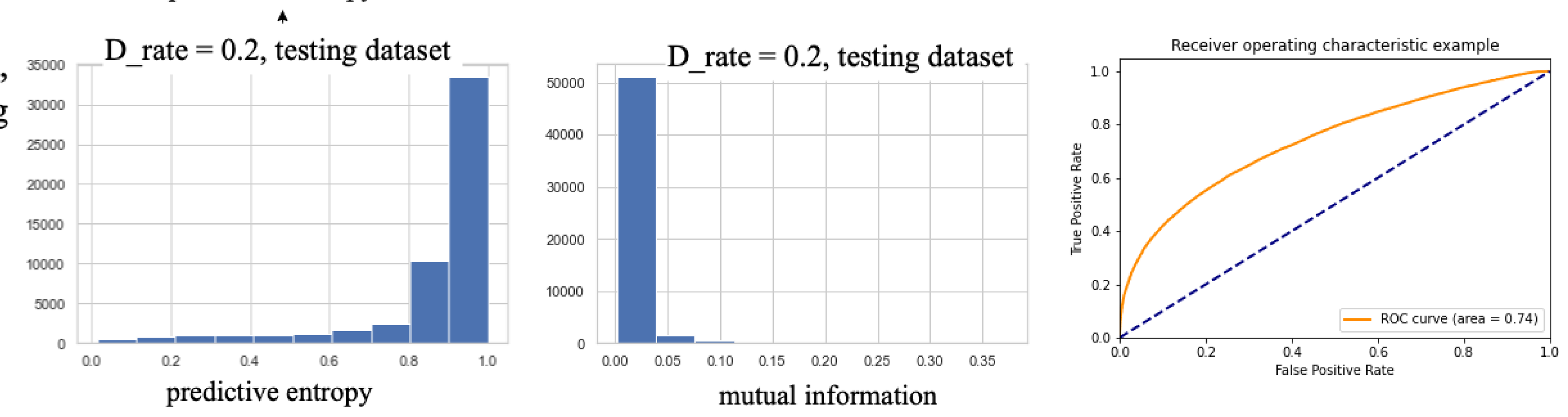
D_{rate}	Acc	Prec	Rec	F1	AUC
0.0	0.77	0.79	0.77	0.78	0.85
0.2	0.76	0.77	0.78	0.77	0.84
0.5	0.74	0.76	0.74	0.75	0.82
0.9	0.53	0.53	1.00	0.69	0.50



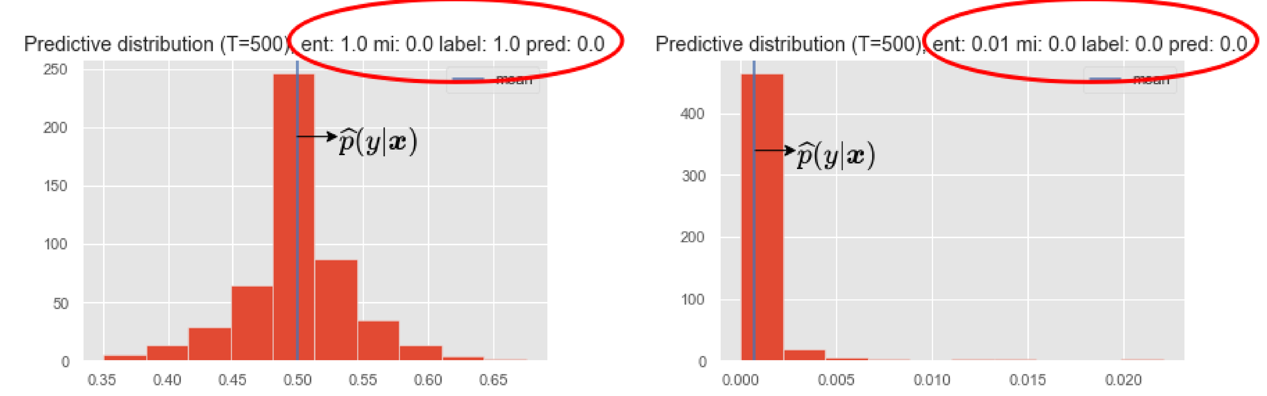
ω meson dataset

With with four hidden layers: [128, 256, 256, 128], using undersampling to balance the data.

D_{rate}	Acc	Prec	Rec	F1	AUC
0.0	0.68	0.70	0.61	0.65	0.74
0.2	0.68	0.70	0.61	0.65	0.74
0.5	0.66	0.70	0.56	0.62	0.71
0.9	0.63	0.69	0.45	0.55	0.68



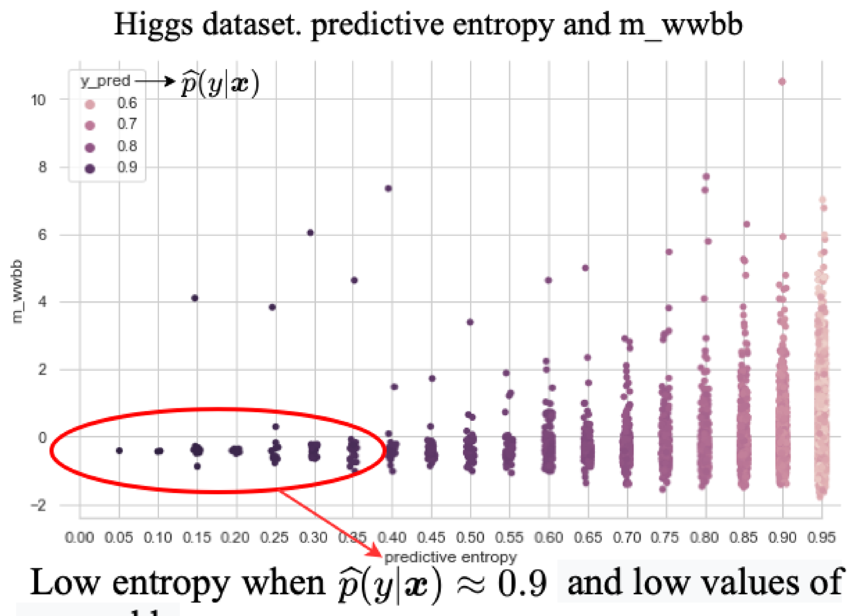
Predictive distribution example



Particular event of Higgs dataset, with high entropy certain prediction

Particular event of Higgs dataset, with low entropy uncertain prediction

Higgs dataset. predictive entropy and m_wwbb



Low entropy when $\widehat{p}(y|\boldsymbol{x}) \approx 0.9$ and low values of m_wwbb.

6. Conclusions and Future Work

- Preliminary results showed best performance using MC Dropout $D_{rate} = 0.2$, but we still need to improve classification performance. • High predictive entropy $\to \widehat{p}(y|\boldsymbol{x}) \approx 0.5$, and low mutual information $\to$ model gives similar probabilities in multiple forward passes. • Future tasks: to include the uncertainty estimations in the training stage to improve performance and to combine BDL with eXplainable Artificial Intelligence techniques, like SHAP.

GitHub: `https://github.com/rpezoa/MCDropout_HEP_classif`

7. References

[1] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[2] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[3] https://shap.readthedocs.io/.

Acknowledgments: This research was supported by FONDECYT Postdoc Project N. 3190740 and ANID PIA/APOYO AFB180002.