Predicting the performance of a GNN for Particle-Flow reconstruction using (Quantum-)SVR in CoE RAISE

Student: ***Juan Pablo García Amboage***
Supervisor: ***Eric Wulff, Maria Girone***

Openlab Summer Student Program 2022

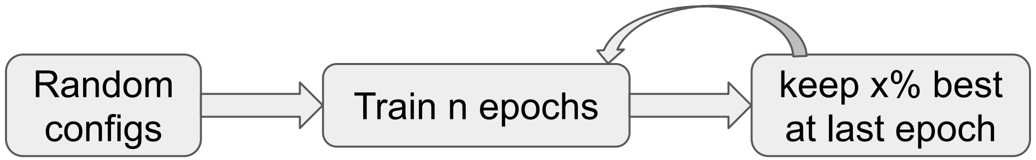
Introduction and some definitions

- ***Machine-Learned Particle-Flow (MLPF) [1]***: Graph Neural Network (GNN) aimed at performing **efficient particle flow reconstruction**.

- ***Hyperparameters (HP)***: **Non trainable parameters**. Define how the training process is carried out as well as the model architecture.

- ***Hyperparameter optimization (HPO)***: Automatic **search for the best** HP **configuration**. Very computationally expensive.

💡 *IDEA*: ***Early Stopping:*** **Terminate less promising configurations based on their relative performance at a specific epoch** in order to allocate more resources to the promising ones. Problem: learning curves may intersect.



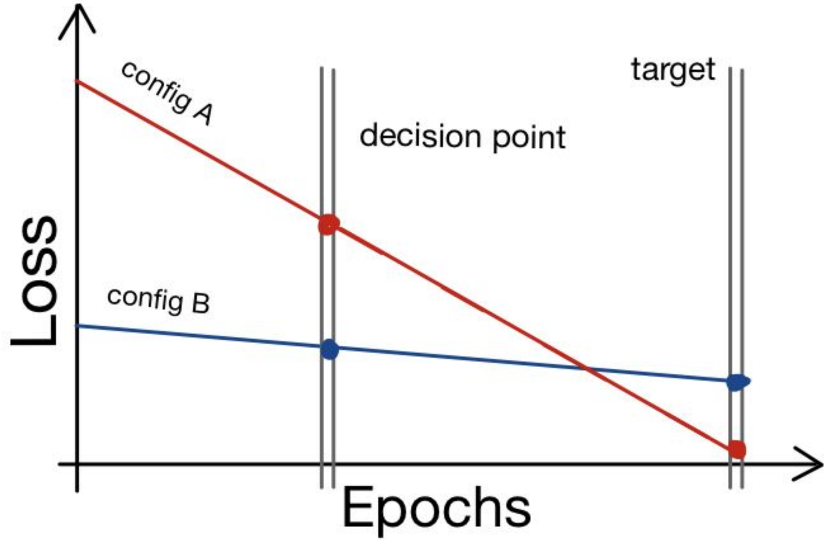
[1] J. Pata et al. Eur. Phys. J. C (2021) 81: 381

Performance prediction for HPO

Experts can **detect less promising configurations** when **looking** at their **partial learning curves** and terminate them early affording resources.

Use a classical (fast to train) **regressor to predict the final performance of the NN** given its configuration and information from its partial learning curve.

Inspired by:
Baker et al: Accelerating Neural Architecture Search using Performance Prediction.



Feature vector

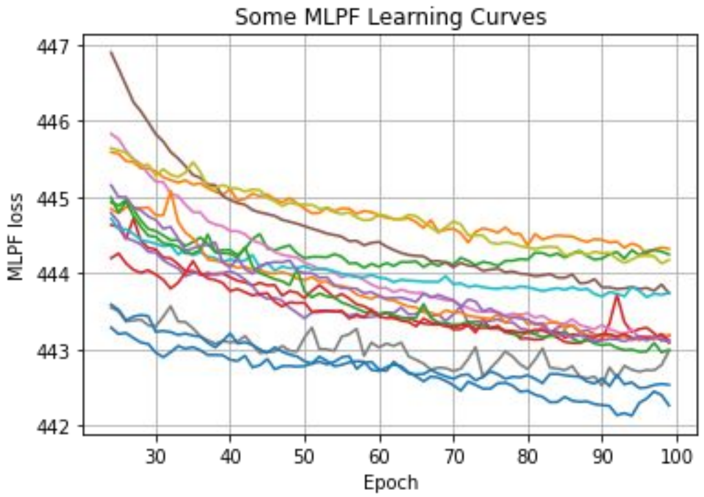
Configuration	Partial Learning Curve	1^{st} order diffs	2^{nd} order diffs

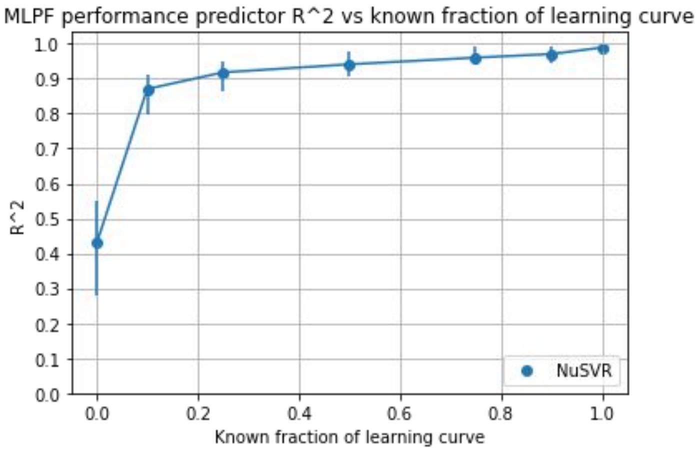
Target

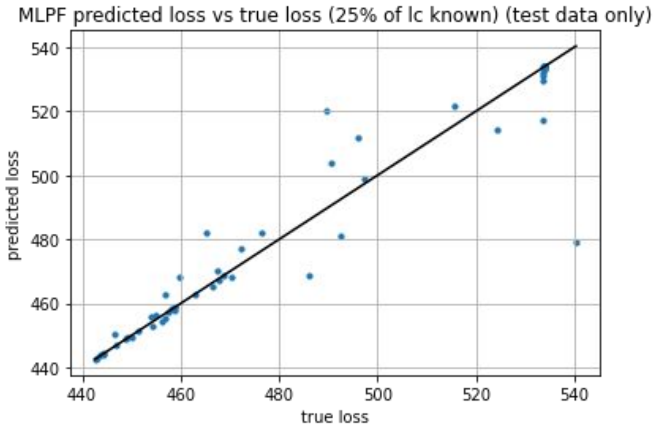
Final loss

Classical Performance Predictor

- **Generate a training set** to train the performance predictor.
  - Train MLPF for **100 epochs** with random configurations varying **7 HPs** on the publicly available Delphes dataset (https://doi.org/10.5281/zenodo.4559324)
- Train the predictor: classical **NuSVR**.
- **Evaluate** how good the predictor is **using R^2 score.**
  - Measure of how much of the variation on the target value is explained by the model. Perfect if $R^2=1$.







Accessing D-Wave Quantum Annealer in CoE RAISE

- **Quantum Annealer:** quantum computer that can solve **specific optimization problems**.
  - Quadratic **U**nconstrained **B**inary **O**ptimization

$$\text{Minimize:} \quad f(x) = \sum_{i<j}^{N} Q_{i,j} x_i x_j + \sum_{i}^{N} Q_{i,i} x_i$$

$x_i \in \{0,1\}$ and Q is a $N \times N$ symmetric matrix

- Represent the SVR with **QUBO** formulation.

- **Annelar returns multiple solutions.**
  - Combined to **reconstruct** the **trained QSVR**.

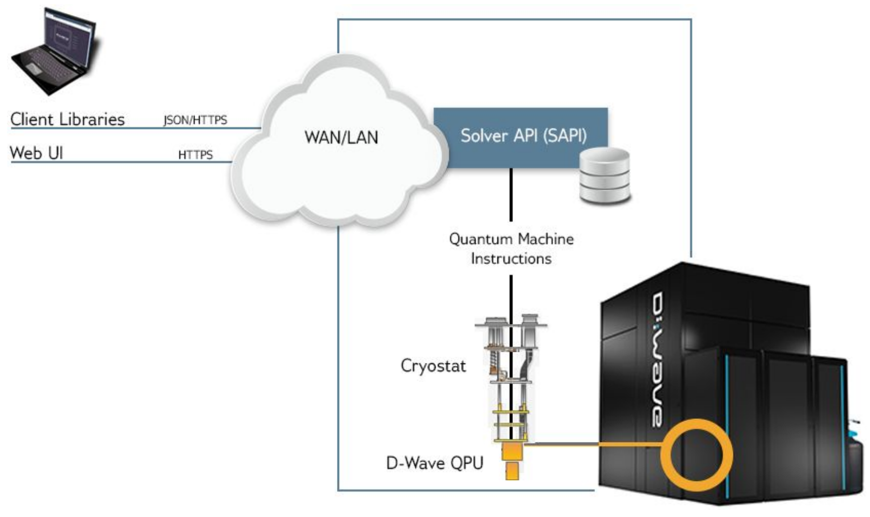


Image from D-Wave documentation

PROBLEMS:

- **Only ~20 points** can be used for **training.**
- **Unstable results**.
- **Limited** quantum computing **time** available

Quantum-SVR

PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

Quantum-SVR

PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

- **Use classical SVR and Simulated Annealing to study QSVR HPs**

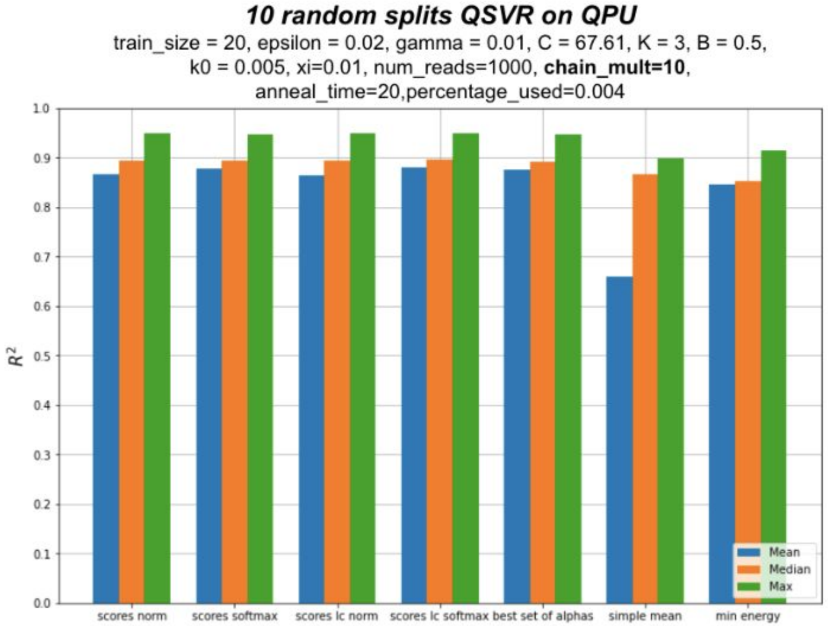
Quantum-SVR

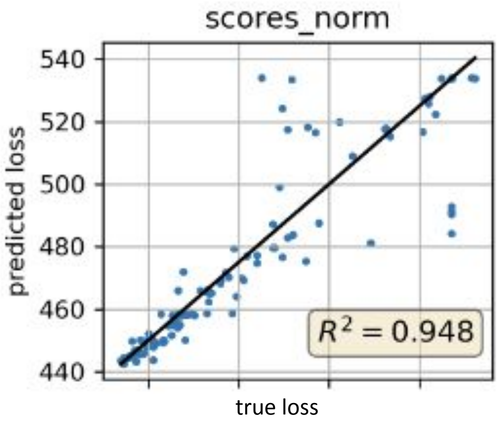
PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

- **Use classical SVR and Simulated Annealing to study QSVR HPs**





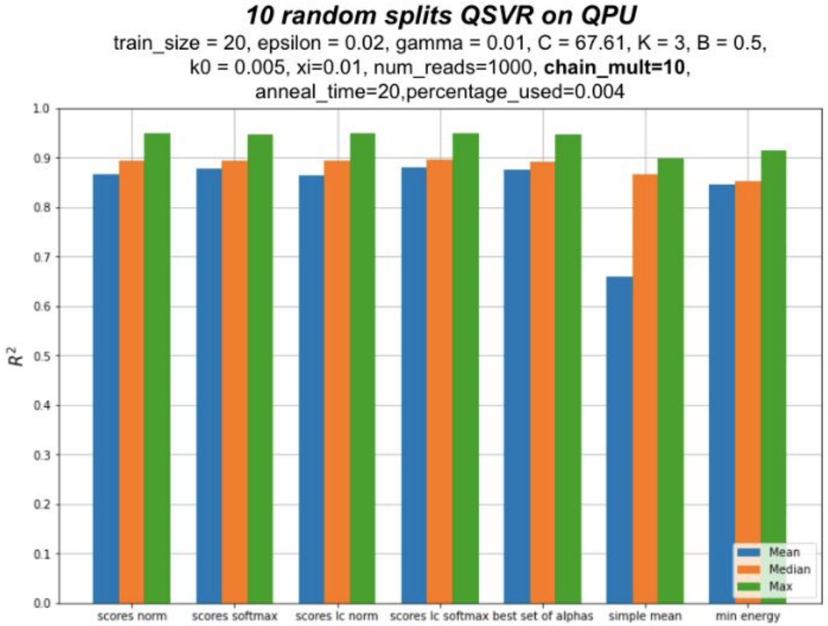
Quantum-SVR

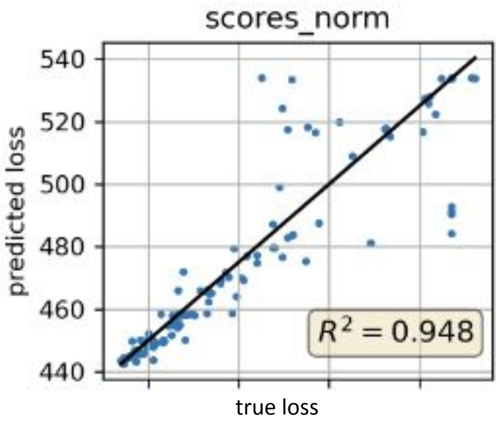
PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

- **Use classical SVR and Simulated Annealing to study QSVR HPs**
- Split larger training set and **combine predictions of multiple QSVRs**





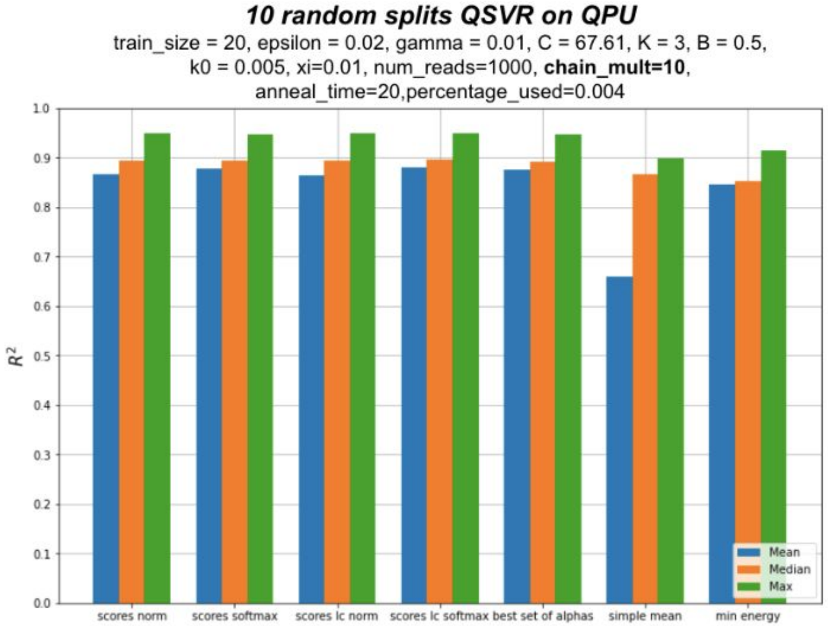
Quantum-SVR

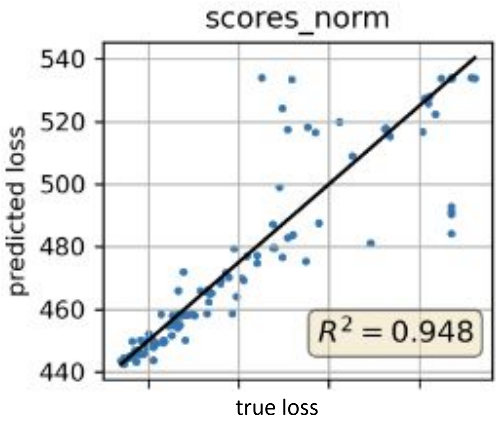
PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

- **Use classical SVR and Simulated Annealing to study QSVR HPs**
- Split larger training set and **combine predictions of multiple QSVRs**





R^2 Summary	
Maximum	0.927
Minimum	0.857
Mean	0.899
Median	0.897
Standard Deviation	0.019

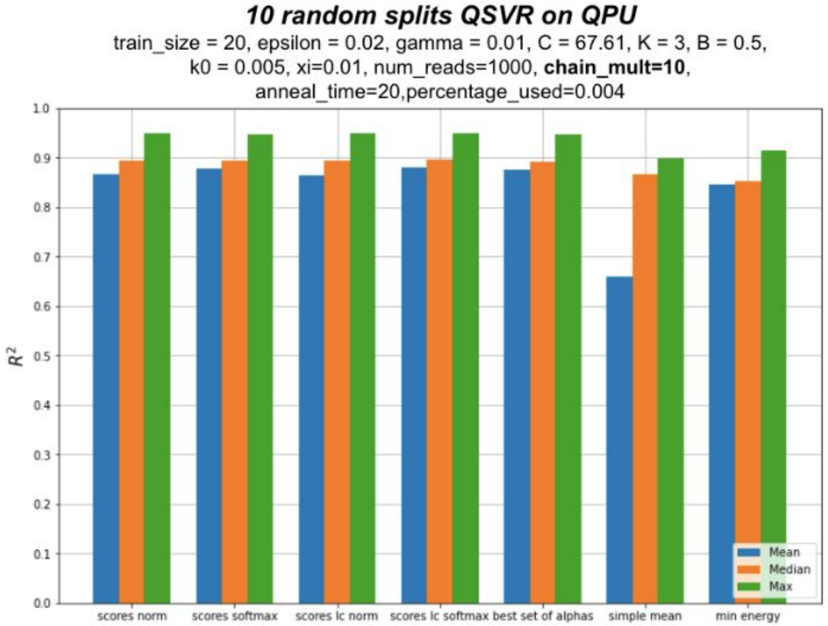
Quantum-SVR

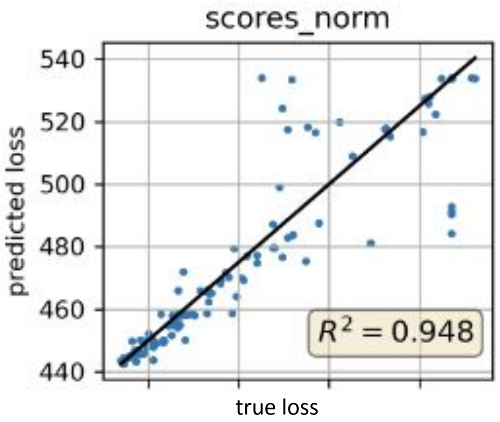
PROBLEMS:

- **Unstable results**.
- **Limited** quantum computing **time** available.
- **Only ~20 points** can be used for **training.**

SOLUTIONS:

- **Use classical SVR and Simulated Annealing to study QSVR HPs**
- Split larger training set and **combine predictions of multiple QSVRs**





Increases Stability ✅

R^2 Summary	
Maximum	0.927
Minimum	0.857
Mean	0.899
Median	0.897
Standard Deviation	0.019

Conclusions

- **Performance Predictors** have strong **potential to accelerate** hyperparameter **optimization**.

- Despite current limitations, it was possible to **achieve QSVR performance comparable to classical SVR**.

- **Quantum technologies** can potentially **have a place in the** classical **machine learning** design and tuning **pipeline**.

- **Further work in CoE RAISE**:
    - Study other ways to translate SVR to and from the quantum machine.
    - Study other QSVR **combination techniques.**
    - Make **use** of **the predictor** for hyperparameter optimization.

QUESTIONS?

juanpablo.garcia@rai.usc.es