REVIEW OF QUANTUM SUPPORT VECTOR MACHINE APPIICATION IN IHEP

Hirmans Tabaharizato

May 8, 2023

Contents

ML Tools

- The process of discovering the right setting of knobs coerching (the result from training model).
- Train the model with data:
  (1) start off with a randomly initialized that cannot do anything useful,
  (2) Grab same off the data,
  (3) Tweak the knobs to make the model perform better,
  (4) Repeat steps 2 and 3 until model is awesome.

Training process Algorithms TPA

- It can be represented as follows:

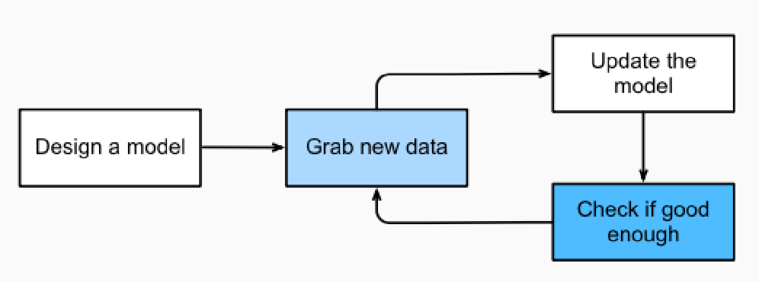

Figure: A typically training process

Key properties of datasets

- Work with data typically needs to come up with a **suitable numerical representation.**
- dataset instance ⇔ **Collection of Examples** $C.e$:

$$C.e := \{\underbrace{e_1, e_2, e_3, e_4, ..., e_{n-2}, e_{n-1} e_n}_{\text{data point, data instance, sample}}\}(n \in \mathbb{N}), \tag{1}$$

- each data point consist of set of attributes, called features (covariates or inputs).
  - based on each model must make its PREDICTIONS.

SL Problem

- GOAL $\Rightarrow$ predict the value of a special attribute, label (or target) (not model's input part),
  - Example the **Case of IMAGE DATA**:
    1. Feature $\Leftrightarrow$ an individual photograph,
    2. label $\Leftrightarrow$ a **number** that represents the category to which photograph belongs.

    - **Photograph** as three grids of numerical value $\Leftrightarrow$ brightness of red, green, and blue light at each pixel.

Support Vector Machine (SVM)

- A supervised machine learning algorithm that is widely used in classification problems.
  - SVM Algorthm based on concept of finding the optimal hyperplane that can split different classes in the feature space.
- Suppose we have a training dataset of n points

$$(\vec{x}_i, y_i) \dots (\vec{x}_n, y_n)$$

if these data points are not linearly separable

$$k_{ij}(\vec{x}_i, \vec{x}_j) = \langle f(\vec{x}_i), f(\vec{x}_j) \rangle$$

There are different form for the $f(\vec{x})$ function

- Radial basis function $f(\vec{x}_i) = e^{-\frac{\vec{x}_i^2}{2\sigma^2}}$
- Polynomial $f(\vec{x}_i) = \left(\gamma \cdot \vec{x}_i^T + r\right)^d ; \gamma > 0$
- Sigmoid $f(\vec{x}_i) = \tanh\left(\gamma \cdot \vec{x}_i^T + r\right)$

SVM

- an essential path to learn of how to classify data into two different sets.

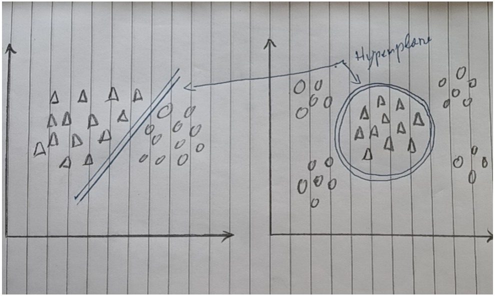

Figure: Data location separated by Hyperplane

Future Map

- Map the location data point from original data into a higher dimension "K", can allow to compute the **optimal hyperplane** (in higher dimensional space) that can separate the two classes.
  - Involve computing the the distance between data points within higher dimensional space.
- In case where "K" very large, determination of these distances is very difficult and computationally expensive.
  - Kernel. Collection of inner products of each pair of data points in the new feature map

$$Ker := \{K_1, K_2, ..., K_n\}$$

Quanturm Support Vector Machine

Quantum Kernel estimation

- In a quantum kernel, a classical feature $\vec{x}$ is mapped to higher dimension Hilbert space like $|\phi(\vec{x})\rangle \langle\phi(\vec{x})|$ in such a way that:

$$k_{ij}(\vec{x_i}, \vec{x_j}) = |\langle\phi(\vec{x_i})\,|\phi(\vec{x_j})\rangle|^2 \tag{2}$$

where,

$$|\phi(\vec{x_i})\rangle = \mathcal{U}_{\phi(\vec{x_i})}\left|0^{\otimes N}\right\rangle$$

- The mathematical implication of the kernel entry is the distance between the two data points in the high-dimensional quantum state space.

Application of QSVM Algorithm in High Energy Physics

- Running SVM in classical computers and a simulated quantum computer using a state vector simulator from IBM and data samples from the Central Electron Positron Collider (CEPC).

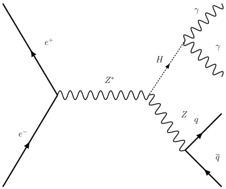

Figure: Representation of Feynman diagram for the $e^+e^- \rightarrow ZH \rightarrow q\bar{q}\gamma\gamma$

- $e^+e^- \rightarrow ZH$ signal and its related backgrounds are utilised for the study where the $H \rightarrow \gamma\gamma$ and $Z \rightarrow q\bar{q}$.

Quantum feature map (4 qubits)

```
Using the following variables:
 ['minDeltaR_y_j', 'DeltaPhi_yy', 'DeltaP_yy_jj', 'p_yy']
number of signal: 1500 number of background: 1500
*==========================================================
```

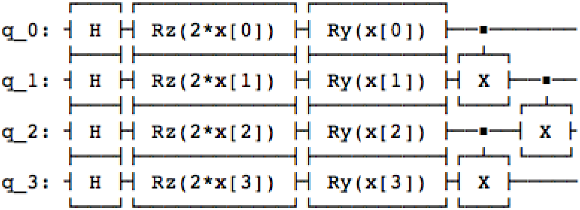

Figure: Quantum circuit of the quantum feature map $\mathcal{U}_{\phi(\vec{x})}$, is constituated by signle-qubit rotation gates (H, R_z and R_y) and two-qubit CNOT entangling gates.

Receiver Operating Characteristic (ROC) curves

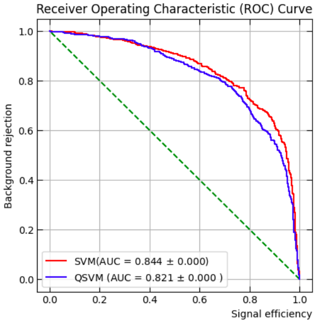

Figure: ROC curves of machine learning classifiers using the $e^+e^- \rightarrow ZH$ analysis datasets with 1500 events. The plot overlays the results of the QSVM-Kernel algorithm (blue) and classical SVM algorithm (red).

Quantum feature map (4 qubits)

```
Using the following variables:
 ['minDeltaR_y_j', 'DeltaPhi_yy', 'DeltaP_yy_jj', 'p_yy', 'e_yy', 'recoM_jj']
number of signal: 1500 number of background: 1500
*==============================================================================
```

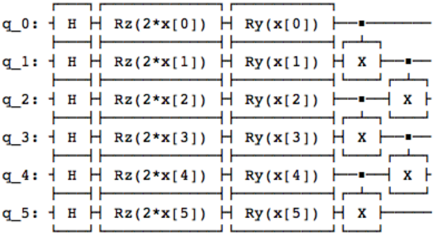

Figure: Quantum circuit of the quantum feature map $\mathcal{U}_{\phi(\vec{x})}$, is constituated by signle-qubit rotation gates (H, R_z and R_y) and two-qubit CNOT entangling gates.

Receiver Operating Characteristic (ROC) curves

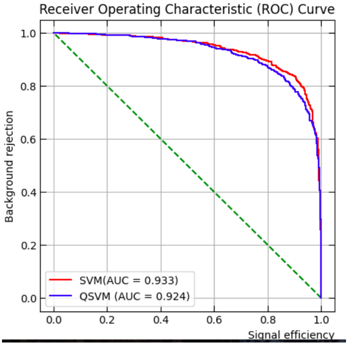

Figure: ROC curves of machine learning classifiers using the $e^+e^- \to ZH$ analysis datasets with 1500 events. The plot overlays the results of the QSVM-Kernel algorithm (blue) and classical SVM algorithm (red).

Conclusion

- Compare to the result that we had SVM algorithm is still better than QSVM algorithm for 4 qubits. However the result is quite similar once increasing the number of qubits (6 qubits).
- QSVM-Kernel, a QML that can leverage high-dimensional quantum state space for identifying a signal from backgrounds.
- Using 6-qubits on quanutm computer simulation, we can optimized the QSVM-Kernel algorithm's quantum circuits for our particles data analysis. And we had a similar classification performance to the classical SVM algorithm for 1500events.