Google
Summer of Code

ML 4 SCI
Machine Learning
for Science

# ML4Sci:
# Quantum Contrastive Learning

**Project:** Learning quantum representations of classical high energy physics data with contrastive learning

**Mentors:** Dr. Sergei Gleyzer, KC Kong, Konstanin Matchev, Katia Matchev, Myeonghun Park,Tom Magorsch, Gopal Ramesh Dahale

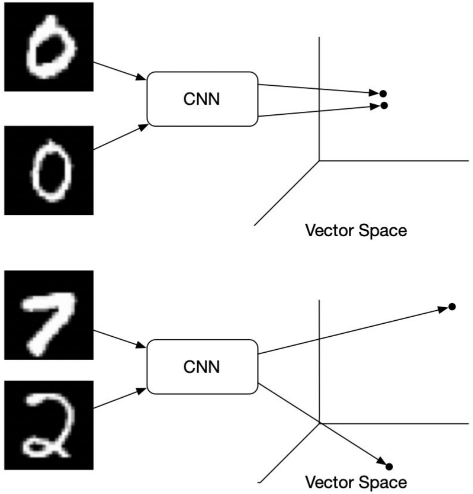
**Mentee:** Sanya Nanda

# Self Supervised Learning: Contrastive Learning

**Contrastive Learning:** They seek to quantify the similarity or dissimilarity between data elements.

**Contrastive Pair Loss**: This loss function aims to minimize the distance between similar pairs and maximize the distance between dissimilar pairs.

$$L = \frac{1}{2N} \sum_{i=1}^{N} [y_i d_i^2 + (1 - y_i) \max(0, m - d_i)^2]$$




Breaking down the loss function further:

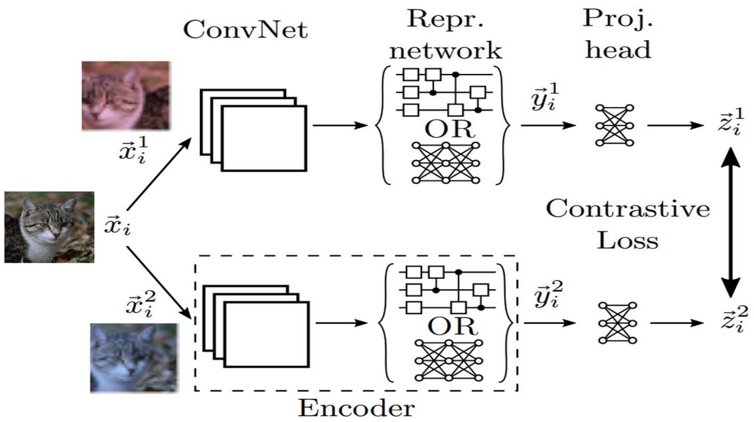
- For **positive pairs y_i = 1**, the loss encourages the distance **d_i** to be small
- For **negative pairs y_i = 0**, the loss encourages the distance **d_i** to be larger than the margin

# Siamese Network

**Siamese Network:**

Siamese networks are neural architectures used for similarity learning. They are particularly useful for our case of image similarity estimation, where we want to determine the similarity or dissimilarity between two input samples.

They consist of **two identical subnetworks (twins or branches)** that share the same weights. Each subnetwork takes an **input sample** (e.g., an image) and **produces an embedding vector** (a compact representation) for that input.



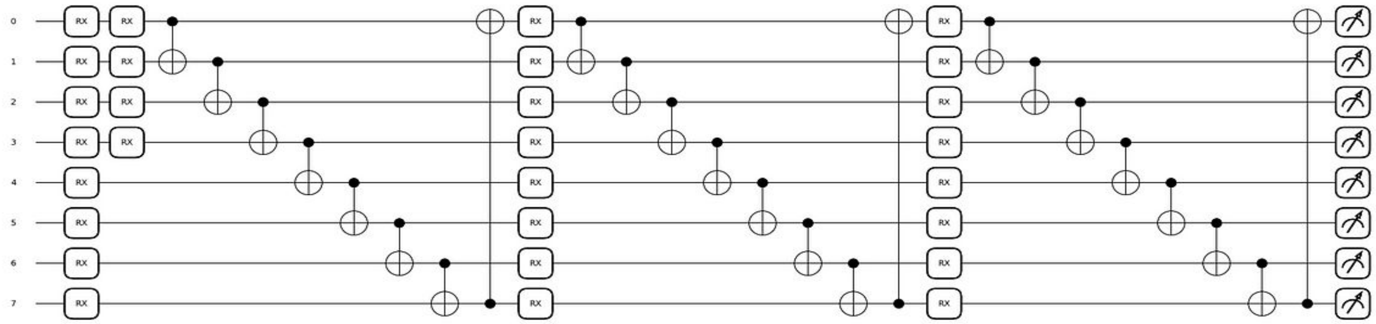

Reference: Jaderberg, B., Anderson, L.W., Xie, W., Albanie, S., Kiffner, M. and Jaksch, D., 2022. Quantum self-supervised learning. *Quantum Science and Technology*, 7(3), p.035005.

**Reference**: Oh, S., Choi, J. and Kim, J., 2020, October. A tutorial on quantum convolutional neural networks (QCNN). In *2020 International Conference on Information and Communication Technology Convergence (ICTC)* (pp. 236–239). IEEE.

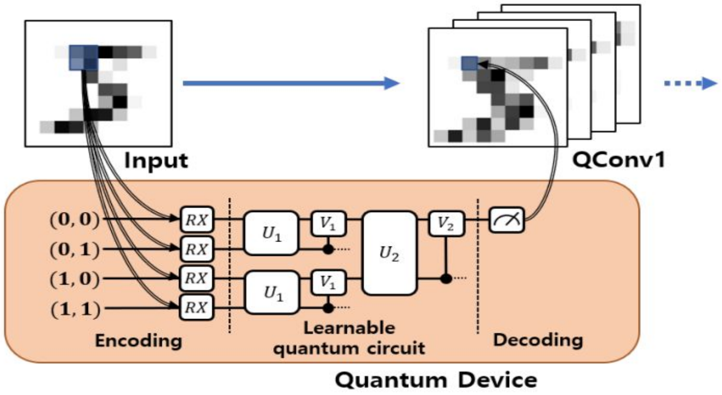
Google Summer of Code @ ML4SCI Machine Learning for Science

# Quantum Circuit and Hybrid Model

```
[3]:  (<Figure size 3100x900 with 1 Axes>, <Axes: >)
```




n_qubits = 8, n_layers = 3




Quantum Hybrid Model

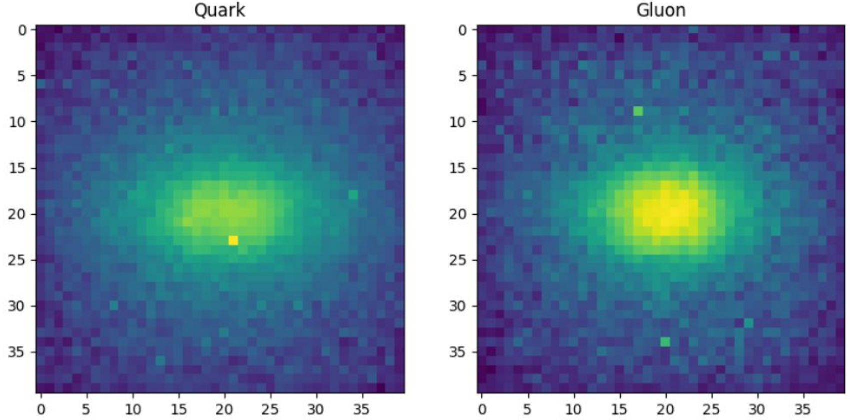
base_quantum_hybrid_model.py

```python
# Hybrid Model
class QuantumCNN:
    def __init__(self, input_shape, quantum_layer, n_qubits=4):
        self.input_shape = input_shape
        self.quantum_layer = quantum_layer
        self.n_qubits = n_qubits

    def create_model(self, return_embeddings=False):
        model = models.Sequential()
        model.add(layers.Input(shape=input_shape))
        model.add(layers.Conv2D(32, (3, 3), activation='relu')) # Conv layer 1
        model.add(layers.MaxPooling2D((2, 2)))
        model.add(layers.Conv2D(64, (3, 3), activation='relu')) # Conv layer 2
        model.add(layers.Flatten())
        model.add(layers.Dense(64, activation='relu'))

        # ------ Quantum layer
        # Reducing dimensions to match n_qubits
        model.add(layers.Dense(n_qubits))
        # Quantum layer
        model.add(quantum_layer)
        # Dense layer after quantum layer
        model.add(layers.Dense(n_qubits, activation='relu'))
        if return_embeddings:
            return model
        return model
```

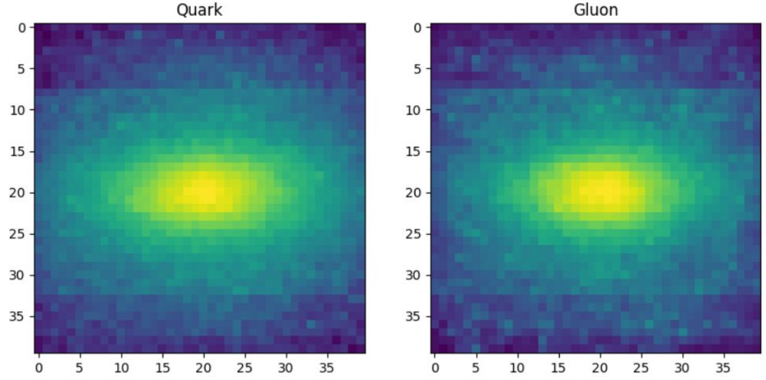
# Datasets

Google @ ML4SCI Machine Learning

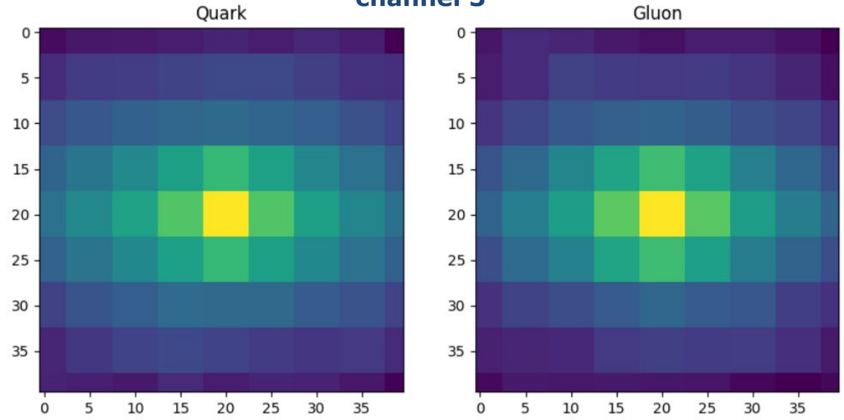
**Quark Gluon: channel 1**



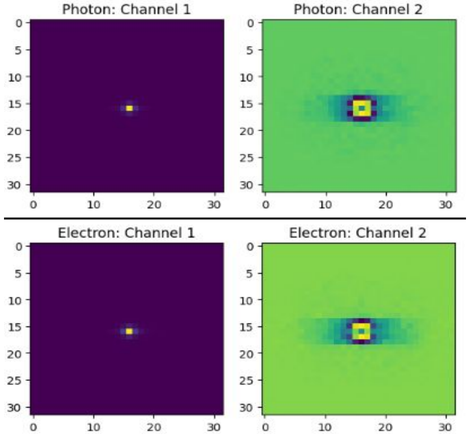

**channel 2**



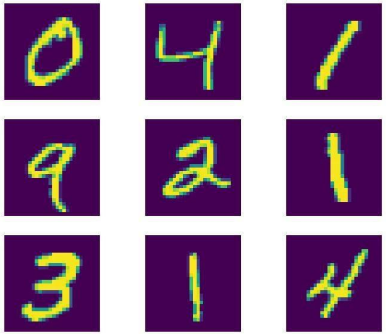

**channel 3**




**Electron and Photon**
**Left: energy channel, Right: time channel**

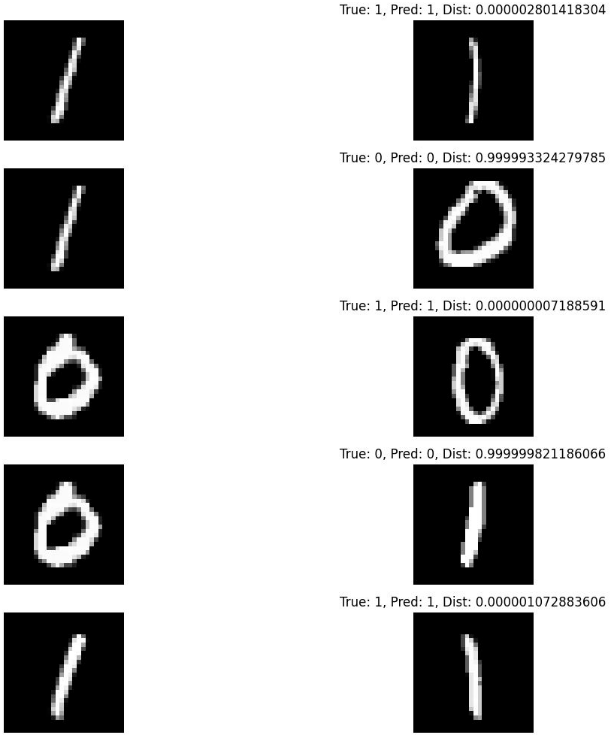


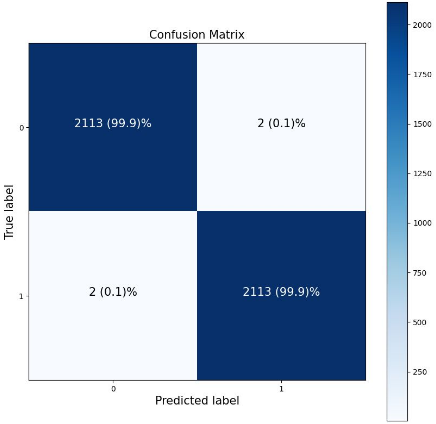

**MNIST**

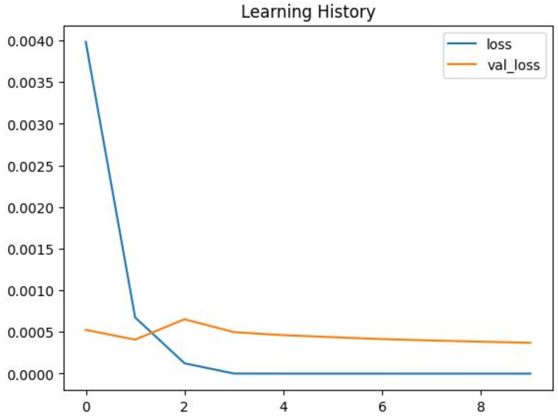


# Results on MNIST pair 0 and 1

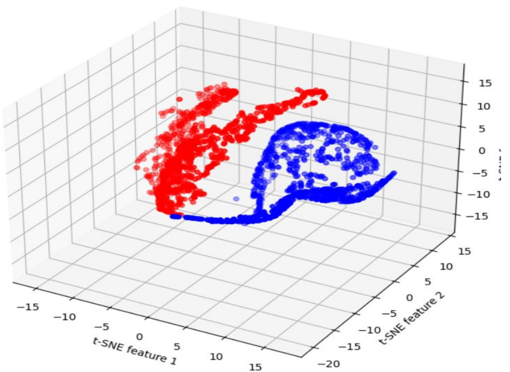
## Base QCNN model + Siamese Network + Contrastive Pair loss on MNIST dataset




3D t-SNE visualization of MNIST 0 and 1 embeddings




$$\mathcal{L}\text{InfoNCE} = -\log \frac{\exp(sim(f_i,f_j^+)/\tau)}{\sum_{k=1}^{N}\exp(sim(f_i,f_k)/\tau)}$$

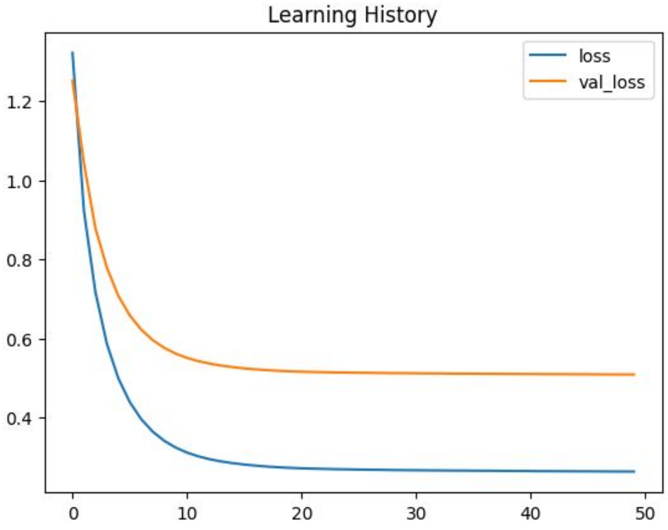
$F(\rho_1,\rho_2) = |\langle\psi_1|\psi_2\rangle|^2$ where $|\psi_1\rangle$ and $|\psi_2\rangle$

$$\mathcal{L}\text{total} = \alpha\mathcal{L}\text{InfoNCE} + (1-\alpha)(1-F(\rho_1,\rho_2))$$

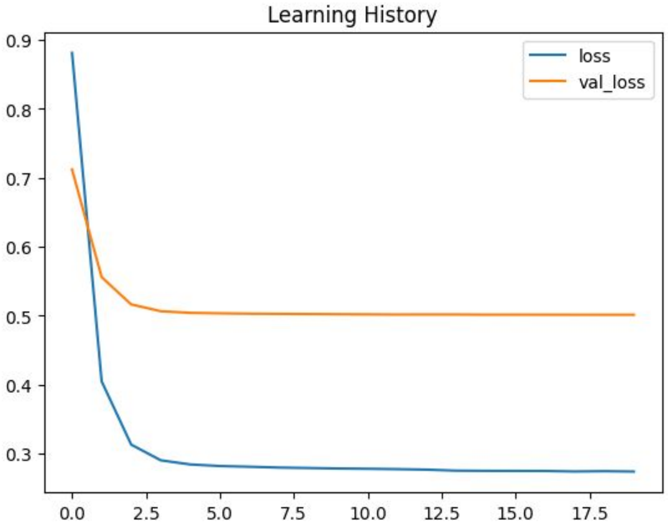
# Learning History: electron-photon

**Implemented Classical and Hybrid Base models on electron photon dataset**

**Classical**



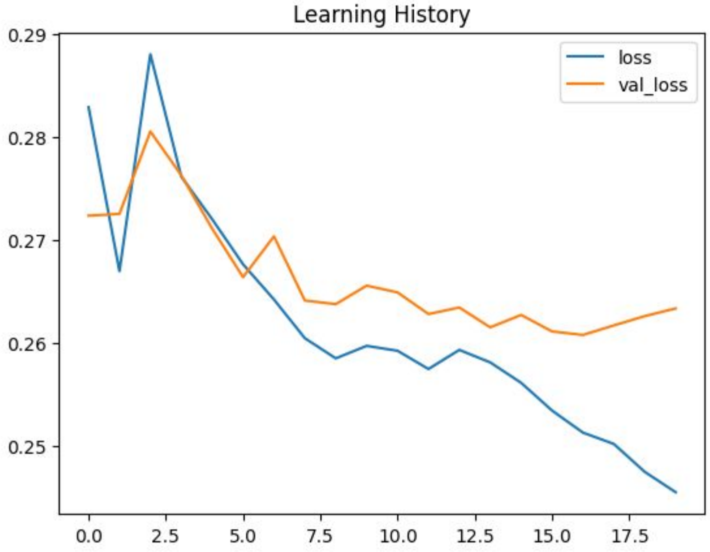

**Hybrid**



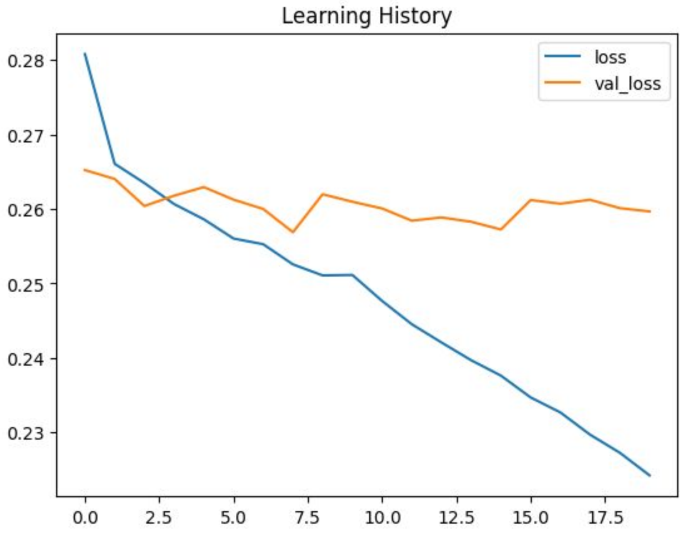
# Learning History: quark-gluon

Google Summer of Code @ ML4SCI Machine Learning for Science

## channel 1




## channel 2



Google Summer of Code @ ML4SCI Machine Learning for Science

## Next Steps

- Experimenting with **other contrastive loss functions** like triplet, SimCLR, SwaV combined with fidelity of the states

- **Quantum Graph Contrastive Learning** as graphs work better on sparse datasets which is the case in LHC-HEP datasets. Try out Vision Transformers (**QVIT**)

- **Continued iterations of experimentation** on the 3 datasets

# Thankyou!