EEE606 Project : SAMPLE CODE

Submitted by : Aradhita Sharma

Python language is used for this project.
Import necessary libraries (matplotlib, pennylane, sklearn, seaborn, pandas).
Develop two variational quantum circuits (adaptive system) with 2 and 3 qubits using the functions : Cost_fn_2qubits() and cost_fn_3qubits() , which returns the cost using qml.expval() function.
Qml.draw() for drawing the variational circuit.

PART 1 : Convergence for different optimizations :

```
Optimizers = [VGD, ADAM, QNG]
For i in optimizers :
    For j in range(0,500) # 500 iteration steps
        • Give initial params to cost_fn
        • Obtain the updated params by evaluating the cost_fn gradient
          optimizers(i)
        • Append the cost_fn error value to separate cost arrays for different
          optimizers
```

Plot the cost_fn convergence curves for different optimizers.

PART 2 : Binary Classification :

Import dataset from sklearn library and split the dataset into training and testing data (80-20%).

Quantum binary Classification :

- Construct quantum circuit with angle encoding of input values, and adaptive parameters as entangled layers, which is the variational classifier which returns +1 and -1 as two different class outputs.
- Split the data into 5 batches.
- Run the above convergence for 5 epochs with adam optimization.
- Test the model weights on testing data, and obtain the accuracy and confusion matrix.

Classical binary Classification :

- Input values are passed through the model with 4 inputs and 1 binary output,
- Sigmoid function is the activation function.
- Weights are updated for 5 epochs with Vanilla gradient descent optimization (calculating the gradient of sigmoid).
- Test the model weights on testing data, and obtain the accuracy and confusion matrix.