1st talk Jupyter notebook NITheP_mini_school_L1-qubit_rot

September 15, 2020

1 Optimising a qubit rotation model with PennyLane

Source: https://pennylane.ai/qml/demos/tutorial_qubit_rotation.html

1.1 Imports

[1]:
```
# insert imports
import pennylane as qml
from pennylane import numpy as np
```

1.2 Create a device

[2]:
```
# default.qubit, other devices available
dev = qml.device('default.qubit', wires=1)
```

1.3 Construct a QNode with your circuit

[3]:
```
# wrap device in qml.qnode
# define circuit, RX and RY gates, <Z>
@qml.qnode(dev)
def circuit(params):
    qml.RX(params[0], wires=0)
    qml.RY(params[1], wires=0)
    return qml.expval(qml.PauliZ(0))
```

[5]:
```
# set some parameters & print the circuit
params = [0,0]
circuit(params)
```

[5]:
```
1.0
```

1.4 Define the cost function

[6]:
```
#define cost(x)
def cost(x):
    return circuit(x)
```

[7]:
```
#define init_params as np array and check cost
init_params = np.array([0.11, 0.5])
cost(init_params)
```

[7]:
```
0.8722785388513962
```

1.5 Optimise the cost function using gradient descent

[8]:
```
# initialise the optimizer with stepsize
opt = qml.GradientDescentOptimizer(stepsize=0.4)

# set the number of steps
steps = 100
# set the initial parameter values
params = init_params
```

[9]:
```
for i in range(steps):
    # update the circuit parameters after each step relative to cost & params
    params = opt.step(cost, params)

    if (i + 1) % 5 == 0:
        print("Cost after step {:5d}: {: .7f}".format(i + 1, cost(params)))

print("Optimized rotation angles: {}".format(params))
```

```
Cost after step     5: -0.4151640
Cost after step    10: -0.9926820
Cost after step    15: -0.9999554
Cost after step    20: -0.9999997
Cost after step    25: -1.0000000
Cost after step    30: -1.0000000
Cost after step    35: -1.0000000
Cost after step    40: -1.0000000
Cost after step    45: -1.0000000
Cost after step    50: -1.0000000
Cost after step    55: -1.0000000
Cost after step    60: -1.0000000
Cost after step    65: -1.0000000
Cost after step    70: -1.0000000
Cost after step    75: -1.0000000
Cost after step    80: -1.0000000
Cost after step    85: -1.0000000
Cost after step    90: -1.0000000
Cost after step    95: -1.0000000
Cost after step   100: -1.0000000
Optimized rotation angles: [7.76477091e-17 3.14159265e+00]
```

[ ]: