Liam Cui
CSCI 739 HW4

1)
a.
Angle encoding:

$$R_y(\pi x)|0\rangle = \cos\left(\frac{\pi x}{2}\right)|0\rangle + \sin\left(\frac{\pi x}{2}\right)|1\rangle$$

$$after\ applying\ M\ and\ U(\theta)$$

$$f_{QNN}(x;\theta) = \sum_k a_k(\theta)\cos(k\pi x) + b_k(\theta)\sin(k\pi x)$$

This is a trigonometric polynomial because the applied $U(\theta)$ are made of trigonometric functions introduced by the angle encoding.

b.

$$R_y(\theta)R_y(\pi x) = R_y(\theta + \pi x)$$

$$|\theta\rangle = R_y(\theta + \pi x)|0\rangle = \cos\left(\frac{\theta + \pi x}{2}\right)|0\rangle + \sin\left(\frac{\theta + \pi x}{2}\right)|1\rangle$$

$$\langle Z\rangle = |\langle 0|\phi\rangle|^2 + |\langle 1|\phi\rangle|^2 = \cos^2\left(\frac{\theta + \pi x}{2}\right) - \sin^2\left(\frac{\theta + \pi x}{2}\right) = \cos\left(2\left(\frac{\theta + \pi x}{2}\right)\right)$$
$$= \cos(\theta + \pi x)$$

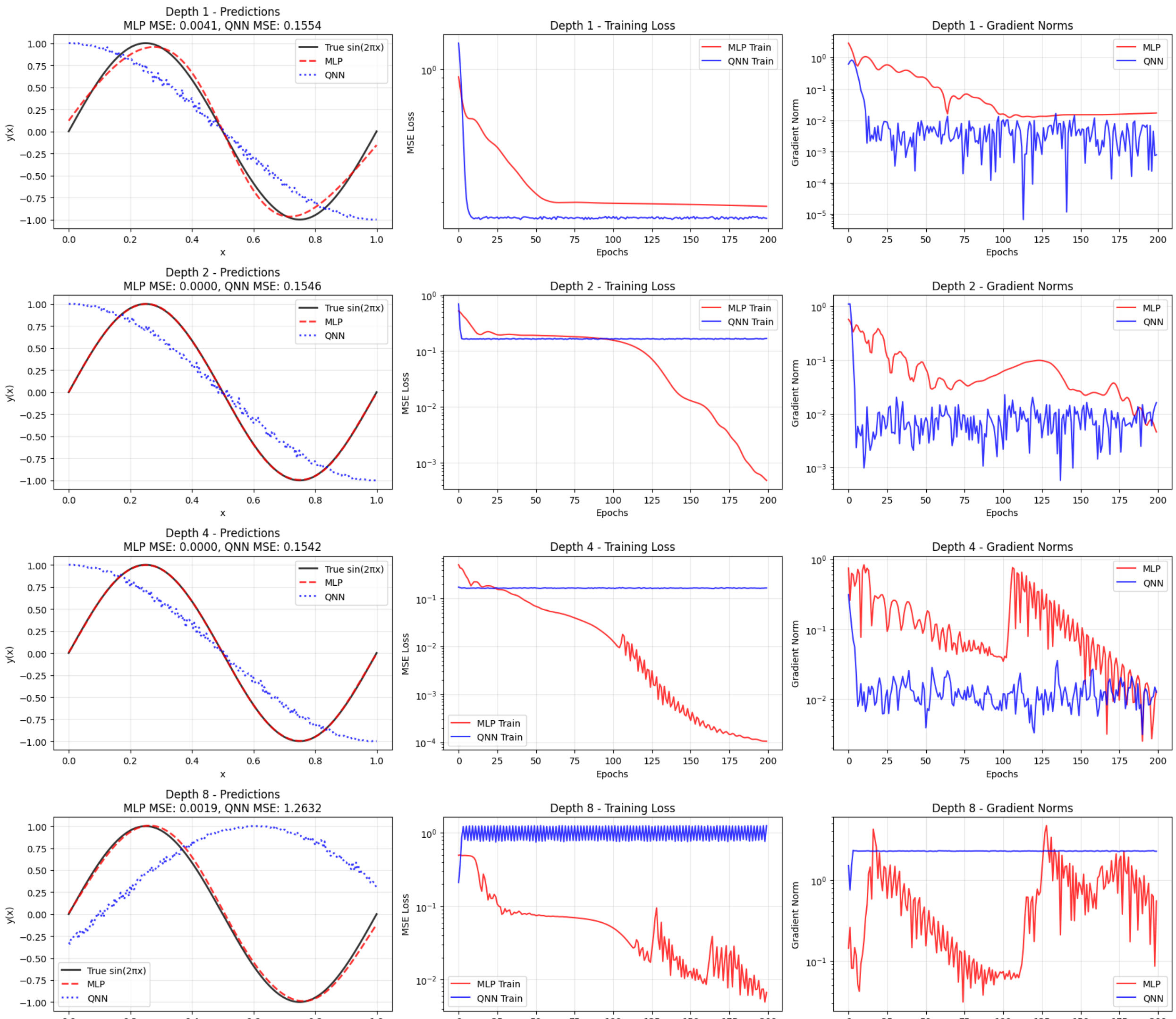
$$f_{QNN}(x;\theta) = \cos(\theta + \pi x)$$

This is a simple polynomial with a fixed amplitude which phases are controlled by theta parameter.

c. Angle encoding produces trig functions by rotating the inputs. Doing this with multiple qubits allows us to create larger series of trigonometric polynomials based on how many qubits. By using the layered circuit with parametrized unitarizes, we can combine these into larger degree polynomials. So when circuit depth increase we are allowing for more combining of the functions into more and more complex trig functions. Therefore, given sufficient qubit counts and circuit depth, we can theoretically approximate any continuous function. This is like how classical can also model a continuous function through a combination of smaller functions except instead of trig functions they would be using nonlinear activation functions like ReLu.

d. If we are only using linear encodings, we cannot utilize the parametrized unitaries like we did in angle encoding to create more complex shapes. This results in extremely limited shapes and therefore a much worse function approximation.

1)



FINAL PERFORMANCE COMPARISON

Depth	Model	Train MSE	Test MSE	Final Grad Norm
1	MLP	0.004279	0.004062	0.004583
1	QNN	0.164703	0.155441	0.000780
2	MLP	0.000014	0.000013	0.000049
2	QNN	0.168675	0.154644	0.016112

As you can see by the QNNs failed to match the true sin regardless of the depth. This is likely because we are only using a single qubit Ry gate and so we don't have enough qubits to generate enough trigonometric functions for an accurate expression of the true sin curve and since we don't have enough functions, we won't be able to leverage the higher depths as we can't merge non-existent functions. Contrarily the highest depth showed a large increasing change to our grad norms. As we know increasing the depth will increase the density of combinations by a polynomial factor. This also increases the number of times any minimizing factor can be applied which in exacerbates the effects of a vanishing gradient.