## Comparison between Pi-Net and PolinomialActivadedNet

# Pinet

In [ ]:
```
import tensorflow as tf
from tensorflow.keras.layers import Input, Add, Dense, Layer
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam

def polynomial_activation(x, degree=1):
    if degree == 1:
        return x
    elif degree == 2:
        return x * x
    elif degree == 3:
        return x**3
    else:
        raise ValueError("Invalid degree specified, only 1st, 2nd and 3rd degree po

class TensorDecompositionLayer(Layer):
    def __init__(self, rank, **kwargs):
        self.rank = rank
        super(TensorDecompositionLayer, self).__init__(**kwargs)

    def build(self, input_shape):
        self.factors_a = self.add_weight(shape=(input_shape[-1], self.rank),
                                         initializer='random_normal',
                                         trainable=True)
        self.factors_b = self.add_weight(shape=(self.rank, input_shape[-1]),
                                         initializer='random_normal',
                                         trainable=True)
        super(TensorDecompositionLayer, self).build(input_shape)

    def call(self, x):
        return tf.matmul(tf.matmul(x, self.factors_a), self.factors_b)

def resnet_block(x, filters, activation_degree, rank=None):
    shortcut = x
    x = Dense(filters)(x)
    x = tf.keras.layers.Lambda(lambda y: polynomial_activation(y, degree=activation

    if rank is not None:
        x = TensorDecompositionLayer(rank)(x)

    x = Dense(filters)(x)
    x = Add()([x, shortcut])
    return x

def build_model(input_shape, num_blocks, filters, activation_degree, rank=None):
    input_layer = Input(shape=input_shape)
    x = input_layer
```

```
    for _ in range(num_blocks):
        x = resnet_block(x, filters, activation_degree, rank)

    output_layer = Dense(1)(x)
    model = Model(inputs=input_layer, outputs=output_layer)

    return model

input_shape = (1,)
num_blocks = 3
filters = 16
activation_degree = 3  # Change to 1 for 1st degree polynomial, 2 for 2nd degree, a
rank = 4  # Tensor decomposition rank, set to None if you don't want to use tensor

model = build_model(input_shape, num_blocks, filters, activation_degree, rank)
model.compile(optimizer='adam', loss='mse')
```

In [ ]:

```
import numpy as np

np.random.seed(42)
n_samples = 10000
# lower_bound = -2 * np.pi
# upper_bound = 2 * np.pi
lower_bound = -10
upper_bound = 10

X = np.random.uniform(lower_bound, upper_bound, size=(n_samples, 1))
y = np.cos(X)

from sklearn.model_selection import train_test_split

X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state

batch_size = 64
epochs = 200

history = model.fit(X_train, y_train,
                    batch_size=batch_size,
                    epochs=epochs,
                    verbose=1,
                    validation_data=(X_val, y_val))
```

```
Epoch 1/200
125/125 [==============================] - 1s 2ms/step - loss: 731.6659 - val_loss:
0.9881
Epoch 2/200
125/125 [==============================] - 0s 1ms/step - loss: 0.7826 - val_loss: 0.
7283
Epoch 3/200
125/125 [==============================] - 0s 1ms/step - loss: 0.7015 - val_loss: 0.
6956
Epoch 4/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6764 - val_loss: 0.
6742
Epoch 5/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6572 - val_loss: 0.
6585
Epoch 6/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6441 - val_loss: 0.
6454
Epoch 7/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6323 - val_loss: 0.
6329
Epoch 8/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6208 - val_loss: 0.
6221
Epoch 9/200
125/125 [==============================] - 0s 1ms/step - loss: 0.6130 - val_loss: 0.
6203
Epoch 10/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5988 - val_loss: 0.
6029
Epoch 11/200
125/125 [==============================] - 0s 1000us/step - loss: 0.5905 - val_loss:
0.5936
Epoch 12/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5843 - val_loss: 0.
5881
Epoch 13/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5778 - val_loss: 0.
5882
Epoch 14/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5727 - val_loss: 0.
5783
Epoch 15/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5647 - val_loss: 0.
5698
Epoch 16/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5622 - val_loss: 0.
5635
Epoch 17/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5548 - val_loss: 0.
5586
Epoch 18/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5446 - val_loss: 0.
5501
Epoch 19/200
125/125 [==============================] - 0s 992us/step - loss: 0.5401 - val_loss:
```

```
0.5485
Epoch 20/200
125/125 [==============================] - 0s 1000us/step - loss: 0.5384 - val_loss: 
0.5347
Epoch 21/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5415 - val_loss: 0.
5469
Epoch 22/200
125/125 [==============================] - 0s 988us/step - loss: 0.5218 - val_loss: 
0.5194
Epoch 23/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5157 - val_loss: 0.
5315
Epoch 24/200
125/125 [==============================] - 0s 988us/step - loss: 0.5089 - val_loss: 
0.5438
Epoch 25/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5097 - val_loss: 0.
4984
Epoch 26/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4955 - val_loss: 0.
4933
Epoch 27/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4947 - val_loss: 0.
5532
Epoch 28/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4918 - val_loss: 0.
4717
Epoch 29/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5128 - val_loss: 0.
4885
Epoch 30/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4737 - val_loss: 0.
4602
Epoch 31/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4504 - val_loss: 0.
4612
Epoch 32/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4412 - val_loss: 0.
4512
Epoch 33/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4289 - val_loss: 0.
4136
Epoch 34/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4171 - val_loss: 0.
4056
Epoch 35/200
125/125 [==============================] - 0s 996us/step - loss: 0.3984 - val_loss: 
0.3830
Epoch 36/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3880 - val_loss: 0.
3967
Epoch 37/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3583 - val_loss: 0.
3538
Epoch 38/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.3477 - val_loss: 0.
3461
Epoch 39/200
125/125 [==============================] - 0s 996us/step - loss: 0.3370 - val_loss:
0.3150
Epoch 40/200
125/125 [==============================] - 0s 996us/step - loss: 0.3111 - val_loss:
0.2976
Epoch 41/200
125/125 [==============================] - 0s 992us/step - loss: 0.3119 - val_loss:
0.2815
Epoch 42/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2718 - val_loss: 0.
2682
Epoch 43/200
125/125 [==============================] - 0s 1000us/step - loss: 0.2772 - val_loss:
0.2568
Epoch 44/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2512 - val_loss: 0.
2531
Epoch 45/200
125/125 [==============================] - 0s 988us/step - loss: 0.2526 - val_loss:
0.2137
Epoch 46/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2173 - val_loss: 0.
1949
Epoch 47/200
125/125 [==============================] - 0s 992us/step - loss: 0.1936 - val_loss:
0.1776
Epoch 48/200
125/125 [==============================] - 0s 984us/step - loss: 0.1858 - val_loss:
0.1643
Epoch 49/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2026 - val_loss: 0.
1462
Epoch 50/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1388 - val_loss: 0.
1271
Epoch 51/200
125/125 [==============================] - 0s 988us/step - loss: 0.1233 - val_loss:
0.1245
Epoch 52/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1197 - val_loss: 0.
1109
Epoch 53/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1073 - val_loss: 0.
0949
Epoch 54/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0977 - val_loss: 0.
1121
Epoch 55/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0843 - val_loss: 0.
0727
Epoch 56/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0723 - val_loss: 0.
0640
```

```
Epoch 57/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0595 - val_loss: 0.
0563
Epoch 58/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0520 - val_loss: 0.
0489
Epoch 59/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0470 - val_loss: 0.
0428
Epoch 60/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0427 - val_loss: 0.
0408
Epoch 61/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0365 - val_loss: 0.
0609
Epoch 62/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0324 - val_loss: 0.
0283
Epoch 63/200
125/125 [==============================] - 0s 996us/step - loss: 0.0266 - val_loss:
0.0264
Epoch 64/200
125/125 [==============================] - 0s 992us/step - loss: 0.0225 - val_loss:
0.0226
Epoch 65/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0224 - val_loss: 0.
0194
Epoch 66/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0178 - val_loss: 0.
0160
Epoch 67/200
125/125 [==============================] - 0s 1000us/step - loss: 0.0160 - val_loss:
0.0158
Epoch 68/200
125/125 [==============================] - 0s 992us/step - loss: 0.0148 - val_loss:
0.0136
Epoch 69/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0136 - val_loss: 0.
0167
Epoch 70/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0131 - val_loss: 0.
0131
Epoch 71/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0118 - val_loss: 0.
0171
Epoch 72/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0151 - val_loss: 0.
0110
Epoch 73/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0148 - val_loss: 0.
0105
Epoch 74/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0134 - val_loss: 0.
0124
Epoch 75/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0137 - val_loss: 0.
```

```
0131
Epoch 76/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0111 - val_loss: 0.
0082
Epoch 77/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0131 - val_loss: 0.
0130
Epoch 78/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0126 - val_loss: 0.
0365
Epoch 79/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0104 - val_loss: 0.
0082
Epoch 80/200
125/125 [==============================] - 0s 988us/step - loss: 0.0088 - val_loss:
0.0086
Epoch 81/200
125/125 [==============================] - 0s 988us/step - loss: 0.0090 - val_loss:
0.0109
Epoch 82/200
125/125 [==============================] - 0s 988us/step - loss: 0.0085 - val_loss:
0.0058
Epoch 83/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0091 - val_loss: 0.
0072
Epoch 84/200
125/125 [==============================] - 0s 1000us/step - loss: 0.0084 - val_loss:
0.0098
Epoch 85/200
125/125 [==============================] - 0s 996us/step - loss: 0.0103 - val_loss:
0.0160
Epoch 86/200
125/125 [==============================] - 0s 992us/step - loss: 0.0062 - val_loss:
0.0052
Epoch 87/200
125/125 [==============================] - 0s 1000us/step - loss: 0.0065 - val_loss:
0.0080
Epoch 88/200
125/125 [==============================] - 0s 996us/step - loss: 0.0108 - val_loss:
0.0059
Epoch 89/200
125/125 [==============================] - 0s 984us/step - loss: 0.0089 - val_loss:
0.0043
Epoch 90/200
125/125 [==============================] - 0s 988us/step - loss: 0.0058 - val_loss:
0.0055
Epoch 91/200
125/125 [==============================] - 0s 992us/step - loss: 0.0071 - val_loss:
0.0046
Epoch 92/200
125/125 [==============================] - 0s 992us/step - loss: 0.0063 - val_loss:
0.0037
Epoch 93/200
125/125 [==============================] - 0s 996us/step - loss: 0.0062 - val_loss:
0.0040
Epoch 94/200
```

```
125/125 [==============================] - 0s 996us/step - loss: 0.0062 - val_loss:
0.0034
Epoch 95/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0043 - val_loss: 0.
0025
Epoch 96/200
125/125 [==============================] - 0s 988us/step - loss: 0.0038 - val_loss:
0.0045
Epoch 97/200
125/125 [==============================] - 0s 1000us/step - loss: 0.0050 - val_loss:
0.0047
Epoch 98/200
125/125 [==============================] - 0s 992us/step - loss: 0.0043 - val_loss:
0.0029
Epoch 99/200
125/125 [==============================] - 0s 996us/step - loss: 0.0034 - val_loss:
0.0035
Epoch 100/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3068 - val_loss: 0.
1687
Epoch 101/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0753 - val_loss: 0.
0330
Epoch 102/200
125/125 [==============================] - 0s 988us/step - loss: 0.0270 - val_loss:
0.0830
Epoch 103/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0314 - val_loss: 0.
0155
Epoch 104/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0128 - val_loss: 0.
0116
Epoch 105/200
125/125 [==============================] - 0s 992us/step - loss: 0.0118 - val_loss:
0.0084
Epoch 106/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0036 - val_loss: 0.
0032
Epoch 107/200
125/125 [==============================] - 0s 992us/step - loss: 0.0019 - val_loss:
0.0014
Epoch 108/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0017 - val_loss: 0.
0136
Epoch 109/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0231 - val_loss: 0.
0072
Epoch 110/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0021 - val_loss: 0.
0039
Epoch 111/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0012 - val_loss: 0.
0013
Epoch 112/200
125/125 [==============================] - 0s 996us/step - loss: 0.0019 - val_loss:
0.0027
```

```
Epoch 113/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0024 - val_loss: 0.
0018
Epoch 114/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0034 - val_loss: 0.
0012
Epoch 115/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0018 - val_loss: 0.
0014
Epoch 116/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0103 - val_loss: 0.
0030
Epoch 117/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0018 - val_loss: 5.
0540e-04
Epoch 118/200
125/125 [==============================] - 0s 992us/step - loss: 0.0023 - val_loss:
0.0072
Epoch 119/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0032 - val_loss: 7.
3982e-04
Epoch 120/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0117 - val_loss: 0.
0148
Epoch 121/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0016 - val_loss: 4.
9195e-04
Epoch 122/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0021 - val_loss: 0.
0013
Epoch 123/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0024 - val_loss: 6.
4237e-04
Epoch 124/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0060 - val_loss: 0.
0048
Epoch 125/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0019 - val_loss: 0.
0076
Epoch 126/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0021 - val_loss: 2.
9001e-04
Epoch 127/200
125/125 [==============================] - 0s 1ms/step - loss: 6.3122e-04 - val_los
s: 3.2957e-04
Epoch 128/200
125/125 [==============================] - 0s 1ms/step - loss: 6.1535e-04 - val_los
s: 6.6291e-04
Epoch 129/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0012 - val_loss: 4.
6997e-04
Epoch 130/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0029 - val_loss: 0.
0017
Epoch 131/200
125/125 [==============================] - 0s 1ms/step - loss: 8.0663e-04 - val_los
```

```
s: 8.9463e-04
Epoch 132/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0058 - val_loss: 0.
0092
Epoch 133/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0518 - val_loss: 0.
0050
Epoch 134/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0024 - val_loss: 0.
0031
Epoch 135/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0011 - val_loss: 7.
5347e-04
Epoch 136/200
125/125 [==============================] - 0s 1ms/step - loss: 7.3661e-04 - val_los
s: 7.7171e-04
Epoch 137/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0016 - val_loss: 0.
0015
Epoch 138/200
125/125 [==============================] - 0s 1ms/step - loss: 8.4196e-04 - val_los
s: 6.0947e-04
Epoch 139/200
125/125 [==============================] - 0s 1ms/step - loss: 7.7339e-04 - val_los
s: 6.7521e-04
Epoch 140/200
125/125 [==============================] - 0s 1ms/step - loss: 8.0197e-04 - val_los
s: 2.8393e-04
Epoch 141/200
125/125 [==============================] - 0s 1000us/step - loss: 5.8177e-04 - val_l
oss: 4.4587e-04
Epoch 142/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0017 - val_loss: 0.
0158
Epoch 143/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0063 - val_loss: 0.
0011
Epoch 144/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0010 - val_loss: 4.
6336e-04
Epoch 145/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0022 - val_loss: 0.
0087
Epoch 146/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0018 - val_loss: 4.
6050e-04
Epoch 147/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0057 - val_loss: 3.
9938e-04
Epoch 148/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0021 - val_loss: 0.
0102
Epoch 149/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0054 - val_loss: 5.
9962e-04
Epoch 150/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 7.2370e-04 - val_los
s: 4.0338e-04
Epoch 151/200
125/125 [==============================] - 0s 1ms/step - loss: 4.9453e-04 - val_los
s: 1.5771e-04
Epoch 152/200
125/125 [==============================] - 0s 1ms/step - loss: 6.1826e-04 - val_los
s: 1.8318e-04
Epoch 153/200
125/125 [==============================] - 0s 1ms/step - loss: 2.3298e-04 - val_los
s: 4.4925e-04
Epoch 154/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0026 - val_loss: 0.
0135
Epoch 155/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0253 - val_loss: 0.
0031
Epoch 156/200
125/125 [==============================] - 0s 1ms/step - loss: 5.4216e-04 - val_los
s: 1.3836e-04
Epoch 157/200
125/125 [==============================] - 0s 1ms/step - loss: 1.1976e-04 - val_los
s: 6.8517e-05
Epoch 158/200
125/125 [==============================] - 0s 1ms/step - loss: 7.4420e-05 - val_los
s: 1.7367e-05
Epoch 159/200
125/125 [==============================] - 0s 1000us/step - loss: 2.6483e-05 - val_l
oss: 3.0915e-05
Epoch 160/200
125/125 [==============================] - 0s 996us/step - loss: 2.2309e-04 - val_lo
ss: 6.2748e-04
Epoch 161/200
125/125 [==============================] - 0s 1000us/step - loss: 1.8263e-04 - val_l
oss: 1.1292e-05
Epoch 162/200
125/125 [==============================] - 0s 1ms/step - loss: 2.4105e-05 - val_los
s: 2.0065e-05
Epoch 163/200
125/125 [==============================] - 0s 996us/step - loss: 1.7320e-05 - val_lo
ss: 1.0761e-05
Epoch 164/200
125/125 [==============================] - 0s 1ms/step - loss: 1.5893e-05 - val_los
s: 7.0227e-05
Epoch 165/200
125/125 [==============================] - 0s 1ms/step - loss: 5.8411e-05 - val_los
s: 4.7416e-05
Epoch 166/200
125/125 [==============================] - 0s 1ms/step - loss: 1.6442e-05 - val_los
s: 2.8428e-05
Epoch 167/200
125/125 [==============================] - 0s 1ms/step - loss: 9.0111e-05 - val_los
s: 7.2689e-05
Epoch 168/200
125/125 [==============================] - 0s 992us/step - loss: 0.0375 - val_loss:
0.0125
```

```
Epoch 169/200
125/125 [==============================] - 0s 988us/step - loss: 0.0031 - val_loss:
3.3482e-04
Epoch 170/200
125/125 [==============================] - 0s 1ms/step - loss: 2.0659e-04 - val_los
s: 1.2027e-04
Epoch 171/200
125/125 [==============================] - 0s 1ms/step - loss: 1.7571e-04 - val_los
s: 1.5071e-04
Epoch 172/200
125/125 [==============================] - 0s 1ms/step - loss: 1.5659e-04 - val_los
s: 9.2055e-05
Epoch 173/200
125/125 [==============================] - 0s 996us/step - loss: 9.1672e-05 - val_lo
ss: 3.3462e-05
Epoch 174/200
125/125 [==============================] - 0s 1ms/step - loss: 9.4642e-05 - val_los
s: 7.2600e-05
Epoch 175/200
125/125 [==============================] - 0s 1ms/step - loss: 1.0831e-04 - val_los
s: 3.2469e-05
Epoch 176/200
125/125 [==============================] - 0s 1000us/step - loss: 2.3483e-04 - val_l
oss: 2.1112e-04
Epoch 177/200
125/125 [==============================] - 0s 1ms/step - loss: 1.9553e-04 - val_los
s: 2.5477e-05
Epoch 178/200
125/125 [==============================] - 0s 1ms/step - loss: 7.9329e-05 - val_los
s: 2.3553e-04
Epoch 179/200
125/125 [==============================] - 0s 1ms/step - loss: 9.3922e-04 - val_los
s: 9.0597e-04
Epoch 180/200
125/125 [==============================] - 0s 996us/step - loss: 0.0049 - val_loss:
0.0084
Epoch 181/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0080 - val_loss: 3.
7035e-04
Epoch 182/200
125/125 [==============================] - 0s 1000us/step - loss: 1.6753e-04 - val_l
oss: 5.7127e-05
Epoch 183/200
125/125 [==============================] - 0s 1ms/step - loss: 4.7675e-05 - val_los
s: 3.6015e-05
Epoch 184/200
125/125 [==============================] - 0s 1ms/step - loss: 4.3249e-05 - val_los
s: 1.8672e-05
Epoch 185/200
125/125 [==============================] - 0s 1ms/step - loss: 2.2453e-05 - val_los
s: 1.6901e-05
Epoch 186/200
125/125 [==============================] - 0s 1ms/step - loss: 6.2579e-05 - val_los
s: 9.5858e-05
Epoch 187/200
125/125 [==============================] - 0s 1ms/step - loss: 1.5720e-04 - val_los
```

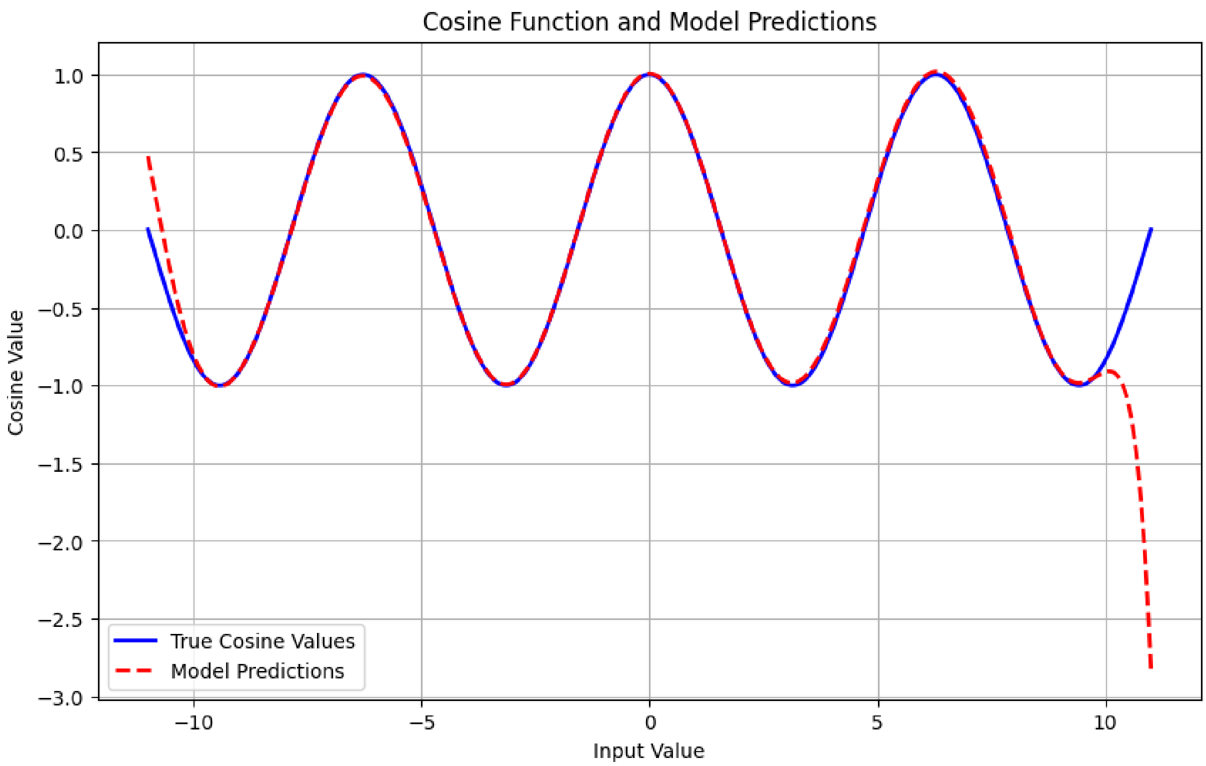
```
s: 4.3822e-04
Epoch 188/200
125/125 [==============================] - 0s 1ms/step - loss: 9.6473e-05 - val_los
s: 1.0051e-04
Epoch 189/200
125/125 [==============================] - 0s 1ms/step - loss: 3.8184e-05 - val_los
s: 3.1834e-05
Epoch 190/200
125/125 [==============================] - 0s 1ms/step - loss: 6.8509e-05 - val_los
s: 5.0552e-05
Epoch 191/200
125/125 [==============================] - 0s 1ms/step - loss: 3.5264e-04 - val_los
s: 0.0070
Epoch 192/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0995 - val_loss: 0.
0034
Epoch 193/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0059 - val_loss: 5.
3645e-04
Epoch 194/200
125/125 [==============================] - 0s 1ms/step - loss: 4.2885e-04 - val_los
s: 2.9573e-04
Epoch 195/200
125/125 [==============================] - 0s 1ms/step - loss: 3.0953e-04 - val_los
s: 2.8138e-04
Epoch 196/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0010 - val_loss: 0.
0024
Epoch 197/200
125/125 [==============================] - 0s 1000us/step - loss: 8.1029e-04 - val_l
oss: 1.4913e-04
Epoch 198/200
125/125 [==============================] - 0s 996us/step - loss: 2.2466e-04 - val_lo
ss: 1.3704e-04
Epoch 199/200
125/125 [==============================] - 0s 996us/step - loss: 1.8371e-04 - val_lo
ss: 9.7466e-05
Epoch 200/200
125/125 [==============================] - 0s 1ms/step - loss: 2.0418e-04 - val_los
s: 2.3438e-04
```

In [ ]:

```
val_loss = model.evaluate(X_val, y_val, verbose=0)
print(f"Validation loss: {val_loss}")

import matplotlib.pyplot as plt

num_test_samples = 1000
X_test = np.linspace(-11, 11, num=num_test_samples).reshape(-1, 1)
y_true = np.cos(X_test)
y_pred = model.predict(X_test)

plt.figure(figsize=(10, 6))
plt.plot(X_test, y_true, label='True Cosine Values', color='b', linewidth=2)
plt.plot(X_test, y_pred, label='Model Predictions', color='r', linestyle='--', line
plt.xlabel('Input Value')
plt.ylabel('Cosine Value')
```

```
plt.title('Cosine Function and Model Predictions')
plt.legend()
plt.grid()
plt.show()
```

```
Validation loss: 0.00023438122298102826
32/32 [==============================] - 0s 544us/step
```




# Polinomial Architecture

In [ ]:
```
import tensorflow as tf
from tensorflow.keras.layers import Input, Add, Dense, Layer
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam

class PolynomialLayer(Layer):
    def __init__(self, **kwargs):
        super(PolynomialLayer, self).__init__(**kwargs)

    def build(self, input_shape):
        self.a = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        self.b = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        self.c = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        super(PolynomialLayer, self).build(input_shape)
```

```
    def call(self, x):
        return self.a + self.b * x + self.c * x**2

class FunctionCompositionLayer(Layer):
    def __init__(self, activation_1, activation_2, **kwargs):
        self.activation_1 = activation_1
        self.activation_2 = activation_2
        super(FunctionCompositionLayer, self).__init__(**kwargs)

    def call(self, x):
        x = self.activation_1(x)
        x = self.activation_2(x)
        return x

class TensorDecompositionLayer(Layer):
    def __init__(self, rank, **kwargs):
        self.rank = rank
        super(TensorDecompositionLayer, self).__init__(**kwargs)

    def build(self, input_shape):
        self.factors_a = self.add_weight(shape=(input_shape[-1], self.rank),
                                         initializer='random_normal',
                                         trainable=True)
        self.factors_b = self.add_weight(shape=(self.rank, input_shape[-1]),
                                         initializer='random_normal',
                                         trainable=True)
        super(TensorDecompositionLayer, self).build(input_shape)

    def call(self, x):
        return tf.matmul(tf.matmul(x, self.factors_a), self.factors_b)

def polynomial_activation(x, degree=1):
    if degree == 1:
        return x
    elif degree == 2:
        return x * x
    elif degree == 3:
        return x**3
    elif degree ==4:
        return x**4
    else:
        raise ValueError("Invalid degree specified, only 1st, 2nd and 3rd degree po

def resnet_block(x, filters, activation_1, activation_2, rank=None):
    shortcut = x
    x = Dense(filters)(x)
    x = PolynomialLayer()(x)
    x = FunctionCompositionLayer(activation_1, activation_2)(x)

    if rank is not None:
        x = TensorDecompositionLayer(rank)(x)

    x = Dense(filters)(x)
    x = Add()([x, shortcut])
    x = TensorDecompositionLayer(rank)(x)
```

```
    return x

def build_model(input_shape, num_blocks, filters, activation_1, activation_2, rank=
    input_layer = Input(shape=input_shape)
    x = input_layer

    for _ in range(num_blocks):
        x = resnet_block(x, filters, activation_1, activation_2, rank)

    output_layer = Dense(1)(x)
    model = Model(inputs=input_layer, outputs=output_layer)

    return model

input_shape = (1,)
num_blocks = 3
filters = 16
# activation_1 = tf.keras.activations.relu
activation_1 = lambda x: polynomial_activation(x, degree=2)
#activation_2 = tf.keras.activations.linear
activation_2 = lambda x: polynomial_activation(x, degree=1)

rank = 2

model = build_model(input_shape, num_blocks, filters, activation_1, activation_2, r
optimizer = Adam(learning_rate=0.00001) # Reduce learning rate
model.compile(optimizer='adam', loss='mse')
```

## Both architectures have 10,000 samples in range -10, 10

## The main difference is that Pi net requires 200 + epochs to train

## Polinomial Network requires less than 200 epochs and the the loss function is smaller

Loss Pi Net 0.00023438122298102826 Loss Polinomial Func 2.0201952793286182e-05

In [ ]:
```
import numpy as np

np.random.seed(42)
n_samples = 10000
#lower_bound = -2 * np.pi
#upper_bound = 2 * np.pi
lower_bound = -10
upper_bound = 10

X = np.random.uniform(lower_bound, upper_bound, size=(n_samples, 1))
y = np.cos(X)

from sklearn.model_selection import train_test_split

X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state

batch_size = 64
epochs = 200
```

```
history = model.fit(X_train, y_train,
                    batch_size=batch_size,
                    epochs=epochs,
                    verbose=1,
                    validation_data=(X_val, y_val))
```

```
Epoch 1/200
125/125 [==============================] - 2s 3ms/step - loss: 0.5207 - val_loss: 0.
5268
Epoch 2/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4919 - val_loss: 0.
5132
Epoch 3/200
125/125 [==============================] - 0s 2ms/step - loss: 0.4825 - val_loss: 0.
4455
Epoch 4/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4210 - val_loss: 0.
3910
Epoch 5/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3647 - val_loss: 0.
4673
Epoch 6/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3381 - val_loss: 0.
3112
Epoch 7/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3579 - val_loss: 0.
4811
Epoch 8/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3864 - val_loss: 0.
3199
Epoch 9/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3025 - val_loss: 0.
2975
Epoch 10/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3042 - val_loss: 0.
3102
Epoch 11/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2823 - val_loss: 0.
2785
Epoch 12/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2801 - val_loss: 0.
2898
Epoch 13/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2775 - val_loss: 0.
2795
Epoch 14/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2779 - val_loss: 0.
2792
Epoch 15/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2721 - val_loss: 0.
2770
Epoch 16/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2742 - val_loss: 0.
2808
Epoch 17/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2766 - val_loss: 0.
3229
Epoch 18/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2790 - val_loss: 0.
2951
Epoch 19/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2767 - val_loss: 0.
```

```
2800
Epoch 20/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2771 - val_loss: 0.
2746
Epoch 21/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2822 - val_loss: 0.
2778
Epoch 22/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2759 - val_loss: 0.
3046
Epoch 23/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2703 - val_loss: 0.
2728
Epoch 24/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2708 - val_loss: 0.
2734
Epoch 25/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2679 - val_loss: 0.
2761
Epoch 26/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2822 - val_loss: 0.
2844
Epoch 27/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2691 - val_loss: 0.
2803
Epoch 28/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2781 - val_loss: 0.
2708
Epoch 29/200
125/125 [==============================] - 0s 2ms/step - loss: 0.2649 - val_loss: 0.
2706
Epoch 30/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2652 - val_loss: 0.
2724
Epoch 31/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2685 - val_loss: 0.
2790
Epoch 32/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2664 - val_loss: 0.
2695
Epoch 33/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2570 - val_loss: 0.
2581
Epoch 34/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2514 - val_loss: 0.
2343
Epoch 35/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2486 - val_loss: 0.
2264
Epoch 36/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2037 - val_loss: 0.
1800
Epoch 37/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1724 - val_loss: 0.
1592
Epoch 38/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.5231 - val_loss: 0.
5275
Epoch 39/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5187 - val_loss: 0.
5043
Epoch 40/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4454 - val_loss: 0.
4243
Epoch 41/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4263 - val_loss: 0.
6313
Epoch 42/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5296 - val_loss: 0.
5291
Epoch 43/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5215 - val_loss: 0.
5278
Epoch 44/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5207 - val_loss: 0.
5272
Epoch 45/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5203 - val_loss: 0.
5269
Epoch 46/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5202 - val_loss: 0.
5268
Epoch 47/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5200 - val_loss: 0.
5266
Epoch 48/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5199 - val_loss: 0.
5265
Epoch 49/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5196 - val_loss: 0.
5259
Epoch 50/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5185 - val_loss: 0.
5237
Epoch 51/200
125/125 [==============================] - 0s 1ms/step - loss: 0.5064 - val_loss: 0.
4795
Epoch 52/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4340 - val_loss: 0.
4114
Epoch 53/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3962 - val_loss: 0.
3777
Epoch 54/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3825 - val_loss: 0.
3956
Epoch 55/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4023 - val_loss: 0.
4039
Epoch 56/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3931 - val_loss: 0.
4693
```

```
Epoch 57/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3819 - val_loss: 0.
3453
Epoch 58/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3276 - val_loss: 0.
2982
Epoch 59/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2712 - val_loss: 0.
2696
Epoch 60/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2815 - val_loss: 0.
2662
Epoch 61/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2551 - val_loss: 0.
2579
Epoch 62/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3185 - val_loss: 0.
4006
Epoch 63/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3140 - val_loss: 0.
2709
Epoch 64/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2441 - val_loss: 0.
2372
Epoch 65/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2266 - val_loss: 0.
2118
Epoch 66/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1874 - val_loss: 0.
1767
Epoch 67/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1927 - val_loss: 0.
1644
Epoch 68/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2172 - val_loss: 0.
2009
Epoch 69/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1862 - val_loss: 0.
1859
Epoch 70/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1755 - val_loss: 0.
1755
Epoch 71/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1628 - val_loss: 0.
2211
Epoch 72/200
125/125 [==============================] - 0s 1ms/step - loss: 0.4846 - val_loss: 0.
4021
Epoch 73/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2894 - val_loss: 0.
1816
Epoch 74/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1714 - val_loss: 0.
1692
Epoch 75/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1664 - val_loss: 0.
```

```
1663
Epoch 76/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1632 - val_loss: 0.
1618
Epoch 77/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1462 - val_loss: 0.
1364
Epoch 78/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1172 - val_loss: 0.
1290
Epoch 79/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0724 - val_loss: 0.
0682
Epoch 80/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0738 - val_loss: 0.
0641
Epoch 81/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0636 - val_loss: 0.
0582
Epoch 82/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0697 - val_loss: 0.
0614
Epoch 83/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0630 - val_loss: 0.
0624
Epoch 84/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0623 - val_loss: 0.
0555
Epoch 85/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0595 - val_loss: 0.
0562
Epoch 86/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0582 - val_loss: 0.
0557
Epoch 87/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0564 - val_loss: 0.
0557
Epoch 88/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0588 - val_loss: 0.
0545
Epoch 89/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0571 - val_loss: 0.
0534
Epoch 90/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0545 - val_loss: 0.
0504
Epoch 91/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0589 - val_loss: 0.
1444
Epoch 92/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0609 - val_loss: 0.
0523
Epoch 93/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0488 - val_loss: 0.
0440
Epoch 94/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.0431 - val_loss: 0.
0404
Epoch 95/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0406 - val_loss: 0.
0390
Epoch 96/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0407 - val_loss: 0.
0383
Epoch 97/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0396 - val_loss: 0.
0401
Epoch 98/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0399 - val_loss: 0.
0436
Epoch 99/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0385 - val_loss: 0.
0405
Epoch 100/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0381 - val_loss: 0.
0413
Epoch 101/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0396 - val_loss: 0.
0412
Epoch 102/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0389 - val_loss: 0.
0414
Epoch 103/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0378 - val_loss: 0.
0367
Epoch 104/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0379 - val_loss: 0.
0362
Epoch 105/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0376 - val_loss: 0.
0375
Epoch 106/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0397 - val_loss: 0.
0412
Epoch 107/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0376 - val_loss: 0.
0358
Epoch 108/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0379 - val_loss: 0.
0362
Epoch 109/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0378 - val_loss: 0.
0360
Epoch 110/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0381 - val_loss: 0.
0411
Epoch 111/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0368 - val_loss: 0.
0381
Epoch 112/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0372 - val_loss: 0.
0361
```

```
Epoch 113/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0370 - val_loss: 0.
0368
Epoch 114/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0386 - val_loss: 0.
0355
Epoch 115/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0369 - val_loss: 0.
0421
Epoch 116/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0381 - val_loss: 0.
0390
Epoch 117/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0372 - val_loss: 0.
0369
Epoch 118/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0366 - val_loss: 0.
0455
Epoch 119/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0497 - val_loss: 0.
0703
Epoch 120/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0389 - val_loss: 0.
0368
Epoch 121/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0380 - val_loss: 0.
0358
Epoch 122/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0386 - val_loss: 0.
0366
Epoch 123/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0370 - val_loss: 0.
0355
Epoch 124/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0369 - val_loss: 0.
0365
Epoch 125/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0367 - val_loss: 0.
0367
Epoch 126/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0779 - val_loss: 0.
2611
Epoch 127/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0711 - val_loss: 0.
0390
Epoch 128/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0389 - val_loss: 0.
0382
Epoch 129/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0374 - val_loss: 0.
0359
Epoch 130/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0366 - val_loss: 0.
0362
Epoch 131/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0376 - val_loss: 0.
```

```
0374
Epoch 132/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0366 - val_loss: 0.
0372
Epoch 133/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0364 - val_loss: 0.
0370
Epoch 134/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0370 - val_loss: 0.
0374
Epoch 135/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0372 - val_loss: 0.
0357
Epoch 136/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0370 - val_loss: 0.
0370
Epoch 137/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0366 - val_loss: 0.
0356
Epoch 138/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0394 - val_loss: 0.
0390
Epoch 139/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0393 - val_loss: 0.
0363
Epoch 140/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0366 - val_loss: 0.
0367
Epoch 141/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0369 - val_loss: 0.
0365
Epoch 142/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0371 - val_loss: 0.
0370
Epoch 143/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0373 - val_loss: 0.
0358
Epoch 144/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0372 - val_loss: 0.
0374
Epoch 145/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0369 - val_loss: 0.
0346
Epoch 146/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0357 - val_loss: 0.
0345
Epoch 147/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0332 - val_loss: 0.
0303
Epoch 148/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0180 - val_loss: 0.
0030
Epoch 149/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0042 - val_loss: 4.
6372e-04
Epoch 150/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 3.1775e-04 - val_los
s: 1.8248e-04
Epoch 151/200
125/125 [==============================] - 0s 1ms/step - loss: 2.2909e-04 - val_los
s: 3.2357e-04
Epoch 152/200
125/125 [==============================] - 0s 1ms/step - loss: 1.7770e-04 - val_los
s: 8.8422e-05
Epoch 153/200
125/125 [==============================] - 0s 1ms/step - loss: 1.1869e-04 - val_los
s: 6.8204e-05
Epoch 154/200
125/125 [==============================] - 0s 1ms/step - loss: 7.5283e-05 - val_los
s: 1.6333e-04
Epoch 155/200
125/125 [==============================] - 0s 1ms/step - loss: 1.5687e-04 - val_los
s: 4.0484e-05
Epoch 156/200
125/125 [==============================] - 0s 1ms/step - loss: 7.1478e-05 - val_los
s: 1.2993e-04
Epoch 157/200
125/125 [==============================] - 0s 1ms/step - loss: 4.0263e-05 - val_los
s: 7.2322e-05
Epoch 158/200
125/125 [==============================] - 0s 1ms/step - loss: 5.2199e-05 - val_los
s: 2.3157e-05
Epoch 159/200
125/125 [==============================] - 0s 1ms/step - loss: 6.9700e-05 - val_los
s: 6.4906e-05
Epoch 160/200
125/125 [==============================] - 0s 1ms/step - loss: 2.4629e-04 - val_los
s: 1.0639e-04
Epoch 161/200
125/125 [==============================] - 0s 1ms/step - loss: 1.8494e-05 - val_los
s: 1.2270e-05
Epoch 162/200
125/125 [==============================] - 0s 1ms/step - loss: 2.1198e-05 - val_los
s: 1.6376e-04
Epoch 163/200
125/125 [==============================] - 0s 1ms/step - loss: 5.6230e-05 - val_los
s: 9.6865e-06
Epoch 164/200
125/125 [==============================] - 0s 1ms/step - loss: 3.7163e-05 - val_los
s: 9.4179e-06
Epoch 165/200
125/125 [==============================] - 0s 1ms/step - loss: 1.9707e-05 - val_los
s: 6.3048e-06
Epoch 166/200
125/125 [==============================] - 0s 1ms/step - loss: 3.9989e-05 - val_los
s: 9.7909e-05
Epoch 167/200
125/125 [==============================] - 0s 1ms/step - loss: 6.8755e-05 - val_los
s: 4.2429e-05
Epoch 168/200
125/125 [==============================] - 0s 1ms/step - loss: 1.7283e-05 - val_los
s: 7.8539e-05
```

```
Epoch 169/200
125/125 [==============================] - 0s 1ms/step - loss: 1.1887e-04 - val_los
s: 4.2889e-05
Epoch 170/200
125/125 [==============================] - 0s 1ms/step - loss: 3.2310e-04 - val_los
s: 0.0014
Epoch 171/200
125/125 [==============================] - 0s 1ms/step - loss: 4.4768e-04 - val_los
s: 2.7516e-05
Epoch 172/200
125/125 [==============================] - 0s 1ms/step - loss: 2.2458e-05 - val_los
s: 2.1761e-05
Epoch 173/200
125/125 [==============================] - 0s 1ms/step - loss: 7.6109e-05 - val_los
s: 1.4129e-04
Epoch 174/200
125/125 [==============================] - 0s 1ms/step - loss: 6.2462e-04 - val_los
s: 0.0014
Epoch 175/200
125/125 [==============================] - 0s 1ms/step - loss: 8.7466e-04 - val_los
s: 4.1787e-05
Epoch 176/200
125/125 [==============================] - 0s 1ms/step - loss: 4.8425e-05 - val_los
s: 2.5119e-05
Epoch 177/200
125/125 [==============================] - 0s 1ms/step - loss: 1.7355e-05 - val_los
s: 2.0007e-05
Epoch 178/200
125/125 [==============================] - 0s 1ms/step - loss: 3.7629e-05 - val_los
s: 2.8122e-05
Epoch 179/200
125/125 [==============================] - 0s 1ms/step - loss: 8.8152e-05 - val_los
s: 2.3053e-05
Epoch 180/200
125/125 [==============================] - 0s 1ms/step - loss: 2.4955e-05 - val_los
s: 4.0557e-05
Epoch 181/200
125/125 [==============================] - 0s 1ms/step - loss: 8.0445e-05 - val_los
s: 1.2571e-04
Epoch 182/200
125/125 [==============================] - 0s 1ms/step - loss: 9.4484e-04 - val_los
s: 2.7373e-04
Epoch 183/200
125/125 [==============================] - 0s 1ms/step - loss: 5.1645e-05 - val_los
s: 3.1514e-05
Epoch 184/200
125/125 [==============================] - 0s 1ms/step - loss: 1.7592e-04 - val_los
s: 0.0020
Epoch 185/200
125/125 [==============================] - 0s 1ms/step - loss: 7.3331e-04 - val_los
s: 1.0835e-04
Epoch 186/200
125/125 [==============================] - 0s 1ms/step - loss: 3.1208e-05 - val_los
s: 3.3425e-05
Epoch 187/200
125/125 [==============================] - 0s 1ms/step - loss: 2.5024e-05 - val_los
```

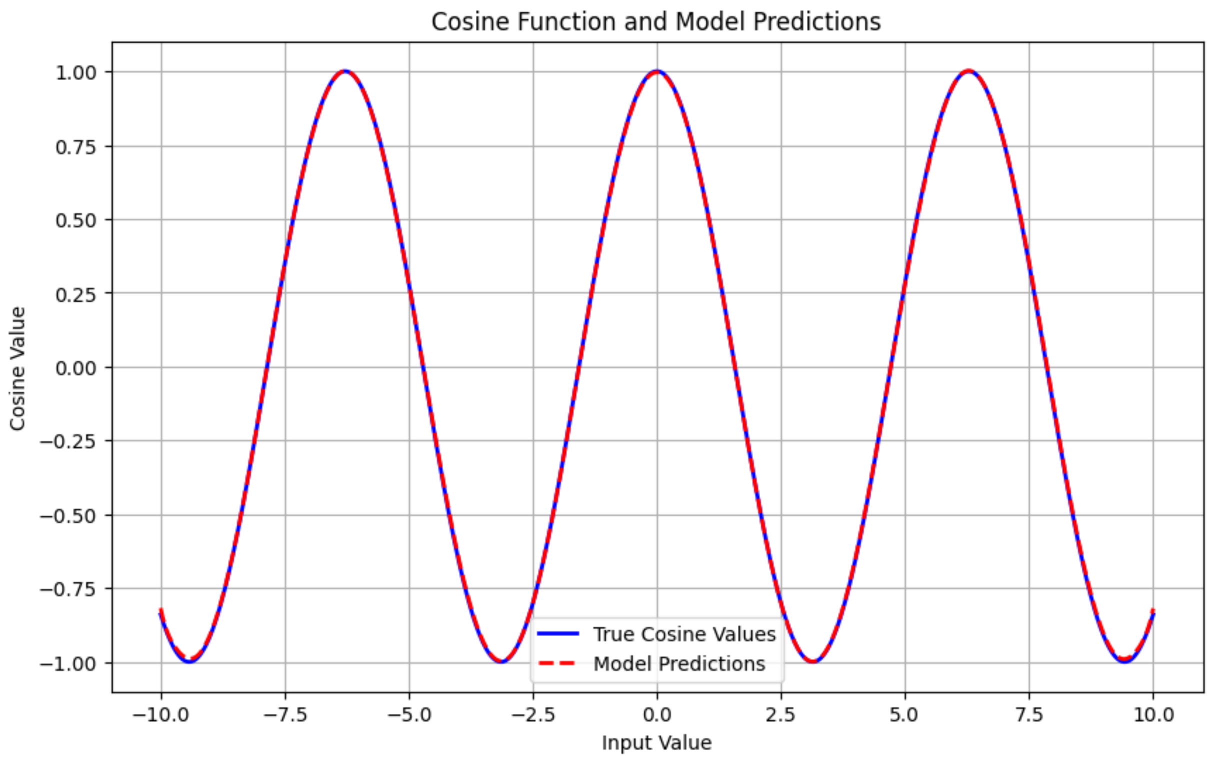
```
s: 7.3417e-06
Epoch 188/200
125/125 [==============================] - 0s 1ms/step - loss: 1.1051e-05 - val_los
s: 1.7460e-05
Epoch 189/200
125/125 [==============================] - 0s 1ms/step - loss: 1.4522e-04 - val_los
s: 2.1510e-04
Epoch 190/200
125/125 [==============================] - 0s 1ms/step - loss: 2.6239e-04 - val_los
s: 1.2237e-04
Epoch 191/200
125/125 [==============================] - 0s 1ms/step - loss: 4.0673e-04 - val_los
s: 2.6642e-04
Epoch 192/200
125/125 [==============================] - 0s 1ms/step - loss: 4.8035e-05 - val_los
s: 1.4864e-05
Epoch 193/200
125/125 [==============================] - 0s 1ms/step - loss: 8.6267e-05 - val_los
s: 1.7694e-04
Epoch 194/200
125/125 [==============================] - 0s 1ms/step - loss: 1.2433e-04 - val_los
s: 7.4939e-06
Epoch 195/200
125/125 [==============================] - 0s 1ms/step - loss: 6.7194e-05 - val_los
s: 5.7057e-05
Epoch 196/200
125/125 [==============================] - 0s 1ms/step - loss: 1.4589e-04 - val_los
s: 2.0333e-05
Epoch 197/200
125/125 [==============================] - 0s 1ms/step - loss: 1.0528e-04 - val_los
s: 4.3249e-05
Epoch 198/200
125/125 [==============================] - 0s 1ms/step - loss: 3.2426e-04 - val_los
s: 7.3892e-06
Epoch 199/200
125/125 [==============================] - 0s 1ms/step - loss: 1.0935e-05 - val_los
s: 6.6592e-05
Epoch 200/200
125/125 [==============================] - 0s 1ms/step - loss: 6.4278e-05 - val_los
s: 2.0202e-05
```

In [ ]:

```
val_loss = model.evaluate(X_val, y_val, verbose=0)
print(f"Validation loss: {val_loss}")

import matplotlib.pyplot as plt

num_test_samples = 1000
X_test = np.linspace(lower_bound, upper_bound, num=num_test_samples).reshape(-1, 1)
y_true = np.cos(X_test)
y_pred = model.predict(X_test)

plt.figure(figsize=(10, 6))
plt.plot(X_test, y_true, label='True Cosine Values', color='b', linewidth=2)
plt.plot(X_test, y_pred, label='Model Predictions', color='r', linestyle='--', line
plt.xlabel('Input Value')
plt.ylabel('Cosine Value')
```

```
plt.title('Cosine Function and Model Predictions')
plt.legend()
plt.grid()
plt.show()
```

```
Validation loss: 2.0201952793286182e-05
32/32 [==============================] - 0s 640us/step
```




### Less epochs means less training time, but in this case with higher accuracy

# Differences

- Pi net structure has a resnet architecture with a polinomial activation of degree 3
- It has a decomposition layer that helps keeping the tensors within a range and does not explodes the learning.

- Polinomial structure also has a resnet architecture but it has a layer in the form of a polinomial a + bx + cx*x
- This also has a Tensor Decomposition layer that helps with the learning
- It has two activation functions where one is X*X and the second one is just X

# PI NET WITH POLI LAYER

In [ ]:
```
import tensorflow as tf
from tensorflow.keras.layers import Input, Add, Dense, Layer
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam

class PolynomialLayer(Layer):
    def __init__(self, **kwargs):
        super(PolynomialLayer, self).__init__(**kwargs)

    def build(self, input_shape):
        self.a = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        self.b = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        self.c = self.add_weight(shape=(input_shape[-1],),
                                 initializer='random_normal',
                                 trainable=True)
        super(PolynomialLayer, self).build(input_shape)

    def call(self, x):
        return self.a + self.b * x + self.c * x**2

def polynomial_activation(x, degree=1):
    if degree == 1:
        return x
    elif degree == 2:
        return x * x
    elif degree == 3:
        return x**3
    else:
        raise ValueError("Invalid degree specified, only 1st, 2nd and 3rd degree po

class TensorDecompositionLayer(Layer):
    def __init__(self, rank, **kwargs):
        self.rank = rank
        super(TensorDecompositionLayer, self).__init__(**kwargs)

    def build(self, input_shape):
        self.factors_a = self.add_weight(shape=(input_shape[-1], self.rank),
                                         initializer='random_normal',
                                         trainable=True)
        self.factors_b = self.add_weight(shape=(self.rank, input_shape[-1]),
                                         initializer='random_normal',
                                         trainable=True)
        super(TensorDecompositionLayer, self).build(input_shape)

    def call(self, x):
        return tf.matmul(tf.matmul(x, self.factors_a), self.factors_b)

def resnet_block(x, filters, activation_degree, rank=None):
    shortcut = x
    x = Dense(filters)(x)
    x = PolynomialLayer()(x)
```

```
    x = tf.keras.layers.Lambda(lambda y: polynomial_activation(y, degree=activation

    if rank is not None:
        x = TensorDecompositionLayer(rank)(x)

    x = Dense(filters)(x)
    x = Add()([x, shortcut])
    return x

def build_model(input_shape, num_blocks, filters, activation_degree, rank=None):
    input_layer = Input(shape=input_shape)
    x = input_layer

    for _ in range(num_blocks):
        x = resnet_block(x, filters, activation_degree, rank)

    output_layer = Dense(1)(x)
    model = Model(inputs=input_layer, outputs=output_layer)

    return model

input_shape = (1,)
num_blocks = 3
filters = 16
activation_degree = 3  # Change to 1 for 1st degree polynomial, 2 for 2nd degree, a
rank = 4  # Tensor decomposition rank, set to None if you don't want to use tensor

model = build_model(input_shape, num_blocks, filters, activation_degree, rank)
model.compile(optimizer='adam', loss='mse')
```

In [ ]:

```
import numpy as np

np.random.seed(42)
n_samples = 10000
# lower_bound = -2 * np.pi
# upper_bound = 2 * np.pi
lower_bound = -10
upper_bound = 10

X = np.random.uniform(lower_bound, upper_bound, size=(n_samples, 1))
y = np.cos(X)

from sklearn.model_selection import train_test_split

X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state

batch_size = 64
epochs = 200

history = model.fit(X_train, y_train,
                    batch_size=batch_size,
                    epochs=epochs,
                    verbose=1,
                    validation_data=(X_val, y_val))
```

```
Epoch 1/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2288 - val_loss: 0.
2514
Epoch 2/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2138 - val_loss: 0.
2169
Epoch 3/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2127 - val_loss: 0.
3209
Epoch 4/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2783 - val_loss: 0.
2117
Epoch 5/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2079 - val_loss: 0.
1890
Epoch 6/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1944 - val_loss: 0.
1887
Epoch 7/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2023 - val_loss: 0.
1839
Epoch 8/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2176 - val_loss: 0.
1885
Epoch 9/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1850 - val_loss: 0.
1752
Epoch 10/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1954 - val_loss: 0.
1970
Epoch 11/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3762 - val_loss: 0.
2395
Epoch 12/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1942 - val_loss: 0.
1753
Epoch 13/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1733 - val_loss: 0.
1697
Epoch 14/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1680 - val_loss: 0.
1607
Epoch 15/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1626 - val_loss: 0.
1586
Epoch 16/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1711 - val_loss: 0.
1655
Epoch 17/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1614 - val_loss: 0.
1946
Epoch 18/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1565 - val_loss: 0.
1524
Epoch 19/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1649 - val_loss: 0.
```

```
1439
Epoch 20/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1611 - val_loss: 0.
1643
Epoch 21/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1610 - val_loss: 0.
1470
Epoch 22/200
125/125 [==============================] - 0s 1ms/step - loss: 0.3194 - val_loss: 1.
3873
Epoch 23/200
125/125 [==============================] - 0s 1ms/step - loss: 0.2266 - val_loss: 0.
1452
Epoch 24/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1445 - val_loss: 0.
1392
Epoch 25/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1420 - val_loss: 0.
1388
Epoch 26/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1344 - val_loss: 0.
1462
Epoch 27/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1320 - val_loss: 0.
1346
Epoch 28/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1280 - val_loss: 0.
1262
Epoch 29/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1247 - val_loss: 0.
1224
Epoch 30/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1175 - val_loss: 0.
1119
Epoch 31/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1382 - val_loss: 0.
1536
Epoch 32/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1147 - val_loss: 0.
1141
Epoch 33/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1079 - val_loss: 0.
1089
Epoch 34/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1086 - val_loss: 0.
1042
Epoch 35/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1144 - val_loss: 0.
0983
Epoch 36/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1015 - val_loss: 0.
1077
Epoch 37/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1041 - val_loss: 0.
0981
Epoch 38/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.1004 - val_loss: 0.
0946
Epoch 39/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1005 - val_loss: 0.
1038
Epoch 40/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1374 - val_loss: 0.
1477
Epoch 41/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0974 - val_loss: 0.
0907
Epoch 42/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0931 - val_loss: 0.
0938
Epoch 43/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0966 - val_loss: 0.
0940
Epoch 44/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0929 - val_loss: 0.
1029
Epoch 45/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0972 - val_loss: 0.
0891
Epoch 46/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0889 - val_loss: 0.
1214
Epoch 47/200
125/125 [==============================] - 0s 1ms/step - loss: 0.1076 - val_loss: 0.
0897
Epoch 48/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0810 - val_loss: 0.
0841
Epoch 49/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0878 - val_loss: 0.
0755
Epoch 50/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0786 - val_loss: 0.
0745
Epoch 51/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0783 - val_loss: 0.
0710
Epoch 52/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0921 - val_loss: 0.
0904
Epoch 53/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0739 - val_loss: 0.
0817
Epoch 54/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0730 - val_loss: 0.
0706
Epoch 55/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0715 - val_loss: 0.
0614
Epoch 56/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0681 - val_loss: 0.
0658
```

```
Epoch 57/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0938 - val_loss: 0.
1051
Epoch 58/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0693 - val_loss: 0.
0593
Epoch 59/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0601 - val_loss: 0.
0558
Epoch 60/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0619 - val_loss: 0.
0545
Epoch 61/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0729 - val_loss: 0.
0649
Epoch 62/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0560 - val_loss: 0.
2363
Epoch 63/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0850 - val_loss: 0.
0493
Epoch 64/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0495 - val_loss: 0.
0441
Epoch 65/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0596 - val_loss: 0.
0558
Epoch 66/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0464 - val_loss: 0.
0412
Epoch 67/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0432 - val_loss: 0.
0414
Epoch 68/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0500 - val_loss: 0.
0447
Epoch 69/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0373 - val_loss: 0.
0391
Epoch 70/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0440 - val_loss: 0.
0555
Epoch 71/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0478 - val_loss: 0.
0333
Epoch 72/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0359 - val_loss: 0.
0397
Epoch 73/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0349 - val_loss: 0.
0255
Epoch 74/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0309 - val_loss: 0.
0396
Epoch 75/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0430 - val_loss: 0.
```

```
0650
Epoch 76/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0377 - val_loss: 0.
0836
Epoch 77/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0356 - val_loss: 0.
0242
Epoch 78/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0345 - val_loss: 0.
0222
Epoch 79/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0321 - val_loss: 0.
0232
Epoch 80/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0232 - val_loss: 0.
0263
Epoch 81/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0254 - val_loss: 0.
0552
Epoch 82/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0705 - val_loss: 0.
0222
Epoch 83/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0234 - val_loss: 0.
0204
Epoch 84/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0231 - val_loss: 0.
0238
Epoch 85/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0213 - val_loss: 0.
0180
Epoch 86/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0298 - val_loss: 0.
0167
Epoch 87/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0200 - val_loss: 0.
0247
Epoch 88/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0221 - val_loss: 0.
0211
Epoch 89/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0857 - val_loss: 0.
0435
Epoch 90/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0272 - val_loss: 0.
0190
Epoch 91/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0203 - val_loss: 0.
0213
Epoch 92/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0204 - val_loss: 0.
0191
Epoch 93/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0160 - val_loss: 0.
0147
Epoch 94/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.0154 - val_loss: 0.
0116
Epoch 95/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0264 - val_loss: 0.
0150
Epoch 96/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0162 - val_loss: 0.
0118
Epoch 97/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0117 - val_loss: 0.
0136
Epoch 98/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0342 - val_loss: 0.
0125
Epoch 99/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0105 - val_loss: 0.
0095
Epoch 100/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0100 - val_loss: 0.
0096
Epoch 101/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0094 - val_loss: 0.
0077
Epoch 102/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0141 - val_loss: 0.
0193
Epoch 103/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0236 - val_loss: 0.
0100
Epoch 104/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0085 - val_loss: 0.
0074
Epoch 105/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0074 - val_loss: 0.
0075
Epoch 106/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0091 - val_loss: 0.
0103
Epoch 107/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0101 - val_loss: 0.
0130
Epoch 108/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0329 - val_loss: 0.
0089
Epoch 109/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0081 - val_loss: 0.
0075
Epoch 110/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0079 - val_loss: 0.
0063
Epoch 111/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0168 - val_loss: 0.
0071
Epoch 112/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0083 - val_loss: 0.
0093
```

```
Epoch 113/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0072 - val_loss: 0.
0060
Epoch 114/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0076 - val_loss: 0.
0085
Epoch 115/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0070 - val_loss: 0.
0062
Epoch 116/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0114 - val_loss: 0.
0051
Epoch 117/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0097 - val_loss: 0.
0250
Epoch 118/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0085 - val_loss: 0.
0074
Epoch 119/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0076 - val_loss: 0.
0050
Epoch 120/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0086 - val_loss: 0.
0049
Epoch 121/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0053 - val_loss: 0.
0062
Epoch 122/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0098 - val_loss: 0.
0208
Epoch 123/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0086 - val_loss: 0.
0040
Epoch 124/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0055 - val_loss: 0.
0065
Epoch 125/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0127 - val_loss: 0.
0090
Epoch 126/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0060 - val_loss: 0.
0105
Epoch 127/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0091 - val_loss: 0.
0045
Epoch 128/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0056 - val_loss: 0.
0046
Epoch 129/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0056 - val_loss: 0.
0041
Epoch 130/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0044 - val_loss: 0.
0058
Epoch 131/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0384 - val_loss: 0.
```

```
0116
Epoch 132/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0065 - val_loss: 0.
0058
Epoch 133/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0049 - val_loss: 0.
0032
Epoch 134/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0086 - val_loss: 0.
0082
Epoch 135/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0041 - val_loss: 0.
0038
Epoch 136/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0040 - val_loss: 0.
0023
Epoch 137/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0028 - val_loss: 0.
0020
Epoch 138/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0148 - val_loss: 0.
0188
Epoch 139/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0058 - val_loss: 0.
0025
Epoch 140/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0023 - val_loss: 0.
0022
Epoch 141/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0022 - val_loss: 0.
0015
Epoch 142/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0029 - val_loss: 0.
0020
Epoch 143/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0029 - val_loss: 0.
0019
Epoch 144/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0025 - val_loss: 0.
0061
Epoch 145/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0027 - val_loss: 0.
0021
Epoch 146/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0024 - val_loss: 0.
0024
Epoch 147/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0060 - val_loss: 0.
0830
Epoch 148/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0104 - val_loss: 0.
0012
Epoch 149/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0014 - val_loss: 0.
0010
Epoch 150/200
```

```
125/125 [==============================] - 0s 1ms/step - loss: 0.0013 - val_loss: 9.
3371e-04
Epoch 151/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0022 - val_loss: 0.
0068
Epoch 152/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0017 - val_loss: 0.
0018
Epoch 153/200
125/125 [==============================] - 0s 1ms/step - loss: 9.9917e-04 - val_los
s: 7.3135e-04
Epoch 154/200
125/125 [==============================] - 0s 1ms/step - loss: 9.6890e-04 - val_los
s: 0.0021
Epoch 155/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0038 - val_loss: 0.
0039
Epoch 156/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0058 - val_loss: 0.
0019
Epoch 157/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0015 - val_loss: 0.
0012
Epoch 158/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0019 - val_loss: 0.
0013
Epoch 159/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0012 - val_loss: 7.
6297e-04
Epoch 160/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0020 - val_loss: 0.
0015
Epoch 161/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0014 - val_loss: 0.
0014
Epoch 162/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0011 - val_loss: 5.
7941e-04
Epoch 163/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0090 - val_loss: 0.
0088
Epoch 164/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0020 - val_loss: 9.
7794e-04
Epoch 165/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0011 - val_loss: 0.
0029
Epoch 166/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0095 - val_loss: 0.
0167
Epoch 167/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0046 - val_loss: 6.
2490e-04
Epoch 168/200
125/125 [==============================] - 0s 1ms/step - loss: 7.9254e-04 - val_los
s: 6.7904e-04
```

```
Epoch 169/200
125/125 [==============================] - 0s 1ms/step - loss: 7.7004e-04 - val_los
s: 4.7294e-04
Epoch 170/200
125/125 [==============================] - 0s 1ms/step - loss: 5.2595e-04 - val_los
s: 4.5065e-04
Epoch 171/200
125/125 [==============================] - 0s 1ms/step - loss: 4.5274e-04 - val_los
s: 4.9315e-04
Epoch 172/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0114 - val_loss: 0.
0055
Epoch 173/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0013 - val_loss: 9.
5678e-04
Epoch 174/200
125/125 [==============================] - 0s 1ms/step - loss: 5.0358e-04 - val_los
s: 0.0012
Epoch 175/200
125/125 [==============================] - 0s 1ms/step - loss: 7.6646e-04 - val_los
s: 9.8130e-04
Epoch 176/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0014 - val_loss: 0.
0015
Epoch 177/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0012 - val_loss: 8.
2351e-04
Epoch 178/200
125/125 [==============================] - 0s 1ms/step - loss: 5.8861e-04 - val_los
s: 7.1649e-04
Epoch 179/200
125/125 [==============================] - 0s 1ms/step - loss: 6.2896e-04 - val_los
s: 0.0013
Epoch 180/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0055 - val_loss: 0.
0235
Epoch 181/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0067 - val_loss: 0.
0082
Epoch 182/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0031 - val_loss: 7.
4575e-04
Epoch 183/200
125/125 [==============================] - 0s 1ms/step - loss: 4.5805e-04 - val_los
s: 4.8362e-04
Epoch 184/200
125/125 [==============================] - 0s 1ms/step - loss: 5.8883e-04 - val_los
s: 0.0018
Epoch 185/200
125/125 [==============================] - 0s 1ms/step - loss: 7.8210e-04 - val_los
s: 4.6145e-04
Epoch 186/200
125/125 [==============================] - 0s 1ms/step - loss: 3.4047e-04 - val_los
s: 2.3363e-04
Epoch 187/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0014 - val_loss: 3.
```

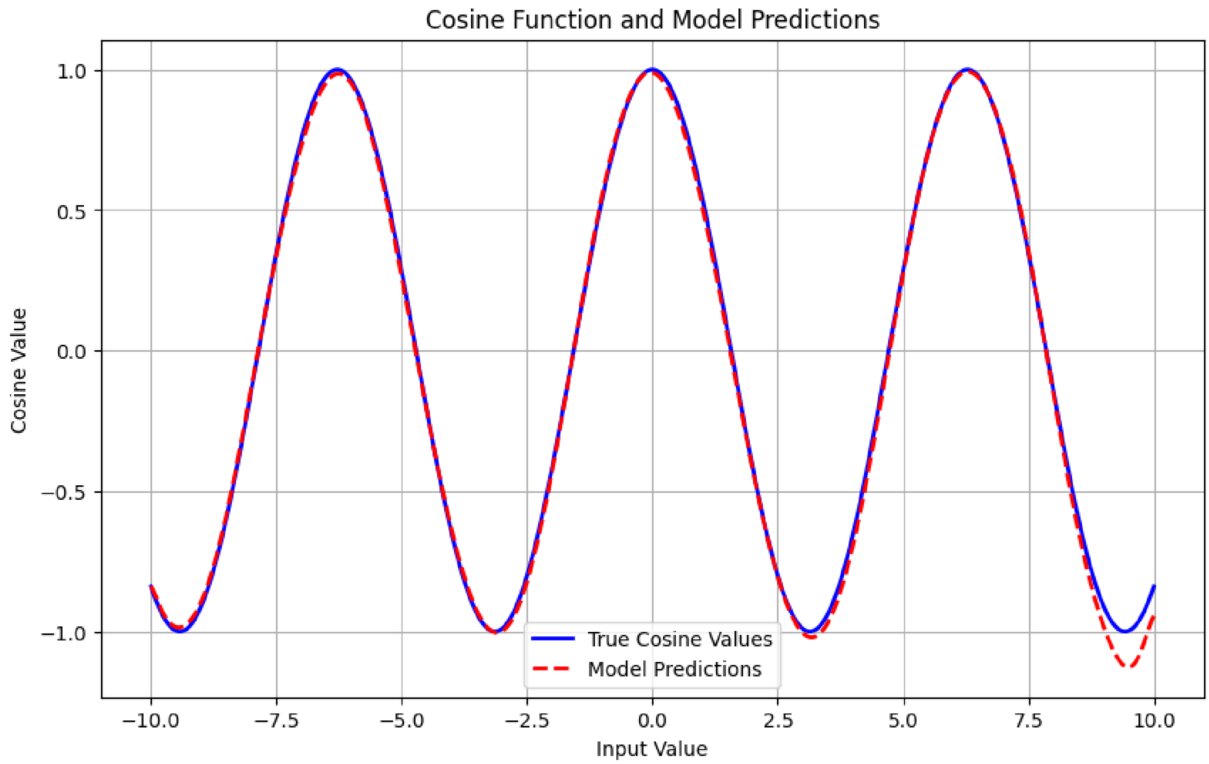
```
8830e-04
Epoch 188/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0011 - val_loss: 5.
7208e-04
Epoch 189/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0053 - val_loss: 0.
0012
Epoch 190/200
125/125 [==============================] - 0s 1ms/step - loss: 8.1463e-04 - val_los
s: 3.3330e-04
Epoch 191/200
125/125 [==============================] - 0s 1ms/step - loss: 3.2481e-04 - val_los
s: 1.9583e-04
Epoch 192/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0034 - val_loss: 0.
0014
Epoch 193/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0019 - val_loss: 4.
3428e-04
Epoch 194/200
125/125 [==============================] - 0s 1ms/step - loss: 4.0004e-04 - val_los
s: 8.2877e-04
Epoch 195/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0035 - val_loss: 0.
0028
Epoch 196/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0025 - val_loss: 2.
5500e-04
Epoch 197/200
125/125 [==============================] - 0s 1ms/step - loss: 4.6973e-04 - val_los
s: 4.6321e-04
Epoch 198/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0011 - val_loss: 0.
0026
Epoch 199/200
125/125 [==============================] - 0s 1ms/step - loss: 0.0014 - val_loss: 3.
6140e-04
Epoch 200/200
125/125 [==============================] - 0s 1ms/step - loss: 4.8285e-04 - val_los
s: 9.5041e-04
```

In [ ]:

```
val_loss = model.evaluate(X_val, y_val, verbose=0)
print(f"Validation loss: {val_loss}")

import matplotlib.pyplot as plt

num_test_samples = 1000
X_test = np.linspace(-10, 10, num=num_test_samples).reshape(-1, 1)
y_true = np.cos(X_test)
y_pred = model.predict(X_test)

plt.figure(figsize=(10, 6))
plt.plot(X_test, y_true, label='True Cosine Values', color='b', linewidth=2)
plt.plot(X_test, y_pred, label='Model Predictions', color='r', linestyle='--', line
plt.xlabel('Input Value')
plt.ylabel('Cosine Value')
```

```
plt.title('Cosine Function and Model Predictions')
plt.legend()
plt.grid()
plt.show()
```

```
Validation loss: 0.0009504059562459588
32/32 [==============================] - 0s 624us/step
```




```
In [ ]:
```