```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.datasets import cifar10
import matplotlib.pyplot as plt

# Load dataset
(x_train, y_train), (x_test, y_test) = cifar10.load_data()

# Normalize inputs
x_train = x_train.astype('float32') / 255.0
x_test = x_test.astype('float32') / 255.0

# One-hot encode labels
y_train = to_categorical(y_train, 10)
y_test = to_categorical(y_test, 10)

# Create validation set
x_val = x_train[-5000:]
y_val = y_train[-5000:]
x_train = x_train[:-5000]
y_train = y_train[:-5000]

# Build CNN model
model = Sequential()
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(10, activation='softmax'))

# Compile model
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

# Train model
history = model.fit(x_train, y_train,
                    epochs=20,
                    batch_size=64,
                    validation_data=(x_val, y_val))

# Evaluate
test_loss, test_acc = model.evaluate(x_test, y_test, verbose=0)
print(f"Test Accuracy: {test_acc:.4f}")
print(f"Test Loss: {test_loss:.4f}")

# Plot performance
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plt.plot(history.history['accuracy'], label='Train Acc')
plt.plot(history.history['val_accuracy'], label='Val Acc')
plt.title('Accuracy over Epochs')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend()

plt.subplot(1, 2, 2)
plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.title('Loss over Epochs')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()

plt.tight_layout()
plt.show()
```

```
/usr/local/lib/python3.11/dist-packages/keras/src/layers/convolutional/base_conv.py:107: UserWarning: Do not pass an `input_shape`/`in
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
Epoch 1/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 57s 79ms/step - accuracy: 0.3001 - loss: 1.8850 - val_accuracy: 0.5198 - val_loss: 1.3570
Epoch 2/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 81s 77ms/step - accuracy: 0.4925 - loss: 1.4156 - val_accuracy: 0.5798 - val_loss: 1.1823
Epoch 3/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 55s 79ms/step - accuracy: 0.5464 - loss: 1.2626 - val_accuracy: 0.6156 - val_loss: 1.1000
Epoch 4/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 55s 78ms/step - accuracy: 0.5901 - loss: 1.1637 - val_accuracy: 0.6460 - val_loss: 1.0230
Epoch 5/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 81s 76ms/step - accuracy: 0.6142 - loss: 1.0974 - val_accuracy: 0.6344 - val_loss: 1.0392
Epoch 6/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 83s 77ms/step - accuracy: 0.6363 - loss: 1.0448 - val_accuracy: 0.6824 - val_loss: 0.9337
Epoch 7/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 81s 76ms/step - accuracy: 0.6477 - loss: 0.9972 - val_accuracy: 0.6816 - val_loss: 0.9094
Epoch 8/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 54s 76ms/step - accuracy: 0.6567 - loss: 0.9685 - val_accuracy: 0.6760 - val_loss: 0.9124
Epoch 9/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 81s 76ms/step - accuracy: 0.6710 - loss: 0.9191 - val_accuracy: 0.6934 - val_loss: 0.8666
Epoch 10/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 83s 77ms/step - accuracy: 0.6857 - loss: 0.8825 - val_accuracy: 0.7002 - val_loss: 0.8733
Epoch 11/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 82s 77ms/step - accuracy: 0.7023 - loss: 0.8493 - val_accuracy: 0.6908 - val_loss: 0.8828
Epoch 12/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 84s 80ms/step - accuracy: 0.7045 - loss: 0.8259 - val_accuracy: 0.7050 - val_loss: 0.8645
Epoch 13/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 78s 75ms/step - accuracy: 0.7167 - loss: 0.8008 - val_accuracy: 0.7042 - val_loss: 0.8630
Epoch 14/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 83s 77ms/step - accuracy: 0.7216 - loss: 0.7821 - val_accuracy: 0.7076 - val_loss: 0.8566
Epoch 15/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 82s 78ms/step - accuracy: 0.7272 - loss: 0.7633 - val_accuracy: 0.7018 - val_loss: 0.8957
Epoch 16/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 55s 78ms/step - accuracy: 0.7376 - loss: 0.7314 - val_accuracy: 0.7198 - val_loss: 0.8407
Epoch 17/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 54s 76ms/step - accuracy: 0.7457 - loss: 0.7125 - val_accuracy: 0.7016 - val_loss: 0.8751
Epoch 18/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 53s 76ms/step - accuracy: 0.7507 - loss: 0.7019 - val_accuracy: 0.7106 - val_loss: 0.8668
Epoch 19/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 53s 76ms/step - accuracy: 0.7504 - loss: 0.6846 - val_accuracy: 0.7174 - val_loss: 0.8546
Epoch 20/20
704/704 ━━━━━━━━━━━━━━━━━━━━ 83s 77ms/step - accuracy: 0.7594 - loss: 0.6726 - val_accuracy: 0.7132 - val_loss: 0.8667
Test Accuracy: 0.7126
Test Loss: 0.8721
```