```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split

# Generate synthetic non-linear data
np.random.seed(0)
X = np.sort(5 * np.random.rand(100, 1), axis=0)
y = np.sin(X).ravel() + np.random.normal(0, 0.2, X.shape[0])  # Non-
linear function with noise

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Linear Regression
lin_reg = LinearRegression()
lin_reg.fit(X_train, y_train)
y_pred_lin = lin_reg.predict(X_test)

# Polynomial Regression (degree = 4)
poly = PolynomialFeatures(degree=4)
X_train_poly = poly.fit_transform(X_train)
X_test_poly = poly.transform(X_test)

poly_reg = LinearRegression()
poly_reg.fit(X_train_poly, y_train)
y_pred_poly = poly_reg.predict(X_test_poly)

# Evaluation
print("Linear Regression R² Score:", r2_score(y_test, y_pred_lin))
print("Polynomial Regression R² Score:", r2_score(y_test, y_pred_poly))

# Visualization
X_range = np.linspace(0, 5, 100).reshape(-1, 1)
y_range_lin = lin_reg.predict(X_range)
y_range_poly = poly_reg.predict(poly.transform(X_range))

plt.scatter(X, y, color='blue', label='Actual Data')
plt.plot(X_range, y_range_lin, color='red', label='Linear Regression')
plt.plot(X_range, y_range_poly, color='green', label='Polynomial
Regression (deg=4)')
plt.title("Comparison of Linear vs Polynomial Regression")
plt.xlabel("X")
plt.ylabel("y")
plt.legend()
plt.show()
```