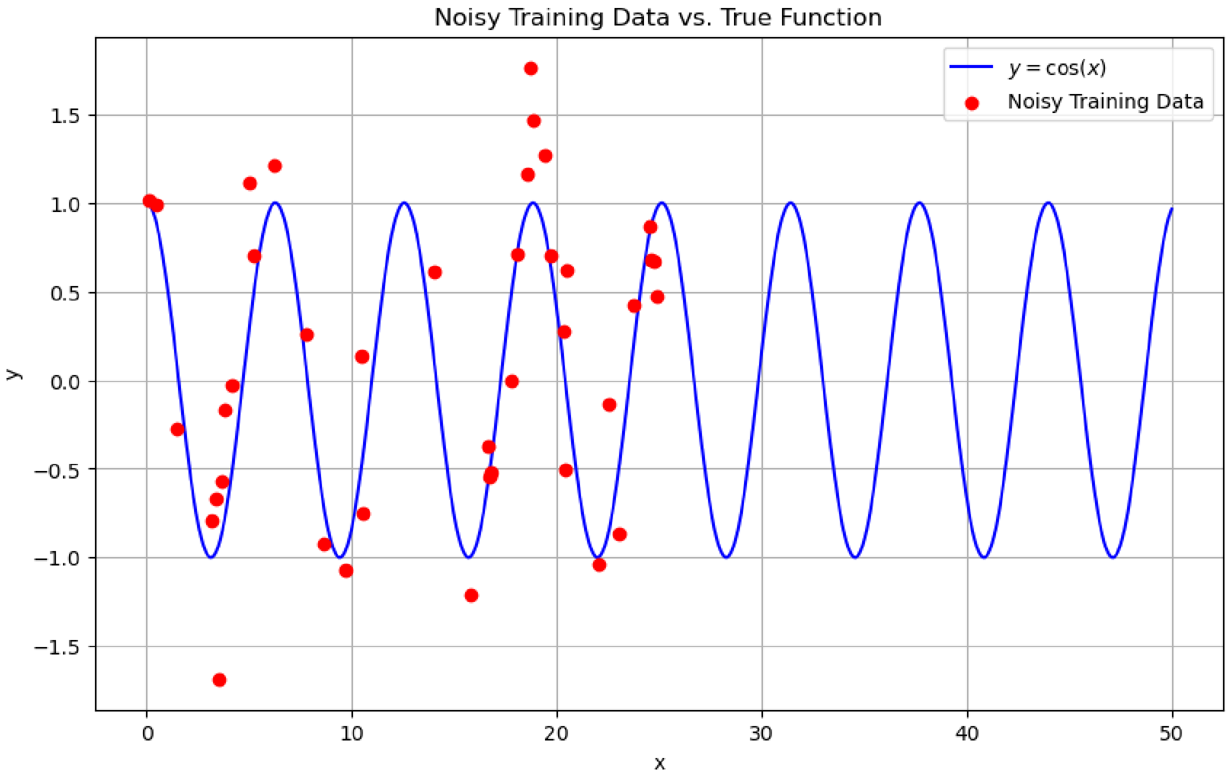
```python
# Import Dependencies
import numpy as np
import matplotlib.pyplot as plt
from sklearn.gaussian_process import GaussianProcessRegressor
from sklearn.gaussian_process.kernels import RBF, WhiteKernel,
ExpSineSquared, ConstantKernel
from sklearn.kernel_ridge import KernelRidge
```

Midterm 1 Project, Problem 2-1

```python
# Generate 1000 evenly spaced x data points in the range [0, 50]
x = np.linspace(0, 50, 1000)

# True function y = cos(x)
y_true = np.cos(x)

# Randomly select 40 points from the first 500 data points (i.e., x ∈
[0, 25])
np.random.seed(42)  # Ensure reproducibility
indices = np.random.choice(np.arange(500), size=40, replace=False)

# Add i.i.d. random noise (mean 0, variance 0.16) to the 40 selected
points
noise = np.random.normal(0, np.sqrt(0.16), size=40)
x_train = x[indices]  # Select 40 x points
y_train_noisy = y_true[indices] + noise  # Add noise to the
corresponding y_true points

# Plot the results
plt.figure(figsize=(10, 6))

# Plot the true cos(x) function
plt.plot(x, y_true, label=r"$y = \cos(x)$", color='blue')

# Plot the noisy training data points
plt.scatter(x_train, y_train_noisy, label="Noisy Training Data",
color='red', zorder=5)

# Set labels, title, and legend
plt.title("Noisy Training Data vs. True Function")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.grid(True)

# Show the plot
plt.show()
```


Noisy Training Data vs. True Function
$y = \cos(x)$
Noisy Training Data
y
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
10
20
30
40
50
x