GPredict: Gaussian Process Regression

Kevin Mota da Costa

January 12, 2026

Abstract

This document presents the theoretical background and implementation details of *GPredict*, a Python framework for Gaussian Process (GP) regression. The project emphasizes explicit kernel and mean function definitions, uncertainty propagation, and visualization of prior and posterior distributions. Example scripts demonstrate the workflow, including prior and posterior GP plots.

1 Introduction

Gaussian Processes provide a flexible, non-parametric framework for regression. *GPredict* demonstrates:

- Flexible mean and kernel functions for GP modeling,
- Exact GP inference with predictive mean and covariance,
- Sampling from the posterior for visualization,
- Modular Python implementation suitable for illustrative examples.

All components are implemented explicitly to highlight the underlying mathematical and computational principles.

2 Gaussian Process Regression

Given training inputs $X = \{x_i\}_{i=1}^N$ and targets $y = \{y_i\}_{i=1}^N$, a Gaussian Process is defined as a distribution over functions $f(x)$:

$$f(x) \sim \mathcal{GP}\big(m(x), k(x, x')\big), \tag{1}$$

where $m(x)$ is the mean function and $k(x, x')$ the kernel (covariance) function. For observed data y_i with uncertainties δy_i, the likelihood is

$$p(y|f) = \prod_{i=1}^{N} \mathcal{N}\big(y_i | f(x_i), \delta y_i^2\big). \tag{2}$$

2.1 Mean Functions

GPredict supports:

- Constant mean: $m(x) = b$,
- Linear mean: $m(x) = wx + b$.

2.2 Kernel Functions

Supported kernels include:

- Radial Basis Function (RBF):

$$k_{\mathrm{RBF}}(x, x') = \sigma_f^2 \exp\left(-\frac{(x - x')^2}{2\ell^2}\right)$$

- Matern (with $\nu = 3/2$):

$$k_{\mathrm{Matern}}(x, x') = \sigma_f^2 \left(1 + \frac{\sqrt{3}|x - x'|}{\ell}\right) \exp\left(-\frac{\sqrt{3}|x - x'|}{\ell}\right)$$

3 Observational Model

For each observation, *GPredict* allows an associated uncertainty δy_i. Training data are represented as $(x_i, y_i, \delta y_i)$, and predictive distributions propagate these uncertainties:

$$y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \delta y_i^2). \tag{3}$$

4 Inference and Prediction

4.1 Posterior Distribution

Given training data (X, y) and test points X_*, the predictive mean μ_* and covariance Σ_* are:

$$\mu_* = m(X_*) + K(X_*, X)K(X, X)^{-1}(y - m(X)), \tag{4}$$
$$\Sigma_* = K(X_*, X_*) - K(X_*, X)K(X, X)^{-1}K(X, X_*), \tag{5}$$

where $K(A, B)$ denotes the kernel matrix evaluated at points A and B.

4.2 Sampling from the Posterior

Posterior samples are drawn as:

$$f_* \sim \mathcal{N}(\mu_*, \Sigma_*), \tag{6}$$

allowing visualization of plausible functions consistent with the observed data.

5 Visualization

GPredict provides functions to plot:

- Training points with uncertainty bars,
- Predictive mean with 1-sigma confidence intervals,
- Posterior samples from the GP,
- Prior function samples to illustrate the GP prior distribution.

Example prior and posterior plots:

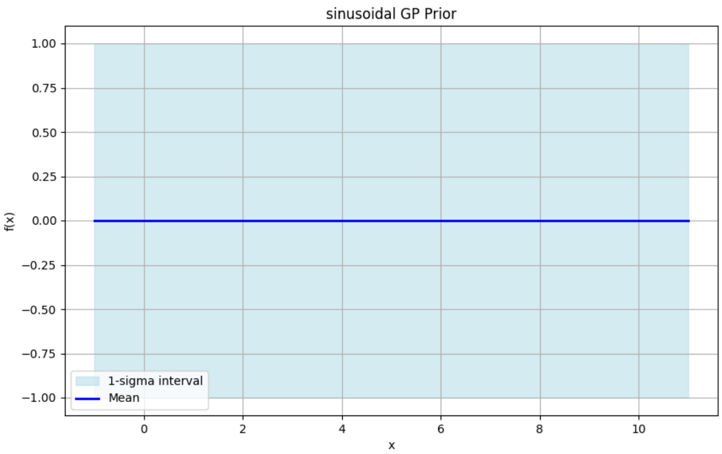

Figure 1: GP prior for sinusoidal example.

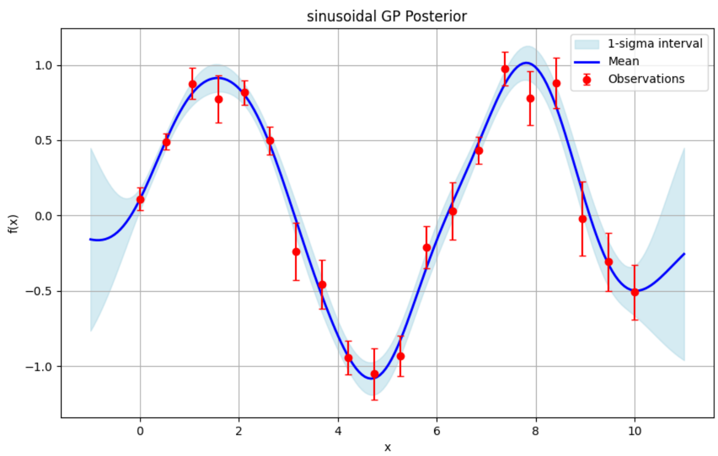

Figure 2: GP posterior for sinusoidal example, showing mean and uncertainty.

6 Implementation Overview

The project is organized into the following modules:

- `gp.py`: GP fitting, prediction, and posterior sampling,
- `kernels.py`: kernel function definitions,
- `means.py`: mean function definitions,
- `utils.py`: plotting, saving predictions,
- `examples.py`: example scripts demonstrating GP functionality,
- `main.py`: entry point to run experiments and generate plots.

7 Conclusion

GPredict provides a technical framework for understanding and implementing Gaussian Process regression. By explicitly computing kernel matrices, predictive means, and covariances, and visualizing prior and posterior distributions, the project illustrates both the theoretical and practical aspects of GP modeling.

8 References

1. Rasmussen, C. E., & Williams, C. K. I., *Gaussian Processes for Machine Learning*, MIT Press (2006).
2. Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer (2006).
3. Murphy, K. P., *Machine Learning: A Probabilistic Perspective*, MIT Press (2012).