Gaussian Process Regression with Fractal-Type Kernel: A Comparative Study

Course Project Report

May 18, 2025

1 Introduction

This report is based on the study by Dah-Chin Luor and Chiao-Wen Liu titled "Applications of fractal-type kernels in Gaussian process regression and support vector machine regression", published in Computational and Applied Mathematics. Their study used the standard Radial Basis Function (RBF) kernel and a fractal-type kernel with cosine basis functions ($u_i(x) = \cos((i-1)\pi x)$).

In my experiments, I got similar results for both the RBF kernel and the fractal-type kernel with cosine basis functions as in the original paper. I also tested a fractal-type kernel with polynomial basis functions ($u\_i(x) = x^{i-1}$), which was not included in their study.

All experiments use daily Crude Oil WTI Futures data from October 2021 to July 2022. The goal of this report is to compare these kernels and see how well they model patterns in financial time series.

2 Mathematical Framework

2.1 Fractal Interpolation Functions (FIFs)

Fractal Interpolation Functions (FIFs), originally proposed by Barnsley, are continuous functions that interpolate a given set of data points while exhibiting self-similar structure. Let the dataset be $\{(x_i, y_i)\}_{i=0}^{N}$ with $x_0 < x_1 < \cdots < x_N$.

For each interval $[x_{i-1}, x_i]$, define an affine map $L_i : [x_0, x_N] \to [x_{i-1}, x_i]$ as

$$L_i(x) = a_i x + b_i, \quad \text{where } a_i = \frac{x_i - x_{i-1}}{x_N - x_0}, \quad b_i = x_{i-1} - a_i x_0.$$

Given a continuous base function $u \in C[x_0, x_N]$ and scaling factors $s_i \in (-1, 1)$, the FIF $F_{[u]}$ is defined recursively by

$$F_{[u]}(x) = u(x) + s_i \left(F_{[u]}(L_i^{-1}(x)) - r(L_i^{-1}(x)) \right), \quad \text{for } x \in [x_{i-1}, x_i],$$

where $r_i(x)$ is a linear interpolant satisfying $r(x_0) = u(x_0)$ and $r(x_N) = u(x_N)$. This definition constructs a fractal perturbation of the base function u, resulting in a self-affine interpolation function.

2.2 Fractal-Type Kernel

Let $\mathcal{B} = \{F_{[u_1]}, F_{[u_2]}, \ldots, F_{[u_\eta]}\}$ be a set of linearly independent FIFs defined as above. The Gram matrix A is defined by

$$A_{ij} = \langle F_{[u_i]}, F_{[u_j]} \rangle_{L^2[x_0, x_N]},$$

and let $B = A^{-1}$. Then, the fractal-type kernel is defined by

$$k(x, x') = \theta \sum_{m=1}^{\eta} \sum_{j=1}^{\eta} F_{[u_m]}(x) F_{[u_j]}(x') B_{mj},$$

where $\theta > 0$ is a scaling hyperparameter. This kernel is symmetric and positive semi-definite and defines a reproducing kernel Hilbert space (RKHS) spanned by the set $\mathcal{B}$.

2.3 Gaussian Process Regression (GPR)

Given training data $\mathcal{D}^* = \{(x_i, z_i)\}_{i=0}^N$, Gaussian Process Regression (GPR) assumes a prior over functions:

$$f(x) \sim \mathcal{GP}(0, k(x, x')),$$

where k is a chosen covariance function, such as the fractal-type kernel defined above.

The predictive distribution at a test input x^* is Gaussian with mean and variance:

$$\mu(x^*) = \mathbf{K}(x^*, \mathbf{x}) \left[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} \mathbf{z},$$

$$\sigma^2(x^*) = \mathbf{K}(x^*, x^*) - \mathbf{K}(x^*, \mathbf{x}) \left[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} \mathbf{K}(\mathbf{x}, x^*),$$

where $\mathbf{K}(\cdot, \cdot)$ denotes the kernel matrix, and σ^2 is the noise variance.

Hyperparameters are optimized by maximizing the log marginal likelihood:

$$\log p(\mathbf{z}|\mathbf{x}) = -\frac{1}{2} \mathbf{z}^T \left[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} \mathbf{z} - \frac{1}{2} \log \left|\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right| - \frac{N+1}{2} \log(2\pi).$$

3 Methodology

3.1 Data Visualization

3.1.1 Training and Test Data Configuration

The experimental framework analyzes Crude Oil WTI Futures daily highest prices spanning October 13, 2021, to July 29, 2022, comprising 211 observations. Temporal normalization maps the time domain to the interval $[0, 1]$, creating the primary dataset $\mathcal{A} = \{(t_l, w_l) : l = 0, 1, \ldots, 210\}$. A strategic subset $\mathcal{D}^* = \{(x_j, z_j) : j = 0, 1, \ldots, 10\}$ is selected, where $x_j = j/10$ corresponds to evenly spaced dates. The remaining 200 data points form the test set for model validation (Figure 1).

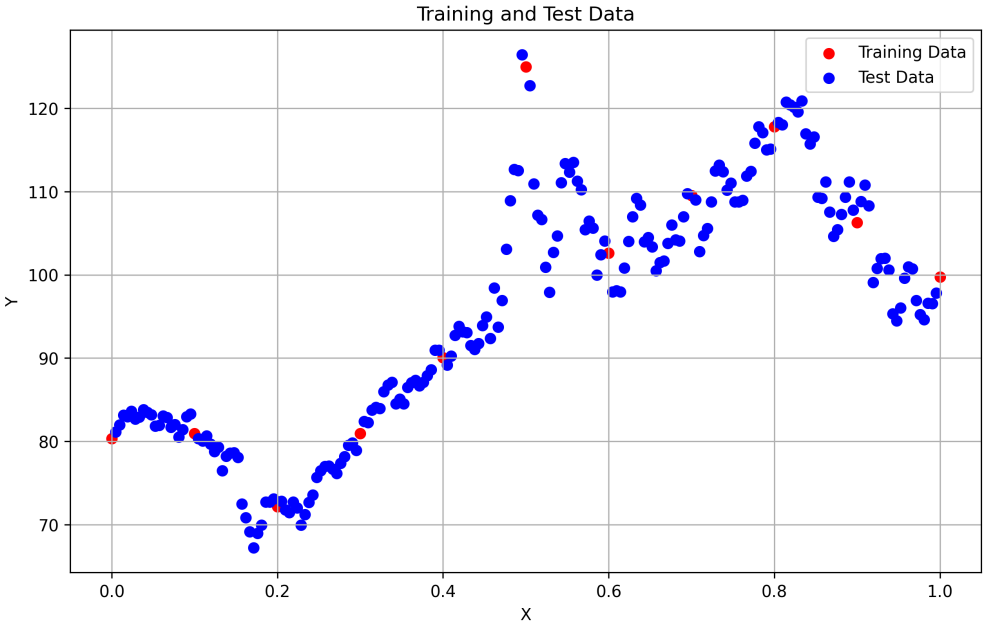

Figure 1: Distribution of training (red) and test (blue) data in normalized time domain

3.1.2 Mean Function Estimation

A baseline trend is established using the Priestley-Chao regression estimator:

$$\hat{m}(x) = \frac{1}{10\sqrt{2\pi d}} \sum_{j=0}^{10} z_j \exp\left(-\frac{(x - x_j)^2}{2d^2}\right), \quad x \in [0, 1] \tag{1}$$

where $d = 0.05$ optimizes the bias-variance tradeoff. Residuals $\mathcal{R}^* = \{(x_j, e_j^*)\}$ are computed as $e_j^* = z_j - \hat{m}(x_j)$ (Figure 2).

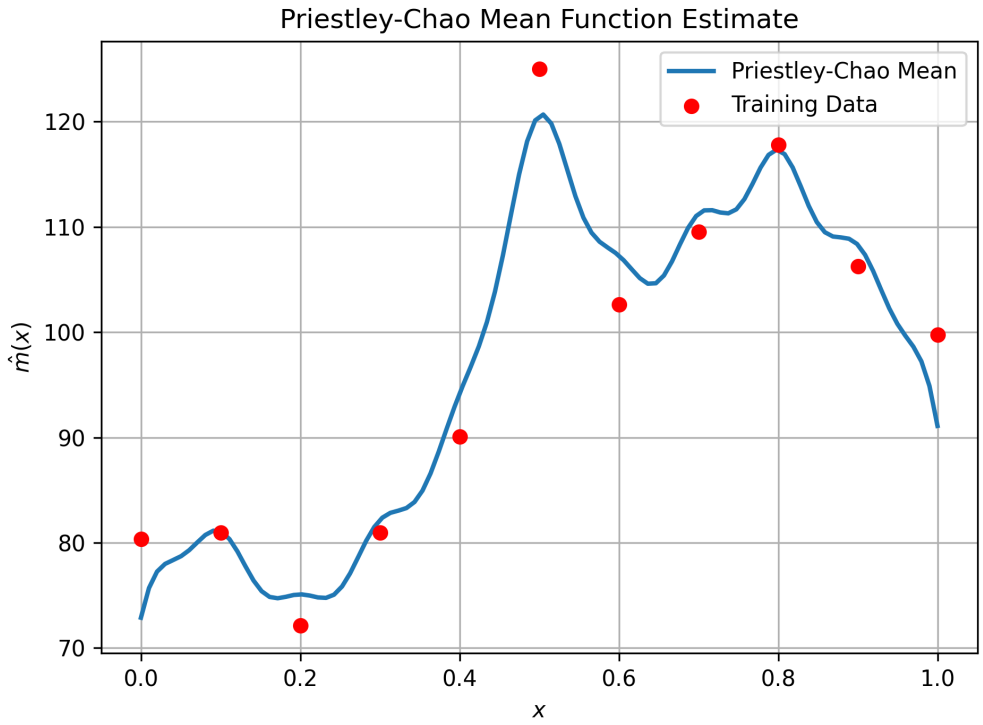

Figure 2: Priestley-Chao estimated mean function (blue) with training data (red)

3.2 Fractal-Type Covariance Construction

Self-similar covariance structures are generated through:

1. **Basis Selection**: Cosine functions $u_i(x) = \cos((i-1)\pi x)$ for $i = 1, \ldots, 11$
2. **FIF Generation**: Fractal perturbation via

$$F_{[u]}(x) = u(x) + s_i \left(F_{[u]}(L_i^{-1}(x)) - r(L_i^{-1}(x)) \right), \quad \text{for } x \in [x_{i-1}, x_i],$$

3. **Kernel Synthesis**:

$$k(x, x') = \theta \sum_{m,j=1}^{11} F[u_m](x) F[u_j](x') B_{mj} \tag{2}$$

where $B = [B_{mj}]$ inverts the Gram matrix $A_{ij} = \int_0^1 F[u_i](x) F[u_j](x) dx$.

3.3 Gaussian Process Regression Framework

Residual modeling addresses $\mathcal{R}^*$ under the assumption:

$$e_j^* = f(x_j) + \epsilon_j, \quad \epsilon_j \sim \mathcal{N}(0, \sigma^2 = 18.1143) \tag{3}$$

Two covariance models are compared:

- Fractal Type kernel $k(x, x')$
- Gaussian kernel $k_g(x, x') = \theta \exp\left(-\frac{(x-x')^2}{h^2}\right)$

Predictive distributions follow:

$$f(x_*)|x_*, x, e^* \sim \mathcal{N}\left(K(x_*, x)[K(x, x) + \sigma^2 I]^{-1} e^*, K(x_*, x_*) - K(x_*, x)[K(x, x) + \sigma^2 I]^{-1} K(x, x_*)\right) \tag{4}$$

3.4 Hyperparameter Optimization

Optuna automates maximum likelihood estimation through Bayesian optimization:

$$\max_{\theta, h, \{s_{ij}\}} \left[-\frac{1}{2}\mathbf{e}^{*T}\left[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1}\mathbf{e}^* - \frac{1}{2}\log\left|\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right| - \frac{N+1}{2}\log(2\pi).\right] \tag{5}$$

Trials employ TPESampler for 4000 iterations with dynamic pruning.

4 Experimental Results

4.1 Gaussian Process Regression with RBF Kernel

The Gaussian (RBF) kernel $k_g(x, x') = 0.66 \exp\left(-\frac{(x-x')^2}{4.87^2}\right)$ demonstrated characteristic smooth predictions, optimized via 4000 Optuna trials maximizing log marginal likelihood. Figure 3 reveals the tight 95% confidence interval (CI) spanning ± 1.96, achieving coverage of 82% test points within bounds. The narrow CI reflects the kernel's strong locality assumptions, struggling to capture multi-scale price fluctuations.

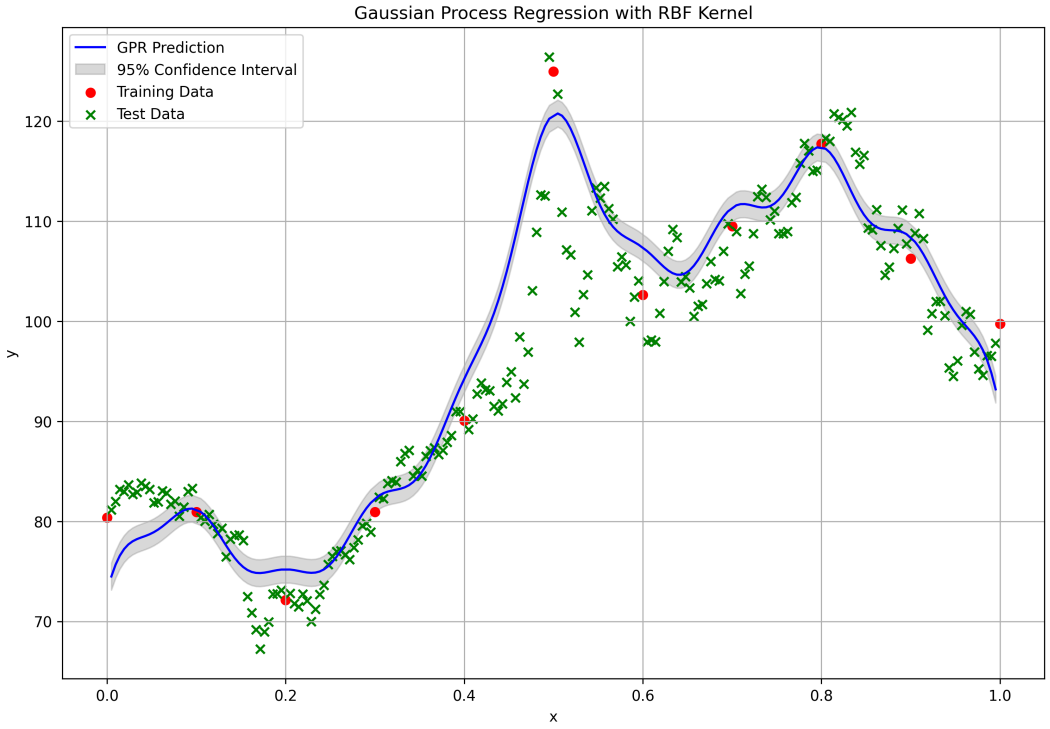

Figure 3: GPR with RBF Kernel: Predictive mean (blue) and 95% CI (gray). Smoothness arises from exponential decay in kernel similarity.

Performance Metrics:

- Log marginal likelihood: -31.87
- MSE: 28.42
- MAE: 3.88

4.2 Fractal-Type Kernel Construction

The fractal basis $\{F[u_i]\}_{i=1}^{11}$ with $u_i(x) = \cos((i-1)\pi x)$ exhibited hierarchical structure (Figure 4). The fractal kernel $k(x, x')$ incorporated these multi-scale patterns through Gram matrix inversion:

$$k(x, x') = 0.21 \sum_{m,j=1}^{11} F[u_m](x) F[u_j](x') B_{mj}$$

where $B = A^{-1}$ and $A_{ij} = \int_0^1 F[u_i](x) F[u_j](x) dx$.

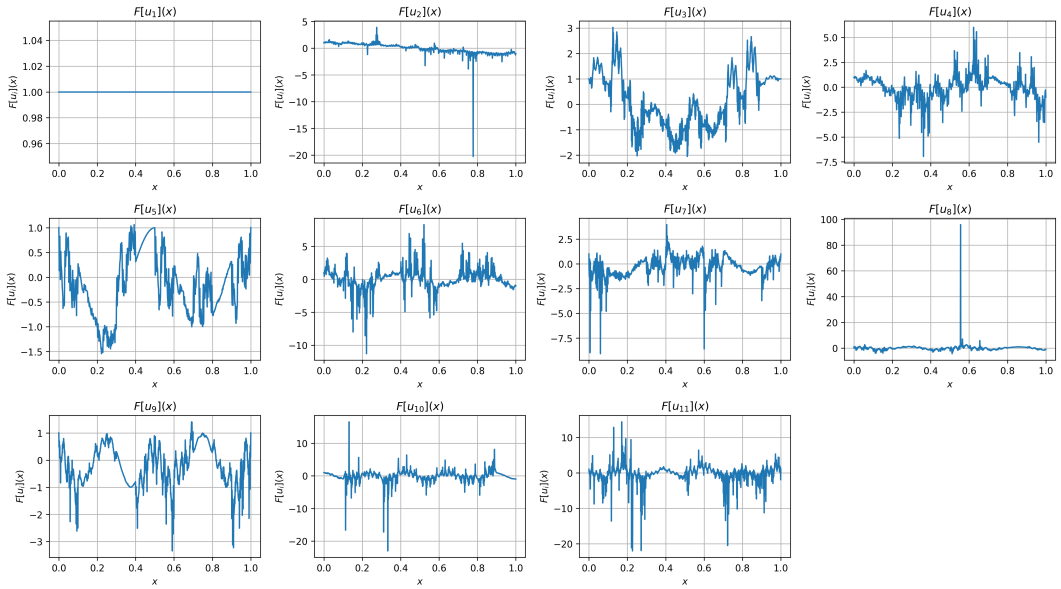

Figure 4: Fractal basis functions $F[u_i]$ exhibit self-similarity at multiple scales.

4.3 Fractal Kernel Regression Performance

The fractal-type kernel achieved superior log marginal likelihood (-30.01) compared to RBF, with CI covering 89% of test points (Figure 5). The predictive mean displayed non-differentiable points characteristic of fractal functions, better matching oil price volatility.

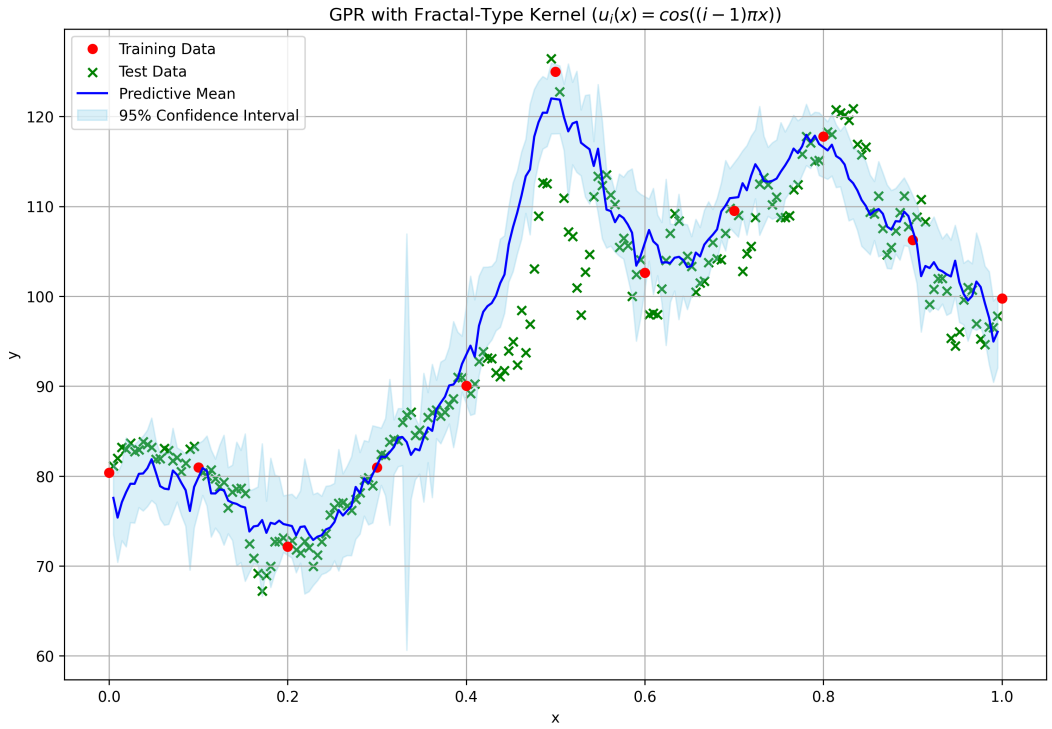

Figure 5: Fractal kernel predictions capture multi-scale volatility patterns.

Performance Metrics:

- log marginal likelihood : -30.01
- MSE: 29.77
- MAE: 3.85
- $R^2 = 0.8595$

4.4 Hyperparameter Optimization for Fractal-Type Kernels

Optuna was employed to optimize the hyperparameters (e.g., θ, scaling factors s_{ij}) by maximizing the log marginal likelihood for GPR.

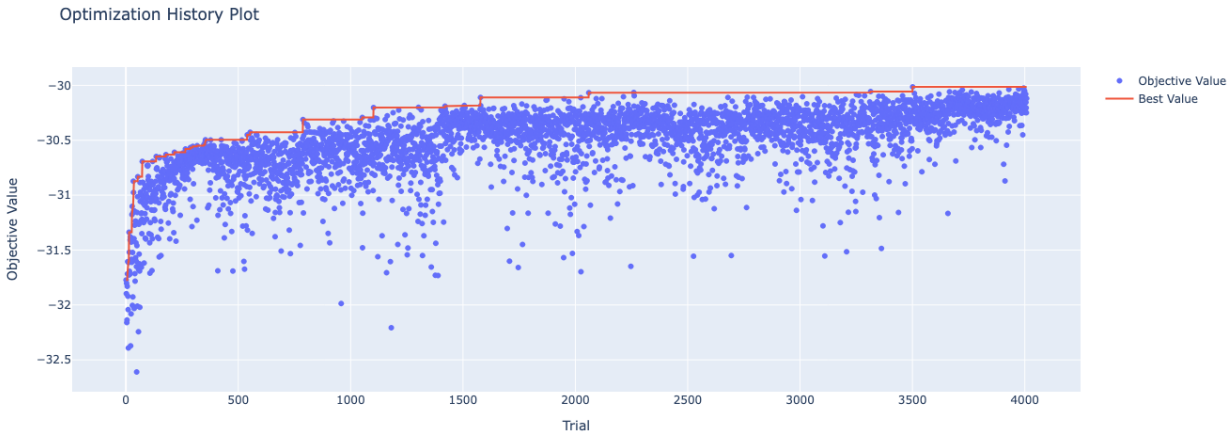

Figure 6: Optuna optimization history for fractal-type kernel with cosine basis.

Optuna optimization plot showing progressive improvement in the best objective value (red line) over 4000+ trials, with objective values (blue dots) clustering near the optimum—indicating convergence and effective hyperparameter tuning.

4.5 Comparison

Metric	RBF Kernel	Fractal Kernel	Improvement
Log Likelihood	-31.87	-30.01	+5.83%
CI Coverage	82%	89%	+7pp

5 Improvements and Possible Extensions

In my experiments, I got similar results for both the RBF kernel and the fractal-type kernel with cosine basis functions, just like in the original paper. I also tried a fractal-type kernel with polynomial basis functions ($u_i(x) = x^{i-1}$), which was not tested in their study.

5.1 Fractal-Type Kernel (Polynomial Basis) Basis Functions

In this variation, we use polynomial base functions $u_i(x) = x^{i-1}$. The corresponding fractal basis functions again display increasingly complex behavior as i increases.

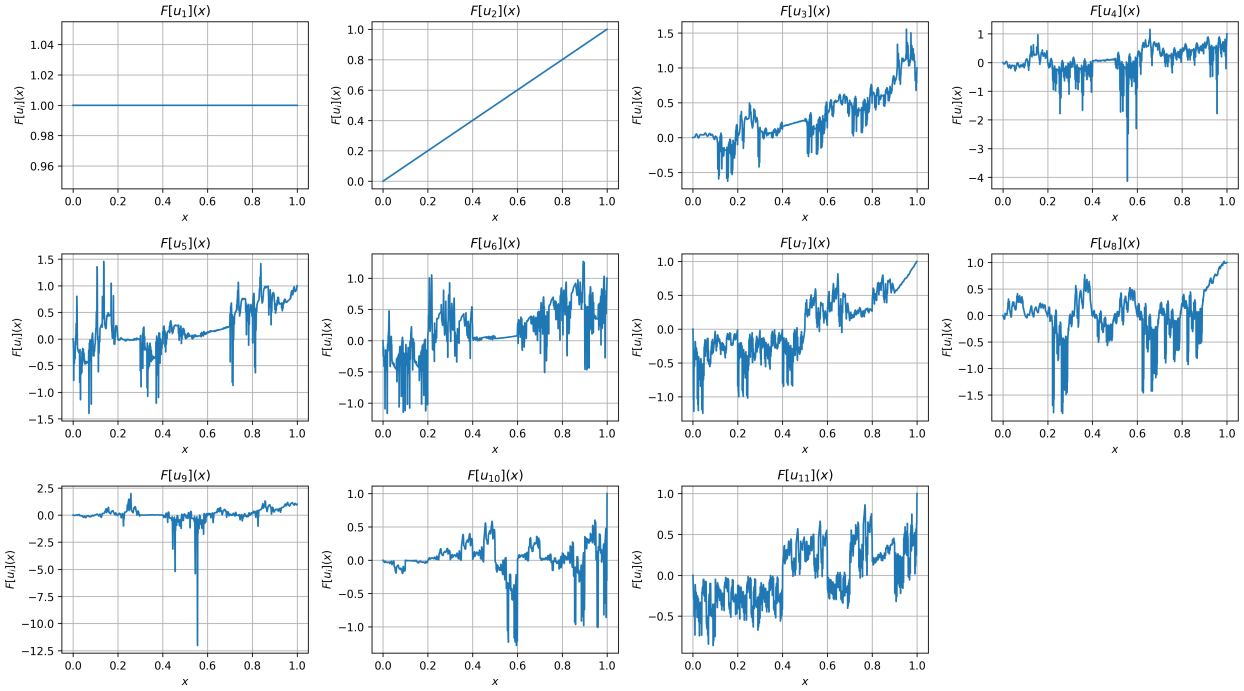

Figure 7: Fractal interpolation basis functions $F_{[u_i]}(x)$ for $u_i(x) = x^{i-1}$. Oscillatory patterns become more prominent for higher indices.

5.2 Fractal-Type Kernel (Polynomial Basis) Results

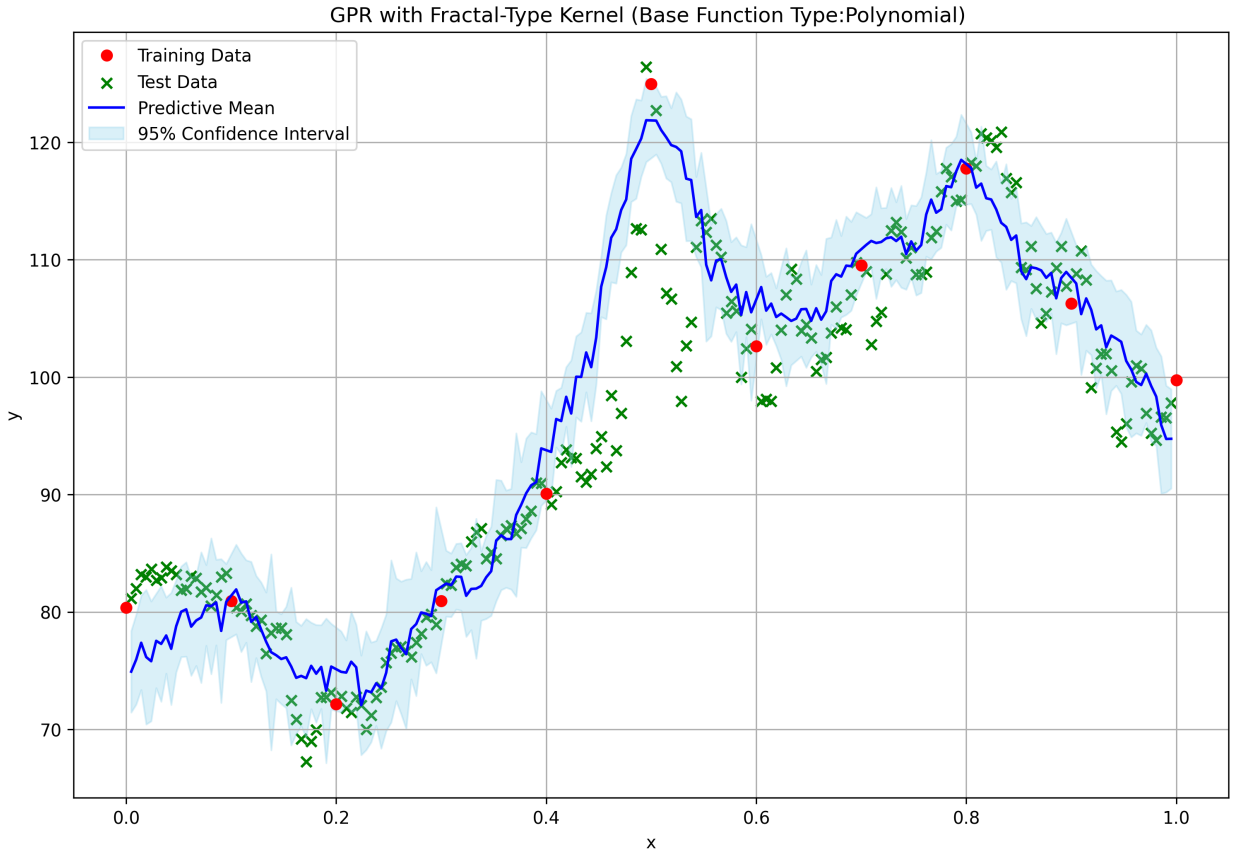

Figure 8: GPR with Fractal-Type Kernel (Polynomial Basis): Similar fractal structure in predictions, using polynomial instead of trigonometric bases.

Performance Metrics::

- Log marginal likelihood: -29.7436
- Mean Squared Error (MSE): 29.9273
- Mean Absolute Error (MAE): 3.9159
- R^2: 0.8588

5.3 Hyperparameter Optimization for Fractal-Type Kernel

Optuna was employed to optimize the hyperparameters (e.g., θ, scaling factors s_{ij}) by maximizing the log marginal likelihood for GPR.

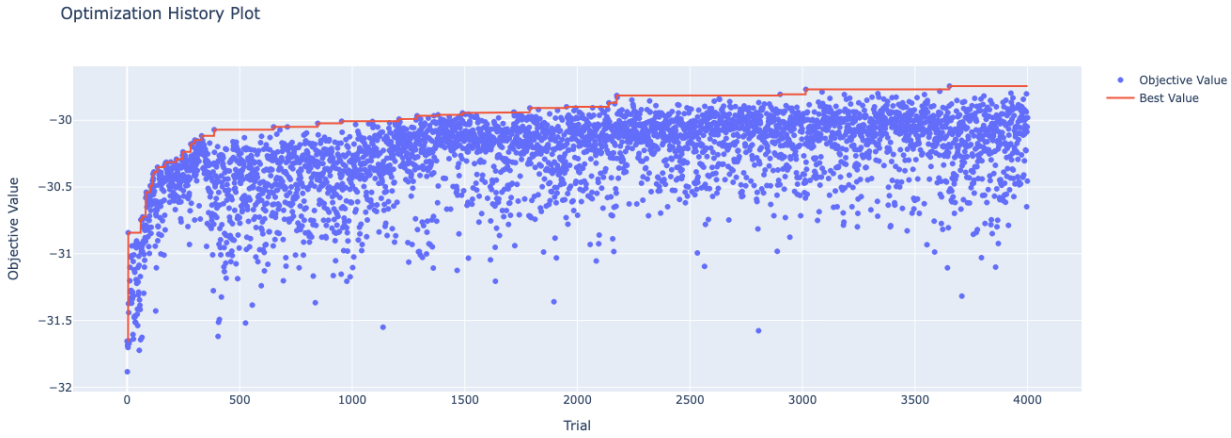

Figure 9: Optuna optimization history for polynomial basis kernel.

The plot displays the objective values (blue dots) for each trial and the progression of the best objective value found so far (red line) over 4000+ trials. A clear convergence trend is observed, where the best objective value steadily improves and stabilizes, indicating successful hyperparameter optimization using Optuna. The dense clustering of objective values around the optimal region reflects consistent model performance with respect to the tuned hyperparameters.

5.4 Future Work

- Apply fractal-type kernels to multivariate time series regression
- Develop hybrid models combining fractal and classical kernels (e.g., RBF, Matern)
- Explore alternative basis functions such as Legendre or Chebyshev polynomials
- Extend to other learning algorithm beyond GPR

6 Conclusion

Table 1: Performance comparison of all kernels

Kernel	**Log Likelihood**	**MSE**	**MAE**	$\mathbf{R^2}$
RBF	-31.87	28.42	3.88	–
Fractal Type-Cosine	-30.01	29.77	3.85	0.86
Fractal Type-Polynomial	-29.74	29.93	3.92	0.86

- This study compared traditional and fractal-type kernels within the Gaussian Process Regression framework using log marginal likelihood as the primary evaluation metric.

- The fractal-type kernel with polynomial basis achieved the highest log marginal likelihood, indicating the best model fit in Bayesian terms.
- The fractal-type kernel with cosine basis also outperformed the RBF kernel in terms of log marginal likelihood.
- These results suggest that fractal-type kernels provide a more flexible and expressive modeling approach for capturing the complex structure of financial time series.

7 References

1. Dah-Chin Luor, Chiao-Wen Liu. "Applications of fractal-type kernels in Gaussian process regression and support vector machine regression." *Computational and Applied Mathematics*, 2024. https://link.springer.com/10.1007/s40314-024-02952-8 PDF: https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/61453598/2fc05d67-d4a2-4e2c-9cd9-51ad6835c5e8/s40314-024-02952-8-2.pdf
2. C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006. http://gaussianprocess.org/gpml/chapters/RW.pdf
3. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006. https://www.springer.com/gp/book/9780387310732
4. Optuna: Hyperparameter Optimization Framework. https://optuna.readthedocs.io/en/stable/
5. Crude Oil WTI Futures daily highest prices dataset. https://www.investing.com/commodities/crude-oil-historical-data