GPCell: A Performant Framework for Gaussian Processes in Bioinformatics

Tristan Sones-Dykes

School of Mathematics and Statistics, The University of St Andrews

2025-04-01

Agenda

- Motivation & Background
- Problem Statement
- The Solution: GPCell
- Results
- Discussion
- Q&A

Motivation & Background

- **Historical Foundations**
  - Jacob and Monod (1961): Demonstrated gene expression as a result of external stimuli through mRNA modulation
  - Hardin, Hall, and Rosbash (1990): Demonstrated feedback mechanisms in gene expression, revealing oscillatory behavior in circadian rhythms.
  - Phillips et al. (2017): Applied Gaussian processes to classify gene expression time series in neural progenitor cells, highlighting oscillatory versus aperiodic dynamics.
- **Introduction to Gaussian Processes**
  - A non-parametric, probabilistic modeling approach that offers flexibility in fitting complex biological signals and quantifying uncertainty.

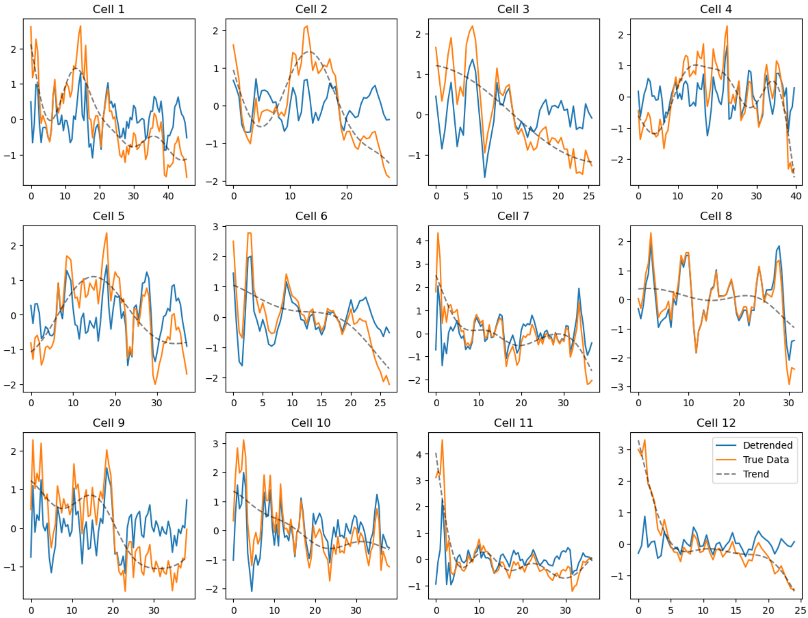

Figure 1: Example of fit GPs

Problem Statement

- **Biological Background**
  - Gene expression regulation is central to cellular function and its oscillatory dynamics can indicate critical biological processes such as differentiation.
- **Current implementation**
  - MATLAB-based Approach: Previous work relied on MATLAB, limiting accessibility and integration with modern machine learning libraries.
  - Performance Issues: Traditional implementations faced slow model fitting and lacked support for parallel processing.
  - Limited Extensibility: The original system was tailored to a specific context and difficult to extend with new inference techniques (e.g., MCMC, Variational Inference).

GPCell

- **Overview:**
  - A generalised Python library, based on GPflow (Tensorflow Probability), to facilitate model fitting and oscillation detection.
- **Key Features:**
  - **OscillatorDetector:**: A class that handles the entire analysis pipeline (background noise estimation, detrending, and GP model fitting).
  - Extensible **GaussianProcess** class: Allows for adding different models (e.g., MCMC for probabilistic inference) and various fitting algorithms.
  - **Utils** module: for fitting models/generating data quickly in parallel.
  - **Robust Pipeline**: Incorporates a strong type system, automated CI/CD, and comprehensive testing for reproducibility.

Methods & Results

- **Validation:**
  - GPCell has been rigorously tested on both synthetic and real gene expression datasets.
  - Unit tests (see correctness.py and gpflow_tests.py) ensure that background noise, detrending, and GP parameter estimation are accurate.
- **Performance:**
  - Leverages parallel processing via Joblib (see fit_processes in utils.py) to dramatically reduce computation time
  - Demonstrated improvements in model fitting speed and oscillation detection accuracy.
- **Visuals:**
  - Results include performance charts and model prediction plots (e.g., mean and variance estimates from the GaussianProcess class).

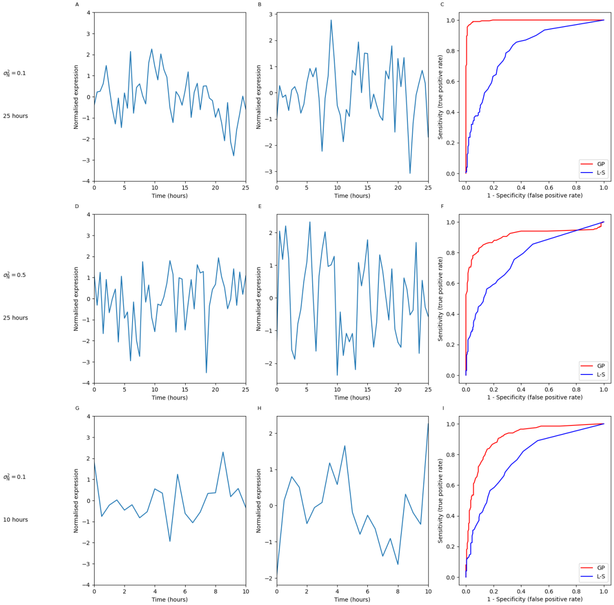

Figure 2: ROC of Gaussian Process and Lomb-Scargle fits

Figures

Performance Comparisons: Data simulation

Table 1: Average data simulation times

Implementation	Time..s.
Python, parallel and njit	58.64
Python, parallel	315.23
Python	3722.16
MATLAB	Approx 1.5 hours

Performance Comparisons: BIC pipeline

Table 2: Average BIC pipeline times

Implementation	Time..s.
Python, parallel	12.54
Python	37.03
MATLAB	44.41

MCMC

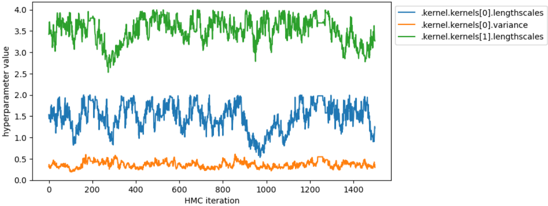

Figure 3: Example MCMC fit using GPCell

Discussion & Impact

- **Strengths:**
  - Modularity: Each component (noise estimation, detrending, GP fitting) is self-contained, making the system easy to maintain and extend.
  - Flexibility: Supports multiple inference techniques (BIC, bootstrap, MCMC) and custom kernel composition.
- **Challenges:**
  - Parameter Tuning: Fine-tuning priors and hyperparameters remains challenging in heterogeneous datasets.
  - Scalability: Handling extremely large datasets may require additional optimizations.
  - Previous Work: Inconsistent choices made (e.g priors) that need to be investigated.
- **Broader Impact:**
  - By transitioning from MATLAB to Python, GPCell enhances reproducibility and accessibility for bioinformatics researchers.

Q & A

Thank you!
Questions?

Hardin, Paul E., Jeffrey C. Hall, and Michael Rosbash. 1990. "Feedback of the Drosophila Period Gene Product on Circadian Cycling of Its Messenger RNA Levels." *Nature* 343 (6258): 536–40. https://doi.org/10.1038/343536a0.

Jacob, François, and Jacques Monod. 1961. "Genetic Regulatory Mechanisms in the Synthesis of Proteins." *Journal of Molecular Biology* 3 (3): 318–56. https://doi.org/10.1016/S0022-2836(61)80072-7.

Phillips, Nick E., Cerys Manning, Nancy Papalopulu, and Magnus Rattray. 2017. "Identifying Stochastic Oscillations in Single-Cell Live Imaging Time Series Using Gaussian Processes." *PLOS Computational Biology* 13 (5): e1005479. https://doi.org/10.1371/journal.pcbi.1005479.