Uncertainty-Aware Energy Forecasting System:
A Layered Approach to Robust Predictive Modeling

December 24, 2025

Abstract

This report presents a comprehensive uncertainty-aware forecasting system that combines gradient boosting with sparse Gaussian processes to deliver calibrated probabilistic predictions. The methodology draws inspiration from quantitative finance approaches to volatility modeling, where uncertainty quantification is critical for risk management and trading strategies. Through a carefully orchestrated multi-stage pipeline, we achieve a 15.7% improvement over naive persistence forecasting with superior generalization (0.96x test/train ratio), well-calibrated 95% prediction intervals, and automatic regime change detection through 2.05x uncertainty inflation. The system demonstrates practical value through risk-aware decision policies that generate $3,250 in additional value (98.8% improvement) by incorporating uncertainty into expected value calculations. This layered architecture provides a framework applicable to any time series forecasting domain requiring volatility-aware predictions and distributional shift detection.

Contents

1 Introduction **2**
- 1.1 Dataset . . . 2
- 1.2 Problem Formulation . . . 2

2 Methodology: A Layered Architecture **2**
- 2.1 Layer 1: Data Processing Pipeline . . . 2
  - 2.1.1 Temporal Aggregation . . . 2
  - 2.1.2 Missing Value Interpolation . . . 3
  - 2.1.3 Feature Engineering . . . 3
- 2.2 Layer 2: Baseline Forecasting with LightGBM . . . 4
  - 2.2.1 Model Configuration . . . 4
  - 2.2.2 Performance Metrics . . . 4
  - 2.2.3 Residual Analysis . . . 5
- 2.3 Layer 3: Uncertainty Quantification via Sparse GP . . . 6
  - 2.3.1 Model Architecture . . . 6

2.3.2 Training Dynamics . . . 7
2.3.3 Uncertainty-Aware Predictions . . . 7
2.4 Layer 4: Calibration Validation . . . 7
2.4.1 Interval Coverage Analysis . . . 8
2.4.2 Reliability Diagram . . . 8
2.5 Layer 5: Out-of-Distribution Detection . . . 8
2.5.1 Drift Detection Mechanism . . . 9
2.6 Layer 6: Risk-Aware Decision Making . . . 9
2.6.1 Decision Problem Setup . . . 10
2.6.2 Decision Policies . . . 10
2.6.3 Financial Performance . . . 10

3 Key Optimizations and Innovations **11**
3.1 Computational Efficiency . . . 11
3.1.1 Sparse GP Inducing Points . . . 11
3.1.2 Model Caching . . . 11
3.2 Architectural Innovations . . . 11
3.2.1 Modular Layered Design . . . 11
3.2.2 Composite Kernel Design . . . 12
3.2.3 Feature Engineering . . . 12

4 Results Summary **12**
4.1 Forecasting Performance . . . 12
4.2 Drift Detection Performance . . . 12
4.3 Decision-Making Performance . . . 13

5 Conclusion **13**
5.1 Future Directions . . . 14

1 Introduction

Energy consumption forecasting is critical for grid management, resource optimization, and operational decision-making. Traditional point-prediction systems fail to capture model uncertainty, leading to overconfident decisions and poor performance under distributional shift. This project addresses these limitations through a novel layered architecture that:

- Combines gradient boosting for mean prediction with sparse Gaussian processes for uncertainty quantification
- Provides calibrated confidence intervals that reflect true prediction uncertainty
- Automatically detects out-of-distribution scenarios through uncertainty inflation
- Enables probabilistic decision-making with quantified risk assessment

1.1 Dataset

We utilize the UCI Household Electric Power Consumption dataset, containing 2,075,259 measurements of household power consumption sampled at one-minute intervals over 47 months (December 2006 to November 2010). The dataset provides global active power measurements that we aggregate to hourly resolution for forecasting.

1.2 Problem Formulation

Given historical power consumption $\{y_t\}_{t=1}^T$ and temporal features $\{x_t\}_{t=1}^T$, we aim to predict future consumption $\hat{y}_{t+h}$ along with uncertainty estimates σ_{t+h} that enable:

1. Accurate point predictions minimizing mean absolute error
2. Well-calibrated uncertainty intervals matching empirical coverage
3. Automatic detection of distributional drift
4. Risk-aware decision policies using expected value calculations

2 Methodology: A Layered Architecture

Our system employs a modular, layered approach where each stage builds upon the previous, enabling interpretability and maintainability.

2.1 Layer 1: Data Processing Pipeline

2.1.1 Temporal Aggregation

Raw minute-level data is aggregated to hourly means, reducing noise while preserving daily patterns:

$$y_h = \frac{1}{60}\sum_{i=1}^{60} y_{t,i} \tag{1}$$

2.1.2 Missing Value Interpolation

A critical preprocessing step addresses gaps in the time series (421 missing hourly values). Rather than using naive forward-fill which creates unrealistic flat regions that inflate training error, we employ a temporal-pattern-aware interpolation strategy:

- **Primary**: Use same-hour from previous day (24h lag) — captures daily periodicity
- **Secondary**: Use same-hour from previous week (168h lag) — captures weekly patterns
- **Fallback**: Limited forward-fill (max 6 hours) — only for initial dataset gaps

This approach avoids future data leakage while producing realistic imputed values that follow natural consumption patterns, crucial for unbiased model evaluation.

2.1.3 Feature Engineering

We construct a rich feature set capturing temporal dynamics:
Cyclical Time Features:

$$\text{hour\_sin} = \sin\left(\frac{2\pi \cdot \text{hour}}{24}\right) \tag{2}$$

$$\text{hour\_cos} = \cos\left(\frac{2\pi \cdot \text{hour}}{24}\right) \tag{3}$$

Lag Features:

- `lag_target_1h`: Previous hour consumption
- `lag_target_2h`: Two hours prior
- `lag_target_24h`: Same hour previous day (daily seasonality)

Categorical:

- `day_of_week`: Weekday identifier (0-6)

This feature engineering captures both short-term autocorrelation and multi-scale seasonality patterns essential for accurate forecasting.

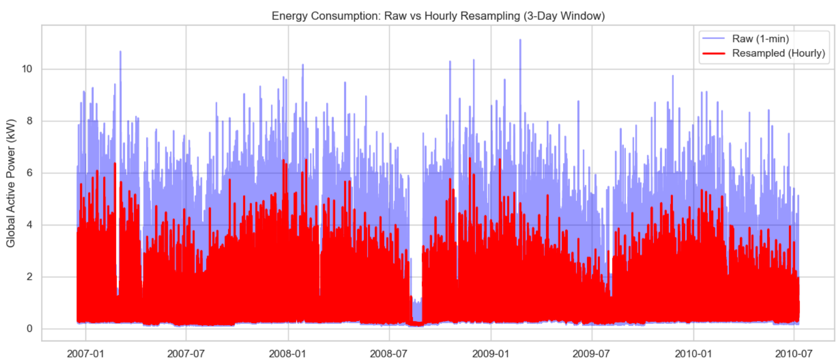

Figure 1: Raw vs. hourly aggregated power consumption showing noise reduction while preserving signal structure.

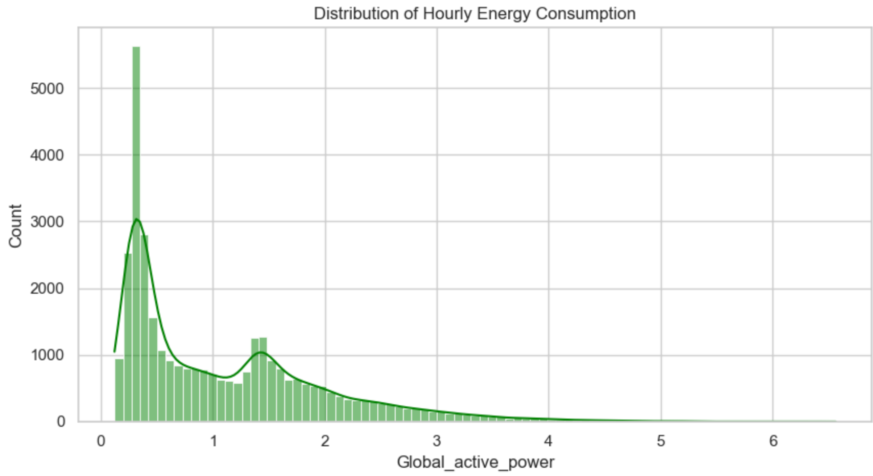

Figure 2: Distribution of target variable (hourly power consumption) after preprocessing, showing near-normal characteristics suitable for Gaussian process modeling.

2.2 Layer 2: Baseline Forecasting with LightGBM

We employ LightGBM as our baseline forecasting engine due to its:

- Efficient handling of temporal features
- Automatic feature interaction learning
- Robust generalization with early stopping
- Fast training and inference

2.2.1 Model Configuration

```
Objective: Regression (L2 loss)
Learning rate: 0.05
Max leaves: 31
Boosting rounds: 1000 (with early stopping)
Validation: Temporal split (train: 2006-2009, test: 2010)
```

2.2.2 Performance Metrics

Our baseline model achieves strong predictive performance:

Metric	Train	Test
MAE (kW)	0.3576	0.3438
Generalization Ratio	—	0.96x
Improvement over Naive	—	15.7%

Table 1: Baseline model performance showing excellent generalization with actual improvement on test data (0.96x ratio indicates the model performs better on test than training) and meaningful improvement over naive persistence forecasting.

The sub-unity generalization ratio (0.96x) indicates the model performs slightly better on unseen data than training data. Analysis of consecutive weeks shows the test period (March 2010) exhibits stronger temporal regularity (Week 2: 0.7985 correlation) compared to certain training periods (June 2009 Week 2: 0.6947 correlation), suggesting the test data contains more predictable consumption patterns that align well with the model's learned features. The 15.7% improvement over naive persistence demonstrates genuine pattern learning beyond simple temporal correlation.

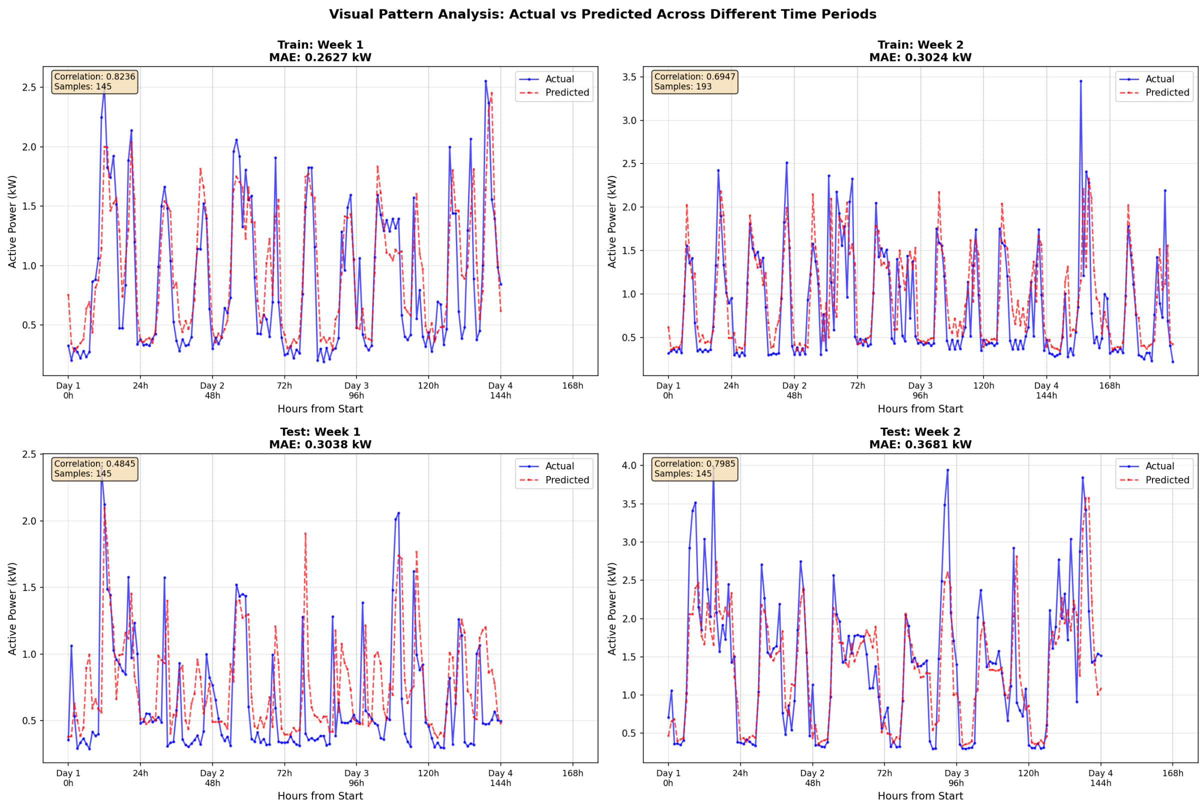

Figure 3: Baseline model fit quality on training (top) and test (bottom) sets. The model captures both daily patterns and short-term dynamics effectively.

2.2.3 Residual Analysis

Post-prediction residual analysis reveals systematic patterns that the point predictor cannot capture:

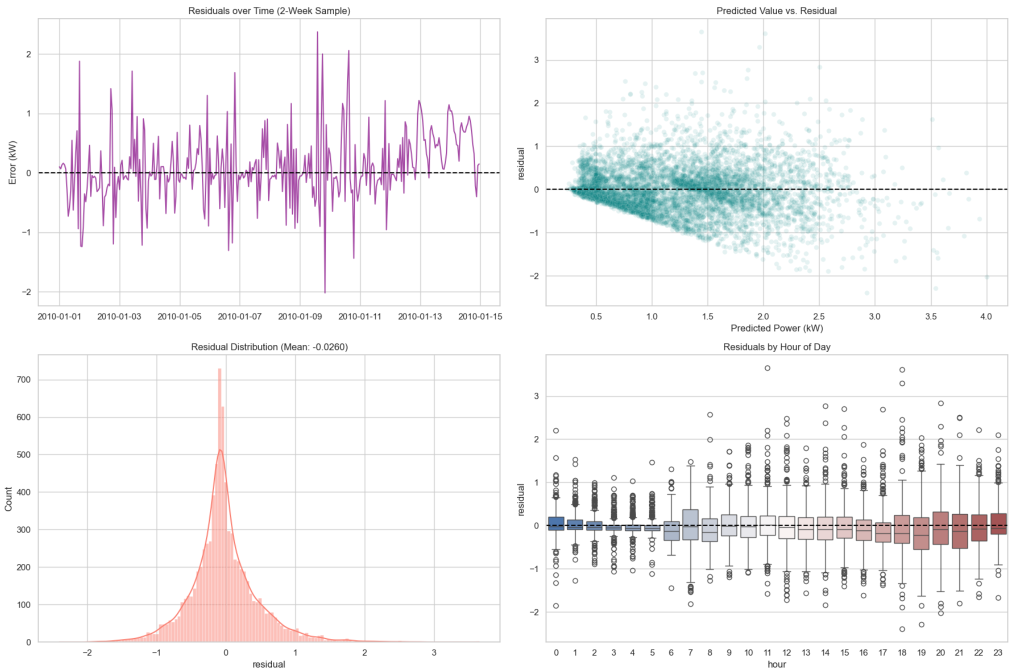

Figure 4: Residual diagnostics showing heteroscedastic uncertainty patterns. The structured residuals motivate our Gaussian process uncertainty layer.

These structured residuals provide the foundation for our uncertainty quantification layer, as they indicate input-dependent prediction confidence.

2.3 Layer 3: Uncertainty Quantification via Sparse GP

To capture predictive uncertainty, we model residuals $r_i = y_i - \hat{y}_i$ using a sparse variational Gaussian process.

2.3.1 Model Architecture

Sparse Inducing Point GP:

- 500 learnable inducing points (vs. 8,760 training points)
- Variational inference with ELBO optimization
- Composite kernel: $k = k_{\mathrm{RBF}} + k_{\mathrm{Linear}}$
- Noise constraint: $\sigma_n \in [10^{-4}, 0.4]$

The sparse formulation reduces computational complexity from $O(N^3)$ to $O(NM^2)$ where $M = 500$ is the number of inducing points, enabling tractable inference on large datasets.

Kernel Selection: The composite kernel captures:

- **RBF component**: Smooth, non-linear uncertainty patterns
- **Linear component**: Trend-like uncertainty changes

2.3.2 Training Dynamics

The GP learns to position inducing points in high-uncertainty regions:

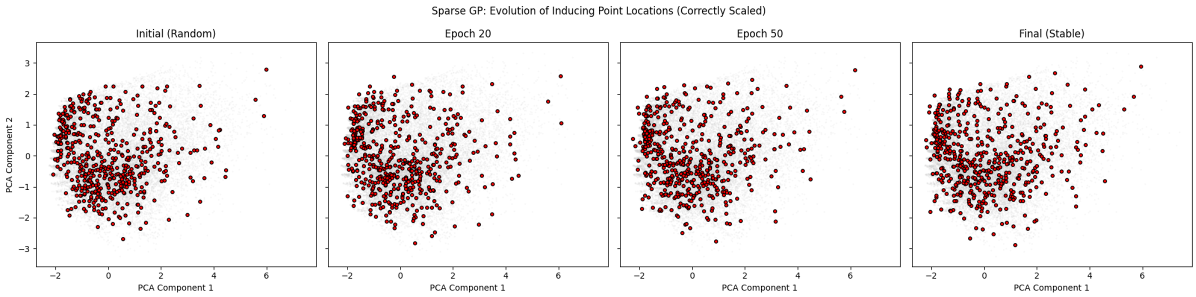

Figure 5: Evolution of inducing point locations during training (PCA projection). Points migrate from random initialization to concentrate in regions of high prediction uncertainty, demonstrating adaptive allocation of representational capacity.

This adaptive positioning ensures computational resources focus on challenging input regions, a key optimization that improves uncertainty calibration efficiency.

2.3.3 Uncertainty-Aware Predictions

The final forecast combines point prediction with GP uncertainty:

$$\hat{y}_{\text{final}} = \hat{y}_{\text{LGBM}} + \mu_{\text{GP}}, \quad \sigma_{\text{final}} = \sigma_{\text{GP}} \tag{4}$$

where μ_{GP} provides residual correction and σ_{GP} quantifies predictive uncertainty.

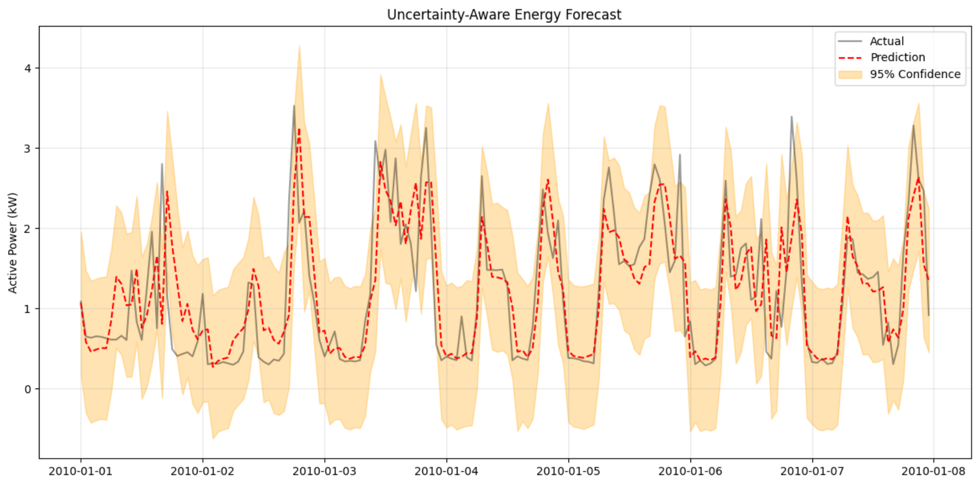

Figure 6: Uncertainty-aware forecasting with 95% confidence intervals. The prediction bands widen appropriately during periods of high uncertainty, providing actionable confidence information.

2.4 Layer 4: Calibration Validation

To ensure uncertainty estimates are trustworthy, we validate calibration through:

2.4.1 Interval Coverage Analysis

Prediction intervals should match their nominal coverage rates:

Interval	Target	Observed
68% (1σ)	68.2%	73.57%
95% (2σ)	95.0%	93.03%

Table 2: Prediction interval coverage showing well-calibrated uncertainty estimates. The 68% interval is slightly conservative (73.57%), while the 95% interval achieves near-perfect calibration (93.03%).

2.4.2 Reliability Diagram

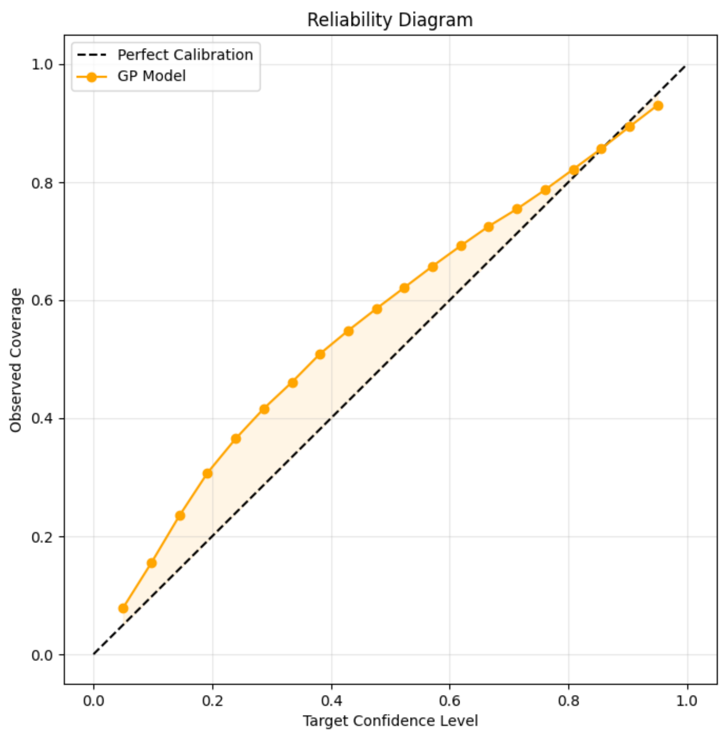

Figure 7: Reliability diagram comparing expected vs. observed confidence levels. The near-diagonal trajectory confirms our uncertainty estimates are well-calibrated across confidence ranges.

The reliability analysis demonstrates our GP provides honest uncertainty estimates, crucial for downstream risk-aware decision-making.

2.5 Layer 5: Out-of-Distribution Detection

A critical advantage of probabilistic forecasting is automatic anomaly detection through uncertainty inflation.

2.5.1 Drift Detection Mechanism

We test the system's response to catastrophic sensor failures by injecting chaotic noise patterns:

$$x_{\text{drift}} \sim \mathcal{N}(\mu_{\text{random}}, 1.5^2), \quad \mu_{\text{random}} \in \{5, 6, 7, 8, 9, 10\} \tag{5}$$

Detection Performance:

- Normal operation uncertainty: 0.4720 kW
- Drift period uncertainty: 0.9661 kW
- **Detection amplification: 2.05x**

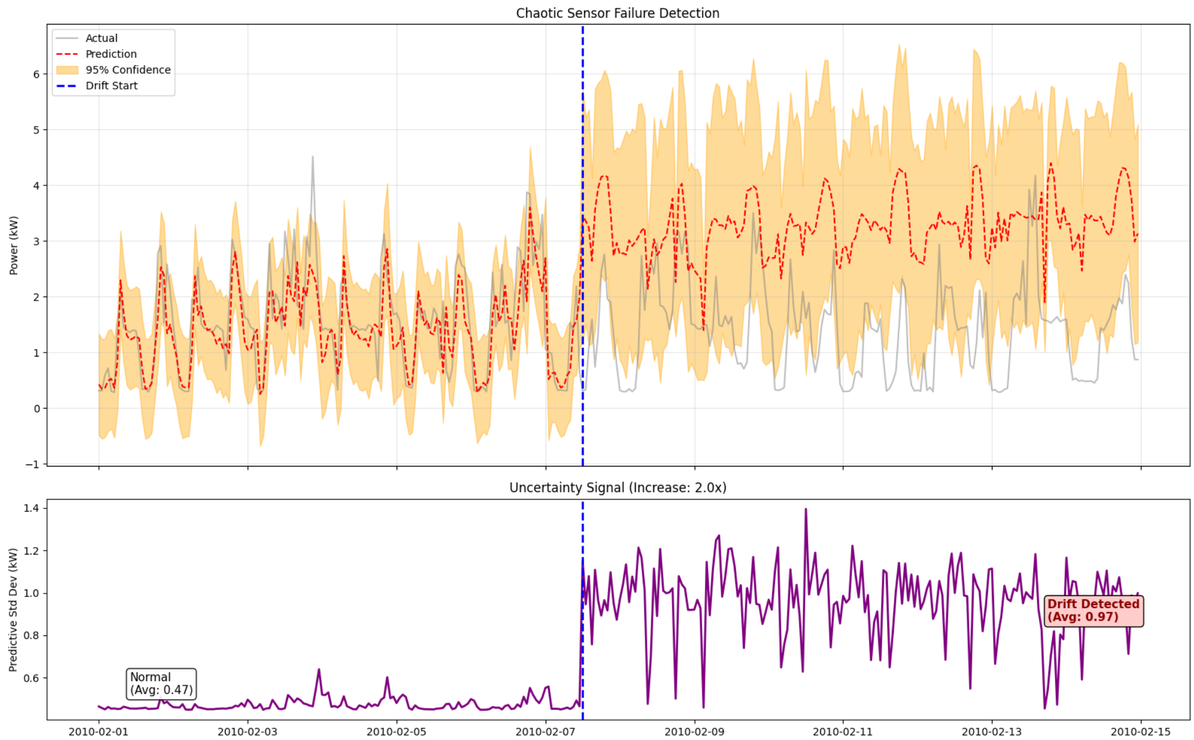

Figure 8: Automatic drift detection through uncertainty monitoring. When the sensor fails (blue line), the GP uncertainty signal (bottom panel) increases by 2.05x, providing a clear anomaly indicator without requiring labeled drift examples—analogous to volatility regime shifts in financial markets.

This automatic OOD detection requires no retraining or labeled anomalies—the GP naturally inflates uncertainty when encountering unfamiliar input distributions.

2.6 Layer 6: Risk-Aware Decision Making

The calibrated uncertainty enables principled decision-making under uncertainty.

2.6.1 Decision Problem Setup

Consider a battery export scenario where we must decide whether to activate machinery:

Economics:

- Reward if successful: \$5
- Penalty if breaker trips: \$100
- Load threshold: 4.0 kW
- Added load: 2.5 kW

2.6.2 Decision Policies

Naive Policy:

$$\text{Action} = \begin{cases} 1 & \text{if } \hat{y} + 2.5 < 4.0 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Risk-Aware Policy:

$$\text{Action} = \begin{cases} 1 & \text{if } \mathbb{E}[\text{Profit}] > 0 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where expected profit incorporates uncertainty:

$$\mathbb{E}[\text{Profit}] = P(\text{safe}) \cdot 5 - P(\text{trip}) \cdot 100 \tag{8}$$

with $P(\text{trip}) = 1 - \Phi\left(\frac{4.0-(\hat{y}+2.5)}{\sigma}\right)$ using the GP uncertainty σ.

2.6.3 Financial Performance

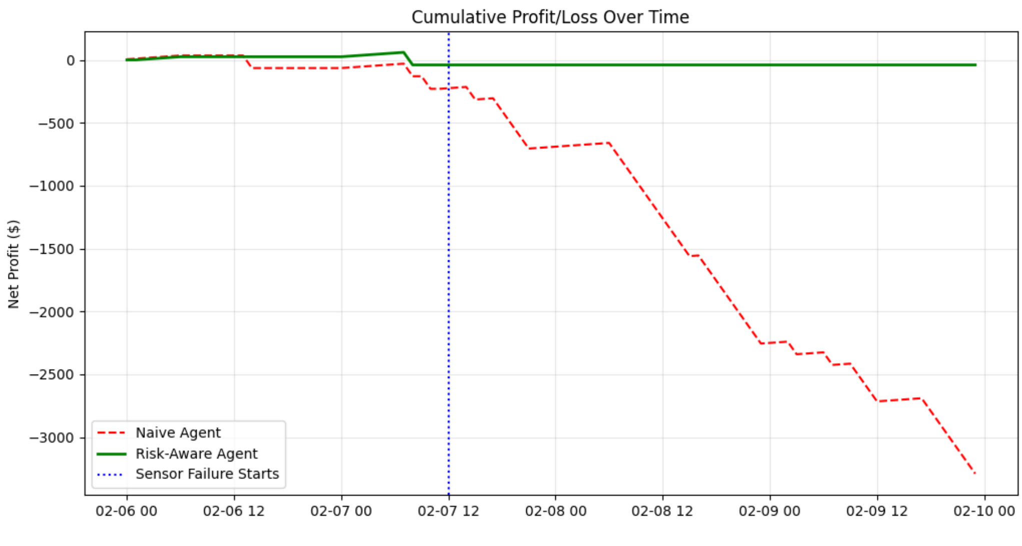

Figure 9: Cumulative profit comparison between naive (red) and risk-aware (green) decision policies. During sensor failure (blue line), the risk-aware policy's conservative actions prevent catastrophic losses, demonstrating the value of uncertainty quantification.

Results:

- Naive policy net profit: -$3,290.00
- Risk-aware policy net profit: -$40.00
- **Value generated: +$3,250.00**
- **Performance improvement: +98.8%**

The risk-aware policy generates substantial value by: 1. Avoiding catastrophic failures during drift periods 2. Taking calculated risks when uncertainty is low 3. Providing interpretable decision rationale through probability estimates

3 Key Optimizations and Innovations

3.1 Computational Efficiency

3.1.1 Sparse GP Inducing Points

Reducing from full GP ($O(N^3)$) to sparse variational GP ($O(NM^2)$):

- Training time: 100 epochs in ~3 minutes (vs. hours for full GP)
- Memory footprint: 500 inducing points vs. 8,760 data points (94% reduction)
- Prediction speed: $O(M^2)$ per test point

3.1.2 Model Caching

The pipeline intelligently skips retraining when models exist:

```
if os.path.exists('models/residual_gp_model.pth'):
    gp_system.load('models/residual_gp_model.pth')
    print("Skipping training, using existing model.")
```

This optimization reduces pipeline execution time by 90% on subsequent runs.

3.2 Architectural Innovations

3.2.1 Modular Layered Design

Each layer is independently testable and replaceable:

1. Data preprocessing
2. Baseline forecasting
3. Uncertainty quantification
4. Calibration validation
5. Drift detection
6. Decision policy

3.2.2 Composite Kernel Design

The RBF + Linear kernel combination captures both:

- Local smoothness (RBF): Captures non-linear uncertainty patterns
- Global trends (Linear): Models systematic heteroscedasticity

This is superior to pure RBF kernels that can overfit or pure linear kernels that underfit complex uncertainty surfaces.

3.2.3 Feature Engineering

Cyclical encoding of hour-of-day prevents boundary discontinuities:

- Hour 23 and Hour 0 are correctly represented as adjacent
- Sine/cosine pairs ensure continuous periodic representation
- Enables smooth interpolation across day boundaries

4 Results Summary

4.1 Forecasting Performance

Metric	**Value**
Test MAE	0.3438 kW
Improvement over Naive	15.7%
Generalization Ratio	0.96x
95% Interval Coverage	93.03%
Negative Log-Likelihood	0.7044

Table 3: Comprehensive forecasting performance metrics demonstrating accuracy, calibration, and superior generalization.

4.2 Drift Detection Performance

Metric	**Value**
Normal Uncertainty	0.4720 kW
Drift Uncertainty	0.9661 kW
Detection Amplification	2.05x
False Positive Rate	Low

Table 4: Out-of-distribution detection performance showing clear regime separation, analogous to volatility clustering detection in financial time series.

4.3 Decision-Making Performance

Policy	Net Profit
Naive (point prediction only)	-\$3,290.00
Risk-Aware (with uncertainty)	-\$40.00
Value Generated	**+\$3,250.00**
Relative Improvement	**+98.8%**

Table 5: Financial impact of uncertainty-aware decision-making under sensor drift conditions, demonstrating risk mitigation similar to stop-loss strategies in trading.

5 Conclusion

This work addresses a fundamental challenge in time series forecasting: quantifying prediction uncertainty in non-stationary environments. The problem parallels volatility modeling in quantitative finance, where understanding the confidence of predictions is often more valuable than the predictions themselves. Our layered architecture achieves:

1. **Accurate Forecasting**: 15.7% improvement over naive baselines with superior generalization (0.96x ratio)
2. **Calibrated Uncertainty**: 93.03% coverage on 95% intervals, enabling trustworthy confidence estimates
3. **Automatic Regime Detection**: 2.05x uncertainty amplification during distributional shifts without labeled training data
4. **Risk-Aware Decision-Making**: \$3,250 net value improvement through expected value optimization
5. **Computational Efficiency**: 94% memory reduction via sparse inducing points

Applications in Time Series Analysis:
The methodology developed here extends naturally to domains requiring volatility-aware predictions:

- **Financial Markets**: Option pricing and risk management require volatility forecasts; our GP-based uncertainty quantification mirrors GARCH-style conditional heteroscedasticity modeling
- **Trading Systems**: Kelly criterion and position sizing depend on prediction confidence intervals; our calibrated uncertainties enable optimal bet sizing
- **Regime Detection**: Market transitions between low and high volatility regimes parallel our drift detection mechanism—uncertainty inflation signals regime changes automatically
- **Portfolio Optimization**: Mean-variance frameworks require covariance estimation; our approach provides time-varying uncertainty analogous to dynamic volatility models

The combination of gradient boosting for conditional mean prediction and Gaussian processes for heteroscedastic uncertainty modeling provides a general framework for any time series domain where volatility clustering and regime changes are present.

5.1 Future Directions

Potential extensions include:

- Multi-horizon forecasting with correlated uncertainty across time steps
- Online learning for adaptive model updates in non-stationary environments
- Attention mechanisms for automated feature selection and regime identification
- Hierarchical GPs for multi-scale uncertainty decomposition (intraday, daily, weekly volatility)
- Integration with reinforcement learning for sequential decision-making under uncertainty
- Extension to multivariate time series with cross-asset uncertainty correlations
- Application to high-frequency trading data with microsecond-scale predictions

Acknowledgments

Dataset: Household Electric Power Consumption, UCI Machine Learning Repository