Warped Gaussian Processes for Power Grid Data Analysis

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Emma Foley

August 2025

For the Belles of Saint Mary's — your example showed me what was possible.

Acknowledgments

I am especially grateful to my advisor, Dr. Peter Fuhr, for his excellent guidance, support, and encouragement throughout this process. His academic and professional mentorship has been invaluable.

I would also like to extend my sincere thanks to my committee members, Dr. Ali Ekti, Dr. Fran Li, and Dr. Yilu Liu, for their time and insightful feedback, which improved the quality of my research.

I am also thankful to my fellow group members at ORNL for their support and collaboration, especially Dr. Elizabeth Piersall, Bill Monday, Jason Richards, Chris Cooper, David Pesin, Dr. Ali Boyaci, and Coreen Prysmont.

My most heartfelt appreciation goes to my family, especially my parents, who always encouraged me to pursue my education. Your support is the foundation of this achievement.

Abstract

Reliable power grid operation is critical for supporting nearly every aspect of modern life. Outages due to extreme weather, equipment malfunction, or even cyber attacks can disrupt infrastructure and create serious, lasting problems for communication and emergency services. The Grid Communications and Security group at Oak Ridge National Laboratory (ORNL) works with industry partners to develop the Autonomous Intelligence Measurements and Sensor Systems (AIMS) project. AIMS is a system using a machine-controlled response team with sensors, mobile platforms, communications and data storage and analysis components. When it comes to monitoring the power grid, leveraging drone and sensing technology provides new opportunities to improve operations. The different data streams comprise an interrelated, complex system with significant amounts of data points. All the data collected by the system will need to be analyzed in order to transform it into useful information. A Gaussian process (GP) is a probabilistic method that has been successfully applied to grid data in the past. However, the assumption that the data and its noise follows a Gaussian distribution is not always realistic. Warped Gaussian Processes (WGP), a generalization of a GP, retains the power of a GP but is not reliant on that assumption. Given the non-stationary and non-linear data that tends to accompany the power grid, WGP is an underutilized method with potential to improve grid modeling. A Pandapower IEEE 14-Bus system was used to generate power grid data to evaluate the performance of WGP as a method for data analysis in the grid context. Other data include real-world signal data from the Grid Event Signature Library and weather-load data from Panama's national grid operator, CND. This analysis demonstrates that WGP is suitable for power grid data analysis because it improves upon GP and other traditional statistical methods in terms of parameter estimation, signal filtering, and load forecasting. Future

work should focus on managing the computational complexity of this method and integrating artificial intelligence and other machine learning techniques into data analysis pipelines.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Combining statistical methods with remote sensing applications is a significant motivating factor behind this research. A portion of this chapter draws from my prior publication titled "Leveraging Gaussian Processes in Remote Sensing" published in *Energies* [1]. Power grid reliability is crucial to supporting critical infrastructure. Service interruptions can have devastating consequences for communication, emergency, and transportation services, just to name a few. In addition to infrastructure, nearly every element of modern day living relies on a stable and functioning power grid, including heating and cooling systems, lighting, refrigeration, telecommunications, and internet access. Interruptions in the power supply can range from minor inconveniences, such as the temporary loss of internet or television services, to more significant problems, like the inability to access medical equipment or emergency services during critical situations [2]. As climate conditions worsen, the reliability of the power grid becomes even more significant. Extreme weather events, such as hurricanes, wildfires, or severe storms, can cause serious damage to power infrastructure, leading to prolonged power outages. Rising temperatures and changing weather patterns can strain the grid [3], resulting in increased demand for electricity, which in turn stresses transmission and distribution systems. Furthermore, much of the existing power grid infrastructure in the United States is aging and needs to be replaced [4].

Recently, Hurricane Helene highlighted the complex nature of power restoration under volatile weather conditions. Millions lost power and complete recovery took weeks. Energy providers like Duke used drones to assess hard to reach areas and aid in recovery [5]. A rising number of cyber and physical attacks also present service concerns [6]. An attack on a substation in North Carolina left thousands without power and cost millions of dollars [7] and a terrorist plot revealed efforts to sabotage certain substations to maximize damage to the grid [8]. Cyber threats to utilities continue to rise and can be carried out remotely, leaving the United States vulnerable to domestic and international threats.

Managing the power grid is no easy task in ideal conditions, to say nothing of the threats of weather and attacks. The Grid Communications and Security group at Oak Ridge National Laboratory (ORNL) researches integrated sensing and communications platforms to ensure efficient and reliable grid communications. Autonomous Intelligence Measurements and Sensor Systems (AIMS) is a system with a machine-controlled response team with sensors, mobile platforms, communications and data storage and analysis components [9]. Figure 1.1 presents a graphical overview of AIMS. This team will be capable of autonomous inspection and reporting with the explicit goal of improved grid reliability. The system will be comprised of advanced sensors, generative design AI, communications, autonomic systems, swarming technologies, energy harvesting, collaborative operation between drone and command center, and training of grid state measurements. This project intends to work in conjunction with existing Supervisory Control and Data Acquisition (SCADA) systems utilities already have in place, allowing for easy adoption by industry partners. Related projects involve the United States Marine Core (USMC) and utilities on Native American Reservations as clients for practical implementation of the AIMS use case [10]. These projects have specific focus on wildfire detection and management. AIMS has significant future implications for infrastructure management.

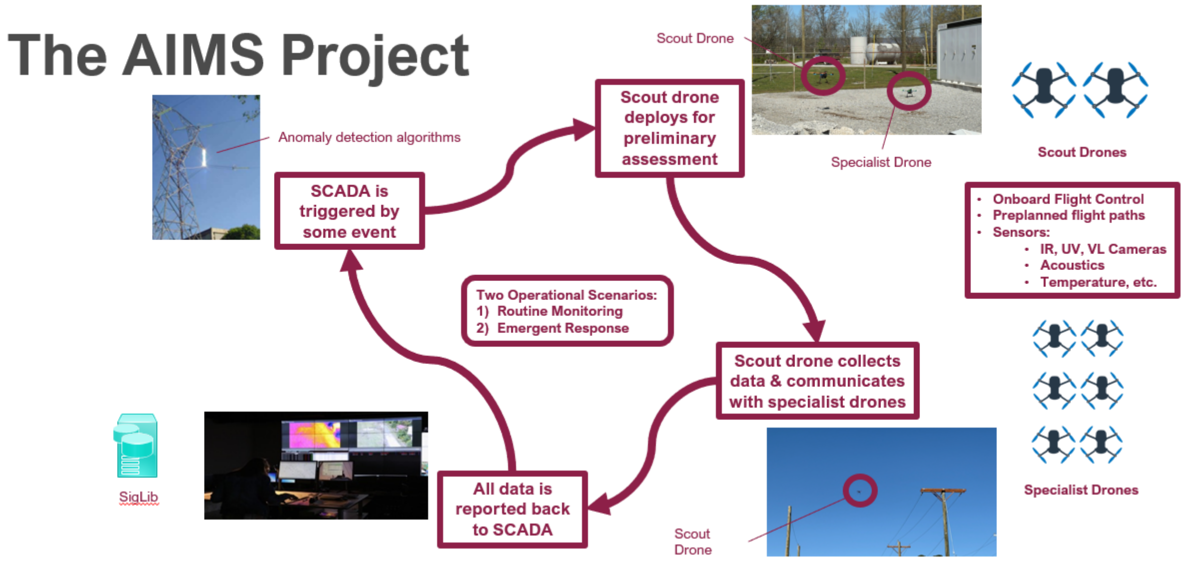

Figure 1.1: Autonomous Intelligence Measurements and Sensor Systems (AIMS) is a project currently underway in the Grid Communications and Security Group at ORNL. This shows an overview of the AIMS operational scenario under some triggering event that requires monitoring [11].

1.2 Background

1.2.1 Remote Sensing

Remote sensing applications have gained significant traction in autonomous surveillance [12, 13, 14], enabling the monitoring and collection of data related to environmental and industrial processes. Various devices, such as satellites, unmanned aerial vehicles (UAVs) or drones, and ocean crawlers, offer valuable tools for data acquisition in diverse contexts. When it comes to monitoring the power grid, leveraging drone and sensing technology provides a new opportunity. Drones equipped with sensors and cameras can be deployed to capture high-resolution imagery and collect data related power infrastructure, such as transmission lines, substations, and power distribution equipment. This data can include visual imaging, thermal imaging, LiDAR (Light Detection and Ranging) data, or other relevant sensor readings.

Machine learning (ML) algorithms can analyze available data to enable more efficient and targeted maintenance and inspection activities, leading to improved grid reliability and reduced downtime. The operational use of drones is efficient and cost-saving. Autonomous drones can quickly cover large areas, making it feasible to monitor extensive power infrastructure without the need for human pilots. They can capture data from challenging or inaccessible locations, such as remote areas or areas with difficult terrain, more easily than traditional manual inspections. This in turn means that human inspectors can avoid hazardous or risky situations, such as working at heights or navigating dangerous environments, minimizing potential risks. Furthermore, drones can be deployed for regular or periodic monitoring, providing real-time or near real-time data on the grid's status. This enables proactive maintenance and response to changing conditions promptly. Drones equipped with advanced sensors can collect precise measurements, providing detailed and accurate data on the condition of power grid infrastructure. This allows for better assessment of the grid's health and can aid in detecting early signs of potential issues.

There are a variety of useful sensors - cameras (infrared, ultra-violet, and visible light), acoustic, radio frequency, temperature, and methane - that provide insights into the power grid. The AIMS projects uses multiple drones in order to accommodate specialty sensors. The output of this monitoring will consist of multiple types of data in multiple formats, so robust methods are necessary for analysis. The Gaussian process is a well-established stochastic method that can model complex relationships while maintaining interpretability. This method continues to be successful in remote sensing applications using satellites and physics-based modeling. However, there is little application of Gaussian processes using drones in a power grid setting. Drones provide high resolution images and much of the data relevant to the power grid is rooted in physics, lending itself to Gaussian applications.

1.2.2 Gaussian Processes

A Gaussian process (GP) is a machine learning method that uses a generalization of the Gaussian distribution. It is a stochastic process that can be used for regression (GPR) and classification (GPC) [15, 16]. Statistically speaking, a GP is a collection of random variables which have a joint Gaussian distribution [17]. Since this method is rooted in probabilistic methods, it provides reliable measures of its own uncertainty and is fairly interpretable [18]. Interpretability is an ideal feature for "physics based" modeling since it can lead to the discovery of factors that influence physical processes, as opposed to the so called "black box" of some ML algorithms. Chapter 2 shows the wide use of GPs in the existing literature for power grid data analysis as well as remote sensing applications. However, GPs can be complex and have high computational costs - the computational complexity of GPs typically grows cubically with the number of data points, limiting their scalability [17]. This is a challenge when dealing with large remote sensing datasets, where acquiring a significant amount of data is common. There are different methods for reducing the computational cost discussed in Chapter 2.

In addition to its complexity, the assumption that the data and its noise follows a Gaussian distribution is not always realistic. Warped Gaussian Processes (WGP), a generalization of a GP, create a "warping" function to transform the data to a latent space in such way that it is modeled by a Gaussian distribution in that latent space [19]. Due to this transformation in latent space, WGP is capable of modeling non-Gaussian and non-stationary data. Furthermore, the warping function is discovered from the data itself and is fully recoverable, unlike many deep learning methods. Since the warping function is specific to the data, it leads to more precise predictions than that of traditional GP (see Figure 1.2). However, similar to traditional GP, WGP has high computational complexity and costs.

WGP is a method that is well-suited to the diverse data that contribute to power grid management. Ambient climate data (temperature, wind, humidity, dew point, etc.) as well as data intrinsic to the power grid (power flow, voltage sags/surges, damping coefficients, etc.) under real world conditions are non-stationary and difficult to approximate, but crucial to operations [20, 21]. The importance of power grid reliability strengthens the need for accurate and timely estimations and predictions. WGP leverages empirical data to create a specific warping function, leading to more accurate prediction while maintaining the stochastic advantages of GPs. WGPs have exciting implications for managing power grid operations. Parameter estimation, load forecasting, and anomaly detection are all capabilities of WGPs, just to name a few. Many aspects of power grid operations are amenable to physics based modeling given their well established scientific properties (voltage, acoustics, temperature, etc.). As previously mentioned, the AIMS project and use of drones in the field will generate more data with similar noise and non-Gaussian considerations.

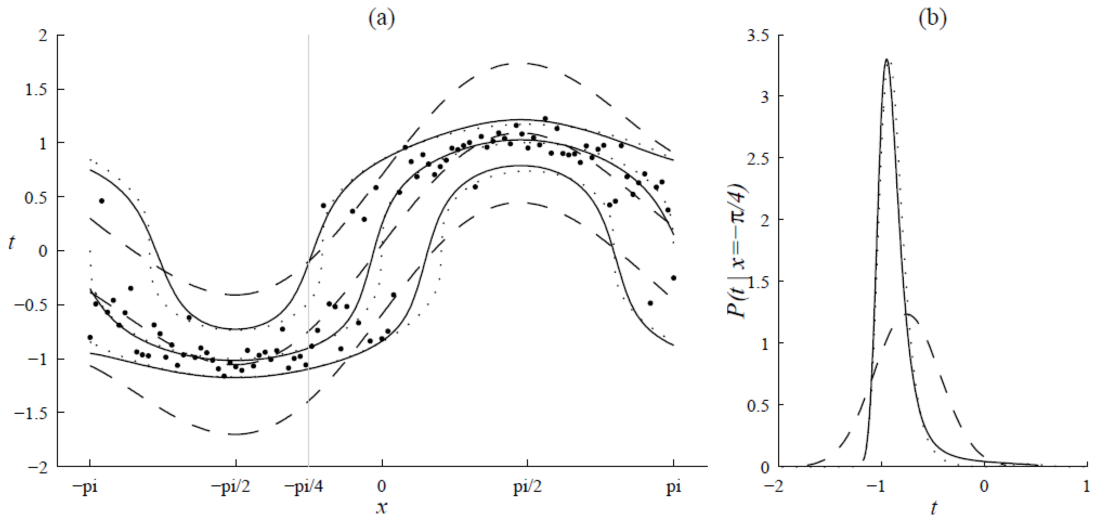

Figure 1.2: Results of a 1-D regression task: (a) The dotted lines show the actual data, the dashed lines show GP predictions and the 2σ percentiles, and the solid lines show the WGP predictions and 2σ percentiles. (b) Predictive probability densities at $x = -\pi/4$. [19].

WGP algorithms can be used to analyze this data on the front end (detect problems that trigger a drone dispatch) and the back end (classification/estimation on the data collected). Online/offline processes would allow for this data to be analyzed after the fact and be used for training as well as for analysis in real time. The power grid is a complex system that provides a critical service. This system would benefit from the flexible modeling and the interpretable results of WGPs.

The purpose of this research is to evaluate WGPs performance as an effective method for analyzing grid data. I applied WGP to load modeling, signal analysis, and weather forecasting. Chapter 2 provides a literature review that draws extensively from my prior publication found in [1].Chapter 3 defines WGPs and outlines the methodology used in this analysis. Chapter 4 summarizes the results, Chapter 5 makes recommendations for future work in this area, and Chapter 6 concludes this paper.

Chapter 2

Literature Review

2.1 Power Flow and Grid Data

Efficient operation of the power grid relies on forecasting. Forecasting in power grid operations is difficult because of its complexity - not only in its technical operation, but also because of climate factors associated with outdoor operation. Wind forces, for example, have serious consequences for power grid operation, but are very difficult to model.

A physics-informed Gaussian process regression (PI-GPR) takes advantage of established physical processes to predict the phase angle, angular speed, and wind mechanical power of a three-generator power grid system [22]. The proposed method is similar to a traditional GPR but uses stochastic differential equations to estimate prior statistics instead of the likelihood. The authors compared the PI-GPR to a standard GPR model, the ensemble Kalman filter (EnKF) method, and autoregressive integrated moving average (ARIMA). They used log predictive probability to measure and compare accuracy. PI-GPR is slightly less accurate but comparable to GPR and had smaller standard deviation overall. EnKF is an online method while PI-GPR is offline, therefore, EnKF is less computationally complex. However, because PI-GPR does all of the prior estimating offline, the actual forecast is less computationally complex than EnKF. In terms of performance, PI-GPR is comparable to EnKF. PI-GPR outperformed ARIMA and was more robust to noisy data. Although PI-GPR does not significantly outperform other methods overall, the first two seconds of forecasting are extremely precise. Figure 2.1 shows a comparison of PI-GPR and data-driven forecasting.

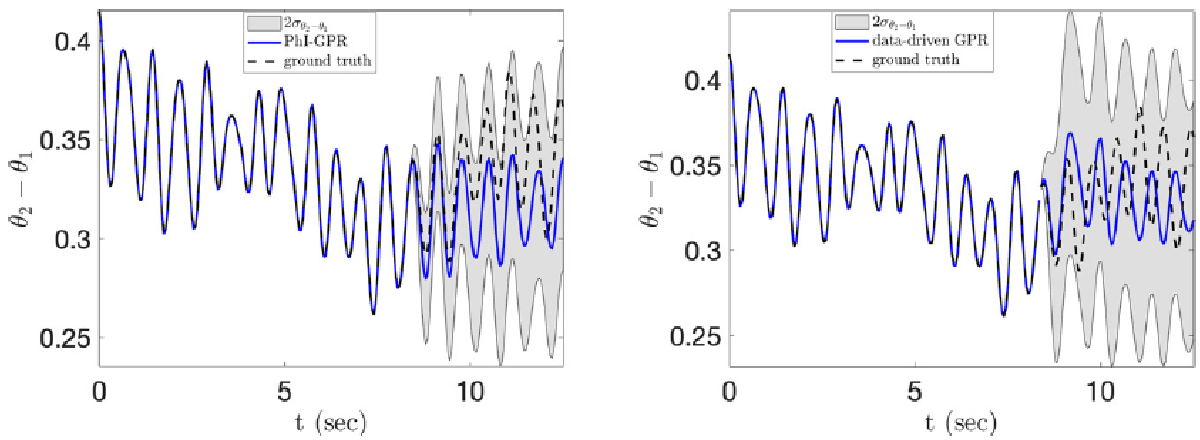

Figure 2.1: PhI-GPR (left) and data-driven GPR (right) forecast of $\theta_2(t) - \theta_1(t)$ with θ_k and ω_k (k = 1, 2, 3) measurements available for t $<$ 8.3375 s every 0.05 s [22].

Another successful method for short-term prediction of electric loads is multi-output Gaussian process regression (MOGP) [23]. The future predictions (24-hour) are based on previous load, temperature, and dew point and the authors used mean absolute percentage error (MAPE) as an accuracy measure in all cases. Simulation cases for prediction included using previous load, using previous load and dew point data, previous load and temperature data, and previous load, dew point, and temperature data. Six different kernels were tested on MOGP performance: linear, Matérn, polynomial, rational quadratic (RQ), piecewise (P.) polynomial, and the radial basis function (RBF). Prediction using previous load performed best with linear kernel. In all other cases, best performance occurred using an RQ kernel with 7 days of previous load and 2 days of previous dew point and temperature data. The results showed that prediction using previous load and temperature data outperformed that of previous load and dew point data and that prediction using all three outperformed all other cases. Opportunities to understand how climate affects the grid and how it can be used to influence grid operations is an interesting area for future work.

Chance constrained optimal power flow (CC-OPF) is a stochastic method used for modeling demand in power grid operations under conditions that allow a ”violation probability” [24]. This method is meant to account for the uncertainties that accompany a complex system like the power grid, for example, significant and unexpected weather events. A major drawback of CC-OPF is the associated computational complexity. A novel framework inserting GP into this process using Python [24] overcomes this issue while maintaining accuracy. Under this framework, the need for knowledge of grid configuration and parameters is eliminated because the GP is data-driven. The authors tested the proposed method using synthetic data from multiple test cases from the Institute of Electrical and Electronics Engineers (IEEE) Bus Systems [25]: the IEEE 9-bus Test System with three generators and three loads and the IEEE 39-bus Test System with ten generators and twenty-one loads. Both test systms had the same voltage level and power ratio and the loads and output from renewable sources are unknown. Root Mean Square Error (RMSE) measures the accuracy of GP prediction. The computation time, generation cost, and the empirical spread of output variables were also used to evaluate the overall performance. In general, RMSE is low, but computational complexity remains an issue in this application and an area for future work.

Initial choices for kernel and basis function in GPR will have different effects on performance. Evaluating load forecasts using multiple kernel and basis functions [26] gives insight into this effect. The authors used MAPE to consider three basis functions (constant, zero, and linear) and six kernel functions (exponential, squared exponential, Matérn 3/2, Matérn 5/2, rational quadratic, Ard exponential). Best performance varies by location, with different basis-kernel combinations and month of year (see Table 2.1). This does make sense in that weather plays a large factor in this model, but it is not particularly revealing of which combinations should be used. That said, in comparison to ANN and regression tree applications, GPR outperformed both in terms of MAPE.

As mentioned previously, the ACPF solution is non-linear and difficult to solve. A general solution framework using voltage and bus injection using GPR is proposed [27] in a simulated voltage limit violation scenario. The authors found a strong relationship between mean absolute error (MAE) and voltage variance. Thus, higher predicted standard deviation leads to higher approximation error. The proposed method outperformed standard methods including data based methods and is not computationally burdensome. In this case, GPR has shown promise in evaluating non-linear processes while maintaining interpretability.

Table 2.1: Results of GPR models with various kernel and basis functions for Indian city 'Aurangabad' in terms of MAPE (%) [26].

Kernel function \ Basic function	Linear	Constant	Zero
Exponential	1.827	0.9	1.313
Squared exponential	2.239	0.844	0.859
Matern 3/2	0.236	0.374	1.403
Matern 5/2	0.962	1.091	2.754
Rational quadratic	0.140	3.172	0.221
Ard-exponential	0.987	0.774	1.356

GPR, Gaussian process regression; MAPE, mean average percentage error.

Load forecasting during anomalous events is difficult given that most applications use historic data that may not reflect the uncertainty of these rare events. Deep Gaussian process (DGP) analysis accounts for limited data in this scenario by mimicking the layering idea of a neural network (NN) [28]. In this study, DGP outperforms back-propagation NN, sparse GP (SGP) and variational autoencoder DGP (VAE-DGP) in terms of MAPE. In general, the proposed DGP is better at capturing the uncertainty of the limited data during an uncertain event. It is clear in Figure 2.2 that DGP captures all of the data throughout the entire distribution.

GPs are not always the most suitable method for a problem, specifically if the data or noise is non-Gaussian. A distributionally robust stochastic method for optimal power flow accounts for non-Gaussian forecast error while maintaining interpretability [29]. Other stochastic methods (like Gaussian methods) must make assumptions about the data that can lead to overestimation or underestimation of risk. The authors contend that a data-driven approach that does not require these assumptions will allow them to more correctly control the tradeoffs between economic efficiency and risk constraint violation, providing an optimal solution to the power flow problem. This method is successful in demonstrating the control over parameters in applications to overvoltages related to photovoltaic power and N - 1 security line flow constraints related to wind generators [30]. The strength of this approach

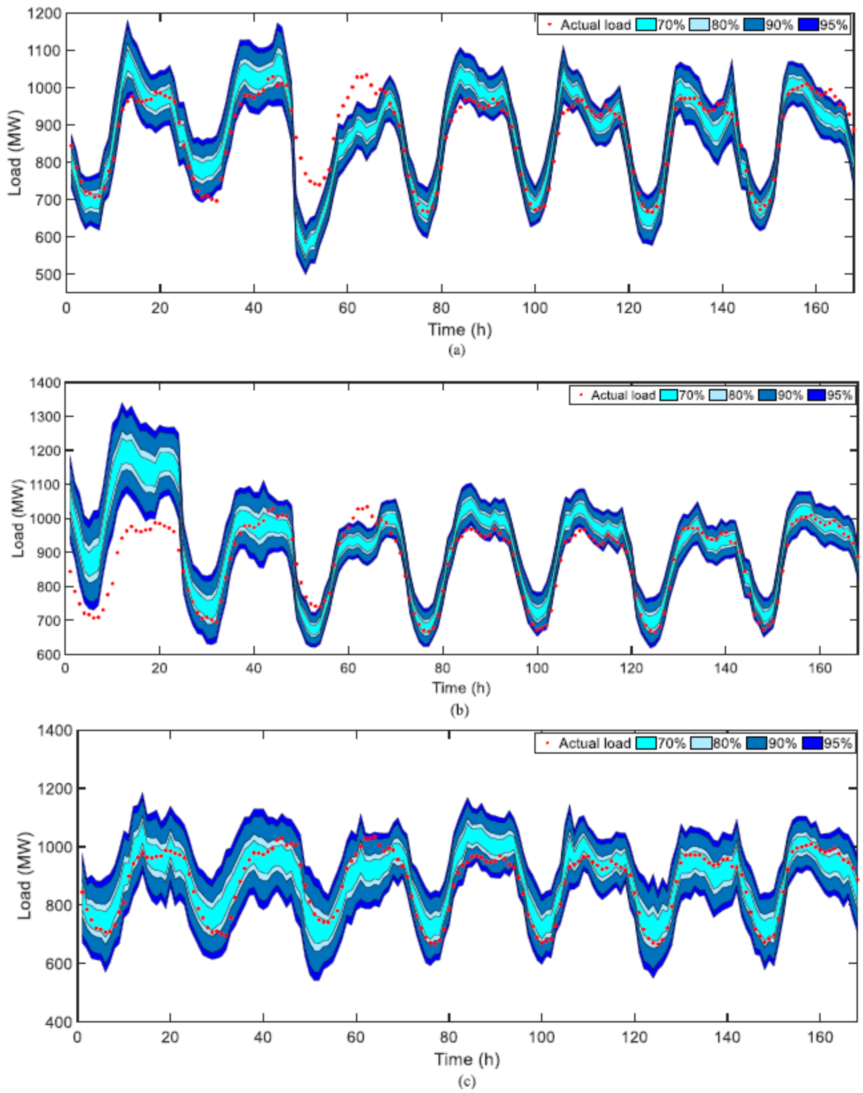

Figure 2.2: Probabilistic load forecasting results of various methods for Seattle. (a) SGP. (b) VAE-DGP. (c) Proposed DGP method. [28] The results clearly show that the proposed DGP method (c) more closely follows the underlying load and captures all the data within a 95% confidence interval.

is in its modeling of forecast error, which leads to improved performance. That said, this approach is theoretically demanding and was not directly compared to other methods.

Real-time anomaly detection is an important factor of power grid monitoring and key to AIMS' goals. Current methods are limited by the assumption that signals are stationary, which is not practical in a real-world scenario. A novel method using the spectral correlation function (SCF) allows for improved characterization of grid-signal distortions because it evaluates non-stationary signals [31]. This method uses the correlation between spectral components of a signal to extract meaningful features. The authors applied SCF to real-world data from the Grid Event Signature Library and compared it to the methods amplitude-phase (AP), Fast Fourier Transform (FFT), power spectral density (PSD) using t-Distributed Stochastic Neighbor Embedding (t-SNE). SCF outperformed the other methods in terms of distinctive grouping and distinguished between anomaly events. Future work includes classifying multiple types of grid signatures and using non-linear models.

Signal distortions are further studied in a solar application. The authors propose that fast amplitude estimation during harmonics and noise will aid in the control of voltage source inverters [32]. They used a discrete Fourier transform with an interpolating window in order to reduce "spectral leakage." This method was tested on simulated data to derive theoretical equations. The strength of this method is in the tractability. The theoretical equations can aid in creating measurement systems because they can account for the interference from both noise and spectral elements.

The authors present the Forecasting Aided States Estimation combined with an Extended Kalman Filter to quickly identify anomalies in PMUs [33] in a Real Time Digital Simulator. This method was compared to the traditional method using largest normalized innovation combined with an Unscented Kalman Filter. The proposed method uses skewness in addition to LNI to improve on the method. Both the traditional method and the improved method were able to detect anomalies, but the improved method is not significantly better than the traditional method, suggesting that filter choice may not play a large role in anomaly detection. Although this method is viable, computational complexity is an issue even with the Extended Kalman Filter.

Signal decomposition is addressed in [34] using a warping function. The authors are building on Variational Mode Decomposition (VMD) specifically applied to the non-stationary nature of rotating machinery. The proposed Warped VMD (WVMD) improves upon VMD because it accounts for the time-series element in rotation which was poorly modeled by VMD. The authors applied this method to simulated and real world data. However, WVMD still needs to be applied to more complicated signals that may not be harmonic in nature. This application leaves a path for future research in a grid application using WGP.

2.2 Estimating Parameters and Unknown/Missing Data

Physics-based simulation is a process to model a dynamic system using the laws of physics. The appeal of physics-based models is the innate reflection of the physical world (time and space) and generalizability. Artificial intelligence methods like machine learning perform quickly and with high accuracy, but at times are intractable and may not be utilizing statistical frameworks to their advantage. The drawback of statistical methods in big data applications is the large computational costs associated with their use. However, new methods have developed that reduce computational time while using the power of statistics in data analysis. WGPs are not widely used in remote sensing and power grid operations, for that reason, much of the literature reviewed has a focus on GPs and other stochastic methods. The following applications are related to remote sensing and many apply their methods to satellite images. This section is divided into categories which reflect the underlying structure exploited by the authors in their analysis: Spatial Statistics, Low Rank and Sparse Approximation, and Spectral Applications.

2.2.1 Spatial Statistics

A common problem in large datasets from remote sensing applications is incomplete lattice, or missing data. A new approach using Bayesian and Maximum Likelihood Estimation (MLE) is proposed in [35] to address this problem. The authors used a Markov chain

Monte Carlo (MCMC) approach for Bayesian inference and a Monte Carlo expectation-maximization algorithm for MLE. The approach was evaluated using both simulated data and real world data from the satellite sea surface temperatures in the Pacific Ocean. Figure 2.3 shows the process of estimating missing data using this method. Bayesian estimation incorporates prior knowledge and quantifies uncertainty but is computationally expensive and statistically complex. MLE offers simplicity and computational efficiency at the expense of explicit uncertainty quantification. Further research should explore hybrid approaches that combine the strengths of both methods for parameter estimation in GPs.

A divide-and-conquer strategy within the Bayesian paradigm addresses scalability in spatial process modeling for geostatistical data [36]. The authors introduce the concept of "spatial meta-kriging" (SMK), which involves partitioning the data into subsets and analyzing each subset on its own. Then, the approximate posterior for the entire dataset can be found by combining the individual posterior distributions from each subset. The authors tested this method using simulations and Pacific Ocean Sea surface temperature data. This article highlights the importance of addressing scalability issues in spatial process modeling and provides a practical statistical approach for large datasets. Areas of future work include discrete and arbitrary spatiotemporal interpolation and multivariate spatial data analysis.

New methods proposed in [37] highlight the limitations of using stationary kernels and the need for non-stationary kernels to model complex spatio-temporal processes. The authors introduce a new class of non-stationary kernels that can be learned from the data using a spectral representation of the covariance function and a GP regression model. Non-stationary random Fourier features are a type of kernel method that can be used to model nonstationary covariance functions in spatial statistics. Non-stationary kernels depend on the inputs, unlike stationary kernels, which only depend on the lag vector. The frequencies used in the feature maps are treated as kernel parameters that are optimized by maximizing the log-marginal likelihood. Other examples of non-stationary kernels for this type of modeling are the polynomial kernel and the neural network kernel. The authors applied this method to daily high stock prices and temperature data. Overall, the results show that arbitrarily complex non-stationary kernels were as easy to implement as stationary kernels but gain in computational efficiency.

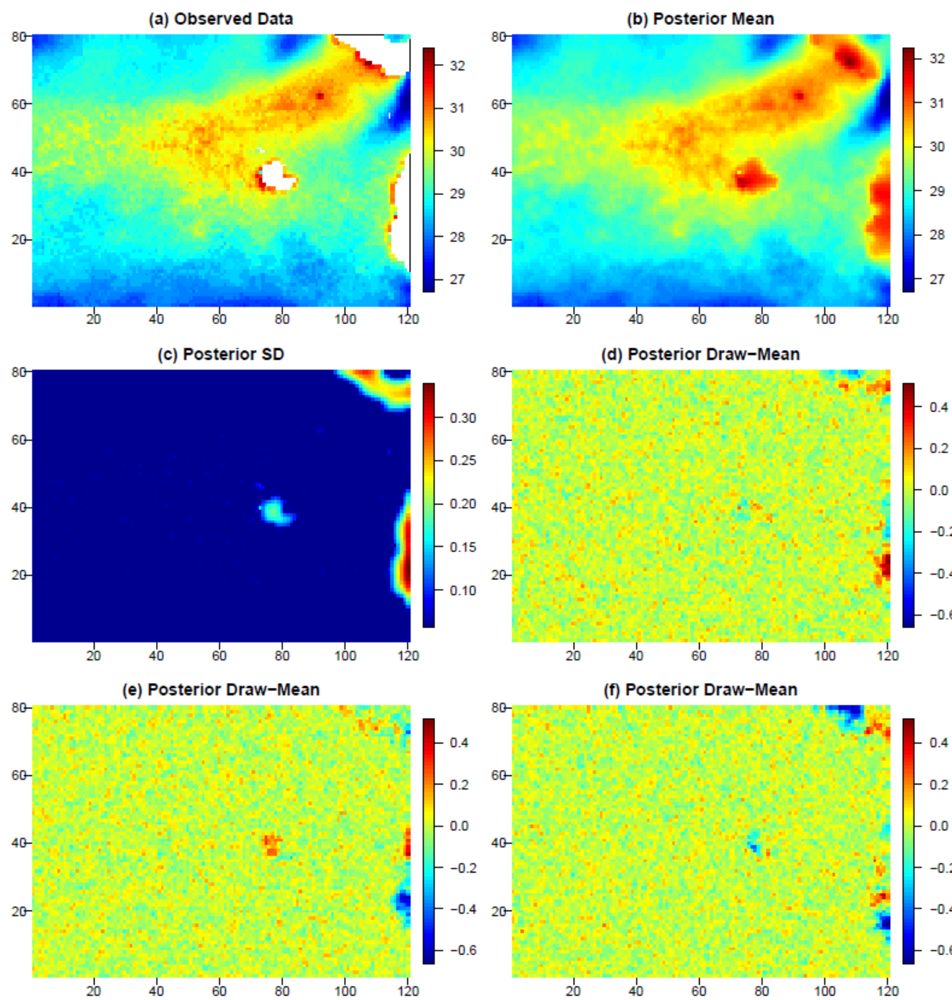

Figure 2.3: Results of missing data reconstruction using TMI satellite data. (a) Observed data with missing portion, (b)-(c) Posterior mean and standard deviation for MCMC MLE estimation. (d)-(f) Posterior draws minus the posterior mean for MCMC MLE estimation. [35]. This collection of figures shows the progression from the observation with missing data to predictions with low standard deviation and low uncertainty for three posterior draws.

Much of the image processing in the past was focused on aerial and satellite images with low spatial resolution. As very high-resolution imaging becomes more accessible, complex scenes require different methods to classify images. A novel framework using semantic segmentation and higher-order conditional random fields (HCRF) exploits the range of contextual information available with higher order CRF [13]. The author's approach uses a harmonic label co-existence (which is typical), but they also introduce local object co-occurence. This method ensures more practical labeling by penalizing unexpected label combinations and labels that rarely occur together (i.e., a tree segment should not appear to be surrounded by car segments). The authors combat high computational costs with movemaking graph cuts. Compared to other methods, HCRF had the highest accuracy, cleared noisy labeling, and resulted in the most practical labels.

2.2.2 Low Rank and Sparse Approximation

Hierarchical partitioning creates data subsets based on hierarchical groupings to improve efficiency. One such method was developed using a hierarchical low rank approximation scheme for maximum likelihood estimation that is computationally efficient for large datasets [38]. The generalized hierarchical low rank approximation method (HLR) diverges from other methods by completing a low rank approximation for each hierarchy, meaning the approximated covariance is not low rank. Computational efficiency is gained by doing many more small calculations of eigenvalues and eigenvectors but avoiding the entire eigendecompositon. A numerical study, simulation study, and real-world study were conducted comparing HLR to several other methods. The numerical study shows HLR outperforms other methods in terms of dependence, smoothness, and noise level, and improves approximation. The simulation study reveals HLR outperforms other methods in terms of mean square error (MSE) and the real-world study shows better performance based on likelihood. However, in the real-world study, a neural network performed faster than HLR.

Alternate formulations of the proposed hierarchical nearest neighbor Gaussian process (NNGP) models [39] expand on the previous work. In this case, the focus is on exploiting the sparsity of large spatial datasets. The NNGP model took multiple forms: the collapsed

NNGP, the response NNGP, and the conjugate NNGP. The collapsed NNGP uses an MCMC to sample from the posterior and create an NNGP model capable of posterior predictive inference. This model operates most closely to a full GP model and remains computationally expensive since it recovers random spatial information and makes predictions. The response model similarly samples from the posterior to create a posterior predictive inference response NNGP, however, it does not fully recover latent information. To its advantage, this reduces computational cost while allowing for full Bayesian analysis and prediction at arbitrary locations. The conjugate NNGP is the most well-rounded because it retains the computational efficiency of the response and gives exact Bayesian inference. The computational advantage is the result of sampling from the marginal posterior distributions instead of the posterior. However, the conjugate NNGP introduces a hyperparameter that requires tuning. The authors' work is the first to successfully use full Bayesian analysis on a spatial dataset at a scale of five million locations. Future work in this area would include analysis of spatial datasets from non-stationary processes, an appropriate application for AIMS, which will use drones equipped with LiDAR for the purposes of monitoring power grid assets.

Fast evaluation of the likelihood for parameter inference in large datasets is also explored with Vecchia approximations [15]. Vecchia approximations are a class of approximations that reduce the computational complexity of GPs by exploiting the spatial structure of the data. The authors discuss two different variable orderings for the Vecchia approximation: response-first ordering and latent-first ordering. In response-first ordering, the response variables are ordered first, followed by the latent variables. This ordering is useful when the response variables are of primary interest and the latent variables are used to model the spatial dependence (the opposite for latent-first ordering). There are three different schemes for response-first ordering: full conditioning, partial conditioning, and no conditioning and two different schemes for latent-first ordering: full conditioning and no conditioning. Figure 2.4 shows the results of parameter estimation using the proposed method. A simulation reveals that the low-rank and sparse approximations did well in both accuracy and computational efficiency. Furthermore, the choice of variable ordering and conditioning have significant

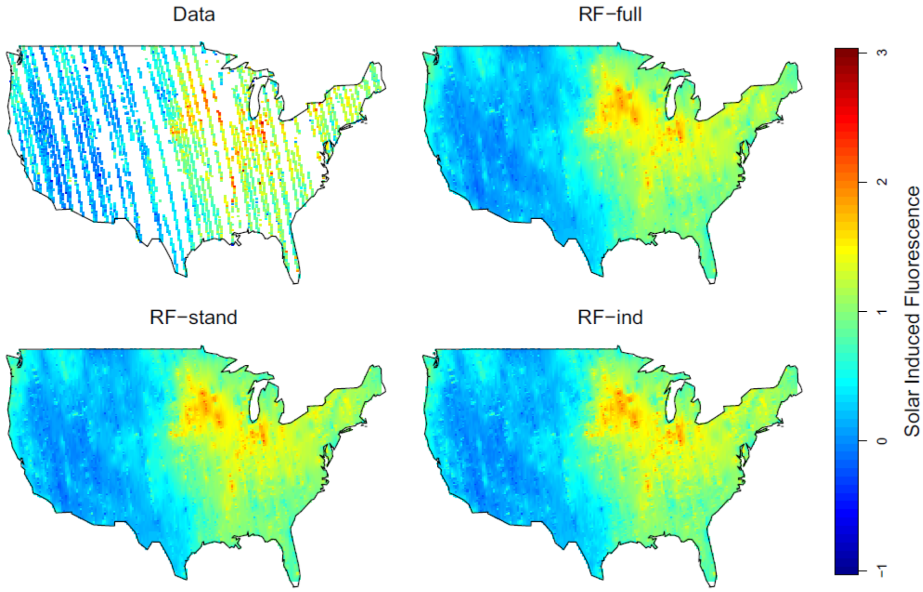

Figure 2.4: Results of applying ordering Vecchia approximations on data from a real-world Orbiting Carbon Observatory 2 satellite to predict solar-induced chlorophyll fluorescence for the RF-full, RF-stand, and RF-ind methods. These methods perform similarly, but the RF-full shows less noise than RF-stand and RF-ind, as evidenced by the streakier quality the of latter two. These results provide evidence for fast and accurate evaluation in large, sparse datasets [15].

impact on both. The first-order methods were also applied to data from a real-world Orbiting Carbon Observatory 2 satellite with similar results.

2.2.3 Spectral Approximation

Class specific random forest (CSRF) successfully combines machine learning and stochastic processes [40]. CSRF, created by the authors, combines cross-correlation measures into random forest applications for classifying hyperspectral images. Spectral-spatial cross-correlation preserves the spectral-spatial components of the image and then uses class specific trees to classify the image. Applying CSRF to an Indian Pines Dataset and a Washington, D.C. mall Dataset reveals that CSRF outperforms support vector machine (SVM) methods and achieves the highest accuracy when combined with a composite kernel. That said, CSRF would benefit from future efforts in parameter tuning and computational cost analysis.

Change detection is the process of identifying and quantifying changes in remotely sensed images [41]. Detecting changes in these images is challenging due to variations in lighting conditions, sensor characteristics, and geometric distortions. This is only made more complicated by multimodal sensor systems. The proposal of an unsupervised pixel pairwise-based Markov Random Field (MRF) model for multimodal change detection in remote sensing images is explored with a Bayesian approach [12]. MRF captures spatial relationships between pixels in remotely sensed images by incorporating contextual information and dependencies between neighboring pixels. Figure 2.5 gives a visual representation of this framework. The MRF framework effectively models the spatial interactions by defining an energy function that incorporates the pairwise potentials and assigns labels to each pixel. This approach includes local and global contextual cues, leading to improved accuracy. MRF outperformed other models in both accuracy and computational efficiency. Future work should focus scalability and the potential for real-time change detection applications.

2.3 Gaussian Process Emulators

One particular type of physics-based GP is called an emulator. An emulator, also called a surrogate or proxy model, builds a statistical model meant to mimic a particular process [42, 14]. The advantage of emulators over full GPs is the increased computational efficiency while maintaining tractable models that can produce confidence intervals. Emulators focus on the use of physics-aware GPs in remote sensing and their implications for accurate and interpretable predictions. A Joint GP model combining actual data with simulated data, a latent force model (LFM) that encodes ordinary differential equations, and Automatic Gaussian Process Emulator (AGAPE) for forward and inverse modeling with automatic emulation [42] explore emulators as generators. Simulated data from a same-site scenario versus a cross-site scenario was evaluated to determine if the inclusion improved prediction accuracy. Although the Joint GP model did not outperform a full GP method, it did show that it is a safe way to combine simulated data with real data. The GP-LFM model combines data-driven Bayesian approaches with purely mechanistic models such that it can be a generative model, maintain interpretability, and show good extrapolation capabilities. However, there is significant computational cost to this method and it is only applicable to differentiable processes. In a toy example with multi-output and scalar inputs, AGAPE

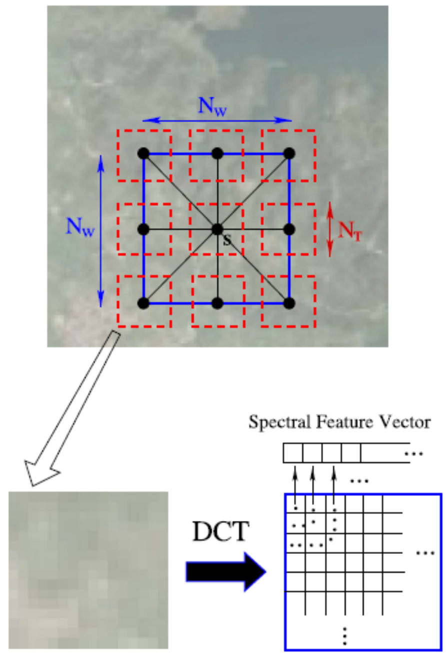

Figure 2.5: The MRF change detection model takes each pixel and includes a subsample window to create a spectral feature vector that incorporates neighboring pixels into the change detection algorithm [12].

outperformed other methods in approximating the underlying function it was emulating. However, this method's success is very dependent on choosing good starting points.

Emulators are further studied by using active learning to mimic complex and costly computational codes in a multioutput setting [14]. Here, the authors refer to a generic active emulation (AE) function and active multi-output Gaussian process emulator (AMOGAPE). Figure 2.6 shows the general idea behind Automatic Emulation. They focus on a sequential method that uses an acquisition function that is optimized via gradient based methods. In this case, the authors apply AMOGAPE as a GP interpolator and regression formulation to synthetic and real-world data. They compared AMOGAPE to other sequential and non-sequential methods using synthetic data. In both cases, AMOGAPE outperformed other available methods as the number of nodes increased. Finally, the authors applied AMOGAPE to a leaf canopy PROSAIL radiative transfer model (RTM) in two dimensional and three dimensional space. Once again, AMOGAPE outperforms random sampling in both cases.

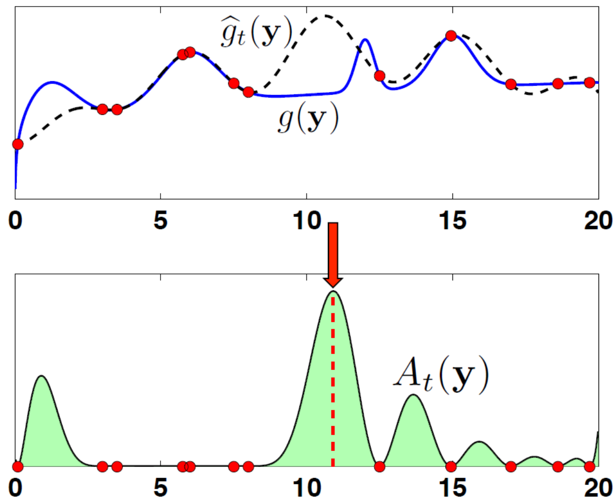

Figure 2.6: General sketch of an Automatic Emulation (AE) procedure. The actual model $g(y)$ (top - solid line), its approximation $\hat{g}_t(y)$ (top - dashed line) and an acquisition function $A_t(y)$. The maximum represents the new node [14]. This is an iterative process that continues produces new nodes until some condition has been met. The activation function is physics-informed since it captures the underlying distribution of the data and its geometry.

Future work involves using alternative acquisition and kernel functions as well as performance measures other than RMSE.

2.4 Additional Gaussian Applications

The aforementioned problems with traditional GPs are addressed by utilizing Deep Gaussian Processes (DGP) that account for complex kernel processes, scale to large datasets, and improve prediction accuracy [16]. DGP creates hierarchies of GP models in a similar structure to deep neural networks. This setup improves computational efficiency since it can model multiple simple kernels, as opposed to a complex kernel required by some GP models. Figure 2.7 shows a graphical representation of the proposed framework. The authors applied this method to superspectral sounding data with multiple models: a full GP model, a sparse GP model, and DGPs with one to four layers. DGPs with two or more layers outperform the full GP model and are more computationally efficient.

Learning control-oriented models involving the development of optimal experiment design, receding horizon optimal control, and continuous model improvement are also available

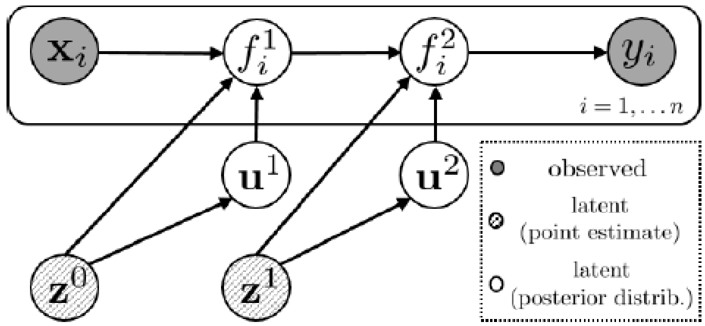

Figure 2.7: Graphical representation of the DGP model [16].

for GP applications [18]. Data-driven control strategies are possible with the integration of GPs in control frameworks because they can model system uncertainties/anomalies and predict behavior. Over time, the incorporation of prior knowledge will allow for optimized control actions and more adaptability within system dynamics. A case study of a simulated electricity grid with real world data verified this method in a "demand response" problem. This research provides a valuable framework for using data-driven machine learning for physical systems controls with GPs. Further research should focus on addressing scalability and real-time implementation.

In practice, data does not always follow a Gaussian distribution, so it is transformed (usually with a logarithmic or exponential function). A Warped Gaussian Process (WGP) addresses this by using a model that automatically learns the optimal transformation by "warping" the predictions of a standard GP model and using continuous optimization to infer the parameters [43]. Another advantage of WGP is that the learned warping function is accessible. The authors compared the performance of the novel WGP to standard GP and variational heteroscedastic GP (VHGP) through three different experimental applications: the estimation of ocean chlorophyll concentration, vegetation parameter retrieval, and causal inference on a set of 28 geosciences problems. The chlorophyll experiment revealed that WGP outperforms standard GP and VHGP in all comparisons, the vegetation parameter retrieval showed that WGP had lower uncertainty and more homogeneous results, and WGP performed better than the other methods in causal inference. These combined results show that WGP performs well and has explanatory capabilities. Future work would benefit from WGP applications to non-parametric functions.

2.5 Temperature Detection in Images

Temperature sensing for identifying hot spots on power lines and transformers is valuable to power grid operations. Hot spots can be indicative of failure or potential failure and affect efficiency. Infrared (IR) is a convenient way to estimate temperature and IR cameras can be easily mounted on drones for inspection, as in the AIMS project. A new method with LiDAR and long-distance IR cameras [44] uses Temperature-Radiation and Radiation-Digital distance mapping to approximate temperature more accurately. The process uses an offline method for data collection and interpolation (calculating distance through digital count and radiation via Planck's law) and an online method for analysis. In the online process, LiDAR provides the distance measure and the IR camera provides the radiation measure. In a blackbody application, the proposed method outperformed forward looking IR and single mapping method at 3.5 and 7 m according to MSE. That said, under non-controlled conditions, atmospheric elements like humidity may have negative effects on results.

An application to stream management does consider the problems atmospheric conditions can add to temperature estimation from high and low resolution IR images [45]. Spectral mixing analysis (SMA) is the method of choice since it accounts for spectral signature, the topography, shadowing effects and the spatial resolution. This method was not compared to others and is limited by instrumentation of the time (2001). Although this analysis is focused on stream management, it is relevant to power grid applications in that the stream is an asset that must be distinguished from air temperature surrounding it, much like power lines. This method could be revisited with new technology.

A direct application of IR imaging for hot spot identification [46] using drones leverages information from both IR images and visible light images using image registration methodology. This process involves first identifying the power lines in the image using a Gaussian filter, edge detection, and Hough transform. After identifying power lines, a histogram threshold is used for hot spot identification. Once again, this method was to test functionality and was not compared to other methods, nor did it estimate temperature directly. Future work in this area should make comparisons to other methods and attempt to measure temperature.

2.6 Wind Speed Forecasting

Wind speed is difficult to forecast because it is random in nature and therefore there is no probability distribution that totally describes it. [47] compares four different distributions: a 2-parameter Weibull, 3-parameter Weibull, Gamma, and Lognormal distributions to ascertain the most suitable. Table 2.2 shows the results applied to wind data. The 3-parameter Weibull outperforms the other distributions, but none are significantly fit. Statistically, these distributions are interpretable, but a more flexible method like WGP may be useful here.

Statistical models are usually limited to short term prediction, but hybrid models seek to overcome this [48]. GP regression combined with other meteorological data and atmospheric stability ratings on public data from UK weather stations outperformed the current numerical method. However, it was somewhat location dependent. Furthermore, this method was interested in applications to improve wind power output prediction more accurately, not specifically for forecasting power grid loads. Other research in hybrid models combing GPs with RNN similarly seeks to increase prediction accuracy while maintaining the probabilistic properties of GPs [49]. This method uses GP to provide necessary information to enhance the RNN model. The authors applied this method to data from the National Wind Energy Technology Center to assess its accuracy. The method outperforms other state of the art methods in terms of RMSE, but accuracy decreases as the forecast step increases and computational time was not reported.

A sparse GP with moving window strategy is proposed for predicting wind gusts [50]. Historical data from numerical weather prediction (NWP) and the European Center for Medium-Range Weather forecasts (ECMWF), as well as observed data from the 2022 Winter Olympics were used to train and validate the proposed method. The proposed method outperforms the others (ECMWF, GP regression, random forest, ECMWF-Sparse

Table 2.2: Results from a comparison of four distributions for wind speed modeling [47].

Distribution	MLM estimates of the parameters	Statistical test					
		χ^2 -test ($p-value$)	KS test ($p-value$)	AIC	BIC	R^2	RMSE
Weibull	$\alpha = 2.5009,\ \beta = 1.4954$	0.2167 (0.6415)	0.0894 (0.9248)	57.5254	60.1934	0.9781	0.0447
3-parameter Weibull	$\alpha = 3.4996,\ \beta = 1.9692,$ $\theta = -0.4383$	0.5091 (0.4755)	0.0773 (0.9925)	58.5468	62.8488	0.9860	0.0357
Gamma	$\alpha = 3.9409,\ \beta = 0.3375$	0.9901 (0.6096)	0.1442 (0.5569)	61.5560	64.4240	0.9510	0.0643
Lognormal	$\mu = 0.1530,\ \sigma = 0.5915$	6.5611 (0.0873)	0.1707 (0.3289)	67.9062	70.7742	0.9131	0.0819

GP regression, and ECMWF-multiple linear regression models) in both short term and long term forecasting. The authors note that the spatial correlation of the input variables may be affecting the accuracy of this method. Furthermore, the moving window is meant to reduce computational complexity, but computational times were not reported.

An application using WGP is presented in [51]. The authors use sparse online WGP in order to account for the time-based data and the high computational cost associated with WGP. Once again, the purpose of this analysis is to better predict wind power production. On data from an actual wind farm, the proposed method was competitive with GP and splines quantile regression. However, future work would improve hyperparameter estimation, include heteroscedasticity, changes in wind direction, and terrain. Figure 2.8 shows the wind power prediction compared to the observed data and with confidence intervals. The results show the flexibility of WGP in action.

2.7 Drone Functionality

2.7.1 Exploration

A truly autonomous system requires no user intervention. One element of autonomous drone navigation is exploration. Programming a specific path for a drone can be efficient in scenarios where the operation is routine, however, in the case that an event is not routine, an autonomous drone needs to be able to make path decisions. In these events, a drone may need to be deployed to gather information and would ideally know when it has collected enough or must continue searching. GPs can determine the best trajectory based on available

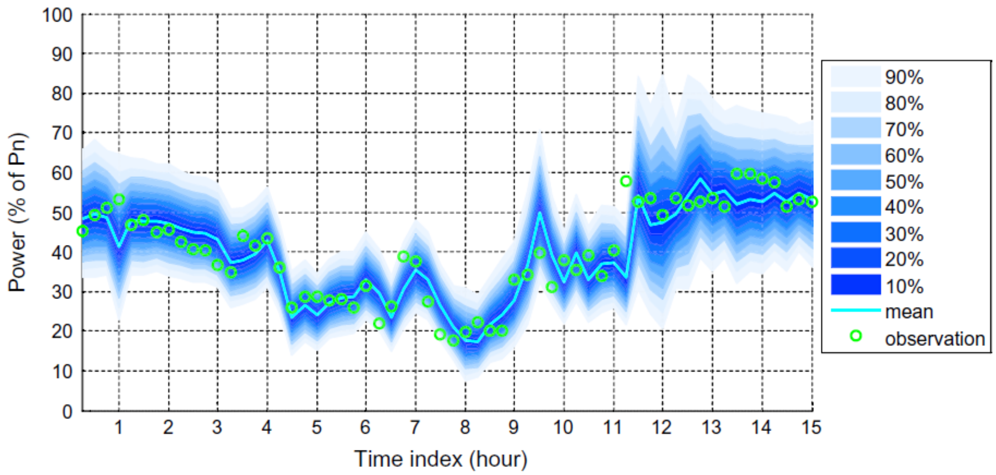

Figure 2.8: Results of one-hour ahead wind power forecasting using sparse online WGP [51]. This figure highlights not only the accuracy of the WGP method, but also the flexibility of WGP. The prediction intervals adjust to capture fluctuations in the data.

data while measuring the uncertainty of the unexplored area, i.e., what is the probability that an unexplored area would have new, valuable information.

Modern remote sensing provides a novel way to characterize important features of unexplored phenomena of interest [52]. The authors treat the combination of the available measurements and the probable uncertainty as an optimization problem. Parameters that estimate the cost of maneuvering are also incorporated. The authors evaluate this process in a simulated space exploration with test cases that emphasize and de-emphasize exploration. Case 1 favored exploration over data refinement, Case 2 placed less emphasis on exploring the whole field of interest, Case 3 decreases emphasis on exploration over time as data is collected, and Case 4 made emphasis on exploration dependent on the cost to perform maneuvering. Best performance occurred under conditions that favor exploration.

The autonomous exploration of unknown phenomena is further investigated via a process termed "informative planning" [53]. In this case, the focus is on long-term planning, where the computational expense remains a continuous problem as data accumulates. Therefore, the authors use Sparse Online Gaussian Processes (SOGP) to reduce high computational costs. Online learning is pertinent here as the SOGP needs to incorporate the temporal variation of the environment, re-estimating the hyperparameters when needed. This process needs spatio-temporal cues to justify the computational cost of any updates. The authors applied this method to a simulated ocean environment using real world salinity data. When hyperparameters were manually set, performance was weak, but data-driven selection improved performance and reflected ground truth.

2.7.2 Energy Harvesting

Energy harvesting is crucial in remote sensing operations. EH is defined as the process of using energy from external sources (i.e. wind, solar, etc.) and storing it as usable electricity [54]. A critical component of the AIMS project is the use of drones and sensors, which have a finite battery life. Charging stations, drones, sensors, etc. need to record battery level so that the system can make decisions accordingly. Drones need to retain enough battery to return to their charging stations and sensors need to indicate low battery life to the main control system to ensure their timely replacement. Adding stochastic processes to this system can create more efficient operation over drone and sensor battery life. Furthermore, electric field energy harvesting is of interest to power grid operations in general [55].

A novel method using an Age of Information (AOI) based status update system for a two-state stochastic process extends recent AOI research by considering the state of the stochastic process (normal or alarm) and the status changes unknown to the monitoring application [56]. The authors use an EH sensor that sends status updates to a destination node under the condition that the stochastic process will be in the normal state or the alarm state for a certain amount of time. At every time slot, the sensor generates a status update and then determines if the destination node and monitor are advised. If there is a state change, the system should leverage that with the AOI at the destination node to determine if an update is desirable given energy resources. Consideration of all these variables is a cost minimization problem to determine the best energy harvesting policy, achieved using a Markov Decision Process. The authors conclude that reserving energy is necessary to prepare for unexpected, long periods of an alarm state. In practical terms, status updates will be less frequent during a normal state and much more frequent during an alarm state. Figure 2.9 depicts the cost of different combinations of state transitions.

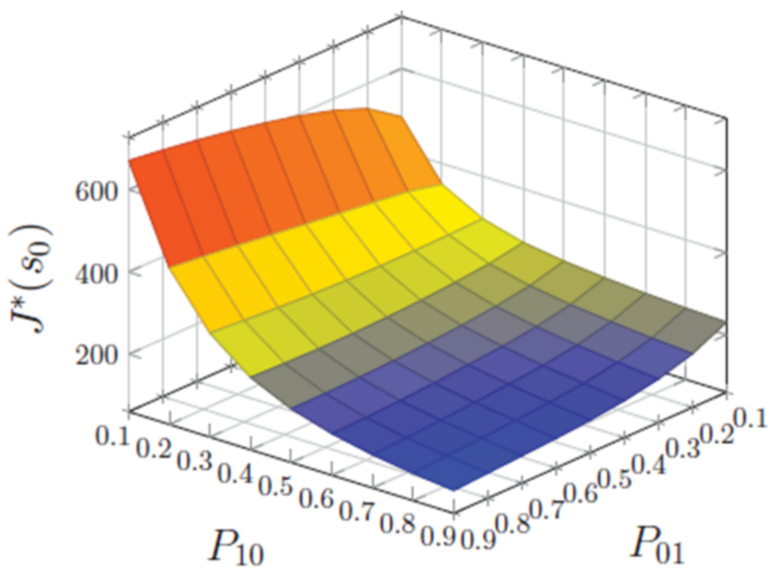

Figure 2.9: The impact of different combinations of stochastic process' state transition (P10 is normal, P01 is alarm) probabilities on the cost function, J∗(s0) [56]. Cost remains high during scenarios where there is a high probability of transitioning from a normal state into an alarm state. The lowest cost occurs when both the probability of transitioning to either state is high, suggesting that constant state transitions maintain low cost but are likely not collecting a large amount of information.

Chapter 3

Methodology

3.1 The Warped Gaussian Process

3.1.1 The Gaussian Process

As the name suggests, the Warped Gaussian Process (WGP) is built upon the traditional Gaussian process (GP). Therefore, in order to explain the WGP, I will begin by mathematically defining a GP. A portion of this section draws from my prior publication, "A Warped Gaussian Process Tutorial using a Simple Example" [57]. As stated in Chapter 1, a GP is a collection of random variables that have a joint Gaussian distribution [17]. If $\mathbf{X}$ is a vector of random variables that follow a Gaussian distribution, then $\mathbf{X} \sim \mathcal{N}(\mu, \Sigma)$ where μ is the mean and Σ is the covariance matrix. $\mathbf{X}$ can be written in vector notation as $X = \begin{bmatrix} X_1 & X_2 & \dots & X_n \end{bmatrix}^T$ where N is the number of input vectors. It follows that the corresponding function values $f(x) = \begin{bmatrix} f(x_1) & f(x_2) & \dots & f(x_n) \end{bmatrix}^T$ also follow a Gaussian distribution [58], so $f(\mathbf{x})$ is a Gaussian process. The GP, $f(\mathbf{x})$, is defined as follows:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \tag{3.1}$$

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] \tag{3.2}$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] + \delta_{ij}\sigma^2 \tag{3.3}$$

where $m(\mathbf{x})$ is the mean function and $k(\mathbf{x}, \mathbf{x}')$ is the covariance function, also called a kernel. δ is the Kronecker delta (see Appendix A). This definition assumes independent and identically distributed (i.i.d.) Gaussian noise, denoted as $\epsilon \sim \mathcal{N}(0, \sigma^2)$. The joint distribution of $f(\mathbf{x})$ and some function value $f(\mathbf{x}_*)$ at test point $\mathbf{x}_*$ can be written as

$$\begin{bmatrix} f(\mathbf{x}) \\ f(\mathbf{x}_*) \end{bmatrix} \sim \mathcal{N}(m(\mathbf{x}_*), \begin{bmatrix} K(X, X + I\sigma_n^2) & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*). \end{bmatrix}) \tag{3.4}$$

where the $I\sigma_n^2$ term represents the i.i.d. noise. The conditional probability is defined as follows:

$$f(\mathbf{x}_*)|X, f(\mathbf{x}), X_* \sim \mathcal{N}(m(\mathbf{x}_*), K(X, X_*)) \tag{3.5}$$

$$\bar{\mathbf{f}}_* = \mathbb{E}[f(\mathbf{x}_*)|X, f(\mathbf{x}), X_*] = K(X_*, X)[K(X, X + I\sigma_n^2)]^{-1} f(\mathbf{x}) \tag{3.6}$$

$$cov(\mathbf{f}_*) = K(X_*, X_*) - K(X_*, X)[K(X, X + I\sigma_n^2)]^{-1} K(X, X_*) \tag{3.7}$$

where $\bar{\mathbf{f}}_*$ is the mean and $cov(\mathbf{f}_*)$ is the covariance. One method for determining the best model hyperparameters is done by maximizing the log marginal likelihood function. The marginal likelihood, or the probability of y given X is defined below:

$$p(\mathbf{y}|X) = \int p(\mathbf{y}|\mathbf{f}, X) p(\mathbf{f}|X) d\mathbf{f}. \tag{3.8}$$

Taking the log of the previous equation results in the log marginal likelihood:

$$\log p(\mathbf{y}|X) = -\frac{1}{2} y^T K(X, X + I\sigma_n^2)^{-1} y - \frac{1}{2} \log|K(X, X + I\sigma_n^2)| - \frac{n}{2} \log(2\pi). \tag{3.9}$$

Solving the integral in Eqn. 3.8 not straightforward. The full derivation can be found in [17]. Recall that GPs are considered computationally expensive. The derivation shows why: solving for mean and covariance requires matrix inversion, which is computationally demanding. There are several methods to address this that were outlined in Chapter 2, but this is one of the main drawbacks of using a GP.

3.1.2 The Warped Gaussian Process

In this section I will introduce the Warped Gaussian Process. This method uses a warping function to project the observation space into a latent space in such a way that the data is well-modeled by a Gaussian distribution. In order to warp the observation space target values ($\mathbf{y_N}$) to latent space, let $\mathbf{z_N}$ be a vector of latent target values modeled by a GP. The monotonic function f maps all the entries from the actual target space to the latent space, as defined below:

$$z_n = f(y_n; \Psi) \tag{3.10}$$

where Ψ denotes the set of parameters for the warping function f. The only restriction on the function f is that it must be monotonic or the probability distribution will not be valid over $f(\mathbf{x})$ [19]. Going forward, the hyperbolic tangent function will serve as the warping function:

$$f(y, \Psi) = y + \sum_{i=1}^{I} a_i \tanh(b_i(c_i + y) \tag{3.11}$$

where $a_i, b_i \geq 0 \; \forall i$. This function generates smooth steps over the target values. In this case, a represents the magnitude, b represents the slope, and c represents the position of the steps. The number of hyperparameters a, b, and c is determined by how many iterations, I, of the warping function are used in the transformation. The number chosen must be weighed against the computational complexity that the warping function will produce. I used only the number of iterations necessary to improve upon the standard GP, either $I = 2$ or $I = 3$.

Continuing the warping process, the log marginal likelihood undergoes the transformation, thus $\log p(\mathbf{z}_N|\mathbf{X}_N, \Theta)$ and $\log p(\mathbf{y}_N|\mathbf{X}_N, \Psi)$ are defined below:

$$\log\, p(\mathbf{z_N}|\mathbf{X}_N, \Theta) = -\frac{1}{2}\mathbf{z}_n^T K(X, X + I\sigma_n^2)^{-1}\mathbf{z}_n - \frac{1}{2}\log|K(X, X + I\sigma_n^2)| - \frac{n}{2}\log(2\pi) \quad (3.12)$$

$$\log\, p(\mathbf{y_N}|\mathbf{X}_N, \Psi) = -\frac{1}{2}f(\mathbf{y}_n)^T K^{-1} f(\mathbf{y}_n) - \frac{1}{2}\log|K| + \sum_{n=1}^{N}\log\frac{\partial f(t)}{\partial t}|_{t_n} - \frac{n}{2}\log(2\pi) \quad (3.13)$$

where $K = K(X, X + I\sigma_n^2)$ and the Jacobian accounts for the warping. (The Jacobian is defined in more detail in Appendix A). Now, prediction in the latent space moves forward with a standard GP as defined earlier in this chapter - the latent space targets, $\mathbf{z_n}$ are predicted using $\mathbf{X_N}$ with GP parameters Θ. To return the data to the observation space, the targets $\mathbf{z}_N$ are passed back through the warping function to predict the original target space, $\mathbf{y}_n$:

$$P(y_{N+1}|x_{N+1}, \Theta, \Psi) = \frac{f'(y_{N+1})}{\sqrt{2\pi\sigma_{N+1}^2}}\exp[-\frac{1}{2}(\frac{f(\mathrm{y_{N+1}}) - \mathrm{z_{N+1}}}{\sigma_{\mathrm{N+1}}})^2] \quad (3.14)$$

As seen from Eqn. 3.14, predicting the observation space targets $\mathbf{y_n}$ from $\mathbf{X_N}$ depends on the parameters from the latent space GP, Θ, and the parameters from the warping function, Ψ. Recall Ψ contains the hyperparameters a, b, and c which control the shape of the warping function. The GP parameters Θ depend on kernel choice, for example, the length scale, l, is a hyperparameter that controls the smoothness of the GP. Kernels will be discussed in more detail in the next section.

WGP provides the complete probability distribution with the equation above and can be used to make point predictions for the median by inverting the warping function directly. I used the median for prediction, so all forecasting loss functions are based on mean absolute error (MAE) [59]:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \quad (3.15)$$

where n is the number of data points, y_i is the actual value, $\hat{y}_i$ is the predicted value, and $|y_i - \hat{y}_i|$ is the absolute error for each data point. Figure 3.1 shows the basic code for a WGP in Python for $I = 2$.

3.1.3 Kernels

The kernel function is the heart of the GP. There are several well-defined kernel functions, and kernels can be combined and scaled to best model the data [60]. The choice of kernel will not only affect the outcome of the model, but also the associated computational complexity. The best choice for kernel function will depend on domain knowledge, the underlying data structure, and hyperparameter optimization. In this paper, I used single, simple kernels to compare the basic performance of GP vs. WGP. Some well-known kernels that will be referenced later are defined below.

White Noise (WN):

$$k_{WN}(x_i, x_j) = \sigma^2 \delta(i, j) \tag{3.16}$$

where σ^2 is the variance and δ is the Kronecker delta. As its name suggests, this kernel models the i.i.d. Gaussian noise referenced earlier in this chapter.

Squared Exponential (SE):

$$k_{SE}(x_i, x_j) = \sigma^2 \exp[-(\frac{x_i - x_j}{l^2})^2] \tag{3.17}$$

where l is the length scale. This is also known as the Radial Basis Function (RBF) kernel, which is how I refer to it in this paper. This is one of the most commonly used kernels for GP analysis in the literature (see Chapter 2).

Rational Quadratic (RQ):

$$k_{RQ}(x_i, x_j) = \sigma^2 (1 + \frac{(x_i - x_j)^2}{\alpha l^2})^{-\alpha} \tag{3.18}$$

```python
I = 2
a = b = c = [0.5] * I
def warping_func(y, a, b, c, I):
    tanh_sum = 0
    for i in range(0, I):
        tanh_sum += a[i]*np.tanh(b[i] * (y + c[i]))
    z = y + tanh_sum
    return z
space = [
    Real(1e-10, 10, name='length_scale'),
    Real(0, 2, name='a1'),
    Real(0, 2, name='b1'),
    Real(-2, 2, name='c1'),
    Real(2, 2, name='a2'),
    Real(2, 2, name='b2'),
    Real(-2, 2, name='c2')
    ]
def objective(params):
    length_scale, a1, b1, c1, a2, b2, c2 = params
    gp = GaussianProcessRegressor(RBF(length_scale))
    a = np.array([a1, a2])
    b = np.array([b1, b2])
    c = np.array([c1, c2])
    warped_y = warping_func(y_tr, a, b, c, I)
    gp.fit(x, z)
    return -gp.log_marginal_likelihood()

result = gp_minimize(objective, space, n_calls=10)

best_length_scale = result.x[1]
best_a1 = result.x[2]
best_b1 = result.x[3]
best_c1 = result.x[4]
best_a2 = result.x[5]
best_b2 = result.x[6]
best_c2 = result.x[7]

best_a = np.array([best_a1, best_a2])
best_b = np.array([best_b1, best_b2])
best_c = np.array([best_c1, best_c2])

z = warping_func(y_tr, best_a, best_b, best_c, I).flatten()
gp = GaussianProcessRegressor(RBF(best_lenght_scale))
gp.fit(x, z)

z_hat = gp.predict(x)
inv_warp = inversefunc(warping_func, args=(best_a, best_b, best_c, I))
y_hat = inv_warp(warped_normalized.reshape(-1,1)).flatten()
```

Figure 3.1: Basic code for a WGP in Python.

where α is the index, and l is the length scale. The RQ kernel is built off of the SE, and both are suitable to smooth functions.

Matérn (M):

$$k_M(x_i, x_j) = \frac{2^{1-\nu}}{\Gamma(\nu)}(2\sqrt{\nu}\frac{|x_i - x_j|}{l})\mathbb{B}_\nu(2\sqrt{\nu}\frac{|x_i - x_j|}{l}) \tag{3.19}$$

where Γ is the standard Gamma function and $\mathbb{B}$ is the modified Bessel function of second order, ν is the degree of differentiability (typically, $\nu = 3/2, 5/2$), and l is the length scale.

Periodic (P):

$$k_P(x_i, x_j) = C^2\text{exp}(-\frac{2sin^2(\frac{\pi|x_i - x_j|}{p})}{l^2} \tag{3.20}$$

where p is the period and l is the length scale. Aptly named, this kernel is useful for modeling periodic functions.

3.1.4 Time-series analysis

Several of the following analyses will rely on time-series forecasting as the basis for modeling. Classical statistics methods like Autoregressive integrated moving average (ARIMA) and seasonal ARIMA and GP regression are well represented in the literature. WGP offers a method to further investigate kernel choice in forecasting and maintains the statistical interpretability of classic methods. Time-series analysis under WGP is considered a regression problem and will follow the definition outlined earlier in this section. In time series analysis, the input vector is $\mathbf{x}_t = \begin{bmatrix} x_{1,t} & x_{2,t} & \dots & x_{n,t} \end{bmatrix}^T$, representing the i-th variable at time t and the target values y_t are the time-series at time t. ARIMA is an established, widely used statistical method for time-series analysis. ARIMA is denoted as ARIMA(p,d,q) where p is the lag order, d is the degree of differencing, and q is the order of the moving average. Vector Autoregression (VAR) is a generalization of ARIMA for multivariate data. For a detailed explanation of ARIMA and VAR, see Chapter 2 of [61].

3.2 Power Grid Simulation

In this section I will describe the methods and software packages to simulate the power grid datasets. The Institute of Electrical and Electronics Engineers (IEEE) provides different bus test cases based on the American Electric Power system. These standardized cases are commonly used in the literature for power analysis and allow for consistent comparative analysis. The IEEE 14 bus system [62] has 14 buses, 11 loads, and 5 generators. Figure 3.2 shows a graphical depiction of this system. Associated data tables can be found in Appendix 3.

The IEEE Bus 14 test case was implemented using Python's Pandapower to simulate power grid data. Pandapower is an open-source Python library for power system analysis and optimization built on top of Pandas and NumPy [63] to model transmission and distribution systems specifically. All of the data used in the pandapower network is recoverable as pandas dataframes. This output includes the following data points:

1. Voltage magnitude and angle at each bus
2. Voltage estimates from state estimation
3. Power flow through external grid elements
4. Generator active/reactive power output
5. Static generator power output
6. Power consumption by loads
7. Reactive power from shunts
8. Results from storage elements
9. Power flow, current, and loading of lines
10. Power flow and loading in 2-winding transformers
11. Power flow in 3-winding transformers

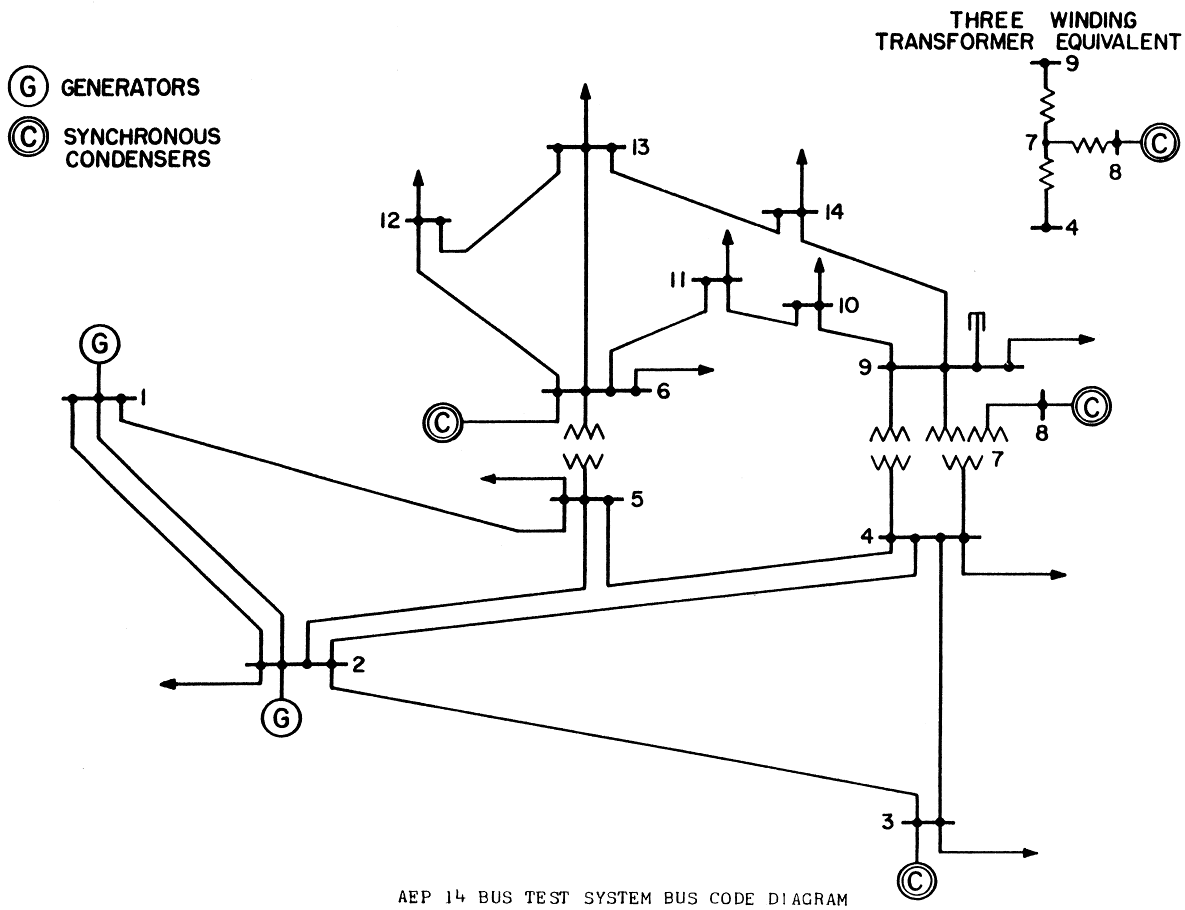

Figure 3.2: Graphical depiction of the IEEE 14-Bus test case system [62].

12. Results for impedance elements
13. Results for switches (open/closed)
14. Results from DC lines

The combined data from all these elements are usually about 100 rows by 32 columns. I used Pandapower to generate grid data for approximating the OPF equations, estimating parameters, and load forecasting. The code can be found in Appendix F.

3.2.1 The Optimal Power Flow Solution

There are several methods in the literature (Chapter 2) for optimal power flow (OPF) solutions, including GPR. However, WGP has not been applied to estimating the OPF solution. The equations associated with OPF are non-linear and complex, therefore, WGP may be better suited to this problem than GP due to its increased flexibility [29].

The AC Optimal Power Flow (ACOPF) problem aims to minimize the total cost while satisfying the necessary constraints [64]. The ACOPF objective function is

$$\min_{\mathbf{P},\mathbf{Q}} \sum_{i=1}^{n} C_i(P_i) \tag{3.21}$$

where $\mathbf{P} = [P_1, P_2, \ldots, P_n]$ is the vector of active power generation, $\mathbf{Q} = [Q_1, Q_2, \ldots, Q_n]$ is the vector of reactive power generation, and $C_i(P_i)$ is the cost function associated with generator i.

The power balance equations ensure that the power injected at each bus matches the power consumed and lost in the transmission network:

$$P_i = \sum_{j=1}^{n} |V_i||V_j|(G_{ij}\cos(\theta_{ij}) + B_{ij}\sin(\theta_{ij})) \tag{3.22}$$

$$Q_i = \sum_{j=1}^{n} |V_i||V_j|(G_{ij}\sin(\theta_{ij}) - B_{ij}\cos(\theta_{ij})) \tag{3.23}$$

where P_i and Q_i are the active and reactive power injections at bus i, G_{ij} and B_{ij} are the real and imaginary parts of the admittance between buses i and j, and θ_{ij} is the phase angle difference between buses i and j.

The inequalities below represent the multiple constraints in this problem: 1) that there are minimum and maximum active and reactive power generation limits for each generator, 2) that all voltage magnitudes at all buses are within acceptable bounds, and 3) that the phase angle differences between adjacent buses is limited.

$$P_{\min,i} \leq P_i \leq P_{\max,i} \tag{3.24}$$

$$Q_{\min,i} \leq Q_i \leq Q_{\max,i} \tag{3.25}$$

$$V_{\min} \leq |V_i| \leq V_{\max} \tag{3.26}$$

$$\theta_{\min,ij} \leq \theta_{ij} \leq \theta_{\max,ij} \tag{3.27}$$

A recurring theme in the OPF literature is finding stochastic methods that can model this problem without having to rely on a Gaussian representation of the data and/or noise. WGP is well-suited to this problem because it can model non-Gaussian problems and captures noise well. Optimizing through WGP can be achieved multiple ways [65, 66]. A general representation of the optimization problem constraint is given by

$$\begin{aligned} \sum_i \tilde{a}_i x_i &\leq b \\ \tilde{a}_i &= g(\mathbf{z}_i) \end{aligned} \tag{3.28}$$

where x_i is a decision variable and $\tilde{a}_i$ depends on the vector of decision variables $z_i \in \mathbb{R}^k$ through some function g (in a WGP context, g is the warping function). The set $\mathcal{U}$ contains the constraints on the $\tilde{a}_i$ parameters. Adapting the constraint problem for WGP optimization [66] gives

$$\min_{x,z\in X} f(\mathbf{x},\mathbf{z})$$
$$\text{s.t.} \sum_i h^{-}1(\mu(z_i))x_i \leq b, i \in [n] \tag{3.29}$$

where $\mathbf{z}$ is the vector of decision variables z_i $\forall i$, f is a known function, and $h^{-}1(\mu(z_i))$ is the inversely warped mean prediction of the GP. An α-confidence ellipsoid ε^{α} can be constructed in latent space:

$$\varepsilon^{\alpha}(\mathbf{z}) = \{\boldsymbol{\xi} : (\boldsymbol{\xi} - \boldsymbol{\mu}(\mathbf{z}))^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{z})(\boldsymbol{\xi} - \boldsymbol{\mu}(\mathbf{z})) \leq F_n^{1-\alpha}\} \tag{3.30}$$

where $F_n^{1-\alpha}$ is the chi-squared cumulative distribution with n degrees of freedom and $\boldsymbol{\Sigma}$ is the covariance matrix. Then the optimization problem can be written as

$$\min_{x,z\in X} f(\mathbf{x},\mathbf{z})$$
$$\text{s.t. } \mathbf{y}^{\mathbf{T}}\mathbf{x} \leq b, \;\; \forall \mathbf{y} : h(\mathbf{y}) \in \varepsilon^{\alpha}(\mathbf{z}) \tag{3.31}$$

Rewriting in terms of the uncertainty set $\mathcal{U}$, the optimization problem can be written as

$$\min_{x,z\in X} f(\mathbf{x},\mathbf{z})$$
$$\text{s.t. } \mathbf{y}^{\mathbf{T}}\mathbf{x} \leq b, \;\; \forall \mathbf{y} \in \mathcal{U}(\mathbf{z}) \tag{3.32}$$

$$\text{where } \mathcal{U}(\mathbf{z}) = \{y \in \mathbb{R}^l : (h(\mathbf{y}) - \boldsymbol{\mu}(\mathbf{z}))^{\mathbf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{z})(h(\mathbf{y}) - \boldsymbol{\mu}(\mathbf{z})) \leq F_n^{1-\alpha}\} \tag{3.33}$$

Figure 3.3 shows a representation of these spaces.

In the literature, GP has been used successfully as a surrogate model for the OPF equations (see Chapter 2). I compared data-driven surrogate models using GP and WGP to estimate the solution to the OPF equations. Once again, the pandapower IEEE 14-Bus system was used to generate the data (100 runs producing 32 outputs per run). The network data was extracted for training and testing, and Pandapower's internal OPF function comprised the true values. Pandapower's OPF function uses PYPOWER, a Python port of

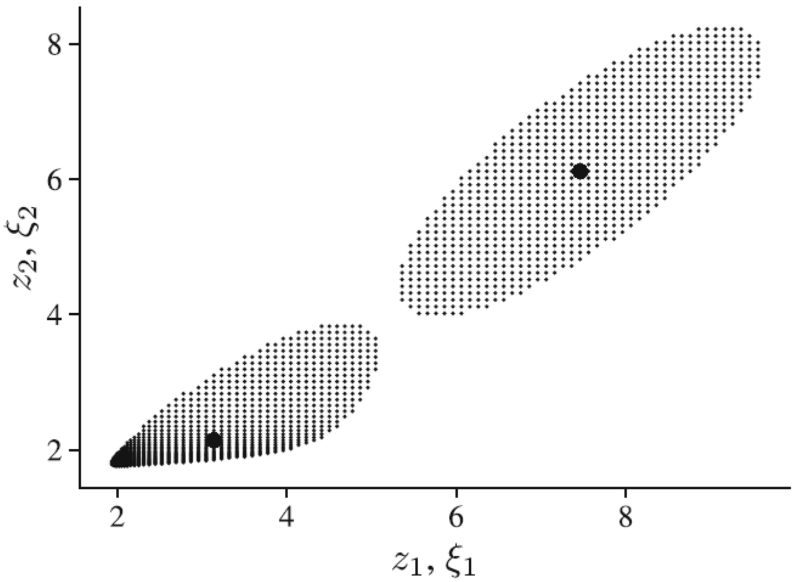

Figure 3.3: Graphical depiction of ε^{α} in latent space and $\mathcal{U}$ in observation space [66].

MATPOWER to solve the OPF equations. This ensures that the constraints discussed in this section are maintained. The benefit of using a surrogate model is reduced computational time as the the OPF solution does not need to be directly solved. However, this data-driven approach means that the GP and WGP surrogate models do not enforce constraints as in the direct OPF equations. The results of this analysis can be found in Chapter 4.

3.2.2 Parameter Estimation

The power grid is a complex system that depends on multiple variables for successful operation. In some power grid system with a set of $\mathcal{N}$ buses, $\mathcal{N} = \{1 \dots N\}$ with generator buses $\mathcal{N}_g$ and load buses $\mathcal{N}_l$, the following represent some of the applicable system variables:

- V_i: Voltage magnitude at bus i.
- θ_i: Voltage phase angle at bus i.
- ω_i: Rotor frequency deviation at bus i.
- P^{M}: Mechanical power input to generator i.
- P^{L}: Power load output from generator i.

- δ_i: Phase angle deviation at bus i.
- $\dot{\delta}_i$: Rate of change of phase angle at bus i.
- M_i: Inertia coefficient of generator i.
- D_i: Damping coefficient of generator i.

Using the variables above, the dynamic power system can be described by the following equations [67]

$$\dot{\delta}_i = \omega_i, i \in \mathcal{N}_g \tag{3.34}$$

$$M_i \dot{\omega}_i = P_i^M - \sum_{j \in \mathcal{N}} B_{ij} \sin(\delta_{ij}) - D_i \omega_i, i \in \mathcal{N}_g \tag{3.35}$$

$$D_i \dot{\delta}_i = -P_i^L - \sum_{j \in \mathcal{N}} B_{ij} \sin(\delta_{ij}), i \in \mathcal{N}_l \tag{3.36}$$

where B_{ij} is the susceptance of the line i, j and $\delta_{ij} = \delta_i - \delta_j$. The inertia and damping coefficients are particular interesting for parameter estimation. In the power grid, inertia is a physical property referring to the energy stored from the rotating generators. This helps maintain reliability by responding to frequency imbalances [68]. Damping, which occurs from mechanical and electrical losses, is critical to maintaining stability [69, 70]. Both are physics-based parameters that are non-linear, non-stationary, and crucial to power grid stability, making WGP regression a suitable and underused method [71]. I extracted the data from a pandapower IEEE 14-bus system to simulate the swing equations in Python (Pandapower code can be found in Appendix F). Over time (200 data points over 5 seconds), I tracked the rotor angular deviation and angular velocity deviation to extract dynamic features: the maximum and final rotor angular deviations and angular velocity deviations. The results of this analysis can be found in Chapter 4.

The power flow equations are not the most practical way to estimate parameters in a power grid setting. Measurements are noisy and may not be available at every grid element at any given time. State estimation (SE) addresses this problem through a measurement model to estimate states. This model, in total, consists of five different functions: Measurement pre-filtering, topology processor, observability analysis, state estimation, and bad data processor [72]. Various statistical methods are applied within this problem, including weighted least squares, chi-squared and residual tests, etc. The full problem definition can be found in Chapter 4 of [72]. Generally, consider an input vector m that contains a set of measurements for each system state:

$$m = \begin{bmatrix} m_1 \\ m_2 \\ \dots \\ m_m \end{bmatrix} = \begin{bmatrix} h_1(x_1, x_2, \dots, x_n) \\ h_2(x_1, x_2, \dots, x_n) \\ \dots \\ h_m(x_1, x_2, \dots, x_n) \end{bmatrix} \tag{3.37}$$

where h_m is a non-linear function of a system state vector x for the i_{th} measurement m_i. There is error, e associated with each measurement, thus $m = h(x) + e$. It should be immediately apparent that this is collection of functions, suitable to WGP analysis.

Other methods, including GPR, are used in the literature for SE, but WGP has not been applied to this problem. WGP is well-suited to SE because it can handle non-linear and noisy data. Managing a SCADA system with WGP also benefits from its interpretability.

3.2.3 Load Forecasting

Electricity demand has peak and non-peak hours, meaning efficient operation relies on properly identifying these hours. Previous research applications in the literature have used traditional GPs or other stochastic methods to model power demand successfully, but WGP is underrepresented in this area, particularly in transmission and distribution applications. WGP is suitable for this problem given that alternate current power flow is usually modeled with non-linear equations and demand forecasting often requires using non-stationary, noisy data. The focus of this research is short-term prediction (24-hours ahead).

The system uses a distributed slack to ensure every generator contributes and one controller for load scaling. One week of data was simulated with one data point per hour for 168 data points total. Figure 3.4 shows the total load power over the week, Figure 3.5

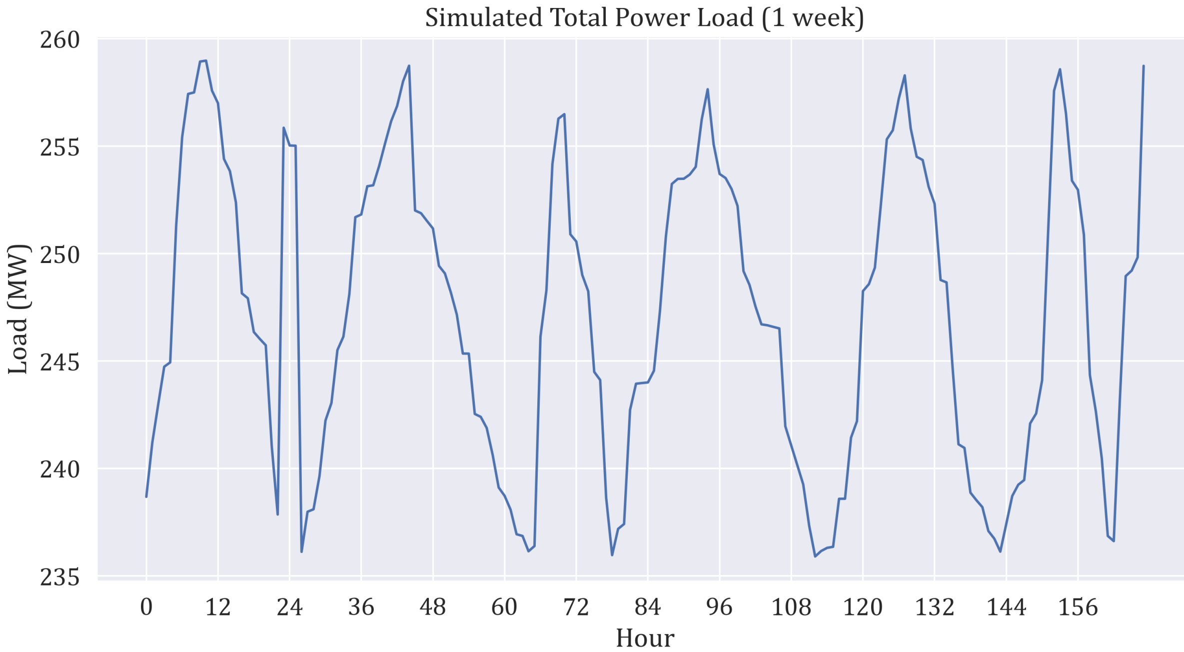

Figure 3.4: Simulated total power load for one week.

shows the generator output, Figure 3.6 shows voltage for each bus, and Figure 3.7 shows the line loading.

The total power load (Figure 3.2) generated from the pandapower IEEE 14-Bus network shows an increase in load in the early hours of each day with a peak in the afternoon/evening, and then a decrease in the night, which is expected. The data show general daily seasonality well suited to time series analysis, however, there is noise present and the peak hour is not the same every day. Summary statistics for total power load are shown in Table 3.1. I compared an ARIMA model, a GP model, and a WGP model to analyze the load profile time series and assess each model's predictive power. As mentioned earlier in this chapter, time series analysis with WGP is a regression problem where the time step t acts as the x variable, in reference to the basic WGP code (Figure 3.1). The results of this analysis can be found in Chapter 4.

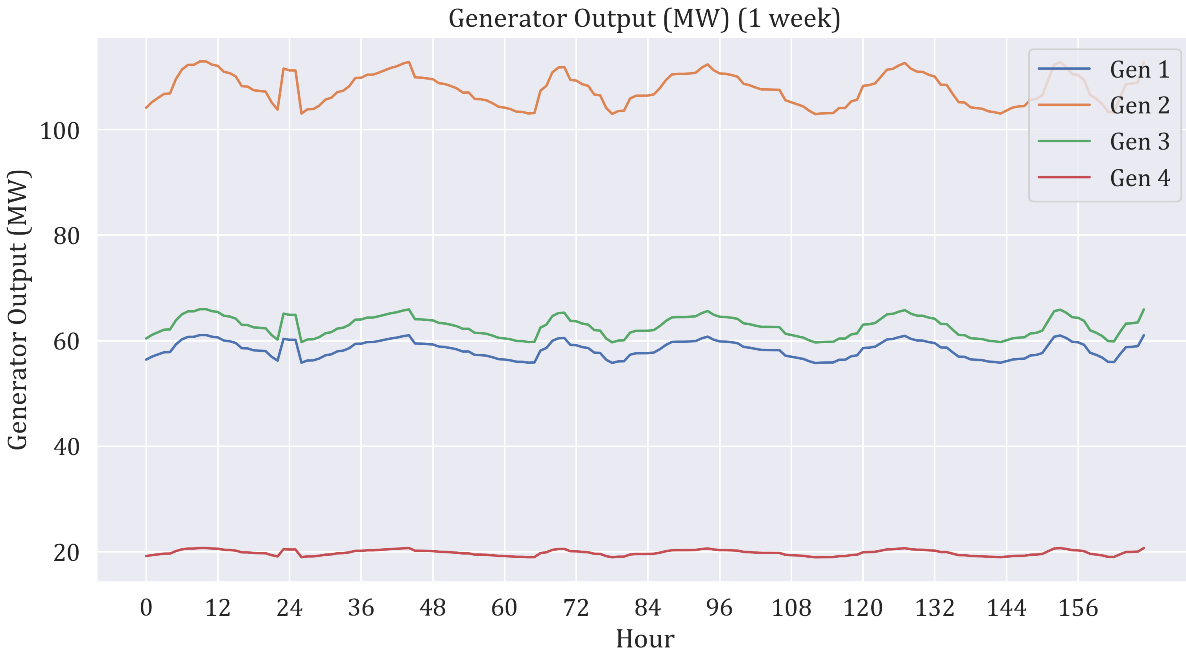

Figure 3.5: Simulated generator loads for one week.

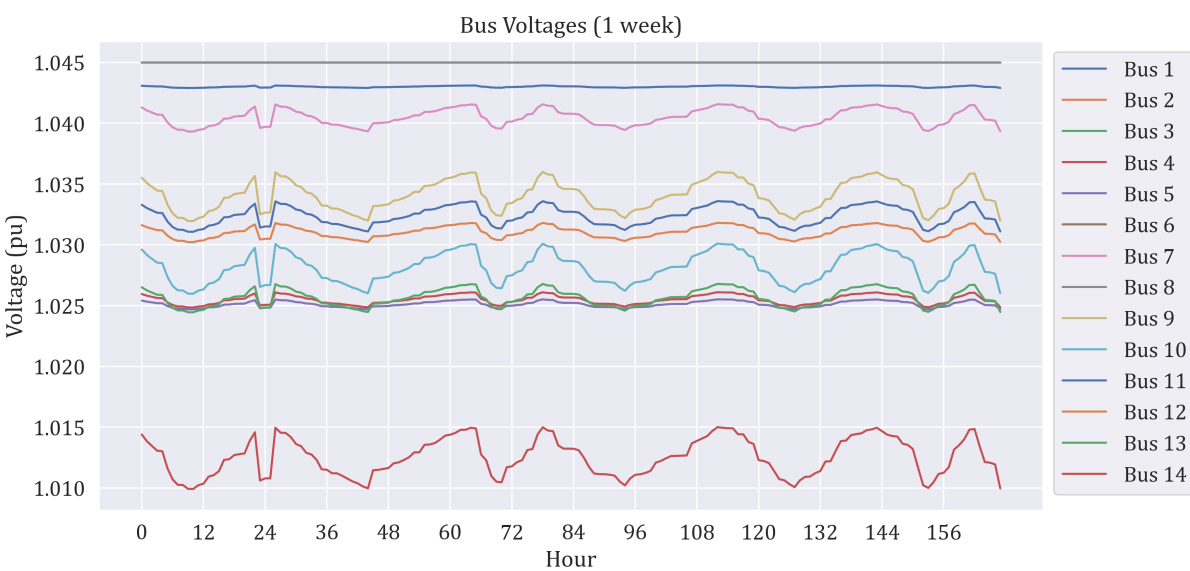

Figure 3.6: Simulated bus voltages for one week.

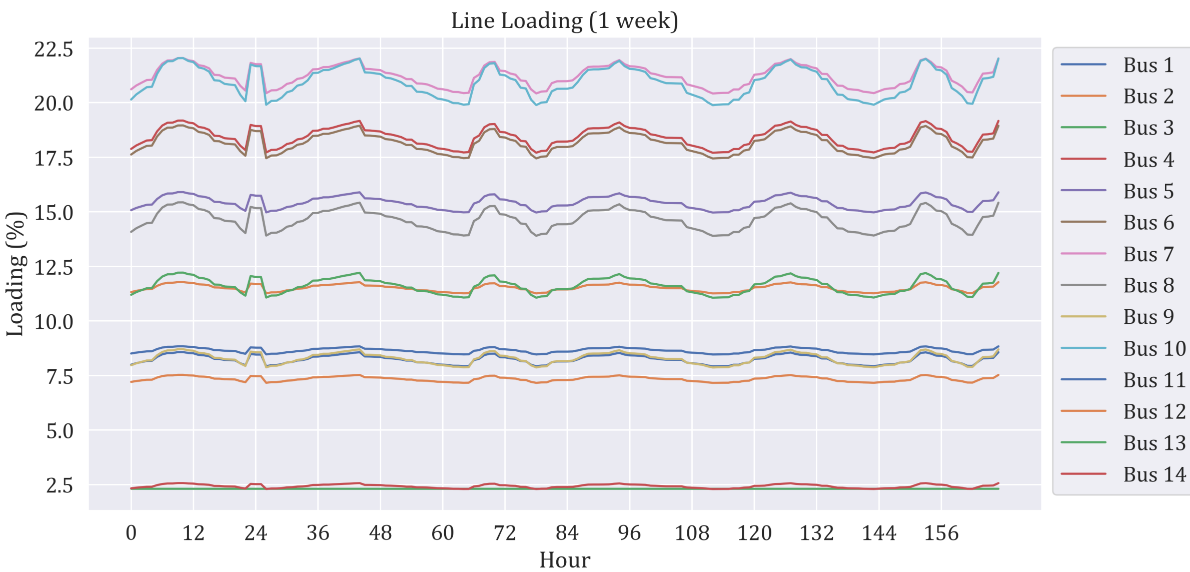

Figure 3.7: Simulated line loading for one week.

Table 3.1: Summary statistics for total power load (MW).

	Total Load (MW)
min	235.8910
25th percentile	241.0552
mean	247.1071
75th percentile	253.2757
max	258.9776
std	6.9663

3.3 Signal Analysis

One key aspect of the AIMS project is the deployment of drones during some trigger event to assess the situation. These "triggers" will likely be the result of some anomaly that has been assessed by the SCADA system. One type of anomaly that can be measured in a power grid is electrical signal distortion [31]. Continuous or transient disturbances can be modeled and analyzed through signal analysis, and these types of disturbances can be thought of as anomalies. Signal analysis in power grid operations is challenging due to the non-linearity and non-stationary nature of the data, not to mention the large number of contributing variables (voltage, current, phase angle, etc.). If a signal is represented by a waveform, $x(t)$, then the detection equations can be represented as follows [73]:

$$x_a(t) = x(t) + j\mathcal{H}[x(t)] \tag{3.38}$$

$$= A(t)e^{j\phi(t)} \tag{3.39}$$

where $x(t)$ is the real-valued signal, $j = \sqrt{-1}$, $\mathcal{H}[\cdot]$ is the Hilbert transform, $A(t)$ is the instantaneous amplitude, and $\phi(t)$ is the instantaneous phase. The instantaneous amplitude can be rewritten for envelope detection as

$$A(t) = \sqrt{x(t)^2 + \mathcal{H}[x(t)]^2} \tag{3.40}$$

and the instantaneous phase is defined below:

$$\phi(t) = \tan^{-1}\left(\frac{\mathcal{H}[x(t)]}{x(t)}\right). \tag{3.41}$$

The Fast Fourier transform (FFT) is calculated with the following equation:

$$FFT_k = \sum_{n=0}^{N-1} e^{-j2\pi \frac{kn}{N}} x_n \tag{3.42}$$

where FFF_k is the k-th frequency component of the transformed signal, x_n is the n-th sample of the input signal, and $e^{-j2\pi nk/N}$ represents the complex exponential representing the k-th frequency basis function. The FFT is an efficient way to calculate the Discrete Fourier Transform, a technique useful for analyzing signals in the frequency domain. More information on DFT can be found in [72].

It is common to model power transmission under the assumption that it is balanced [74], however, this is not a guarantee in practice. WGP is underutilized in signal analysis [75] since it can account for asymmetric and unbalanced data and is more suited for modeling noise than traditional GPs. SCADA is particularly relevant here. Each of these grid aspects separately (i.e. load, damping, inertia, signal, etc.) are important, but it is more useful to integrate the data so that there is feedback between different measurement systems. In the AIMS scenario, this is relevant to the drones "communicating" with each other to share information. With the increase in smart technologies like Phasor Data Concentrators [76], anomaly detection applications will likely see an influx of data that must be transformed into usable information.

The Grid Event Signature Library (GESL) [77] is a repository for power grid waveform data hosted by ORNL and the Lawrence Livermore National Laboratory under the Office of Electricity. The waveforms available are phasor measurement unit (PMU) and point-on-wave (PoW) data over different sampling rates, voltage levels, and measurement parameters. The PMU data analyzed from the GESL are events that can be categorized as anomalies due to lightning strikes, arcing, and fuse blows. There are 82 samples in total: 21 in the lightning strike class, 30 in the arcing class, and 31 in the fuse blow class. In order to expand this dataset for classification, I used the real world data to simulate synthetic signals. Each waveform was augmented by adding noise to the signal, applying random time shifts, and scaling the signal amplitude within a given range (.80-1.1). After this augmentation process, 332 total waveforms were used for classification. Figures 3.8 - 3.10 show examples

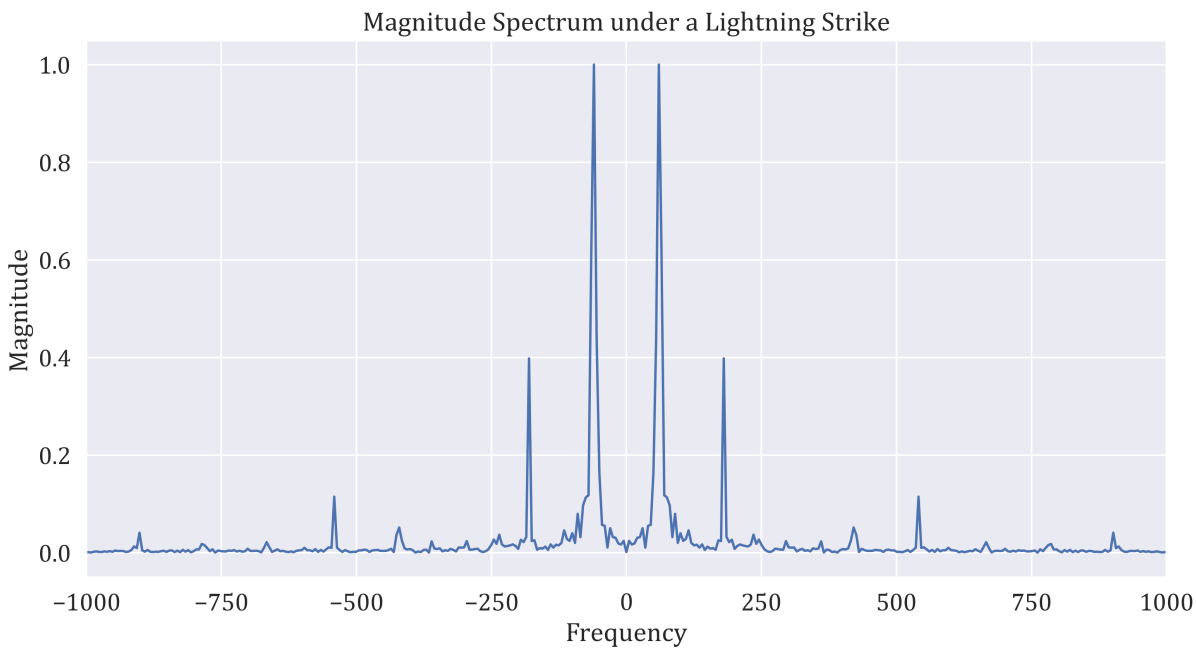

Figure 3.8: Magnitude spectrum for a signal under a lightning strike.

of anomalies and their effect on the magnitude spectrum. There is a clear distinction between all three, suggesting that feature extraction from signal analysis will aid in classification. I compared a GP and WGP filter for classifying anomaly grid events with a random forest classifier. Figure 3.11 shows the diagram for this process.

WGP time-series analysis is another avenue for detecting energy loss anomalies and arcing within a signal. Non-technical energy loss can be due to theft or faulty equipment [78] while arcing is characterized by a large release of energy [79]. In both cases, outlier detection and prediction are necessary, which can be accomplished by WGP analysis. Other methods (parametric and non-parametric) have been used to identify anomalies in time-series data, but WGP is not often used for time series analysis despite its advantages in analyzing non-stationary data, periodicity, and noise - which can hinder deep learning methods [80].

3.4 Load + Weather Forecasting

Climate change presents new challenges for power grid operation. Strong winds can damage infrastructure by causing galloping power lines, flying branches and debris, and downing power poles. Extreme high or low temperatures create significant demand on power grids.

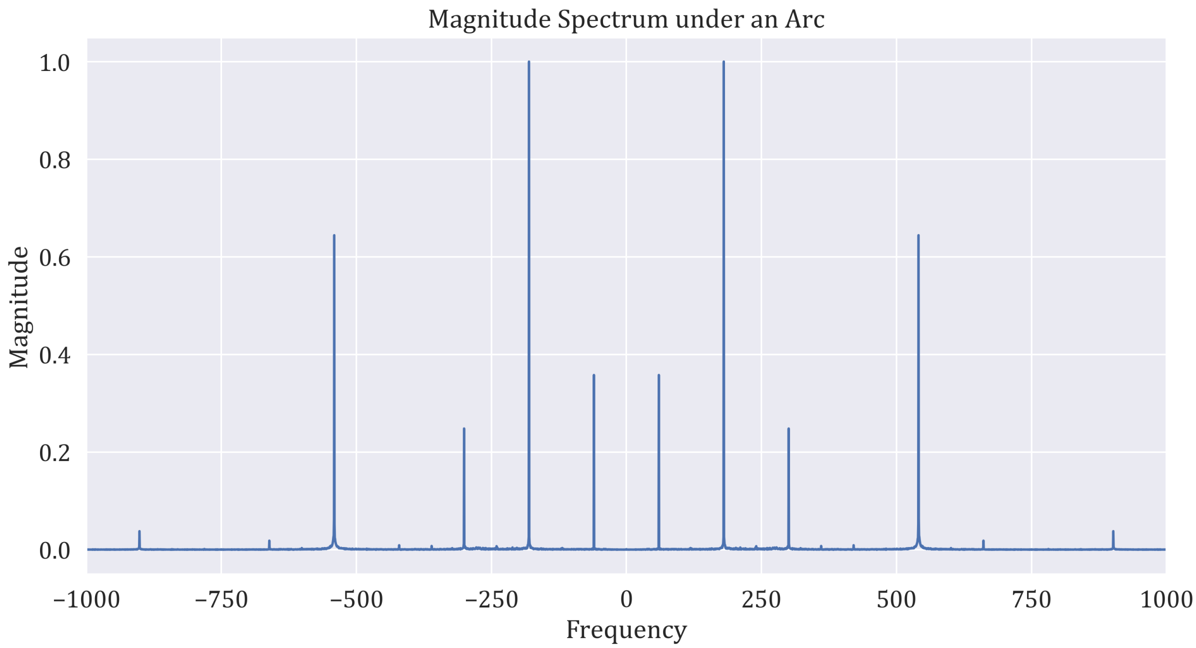

Figure 3.9: Magnitude spectrum for a signal under arcing.

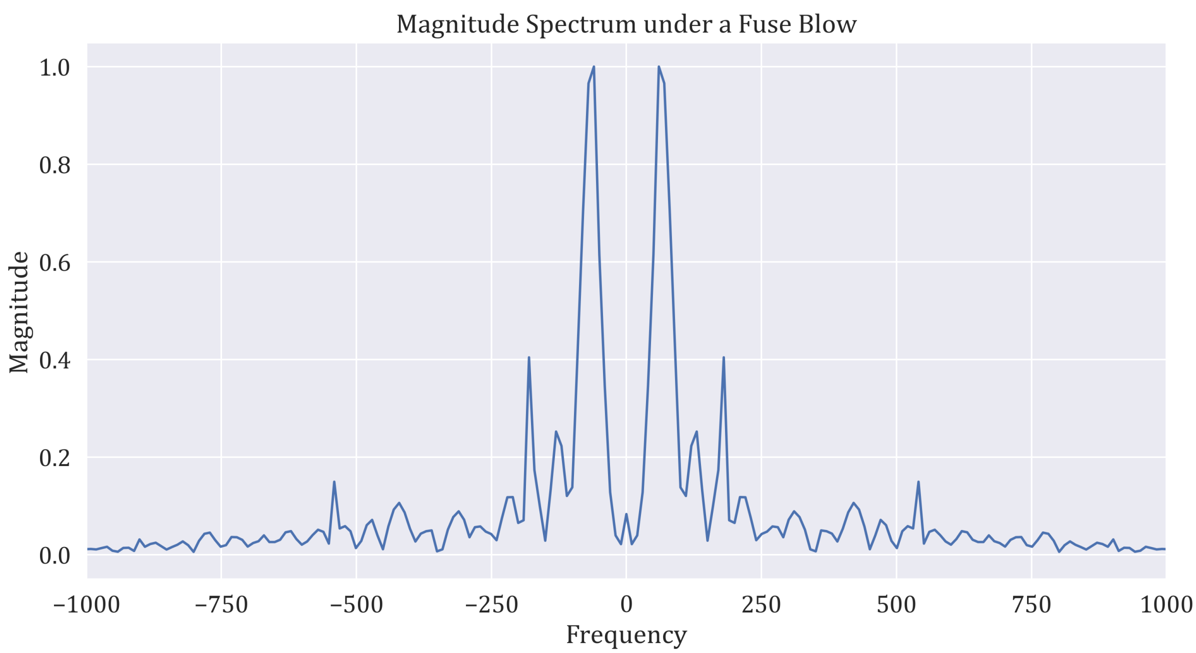

Figure 3.10: Magnitude spectrum for a signal under a fuse blow.

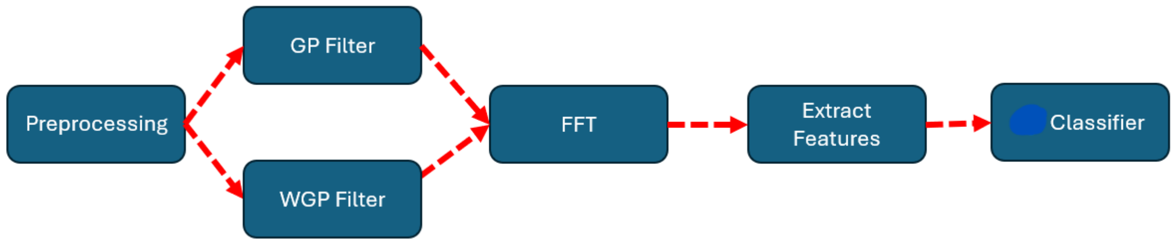

Figure 3.11: Diagram showing the process for classifying signal anomalies.

They can cause power lines to expand or contract due to their material composition. Weather variables are random in nature, therefore difficult to model. Since they do not follow existing distributions, WGP is particularly applicable since it creates a warping function based on empirical data. Moreover, how these variables are related and/or interact is of interest to grid operations, for example, load forecasting is influenced by weather. Wind speed is often used in forecasting temperature and vice versa. Previous research using deep learning methods to forecast weather have reported high accuracy, but do not show consensus on which kernels are most effective and success varied by location and season [81, 82].

Publicly available, well-maintained datasets with both weather and power load measurements are not readily available in the U.S. despite the close relationship between weather and grid operations [83]. I performed the time-series analysis on national electric load data from Panama's national utility, CND [84]. The Panama dataset contains hourly measurements of the national load in Panama, and the temperature, humidity, precipitation, and wind speed for three different cities: Tocumen, Santiago, and David. It also contains holiday category (holiday or no holiday), holiday id, and school category (school in session or out of session). I used the most recent data available for 24-hour ahead prediction (June 20, 2020 to June 25, 2020). Summary statistics for hourly weather data can be found in Table 3.2.

Correlation maps for the hourly weather data are shown in Figure 3.12 - 3.14. Each city shows moderate correlations between humidity and precipitation, which is expected. In all cities, temperature is more positively correlated with national load demand than the other weather variables. In general, the correlation maps differ quite a bit city to city, suggesting the weather and power are uniquely related to location.

Table 3.2: Summary statistics for national load demand and weather data for three different cities: Tocumen (T), David (D) and Santiago (S).

	Nat Load	Temp (T)	Humidity (T)	Precipitation(T)	Wind Speed (T)
min	997.9285	26.7056	0.0191	0.0387	4.6958
25th percentile	1052.3753	26.8453	0.0199	0.0606	8.6619
mean	1106.2777	27.6412	0.0200	0.0884	12.3878
75th percentile	1175.963100	28.5247	0.0203	0.1138	16.4929
max	1225.1657	28.8738	0.0205	0.1525	0.1525
std	73.3156	0.8441	0.0004	0.0349	4.7793
		Temp (D)	Humidity (D)	Precipitation(D)	Wind Speed (D)
min		23.0642	0.0177	0.1509	0.7792
25th percentile		23.3149	.0181	.1611	1.8949
mean		24.3112	0.0183	0.1920	2.3729
75th percentile		25.463075	0.0185	0.2203	2.9236
max		26.1293	0.0189	0.2458	3.1381
std		1.1378	0.0003	0.0321	0.7081
		Temp (S)	Humidity (S)	Precipitation(S)	Wind Speed (S)
min		24.7919	0.0188	0.0739	1.99078
25th percentile		25.1371	0.0191	0.0996	2.7075
mean		26.4440	0.0195	0.1114	3.9597
75th percentile		27.8826	0.0199	0.1225	5.5007
max		28.6414	0.0201	0.1539	6.4854
std		1.4326	0.0004	0.0179	1.5215

Figure 3.15 - 3.17 show the national load demand versus each weather variable. As was suggested by the correlation map, there is a positive trend between load demand and temperature. Each city has separation in terms of temperature and humidity, however, the trend between national load demand and humidity is not as clear. Precipitation is much less separable by city, and there is no clear relationship between that and load demand. Tocumen clearly experiences higher wind speeds than David and Santiago, but again there is no clear relationship between load demand and wind speed.

Figure 3.19 - 3.22 show the time series for each of the weather variables and 3.23 shows the time series for the national load demand. Temperature is showing very periodic behavior reflecting the rise and fall each day, but all of these variables are showing non-stationary behavior. The other weather variables show much more random behavior with significant spiking at certain points. The national load demand is most closely related to the temperature variable by inspection. WGP is a flexible model that is suitable to this type of data. I performed multiple statistical tests prior to performing ARIMA to determine the p, d, and q variables for each variable model. The Augmented Dickey-Fuller test (ADF) and the Kwiatkowski-Phillips-Schmidt-Shin test (KPSS) are both unit root tests to determine stationarity [85, 86]. The test statistics and results of these tests can be found in the Appendix B. I compared WGP with a standard GP and a VAR model using hourly frequencies for short-term (24-hours ahead) evaluation. The results of this analysis can be found in Chapter 4.

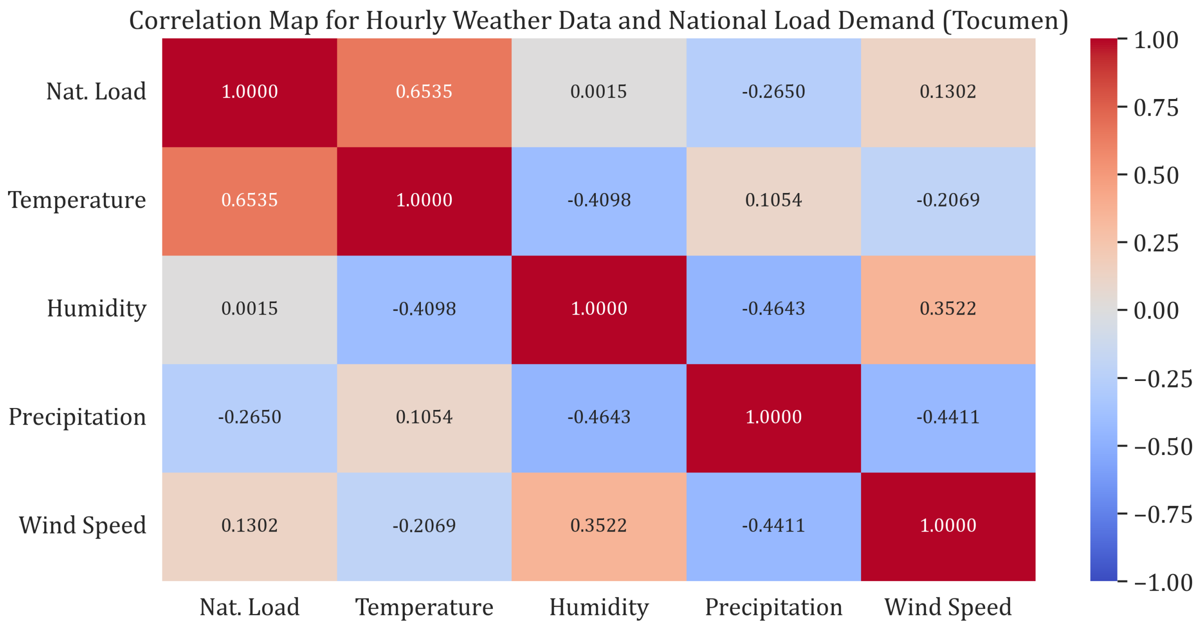

Figure 3.12: Correlation heatmap for hourly weather variables in the city of Tocumen.

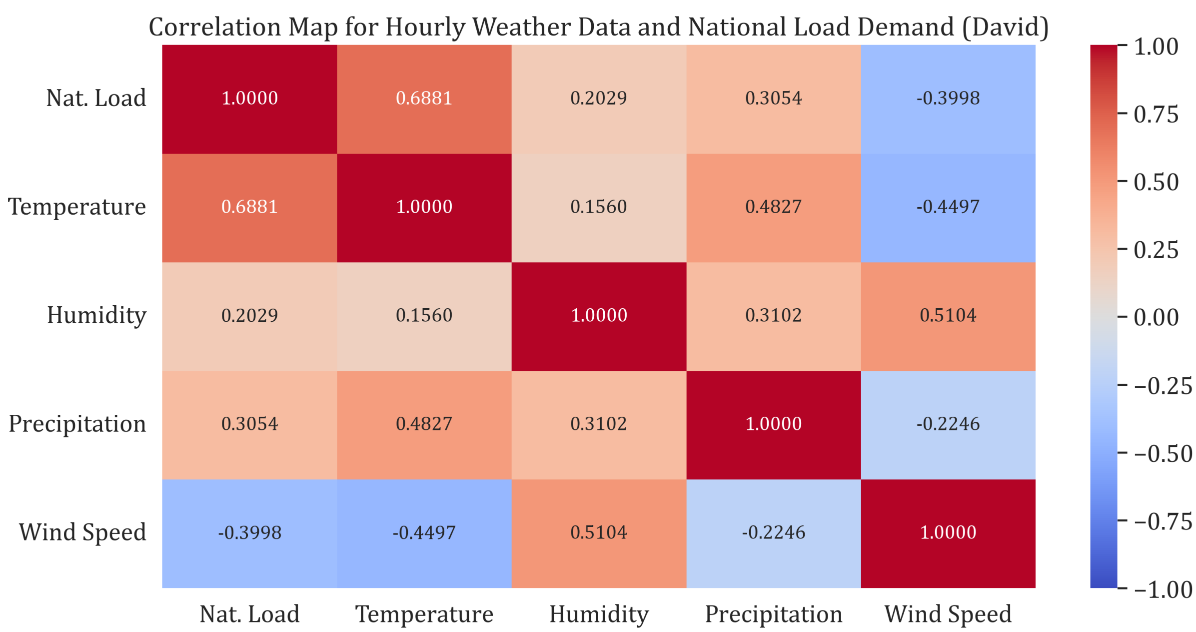

Figure 3.13: Correlation heatmap for hourly weather variables in the city of David.

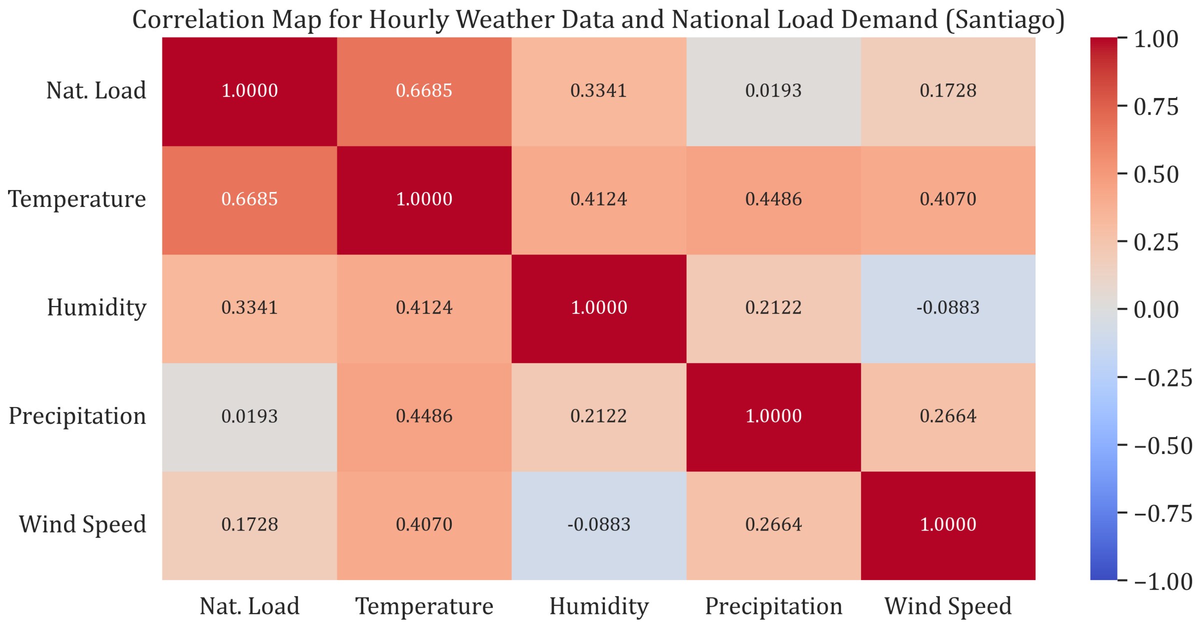

Figure 3.14: Correlation heatmap for hourly weather variables in the city of Santiago.

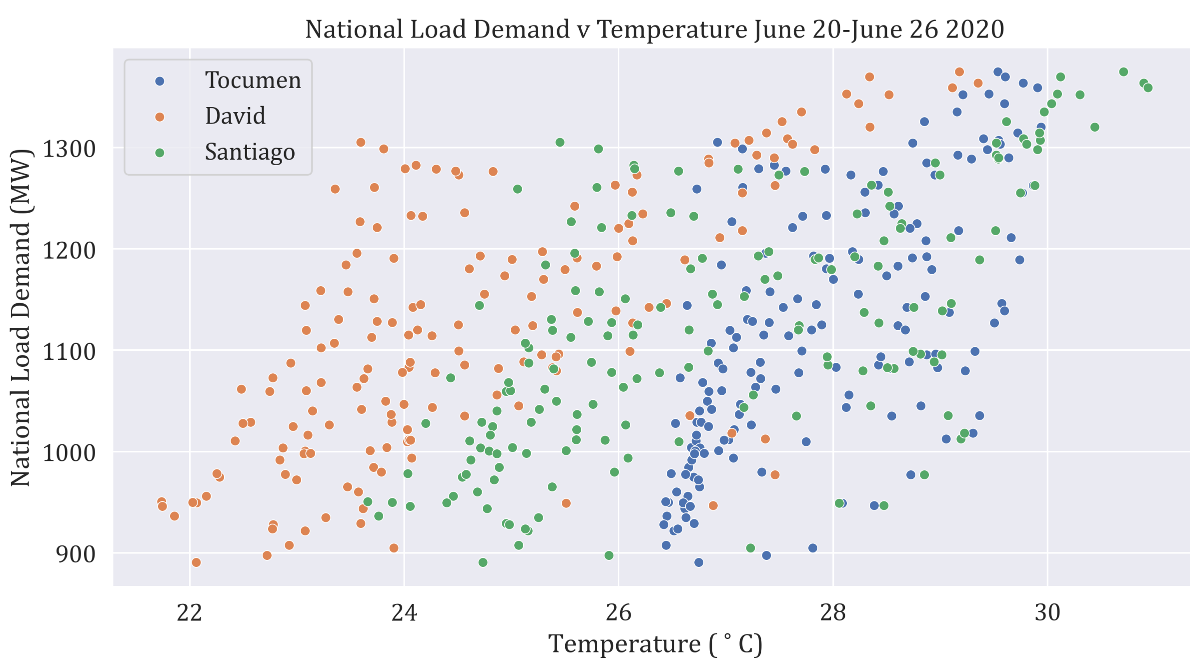

Figure 3.15: National load demand v temperature for each city.

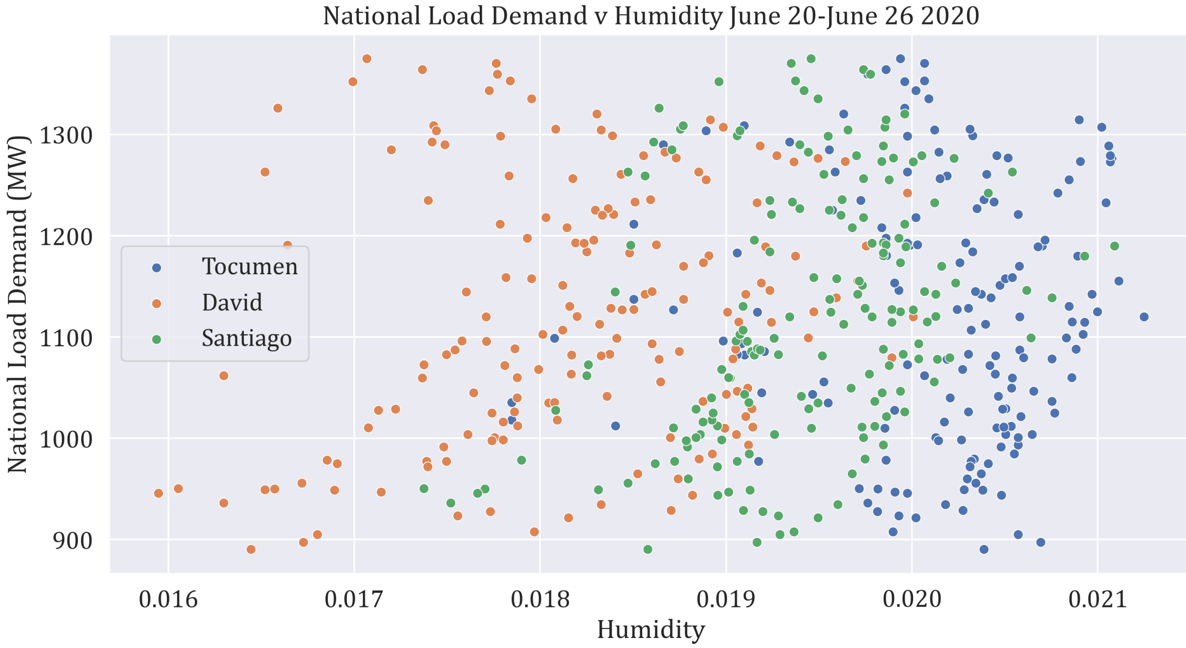

Figure 3.16: National load demand v humidity for each city.

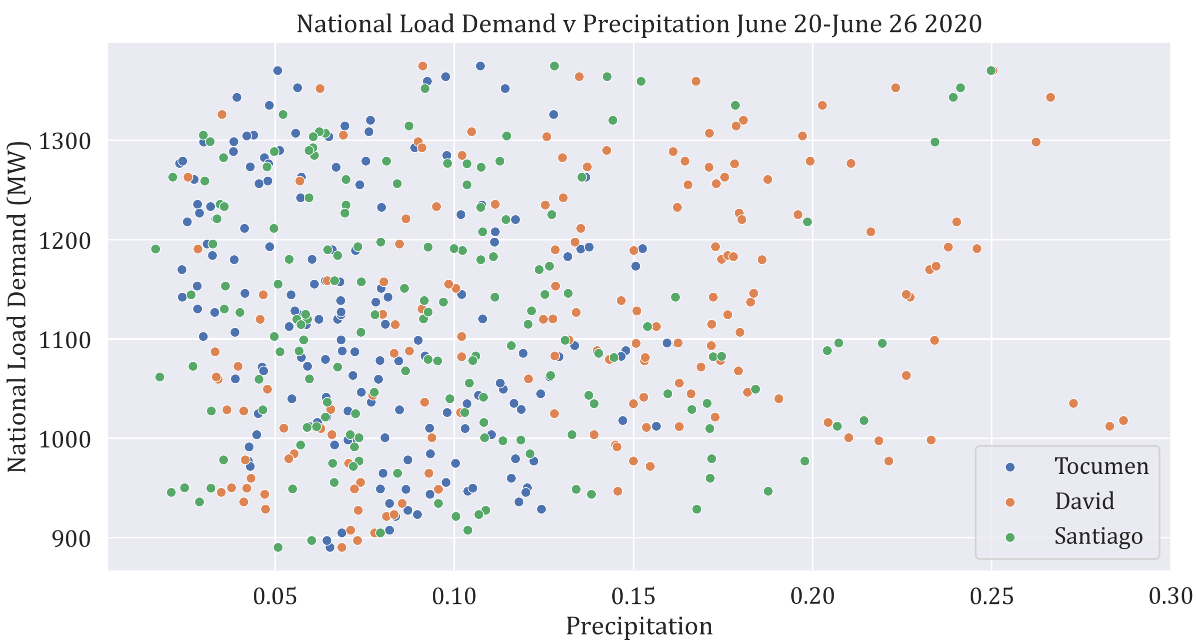

Figure 3.17: National load demand v precipitation for each city.

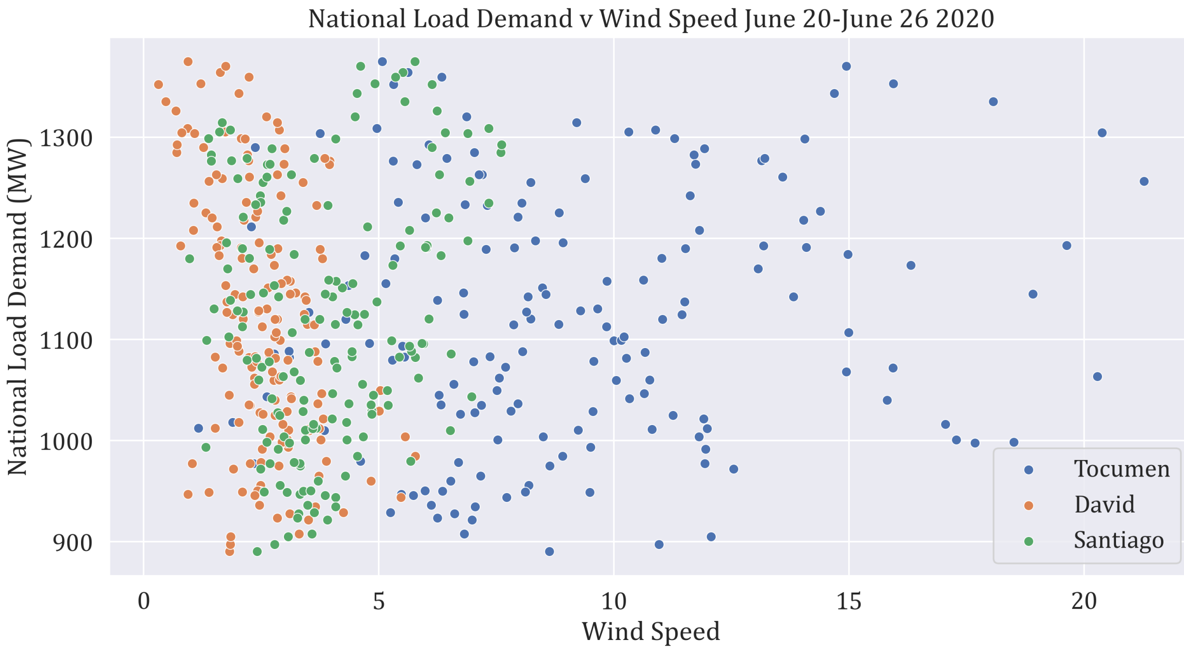

Figure 3.18: National load demand v wind speed for each city.

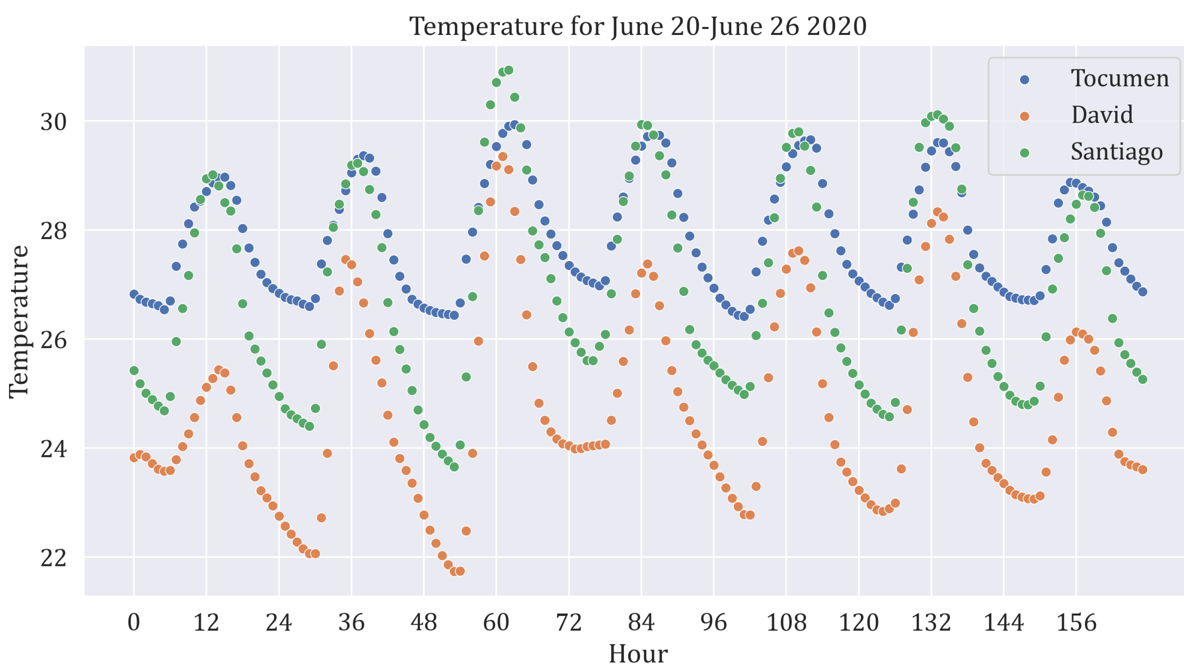

Figure 3.19: Temperature time series for each city.

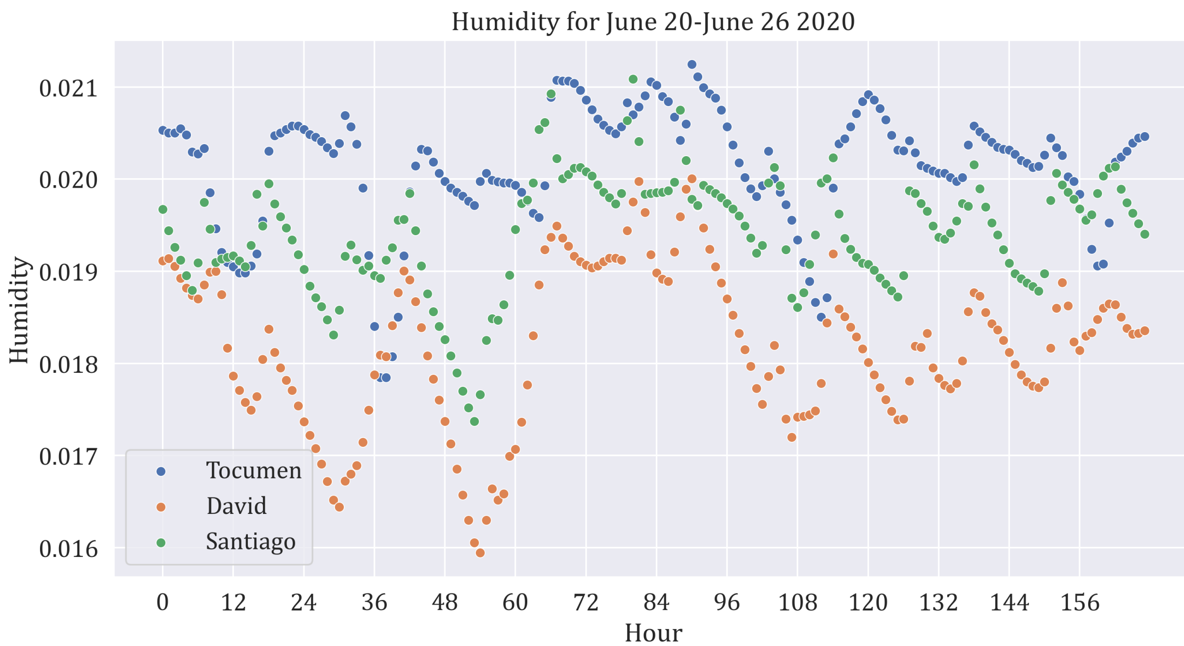

Figure 3.20: Humidity time series for each city.

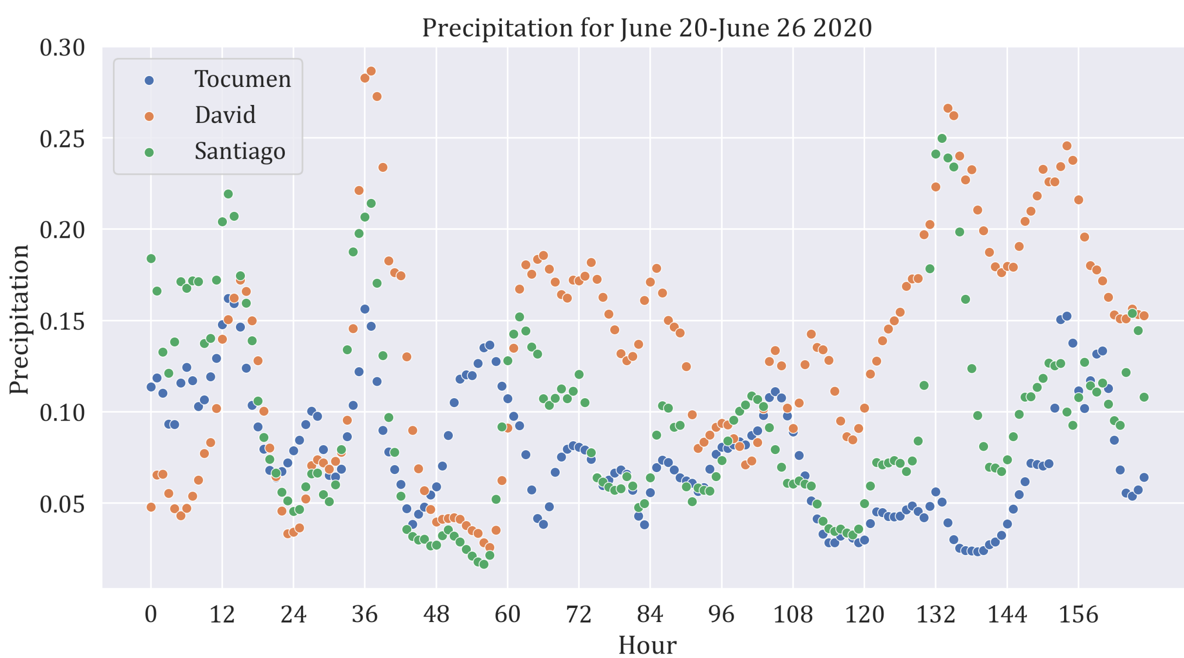

Figure 3.21: Precipitation time series for each city.

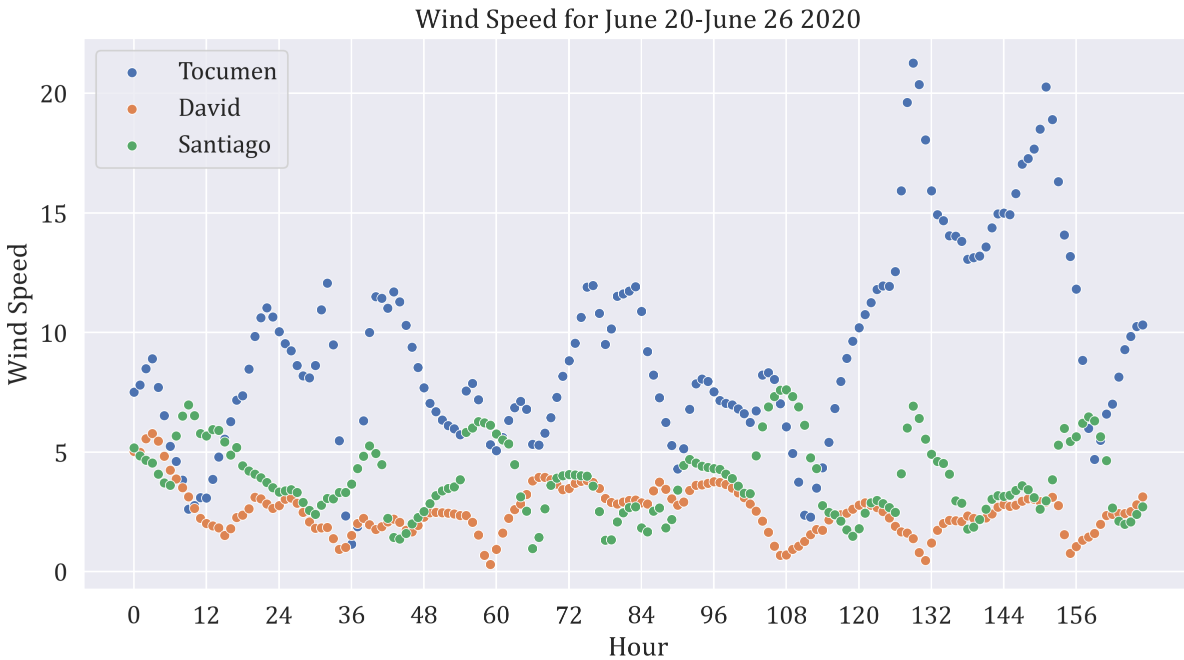

Figure 3.22: Wind Speed time series for each city.

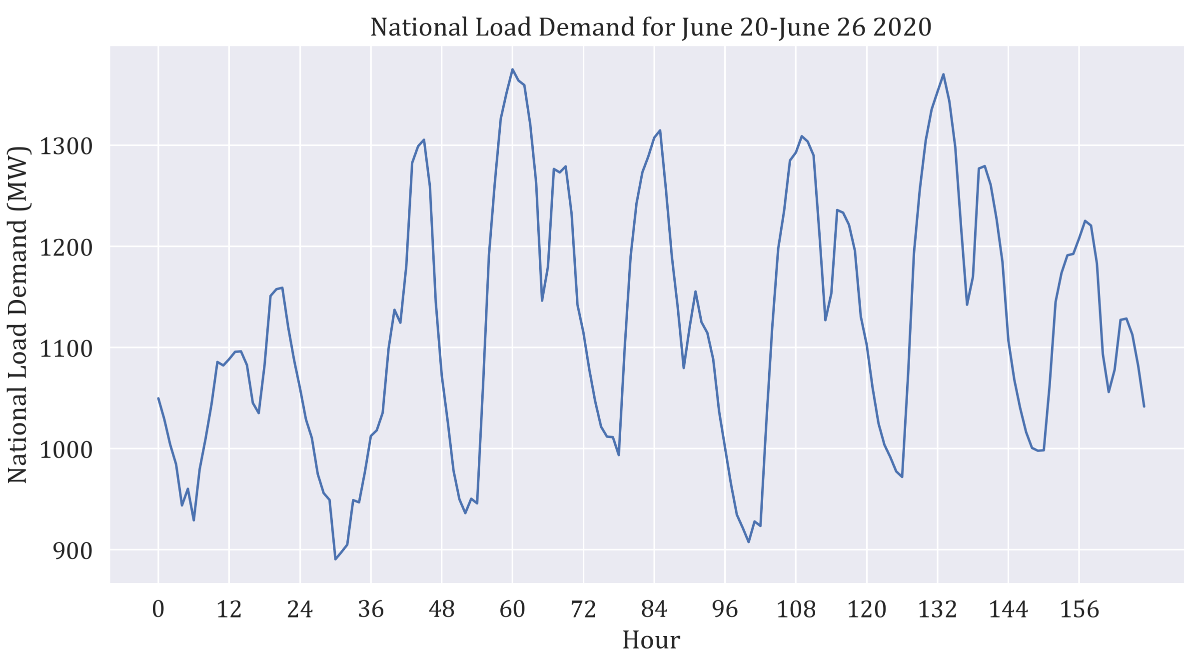

Figure 3.23: National load demand for Panama.

Chapter 4

Results

4.1 Power Grid Modeling

4.1.1 The Optimal Power Flow Equations

This section discusses the results of using WGP as a surrogate model for approximating the OPF equations based on a Pandapower IEEE 14-Bus system. The surrogate model used the network parameters (bus voltages, line loadings, etc.) to approximate the voltage magnitude, voltage angle, and generator active load outputs. Both surrogate models had no line loading violations. Figure 4.1 shows a chart comparison for the MAE at each output for GP and WGP. Bus voltages (VM pu) are represented by indexes 0-13, bus angles (VA degrees) by 14-27, and generator load (P MW) by 28-31. Clearly, GP outperforms WGP in this scenario. Both models show MAE of zero or close to zero for the bus voltages. This is likely because the voltages at the buses are operating under steady state conditions which is easier to estimate compared to bus angles and generator active power loads. MAE starts to spike in estimating bus angles and generator loads. As discussed in Chapter 3, OPF models operate under constraints and produce smooth outputs by design as part of the OPF solutions while surrogate models cannot enforce constraints. WGP is more suited to non-Gaussian and non-linear data, thus a standard GP has better performance under stead state conditions. Essentially, unless all the variables are similarly non-smooth or noisy, the poor fitting variables will drag down accuracy. Notably, the WGP results worsened with

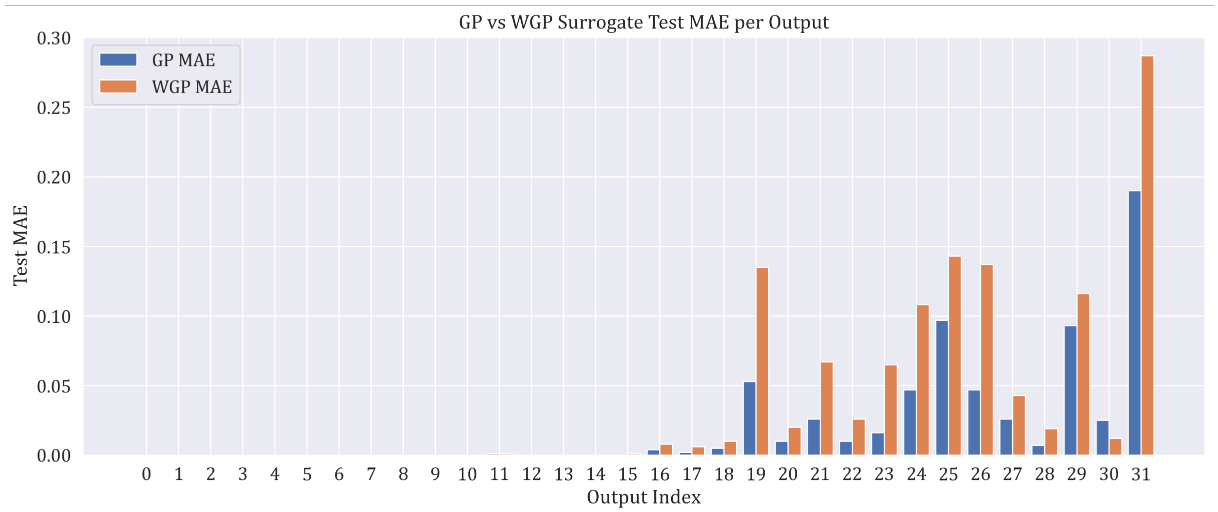

Figure 4.1: GP v WGP MAE for OPF surrogate models based on 100 random runs of a Pandapower IEEE 14-Bus network. Bus voltages (VM pu) are represented by indexes 0-13, bus angles (VA degrees) by 14-27, and generator load (P MW) by 28-31. GP performance is more accurate than WGP in terms of MAE.

additional iterations. The results here show that like most Gaussian models, the underlying assumptions must be met as closely as possible to ensure best performance. WGP accuracy may be improved by choosing only certain elements of network data to include, but in terms of OPF surrogate models in this case, GP is superior.

4.1.2 Parameter Estimation

This section discusses the results of estimating damping and inertia coefficients from Pandapower IEEE 14-Bus power grid data. I used GP and WGP in a data-driven analysis of the output of the Pandapower simulation and swing equations. Table 4.1 shows the MAE for the GP and WGP models. Figures 4.2 and 4.3 show the predicted value vs. true value plots of inertia and damping for each method and

WGP outperforms GP in both cases, but inertia has a more significant improvement. In both cases, the damping predictions are much flatter than that of inertia. This suggests that damping is more difficult to estimate using these models or that the features extracted from the Pandapower output are simply more suited to inertia. In addition, the physical properties of inertia and damping play a role. Damping is more sensitive to noise [87], which

Table 4.1: Results of estimating damping and inertia coefficients from a Pandapower IEEE 14-Bus grid data using GP and WGP. WGP outperforms GP in both cases.

	MAE (GP)	MAE (WGP)
Inertia (H)	0.1359	0.0920
Damping (Kd)	0.0406	0.0385

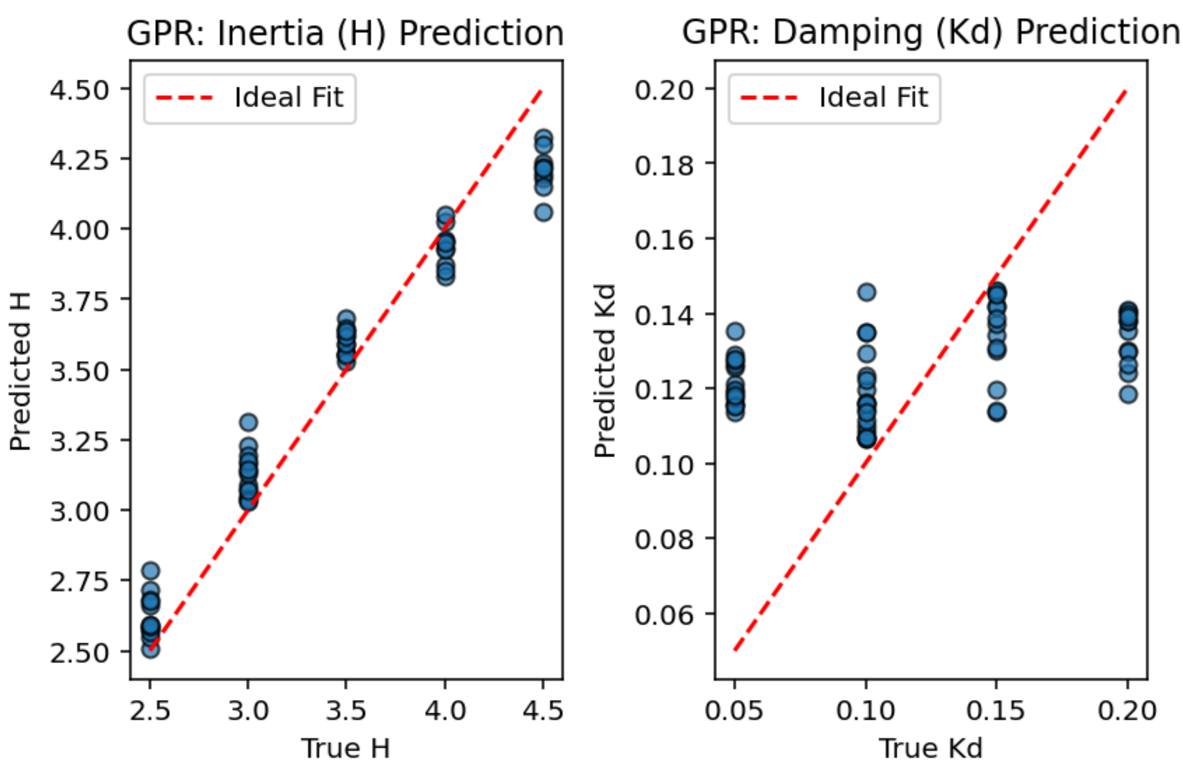

Figure 4.2: Prediction plots for inertia and damping coefficients from a Pandapower IEEE 14-Bus grid using GPR. GPR performance is stronger in inertia prediction.

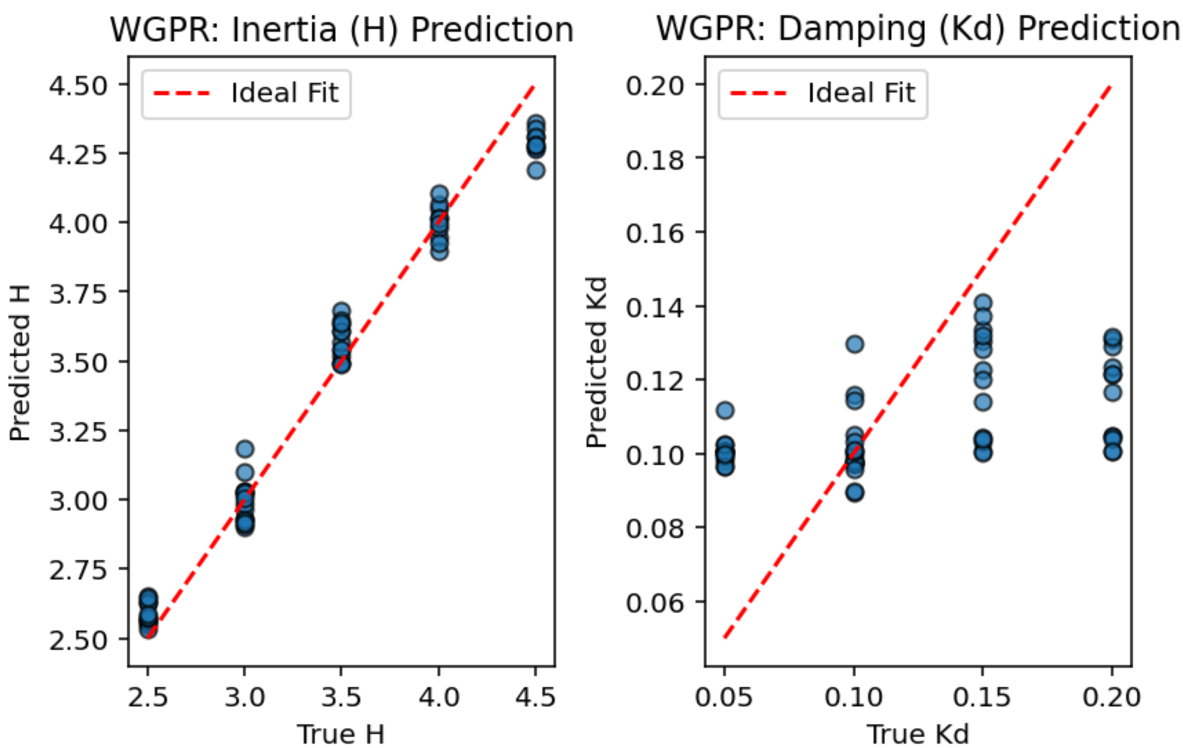

Figure 4.3: Prediction plots for inertia and damping coefficients from a Pandapower IEEE 14-Bus grid using WGPR. WGPR performance is stronger in inertia prediction and improves upon GP prediction for both parameters.

would explain why WGP improves performance but it is still more difficult to approximate than inertia.

Compared to OPF, WGP is more suited to these grid data parameters because of their known non-Gaussian behavior (especially under a disturbance) and relationship to power system states like phase angle or frequency. WGP is ideal for estimating parameters with known non-Gaussian behavior.

4.1.3 Load Forecasting

The results of the time-series analysis for load forecasting using ARIMA, GP, and WGP are discussed in this section. Figures 4.4 - Figure 4.6 show the load forecasts for the ARIMA, GP, and WGP models along with their confidence intervals. This forecast analysis is for short-term prediction of simulated power load data from a Pandapower IEEE 14-Bus system. It consists of six days of training data to predict 24 hours ahead. The GP and WGP figures shown in this chapter are those with the Matérn 2.5 kernel, which had lowest WGP MAE. Figures showing the load forecasts for each kernel can be found in Appendix D.

WGP outperformed both GP and ARIMA in this analysis. Table 4.2 shows the MAE based on different kernels used in analysis (this only applies to the GP and WGP models). I used four different kernels for comparison: the Square Exponential (RBF), the Matérn kernels (M 3/2 and M 5/2), and the Rational Quadratic (RQ). The model with the best overall performance is the WGP model with Matérn 5/2. Table 4.2 illustrates the importance of kernel choice in both GP and WGP analysis - the optimal kernel differed between the two approaches, highlighting that each model captures underlying system behavior in distinct ways. Notably, the worst kernel performance in the WGP model is still more accurate than the best kernel performance in the GP model. The right kernel choice is crucial to reap the full benefits of WGP, but the improvements over a traditional GP in load forecasting are clear.

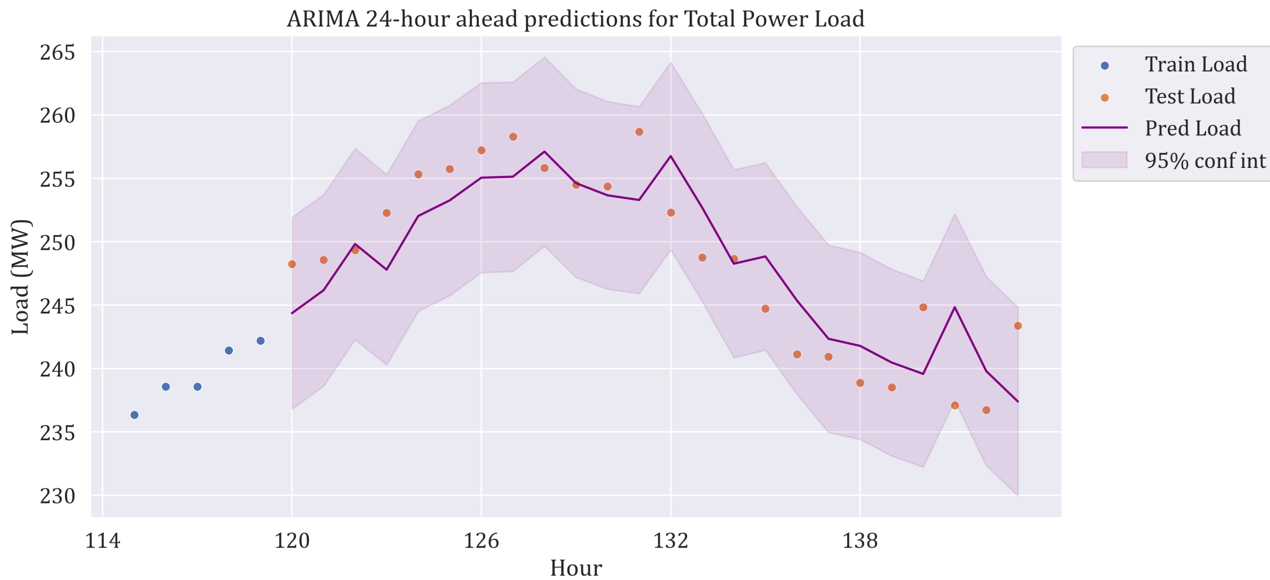

Figure 4.4: ARIMA power load forecast for 24-hour ahead prediction of power load data generated from a Pandapower IEEE 14-Bus system (MAE = 3.1353). ARIMA had the worst performance in terms of MAE.

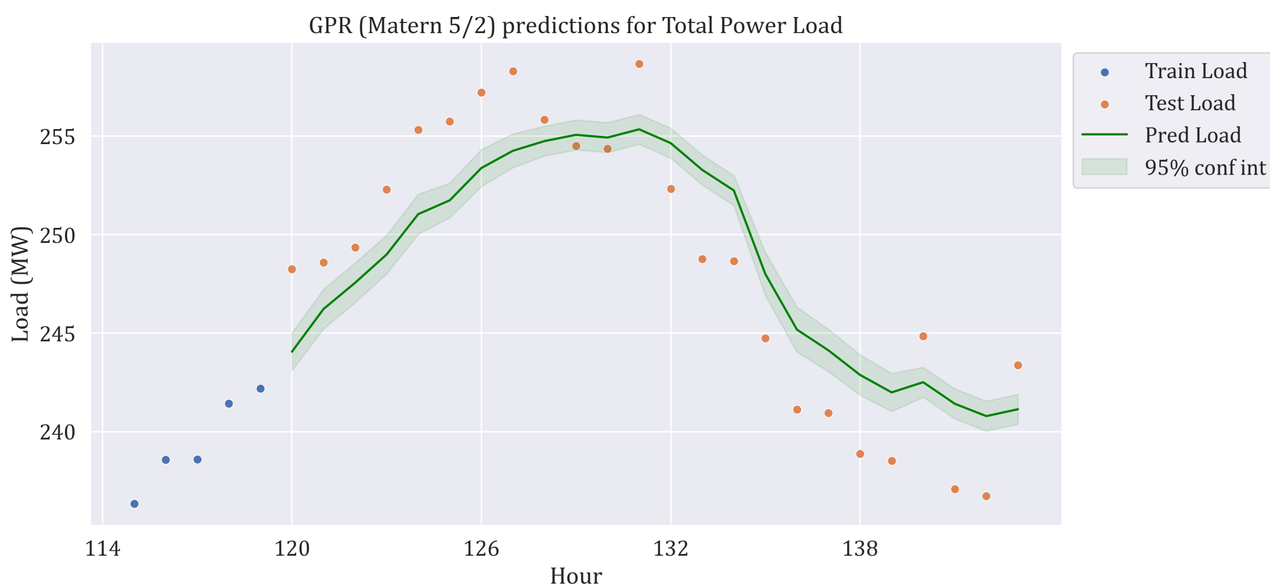

Figure 4.5: GPR (M 5/2) power load forecast for 24-hour ahead prediction of power load data generated from a Pandapower IEEE 14-Bus system. GP outperforms ARIMA but not WGP in terms of MAE.

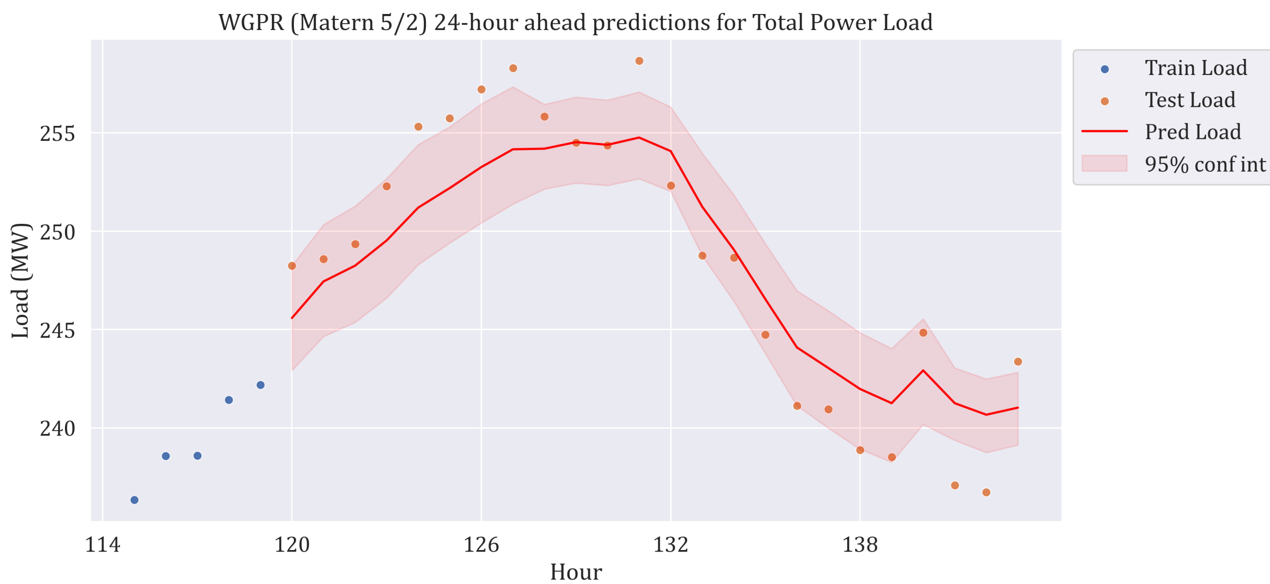

Figure 4.6: WGP (M 5/2) power load forecast for 24-hour ahead prediction of power load data generated from a Pandapower IEEE 14-Bus system. WGP outperforms both ARIMA and GP in terms of MAE.

Table 4.2: Average MAE by kernel for the GP and WGP 24-hour ahead prediction of power load data generated from a Pandapower IEEE 14-Bus system. WGP has better performance overall and better individual kernel performance.

Kernel	**GP**		**WGP**	
	Train	Test	Train	Test
Squared Exponential (RBF)	2.5284	2.9930	2.3465	2.5893
Matérn (M 3/2)	2.4931	3.2100	2.1257	2.5601
Matérn (M 5/2)	2.5043	3.1118	2.1182	2.4475
Rational Quadratic (RQ)	2.5284	2.9934	2.1869	2.6668
Average	2.5136	3.0771	2.1943	2.5659

Figure 4.5 and Figure 4.6 show the flexibility of WGP in action. Not only does the WGP confidence interval capture more of the noise in test data, it also "stretches" further than the GP in response to the noise in the data. Figure 4.7 shows the entire distribution of these models over the training and test data where the "warping" behavior of WGP is obvious. The GP and WGP are not wildly different, but in specific areas (especially around hour 24), it is clear the WGP is more robust to the non-linear and noisy nature of the data. Furthermore, WGP can be improved by adding warping iterations if necessary to reach an accuracy threshold.

However, Figure 4.1 shows the ARIMA model shows improved performance at the end of the forecast. In terms of practical implementation, short term prediction may favor WGP, whereas ARIMA may be more suited to long term prediction. This analysis focused on simple, well-known kernels for implementation illustration. Further study with more sophisticated kernel combinations could improve the long term WGP forecast. Overall, WGP outperforms existing statistical methods, but further improvement is available.

4.2 Signal Analysis

The results of the signal analysis are discussed in this section. I used three features for classification: frequency centroid, rms, and standard deviation. Tables 4.3 - 4.8 show the results of the GP and WGP filtering on the classification of selected signal anomalies: arcing, lighting strikes, and fuse blows. WGP filtering performs best overall in accuracy and F-1 score. Each model classifies the baseline signal most accurately, so detecting an anomaly is more successful than classifying the anomaly. Classifying lightning strikes has poor performance for all models, but best performance from WGP. That said, WGP is least successful at classifying arcing. Arcing and lightning strikes have similar effects on signal data, which affects the model's ability to distinguish between them. Furthermore, lightning and arcing produce fast, high-frequency transients, adding to the detection difficulty. These anomalies can also occur intermittently or propagate unpredictably. WGP is particularly successful at classifying fuse blows. This is likely due to the nature of the anomaly - fuses are designed to blow as a protective measure, and they have a distinct effect on the signal.

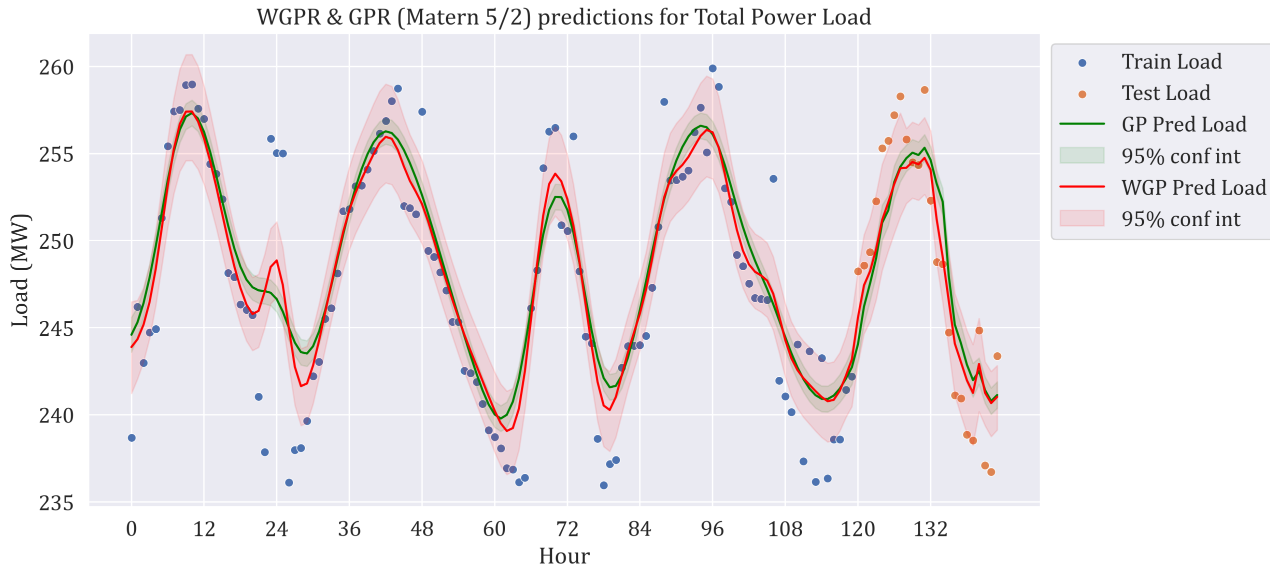

Figure 4.7: GP and WGP (M 5/2) comparison for 24-hour ahead prediction of power load data generated from a Pandapower IEEE 14-Bus system. WGP is more flexible than GP and its 95 percent confidence intervals capture more of the noise in the predictions.

Table 4.3: Classification Report — Unfiltered: Results of the anomaly classification using combined real-world and synthetic signal data from the GESL. Classifying baseline (no anomaly) behavior is most accurate.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Arcing	0.35	0.44	0.39	16
Baseline	0.95	0.90	0.92	20
Fuse Blow	0.33	0.35	0.34	17
Lightning Strike	0.40	0.29	0.33	14
Accuracy		0.52 (67 samples)		
Macro Avg	0.51	0.49	0.50	67
Weighted Avg	0.53	0.52	0.53	67

Table 4.4: Confusion Matrix — Unfiltered

	Arcing	**Baseline**	**Fuse Blow**	**Lightning Strike**
Arcing	7	1	6	2
Baseline	2	18	0	0
Fuse Blow	7	0	6	4
Lightning Strike	4	0	6	4

Table 4.5: Classification Report — GP Filtered: Results of the anomaly classification using combined real-world and synthetic signal data from the GESL. GP filtering performance is the worst, with particularly bad performance in classifying lightning strikes.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Arcing	0.38	0.38	0.38	16
Baseline	0.91	1.00	0.95	20
Fuse Blow	0.33	0.35	0.34	17
Lightning Strike	0.09	0.07	0.08	14
Accuracy		0.49 (67 samples)		
Macro Avg	0.43	0.45	0.44	67
Weighted Avg	0.46	0.49	0.48	67

Table 4.6: Confusion Matrix — GP Filtered

	Arcing	**Baseline**	**Fuse Blow**	**Lightning Strike**
Arcing	6	1	4	5
Baseline	0	20	0	0
Fuse Blow	5	1	6	5
Lightning Strike	5	0	8	1

Table 4.7: Classification Report — WGP Filtered: Results of the anomaly classification using combined real-world and synthetic signal data from the GESL. WGP filtering performance is the best.

Class	**Precision**	**Recall**	**F1-score**	**Support**
Arcing	0.36	0.25	0.30	16
Baseline	0.82	0.90	0.86	20
Fuse Blow	0.45	0.53	0.49	17
Lightning Strike	0.43	0.43	0.43	14
Accuracy		0.55 (67 samples)		
Macro Avg	0.52	0.53	0.52	67
Weighted Avg	0.53	0.55	0.54	67

Table 4.8: Confusion Matrix — WGP Filtered

	Arcing	**Baseline**	**Fuse Blow**	**Lightning Strike**
Arcing	4	2	6	4
Baseline	2	18	0	0
Fuse Blow	3	1	9	4
Lightning Strike	2	1	5	6

Note that GP filtering actually hurt performance. This suggests that the noise filtering may be removing important characteristics of the signal, highlighting WGP's superior handling of non-stationary and non-Gaussian signal components. However, WGP's effect on RF classification accuracy is not strong on its own. Other preprocessing classification methods could bolster WGP's filtering. There are also other features that may need to be considered that better reflect the behavior of the signal anomalies. In general, WGP filtering is more effective at accurately modeling signal noise than GP filtering, but more research is needed to improve performance in anomaly classification. This method is well-suited to modeling noisy signals and shows promise at identifying anomalies in signal data associated with power grid operations.

4.3 Load + Weather Forecasting

This section focuses on the results of the load forecasting in combination with weather data. Weather has a significant impact on grid operations, but both are complex systems that are hard to model. I performed this analysis on a real-world dataset measuring temperature, relative humidity, and wind speed for three separate cities in Panama, as well as the national demand [84]. Figures 4.8 - 4.10. show the results of multivariate forecasting with a VAR model, a GP model, and a WGP model using temperature, wind speed, and humidity.

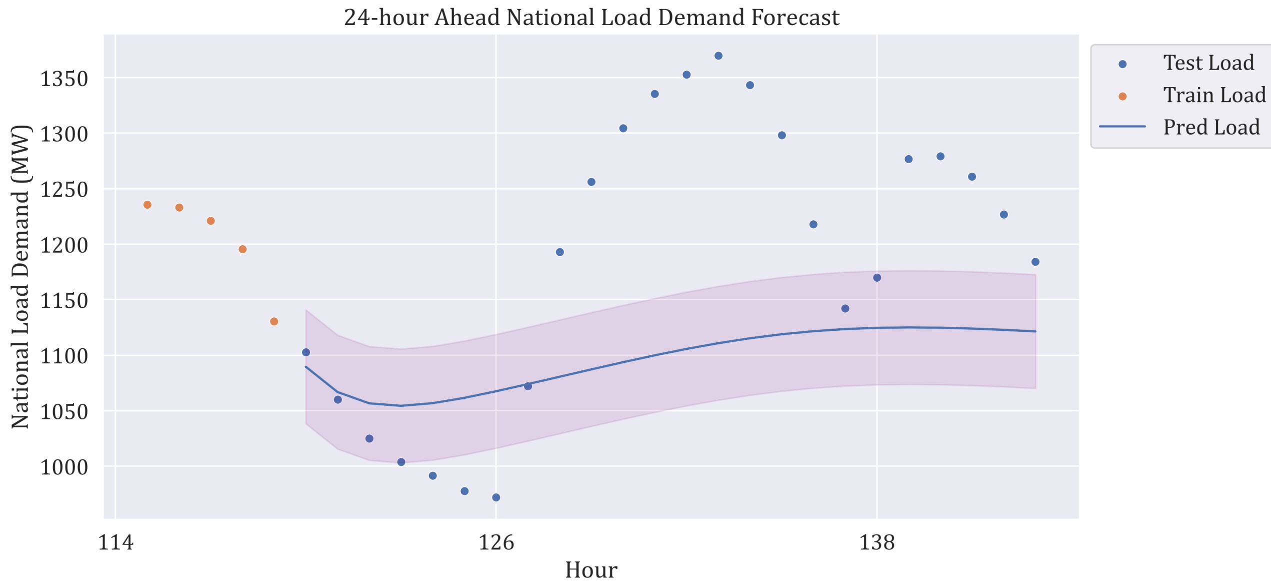

Figure 4.8: Wind speed VAR model for 24-hour ahead prediction of national load demand. VAR had the worst performance.

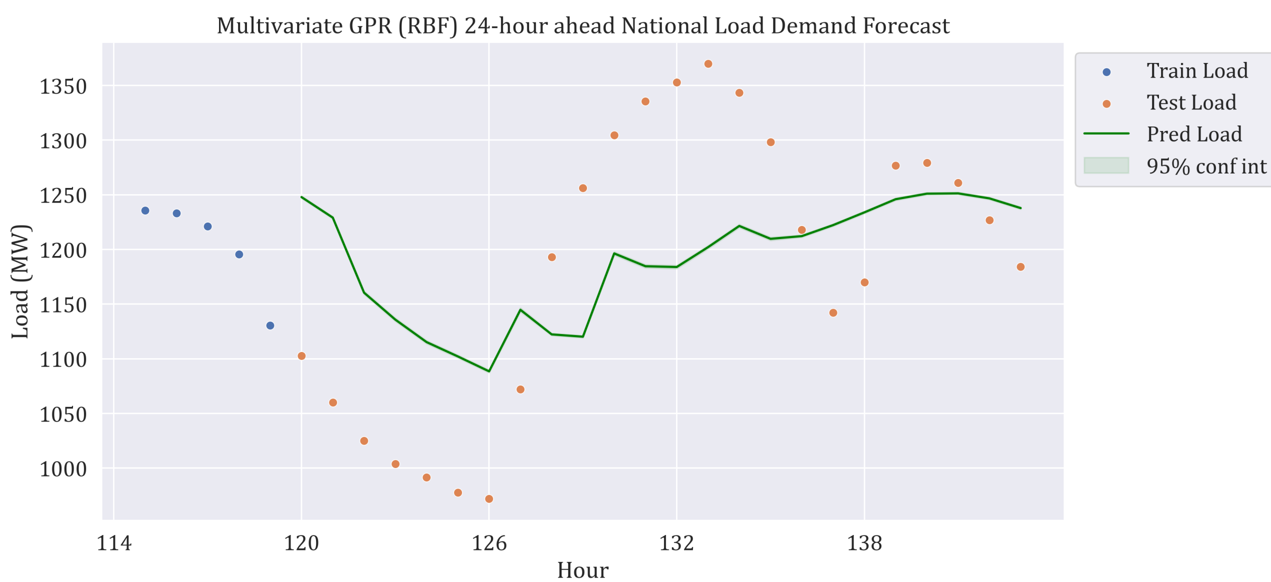

Figure 4.9: Wind speed GPR (RBF) model for 24-hour ahead prediction for national load demand. GPR outperforms VAR but not WGP

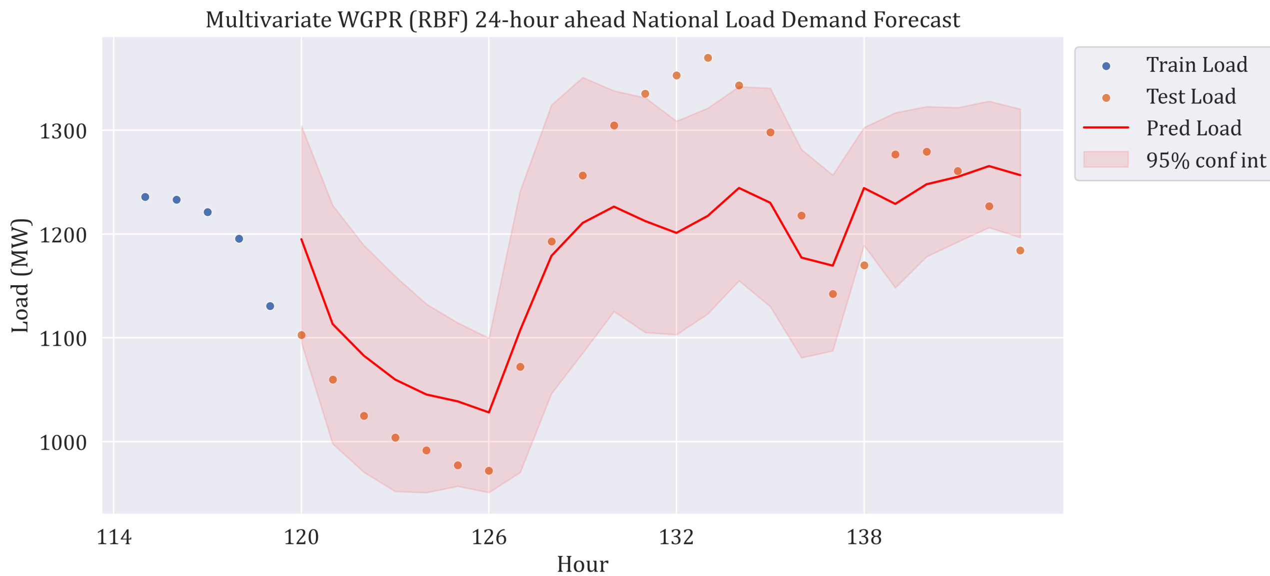

Figure 4.10: Wind speed WGPR (RBF) model for for 24-hour ahead prediction of national load demand. WGP had the best performance.

Table 4.9 shows the MAE for all of the models and weather variables. Overall, the WGP models outperformed the others. The largest error improvement from GP to WGP occurred for the wind model. Wind is particularly random in nature, illustrating WGP's capability at handling complex data with an unclear underlying distribution. Similarly, the humidity data showed randomness and had the worst model performance for VAR, GPR, and WGPR. In comparison, the best performance for all models occurred for the temperature model. Given what was observed in the preliminary data analysis in Figure 3.18, this makes sense. The temperature was most similar in underlying distribution and much less sporadic than the other weather variables. While the GP outperformed the VAR model in test data, it is clear from the figures that it is underfit. Figure 4.11 shows how little of the underlying distribution it captured. Furthermore, the confidence intervals for the GP prediction are extremely small, meaning there is a small window for flexibility in prediction. It is also possible that the kernel choice (RBF) does not reflect the underlying data well. In this case, WGP once again is able to adapt to data better even if the ideal kernel was not chosen. The flexibility of WGP is able to more accurately reflect the past data and predict the future data. These results suggest that temperature is widely the best choice among wind speed and relative humidity to predict future load demand in Panama. However, the WGP model

Table 4.9: MAE by weather variable for the VAR, GP, and WGP models for 24-hour ahead prediction of national load demand. Best performance was seen in the WGP temperature model. WGP outperforms the other methods overall.

	VAR	**GP**	**WGP**
Temp	100.6717	88.2059	60.2273
Wind	115.0599	96.8465	64.0123
Humidity	117.3112	103.4478	82.8670

starts to bridge the gap for wind speed. (Figures for all other weather variables can be found in Appendix E). The data used for this method is somewhat limited by the use of weather variables for three different cities to account for national load demand. Weather variables are significantly affected by location, meaning variables from different regions of the country may be affecting the outcome. Overall, WGP improves upon other statistical methods to incorporate weather data into load demand forecasting. Further improvements via kernels or warping iterations should be explored.

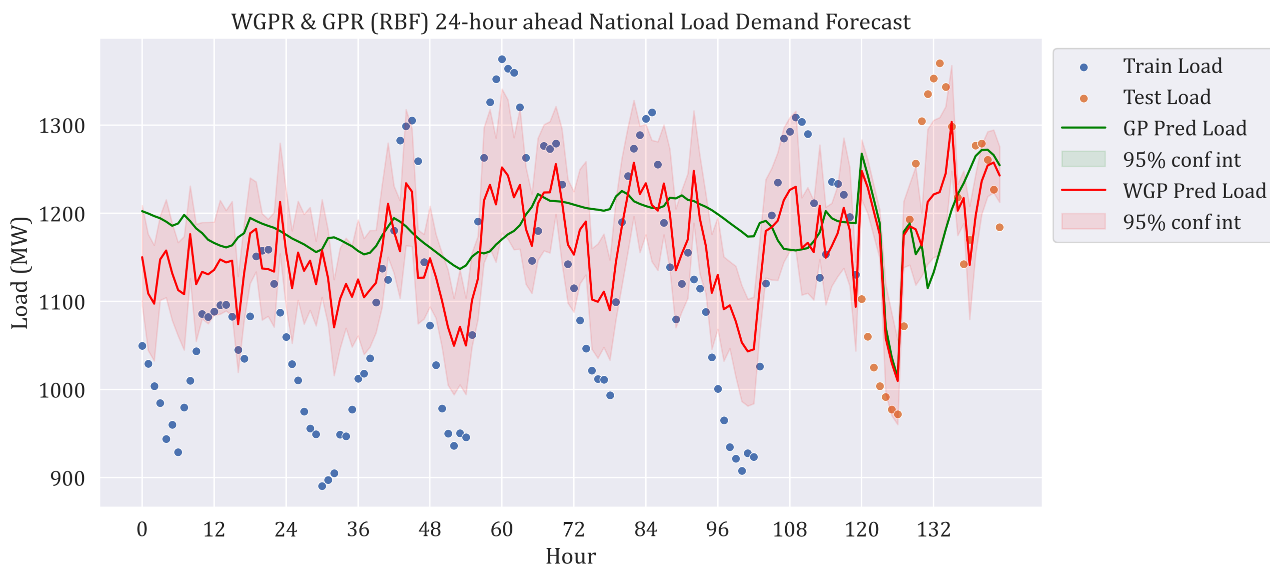

Figure 4.11: Humidity GP and WGP (RBF) 24 hour forecast for load demand. WGP shows more flexibility in representing the underlying distribution of the data.

Chapter 5

Future Work

This research has shown the viability of WGPs for power grid data analysis. However, I simulated the power grid data for this analysis using the Pandapower IEEE 14-Bus system. Future work should expand to gathering real-world data from power utilities, including solar and hydro power plants in addition to traditional transmission and distribution. There are also other complex industrial processes (i.e., manufacturing) that may benefit from WGP in terms of its flexibility with non-Gaussian, noisy data.

As evidenced from the results here and in the literature, kernel selection is an important aspect of any Gaussian method. There are multiple areas for future research on WGP kernels. I used simple kernels in order to compare each one under different conditions, but combining kernels may improve performance and should be investigated in future applications. The analysis revealed that WGP outperformed other methods even with its worst kernel. This suggests that WGP may be able to outperform GPs with more complex kernel methods with the use of simple kernels. This lends itself to tractability and may reduce the computational complexity associated with these methods. Furthermore, this evaluation focused on comparing GPs and WGPs since they are interpretable, generalizable models well-suited for regression problems with noisy, non-Gaussian data. Future efforts should compare WGP to other statistical distributions (Gamma, Weibull, etc.) as well as other ML methods, like Long, Short Term Memory neural nets and Gaussian Mixture Models.

Continued research into the effects of weather on grid functioning is necessary as climate conditions fluctuate. I applied WGP to real-world data, but this sort of analysis is very specific to location. Utilities in Florida are concerned with high temperatures and hurricanes, while utilities in Alaska deal with extreme cold and snow and ice. Furthermore, future research should incorporate prediction during extreme weather events in addition to daily weather. Regular demand from customers certainly strains the grid, however, more analysis on responding to grid events during extreme weather is needed. This research supports data integration from multiple streams to manage grid malfunctions - incorporating the AIMS project in future work could provide a way for utilities to immediately respond to issues.

The warping function is not explored much in the literature, nor was it the focus of this analysis. It is essentially a means to an end to improve accuracy. The warping function acts like an activation function in neural networks, with the exception that it is fully recoverable. I used the hyperbolic tangent function, but this is not the only function available. Further research into the use of other functions like sigmoid or the ReLU variations are needed for comparison. Additionally, since the function itself is learned from the data, it may have more to offer in terms of understanding the underlying relationship between the variables. In addition, the number of warping iterations, I, should be furthered evaluated to determine the accuracy-complexity tradeoff and incorporate I as a parameter itself.

WGP is a computationally complex method, even more so than traditional GP. Recall that the warping function calls for iterating over the data I times. For my purposes, I did not need to use anything more than $I = 3$ to outperform GP. However, there is no limit on how many iterations can be applied, and more complex data may require more iterations. Scaling WGP to big data applications will need to incorporate methods that reduce that complexity. This is a particularly important aspect in grid management because of the need to operate in real time and respond quickly to problems. Minimal research into sparse WGPs can be found in the literature, but methods using rank approximations, nearest neighbor, grouping and permutations, and offline/online processes deserve more research. These methods have all had success in GP analysis, making them applicable to WGP.

5.1 Warped Gaussian Processes and Artificial Intelligence

Any discussion on the future of a ML method like WGP would be incomplete without mentioning Artificial Intelligence (AI). Recent rapid improvements in AI have attracted a lot of attention in both academic and non-academic spaces. The exact definition of AI is not well defined. Figure 5.1, below, is a common variation of the Venn diagram depicting how these categories relate to one another. ML methods are often referred to as subsets of AI, including Gaussian Processes. Statistics and AI are interrelated, and statistics can make up the building blocks of AI [88]. WGPs were developed from from a well established ML method (GPs) that leverages the power of statistics in its analysis. That said, the rationale of how AI models arrive at their results is not easily interpreted. One well known problem with big data and AI is that the quality of the data has strong effects on the reliability, reproducibility, and bias of the results [89]. This is also true of traditional statistics, however, statistical models are less opaque. WGP could benefit AI because it calculates a prior distribution, making it obvious if the model is not capturing the underlying data regardless of accuracy measures. It is also possible to quantify its uncertainty through confidence intervals. AI can also support WGPs [90]. Preprocessing data is necessary for creating quality and relevant datasets, but it is often time consuming. AI can speed up this process and create datasets prepared for WGP analysis. Using AI to reduce data dimensionality would accommodate WGP's computational complexity. Future research should focus on best practices for combining AI and WGP.

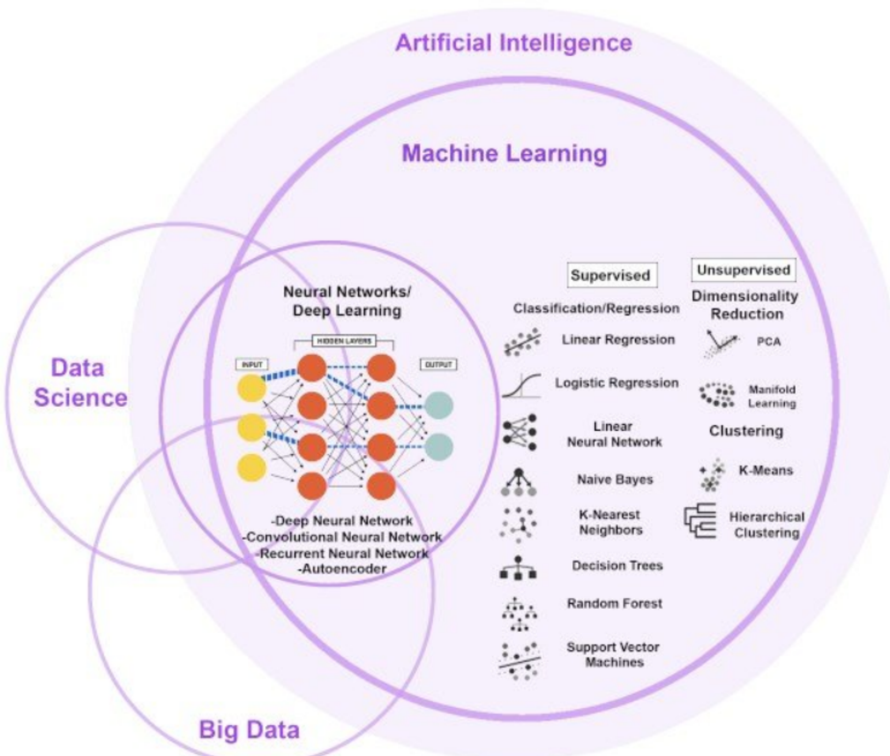

Figure 5.1: A Venn diagram depicting the relationship between artificial intelligence and machine learning (ML) [91]. GP and WGP are both considered supervised ML methods.

Chapter 6

Conclusion

Power grid SCADA systems collect a variety of grid related data and continue to integrate new sources of energy from wind and solar. Additionally, new capabilities in remote sensing, like the AIMS project, provide even more data from the use of advanced sensors. Much of this data may be non-Gaussian, non-stationary, non-linear, and/or noisy. Managing this data is difficult under ideal conditions and is only made more difficult by extra strain due to climate change, physical and cyber attacks, and aging infrastructure. Successful SCADA systems must leverage this data to to ensure reliable service and facilitate quick decision making. Interpretability is key for grid data analysis to ensure trust in model outputs and determine the rationale behind decision making processes. This research shows WGPs are a viable method for power grid data analysis. WGP is a generalization of the well-established GP method. It is a stochastic based method that improves upon standard GP performance when facing non-stationary, non-Gaussian, and noisy data - which describes much of real-world grid data. The flexibility of WGP makes it suitable for power grid data analysis because it can handle multiple data streams (signals, grid parameters, loading, etc.), it is interpretable, and can produce measures of its own uncertainty. This method is underutilized in the power grid space given its many advantages. WGP was applied to a combination of simulated and real-world data from multiple sources, including grid data from a Pandapower IEEE 14-Bus system simulation; real and synthetic signal data from GESL; and real-world weather and load demand data collected by Panama's national utility, CND. The results show WGP is an effective method for estimating grid parameters, reducing noise in signals, forecasting power load demand, and incorporating weather data into power load forecasts. Future areas of research should focus on techniques to manage computational complexity, incorporate real-world grid data, and integrate AI into data analysis pipelines.

Bibliography

[1] E. Foley, "Leveraging gaussian processes in remote sensing," *Energies*, vol. 17, no. 16, 2024. [Online]. Available: https://www.mdpi.com/1996-1073/17/16/3895 1, 6

[2] M. Weiss and M. Weiss, "An assessment of threats to the american power grid," *Energy, Sustainability and Society*, vol. 9, no. 18, 2019. 1

[3] International Energy Agency (2023), *Energy Efficiency 2023*, IEA, Paris, 2023, licence: CC BY 4.0. 1

[4] U.S. Department of Energy, Grid Deployment Office, "What Does It Take to Modernize the U.S. Electric Grid?" https://www.energy.gov/gdo/articles/what-does-it-take-modernize-us-electric-grid, oct 2023. 1

[5] D. E. N. Center. (2024) Duke energy continues rebuild of power infrastructure following hurricane helene. [Online]. Available: https://news.duke-energy.com/releases/duke-energy-continues-rebuild-of-power-infrastructure-following-hurricane-helene 2

[6] Department of Energy, "Electric emergency and disturbance (oe-417) events," 2023. 2

[7] E. Mikkelsen. (2024) What has happened in the 2 years since moore county substation attacks? [Online]. Available: https://myfox8.com/news/investigations/power-grid-attack/2-years-since-moore-county/ 2

[8] D. o. M. United States Attorney's Office. (2025) White supremacist leader found guilty of conspiring to destroy regional power grid. [Online]. Available: https://www.justice.gov/usao-md/pr/white-supremacist-leader-found-guilty-conspiring-destroy-regional-power-grid 2

[9] P. Fuhr, "Pilates (M617000251)," 2022. 2

[10] P. Fuhr and W. Monday, "Vesta (doe proposal number 2318-z668-24)," 2024. 2

[11] E. Foley, "Autonomous intelligence measurement sensors & systems (aims)," Poster for The Social Machines and AI Robotics Technology, May 1-2, 2024, Knoxville, TN. 3

[12] R. Touati, M. Mignotte, and M. Dahmane, "Multimodal change detection in remote sensing images using an unsupervised pixel pairwise-based markov random field model," *IEEE Transactions on Image Processing*, vol. 29, pp. 757–767, 2020. 3, 19, 20

[13] Y. Liu, S. Piramanayagam, S. T. Monteiro, and E. Saber, "Semantic segmentation of remote sensing data using gaussian processes and higher-order crfs," in *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2017, pp. 5454–5457. 3, 16

[14] D. H. Svendsen, L. Martino, and G. Camps-Valls, "Active emulation of computer codes with gaussian processes –application to remote sensing," *Pattern Recognition*, vol. 100, p. 107103, 2019. 3, 19, 20, 21

[15] M. Katzfuss, J. Guinness, W. Gong, and D. Zilber, "Vecchia Approximations of Gaussian-Process Predictions," *Journal of Agricultural, Biological and Environmental Statistics*, vol. 25, no. 3, pp. 383–414, 2020. 4, 17, 18

[16] D. H. Svendsen, P. Morales-Álvarez, R. Molina, and G. Camps-Valls, "Deep gaussian processes for geophysical parameter retrieval," in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018, pp. 6175–6178. 4, 21, 22

[17] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning.* Cambridge, MA: Massachusetts Institute of Technology, 2006. 4, 5, 29, 31, 95

[18] A. Jain, T. Nghiem, M. Morari, and R. Mangharam, "Learning and control using gaussian processes," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, 2018, pp. 140–149. 4, 22

[19] E. Snelson, Z. Ghahramani, and C. Rasmussen, "Warped gaussian processes," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds., vol. 16. MIT Press, 2003. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2003/file/6b5754d737784b51ec5075c0dc437bf0-Paper.pdf 5, 6, 31

[20] U. AlHaddad, A. Basuhail, M. Khemakhem, F. E. Eassa, and K. Jambi, "Towards sustainable energy grids: A machine learning-based ensemble methods approach for outages estimation in extreme weather events," *Sustainability*, vol. 15, no. 16, 2023. 5

[21] R. Rocchetta, L. Bellani, M. Compare, E. Zio, and E. Patelli, "A reinforcement learning framework for optimal operation and maintenance of power grids," *Applied Energy*, vol. 241, pp. 291–301, 2019. 5

[22] A. M. Tartakovsky, T. Ma, D. A. Barajas-Solano, and R. Tipireddy, "Physics-informed gaussian process regression for states estimation and forecasting in power grids," *International Journal of Forecasting*, vol. 39, no. 2, pp. 967–980, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169207022000474 7, 8

[23] A. Ghasempour and M. Martínez-Ramón, "Short-term electric load prediction in smart grid using multi-output gaussian processes regression," in *2023 IEEE Kansas Power and Energy Conference (KPEC)*, 2023, pp. 1–6. 8

[24] M. Mitrovic, O. Kundacina, A. Lukashevich, S. Budennyy, P. Vorobev, V. Terzija, Y. Maximov, and D. Deka, "Gp cc-opf: Gaussian process based optimization tool for chance-constrained optimal power flow," *Software Impacts*, vol. 16, p. 100489, 2023. 8, 9

[25] M. Mitrovic, A. Lukashevich, P. Vorobev, V. Terzija, S. Budennyy, Y. Maximov, and D. Deka, "Data-driven stochastic ac-opf using gaussian process regression," *International Journal of Electrical Power & Energy Systems*, vol. 152, p. 109249, 2023. 9

[26] A. Yadav, R. Bareth, M. Kochar, M. Pazoki, and R. A. E. Sehiemy, "Gaussian process regression-based load forecasting model," *IET Generation, Transmission & Distribution*, vol. 18, no. 5, pp. 899–910, 2024. 9, 10

[27] P. Pareek and H. D. Nguyen, "A framework for analytical power flow solution using gaussian process learning," *IEEE Transactions on Sustainable Energy*, vol. 13, no. 1, pp. 452–463, 2022. 9

[28] D. Cao, J. Zhao, W. Hu, Y. Zhang, Q. Liao, Z. Chen, and F. Blaabjerg, "Robust deep gaussian process-based probabilistic electrical load forecasting against anomalous events," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1142–1153, 2022. 10, 11

[29] Y. Guo, K. Baker, E. Dall'Anese, Z. Hu, and T. H. Summers, "Data-based distributionally robust stochastic optimal power flow—part i: Methodologies," *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1483–1492, 2019. 10, 38

[30] ——, "Data-based distributionally robust stochastic optimal power flow—part ii: Case studies," *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1493–1503, 2019. 10

[31] O. Alaca, A. R. Ekti, A. Wilson, J. Holliman, E. Piersall, S. Yarkan, and N. Stenvig, "Detection of grid-signal distortions using the spectral correlation function," *IEEE Transactions on Smart Grid*, vol. 14, no. 6, pp. 4980–4983, 2023. 12, 47

[32] C. Li, J. Zhang, and H. Wen, "Accurate and fast amplitude estimation of signal distorted by noise and harmonics for control of vsi," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 12, pp. 12 584–12 594, 2021. 12

[33] N. Veerakumar, D. Ćetenović, K. Kongurai, M. Popov, A. Jongepier, and V. Terzija, "Pmu-based real-time distribution system state estimation considering anomaly detection, discrimination and identification," *International Journal of Electrical Power & Energy Systems*, vol. 148, p. 108916, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061522009127 12

[34] S. Chen, Y. Yang, X. Dong, G. Xing, Z. Peng, and W. Zhang, "Warped variational mode decomposition with application to vibration signals of varying-speed rotating machineries," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 8, pp. 2755–2767, 2019. 13

[35] J. R. Stroud, M. L. Stein, and S. Lysen, "Bayesian and maximum likelihood estimation for gaussian processes on an incomplete lattice," *Journal of Computational and Graphical Statistics*, vol. 26, no. 1, pp. 108–120, 2017. 13, 15

[36] J. Guinness, "Permutation and grouping methods for sharpening gaussian process approximations," *Technometrics*, vol. 60, no. 4, pp. 415–429, 2018. 14

[37] J.-F. Ton, S. Flaxman, D. Sejdinovic, and S. Bhatt, "Spatial mapping with gaussian processes and nonstationary fourier features," *Spatial Statistics*, vol. 28, pp. 59–78, 2018. 14

[38] H. Huang and Y. Sun, "Hierarchical low rank approximation of likelihoods for large spatial datasets," *Journal of Computational and Graphical Statistics*, vol. 27, no. 1, pp. 110–118, 2018. 16

[39] A. O. Finley, A. Datta, B. D. Cook, D. C. Morton, H. E. Andersen, and S. Banerjee, "Efficient algorithms for bayesian nearest neighbor gaussian processes," *Journal of Computational and Graphical Statistics*, vol. 28, no. 2, pp. 401–414, 2019. 16

[40] Z. Liu, B. Tang, X. He, Q. Qiu, and F. Liu, "Class-specific random forest with cross-correlation constraints for spectral–spatial hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 2, pp. 257–261, 2017. 18

[41] D. Lu, P. Mausel, E. Brondízio, and E. Moran, "Change detection techniques," *International Journal of Remote Sensing*, vol. 25, no. 12, pp. 2365–2401, 2004. 18

[42] G. Camps-Valls, L. Martino, D. H. Svendsen, M. Campos-Taberner, J. Muñoz-Marí, V. Laparra, D. Luengo, and F. J. García-Haro, "Physics-aware gaussian processes in remote sensing," *Applied Soft Computing*, vol. 68, pp. 69–82, 2018. 19

[43] A. Mateo-Sanchis, J. Muñoz-Marí, A. Pérez-Suay, and G. Camps-Valls, "Warped gaussian processes in remote sensing parameter estimation and causal inference," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 11, pp. 1647–1651, 2018. 22

[44] J. Kim and S. Kim, "Temperature estimation adaptive to variables over distance using infrared–lidar," *Applied Sciences*, vol. 11, no. 9, 2021. 23

[45] J. Kay, R. Handcock, A. Gillespie, C. Konrad, S. Burges, N. Naveh, and D. Booth, "Stream-temperature estimation from thermal infrared images," in *IGARSS 2001.*

Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No.01CH37217), vol. 1, 2001, pp. 112–114. 23

[46] B. Jalil, M. A. Pascali, G. R. Leone, M. Martinelli, D. Moroni, and O. Salvetti, "To identify hot spots in power lines using infrared and visible sensors," in *Multimedia and Network Information Systems*. Springer International Publishing, 2019, pp. 313–321. 23

[47] I. Pobočíkováa, Z. Sedliačkováa, and M. Michalkováa*, "Application of four probability distributions for wind speed modeling," *Procedia Engineering*, vol. 192, pp. 713–718, 2017. 24, 25

[48] V. Hoolohan, A. S. Tomlin, and T. Cockerill, "Improved near surface wind speed predictions using gaussian process regression combined with numerical weather predictions and observed meteorological data," *Renewable Energy*, vol. 126, pp. 1043–1054, 2018. 24

[49] S. Zhu, X. Yuan, Z. Xu, X. Luo, and H. Zhang, "Gaussian mixture model coupled recurrent neural networks for wind speed interval forecast," *Energy Conversion and Management*, vol. 198, p. 111772, 2019. 24

[50] H. Wang, Y. Zhang, and J. Mao, "Sparse gaussian process regression for multi-step ahead forecasting of wind gusts combining numerical weather predictions and on-site measurements," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 220, 2022. 24

[51] P. Kou, F. Gao, and X. Guan, "Sparse online warped gaussian process for wind power probabilistic forecasting," *Applied Energy*, vol. 108, pp. 410–428, 2013. 25, 26

[52] M. Shekaramiz, T. K. Moon, and J. H. Gunther, "Exploration and data refinement via multiple mobile sensors based on gaussian processes," in *Asilomar Conference on Signals, Systems, and Computers*, 2017, pp. 885–889. 26

[53] K.-C. Ma, L. Liu, and G. S. Sukhatme, "Informative planning and online learning with sparse gaussian processes," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4292–4298. 26

[54] T. Sanislav, G. D. Mois, S. Zeadally, and S. C. Folea, "Energy harvesting techniques for internet of things (iot)," *IEEE Access*, vol. 9, pp. 39 530–39 549, 2021. 27

[55] X. Zeng, Z. Yang, P. Wu, L. Cao, and Y. Luo, "Power source based on electric field energy harvesting for monitoring devices of high-voltage transmission line," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7083–7092, 2021. 27

[56] G. Stamatakis, N. Pappas, and A. Traganitis, "Control of status updates for energy harvesting devices that monitor processes with alarms," in *2019 IEEE Globecom Workshops (GC Wkshps)*, 2019, pp. 1–6. 27, 28

[57] E. Foley, "A warped gaussian process tutorial using a simple example," *International Research Journal of Engineering and Technology*, vol. 11, pp. 417–420, 06 2024. [Online]. Available: https://www.irjet.net/volume11-issue6 29

[58] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*. Cambridge, United Kingdom: Cambridge University Press, 2023. 29

[59] C. Sammut and G. Webb, Eds., *Encyclopedia of Machine Learning*. Berlin: Springer, 2010. 32

[60] D. Duvenaud, "Automatic model construction with gaussian processes," PhD thesis, University of Cambridge, Cambridge, England, June 2014, available at https://www.repository.cam.ac.uk/handle/1810/247281. 33

[61] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications (Springer Texts in Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2005. 35

[62] U. of Washington ECE, "Power systems test case archive," Available: http://labs.ece.uw.edu/pstca/pf14/pg_tca14bus.htm. 36, 37, 98

[63] L. Thurner, A. Scheidler, F. Schäfer, J.-H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "pandapower - an open source python tool for convenient modeling, analysis and optimization of electric power systems," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6510–6521, 2018. 36

[64] K. Dvijotham, P. V. Hentenryck, M. Chertkov, S. Misra, and M. Vuffray, "Graphical models for optimal power flow," 2016. 38

[65] J. Snoek, K. Swersky, R. Zemel, and R. Adams, "Input warping for bayesian optimization of non-stationary functions," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 32, no. 2. PMLR, 2014, pp. 1674–1682. 39

[66] J. Wiebe, I. Cecílio, J. Dunlop, and R. Misener, "A robust approach to warped gaussian process-constrained optimization," *Mathematical Programming*, vol. 196, no. 1, pp. 805–839, 2022. 39, 41

[67] S. Lakshminarayana, S. Sthapit, and C. Maple, "A comparison of data-driven techniques for power grid parameter estimation," 2021. 42

[68] P. Denholm, T. Mai, R. W. Kenyon, B. Kroposki, and M. O'Malley, "Inertia and the power grid: A guide without the spin," NREL/TP-6120-73856, Golden, CO: National Renewable Energy Laboratory, 2020. [Online]. Available: https://www.nrel.gov/docs/fy20osti/73856.pdf 42

[69] J. L. Rodríguez-Amenedo and S. A. Gómez, "Damping low-frequency oscillations in power systems using grid-forming converters," *IEEE Access*, vol. 9, pp. 158 984–158 997, 2021. 42

[70] P. Arinushkin and T. Vadivasova, "Nonlinear damping effects in a simplified power grid model based on coupled kuramoto-like oscillators with inertia," *Chaos, Solitons & Fractals*, vol. 152, 2021. 42

[71] A. K. Barnes and A. Mate, "Dynamic state estimation for radial microgrid protection," in *2021 IEEE/IAS 57th Industrial and Commercial Power Systems Technical Conference (I&CPS)*, 2021, pp. 1–9. 42

[72] A. Gomez-Exposito, A. J. Conejo, and C. Canizares, *Electric Energy Systems Analysis and Operation*. Boca Raton: CRC Press, 2018. 43, 48

[73] A. J. Wilson, D. R. Reising, R. W. Hay, and R. C. Johnson, "Automated fuse identification within an operational smart power grid," in *2018 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, 2018, pp. 719–723. 47

[74] H. Verdejo, P. Moreira, W. Kliemann, C. Becker, and J. Delpiano, "An analytical model for small signal stability analysis in unbalanced electrical power systems," *Applied Sciences*, vol. 10, no. 24, 2020. 48

[75] G. B. Giannakis, V. Kekatos, N. Gatsis, S.-J. Kim, H. Zhu, and B. F. Wollenberg, "Monitoring and optimization for power grids: A signal processing perspective," *IEEE Signal Processing Magazine*, vol. 30, no. 5, pp. 107–128, 2013. 48

[76] S. Madabhushi and R. Dewri, "A survey of anomaly detection methods for power grids," *International Journal of Information Security*, vol. 22, pp. 1799–1832, 2023. 48

[77] O. R. N. Laboratory and L. L. N. Laboratory, "Grid event signature library," Available: https://gesl.ornl.gov/. 48

[78] M. Toledo-Orozco, C. Arias-Marin, C. Álvarez Bel, D. Morales-Jadan, J. Rodríguez-García, and E. Bravo-Padilla, "Innovative methodology to identify errors in electric energy measurement systems in power utilities," *Energies*, vol. 14, no. 4, 2021. 49

[79] M. Camponogara, D. P. Bernardon, T. B. Marchesan, V. C. Bender, F. C. Pepe, G. J. G. d. Santos, L. M. d. Chiara, and M. Wolter, "Estimation of electric arc incident energy in electric power distribution grids," *Energies*, vol. 16, no. 7, 2023. 49

[80] W. Danilczyk, Y. L. Sun, and H. He, "Smart grid anomaly detection using a deep learning digital twin," in *2020 52nd North American Power Symposium (NAPS)*, 2021, pp. 1–6. 49

[81] J. Cifuentes, G. Marulanda, A. Bello, and J. Reneses, “Air temperature forecasting using machine learning techniques: A review,” *Energies*, vol. 13, no. 16, 2020. 51

[82] T. T. K. Tran, S. M. Bateni, S. J. Ki, and H. Vosoughifar, “A review of neural networks for air temperature forecasting,” *Water*, vol. 13, no. 9, 2021. 51

[83] E. S. I. Group, “Weather dataset needs for planning and analyzing modern power systems,” Summary Report, Washington, DC, USA, 2023. [Online]. Available: https://www.esig.energy/weather-data-for-power-system-planning/ 51

[84] E. Aguilar Madrid and N. Antonio, “Short-term electricity load forecasting with machine learning,” *Information*, vol. 12, no. 2, 2021. 51, 71

[85] D. Dickey and W. Fuller, “Distribution of the estimators for autoregressive time series with a unit root,” *JASA. Journal of the American Statistical Association*, vol. 74, 06 1979. 53, 96

[86] D. Kwiatkowski, P. C. Phillips, P. Schmidt, and Y. Shin, “Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root?” *Journal of Econometrics*, vol. 54, no. 1, pp. 159–178, 1992. [Online]. Available: https://www.sciencedirect.com/science/article/pii/030440769290104Y 53, 96

[87] Z. Sun, D. Ma, and S. Li, “An oscillation damping method for frequency-detector-less virtual synchronous generators,” *Energy Reports*, vol. 7, pp. 1485–1494, 2021, 2021 International Conference on Energy Engineering and Power Systems. 61

[88] A. S. Association, “The role of statistics in data science and artificial intelligence,” Available: https://www.amstat.org/policy-and-advocacy/asa-board-statements, August 2023. 78

[89] S. Friedrich, G. Antes, S. Behr, H. Binder, W. Brannath, F. Dumpert, K. Ickstadt, H. Kestler, J. Lederer, H. Leitgöb, M. Pauly, A. Steland, A. Wilhelm, and T. Friede, “Is there a role for statistics in artificial intelligence?” *Advances in Data Analysis and Classification*, vol. 16, pp. 823–846, 2022. 78

[90] N. Oz, N. Sochen, D. Mendelovich, and I. Klapp, "Estimating temperatures with low-cost infrared cameras using deep neural networks," 2024. 78

[91] M. Aljubran, C. Nwosu, E. Okoroafor, K. Ochie, C. Smith, and H. Gudmundsdottir, "Recent trends in artificial intelligence for subsurface geothermal applications," in *Workshop on Geothermal Resovoir Engineering*, 02 2022. 79

[92] G. Strang, *Linear Algebra and Its Applications*, 4th ed. Boston: Cengage Learning, 2006. 95

Appendices

A Supplementary Functions and Definitions

A.1 The Kronecker Delta Function

The Kronecker delta [17] is defined as

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

A.2 Jacobian Matrix

Let

$$\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$$

be a vector-valued function defined by

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(x_1, \ldots, x_n) \\ f_2(x_1, \ldots, x_n) \\ \vdots \\ f_m(x_1, \ldots, x_n) \end{bmatrix} \quad \text{where } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}.$$

The Jacobian matrix [92] of $\mathbf{f}$ at $\mathbf{x} \in \mathbb{R}^n$, denoted by

$$J_{\mathbf{f}}(\mathbf{x}) = D\mathbf{f}(\mathbf{x}) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}},$$

is the $m \times n$ matrix of all first-order partial derivatives of the component functions f_i, defined as

$$J_{\mathbf{f}}(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}.$$

Each row of $J_{\mathbf{f}}(\mathbf{x})$ corresponds to the gradient of a component function f_i, and each column corresponds to the partial derivatives of all f_i with respect to a specific variable x_j.

A.3 Augmented Dickey-Fuller Test

The Augmented Dickey-Fuller (ADF) [85] test determines whether a time series has a unit root; or that a time series is non-stationary.

$$H_0 : \gamma = 0 \quad \text{(unit root present, non-stationary)}$$

$$H_1 : \gamma < 0 \quad \text{(no unit root, stationary)}$$

A.4 Kwiatkowski-Phillips-Schmidt-Shin Test

The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) [86] test determines whether a time series has a unit root; or that a time series is non-stationary.

The null and alternative hypotheses of the KPSS test are as follows:

$$H_0 : \text{The time series is stationary (around a deterministic trend or level).}$$

$$H_1 : \text{The time series is non-stationary (has a unit root).}$$

B ARIMA: Statistical Test Results

Table 1: ADF and KPSS test results for IEEE Bus 14 simulated power load forecasting.

	Test Statistic	P-value
ADF	-4.9107	3.3245e-05
KPSS	0.1055	0.1000

Table 2: ADF and KPSS test results for multivariate weather load forecasting.

	ADF statistic	ADF p-value	KPSS statistic	KPSS p-value
Load	-3.9284	0.0018	0.07887	0.1000
Toc Temp	-3.3521	0.0127	0.0448	0.1000
Toc Humidity	-2.7623	0.0639	0.0916	0.1000
Toc Precipitation	-3.0778	0.0282	0.1113	0.1000
Toc Wind	-2.3590	0.1536	0.0953	0.1000
Dav Temp	-3.3014	0.0148	0.0476	0.1000
Dav Humidity	-3.0581	0.0298	0.1003	0.1000
Dav Precipitation	-3.3832	0.0115	0.0607	0.1000
Dav Wind	-4.8688	4.0085	0.0863	0.1000
San Temp	-3.6284	0.0052	0.0411	0.1000
San Humidity	-4.1271	0.0009	0.0995	0.1000
San Precipitation	-4.3556	0.0004	0.1611	0.0374
San Wind	-4.2384	0.0006	0.0616	0.1000

C IEEE 14 Bus System Data Tables

The data tables for the IEEE 14 Bus System [62].

Table 3: IEEE 14-bus system bus data

Bus No.	Bus Type	V	θ	P^M	Q^M	P^L	Q^L
1	swing	1.06	0	0	0	0	0
2	gen	1.045	0	0.4	0.424	0.217	0.127
3	gen	1.01	0	0	0.234	0.942	0.19
4	load	1	0	0	0	0.478	0
5	load	1	0	0	0	0.076	0.016
6	gen	1.07	0	0	0.122	0.112	0.075
7	load	1	0	0	0	0	0
8	gen	1.09	0	0	0.174	0	0
9	load	1	0	0	0	0.295	0.166
10	load	1	0	0	0	0.09	0.058
11	load	1	0	0	0	0.035	0.018
12	load	1	0	0	0	0.061	0.016
13	load	1	0	0	0	0.135	0.058
14	load	1	0	0	0	0.149	0.05

Table 4: IEEE 14-bus system line data

From Bus	To Bus	Resistance	Reactance	Half line charging admittance	Tap Ratio
1	2	0.01938	0.05917	0.0528	1
1	5	0.05403	0.22304	0.0492	1
2	3	0.04699	0.19797	0.0438	1
2	4	0.05811	0.17632	0.0374	1
2	5	0.05695	0.17388	0.0340	1
3	4	0.06701	0.17103	0.0346	1
4	5	0.01335	0.04211	0.0128	1
4	7	0	0.20912	0.0000	0.978
4	9	0	0.55618	0	0.969
5	6	0	0.25202	0	0.932
6	11	0.09498	0.19890	0	1
6	12	0.12291	0.25581	0	1
6	13	0.06615	0.13027	0	1
7	8	0	0.17615	0	1
7	9	0	0.11001	0	1
9	10	0.03181	0.08450	0	1
9	14	0.12711	0.27038	0	1
10	11	0.08205	0.19207	0	1
12	13	0.22092	0.19988	0	1
13	14	0.17093	0.34802	0	1

D IEEE Bus 14 Simulated Load Forecasts

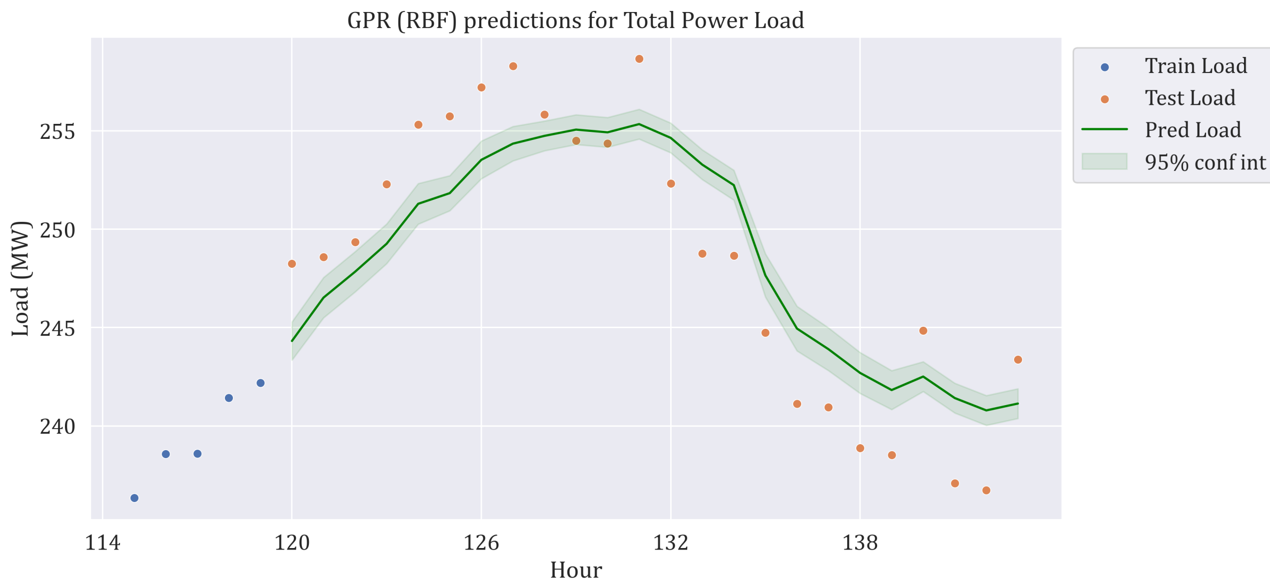

Figure 1: GP (RBF) 24 hour forecast for power load.

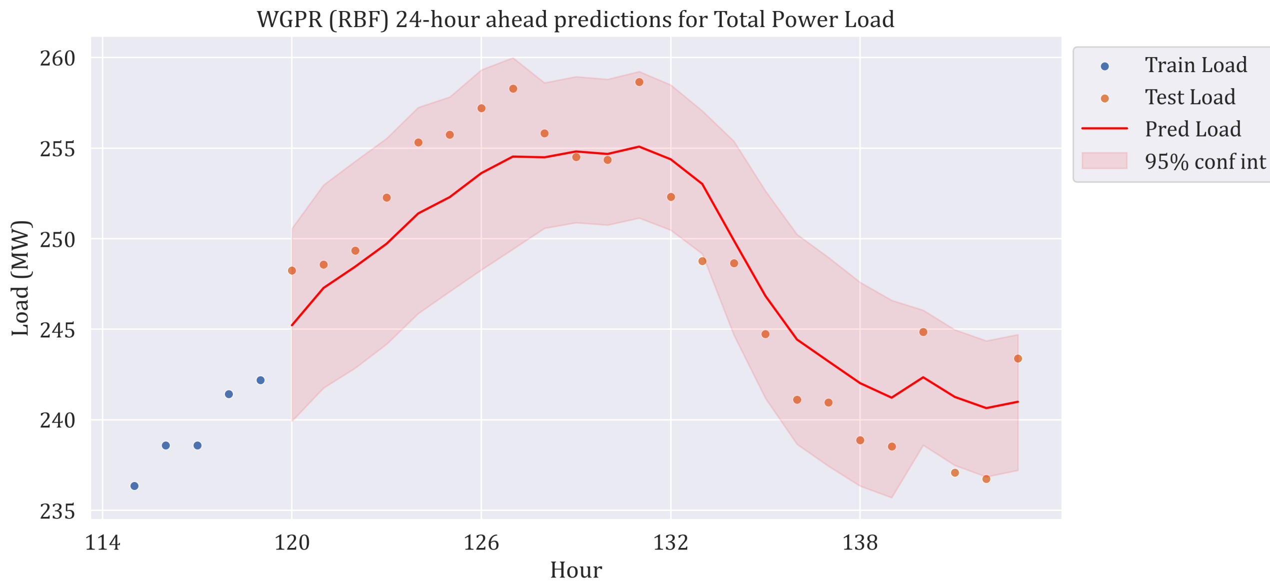

Figure 2: WGP (RBF) 24 hour forecast for power load.

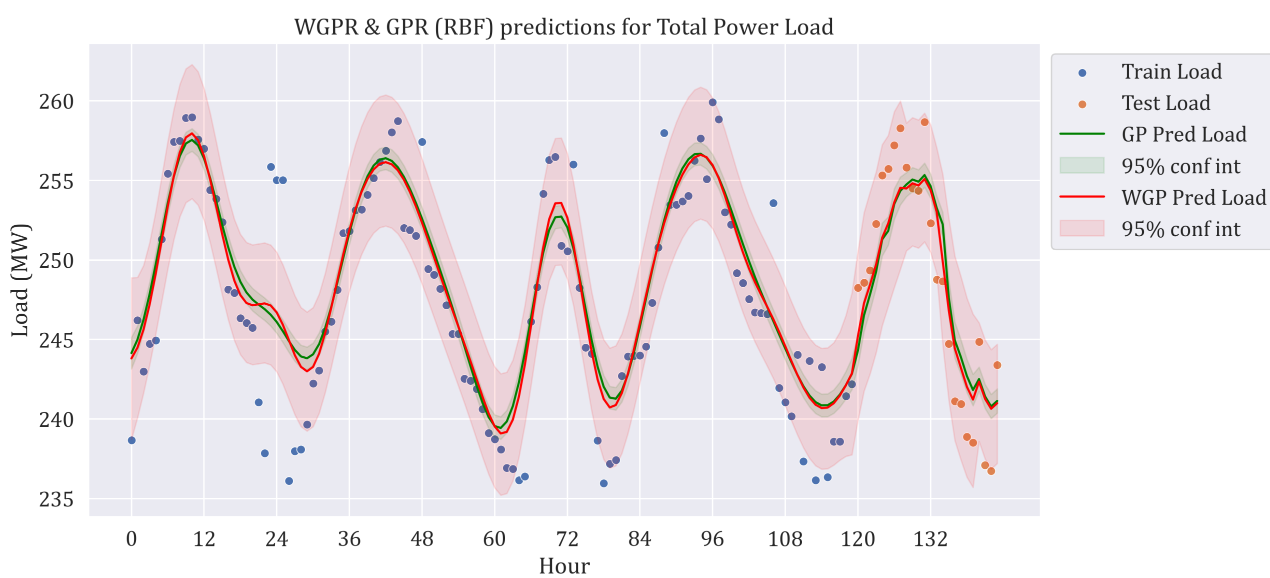

Figure 3: Comparison of GP and WGP models (RBF) over testing and training data.

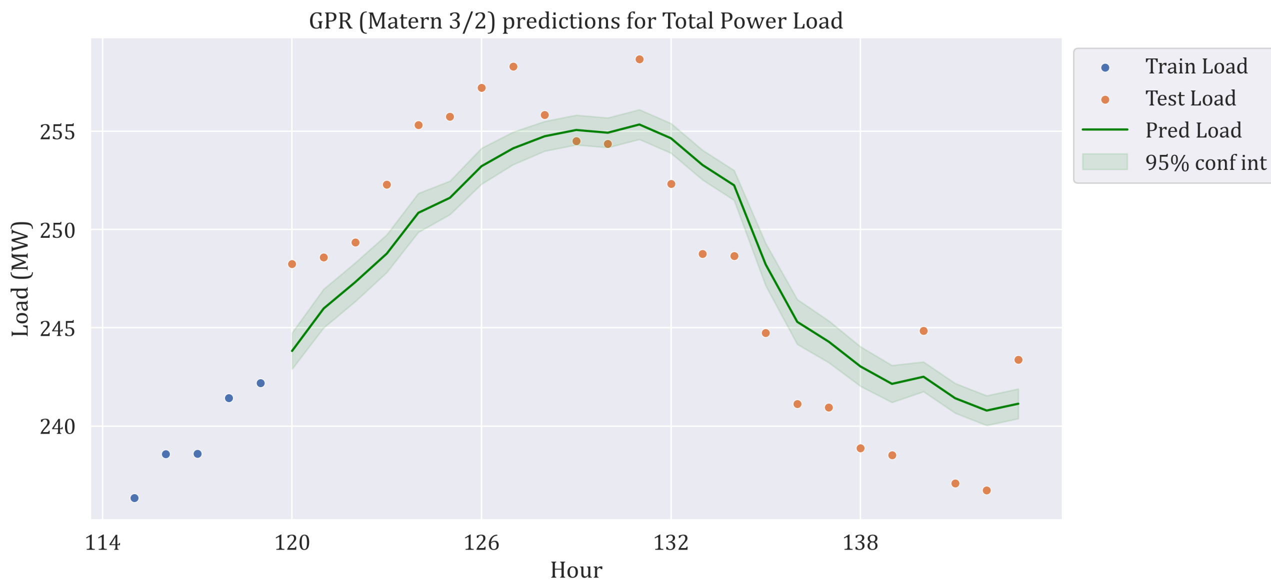

Figure 4: GP (M 3/2) 24 hour forecast for power load.

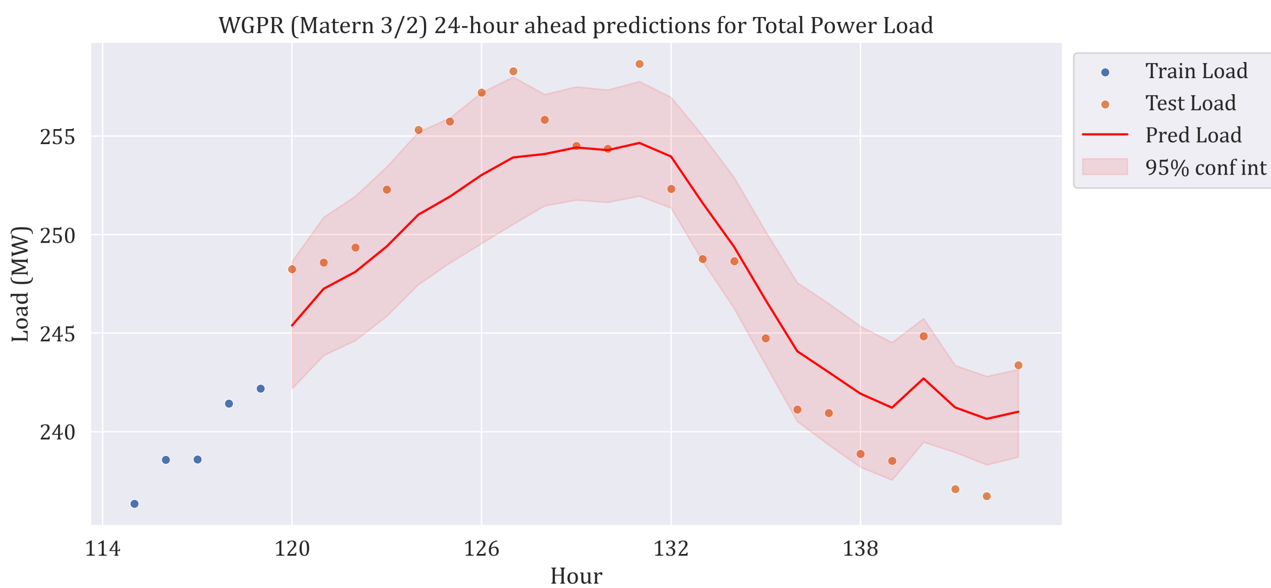

Figure 5: WGP (M 3/2) 24 hour forecast for power load.

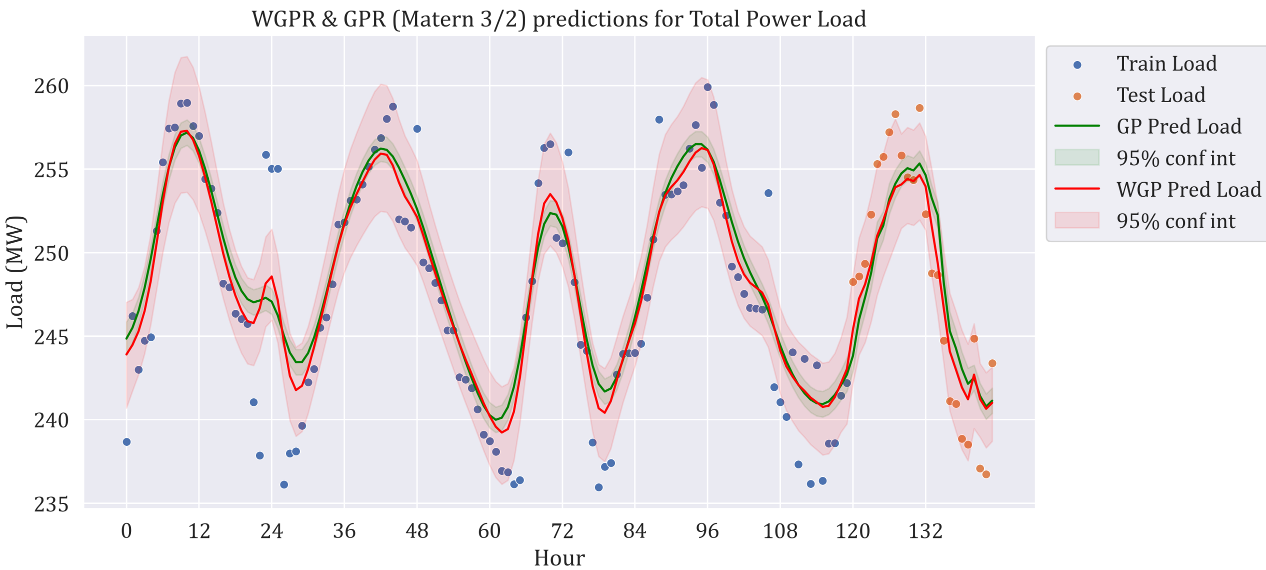

Figure 6: Comparison of GP and WGP models (M 3/2) over testing and training data.

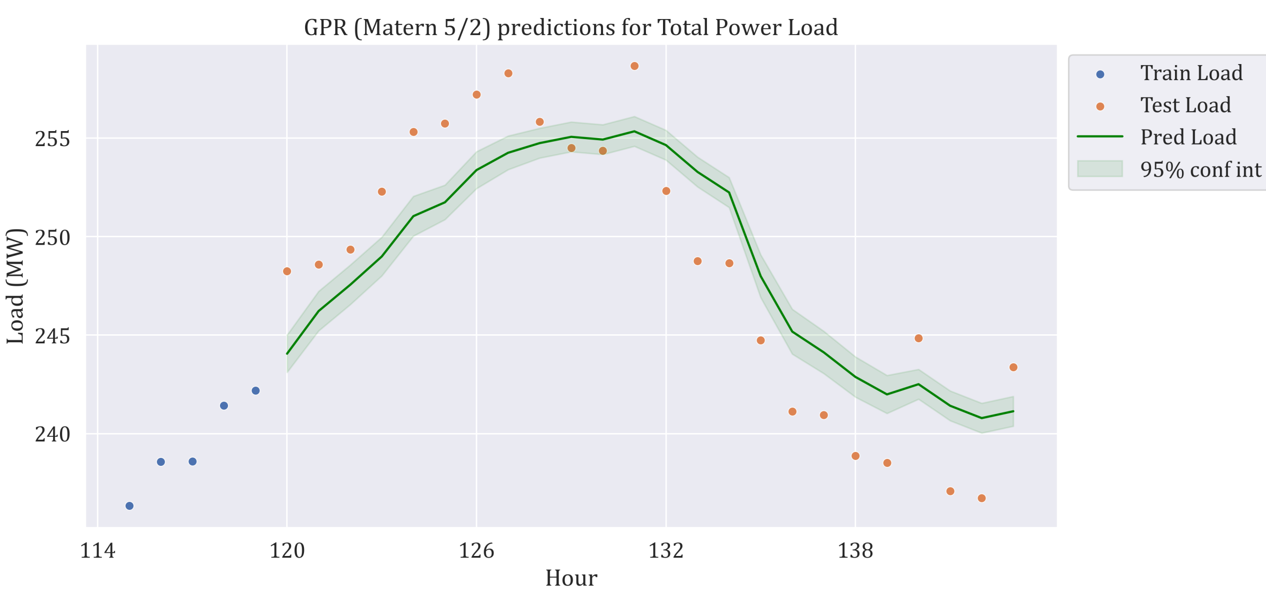

Figure 7: GP (M 5/2) 24 hour forecast for power load.

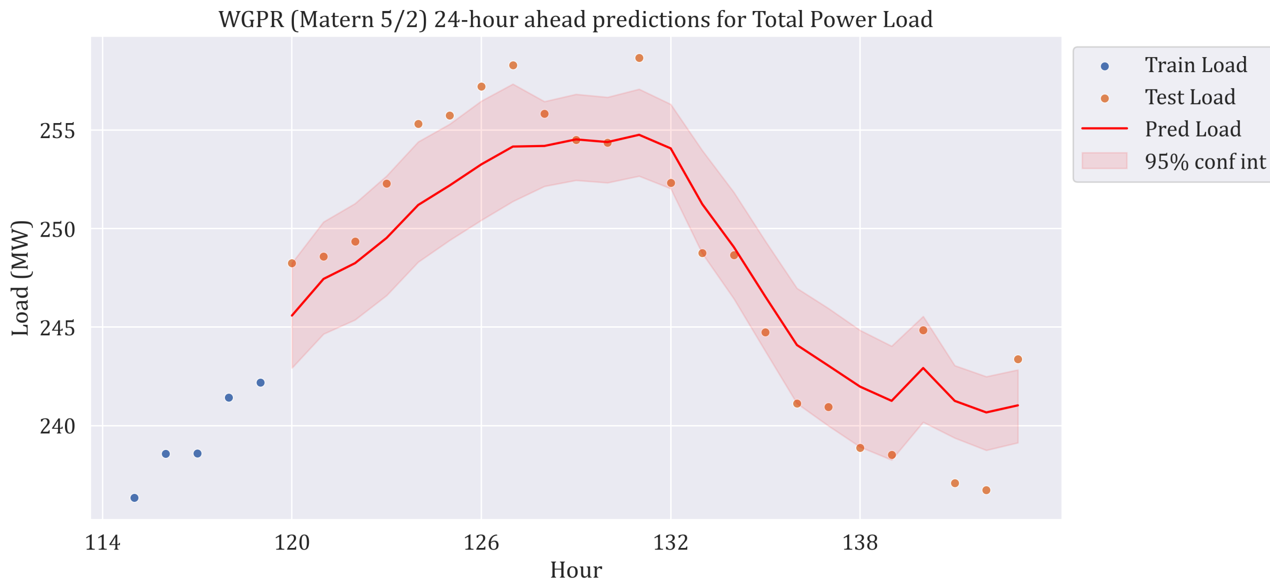

Figure 8: WGP (M 5/2) 24 hour forecast for power load.

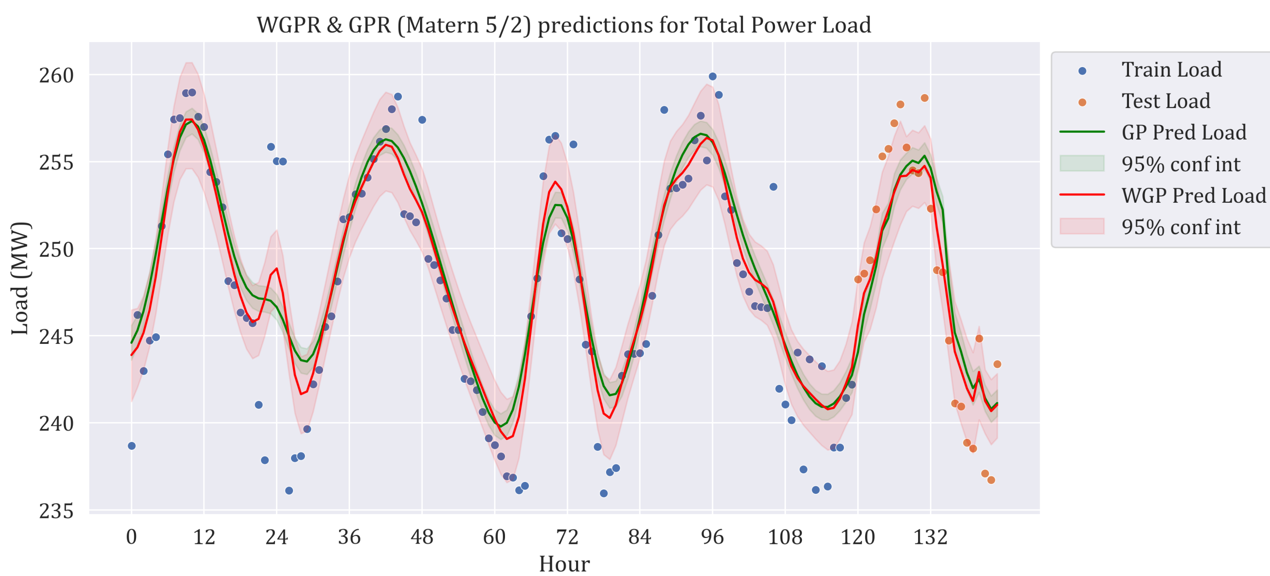

Figure 9: Comparison of GP and WGP models (M 5/2) over testing and training data.

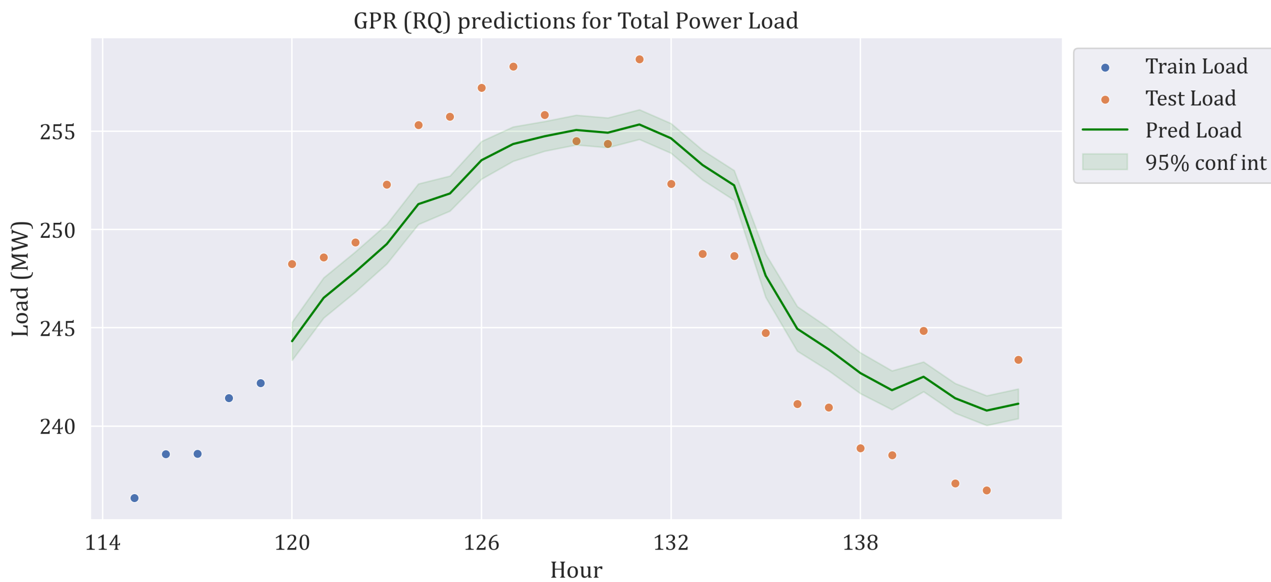

Figure 10: GP (RQ) 24 hour forecast for power load.

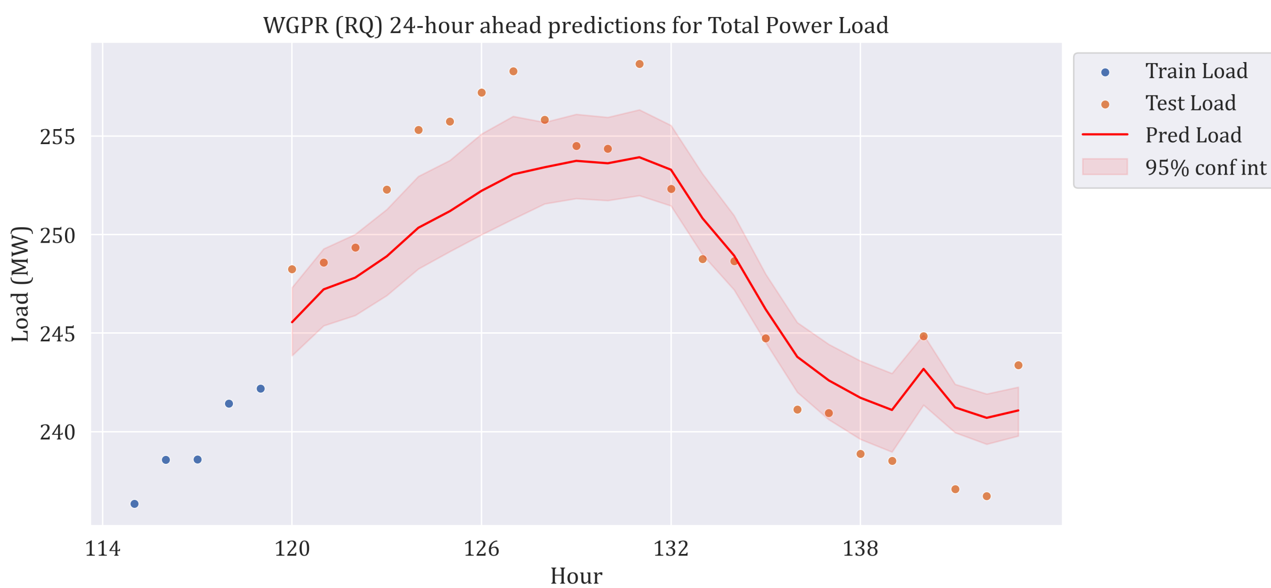

Figure 11: WGP (RQ) 24 hour forecast for power load.

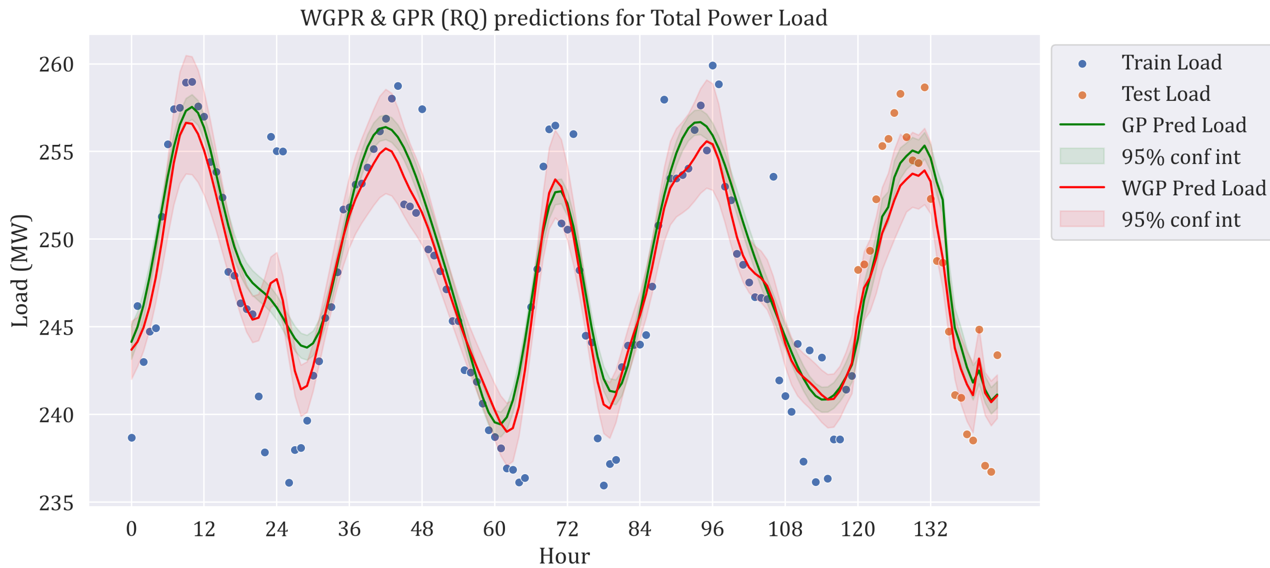

Figure 12: Comparison of GP and WGP models (RQ) over testing and training data.

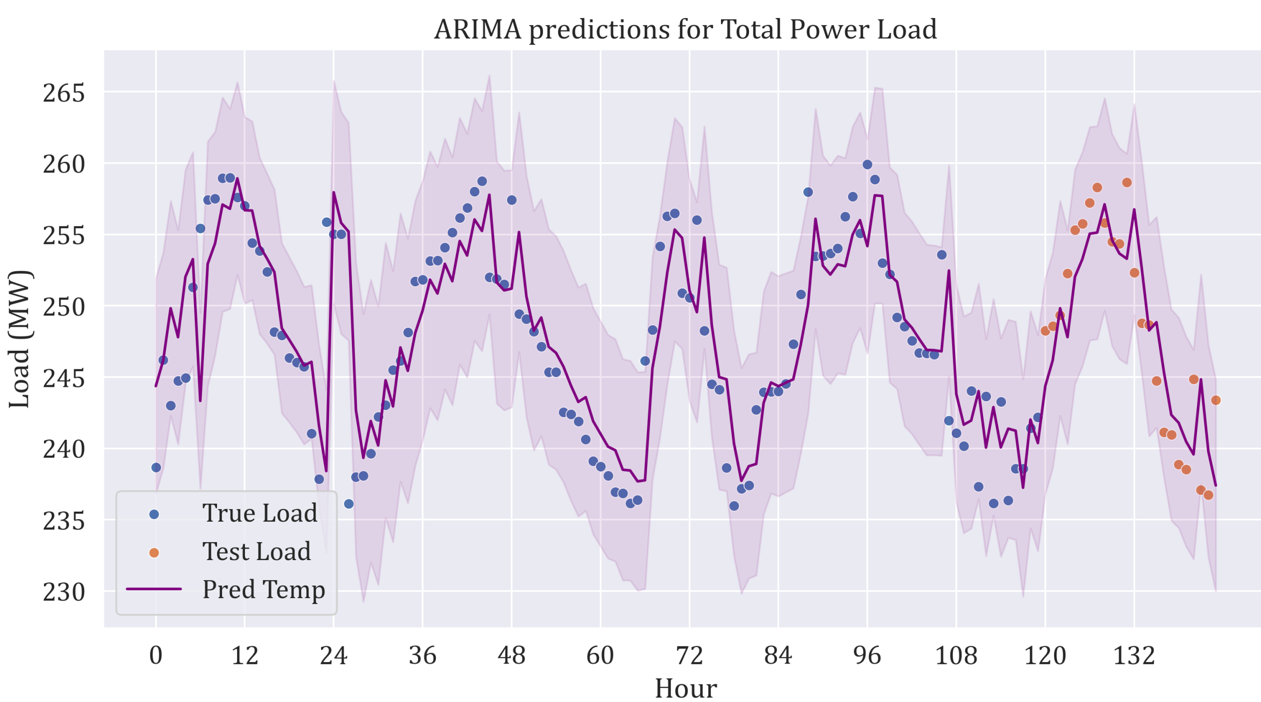

Figure 13: ARIMA model over testing and training data.

E Load + Weather Forecasts

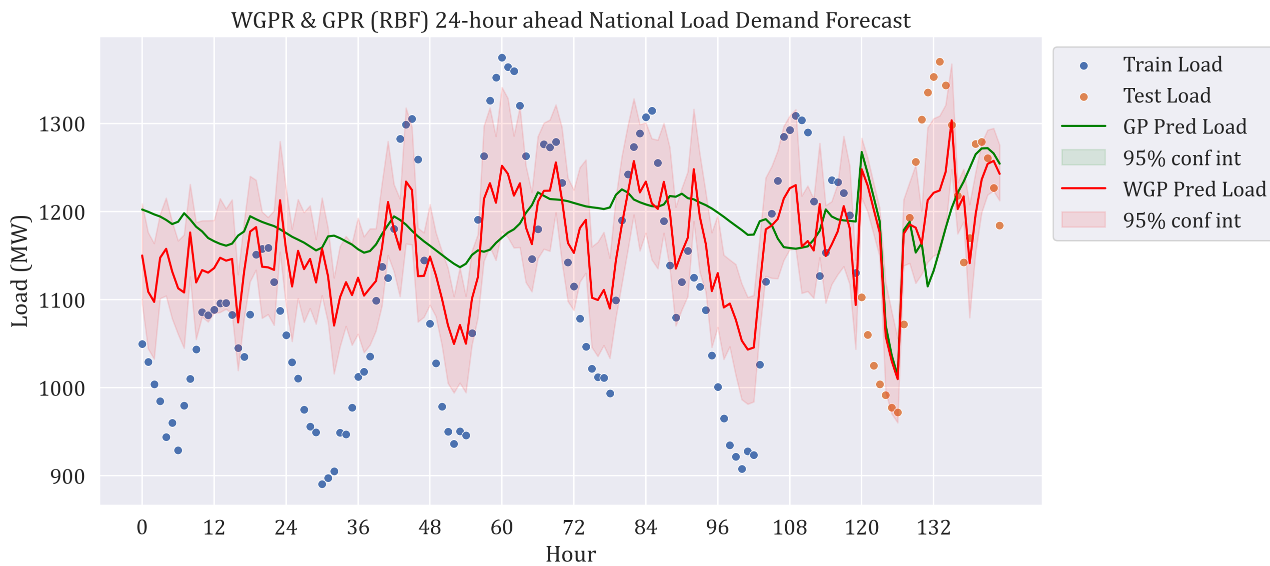

Figure 14: Humidity GP and WGP (RBF) 24 hour forecast for load demand.

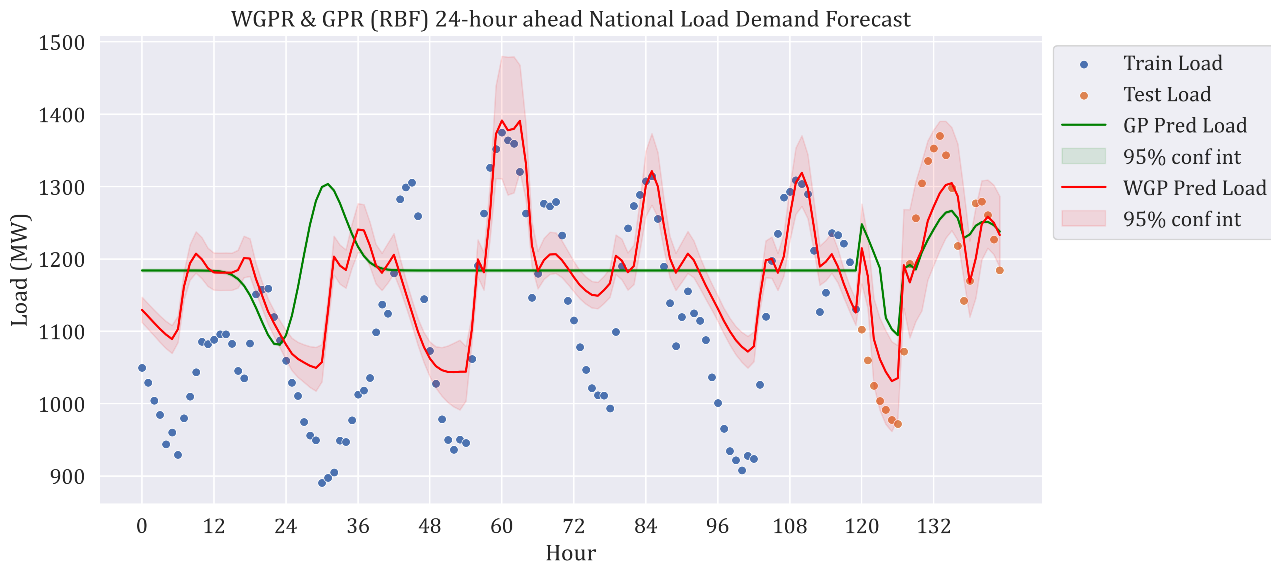

Figure 15: Temperature GP and WGP (RBF) 24 hour forecast for load demand.

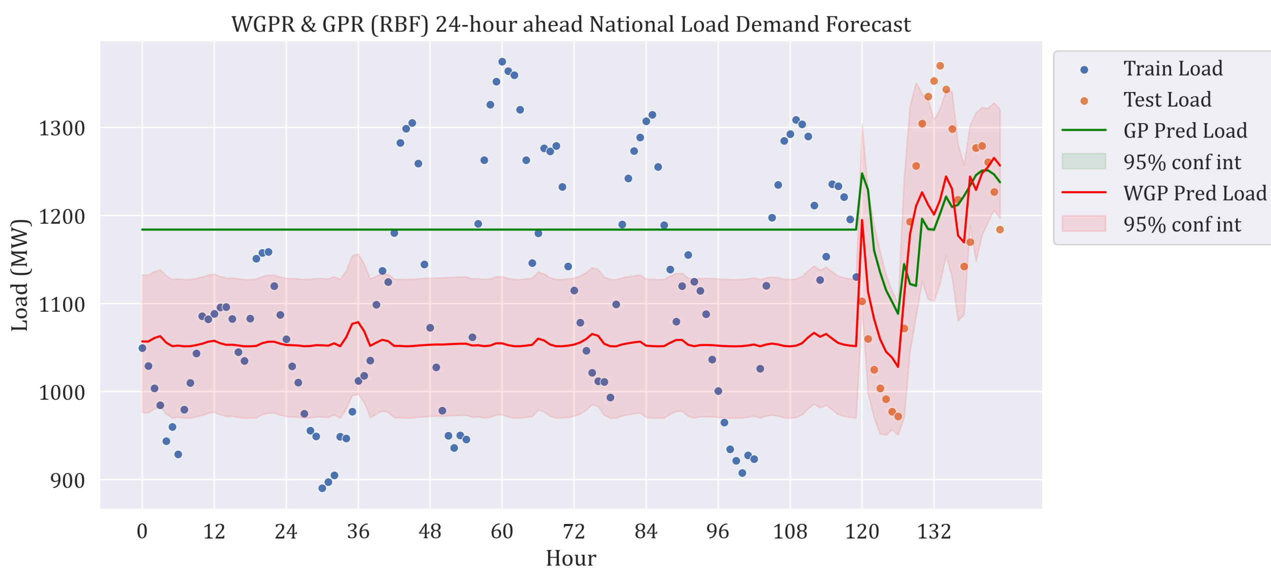

Figure 16: Wind Speed GP and WGP (RBF) 24 hour forecast for load demand.

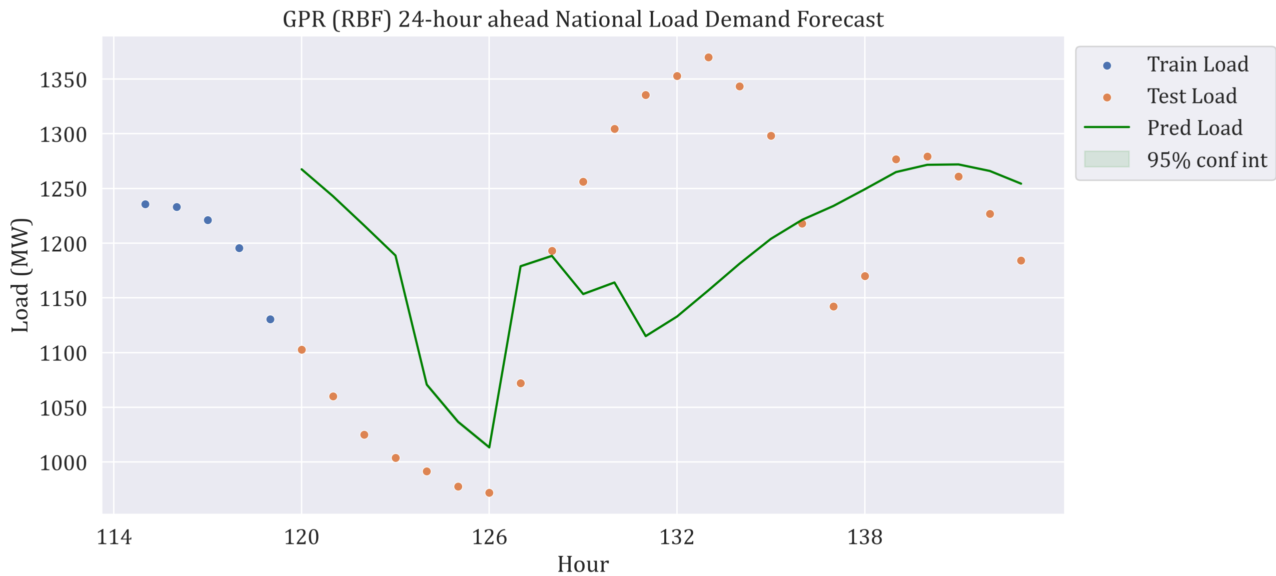

Figure 17: Humidity GP (RBF) 24 hour forecast for load demand.

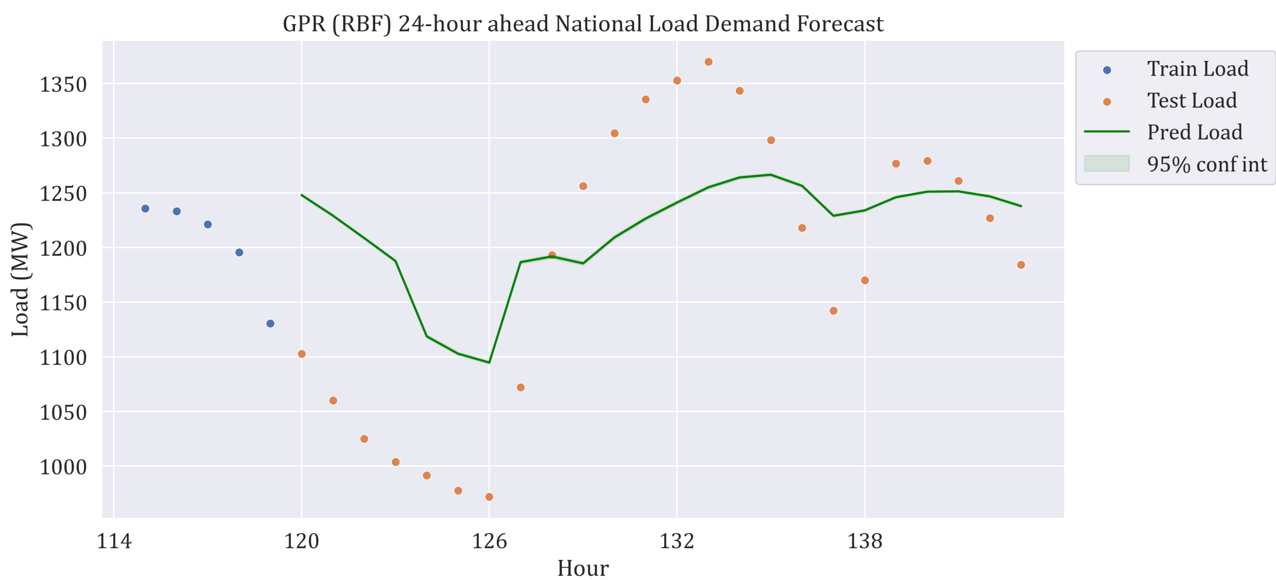

Figure 18: Temperature GP (RBF) 24 hour forecast for load demand.

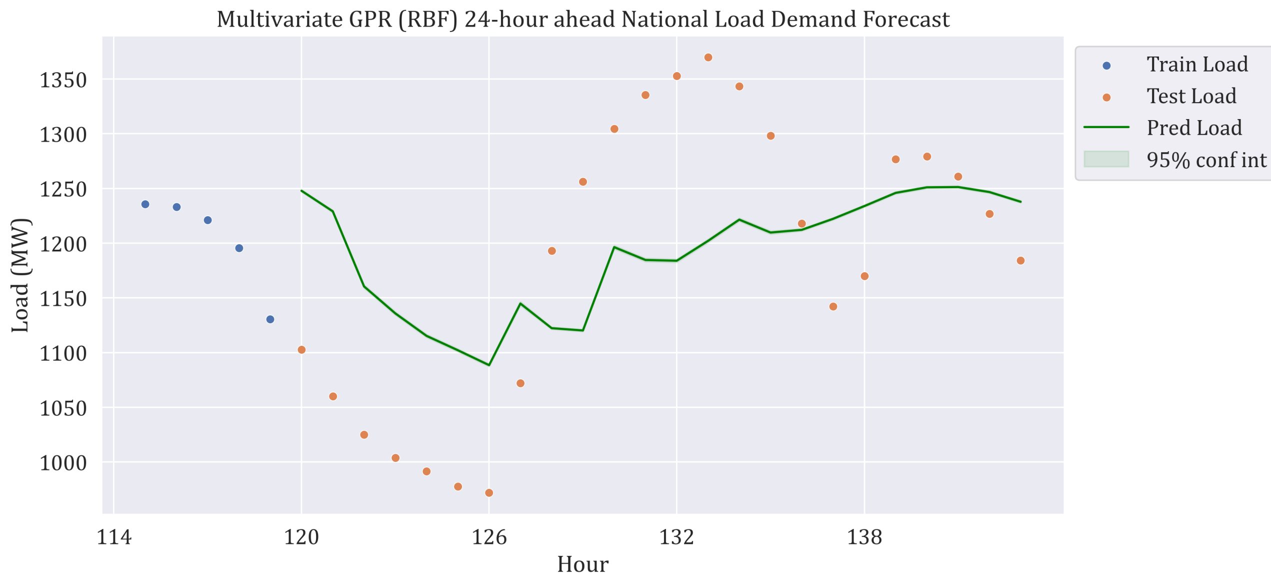

Figure 19: Wind Speed GP (RBF) 24 hour forecast for load demand.

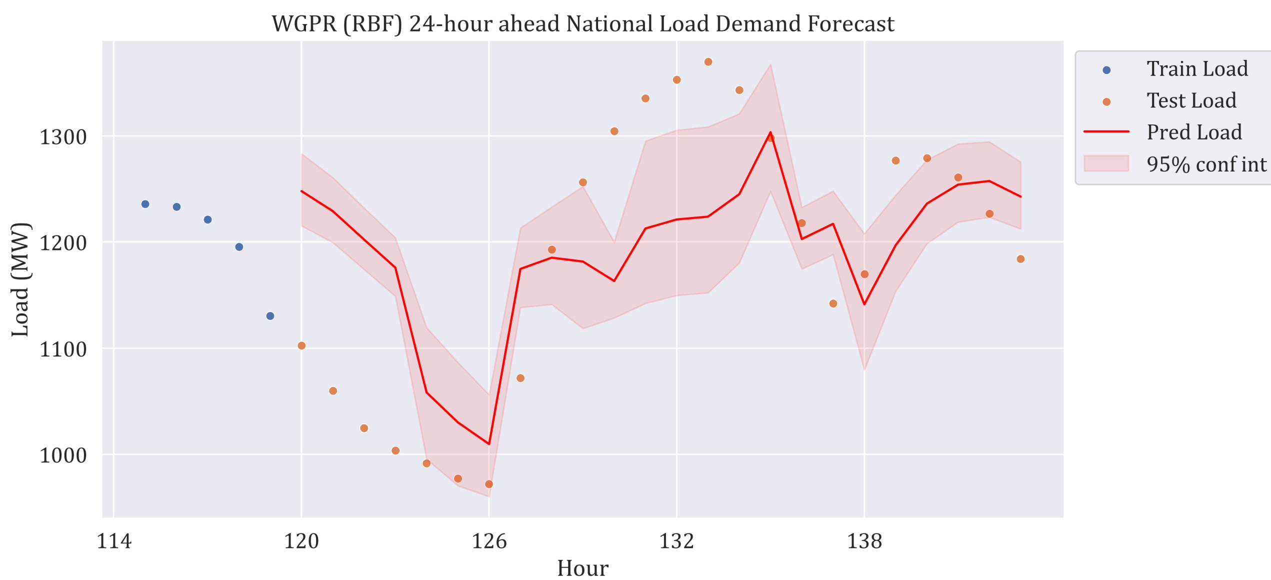

Figure 20: Humidity WGP (RBF) 24 hour forecast for load demand.

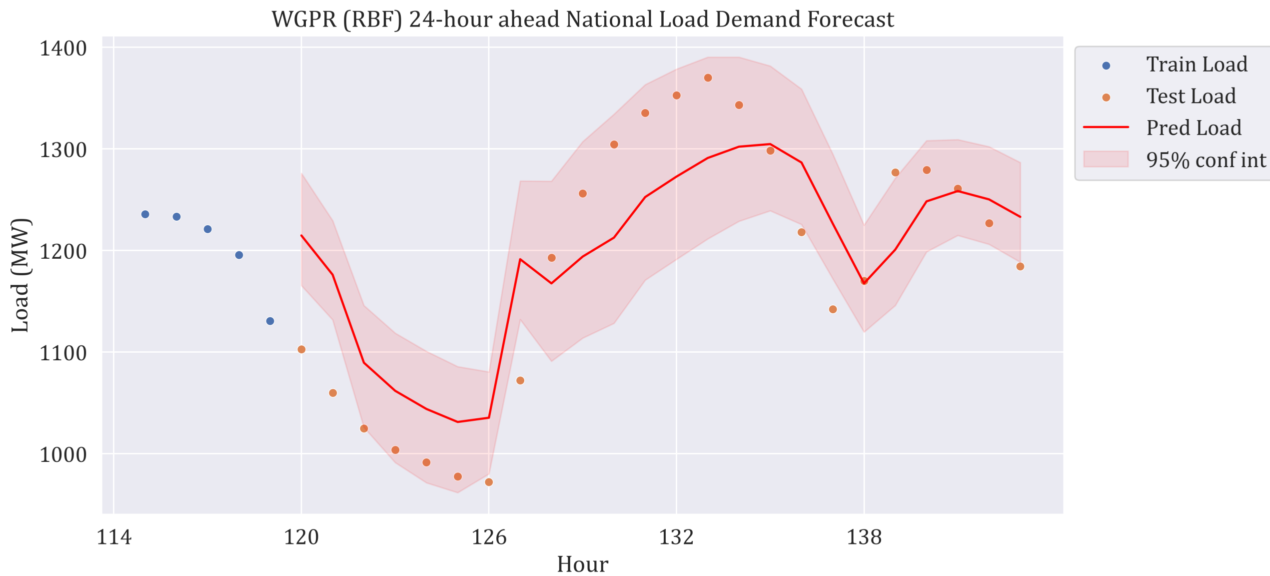

Figure 21: Temperature WGP (RBF) 24 hour forecast for load demand.

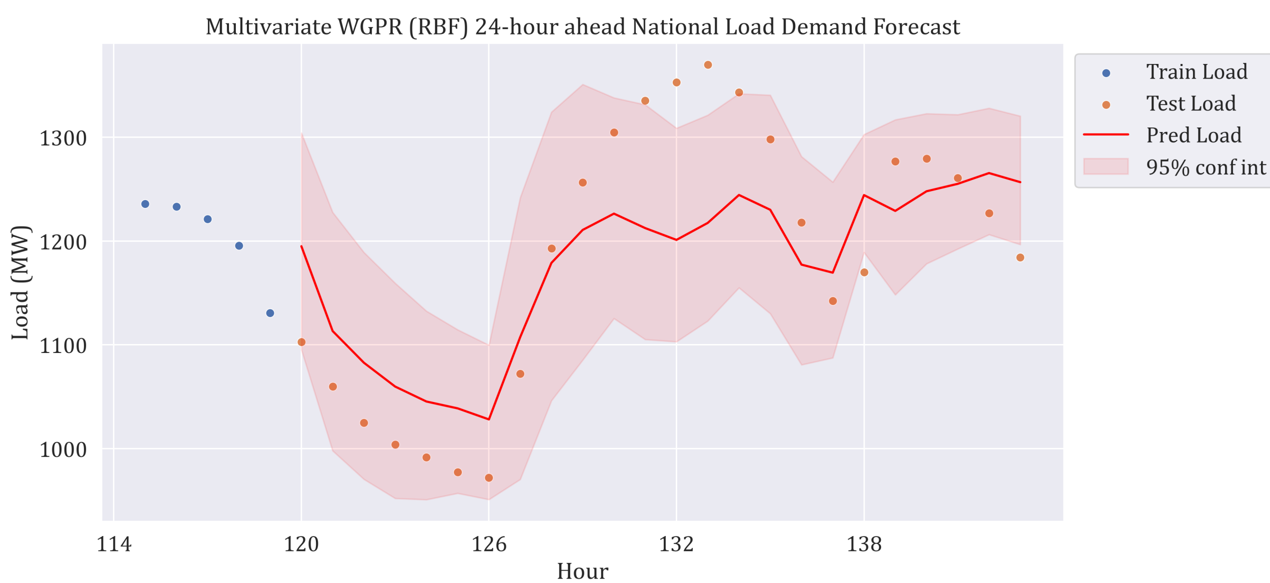

Figure 22: Wind Speed WGP (RBF) 24 hour forecast for load demand.

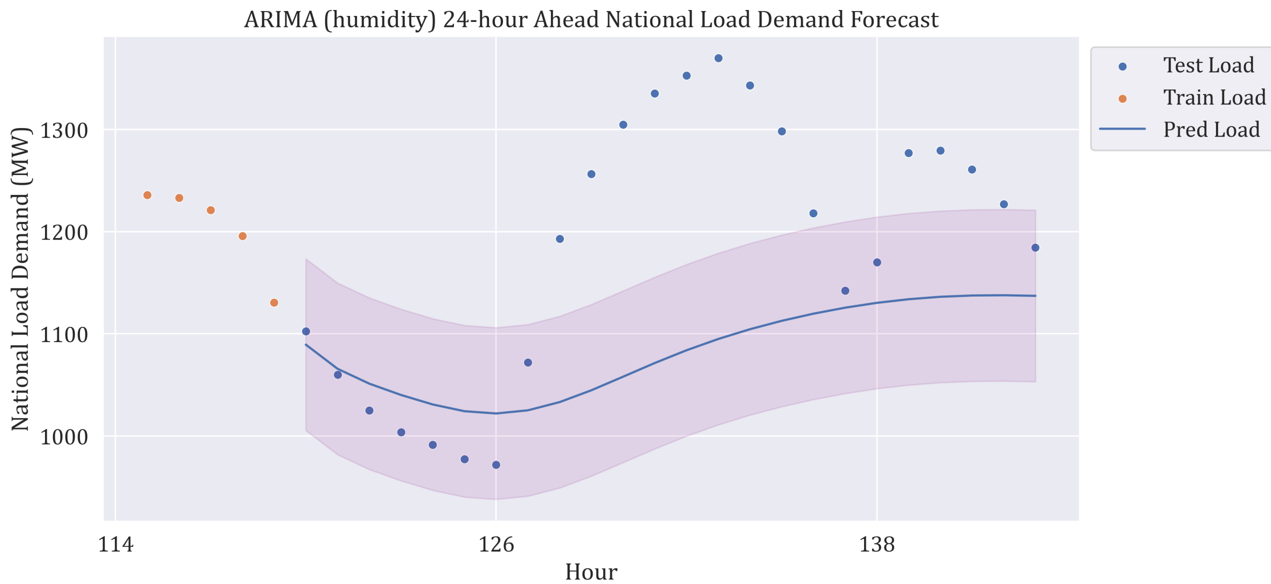

Figure 23: Humidity VAR 24 hour forecast for load demand.

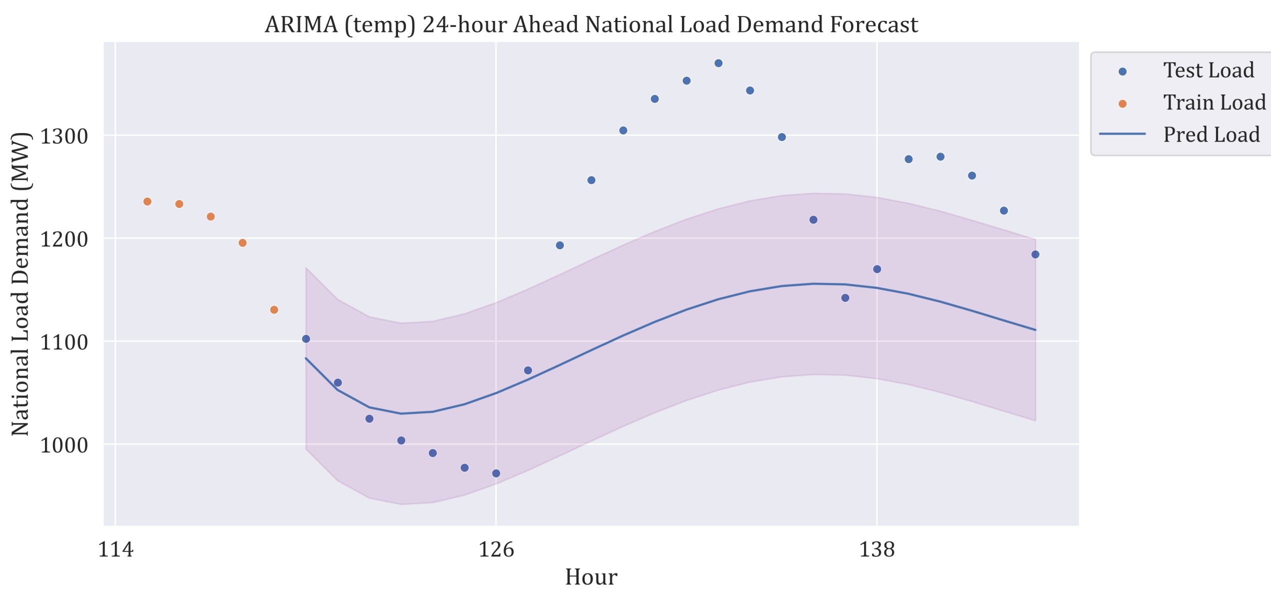

Figure 24: Temperature VAR 24 hour forecast for load demand.

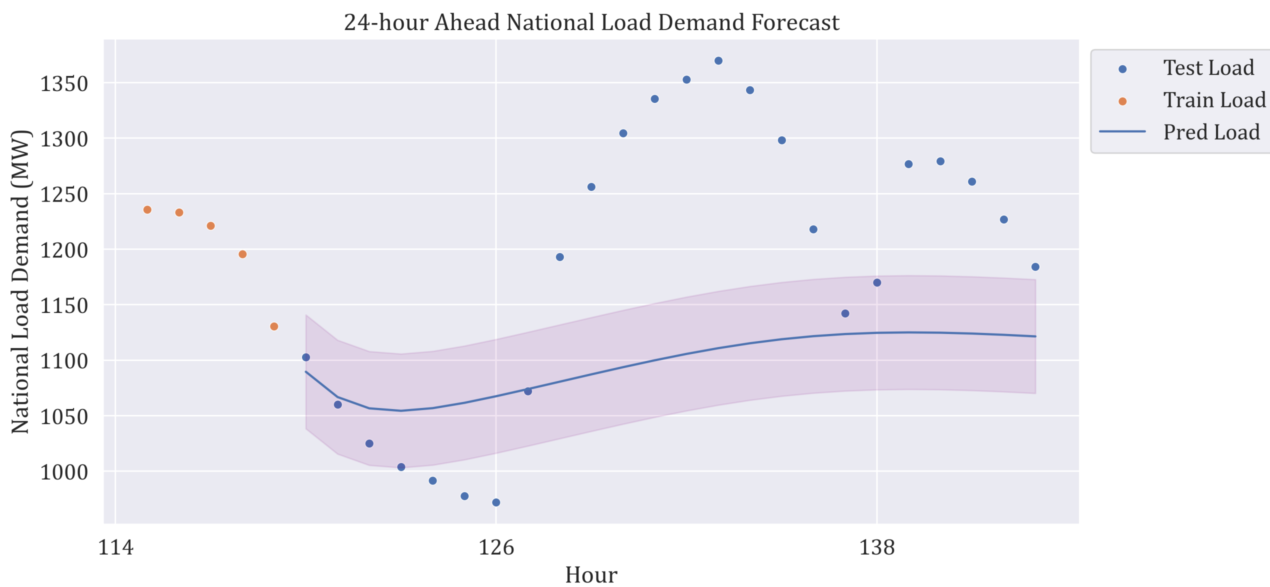

Figure 25: Wind Speed VAR 24 hour forecast for load demand.

F Selected Code - Pandapower

```
net = pn.case14()

def sample_loads(net, n_samples=100):
    base_loads = net.load['p_mw'].values
    sampled_loads = []
    for _ in range(n_samples):
        scale_factors = np.random.uniform(0.8, 1.2, size=base_loads.shape)
        sampled_load = base_loads * scale_factors
        sampled_loads.append(sampled_load)
    return np.array(sampled_loads)

loads = sample_loads(net, n_samples=100)

def run_opf_samples(net, loads):
    results = []
    for load_vec in tqdm(loads, desc="Running OPF"):
        net.load['p_mw'] = load_vec
        try:
            pp.runopp(net)
            vm = net.res_bus.vm_pu.values
            va = net.res_bus.va_degree.values
            gen_p = net.res_gen.p_mw.values
            result_vec = np.concatenate([vm, va, gen_p])
            results.append(result_vec)
        except:
            results.append(None)
    results = [r for r in results if r is not None]
    return np.array(results)

opf_results = run_opf_samples(net, loads)
```

Figure 26: OPF code

```python
def extract_features_from_pandapower(net):
    pp.runpp(net)

    vm_pu = net.res_bus.vm_pu.values
    va_degree = net.res_bus.va_degree.values
    line_loading = net.res_line.loading_percent.values
    p_from_mw = net.res_line.p_from_mw.values
    q_from_mvar = net.res_line.q_from_mvar.values

    if net.load.shape[0] > 0:
        load_p_mw = net.res_load.p_mw.values
        load_q_mvar = net.res_load.q_mvar.values
    else:
        load_p_mw = np.array([])
        load_q_mvar = np.array([])

    features = np.concatenate([
        vm_pu,
        va_degree,
        line_loading,
        p_from_mw,
        q_from_mvar,
        load_p_mw,
        load_q_mvar
    ])

    return features

def scale_loads(net, scale_factor):
    net.load['p_mw'] = net.load['p_mw_orig'] * scale_factor
    net.load['q_mvar'] = net.load['q_mvar_orig'] * scale_factor

def swing_equation(t, y, Pm, Pe, H, Kd):
    delta, omega = y
    d_delta_dt = omega
    d_omega_dt = (Pm - Pe - Kd * omega) / (2 * H)
    return [d_delta_dt, d_omega_dt]

def run_dynamic_simulation(Pm, Pe, H, Kd, t_span=[0, 5], y0=[0, 0], dt=0.01):
    t_eval = np.arange(t_span[0], t_span[1] + dt, dt)
    sol = solve_ivp(swing_equation, t_span, y0, args=(Pm, Pe, H, Kd), t_eval=t_eval)
    return sol.t, sol.y
```

Figure 27: Parameter Estimation Functions

```python
def main():
    net = pn.case14()

    net.load['p_mw_orig'] = net.load['p_mw'].copy()
    net.load['q_mvar_orig'] = net.load['q_mvar'].copy()

    load_scales = np.linspace(0.8, 1.2, 10)
    H_values = np.linspace(2.5, 4.5, 5)
    Kd_values = np.linspace(0.05, 0.2, 4)
    dataset = []

    for scale in load_scales:
        scale_loads(net, scale)
        pp.runpp(net)

        # electrical power output (Pe) from slack bus active power injection
        Pe_mw = net.res_ext_grid.p_mw.values[0]  # slack bus active power injection (MW)
        base_MVA = 100  # IEEE 14 bus base MVA
        Pe = Pe_mw / base_MVA
        Pm = 0.9 * scale
        features = extract_features_from_pandapower(net)
        for H in H_values:
            for Kd in Kd_values:
                t, y = run_dynamic_simulation(Pm, Pe, H, Kd)

                delta = y[0]
                omega = y[1]
                max_delta = np.max(np.abs(delta))
                final_delta = delta[-1]
                max_omega = np.max(np.abs(omega))

                data_row = np.concatenate([
                    features,
                    [max_delta, final_delta, max_omega],
                    [H, Kd]
                ])

                dataset.append(data_row)

    dataset = np.array(dataset)
    num_features = len(features)
    feat_cols = [f"feat_{i}" for i in range(num_features)]
    dyn_cols = ["max_delta", "final_delta", "max_omega"]
    label_cols = ["H", "K_D"]
    cols = feat_cols + dyn_cols + label_cols
    df = pd.DataFrame(dataset, columns=cols)
    df.to_csv("training_dataset_ieee14bus.csv", index=False)

if __name__ == "__main__":
    main()
```

Figure 28: Parameter Estimation Code

Vita

Emma Foley attended Saint Mary's College and The University of Notre Dame where she obtained dual B.S. degrees in Mathematics and Civil Engineering. She worked in transportation engineering for a consulting firm while completing her M.S. in Data Science from Saint Mary's College. Her passion for mathematics and data science led her to the University of Tennessee, Knoxville to obtain a PhD in Data Science and Engineering.

Emma is currently a Graduate Research Assistant at the University of Tennessee. She works with the Grid Communications and Security Group at the Oak Ridge National Laboratory in Oak Ridge, TN under the advisement of Dr. Peter Fuhr. Her current research focuses on working with industry partners to improve grid resilience through the use of drone technology.