Imputation and forecasting for Multi-Output Gaussian Process in Smart Grid

Jiangjiao Xu and Ke Li[*]
Department of Computer Science
University of Exeter

Abstract

Data imputation and prediction is a key component of intelligent upgrading of power systems. Data obtained from the real world may have varying degrees of missing data. These missing components have a significant impact on the outcome of the prediction model. In addition, the single-objective Gaussian process (SOGP) lacks the ability to establish correlation models between multiple datasets, which can not improve the accuracy of data imputation and forecasting. To handle multi-output imputation and forecasting problems, this paper a novel kernel-based multi-output Gaussian process (MOGP) model to achieve data imputation and prediction simultaneously.

1 Introduction

Statistical and machine learning models have been proposed for solar, wind and load generation forecasting to address the technical challenges posed to the grid by stochastic renewable energy generation output Dreidy et al. [2017]. As reported in Panapakidis et al. [2018], the datasets recorded in practice may contain missing values due to a variety of reasons, which will lead to potential bias in data prediction by traditional training methods. However, most existing machine learning-based approaches in smart grid scenarios are based on complete datasets, only a few studies have considered the impact of flawed datasets. In addition, data from different devices in the same region can significantly correlate for multiple prediction tasks. Therefore, using data correlation to build a multi-objective model that accurately interpolates missing data and then uses the interpolated historical data to improve prediction performance is critical to upgrading next-generation power systems.

Compared with other techniques, such as support vector regression (SVR) Drucker et al. [1996], the Gaussian process (GP) predicts the value of unknown points based on input training data using the measurement of homogeneity between points Rasmussen and Williams [2006]. In the field of smart grid engineering, most Gaussian process (GP) models are utilized to forecast tasks using single-output GP (SOGP) model based on one or more time series datasets. In recent years, the multi-output learning (MOL) framework has been successfully applied to various time series prediction problems, and using MOL to learn all tasks jointly can significantly alleviate problems such as overfitting and model instability due to sparse training data, while improving performance compared to SOGP methods Liu et al. [2020]. In contrast to other MOL techniques by sharing features, MOGP uses one model rather than multiple SOGP models to learn concurrent correlations between and within tasks. MOGP has been shown to be effective in multi-output scenarios such as patient monitoring and radiotherapy Durichen et al. [2015], robotic inverse dynamics Chai et al. [2008], classification of electrocardiogram signals Skolidis and Sanguinetti [2011], compiler performance Bonilla et al. [2007] and environmental sensor networks Osborne et al. [2012].

[*] (e-mail: {j.xu, k.li}@exeter.ac.uk).

2022 NeurIPS Workshop on Gaussian Processes, Spatiotemporal Modeling, and Decision-making Systems.

In a multi-output learning problem of M tasks, a MOGP model can be taken to deal with M tasks simultaneously. Given M tasks of length N

$$f(X) = (f_1(x_1), \cdots, f_1(x_N), \cdots, f_M(x_1), \cdots, f_M(x_N))^\top \tag{1}$$

where $\mathbf{x} = \{x_n | n = 1, \cdots, N\}$ and $\mathbf{f} = \{f_m(\cdot) | m = 1, \cdots, M\}$ are the training inputs and outputs for M tasks, respectively. N is the total number of training data for each task.

The covariance function can be defined with the covariance corresponding to tasks and the covariance corresponding to inputs

$$cov(f_m(\mathbf{x}), f_{m'}(\mathbf{x}')) = k(m, m') \times k(\mathbf{x}, \mathbf{x}') \tag{2}$$

where $cov(f_m(\mathbf{x}), f_{m'}(\mathbf{x}'))$ is the covariance function for two independent tasks. $k(m, m')$ and $k(\mathbf{x}, \mathbf{x}')$ are the tasks relationship and inputs relationship, respectively.

Then the covariance matrix K for $\mathbf{f}$ can be expressed as

$$K = K_m(\mathbf{M}, \mathbf{w}_m) \otimes K_t(\mathbf{X}, \mathbf{w}_t) \tag{3}$$

where $K_m(\cdot)$ and $K_t(\cdot)$ are the covariance matrix between tasks and the covariance matrix between the corresponding inputs, respectively. $\mathbf{M} = \{m | m = 1, \cdots, M\}$ and $\mathbf{X} = \{\mathbf{x}_m | m = 1, \cdots, M\}$. $\otimes$ is the Kronecker product operator. Details are presented in Appendix A.

The performance provided by MOGP models is relied on the time series correlation between tasks. In extreme cases with large proportions of data missing, the MOGP based forecasting model is affected if the correlated tasks are temporally shifted. In this paper, the MOGP model will first use the correlation of multiple datasets from multiple tasks to impute missing data or replace anomalous data, and then use the complete data as training data to for multi-output forecasting tasks. In addition, a reconstruction kernel considering periodicity and smoothness is used to capture periodic effects in energy data.

2 Related Work

An increasing number of engineering applications are beginning to consider the use of state-of-the-art multi-output learning techniques Ben-David and Schuller [2003]. To learn the data correlation, Fiot and Dinuzzo [2018] utilized the kernel-based multi-output learning method to forecast the electricity demand. Gilanifar et al. [2020] proposed a Bayesian spatiotemporal Gaussian process model to forecast load consumption. Zhang et al. [2014] utilized a multi-output Gaussian process method to solve the load forecasting problem. Shireen et al. [2018] proposed an iterative multi-output learning framework for time series PV generation forecasting problems. Unfortunately, for many real-life scenarios, task description features are either unavailable or difficult to define correctly. Multi-output Gaussian processes also have been utilized to be effective in multi-output scenarios Durichen et al. [2015], Skolidis and Sanguinetti [2011]. However, these multi-outputs forecasting papers only analyze the prediction performance and do not consider the case of insufficient data or data anomalies which is not reasonable in the real-world case. In this paper, we propose to use the MOGP model to achieve data imputation and prediction simultaneously.

3 Kernel Reconstruction for MOGP

This section will use a more flexible and powerful model based on the intrinsic correlation model (ICM), namely the linear model of coregionalization (LMC) model, by utilizing a sum of Q separable kernels Zhang [2007]. We assume $f_m(\mathbf{x}) = \sum_{q=1}^{Q} f_m^q(\mathbf{x})$, then the entire covariance function in the LMC can be set as

$$cov(f_m(\mathbf{x}), f_{m'}(\mathbf{x}')) = \sum_{q=1}^{Q} k^q(m, m') \times k^q(\mathbf{x}, \mathbf{x}') \tag{4}$$

Then the covariance matrix K of $\mathbf{f}$ corresponding to the LMC model takes the form as follows:

$$K = \sum_{q=1}^{Q} K_m^q(\mathbf{M}, \mathbf{w}_m) \otimes K_t^q(\mathbf{X}, \mathbf{w}_t) \tag{5}$$

where $q = 1, \cdots, Q$ is the number of groups, which means that there are Q groups in total and they are independent. $\mathbf{w}_m$ and $\mathbf{w}_t$ are the vectors associated with the hyperparameters for $K_m^q(\cdot)$ and $K_t^q(\cdot)$, respectively. Meanwhile, each group share the same convariance function. In other words, each $k^q(\mathbf{x}, \mathbf{x}')$ can be different, it means that $k^q(m, m')$ can model the relationship between tasks under each $k^q(\mathbf{x}, \mathbf{x}')$. The ICM can be considered as a specific case of the LMC, with $Q = 1$. It is more limited than the LMC because the ICM applies the same kernel to construct the auto-covariances and cross-covariances between the multiple outputs Altamirano and Tobar [2022].

3.1 Kernel Reconstruction

As discussed in Rasmussen and Williams [2006], there are many existing covariance functions that can be used. Multiplying kernels is the usual method to merge two or more kernels that is an elementwise multiplication of their corresponding covariance matrices Rasmussen and Williams [2006]. To add the flexibility of our model, we consider to design a mixture of two frequently used covariance functions, Rational Quadratic (RQ) and Periodic (PE) covariance functions, to capture the periodicity and smoothness of energy data to vary over longer distances.

$$K_{RP}(x, x') = K_{RQ} * K_{PE} = \sigma^2 \Big(1 + \frac{(x - x')^2}{2\alpha \ell_{RQ}^2}\Big)^{-\alpha} * \exp\Big(-\frac{2sin^2(\pi|x - x'|/p)}{\ell_{PE}^2}\Big) \tag{6}$$

where σ^2 are the output variances of covariance functions, respectively. ℓ_{RQ} and ℓ_{PE} are the length scales of RQ and PE covariance functions, respectively. α is the hyper-parameter that determine the relative weighting of large scale and small scale variations.

More specifically, this kernel function will allow us to model only locally periodic functions and preserve the characteristics of each task. The shape of the repetitive part of the function can be varied over time and the correlations occurring within the task are automatically learned by fitting a covariance function within the MOGP framework.

4 Results and Discussion

Table 1: Comparison results of MSE values obtained by SOGP and MOGP methods for imputation and forecasting

Approach	Demand	PV	Wind
Imputation (1440*3)			
SOGP	7.147E-1(1.58E-2)	6.975E-1(1.42E-4)	6.557E-1(9.06E-3)
MOGP	4.114E-2(7.51E-3)	5.058E-2(6.78E-5)	1.271E-1(7.65E-3)
Prediction (1440*3)			
SOGP	7.469E-1(9.58E-3)	9.157E-1(1.27E-3)	8.491E-1(9.64E-2)
MOGP	2.132E-3(7.98E-9)	1.324E-1(1.87E-7)	8.759E-2(4.98E-8)

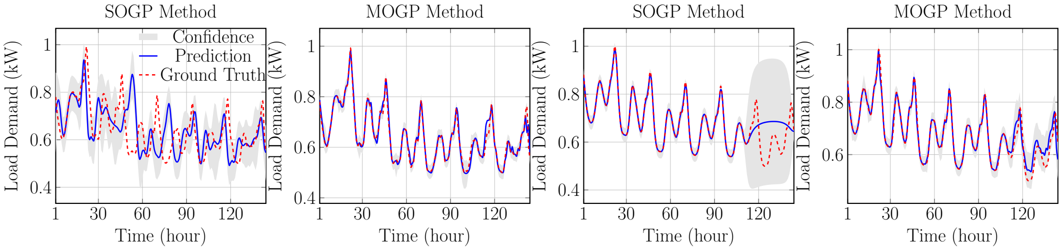

Figure 1: Comparison of the imputation and prediction performance of using the SOGP and MOGP methods for the load demand task.

Our experimental simulation considers the energy datasets for isolated power grid at the English Channel Xu et al. [2021]. We consider three types of sources including the wind generation, the PV generation, and the load demand. The training set and the validation set consist of data collected from three islands including Ushant, Lundy and Isles of Scilly based on the EDF energy company energy company. Details about the performance metric are presented in Appendix B. As introduce in Section 3, by using the LMC as the MOGP model, we directly apply the modified mixtrue kernel to the MOGP method. The historical data of wind generation, PV generation, and load demand for a period of time, sixty days are used to constitute the training datasets while the forecasting performance of different tasks is validated for a period of 24 hours.

The overall comparison results by SOGP and MOGP methods for three different number of datasets are given in Table 1. From these results, we can see that using the MOGP method has a significant impact on the performance of all three types of energy data. To have a better investigation for the performance difference achieved by using the standard SOGP method against the MOGP method, we also show the comparison SMAPE results in Fig. 2. From this result, it is clear to see that the performance has been greatly improved, especially the prediction results for demand. In addition, to have a better visual interpretation of the performance comparison, we plot the imputation and forecasting results for load demand in Fig. 1. From the trajectories shown, we can see that both the imputation and forecasting performance of demand energy of SOGP is poor and there is a better fit to the ground truth with the MOGP model. This is anticipated as the covariance matrix improves performance by reflecting the correlation of multiple data sets from multiple tasks. In contrast, by being equipped with our proposed MOGP method, the imputation and forecasting performance have been improved against the SOGP method and other methods, MTBSGP Gilanifar et al. [2020] and MTLGPTS Shireen et al. [2018], in Table 2.

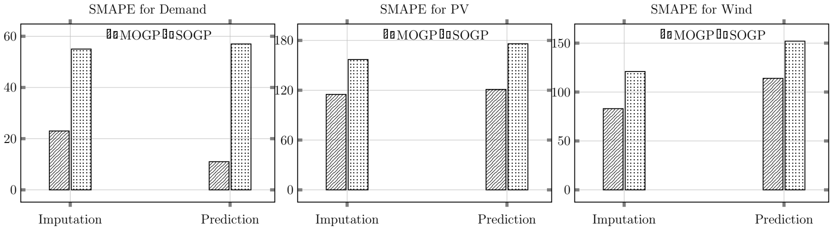

Figure 2: SMAPE value for three datasets.

Table 2: Comparison results obtained by different methods

Type	MTLGPTS	MTBSGP	MOGP
Imputation MSE Values			
Demand	1.062E-1(1.38E-3)	1.833E-1(9.20E-5)	**3.984E-2(2.61E-4)**
PV	4.316E-1(2.57E-2)	3.435E-1(2.20E-3)	**5.817E-2(3.35E-4)**
Wind	2.715E-1(7.43E-4)	2.297E-1(1.41E-3)	**1.436E-1(6.33E-4)**
Imputation SMAPE Values			
Demand	4.587E+1(7.24E-0)	4.746E+1(1.17E+1)	**3.621E+1(3.41E-0)**
PV	4.985E+1(14.58E+1)	5.243E+1(8.25E-0)	**4.038E+1(2.95E-0)**
Wind	5.357E+1(2.25E+1)	6.125E+1(8.65E-0)	**4.259E+1(5.52E-0)**
Prediction MSE Values			
Demand	9.772E-2(9.56E-3)	9.569E-2(4.24E-4)	**2.035E-3(3.42E-5)**
PV	6.263E-1(5.47E-3)	5.346E-1(5.84E-4)	**1.372E-1(6.54E-4)**
Wind	3.552E-1(9.23E-3)	3.254E-1(6.21E-4)	**8.534E-2(4.02E-5)**
Prediction SMAPE Values			
Demand	5.447E+1(8.68E-0)	5.236E+1(1.85E+1)	**3.423E+1(1.54E-0)**
PV	6.056E+1(1.96E+1)	5.971E+1(1.78E+1)	**4.098E+1(1.42E+1)**
Wind	5.127E+1(2.47E+1)	5.972E+1(1.16E+1)	**4.018E+1(3.57E-0)**

5 Conclusions

We utilize the LMC-based MOGP approach, which exploits the MOGP to establish correlations between similar datasets while improving the imputation and prediction performance of multiple imputation and prediction problems in smart grid scenarios. The simplified SOGP model supports data imputation and prediction technologies in the absence of data. However, the MOGP model can further improve performance by using the correlation of data. The MOGP model proposed in this paper can effectively achieve the multi-output results for designing smart grids, and mitigating climate change.

6 Acknowledgment

This work was supported by UKRI Future Leaders Fellowship (MR/S017062/1), NSFC (62076056), EPSRC (2404317), Royal Society (IES/R2/212077) and Amazon Research Award.

References

Matías Altamirano and Felipe Tobar. Nonstationary multi-output gaussian processes via harmonizable spectral mixtures. In *International Conference on Artificial Intelligence and Statistics*, pages 3204–3218. PMLR, 2022.

Shai Ben-David and Reba Schuller. Exploiting task relatedness for multiple task learning. In *Learning theory and kernel machines*, pages 567–580. Springer, 2003.

Edwin V. Bonilla, Kian Ming Adam Chai, and Christopher K. I. Williams. Multi-task gaussian process prediction. In *NIPS'07: Proc. of the 2007 Neural Information Processing Systems Conference*, pages 153–160. Curran Associates, Inc., 2007.

Kian Ming Adam Chai, Christopher K. I. Williams, Stefan Klanke, and Sethu Vijayakumar. Multitask Gaussian process learning of robot inverse dynamics. In *NIPS'08: Proc. of the 2008 Neural Information Processing Systems Conference*, pages 265–272, 2008.

Eric Chuu, Debdeep Pati, and Anirban Bhattacharya. A hybrid approximation to the marginal likelihood. In *AISTATS'21: Proc. of the 20210 International Conference on Artificial Intelligence and Statistics*, volume 130, pages 3214–3222, Apr. 13–15 2021.

Mohammad Dreidy, H Mokhlis, and Saad Mekhilef. Inertia response and frequency control techniques for renewable energy sources: A review. *Renewable and sustainable energy reviews*, 69: 144–155, 2017.

Harris Drucker, Christopher J Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. Support vector regression machines. *Advances in neural information processing systems*, 9, 1996.

R Durichen, MA Pimentel, L Clifton, A Schweikard, and DA Clifton. Multitask Gaussian processes for multivariate physiological time-series analysis. *IEEE Transactions on Biomedical Engineering*, 62(1):314–322, 2015.

EDF energy company. EDF's open data platform on the Ponant Islands. URL `https://opendata-iles-ponant.edf.fr/pages/home/`.

Jean-Baptiste Fiot and Francesco Dinuzzo. Electricity demand forecasting by multi-task learning. *IEEE Transactions on Smart Grid*, 9(2):544–551, 2018. doi: 10.1109/TSG.2016.2555788.

Mostafa Gilanifar, Hui Wang, Lalitha Madhavi Konila Sriram, Eren Erman Ozguven, and Reza Arghandeh. Multitask Bayesian spatiotemporal Gaussian processes for short-term load forecasting. *IEEE Trans. Ind. Electron.*, 67(6):5132–5143, 2020.

Haitao Liu, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. When gaussian process meets big data: A review of scalable gps. *IEEE transactions on neural networks and learning systems*, 31(11): 4405–4423, 2020.

Michael A. Osborne, Stephen J. Roberts, Alex Rogers, and Nicholas R. Jennings. Real-time information processing of environmental sensor network data using Bayesian Gaussian processes. *ACM Trans. Sens. Networks*, 9(1):1–32, 2012.

Ioannis P. Panapakidis, Aggelos S. Bouhouras, and Georgios C. Christoforidis. A missing data treatment method for photovoltaic installations. In *2018 IEEE International Energy Conference (ENERGYCON)*, pages 1–6, 2018. doi: 10.1109/ENERGYCON.2018.8398780.

Carl Edward Rasmussen and Christopher K. I. Williams. Gaussian processes for machine learning. 2006.

Tahasin Shireen, Chenhui Shao, Hui Wang, Jingjing Li, Xi Zhang, and Mingyang Li. Iterative multi-task learning for time-series modeling of solar panel PV outputs. *Applied Energy*, 212(C): 654–662, 2018.

Grigorios Skolidis and Guido Sanguinetti. Bayesian multitask classification with Gaussian process priors. *IEEE Trans. Neural Networks*, 22(12):2011–2021, 2011.

Jiangjiao Xu, Hisham Mahmood, Hao Xiao, Enrico Anderlini, and Mohammad Abusara. Electric water heaters management via reinforcement learning with time-delay in isolated microgrids. *IEEE Access*, 9:132569–132579, 2021.

Hao Zhang. Maximum-likelihood estimation for multivariate spatial linear coregionalization models. *Environmetrics*, 18(2):125–139, 2007.

Yulai Zhang, Guiming Luo, and Fuan Pu. Power load forecasting based on multi-task Gaussian process. *IFAC Proceedings Volumes*, 47(3):3651–3656, 2014.

Appendix A

From the function-space view, a GP is defined as a real process $f(\mathbf{x}) \sim \mathcal{GP}\left(m(x), k(x, x')\right)$ specified by its mean function $m(\mathbf{x})$ and a covariance function $k(\mathbf{x}, \mathbf{x}')$:

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}')]' \end{aligned} \tag{7}$$

where $\mathbf{x} = (x_1, \cdots, x^n) \in \mathbb{R}^n$ is an input vector. The convariance function can be treated as a distance of $\mathbf{x}$ and $\mathbf{x}'$.

Given the n training data set $(\mathbf{x}, \mathbf{y})$ and n^* testing data set $(\mathbf{x}^*, \mathbf{y}^*)$, the joint distribution based on the prior can be given by

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}^* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K(\mathbf{x}, \mathbf{x}) & K(\mathbf{x}, \mathbf{x}^*) \\ K(\mathbf{x}^*, \mathbf{x}) & K(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix}\right) \tag{8}$$

where $K(\mathbf{x}, \mathbf{x}^*)$ represents the $n \times n^*$ matrix of the covariances of the evaluation at all training and testing data points, other entries $K(\mathbf{x}, \mathbf{x})$, $K(\mathbf{x}^*, \mathbf{x})$ and $K(\mathbf{x}^*, \mathbf{x}^*)$ are similar.

The conditional distribution $p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}, \mathbf{y})$ over functions based on the training data points can be written as

$$\begin{aligned} p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}, \mathbf{y}) \sim N(&K(\mathbf{x}^*, \mathbf{x})K(\mathbf{x}, \mathbf{x})^{-1}\mathbf{y}), \\ &K(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, \mathbf{x})K(\mathbf{x}, \mathbf{x})^{-1}K(\mathbf{x}, \mathbf{x}^*) \end{aligned} \tag{9}$$

where the first part is the mean function and the second part is the covariance function. In general, different kernel functions have multiple hyperparameters θ that needs to be adjusted. The hyperparameters for the kernel function can be optimized by minimizing the negative log marginal likelihood Chuu et al. [2021].

$$\begin{aligned} -log(p(\mathbf{y}|\mathbf{x}, \theta)) =& \frac{1}{2}\mathbf{y}^T K(\mathbf{x}, \mathbf{x})^{-1}\mathbf{y} + \frac{1}{2}log(|K(\mathbf{x}, \mathbf{x})|) \\ &+ \frac{n}{2}log(2\pi) \end{aligned} \tag{10}$$

The gradient descent algorithm can be applied into equation (10) iteratively to find the best hyperparameters θ^*.

Appendix B

Performance metric: Here are two widely used metrics. The first one, mean squared error (MSE), is used to evaluate the performance of the imputation and forecasting.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2, \tag{11}$$

where N is number of instances in the testing set, y_i and $\hat{y_i}$ are the ground truth and predicted value, respectively.

Another metric called symmetric mean absolute percentage error (SMAPE) is also utilized to evaluate the performance.

$$\text{SMAPE} = \frac{100\%}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y_i}|}{|y_i| + |\hat{y_i}|}, \tag{12}$$