

Short-term prediction of photovoltaic power generation using Gaussian process regression

Yahya Al Lawati*, Jack Kelly+ and Dan Stowell*+

*School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK

+Open Climate Fix, London, UK

Abstract—Photovoltaic (PV) power is affected by weather conditions, making the power generated from the PV systems uncertain. Solving this problem would help improve the reliability and cost effectiveness of the grid, and could help reduce reliance on fossil fuel plants. The present paper focuses on evaluating predictions of the energy generated by PV systems in the United Kingdom Gaussian process regression (GPR). Gaussian process regression is a Bayesian non-parametric model that can provide predictions along with the uncertainty in the predicted value, which can be very useful in applications with a high degree of uncertainty. The model is evaluated for short-term forecasts of 48 hours against three main factors—training period, sky area coverage and kernel model selection—and for very short-term forecasts of four hours against sky area. We also compare very short-term forecasts in terms of cloud coverage within the prediction period and only initial cloud coverage as a predictor.

I. INTRODUCTION

Fossil fuels are the primary source of energy worldwide, accounting for 84% of primary energy use [1]. However, the world seeks alternatives, and renewable energy has gained interest: its consumption share has grown strongly to over 40% (excluding hydroelectricity) of the global growth in primary energy in 2019 compared to 2018, with solar and wind power being the greatest beneficiaries [1]. The chief concern that accompanies solar energy is its uncertainty, which is mainly affected by weather conditions [2]. Being able to predict power generation mere hours ahead would help control the amount that must be generated from fossil fuels, reducing the amount of carbon dioxide (CO₂) emissions produced. There are two main forecasting methods: model-based approaches and data-driven approaches. An example of model-based approaches is numerical weather prediction (NWP) which employs a set of equations to describe the flow of fluids. Model-based approaches can be complicated and computationally costly. Data-driven approaches, on the other hand, do not use any physical model. They are easier to implement and require no prior knowledge about weather forecasts. Forecasting techniques can be deterministic or probabilistic [2] [3]. Probabilistic approaches can predict a range of values with probabilities, and thus are more appropriate for forecasting applications.

In this work we evaluate the accuracy of short-term predictions of the amount of energy generated from photovoltaic (PV) systems using Gaussian process regression (GPR), trained using historical data of power output from PV systems together with cloud coverage obtained from high-resolution visible (HRV) satellite images.

Integrating PV systems into the grid cannot be efficient unless there is reliable information about power forecast [7]. This is because, whilst it is possible to handle small energy drops by generating slightly more than demand, significant power drops must be replaced within seconds. Fossil fuel plants can take hours to spin from start, and storage battery devices are costly and inefficient for large quantities of photovoltaic panels [8], [9]. Short-term energy prediction can help to determine how much spinning reserve to schedule [8]. Spinning reserve is the unused power capacity which can be activated on the authority of the system operator to serve additional demand [10]. Numerical weather prediction models are useful for predicting elements such as temperature and wind speed, especially in the short term. However, predicting solar irradiance can be inaccurate and very short-term predictions might be inapplicable, for several reasons [8], [11]:

- Numerical weather prediction models require at least an hour to execute. This is a problem, especially for cloud forecasting, as it can change rapidly.
- There is no high demand for solar irradiance.
- Solar irradiance is computationally expensive to calculate.

The gap between the necessity and difficulty of predictions which use conventional forecasts opens the door for machine learning. Different solar forecasting models have been developed throughout the years, some of which have achieved great success in solar forecasting. Artificial neural networks [12], [13] and support vector machines [3], [14] are examples of popular models. Artificial Neural networks are known for their ability to model complex and non-linear relationships. For time-series applications Gaussian processes (GPs) appear to be recommended for their ability to handle complex relationships [10]. Our appendix explains GPs in more detail; we next discuss research conducted on GPs for PV modelling.

Weighted Gaussian Process Regression (WGPR) [2] assigns weights to each data sample, such that high outliers will have lower weights. The forecasting process involves multiple levels, and the model considers eight attributes: solar radiation, photosynthetic active radiation, ambient temperature, wind speed, gust speed, precipitation, wind direction, and humidity. The model has a training data period of approximately 33 days and a validation period of five consecutive days at five-minute intervals. Solar radiation and photosynthetic active radiation (PAR) were the most important of the predictors considered.

Gaussian process quantile regression (GPQR) model is

discussed in [10], [14]. Quantile regression aims to estimate the quantiles of the conditional distribution of predicted values given predictor values, which makes the model more robust against outlier predictions. In addition, it captures the relationship between input and output variables. The GP model is trained with data from 20 days to predict the power load for the following seven days. The results show that GPQR performed better than GPR by 0.26% on average (using mean absolute percentage error, MAPE) for one-hour predictions and 0.49% on average for two-hour predictions.

Other papers have discussed models based on GPs [16]–[18]. Both the models we discussed show better results when compared to GPR in their own comparisons. However, the accuracy difference is not large, and the added sophistication may imply heavier computational demands.

According to [14], clouds are one of the factors that have a significant influence on solar radiation. It is essential to note that, other than the issues mentioned regarding NWP, cloud coverage data are parameterised, whilst depending on satellite images can provide more accurate readings [8]. Parameterisation means that the information is converted to an abstract value that presents general knowledge. For example, instead of resolving clouds, general information is provided, such as whether the day was sunny or cloudy.

The question raised in this paper is: “Can a simple Gaussian Process Regression model accuracy enhance by depending on satellite images instead of Numerical Weather Predictions?” This paper’s contributions can be summarised as:

- Exploring different factors that can enhance short-term forecasts (48 hours) according to satellite images.
- Exploring different factors that can enhance very short-term forecasts (four hours) according to satellite images at each point of the forecast and initial satellite images at the moment of the forecast.
- Comparing forecasts depending on cloud coverage at each point and the initial cloud coverage for very short-term forecasts (four hours).

II. METHODOLOGY

The model was trained based on historical data and cloud coverage. The experiments were divided into two sets. The first set of experiments examined three factors to obtain the most accurate solar power forecasts for the following 48 hours: training period, sky coverage area and kernel structure. The second set of experiments examined very short-term predictions for the subsequent four hours in terms of sky coverage area. In addition, a comparison was made between forecasts with consideration of cloud coverage within the prediction period and forecasts with only initial cloud coverage provided. A third set of experiments was conducted to test the influence of the periodic kernel on model accuracy. However, the model failed to follow the trend of the actual power generated. Hence, this set of experiments is not discussed in the paper.

The results of the first and second sets evaluated amongst four different PV systems in the United Kingdom. The paper assumes that the cloud forecast data will be provided either by

cloud forecasting models or by prediction algorithms in case applied as a real-world application. This section describes two aspects: data preparation and model accuracy evaluation.

A. Data Preparation

The model uses historical data from given PV systems and cloud coverage to predict the amount of energy in the following four and 48 hours. It uses three data sources, including historical data on power generated from different systems and the metadata of those systems, such as location, system capacity and other system details. The third source is satellite images in five-minute intervals. The first step was to identify the geospatial boundary in transverse Mercator projection (metres) of the United Kingdom. The next step in the data preparation was to load PV system metadata and convert latitude and longitude to transverse Mercator projection. Then, systems outside the United Kingdom were identified and removed from the data. After that, data on energy generated from systems over time were loaded, and the period was specified. Data was then aligned with the metadata, in addition to removing systems with missing metadata and corrupt systems that generated power overnight. Next, the satellite data were loaded. The time series rearranged from date and time to integer sequence of numbers where the distance between two numbers equals to five minutes and mapped with the data. Cloud coverage was measured through average visible HRV level in the area above the PV system. Both cloud coverage and time series were merged with the power generated by the PV system to form the training and testing set for the model.

B. Model Accuracy Evaluation

This subsection explores different factors that affect model accuracy. There are several well-known methods for evaluating the performance of a prediction model. Mean square error, root mean square error (RMSE) and MAPE are examples of error-based measurement. In this paper, MAE is used to evaluate the accuracy of the model. It is defined as follows:

$$MAE(Y, Y^*) = \frac{1}{N} \sum_{i=1}^N |y_i - y^*_i| \quad (1)$$

Mean absolute error measures how much error expected on average from a PV system; in this case, it is measured in watts (W). If the model has a MAE of 100 W trained against a PV system, then the predicted value can differ from the actual value by 100 W on average. The reason why MAE was chosen for this study is that it helps quantify the actual difference in power load generated on average. Four PV systems around the United Kingdom were tested in each trial. The tested PV systems had the numbers 709, 1556, 1627 and 1872, which were randomly selected from a pool of PV systems, and had a capacity of 2,460 W; 3,870 W; 2,820 W and 3,960 W, respectively. The experiments were divided into two sets. The first set of experiments predicted the power generated from PV systems over the next 48 hours by testing different factors, given the cloud coverage at each point. The first

factor to examine from the first set is the training period. The drawback with the Gaussian process is that it does not scale well by increasing the number of observations, in which the training complexity can reach $O(N^3)$. Consequently, selecting the features that will be considered in the model and the number of observations that will be trained are crucial for the training process. In addition, a long training period adds seasonal uncertainty and efficiency reduction over time, which may affect accuracy. Four periods are examined to compare training periods and the accuracy of the model. The periods are one week, two weeks, three weeks and one month. The second factor examined is the average HRV value of the sky area above the PV system. If the sky area considered is too large, then the HRV values will not be relevant and will affect the accuracy of the model. If the coverage area is too small, then the solar irradiance captured by a PV system will also be inaccurate. Each pixel on the map represents 1,000 m on the ground. The tested sky areas were 2 by 2 pixels, 6 by 6 pixels and 12 by 12 pixels. The third factor examined is the kernel. The kernel represents the prior knowledge about the data and how the data are correlated. The kernel has a significant effect on the model's accuracy; therefore, it should be carefully selected.

Before discussing different kernel choices, it is essential to note that the white noise kernel is added to the main kernel in all experiments conducted in this paper. This kernel represents the noise in the data by adding uncertainty to the observed data, and it does not change the prediction. The white noise kernel represented by the following:

$$K_{WN}(x_i, x_j) = \sigma^2 \delta(i, j) \quad (2)$$

where δ is the Kronecker delta, and σ^2 is the variance parameter.

Due to the sun's movement, power generated from PV systems is periodic in nature. Accordingly, the main kernel in the model is the periodic kernel. However, the latter is a wrap kernel and must have a base kernel that inherits from a stationary kernel. In other words, the periodic kernel is applied in a domain of a given stationary kernel. The periodic kernel with squared exponential stationary kernel is given in (3). The squared exponential (SE) kernel (4), rational quadratic (RQ) kernel (5) and Matern kernel (6) are examined in this paper:

$$K_{per-SE}(x_j, x_j; h, w, T) = h^2 \exp \left(-\frac{1}{2w^2} \sin^2 \left(\pi \left| \frac{x_i - x_j}{\lambda} \right| \right) \right) \quad (3)$$

$$K_{SE} = h^2 \exp \left[-\left(\frac{x_i - x_j}{\lambda} \right)^2 \right] \quad (4)$$

$$K_{RQ}(x_i, x_j) = h^2 \left(1 + \frac{(x_i - x_j)^2}{\alpha \lambda^2} \right)^{-\alpha} \quad (5)$$

$$K_M(x_i, x_j) = h^2 \frac{1}{\Gamma(v) 2^{v-1}} \left(2\sqrt{v} \frac{|x_i - x_j|}{\lambda} \right) \mathbb{B}_v \left(2\sqrt{v} \frac{|x_i - x_j|}{\lambda} \right) \quad (6)$$

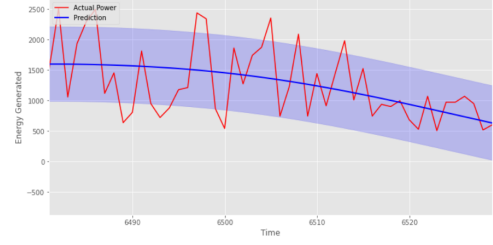


Fig. 1. Energy prediction vs. actual generation for a PV system with initial cloud input on a cloudy day with scattered clouds, over 4 hours. MAE: 327.41.

where h is amplitude, λ is the input scale, T is the period, w is the roughness (similar to role of λ in stationary covariances), $\Gamma()$ is standard Gamma function, $\mathbb{B}()$ is a modified Bessel function of second order, α is known as the index, and v controls the degree of differentiability of the resultant functions [24]. The first two kernels provide a high degree of freedom with relatively few hyperparameters, whilst the third kernel is better for rougher and less smooth variations.

The second set of experiments predicted the energy generated over the next four hours by fixing the training period to three weeks and the periodic kernel to Matern12 as a base kernel. The experiments compared the accuracy of the predictions against a sky coverage of 6 by 6 pixels and 12 by 12 pixels with and without cloud coverage input during the specified period. When cloud coverage was not considered, the last observed cloud coverage was used to predict the following four hours.

III. RESULTS

Results for the first set of experiments (Table I) demonstrate that MAE was lowest with one week of training and the highest with one month of training. However, one week of training may not be enough to cover different weather situations; therefore, three weeks of training was considered as a period in the next set of tests, as it has the second lowest MAE. Moreover, it is worth mentioning that there was a significant difference in training time between each training period, since many GP tasks have complexity of $O(N^3)$.

On sky coverage, the model performed better with 12 by 12 pixel coverage for a 48-hour forecast, with average MAE of 162.32. A kernel selection test found the Matern12 kernel to perform best on average. The conducted experiments showed that a three-week training period, a 12 by 12 pixel sky coverage and a periodic kernel with Matern12 base kernel yielded the best results for 48-hour predictions. As the model made use of cloud cover data, the results may be affected by the accuracy of cloud coverage prediction. Figure 3 shows MAE for the recommended settings per tested days.

We also tested the difference in accuracy with and without cloud cover information, with a much shorter forecasting period of four hours (Table II, Figure 3). We analysed four selected days, each day featuring different weather conditions. The lowest MAE was in a clear sky and highest with scattered clouds. The model was able to follow the general trend in all

TABLE I
SET ONE EXPERIMENTS RESULT

Training Period	Sky Coverage	Kernel Structure	System 709	System 1556	System 1627	System 1872	Average (MAE)
1 week	2X2 pixels	Matern12	112.07	233.14	199.46	143.73	172.1
2 weeks			161.9	418.8	162.1	142.15	221.24
3 weeks			231.54	237.17	161.1	161.08	197.72
1 month			139.38	423.79	177	288.52	257.17
3 weeks	2X2	Matern12	231.54	237.17	161.1	161.08	197.7225
	6X6		105.94	322.23	162.5	154.79	186.365
	12X12		109.59	227.2	165.89	146.6	162.32
3 weeks	2X2	Squared	113.705	236.67	177.54	152.13	170.01125
		Exponential					
		Rational	124.1	223.95	709.615	147.52	301.29625
		Quadratic					
		Matern12	109.59	227.2	165.89	146.6	162.32

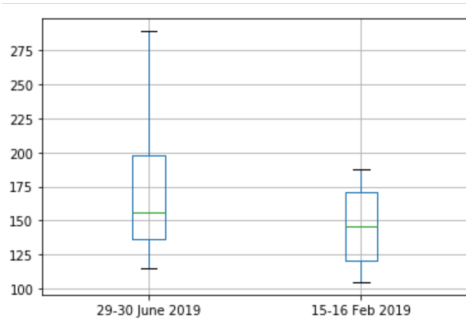


Fig. 2. MAE box plot for 48-hours forecasts presented by testing periods

TABLE II
SET TWO EXPERIMENTS RESULT

System Number	With Cloud Coverage		Without Cloud Coverage	
	6X6	12X12	6X6	12X12
709	145.45	285.99	322.98	327.67
1556	457.81	842.47	527.75	837.665
1627	180.74	192.12	196.57	202.56
1872	240.52	224.62	317.97	320.91
Average	256.04	386.3	341.31	422.2

cases, but cloud cover improved predictions. Note that for PV system 1556, errors were generally larger than for others.

IV. CONCLUSION

This paper evaluated three different factors which affect the GPR model when predicting power generated by PV systems: the training period, sky coverage area and kernel. The main advantage of a GP model is that not only can it predict the amount of power generated, but it also captures the uncertainty of the predicted value at each point. The analysis shows that the Matern12 kernel is the best one amongst the tested kernels, as it was more flexible and could capture the uncertainty presented in the generated power. Furthermore, a training period of three weeks and a sky coverage area of 12 by 12 pixels were the best choices for 48-hour forecasts.

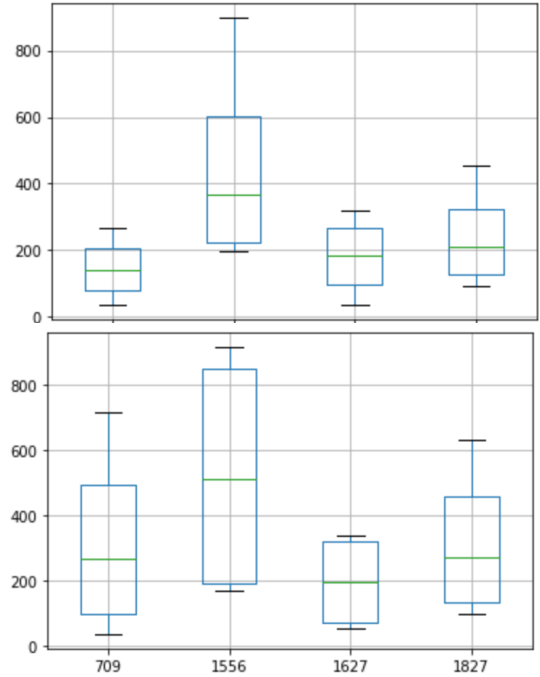


Fig. 3. MAE box plot with outliers for four-hour forecasts. Upper: with consideration of cloud coverage presented to the PV system. Lower: without.

In comparison, a sky coverage area of 6 by 6 pixels was preferable for four-hour forecasts.

The results show that the model was able to accurately predict power generated by PV systems when cloud coverage was stable. However, the model suffered from tracking very rapid changes in cloud coverage when the clouds were scattered. The results also confirmed that the main factor for short-term predictions of solar power is solar irradiance and that cloud coverage has a significant effect on power production.

ACKNOWLEDGMENTS

We thank Prof John Moriarty for advice on the project.

REFERENCES

- [1] BP (2020) BP Statistical Review of World Energy 2020. Available at: <https://www.bp.com/content/dam/bp/business-sites/en/global/corporate/pdfs/energy-economics/statistical-review/bp-stats-review-2020-full-report.pdf> (Accessed: 28 July 2020).
- [2] Sheng, H. et al. (2018) 'Short-term solar power forecasting based on weighted Gaussian process regression', *IEEE Transactions on Industrial Electronics*, 65(1), pp. 300-308. doi: 10.1109/TIE.2017.2714127.
- [3] Liu, J. (2015) 'A novel dynamic-weighted probabilistic support vector regression-based ensemble for prognostics of time series data', *IEEE Transactions on Reliability*, 64(4), pp. 1203-1213. doi: 10.1109/TR.2015.2427156.
- [4] National Weather Service (2020) Numerical Weather Prediction (Weather Models). Available at: <https://www.weather.gov/media/ajk/brochures/NumericalWeatherPrediction.pdf> (Accessed: 3 August 2020).
- [5] Acquah, M.A., Kodaira, D. and Han, S. (2018) 'Real-time demand side management algorithm using stochastic optimization', *Energies*, 11(5), p. 1166. doi: 10.3390/en11051166
- [6] De G. Matthews et al. (2017) 'GPyflow: A Gaussian process library using TensorFlow', *Journal of Machine Learning Research*, 18(1), pp. 1299-1304. doi: 10.5555/3122009.3122049.
- [7] Alessandrini, S. et al. (2015) 'An analog ensemble for short-term probabilistic solar power forecast', *Applied Energy*, 157, pp. 95-110. doi:10.1016/j.apenergy.2015.08.011.
- [8] Kelly, J. (2019) 'Starting work on solar electricity nowcasting', *Open Climate Fix*, Available at: <https://openclimatefix.org/blog/2019-0701-starting-solar-electricity-nowcasting> (Accessed: 1 August 2020).
- [9] Yona, A. et al. (2008), 'Application of neural network to 24-hour-ahead generating power forecasting for PV system', *IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century*, pp. 1-6. doi: 10.1109/PES.2008.4596295.
- [10] Rebours, Y. and Kirschen, D. (2005) 'What is spinning reserve?', *ResearchGate*, 19 September. Available at: https://www.researchgate.net/publication/228364081_What_is_spinning_reserve (Accessed: 1 August 2020).
- [11] Yang, Y. et al. (2018) 'Power load probability density forecasting using Gaussian process quantile regression', *Applied Energy*, 213, pp. 499-509. doi:10.1016/j.apenergy.2017.11.035.
- [12] Yang, Y., Li, S., Li, W. and Qu, M., et al. (2018). 'Power load probability density forecasting using Gaussian process quantile regression'. *Applied Energy*, 213, pp. 499-509. doi:10.1016/j.apenergy.2017.11.035.
- [13] Ding, N. et al. (2015) 'Neural network-based model design for short-term load forecast in distribution systems', *IEEE Transactions on Power Systems*, 31(1), pp. 72-81. doi:10.1109/TPWRS.2015.2390132.
- [14] Shi, J. et al. (2012) 'Forecasting power output of photovoltaic systems based on weather classification and support vector machines', *IEEE Transactions on Industry Applications*, 48(3), pp. 1064-1069. doi: 10.1109/TIA.2012.2190816.
- [15] Chiang, T.C. and Li, J. (2012) 'Stock returns and risk: Evidence from quantile', *Journal of Risk and Financial Management*, 5(1), pp. 20-58. doi:10.3390/jrfm5010020.
- [16] Dahl, A. and Bonilla, E.V., (2019) 'Grouped Gaussian processes for solar power prediction', *Machine Learning*, 108, pp. 1287-1306. doi:10.1007/s10994-019-05808-z.
- [17] Semero, Y.K., Zhang, J. and Zheng, D. (2018) 'PV power forecasting using an integrated GA-PSO-ANFIS approach and Gaussian process regression based feature selection strategy', *CSEE Journal of Power and Energy Systems*, 4(2), pp. 210-218. doi: 10.17775/CSEE-JPES.2016.01920.
- [18] Dahl, A. and Bonilla, E. (2017) 'Scalable Gaussian process models for solar power forecasting', in Woon W., Aung Z., Kramer O. and Madnick S. (eds) *Data Analytics for Renewable Energy Integration: Informing the Generation and Distribution of Renewable Energy*. Springer, pp. 94-106. doi:10.1007/978-3-319-71643-5_9.
- [19] Dahl, A. and Bonilla, E., 2017, September. Scalable Gaussian process models for solar power forecasting. In *International Workshop on Data Analytics for Renewable Energy Integration* (pp. 94-106). Springer, Cham.
- [20] Bacher, P., Madsen, H. and Nielsen, H.A. (2009) 'Online short-term solar power forecasting', *Solar Energy*, 83(10), pp. 1772-1783. doi:10.1016/j.solener.2009.05.016.
- [21] The Gradient (2020) Gaussian processes, not quite for dummies. Available at: <https://thegradient.pub/gaussian-process-not-quite-for-dummies/> (Accessed 1 August 2020).
- [22] Rasmussen, C.E. and Williams, C.K.I. (2006) *Gaussian processes for Machine Learning*. Cambridge: MIT Press.
- [23] Scikit-learn.org (2020) Gaussian processes regression: Basic introductory example. Available at: https://scikit-learn.org/stable/auto_examples/gaussian_process/plot_gpr_noisy_targets.html (Accessed: 1 August 2020).
- [24] Roberts, S. et al. (2013) 'Gaussian processes for time-series modelling', *Philosophical Transactions of the Royal Society*, 371, p. 1-25. doi: 10.1098/rsta.2011.0550.

APPENDIX

Gaussian process (GP) is a probability distribution over possible functions that fit a set of points. Having a joint distribution of these random variables generates a multivariate Gaussian distribution. A univariate Gaussian distribution can be described by mean and variance.

$$\mathcal{N} \sim (\mu, \sigma^2) \quad (7)$$

where \mathcal{N} represents Gaussian distribution with mean μ and variance σ^2 .

The joint univariate Gaussian distribution is described instead by mean and a covariance matrix.

$$\mathcal{N} \sim (\mu, \Sigma) \quad (8)$$

where \mathcal{N} represents Gaussian distribution with mean μ and covariance Σ .

The covariance matrix will contains the variance of each random variable, which expressed diagonally in the matrix, and the covariance between the random variables. The covariance between variables represents how two variables are correlated. Assuming that we have two-variable Gaussian distribution with variables y_1, y_2 having $\mu = 0$ and covariance matrix $\begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}$. Figure 6 shows the possible values that y_2 can have given y_1 , described by the red Gaussian distribution as y_1 and y_2 are correlated.

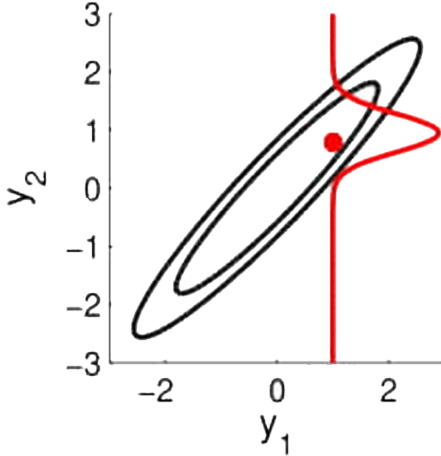


Fig. 4. Possible values for y_2 given y_1

Figure 7 shows an example of a ten-dimensional posterior distribution, with observations on locations 2, 6, and 8. The black line represents the mean and the grey area represents the standard deviation.

Thus far, what has been presented is a discrete set of predictions and observations points. However, in order to model observations with real values, we require arbitrarily large sets and must calculate its covariance or otherwise have the covariance generated by the covariance kernel function, which will provide the covariance element between any arbitrary

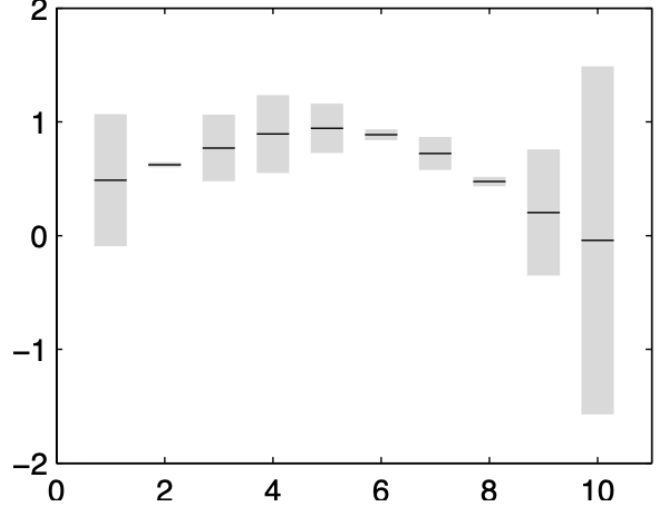


Fig. 5. 10 dimensional posterior distribution with observations on locations 2, 6, and 8 [24]

samples. Choosing the covariance kernel function insufficient on its own. Other properties, such as the right length scale that represents the data more accurately, can still be vague. Such properties are called hyperparameters, and they define the shape of the covariance kernel function. Finding the best values for these hyperparameters that represent the given data is called 'training the model'.

For a set of random variables, such as $X = \{x_1, x_2, \dots, x_n\}$, then we can define the covariance matrix can be defined as

$$\mathbf{K}(X, X) = \begin{pmatrix} k(x_1, x_1) & k(x_1, x_2) & \dots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & \dots & k(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & \dots & k(x_n, x_n) \end{pmatrix} \quad (9)$$

The entire function evaluation associated with points in X is a draw from Multivariate Gaussian distribution.

$$p(Y(X)) = \mathcal{N}(\mu(X), \mathbf{K}(X, X)) \quad (10)$$

where $Y = \{y_1, y_2, \dots, y_n\}$ are dependent function values, evaluated with random variables $X = \{x_1, x_2, \dots, x_n\}$.

Using the Multivariate Gaussian distribution we can obtain posterior distribution over \mathbf{y}^* for test datum \mathbf{x}^* given by

$$p(\mathbf{y}^*) = \mathcal{N}(\mathbf{m}^*, \mathbf{C}^*) \quad (11)$$

where,

$$\mathbf{m}^* = \mu(X^*) + \mathbf{K}(X^*, X) \mathbf{K}(X, X)^{-1} (Y(X) - \mu(X)), \quad (12)$$

$$\mathbf{C}^* = \mathbf{K}(X^*, X^*) - \mathbf{K}(X^*, X) \mathbf{K}(X, X)^{-1} \mathbf{K}(X, X^*)^T. \quad (13)$$

Figure 8 is an example of Gaussian process plot in which the red points represent observations and the blue line represents the function that best fits the observations. The blue area

represents the uncertainty of the output [6] [21] [22] [24].

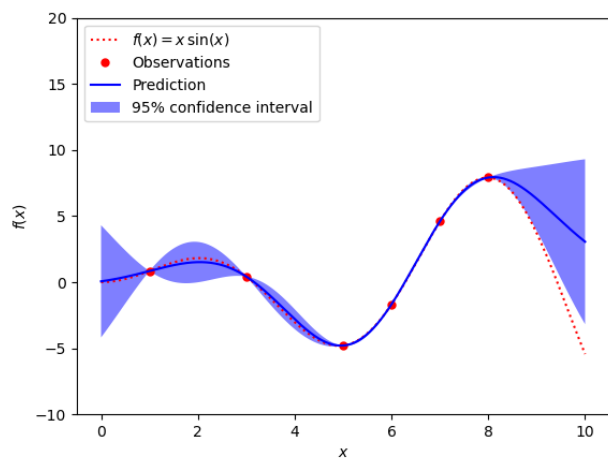


Fig. 6. Example of gaussian Process output [23]