Gaussian Processes

JACK SEYMOUR, Loyola Marymount University, USA
MARIA DOMINGUEZ, Loyola Marymount University, USA
CAMERON SCOLARI, Loyola Marymount University, USA

Gaussian Processes are a type of supervised learning algorithm that excel when modeling data that has complex, non-linear relationships. Additionally, this machine learning model is well-suited for cases where uncertainty in predictions needs to be modeled. This makes Gaussian Processes extremely valuable in areas where understanding prediction confidence is important. The main goals for this report and final project are to learn how to apply Gaussian Processes to data from real-world problems and educate fellow students on the two models of Gaussian Process Regression (GPR) and Gaussian Process Classification (GPC). During implementation, we performed robust hyperparameter tuning to train the best possible models. This paper highlights how Gaussian Processes are an excellent machine learning model choice for situations where uncertainty estimation is essential through their code implementation on two real world datasets and subsequent discussion.

1 Introduction

A Gaussian Process is a random process where any point is assigned a random variable function and the joint distribution of a finite subset of these variables follows a Gaussian distribution [1]. In other words, it is a generalization of the Gaussian probability distribution [2]. Gaussian Processes are powerful regression and classification machine learning models that are also well understood mathematically [3]. Some of the key characteristics of these models are their non-parametric nature, smoothing, interpolation, probabilistic predictions, and marginalization of parameters. The prediction's computational complexity is cubic in the number of datapoints; this can be prohibitive for large sets of data and requires approximate implementations [3].

There are a myriad of advantages that come with using the Gaussian Process algorithm. First of all, the prediction made is probabilistic so the empirical confidence intervals can be calculated to decide whether to refit the prediction in a specific area or region [4]. Gaussian Processes are also versatile as distinct kernels may be specified in addition to the benefit that the prediction interpolates the observations [4]. However, there are also a few disadvantages of using Gaussian Processes including that they lose efficiency in high dimensional spaces and that the implementation is not sparse due to using the entire samples to make predictions [4]. It may also be difficult to understand the exact relationship between the inputs and outputs for Gaussian Processes with complex kernel functions.

"A Gaussian process model describes a probability distribution over possible functions that fit a set of points" [5]. Since a probability distribution is created, we can compute the means to represent the maximum likelihood estimate of the functions, and the variances. Because we have the probability distribution over all possible functions, we can compute the means to represent the maximum likelihood estimate of the function, and the variances as an indicator of prediction confidence. Both classification and regression can be viewed as function approximation problems.

Regression is concerned with the prediction of continuous quantities [2]. GPR is a non-parametric model that does now know the functional relationship between the dependent and independent variables. GPR makes a prior assumption that any smooth function could possibly represent the functional relationship between the data points and regression is performed by defining a distribution over this infinite number of functions. GPR allows for the use of different kernels and alpha values to optimize the model. On top of this, the model optimizes the noise hyperparameter

Authors' Contact Information: Jack Seymour, Loyola Marymount University, Chicago, Illinois, USA; Maria Dominguez, Loyola Marymount University, San Diego, California, USA; Cameron Scolari, Loyola Marymount University, Los Angeles, California, USA.

during training and fits the data points while remaining smooth. GPR not only returns a prediction but also a predicted standard deviation which can be utilized for constructing predicted confidence intervals which can provide a better understanding and point to a broader range of possible predicted values. “GPR models have been widely used in machine learning applications due to their representation flexibility and inherent capability to quantify uncertainty over predictions” [5] and our implementation of this model will highlight the usefulness of quantifying uncertainty.

Classification is when we wish to assign an input pattern x to one of C classes [2]. In GPC, test predictions are in the form of class probabilities, thus this model returns the probabilities for class labels. GPC provides uncertainty estimates for predictions, helping identify low-certainty classifications that may need further review. With GPC, a direct classification is not made, allowing for further flexibility without feature engineering. When interpreting the output of GPC, the curves drawn are viewed as “guesses” based on uncertainty and since generalization to test cases inherently involves some level of uncertainty, it seems natural to attempt to make predictions in a way that reflects these uncertainties [2].

Kernels, or covariance functions, also play a significant role in the predictions for these machine learning models and are used in both GPR and GPC as they “encode our assumptions about the function we wish to learn” [2]. There are an infinite number of kernels to choose from when fitting a model. Kernels define an “infinite” number of previous functions by describing the covariance between any pair of data points in the input space. This allows for a continuous function to be modeled with a high degree of flexibility across the entire domain. The kernel choice greatly influences the model’s performance and how smooth the predictions are and the hyperparameters of kernels are optimized during the initial training.

Two kernels that are commonly used in Gaussian Process are RBF and Matern. With the RBF a distinguishing characteristic of the kernel is that the function is infinitely smooth. The functions it models are infinitely differentiable. This property makes it particularly well suited to data with smooth, continuous patterns. In order to account for noise control, the RBF kernel can manage noisy data well through the length-scale parameter l. This parameter differentiates with the “domain smoothly decaying as the distance between them increases” [6]. With the assumption of infinite smoothness, RBF may lead to overfitting noisy datasets. This can happen when the model tries to interpolate noisy fluctuations instead of the underlying trend. This makes RBF not appropriate for very noisy data as it places too much reliability on continuity.

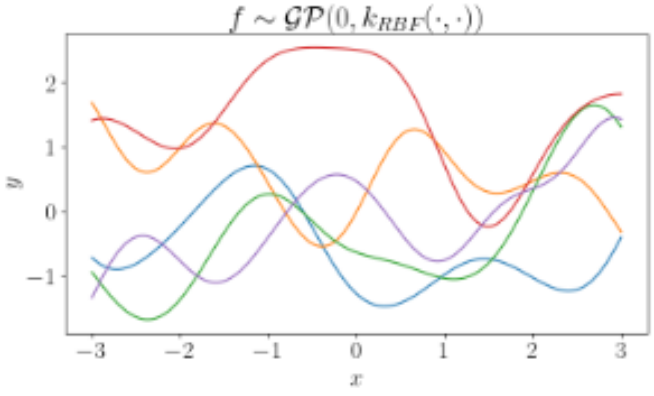

Fig. 1. Gaussian Process Function from RBF.

The Matern kernel differs from the RBF kernel primarily with the introduction of the additional parameter v, which controls the degree of smoothness in the model. This parameter allows Matern greater flexibility to handle a wider variety of datasets. With a lower value of v, the function conforms to rougher functions. So, a lower value for v is better for datasets with sharp transitions. A higher value of v results in smoother functions almost allowing the kernel to behave more like the RBF kernel with minor adjustments associated with the behavior of the Matern kernel. Tuning v to be congruent to a dataset involves choosing a balance between smoothness and adaptability to noisy data.

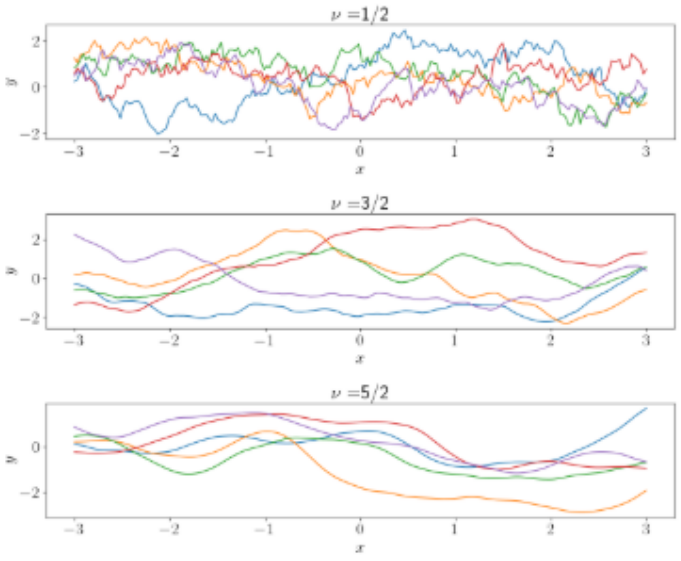

Fig. 2. Gaussian Process Function from Matern differences from variable *v*.

Throughout this course we have learned about several different supervised and unsupervised learning algorithms such as decision trees, k-nearest neighbors, linear and logistic regression, clustering, neural networks, etc. For this final project, we wanted to learn more about a supervised learning model that we had not covered in this class and were intrigued by the topic of Gaussian Processes. We were motivated by the ability of Gaussian Processes to not only make predictions but also provide estimates that are crucial for probabilistic reasoning in machine learning tasks. Through this project, we aim to teach the class about both GPR and GPC. Teaching our fellow students about Gaussian Processes contributes to understanding advanced machine learning concepts by introducing probabilistic modeling, connecting theory and application, and highlighting non-parametric learning. The objective for this report is to help students understand what Gaussian Processes are by demonstrating how this algorithm can be applied to two real world issues.

2 Methods

We applied GPR to a dataset from Kaggle containing observations of cars and their carbon dioxide emissions. The dataset included features like make, model, vehicle class, engine size, and fuel consumption with the target variable being carbon dioxide emissions. The dataset contained over 200,000 samples but due to this large sample size, the code ran slower, so we limited it to only 10,000 observations and still chose to randomly sample 2,000 of those because GPR works better with smaller sample sizes. As the sample size increases, the computational cost increases because the

"computational complexity is O(N3), where N represents the dimension of the covariance matrix" [5]. We first split the data into training and testing data with 1,400 observations belonging to the former and 600 belonging to the latter. We then standardized our data because the feature columns had different scales. The continuous variable columns we used for the model were Engine Size (L), Fuel Consumption City (L/100 km), Fuel Consumption Hwy (L/100 km), Fuel Consumption Comb (L/100 km), Fuel Consumption Comp (mpg), and 'Cylinders'. The categorical features, which were converted to numerical values using one-hot encoding, were Vehicle Class, Transmission, and Fuel Type. To optimize the model, we tested multiple kernels and alpha values. To test these two hyperparameters, we employed a randomized search that randomly tested different inputs, tested each against the training data, and returned the two that gave the best results. To showcase our results, we plotted the actual versus predicted carbon dioxide emissions along with the 95% confidence interval for each data point to highlight regions of the model with high uncertainty. We also referred to an R^2 value to evaluate how well our model was doing. Due to space limitations, we could not visualize all 9 feature variables so Figure 3 helps visualize only two feature variables and our target variable.

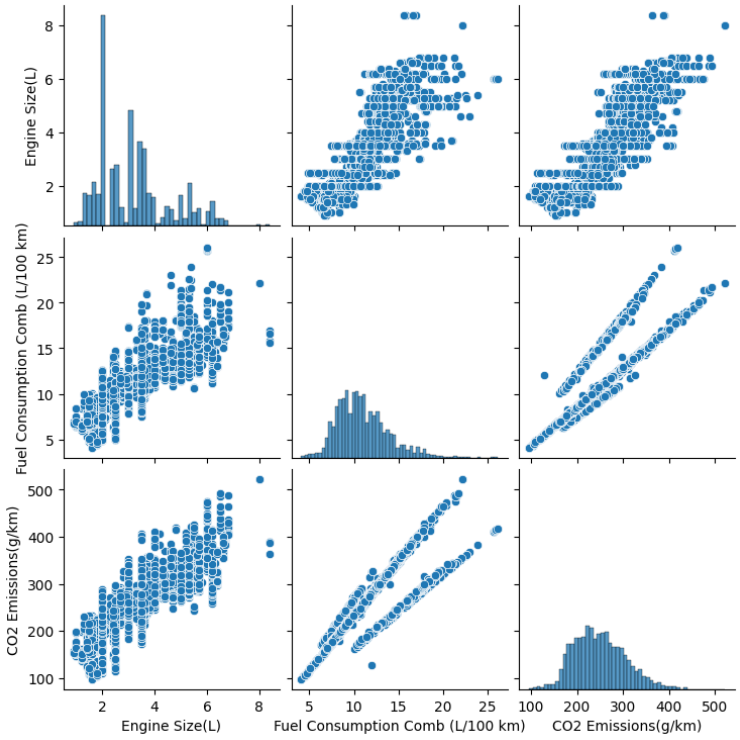

Fig. 3. Pair Plot of Engine Size(L), Fuel Consumption Comb (L/100 km), and CO2 Emissions(g/km).

For applying GPC, we chose a credit card fraud detection dataset including a plethora of credit card transactions made by European cardholders in the year 2023. This dataset includes a unique identifier for each transaction, anonymized features representing different transaction attributes, the amount of the transaction, and a binary class label of whether or not the transaction was fraudulent. The main challenge when incorporating GPC was that the fraud data was massive. The Kaggle data set we selected contained over 550,000 total transactions, and after numerous failed attempts of trying

to push this dataset to GitHub, we decided to just limit it to only 10,000 observations and still chose to randomly sample 2,000 of those because GPC works better with smaller sample sizes. Of those 2,000 samples, 1001 were samples of credit card fraud so roughly half the data were observations with fraud and the other half were observations with no fraud.

In credit card fraud detection, false positives may be an inconvenience for customers, while false negatives can be costly to the credit card users so it is important that our model does a good job at making accurate predictions. We think that it may be advantageous to use recall as our performance statistic because it reduces false negatives meaning we can detect fraudulent behavior on credit cards early and prevent credit card users from losing more money. Figure 4 shows a bar chart to help visualize the distribution of fraudulent vs. non-fraudulent credit card observations in the 10,000 observations from our dataset and since we randomly sampled 2,000 points from this dataset, we can assume our sample resembles this distribution as well.

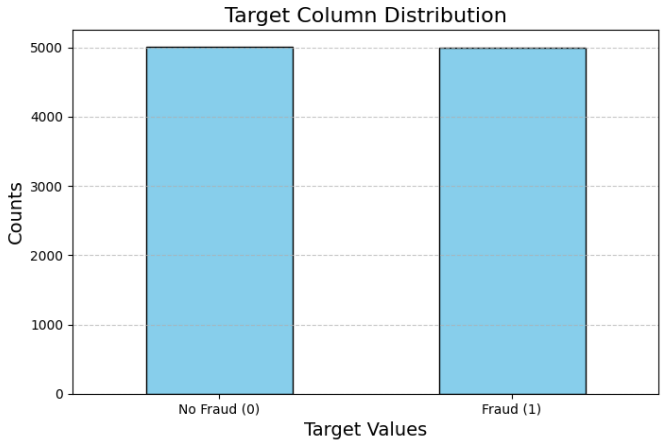

Fig. 4. Bar Chart showing the counts of each class within the dataset.

3 Results

In this section, we will further describe and depict the results for applying GPR to the car observations dataset and GPC in the credit card fraud detection dataset. These results highlight how Gaussian Processes can be applied to real-world problems, especially by showing the strengths and limitations of each GP model as well as how we approached hyperparameters optimization to construct the best models.

Figure 5 shows predicted versus actual carbon dioxide emissions. The confidence intervals were calculated using the predicted standard deviation for each point that is returned as a result of GPR. They create a good visualization for indicating regions of the model with higher uncertainty.

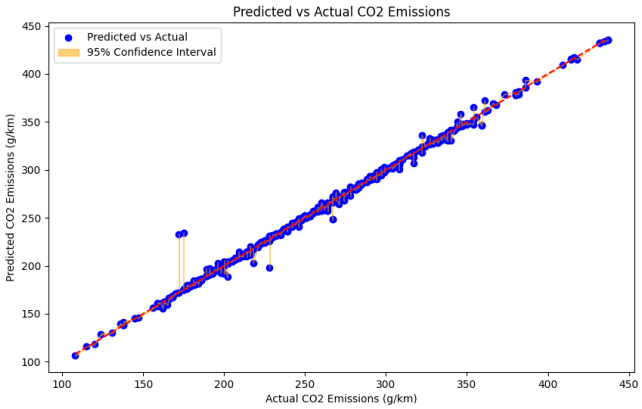

Fig. 5. Predicted vs. actual carbon dioxide emissions.

For GPR, mean-squared error explained on average, how far away our predicted carbon dioxide emissions were from the true carbon dioxide emissions. The mean-squared error for the training data was about 0.066, and the mean-squared error for the test data was about 1.136. These are promising results that suggest that our model is doing a good job at predicting the actual carbon dioxide emissions for each car. On top of that, an R^2 value for both training and testing data was calculated, with each being roughly 99.9%, suggesting that our model does an exceptional job at explaining the variance in the data. Additionally, with the predicted mean and standard deviation values returned for each data point after running GPR, we created 95% confidence intervals for each predicted point and observed that 92% of true values lied within the confidence intervals. These results suggest that the model provides reasonable uncertainty estimates alongside accurate predictions.

As stated previously, the Gaussian Process Classification (GPC) model was applied to a randomly sampled subset of the credit card fraud detection dataset. The model achieved good performance metrics on a test set of 600 transactions.

- `Accuracy`: 97.67
- `Precision`: 0.99 (fraud detection, class 1)
- `Recall`: 0.96 (fraud detection, class 1)
- `F1-Score`: 0.98

Figure 6 shows the confusion matrix, highlighting the classification performance of the GPC.

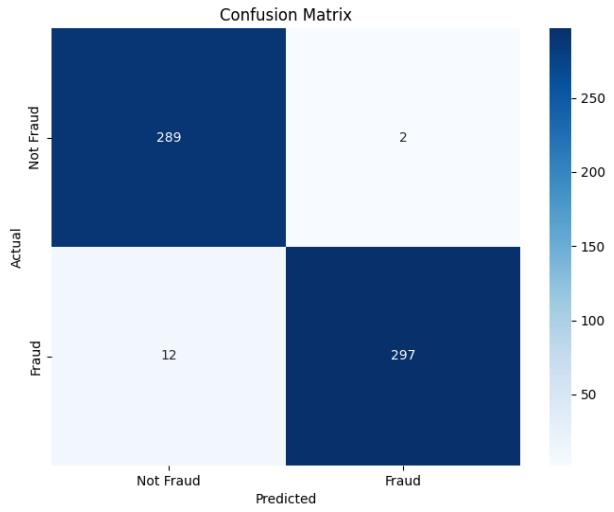

Fig. 6. Confusion matrix for GPC.

Figure 7 shows the probability distribution by class. The model outputs probabilities close to 0 and 1 respectively indicating how the model is fairly confident in distinguishing between the two classes. Despite this, there is a small overlap in the middle region highlighting a region where the model struggles and may be less confident.

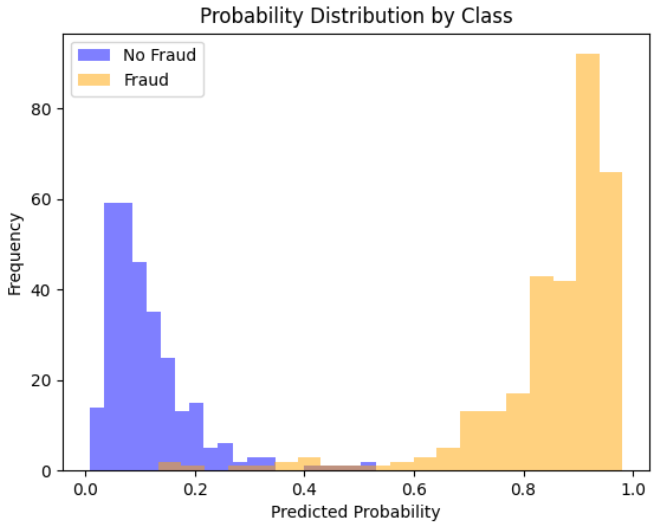

Fig. 7. Probability Distribution by Class.

The model demonstrated a strong fraud detection capability. With a recall of 0.97, the GPC model successfully identified 97% of fraudulent transactions, crucial for preventing financial losses. A Precision of 0.98 reflects that most transactions flagged as fraudulent were indeed fraudulent, reducing customer inconvenience. The F1-score of 0.98 confirms the model's ability to balance precision and recall effectively.

Table 1 provides a nice summary by displaying the training error, test error, training size, and test size for both GPR and GPC. The training and test error for GPR is the mean-squared error which measures the average distance the predicted value was from the actual value. The training and test error for GPC is 1 minus the accuracy score.

Table 1. Display of the training error and test error as well as the training size and test size for both GPR and GPC.

Model	Train Error	Test Error	Training Size	Test Size
GPR	0.066	1.136	1400	600
GPC	0.0014	0.0233	1400	600

4 Ethical Considerations

When using machine learning or any other type of artificial intelligence (AI), there are always certain ethical considerations. Some general ethical principles that should be considered when using AI are that it contributes to society and human well-being, avoids harm, is honest and trustworthy, does not discriminate, respects privacy, and honors confidentiality. It is crucial to be aware of biases in data in order to prevent any unfair outcomes. For example, in the credit card detection dataset, if there were any historical biases relating to specific regions or demographics, then GPC could reinforce these biases and produce unfair results and predictions. Another important thing to keep in mind when utilizing AI is to ensure data encryption and security of data storage, since the information from datasets may be sensitive or private. By addressing these and numerous other ethical considerations, Gaussian Processes can be applied fairly and responsibly to maximize their benefits and decrease any detrimental outcomes.

5 Conclusion

Gaussian Processes are a powerful supervised machine learning model best suited for cases where it is necessary to use uncertainty estimation. Throughout this project, we explored and learned about both Gaussian Process Regression and Gaussian Process Classification. By implementing these algorithms to real-world datasets, we showed how GPR can accurately predict values such as carbon dioxide emissions and how GPC is effectively able to model class probabilities for credit card fraud detection with estimates for uncertainty. We were able to tweak and refine the models so that their performances were reliable as evident by our test error for each model respectively. In this project, we learned that GPR offers reliable predictions that make it ideal to model tasks that need uncertainty quantification. GPC, on the other hand, is effective for classification tasks where understanding prediction confidence is crucial, as seen in the credit card fraud example. The choice of kernels is a significant factor in Gaussian Process performance and the smoothness of the model's predictions. A future study might look deeper into how kernel choice affects results. Finally, we acknowledged the importance of keeping ethical considerations in mind when utilizing AI in order to provide the best results by using data fairly and responsibly.

Acknowledgments

Thank you so much to Dr. Korpusik for her help and guidance throughout this class and for this project. Thank you to Kaggle for providing the initial datasets used in our code. Thank you to the LMU Writing Center, specifically Taylor Crowell, for reviewing both our final report and presentation and providing excellent feedback. Thank you for the references and existing literature that allowed us to strengthen our knowledge on the models.

References

[1] M. Krasser, "Gaussian processes," Mar. 19, 2018. [Online]. Available: https://krasserm.github.io/2018/03/19/gaussian-processes/. [Accessed: Dec. 9, 2024].

[2] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[3] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge, UK: Cambridge University Press, 2012. [Online]. Available: http://web4.cs.ucl.ac.uk/staff/D.Barber/textbook/2006.

[4] scikit-learn developers, "Gaussian processes," *scikit-learn Documentation*. [Online]. Available: https://scikit-learn.org/stable/modules/gaussian_process.html. [Accessed: Dec. 9, 2024].

[5] J. Wang, "An intuitive tutorial to Gaussian process regression," *ArXiv*. [Online]. Available: https://arxiv.org/html/2009.10862v5. [Accessed: Dec. 9, 2024].

[6] A. Jones, "The matérn class of covariance functions," *Andy Jones Technical Blog*. [Online]. Available: https://andrewcharlesjones.github.io/journal/matern-kernels.html. [Accessed: Dec. 9, 2024].