JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Gaussian Process Regression for Climate Modeling: Potentials, Limitations, and Advances in Emulation Techniques

Anmol Chaure
Research Scholar, CSE Department
BIT Durg, India

Ashok Kumar Behera
Associate Professor, CSE Department
BIT Durg, India

Sudip Bhattacharya
Associate Professor, CSE Department
BIT Durg, India

Abstract: **As per the IPCC 6th assessment report, Anthropogenic activities continue to cause increasing and prolonged impacts on the Earth's system. There is a growing need for tools and data driven models to develop adaptation plans, vulnerability assessments and resilience strategies. In various future socioeconomic scenarios fully coupled GCMs, (known as Earth System models) are the most used numerical models for climate change projection generations. These models have been the benchmark of climate modelling since their inception. However due to numerous parameters and an extensive spectrum of spatial and temporal scales, these models entail substantial computational costs which are usually executed as part of international experiments like CMIP (Coupled Model Inter-comparison Project). To circumvent this, one of the obvious solutions is the use of surrogate models of the GCMs. Often termed as emulators, surrogate models are designed to mimic the functionality of a GCM but still being very light in terms of computational as compared to a GCM model. Statistical surrogates have been prevalent for climate modelling since many years. However due to availability of huge volumes of observational, simulation and reanalysis climate data in the recent years, increase in the computational power of modern CPUs/GPUs and growing popularity of data driven modelling techniques, machine learning and deep learning based surrogate models for climate are becoming more prominent. In contrast to statistical surrogates, deep learning surrogates can capture complex nonlinear relationship between data, and can adapt to changing data patterns enabling them to capture even the minute dynamics of climate systems. Although having an advantage on some aspects, deep learning surrogates lack interpretability, can extrapolate values beyond the training points (lack of calibration) and cannot capture model uncertainty to a satisfactory extent. To mitigate these issues, GPR emerges as a viable candidate, as it uses a probabilistic approach to model the uncertainty estimates. GPR based models can be further tuned for better accuracy through the use of composite kernels. With the advancements in machine learning algorithms and increasing number of domains across various engineering streams employing GPR models, It is expected that better composite kernel design techniques will be introduced enabling higher accuracy and resolution in emulation techniques. The present work is a comprehensive summary of the potentials and limitations of GPR, as a surrogate for modelling climatic processes.**

Index Terms **- Earth System Science, Deep learning, Gaussian Process Regression, Climate Modelling**

I. Introduction

The climate of Earth has been predominantly determined by the balance of energy received from the Sun and energy radiated back into space also phrased as Earth's Energy-budget. In the absence of anthropogenic activities, the planet undergoes natural cycles that play a key role in regulating its climate. Anthropogenic Climate Change is closely associated with the usage of fossil fuels, the release of aerosols, and changes in land use through agricultural practices and deforestation, as they are responsible for the emissions of four primary greenhouse gases, that are, carbon dioxide (CO_2), methane (CH_4), nitrous oxide (N_2O) and the halocarbons (a group of gases containing fluorine, chlorine and bromine). [28] Projection of climate change is fundamentally distinct from weather prediction, as forecasting weather is an" initial condition problem", whereby the future state is determined by its present (and past) state, therefore, input accuracy plays a vital role. Even with this information, even a small change in the initial condition can lead to drastically different outcomes due to the phenomenon of" deterministic chaos", which was first demonstrated by Edward Lorenz's three-equation model, on his Royal Mcbee at MIT.

In contrast, climate is a long-term mean of the weather (typically spanning for 3 decades), it is relatively less sensitive to initial conditions. This is because the memory of initial conditions can last for a significant duration primarily in the" inherently slow" components of the Earth system, such as the ocean. Climate projection is a" boundary conditions problem", i.e, it depends on the factors such as the Earth's orbit and the levels of greenhouse gases and aerosols in the atmosphere. It is also heavily dependent on the current ocean state for accurate prediction.

For extended periods of time, climate prediction is increasingly hinged in the trajectory of key forcings like CO_2 emissions, which is mostly governed by anthropogenic activities. In alignment to the goals of the IPCC (Intergovernmental Panel on Climate Change) Paris agreement, which aims to restrict the global average temperature rise to" well below 2°C above pre-industrial levels and

pursue efforts to limit the increase to 1.5°C", there are numerous emission trajectories which cannot all be evaluated because Earth system models are large and have hundreds of tunable parameters, ergo, it takes a lot of time as well as space to run. [24]
To mitigate this the development of Emulators/Surrogates are becoming common, which can be used to replace full scale Earth system models. These emulators can help in the evaluation of various shared socioeconomic pathways (SSPs), which are important for policymakers to assess these trajectories so that they can evaluate the economic and societal and come up with different adaptation strategies.[42]
Emulators, therefore, plays a crucial role in climate modelling and projection, due to their ability to provide computational efficiency enable uncertainty quantification, facilitate parameter analysis, model calibration, and aid in intercomparison. They provide a faster and more systematic way to approximate the behavior of complex climate simulators and allows for cost effective sensitivity analysis.[40]

II. IMPORTANCE OF CLIMATE CHANGE

Climate refers to the typical atmospheric conditions of a location for extended period, usually spanning several years or decades. The effects of climate change are widespread, affecting both human and natural systems. They can be observed in various sectors such as agriculture, ecosystems, energy, and health, and can also worsen existing issues like poverty and political instability. By examining these impacts and our susceptibility to them, we can enhance our understanding of the importance of adaptation and resilience. The general circulation of atmosphere, which includes the semi-tropical trade winds, cyclonic storms accountable for the trade of moisture and energy, etc, are responsible for the regulation of climate.
It is undeniable that climate change is one of the defining challenges of the 21st century, we can no longer rely on the records of past weather, therefore, modelling the future climate has become essential. As climate system is too complex for the human brain to grasp with simple insight, numerous computer applications are available.[40]

III. GENERAL CIRCULATION MODEL(GCMS)

A General Circulation Model, or GCM is a high resolution, grid-based computer simulation of the Earth's atmosphere. The simulation is designed such that it considers various factors that affects the Earth's climate, such as the composition of gases and particulate matter in the atmosphere, heat transfer, fluid dynamics, solar radiations, the interaction between lithosphere, aerosphere and hydrosphere, as well as external factors known as forcings, such as changes in greenhouse gas concentrations, alterations to ocean currents, volcanic eruptions, solar radiation, and other human activities. The resolution in GCM refers to the spatial and temporal discretization used to represent the Earth's atmosphere and other components of the climate system, spatial resolution refers to the size of grid cells, temporal resolution is the time step used for model advancement. Higher spatial and temporal resolutions lead to better depictions of weather and climate phenomena but also requires more computational resources. Enhanced spatial and temporal resolutions typically result in more accurate depictions of weather and climate phenomena, but necessitate increased computational resources and can incur higher computational costs.
GCM uses mathematical equations to model the fundamental physical processes that govern the Earth's climate, and subsequently simulate how the various attributes interrelate with each other over time. These equations are used to study the Earth's climate under different conditions and to predict how the climate may change in the future, helping policymakers to develop strategies for mitigating and adapting to the impacts of climate change.
General Circulation Models (GCMs) consist of two main types of models: Atmospheric Models, which simulate changes in the atmosphere, and Ocean Models, which simulate changes in the ocean. These two types of models are connected to create a complete climate model. The most used models for making predictions of future climate are the Coupled Atmosphere-Ocean General Circulation Models (AOGCMs).[20]
The Canadian Earth System Model version 2 (CanESM2) is a widely used climate model developed by the Canadian Centre for Climate Modelling and Analysis, the ensemble includes the 25 basic predictor variables as well as total precipitation amount at the daily scale. It has a spatial resolution of 2.8°x 2.8° [12]
The Norwegian Earth System Model (NorESM) is a widely used climate model, the NorESM resolution typically ranges from around 1 to 2 degrees in the atmosphere, which means that the Earth's surface is divided into grid cells of size approximately 100-200 kilometers on each side. [6]

3.1. Issues With GCM

While GCMs are useful tools for predicting future climate scenarios, GCMs have several limitations and issues that needs to be addressed. Some of the issues with GCMs include uncertainties in forcings, model tuning, and lack of consensus among different models in their predictions of future climate scenarios.

However, one of the major challenges associated with GCMs is that they are computationally intensive and require significant computing resources to run because they are comprehensive models that simulate the interactions between the atmosphere, ocean, land surface, and other components of the Earth system on a global scale. These models involve complex numerical calculations and require high-performance computing facilities to handle the massive amount of data involved. The computational demands of GCMs increases with the complexity and resolution of the model, which can limit the scope and scale of climate research that can be conducted using these models.

IV. EMULATORS (SURROGATES)

GCMs are comprehensive models that simulate the interactions between the atmosphere, ocean, land surface, and other components of the Earth system on a global scale, they are all encompassing model that aims to capture the dynamics of the entire Earth's climate system. Because of this, GCMs are computationally expensive and not ideal in scenarios where only specific aspects of the Earth's climate system is to be studied. For this purpose, Climate simulators can be used, they are a software tools that simulates

processes, phenomena, or scenarios related to the Earth's climate system in contrast to GCMs which are a type of numerical model used to simulate the global atmospheric circulation patterns.

Climate simulator is a computational model that employs mathematical equations to simulate various processes of the Earth's system. They are built upon intricate set of equations that encapsulates the fundamental physical laws governing climate processes, such as energy and mass conservation, heat transfer, radiative transfer, etc. They are commonly used to simulate the behavior of climate systems under the effect of some kind of forcing, like change in greenhouse concentration, solar radiations, etc.

PLASIM (Planet Simulator) is a widely used global climate simulator, typically used to investigate the dynamics of long-term climate, encompassing factors such as temperature fluctuations, precipitation patterns, atmospheric circulation, and the response of the climate system to external drivers such as changes in greenhouse gas concentrations or solar radiation.[19]

However, due to the modelling of mathematical equation, simulators are computationally very heavy, this can be solved using climate emulators or surrogates. Climate emulator is a simplified statistical or machine learning model that approximates the behavior of a climate simulator. Emulators are often used as a computationally efficient alternative to simulators, as they can provide similar results with lower computational costs. Emulators tries to mimic the simulator output with as little loss as possible. There is trade-off between interpretability and computational efficiency, but the cost of running the emulator is much lesser than its limitation in interpretability. Furthermore, an area of research known as 'interpretability theory' is introduced to explain how machine learning model makes predictions.

The use of surrogates or emulators allows for the examination of intricate systems or processes, experimentation with hypotheses, performance optimization, and prediction making. These tools prove particularly valuable when analyzing systems with a high degree of complexity, uncertainty, or variability.

Therefore, the utilization of surrogates or emulators empowers researchers and engineers to obtain valuable insights into intricate systems and processes, enabling them to arrive at superior decisions and devise more efficient and effective solutions. Some other issues relevant in the discussion of the characteristics of a deep learning-based climate emulator are as follows.

4.1 Physics-Informed Neural Networks (PINNs):

Physics Informed Neural Networks (PINN) are neural networks that applies scientific principles and laws of physics to solve problems that involve Partial Differential Equations (PDEs). PINNs incorporate the governing PDEs as constraints within the neural network architecture, enabling them to learn the underlying physics and make accurate predictions about the system being modeled. This function penalizes deviations from physical laws or principles, thereby constraining the network to conform to known physical behavior. The incorporation of physical constraints into the loss function allows the neural network to learn from data while adhering to fundamental physical laws and principles, resulting in more accurate and reliable predictions.

Additionally, PINNs can learn from observational data to improve the accuracy of the emulated climate models. This combination of physical constraints and data-driven learning makes PINNs a powerful tool for climate emulation, with potential applications in climate forecasting and policy-making. However, the application of PINNs in climate modelling was only limited to deterministic models but Bjorn Lutjens et al. that combines polynomial chaos expansion (PCE), a classic technique for uncertainty propagation, with PINNs. The PCE-PINNs learn a fast surrogate model that is useful for uncertainty propagation of known parameter uncertainties, which is demonstrated by the effectiveness of ocean modelling. [14,27]

4.2 Interpretability Theory:

Numerous deep learning topologies have emerged as promising climate emulation techniques, nonetheless, these models are still thought of as black boxes in the field of climate science, given the difficult interpretation of their inner functioning of climate models.[17]

Interpretability theory is an area of research in machine learning that focuses on developing methods to understand and explain how machine learning models make predictions. In the context of climate emulation, interpretability theory can be used to gain insights into how Physics-Informed Neural Networks (PINNs) are learning to approximate the behavior of more complex climate models. By understanding how the PINN model is making predictions, researchers can identify which aspects of the climate system are being accurately represented and which ones are not. This can help researchers improve the model by identifying areas that require further refinement or additional data.

V. TYPES OF EMULATORS

The different types of surrogates employed over climate data can be classified as Statistical and Deep learning**.**

5.1 Statistical Surrogates

Statistical surrogates are statistical models or algorithms that imitate the behavior of intricate systems or processes. Unlike traditional surrogates or emulators, which are often based on physical or mathematical principles, statistical surrogates rely solely on statistical methods and data-driven techniques. Statistical surrogates can be used to analyze and model complex systems that are difficult to understand or describe mathematically, such as environmental systems. They are also useful when there is a lack of fundamental understanding of the underlying physical principles of the system. The purpose of these models is to mimic the input-output relationship of the system being studied, based on empirical data.

The working of statistical surrogates for GCMs involves training a simplified statistical model to approximate the output of the full GCM. This training is typically done using data generated by the GCM or other sources. The statistical model is designed to capture the key relationships and patterns in the data, allowing it to make accurate predictions of the GCM output for new scenarios or inputs. Once training is done, it can be used to quickly generate predictions for a wide range of scenarios, without the need to run the full GCM. This can significantly reduce the computational time and resources required to explore different climate scenarios and study their potential impacts.

To create a statistical surrogate for a GCM, the first step is to identify a set of input variables that are likely to have a significant influence on the output of the original model. These input variables may include various factors such as atmospheric CO_2 levels, ocean temperature, solar radiation, and others. The surrogate is then trained using a subset of the output data from the original model, which is typically obtained by running simulations for a range of input values. The training process involves finding a mathematical function that can map the input variables to the output variable with a high degree of accuracy.

Once the surrogate is trained, it can be used to predict the output variable for new input scenarios that were not included in the training set.

However, it is important to note that surrogates are typically less accurate than GCMs, and may not capture all the complexity of the climate system. To ensure better results any uncertainty in the predictions should be properly accounted for, and can be used in conjunction with GCMs and other tools to provide a more complete and accurate simulation of the climate system.

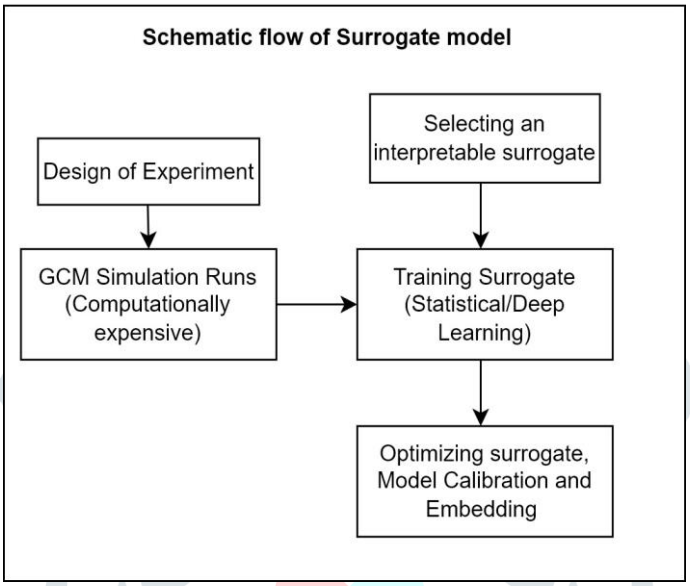

Fig. 1. Surrogate Model workflow

Model	Description	Features
MAGICC	(Model for the Assessment of Greenhouse-gas Induced Climate Change) A climate model used to simulate Earth's climate response induced by the greenhouse gas.	1. Incorporates representations of the radiative forcing of all major greenhouse gases 2. Open Sourced
FAIR	(Finite amplitude impulse response) It is a simple emissions-based climate model, helps in calculating change in global temperature resulting from a sudden increase in atmospheric greenhouse gas concentrations.	1. Can model complex interactions between the atmosphere, ocean, land and ice 2. Can be used to study climate variability and change over long-time scales
MESMERM	(Modular Earth System Model Emulator with Spatially Resolved Output) It is a specific type of modular earth system model emulator that can provide high-resolution, spatially resolved output.	1. Spatially resolved output allows for more detailed analysis 2. Allows for easy modification and customization of different model components
Hector	It is a simple climate model (SCM, also known as a reduced-complexity climate model), a class of models that are extremely versatile with a wide range of applications	1. Computationally efficient 2. Can easily be coupled to other models and used to design scenarios, emulate more complex climate models, and conduct uncertainty analyses.

Table. 1. Popular Statistical Climate Emulators

5.2 Deep Learning Surrogates

Deep learning surrogates refer to artificial neural networks that are used to model and approximate the behavior of a complex system or process. These models are designed to learn complex patterns and non-linear relationships in large datasets and to make predictions based on the learned information. They typically consist of multiple layers of artificial neurons, which are connected by weights that are adjusted during the training process. During training, the model is presented with input data and the corresponding output data, and the weights are adjusted to minimize the difference between the predicted output and the actual output. Once the deep learning surrogate has been trained, it can be used to predict the behavior of the system under different input conditions. For example, a deep learning surrogate for a weather forecasting system may be trained on historical weather data and used to predict future weather patterns.[44]

VI. GAUSSIAN PROCESS REGRESSION

Gaussian Processes (GPs) are extensively employed in Probabilistic modeling applications to replicate computationally expensive models like GCMs, it also minimizes the computational requirements for uncertainty quantification tasks such as uncertainty propagation, sensitivity analysis, and calibration.

The reason behind such widespread popularity of GP is due to their flexibility, ability to account for the uncertainty in the predictions, and mathematical simplicity for various crucial properties such as likelihood function, predictive distribution, and their derivatives.

6.1 Working of GPR

Gaussian Process (GP) regression is data driven machine learning technique used for making predictions. It models the relationship between inputs and outputs as distribution (rather than a fixed function), using a probabilistic approach. To understand this process better, we start with the concept of function. A function is a simple rule that maps inputs and outputs, for instance, the function ($f(x) = x^2$) maps the input x to its squared value. However, in GP regression, instead of assuming a fixed function f(x), we assume that the relationship is governed by a probability distribution. This distribution is known as a Gaussian process. A gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution. It is a way of describing how different data points are related to each other, this distribution is used to make predictions about new data points based on their similarity to existing data. The first step in Gaussian process regression is to define a prior distribution over the possible functions f(x). We assume that f(x) follows a Gaussian process with a mean function m(x) and a covariance function k(x,x'):

$$f(x)\ GP(m(x),\ k(x,x')) \qquad (1)$$

where, the mean function m(x) tells us about the expected value of f(x) for any given x, the covariance function k (x, x') gives us the correlation between f(x) and f(x') and the covariance function defines the shape of the Gaussian process, and it determines how smooth or rough the function f(x) is likely to be.

Given this prior distribution over f(x), we then use the observed data to update our estimate of f(x) using Bayes' rule. The likelihood function is given by:

$$p(y|x,f(x)) = N(y|f(x),\sigma^2) \qquad (2)$$

where ($N(y|f(x),\sigma^2)$) is the normal (Gaussian) distribution with mean f(x) and variance (σ^2) . This likelihood function tells us how likely the observed data y is, given our estimate of f(x) and some noise level (σ^2).

Using Bayes' rule, we can compute the posterior distribution over f(x) given the observed data:

$$p(f(x)|x,y) = N(f(x)|\mu(x),\Sigma(x)) \qquad (3)$$

where $\mu(x) and \Sigma(x)$ are the mean and covariance functions of the posterior distribution, respectively. These can be computed using the prior mean and covariance functions, the likelihood function, and the observed data.

Finally, we can use the posterior distribution over f(x) to make predictions for new values of x. Given a new value of x*, we can compute the predictive distribution over y* using:

$$p\,(y*|x*,x,y) = N(y*|\mu*(x*),\sigma^2*(x*)) \quad (4)$$

where $(\mu*(x*))$ and $(\sigma^2*(x*))$ are the mean and variance of the predictive distribution, respectively. These can be computed using the posterior mean and covariance functions, the likelihood function, and the new value of x*. Therefore, it can be said that, GP regression is a way of estimating the relationship between two variables using a flexible probabilistic model. The model is defined by a prior distribution over possible functions, which is updated using Bayes' rule based on observed data. The resulting posterior distribution over functions is then used to make predictions for new values of the input variable.

6.2 GPR Kernels

By employing a covariance function (or Kernel), we can improve the accuracy of our function estimations using the available data. Therefore, by choosing a covariance function, a prior is chosen or an initial guess is made about the function.
Typical kernels include: constant; linear; radial basis function (RBF; or squared exponential); and Matern 3/2 and´ 5/2 which are only once and twice differentiable respectively. Kernels can be customized to capture any properties of the functions under consideration, such as non-uniformity or periodicity. Kernels play an important role in determining the functions generated by the GP, therefore, the combination of common kernels can provide better control and interpretability of resulting model.[16]

6.2.1 Radial Basis Function kernel

The Radial Basis Function kernel or Squared Exponential kernel are stationary kernels that remain unaffected by translations in the input space. Moreover, they are positive definite, fulfilling the Mercer condition and enabling their usage in kernel methods like Gaussian process regression. Amongst these, the radial basis function kernel stands as the most frequently utilized one. (Also called the quadratic exponential kernel, the squared exponential kernel or the Gaussian kernel):

$$k_{SE}(x,x') = \frac{\sigma^2 exp(-(x-x'))}{2l^2} \quad (5)$$

This kernel is infinitely differentiable, which implies that GPs with this kernel as covariance function have mean square derivatives of all orders, and are thus very smooth.

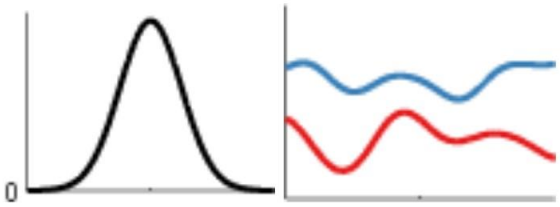

Fig. 2. Radial Basis Function kernel

6.2.2 Rational Quadratic Kernel

The Rational Quadratic kernel can be seen as a scale mixture (an infinite sum) of RBF kernels with different characteristic length-scales. It is parameterized by a length-scale parameter l¿0 and a scale mixture parameter σ ¿ 0 Only the isotropic variant where is a scalar is supported now. The kernel is given by:

$$k_{RQ}(x,x') = \sigma^2(\frac{1+(x-x')^2}{2\sigma l^2})^{-\sigma} \quad (6)$$

The parameter σ determines the relative weighting of largescale and small-scale variations.

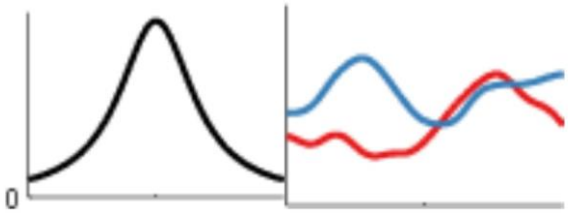

Fig. 3. Rational Quadratic Kernel

6.2.3 Matern kernel

The Matern kernel, which is a more inclusive version of the RBF kernel, is considered to be a stationary kernel. The kernel can be adjusted to vary the smoothness of the output function by modifying an additional parameter υ. This parameter is determined by a length-scale parameter l¿0, which can be either a scalar (representing the isotropic version of the kernel) or a vector that matches the number of dimensions of the inputs x (representing the anisotropic variant of the kernel). The mathematical expression for the kernel is as follows:

$$k_{RQ}(x_i,x_j) = \frac{1}{\Gamma*2^{\upsilon-1}}(\frac{\sqrt{2\upsilon}}{1}d(x_i,x_j))^{\upsilon}(k_{\upsilon}(\frac{\sqrt{2\upsilon}}{1}d(x_i,x_j))^{\upsilon} \quad (7)$$

where $d(x_i,x_j)$is the Euclidean distance, k_{υ} is a modified Bessel function and Γ is the gamma function. As $\upsilon \to \infty$, the

Matern kernel converges to the RBF kernel. When´ υ=1/2, the Matern kernel becomes identical to the absolute exponential´ kernel

6.2.4 Composite Kernels

A composite kernel is simply a combination of two or more individual kernels, which allows for more flexible modeling of the covariance structure.

The composite kernel function can be defined as the sum, product or any linear combination of the individual kernels. For example, a common composite kernel function used in GPR is the sum of the radial basis function (RBF) kernel and the Matern kernel:

$$k(x, x') = k\_RBF(x,x') + k\_Matern(x,x') \quad (8)$$

The RBF kernel is a smooth kernel that assumes that nearby points have similar function values, while the 'Matern kernel' is a more flexible kernel that allows for varying levels of smoothness in the function. By combining these two kernels, the composite kernel can capture both the smooth and non-smooth components of the function.

Composite kernels provide a powerful tool for modeling complex covariance structures in GPR, and can help improve the accuracy of predictions.

VII. GAUSSIAN PROCESS REGRESSION FOR CLIMATE EMULATION

Gaussian processes (GPs) have gained popularity as a preferred option for model emulation because of their elementary formulation and reliable uncertainty estimations, particularly when the amount of training data is relatively limited, and for its ability of non-parametric interpolation (method of estimating values between known data points without assuming a specific function). They are well suited for climate data because they can effectively handle nonlinear relationships between variables and provide reliable estimates of their relationship.

As aforementioned, GPR is a distribution of continuous function, an infinite dimensional normal distribution, which means the distribution is characterized by an infinite dimensional mean function and a covariance operator that maps pairs of functions to real numbers, rather than a mean vector and covariance matrix.

Utilizing the normal distributions and tools of Bayesian inference enables us to employ mathematical techniques to estimate the form of a function given a set of data. This estimation process can be performed in an interpretable and practical manner. A Gaussian process (GP) is a way of making these estimations by using a mean function and a covariance function determines the degree of similarity or difference between different function values and serves as a model for the behavior of the function.

When fitting a GP, the hyper-parameters that need to be optimized are the kernel's smoothness and length-scale.

Although, GPR has numerous advantages, it comes with a few limitations, when applying a Gaussian process to a data set of size n exact inference has complexity $O(n^3)$ with storage demands of $O(n^2)$. This limits the Gaussian process models to approximately 1000 training points, which is not enough training points to learn complicated functions.[13]

To mitigate the time and space complexity, one of the approaches can be Quantum Gaussian process regression, which is proposed using LCU Hamiltonian simulation [11], which has a quadratic acceleration compared with classical counterpart.

Additionally, GPs can be parallelized effectively using GPUs, it can be done by computing covariance matrices (N^2), computing Cholesky factors of positive-definite matrices (N^3), solving triangular systems using back-substitution (N^2), and optimizing hyper-parameters using some kind of parallel global optimization or sampling algorithm.

Model	Advantages	Limitations	Suited for
Gaussian Process Regression (GPR)	1. Provides probabilistic output and uncertainty estimates. 2. Flexibility in handling non-linear and non-parametric models. 3. Can handle missing data well.	1. Computationally expensive for large datasets. O(N^3) 2. Requires careful selection of kernels. 3. May not perform well with high-dimensional datasets.	Small to medium-sized climate datasets, non-linear and non-parametric regression problems, and sparse sdatasets
Linear Regression	1. Simple and computationally efficient. 2. Easy to interpret model coefficients.	1. Assumes a linear relationship between the dependent and independent variables. 2. Sensitive to outliers.	Linear problems with a small number of features.
Ridge Regression	1. Handles multicollinearity well. 2. Regularization reduces overfitting.	1. Requires tuning of regularization parameter. 2. Assumes a linear relationship between the dependent and independent variables.	Linear problems with multicollinearity and overfitting issues.
Lasso Regression	1. Handles multicollinearity well. 2. Performs feature selection.	1. Requires tuning of regularization parameter. 2. Assumes a linear relationship between the dependent and independent variables.	Linear problems with a small number of features and overfitting issues.
Support Vector Regression (SVR)	1. Performs well on high-dimensional datasets. 2. Handles non-linear relationships well	1. Requires careful selection of kernel functions. 2. Can be sensitive to the choice of hyperparameters.	Non-linear problems with a large number of features and high-dimensional datasets.

Table.2. Characteristics of Regression Models

VIII. FRAMEWORKS

8.1 TensorFlow Probability

TensorFlow Probability, is a Python library that leverages TensorFlow and facilitates the integration of deep learning and probabilistic models on contemporary hardware such as GPUs and TPUs. It supports a wide range of probabilistic models, including Gaussian process regression. With TFP, users can easily construct and train Gaussian process regression models, which can be used for tasks such as modeling non-linear dependencies between variables, making predictions, and uncertainty estimation. The library provides a range of useful tools and features for working with Gaussian processes, including kernels, likelihood functions, and various optimization techniques. The *tfd.GaussianProcess(kernel=kernel,index points=Xtrain [...np.newaxis]*,observation noise variance=noise std**2), is used to call the Gaussian process regression function. The 'kernel' argument is used to specify the covariance that describes the relationship between pairs of input points. The most common choice for functions that are smooth but not infinitely differentiable is the Matern 1/2 kernel. The index points argument specifies the input points that the Gaussian Process prior applies to. Xtrain is used as the input points for the prior, since we want to learn a function that interpolates the training data. Finally, observation noise variance specifies the variance of the noise in the training data. We then compute the posterior distribution over functions given the training data and the prior, and use it to make predictions on a set of test points. The posterior () method computes the posterior distribution over functions given the training data and the prior. We pass in Xtest as the input points for the posterior, since we want to make predictions on these points. We then use the mean () and stddev() methods to compute the mean and standard deviation of the predicted function at each test point. Finally, we plot the true function, the training data, and the posterior mean of the predicted function with 2 standard deviations shaded around it.

8.1.1 Features

Some key features of TensorFlow Probability's Gaussian Process Regression:

1.Flexibility in choice of kernel function: TFP offers a variety of kernel functions to choose from, including the Matern family of kernels, the Squared Exponential kernel, and the Rational Quadratic kernel. Users can also define their own custom kernel functions if necessary.

2.Scalability: TFP's Gaussian Process Regression is designed to be scalable to large datasets. The 'Gaussian Process' class uses efficient algorithms for computing the posterior distribution over functions, and can handle high-dimensional input spaces.

3.Integration with TensorFlow: TFP's Gaussian Process Regression is built on top of TensorFlow, which means that it can be easily integrated with other TensorFlow based models or pipelines.

8.2 GPFlow

GPFlow is a Python library that provides tools and functionality for Gaussian process regression, similar to TensorFlow Probability. GPFlow leverages TensorFlow as its computational engine, allowing for efficient and scalable Gaussian process regression. The GPflow interface and Python architecture are heavily influenced by GPy.The library includes a variety of kernel functions,

likelihood functions, and optimization methods for constructing and training Gaussian process regression models. Additionally, GPFlow supports a range of advanced features such as automatic relevance determination (ARD), inducing point methods, and Bayesian optimization. Overall, GPFlow is a powerful and flexible library for Gaussian process regression that can be particularly useful for machine learning and data analysis tasks.

8.2.1 Features

Key features of the library:

1.Variational inference is employed as the main method of approximation to address the dual difficulties posed by non-conjugacy and scale

2.Code that is relatively brief and utilizes automatic differentiation to alleviate the effort required for implementing gradients.

3.The capability to utilize GPU hardware for rapid computing, along with the principles of open-source software.

To create a model, first, the kernel is to be defined *kernel = gpflow.kernels.*(), creates an instance of any of the specified kernel. GPflow implements the following kernels: Matern12, Matern32, Matern52, RBF (radial basis function), Constant, Linear, Cosine, and Periodic. After the kernel is defined *model = gpflow.models.GPR*(*data* = (*X*,*Y*),*kernel = kernel*,*noise variance* = 0.01) we are create an instance of the Gaussian process regression (GPR) model. The data argument is a tuple containing the input data X (a NumPy array of shape (n, d), where n is the number of data points and d is the number of input dimensions) and the output data Y (a NumPy array of shape (n, 1)). The kernel argument is the covariance function used in the GPR model, which we previously defined. The *noise _variance* argument is the variance of the Gaussian noise added to the output values to model measurement noise or other sources of error in the data. This is typically estimated or set using domain knowledge.[15]

IX. Related Work

ESEm v1.0.0 is an open, scalable Earth System Emulator library, introduced by Duncan Watson-Parris [41], provides GP fitting, using the open-source GPFlow[15], a machine learning library that supports the use of Graphical Processing Units(GPUs), which as aforementioned, enhances the training time of GPs. In this paper, ESEm generates emulated response using GP emulation of aerosol optical depth (AAOD) using a 'Bias + linear' kernel for a specific set of parameters, and does an excellent job in reproducing the spatial structure of AAOD and demonstrates reduced error.

Additionally, the use of GPR in climate emulation is discussed [23], for research modelling and predicting precipitation in the Hindu Kush Karakoram Himalayan region. GPs are used because it provides a flexible and easily understandable method for analyzing and forecasting precipitation patterns in this region, because it inherently provides a method for quantifying uncertainty in a principled manner. Also, they allow us to choose from a wide range of priors, based on kernels that helps us in choosing a prior that is more likely to make accurate predictions. By carefully defining the covariance kernel, we can incorporate specific characteristics such as periodic patterns, smoothness, or variation across space into our predictions.

GPR is also utilized as a baseline emulator in ClimateBench[42], which is a bench-marking platform for climate and weather models. Gaussian process regression is used to develop an emulator for a computationally expensive climate model. The emulator is then used to generate a large ensemble of simulations, which enables the exploration of the model's parameter space in a more efficient and systematic manner.

X. CONCLUSION

As climate change continues to impact the Earth's system, there is an increasing need for instruments and knowledge to develop adaptation plans, vulnerability assessments, and resilience strategies. Fully coupled GCMs, or Earth System models, are the benchmark for climate modelling but entail substantial computational costs. To address this, surrogates or emulators have been introduced, with statistical surrogates being used for many years and deep learning models becoming more prominent due to the availability of huge volumes of climate data. However, deep learning surrogates lack interpretability, can extrapolate values beyond the training points, and cannot capture model uncertainty. Gaussian process regression (GPR) emerges as a viable candidate as it uses a probabilistic approach to model the uncertainty estimates and can be improved with the use of composite kernels. The implementation of GPR is facilitated by the availability of new libraries, such as GPFlow, built on TensorFlow. Advancements in machine learning will enable higher accuracy and resolution in emulation techniques. GPR provides a promising approach for improving our understanding of climate systems and developing effective climate adaptation and mitigation strategies.

REFERENCES

[1] Ajagekar, A., & You, F. (2022). Quantum computing and quantum artificial intelligence for renewable and sustainable energy: A emerging prospect towards climate neutrality. Renewable and Sustainable Energy Reviews, 165, 112493. https://doi.org/10.1016/J.RSER.2022.112493

[2] Asch, R. G., Pilcher, D. J., Rivero-Calle, S., & M. Holding, J. (2016). Demystifying Models: Answers to Ten Common Questions That Ecologists Have About Earth System Models. Limnology and Oceanography Bulletin, 25(3), 65–70. https://doi.org/10.1002/lob.10113

[3] Ayala, A., Drazic, C., Hutchinson, B., Kravitz, B., & Tebaldi, C. (2021). Loosely Conditioned Emulation of Global Climate Models with Generative Adversarial Networks. http://arxiv.org/abs/2105.06386

[4] Bagnell, D., Narayanan, V., Koval, M., & Parashar, P. (n.d.). Gaussian Processes (Vol. 12). http://www.gaussianprocess.org/gpml/

[5] Bastos, L. S., & O'hagan, A. (2009a). Diagnostics for gaussian process emulators. Technometrics, 51(4), 425–438. https://doi.org/10.1198/TECH.2009.08019

[6] Bentsen, M., Bethke, I., Debernard, J. B., Iversen, T., Kirkevag, A., ˚ Seland, Ø., Drange, H., Roelandt, C., Seierstad, I. A., Hoose, C., & Kristjansson, J. E. (2013). The Norwegian Earth System Model, ´ NorESM1-M – Part 1: Description and basic evaluation of the physical climate. Geoscientific Model Development, 6(3), 687–720. https://doi.org/10.5194/gmd-6-687-2013

[7] Beusch, L., Gudmundsson, L., & Seneviratne, S. I. (2020). Emulating Earth system model temperatures with MESMER: From global mean temperature trajectories to grid-point-level realizations on land. Earth System Dynamics, 11(1), 139–159. https://doi.org/10.5194/esd-11-1392020

[8] Beusch, L., Nicholls, Z., Gudmundsson, L., Hauser, M., Meinshausen, M., & Seneviratne, S. I. (2022). From emission scenarios to spatially resolved projections with a chain of computationally efficient emulators: coupling of MAGICC (v7.5.1) and MESMER (v0.8.3). Geoscientific Model Development, 15(5), 2085–2103. https://doi.org/10.5194/gmd-152085-2022

[9] Bohning-Gaese,¨ K., & Lemoine, N. (2004). Importance of Climate Change for the Ranges, Communities and Conservation of Birds. Advances in Ecological Research, 35, 211–236. https://doi.org/10.1016/S0065-2504(04)35010-5

[10] Camps-Valls, G., Reichstein, M., Zhu, X., & Tuia, D. (2020). ADVANCING DEEP LEARNING FOR EARTH SCIENCES: FROM HYBRID MODELING TO INTERPRETABILITY. IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, 3979–3982. https://doi.org/10.1109/IGARSS39084.2020.9323558

[11] Chen, M., Lin, S., Guo, G.-D., & Li, J. (2021). Quantum Gaussian process regression.

[12] Chylek, P., Li, J., Dubey, M. K., Wang, M., & Lesins, G. (2011). Canadian Earth System Model CanESM2 Observed and model simulated 20th century Arctic temperature variability: Canadian Earth System Model CanESM2 Canadian Earth System Model CanESM2. Atmos. Chem. Phys. Discuss, 11, 22893–22907. https://doi.org/10.5194/acpd11-22893-2011

[13] Coleman-Smith, C. E., Muller, B., Robert, S., Steen, W., Bass, A., ¨ Chandrasekharan, S., & Kruse, M. (n.d.). Using Gaussian Processes for the Calibration and Exploration of Complex Computer Models.

[14] Cuomo, S., Di Cola, V. S., Giampaolo, F., Rozza, G., Raissi, M., & Piccialli, F. (2022). Scientific Machine Learning Through Physics–Informed Neural Networks: Where we are and What's Next. Journal of Scientific Computing, 92(3). https://doi.org/10.1007/s10915-022-01939-z

[15] De, A. G., Matthews, G., Nickson, T., Fujii, K., Boukouvalas, A., Leon-´ Villagra, P., Ghahramani, Z., & Hensman, J. (2017). GPflow: A Gaussian´ Process Library using TensorFlow Mark van der Wilk. In Journal of Machine Learning Research (Vol. 18). http://jmlr.org/papers/v18/16537.html.

[16] Duvenaud, D. (n.d.). The Kernel Cookbook: Advice on Covariance functions. https://www.cs.toronto.edu/ duvenaud/cookbook/

[17] Gonzalez-Abad, J., B´ a No-Medina, J., & Manuel Guti˜ errez, J. (n.d.). ´ TESTING INTERPRETABILITY TECHNIQUES FOR DEEP STATISTICAL CLIMATE DOWNSCALING. https://www.ecmwf.int/

[18] Hensman, J., Fusi, N. O., & Lawrence, N. D. (n.d.). Gaussian Processes for Big Data.

[19] Holden, P. B., Edwards, N. R., Garthwaite, P. H., Fraedrich, K., Lunkeit, F., Kirk, E., Labriet, M., Kanudia, A., & Babonneau, F. (2014). PLASIM-ENTSem v1.0: A spatiooral emulator of future climate change for impacts assessment. Geoscientific Model Development, 7(1), 433–451. https://doi.org/10.5194/gmd-7-433-2014

[20] Karpatne, A., Atluri, G., Faghmous, J. H., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N., & Kumar, V. (2017). Theoryguided data science: A new paradigm for scientific discovery from data. IEEE Transactions on Knowledge and Data Engineering, 29(10), 2318–2331. https://doi.org/10.1109/TKDE.2017.2720168

[21] Kotamarthi, R., Hayhoe, K., Mearns, L., Wuebbles, D., Jacobs, J., & Jurado, J. (2021). Downscaling Techniques for HighResolution Climate Projections. Cambridge University Press. https://doi.org/10.1017/9781108601269

[22] Kumar, N., Tischbein, B., Kusche, J., Laux, P., Beg, M. K., & Bogardi, J. J. (2017). Impact of climate change on water resources of upper Kharun catchment in Chhattisgarh, India. Journal of Hydrology: Regional Studies, 13, 189–207. https://doi.org/10.1016/j.ejrh.2017.07.008

[23] Lalchand, V., Tazi, K., Cheema, T. M., Turner, R. E., & Hosking, S. (2022). Kernel Learning for Explainable Climate Science. http://arxiv.org/abs/2209.04947

[24] Lenton, T. (2016). Earth System Science: A Very Short Introduction. Oxford University Press. https://doi.org/10.1093/actrade/9780198718871.001.0001

[25] Liu, H., Ong, Y. S., Shen, X., & Cai, J. (2020). When Gaussian Process Meets Big Data: A Review of Scalable GPs. IEEE Transactions on Neural Networks and Learning Systems, 31(11), 4405–4423. https://doi.org/10.1109/TNNLS.2019.2957109

[26] Lu, D., & Ricciuto, D. (2019). Efficient surrogate modeling methods for large-scale Earth system models based on machine-learning techniques. Geoscientific Model Development, 12(5), 1791–1807. https://doi.org/10.5194/gmd-12-1791-2019

[27] Lutjens, B., Crawford, C. H., Veillette, M., & Newman, D. (2021). PCE-¨ PINNs: Physics-Informed Neural Networks for Uncertainty Propagation in Ocean Modeling.

[28] Maslin, M. (n.d.) Climate: a very short introduction.

[29] Matthews, A. G. de G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., Leon-Villagr´ a, P., Ghahramani, Z., & Hensman, ´ J. (2016). GPflow: A Gaussian process library using TensorFlow.http://arxiv.org/abs/1610.08733

[30] Meinshausen, M., Raper, S. C. B., & Wigley, T. M. L. (2008). MAGICC 6.0 Emulating IPCC AR4 atmosphere-ocean and carbon cycle models for projecting global-mean, hemispheric and land/ocean temperatures: MAGICC 6.0 MAGICC 6.0. In Chem. Phys. Discuss (Vol. 8). www.atmos-chem-phys-discuss.net/8/6153/2008/

[31] Morrison, P. (2004). The Discovery of Global Warming the Discovery of Global Warming Spencer R. Weart Harvard U. Press, Cambridge, MA, 2003. 24.95 (228 pp.). ISBN 0-674-01157-0. Physics Today, 57(6), 60–61. https://doi.org/10.1063/1.1784277

[32] Radhakrishnan Kristopher Rand Charles Stern Julius Busecke, A., Abernathey, R., Balaji, V., Kershaw, P., Privette, A., Stephens, A., Naik, N., Nikonov, S., Vahlenkamp, H., Blanusa, M., Banihirwe,A., Blanton, C., Perry Boh, N., Evans, B., Smith, R., Evans, R., Flamig, Z., Gergel, D., Jackson, T., ... Sayibou, Z. (n.d.). CMIP6 Amazon-hosted Data Informational Session CMIP6 Data Informational session, Oct 13th, 2021 Community driven effort 4 Motivation: Nature. https://science.thewire.in/environment/what-the-ipccreport-forecasts-for-india-development-future/

[33] Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physicsinformed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378, 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

[34] Rajan, S. (2014). STATISTICAL DOWNSCALING OF GCM OUTPUT, HYDROLOGICAL SIMULATION AND GENERATION OF FUTURE SCENARIO USING VARIABLE INFILTRATION CAPACITY (VIC) MODEL FOR THE GANGA BASIN, INDIA.

[35] Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019a). Deep learning and process understanding for data-driven Earth system science. Nature, 566(7743), 195–204. https://doi.org/10.1038/s41586-019-0912-1

[36] Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019b). Deep learning and process understanding for data-driven Earth system science. Nature, 566(7743), 195–204. https://doi.org/10.1038/s41586-019-0912-1

[37] Satria Palar, P., Rizki Zuhal, L., & Shimoyama, K. (2020a). Gaussian Process Surrogate Model with Composite Kernel Learning for Engineering Design. AIAA Journal, 58(4), 1864–1880.https://doi.org/10.2514/1.J058807

[39] Wang, J. (2020). An Intuitive Tutorial to Gaussian Processes Regression. http://arxiv.org/abs/2009.10862

[40] Watson-Parris, D. (2020). Machine learning for weather and climate are worlds apart. https://doi.org/10.1098/rsta.2020.0098

[41] Watson-Parris, D., Williams, A., Deaconu, L., & Stier, P. (n.d.). Model calibration using ESEm v1.0.0-an open, scalable Earth System Emulator.

[42] Watson-Parris, D., Rao, Y., Olivie, D., Seland, Ø., Nowack, P., Camps-´ Valls, G., Stier, P., Bouabid, S., Dewey, M., Fons, E., Gonzalez, J., Harder, P., Jeggle, K., Lenhardt, J., Manshausen, P., Novitasari, M., Ricard, L., & Roesch, C. (2022). ETH Library ClimateBench v1.0: A Benchmark for Data-Driven Climate Projections Rights / license: Creative Commons Attribution 4.0 International Funding acknowledgement: 860100-innovative MachIne leaRning to constrain Aerosol-cloud Climate Impacts (EC) ClimateBench v1.0: A Benchmark for Data-Driven Climate Projections Citation. Journal of Advances in Modeling Earth Systems, 14(10). https://doi.org/10.3929/ethz-b-000578488

[43] Weber, T., Corotan, A., Hutchinson, B., Kravitz, B., & Link, R. (2020). Technical note: Deep learning for creating surrogate models of precipitation in Earth system models. Atmospheric Chemistry and Physics, 20(4), 2303–2317. https://doi.org/10.5194/acp-20-2303-2020

[44] Xu, W., Luo, X., Ren, Y., Park, J. H., Yoo, S., & Nadiga, B. T. (2021). Feature Importance in a Deep Learning Climate Emulator.